HAL
open science

Méthodes proximales pour la résolution de problèmes inverses : application à la tomographie par émission de positrons

Nelly Pustelnik

▶ **To cite this version:**

Nelly Pustelnik. Méthodes proximales pour la résolution de problèmes inverses : application à la tomographie par émission de positrons. Autre [cs.OH]. Université Paris-Est, 2010. Français. NNT : 2010PEST1037 . tel-00559126v5

HAL Id: tel-00559126

https://pastel.hal.science/tel-00559126v5

Submitted on 14 Dec 2012

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 PARIS-EST

THÈSE

soutenue le 13/12/2010 pour obtenir
le grade de **Docteur en Sciences** de l'Université Paris-Est

Spécialité : Traitement du Signal et de l'Image

par

Nelly PUSTELNIK

Méthodes proximales pour la résolution de problèmes inverses. Application à la Tomographie par Émission de Positrons.

Composition de la commission d'examen :

Président :	Mohamed NAJIM
Rapporteurs :	Laure BLANC-FÉRAUD
	Jalal FADILI
Examinateurs :	Caroline CHAUX
	Patrick Louis COMBETTES
	Claude COMTAT
	Michel DEFRISE
Directeur de thèse :	Jean-Christophe PESQUET

Remerciements

En premier lieu, je tiens à remercier tout particulièrement mon directeur de thèse Jean-Christophe Pesquet pour m'avoir donné l'opportunité de découvrir le monde passionant de la recherche. Je le remercie pour sa rigueur et son dynamisme qui m'ont permis de repousser un peu plus mes limites à chaque fois, ainsi que pour sa disponibilité et son soutient qui m'ont permis d'effectuer ma thèse dans de si bonnes conditions.

Je souhaite également témoigner toute ma gratitude à Caroline Chaux pour m'avoir guidée sur les chemins de la recherche, pour son soutient et son positivisme sans failles.

Merci à Laure Blanc-Féraud et Jalal Fadili pour avoir tous deux accepté spontannément de rapporter ma thèse et pour leurs remarques avisées. Merci également à mes examinateurs Mohamed Najim, Michel Defrise, Patrick Combettes et Claude Comtat. Un merci particulier à Claude qui m'a guidée, durant ces trois années, à travers les fondements de la TEP et ses subtilités.

Je remercie Patrice Hérault d'avoir toujours été disponible pour me dépétrer aussi bien de problèmes d'informatique que de soucis liés au RER. Je remercie Sylvie Cach dont l'efficacité facilite grandement les démarches administratives relatives à la thèse. Je remercie Line Fonfrède pour m'avoir toujours simplifié mes déplacements en conférences. Je remercie Laurent Duval pour m'avoir transmis sa gestion de la biblio. Merci à Florent Sureau pour son initiation à la TEP. Je remercie Hugues Talbot pour ses remarques enrichissantes. Je remercie également Luis Briceño et Patrick Combettes pour cette collaboration sur les problèmes multicomposantes qui fut très enrichissante.

Je remercie les membres de l'équipe signal et communication. Merci aux thésards (anciens ou actuels), en commencant par Lotfi et Mounir (pour m'avoir supportée comme co-bureau pendant ces 3 ans), Elena (pour ses corrections d'anglais), Ania (pour avoir partagé ces longues soirées passées à MLV), Jérome (pour ses conseils), Pascal, Florian, Mireille, Téodora et Wided.

Merci à mes amis et à la famille de Benoît pour leurs encouragements.

Enfin, un grand merci à ma famille, en commençant bien entendu par mes parents, pour m'avoir toujours soutenue et conseillée et pour tous les bons moments passés avec eux. Merci Papa pour tes remarques sur ma thèse malgré le fait que tu ne sois pas spécialiste du même type de « Poisson ». Merci à mes grand-parents rochelais pour leur tendresse et leur soutient. Enfin, un grand merci à Benoît pour sa patience, son écoute pendant ces trois ans et pour me permettre de m'évader chaque jour grâce à sa joie de vivre et à sa folie.

Table des matières

Résumé **9**

Abstract **11**

Glossaire **13**

1 Introduction **15**
1.1 Contexte de l'étude 15
1.2 Vue d'ensemble du travail effectué 18
1.2.1 État de l'art 18
1.2.2 Algorithmes imbriqués et extension quadratique 18
1.2.3 Algorithme proximal parallèle et régularisation hybride 18
1.2.4 Formulations à l'analyse et à la synthèse 19
1.2.5 Algorithmes proximaux multicomposantes 19
1.2.6 Reconstruction dynamique d'images TEP 19
1.3 Publications 20
1.3.1 Articles de revus acceptés ou publiés 20
1.3.2 Articles soumis 20
1.3.3 Articles de conférences invités 20
1.3.4 Autres articles de conférences 20
1.3.5 Autres présentations 21

2 Introduction aux problèmes inverses et à l'optimisation convexe **23**
2.1 Problèmes inverses 23
2.1.1 Quelques exemples 23
2.1.2 Problème mal posé, mal conditionné et régularisation 26
2.2 Résolution de problèmes inverses 27
2.2.1 Interprétation bayésienne 27
2.2.2 Terme d'attache aux données 28
2.2.3 Terme de régularisation 29
2.3 Optimisation convexe 33
2.3.1 Quelques rappels d'analyse convexe 34
2.3.2 Résultats d'analyse convexe 37
2.3.3 Algorithmes 38
2.3.4 Conclusion sur les algorithmes 43
2.4 Bruit de Poisson : quelle méthode adopter ? 44
2.4.1 Maximum de vraisemblance 46

2.4.2 Approche MAP : traiter le bruit comme s'il sagissait d'un bruit gaussien . . . 47
2.4.3 Approche MAP : transformées stabilisatrices de variance . . . 47
2.4.4 Approches proximales . . . 48
2.5 Conclusion . . . 52

3 Imbrication d'algorithmes proximaux et extension quadratique 55
3.1 Motivations . . . 55
3.2 Opérateur proximal . . . 56
3.2.1 Définition . . . 56
3.2.2 Quelques propriétés . . . 57
3.3 Algorithmes pour minimiser une somme de deux fonctions convexes . . . 65
3.3.1 Algorithme explicite-implicite (FB : forward-backward) . . . 65
3.3.2 Algorithme de Douglas-Rachford (DR) . . . 70
3.4 Opérateur proximal d'une somme de deux fonctions . . . 73
3.4.1 Algorithme explicite-implicite . . . 74
3.4.2 Algorithme de Douglas-Rachford . . . 75
3.4.3 Algorithme de type Dykstra . . . 77
3.5 Algorithmes imbriqués . . . 77
3.5.1 Première méthode proposée : DR[FB] . . . 78
3.5.2 Seconde méthode proposée : FB[DR] . . . 82
3.5.3 Troisième méthode proposée : FB[Dyk] . . . 84
3.6 Résolution de problèmes de restauration d'images . . . 85
3.6.1 Modélisation du problème . . . 86
3.6.2 Comparaison expérimentale des algorithmes imbriqués sur un exemple de restauration en présence de bruit gaussien . . . 88
3.6.3 Approximation inférieure du terme d'attache aux données pour la résolution d'une classe plus large de problèmes de restauration . . . 93
3.7 Conclusion . . . 106

4 Algorithme proximal parallèle et régularisation hybride pour la résolution de problèmes inverses en présence de bruit non-additif 109
4.1 Motivations . . . 109
4.2 Formulation du problème . . . 111
4.3 Formes explicites de quelques opérateurs proximaux . . . 112
4.3.1 Opérateurs proximaux de fonctions usuelles . . . 112
4.3.2 Opérateur proximal impliquant un opérateur linéaire . . . 113
4.3.3 Formes discrètes de la variation totale et opérateurs proximaux associés . . . 123
4.4 Algorithme proximal parallèle (PPXA) . . . 128
4.4.1 Présentation de l'algorithme . . . 128

4.4.2 Version accélérée en présence de trames ajustées 129
4.5 Application à la restauration d'image 133
4.5.1 Régularisation hybride 133
4.5.2 Résultats expérimentaux pour la résolution de problèmes inverses en présence de bruit de Poisson 135
4.6 Conclusion 142

5 Comparaison des approches trame à l'analyse et trame à la synthèse dans le cadre des algorithmes proximaux 149
5.1 Motivations 149
5.2 Formulation à l'analyse et formulation à la synthèse 151
5.2.1 Formulation à l'analyse (FA) 151
5.2.2 Formulation à la synthèse (FS) 152
5.2.3 Comparaison théorique FS versus FA 153
5.3 Algorithme explicite-implicite pour des trames non nécessairement ajustées 155
5.3.1 Algorithme et difficultés rencontrées 155
5.3.2 Calcul de l'opérateur proximal associé à $f_2 \circ F$ 157
5.3.3 Calcul de la constante de trame 160
5.3.4 Comparaison de FA et FS sur un exemple de restauration en présence de bruit gaussien 161
5.4 PPXA+ pour une classe de trames non ajustées 168
5.4.1 Un exemple particulier de trame non-ajustées 168
5.4.2 Résolution des deux formulations avec PPXA+ 169
5.4.3 Résultats expérimentaux 173
5.5 Conclusion 174

6 Algorithmes proximaux pour la résolution de problèmes multicomposantes 177
6.1 Motivations 177
6.2 Notations et rappels 178
6.3 Opérateurs proximaux multicomposantes 179
6.4 Algorithmes proximaux multicomposantes 184
6.4.1 Algorithme explicite-implicite 184
6.4.2 Algorithme de Douglas-Rachford 186
6.4.3 Algorithme proximal parallèle (PPXA) 187
6.4.4 Algorithme de type Dykstra 188
6.5 Applications à la décomposition et à la restauration d'images 190
6.5.1 Restauration d'images stéréoscopiques 190
6.5.2 Débruitage d'images multispectrales 195
6.5.3 Décomposition d'images 201
6.6 Conclusion 210

7 Algorithmes proximaux pour la reconstruction de données dynamiques en TEP 211
7.1 Introduction à la TEP 211
7.1.1 Traceur radioactif 211
7.1.2 Détection des coïcidences 212
7.1.3 Modèle de dégradation spatio-temporel 214
7.1.4 Principe compartimental dans le cas du [^{18}F]-FDG 214
7.2 Méthodes de reconstruction en TEP 218
7.3 Extension quadratique et algorithmes proximaux 221
7.3.1 Extension quadratique 223
7.3.2 Adaptation du FB[DR] pour la reconstruction TEP dynamique . . 225
7.3.3 Matériel et résultats expérimentaux sur données simulées 227
7.3.4 Conclusion 232
7.4 PPXA et régularisation hybride 232
7.4.1 Méthode 232
7.4.2 Matériel et résultats expérimentaux sur données simulées 236
7.4.3 Résultat expérimentaux sur données réelles 252
7.4.4 Conclusion 252

8 Conclusion et perspectives 255
8.1 Bilan 255
8.2 Quelques perspectives 257

Liste des figures 265

Liste des tableaux 267

Bibliographie 281

Résumé

L'objectif de cette thèse est de proposer des méthodes fiables, efficaces et rapides pour minimiser des critères convexes apparaissant dans la résolution de problèmes inverses en imagerie. Ainsi, nous nous intéresserons à des problèmes de restauration/reconstruction lorsque les données sont dégradées par un opérateur linéaire et un bruit qui peut être non additif.

La fiabilité de la méthode sera assurée par l'utilisation d'algorithmes proximaux dont la convergence est garantie lorsqu'il s'agit de minimiser des critères convexes. La quête d'efficacité impliquera le choix d'un critère adapté aux caractéristiques du bruit, à l'opérateur linéaire et au type d'image à reconstruire. En particulier, nous utiliserons des termes de régularisation basés sur la variation totale et/ou favorisant la parcimonie des coefficients du signal recherché dans une trame (*frame*). L'utilisation de trames nous amènera à considérer deux approches : une formulation du critère à l'analyse et une formulation du critère à la synthèse. De plus, nous étendrons les algorithmes proximaux et leurs preuves de convergence aux cas de problèmes inverses multicomposantes. La recherche de la rapidité de traitement se traduira par l'utilisation d'algorithmes proximaux parallélisables.

Les résultats théoriques obtenus seront illustrés sur différents types de problèmes inverses de grandes tailles comme la restauration d'images mais aussi la stéréoscopie, l'imagerie multispectrale, la décomposition en composantes de texture et de géométrie. Une application attirera plus particulièrement notre attention ; il s'agit de la reconstruction de l'activité dynamique en Tomographie par Emission de Positrons (TEP) qui constitue un problème inverse difficile mettant en jeu un opérateur de projection et un bruit de Poisson dégradant fortement les données observées. Pour optimiser la qualité de reconstruction, nous exploiterons les caractéristiques spatio-temporelles de l'activité dans les tissus.

Abstract

The objective of this work is to propose reliable, efficient and fast methods for minimizing convex criteria, that are found in inverse problems for imagery. We focus on restoration/reconstruction problems when data is degraded with both a linear operator and noise, where the latter is not assumed to be necessarily additive.

The methods reliability is ensured through the use of proximal algorithms, the convergence of which is guaranteed when a convex criterion is considered. Efficiency is sought through the choice of criteria adapted to the noise characteristics, the linear operators and the image specificities. Of particular interest are regularization terms based on total variation and/or sparsity of signal frame coefficients. As a consequence of the use of frames, two approaches are investigated, depending on whether the analysis or the synthesis formulation is chosen. Fast processing requirements lead us to consider proximal algorithms with a parallel structure.

Theoretical results are illustrated on several large size inverse problems arising in image restoration, stereoscopy, multi-spectral imagery and decomposition into texture and geometry components. We focus on a particular application, namely Positron Emission Tomography (PET), which is particularly difficult because of the presence of a projection operator combined with Poisson noise, leading to highly corrupted data. To optimize the quality of the reconstruction, we make use of the spatio-temporal characteristics of brain tissue activity.

Glossaire

Notations générales

$\mathcal{H}, \mathcal{G}$	: espaces de Hilbert réels séparables.
$L : \mathcal{H} \to \mathcal{G}$	: opérateur linéaire borné.
$L^* : \mathcal{G} \to \mathcal{H}$	: adjoint de l'opérateur linéaire borné $L : \mathcal{H} \to \mathcal{G}$.
$\Gamma_0(\mathcal{H})$	: ensemble des fonctions de $\mathcal{H}$ dans $]-\infty, +\infty]$, convexes, semi-continues inférieurement et propres (non égales à $+\infty$).
$\mathbb{R}^X$	: espace euclidien de dimension X.
$\operatorname{dom} f$	: domaine de la fonction $f : \mathcal{H} \to]-\infty, +\infty]$.
$\operatorname{int} S$	: intérieur de l'ensemble $S \subset \mathcal{H}$.
$\operatorname{ri} S$	: intérieur relatif de l'ensemble $S \subset \mathcal{H}$.
$\operatorname{sri} S$	: intérieur relatif fort de l'ensemble $S \subset \mathcal{H}$.
C	: ensemble convexe fermé non vide de $\mathcal{H}$.
P_C	: projecteur sur l'ensemble convexe fermé non vide $C \subset \mathcal{H}$.
$\operatorname{Argmin} f$	: ensemble des minimiseurs de la fonction f.
$\arg\min f$	: minimiseur de la fonction f en cas d'unicité.
$\operatorname{tr}(L)$	: trace de la matrice $L \in \mathbb{R}^{X \times Y}$.
$\lVert \cdot \rVert_{\mathrm{F}}$	: norme de Frobenius.
sign	: fonction signum.
Id	: opérateur linéaire identité.
$\lVert \cdot \rVert_p$	: norme ℓ_p.

Notations spécifiques

$y = (y_i)_{1 \leq i \leq N}$	: vecteur associé à une image composée de N pixels.
$\overline{y}$	: image originale (considérée comme image de référence).
$\widehat{y}$	: image restaurée/reconstruite.
A	: opérateur linéaire de dégradation (opérateur de convolution, de projection, ...).
$z = (z_i)_{1 \leq i \leq M}$	: vecteur d'observations composées de M éléments (pixels, bins,...).
$F \in \mathbb{R}^{K \times N}$	: opérateur d'analyse de trame.
$F^* \in \mathbb{R}^{N \times K}$	: opérateur de synthèse de trame.
ν	: coefficient de trame ajustée ($F^* \circ F = \nu \mathrm{Id}$).
$x = (x_k)_{1 \leq k \leq K}$	: vecteur des coefficients de trames.
Q	: taille du noyau de flou.
α	: paramètre du bruit (variance, paramètre d'échelle,...).

CHAPITRE 1

Introduction

« We believe that convergence properties are relevant to clinical medical imaging, since algorithm divergence could have unfortunate consequences. »

J. Fessler & A. Hero, 1995

1.1 Contexte de l'étude

La Tomographie par Emission de Positrons est une modalité d'imagerie médicale qui permet, suite à l'injection au patient d'une molécule marquée radioactivement, d'obtenir une cartographie spatiale d'un paramètre physiologique. Ce dernier se déduit de la mesure dans les tissus de la concentration radioactive de la molécule injectée. Il peut par exemple s'agir de molécules analogues au glucose qui permettent de détecter certaines masses cancéreuses par leur activité métabolique, ou encore de molécules ayant une affinité pour le système dopaminergique qui est impliqué dans les maladies psychiatriques (dépression, schizophrénie,. . .) et certaines maladies neurodégénératives (maladie de Parkinson,. . .).

Les données récupérées en sortie du tomographe ne permettent pas d'effectuer directement la cartographie multidimensionnelle du paramètre physiologique considéré. Retrouver la distribution volumique du traceur injecté dans les organes à partir des observations acquises constitue un problème inverse faisant intervenir un opérateur linéaire de projection et une perturbation aléatoire modélisée par du bruit de Poisson (variance dépendante de l'intensité locale du signal). La précision de la restitution de la concentration radioactive du traceur dans les tissus au cours du temps impactera directement la précision du diagnostic du médecin, d'où l'importance de proposer une méthode fiable et donnant des résultats de bonne qualité.

La résolution de problèmes inverses est présente dans de nombreuses applications de traitement d'images comme l'imagerie multispectrale, l'échantillonnage comprimé ou la stéréoscopie ; elle passe généralement par la minimisation d'un critère. De récentes méthodes d'optimisation convexe, nommées approches proximales, permettent de minimiser des critères non différentiables et ainsi d'exploiter des décompositions sur des trames. Ces dernières sont généralement dérivées d'analyses par ondelettes et elles fournissent

d'une part, des représentations creuses des masses de données traitées, et d'autre part, des « modèles *a priori* » simples et efficaces pour modéliser ces données (par exemple, des données physiologiques).

Les algorithmes proximaux existants gèrent efficacement la résolution de problèmes inverses en présence de bruit gaussien. Cependant, comme nous l'avons précédemment précisé, la TEP fait intervenir du bruit de Poisson. Il est donc nécessaire d'adapter ces méthodes proximales à la reconstruction de l'activité dans les tissus en TEP. Remarquons que les approches proximales possèdent des propriétés théoriques de convergence qui assurent la fiabilité de la solution.

Si les approches proximales ont jusqu'à présent montrées leur efficacité sur des problèmes spécifiques de débruitage, de restauration ou de reconstruction monocomposantes, celles-ci n'ont encore été que peu utilisées pour résoudre des problèmes inverses multicomposantes.

L'un des objectifs de ce travail a été la minimisation du temps de calcul car le médecin souhaite notamment souvent obtenir les résultats au plus vite. Profitant du fait que les performances des ordinateurs ont largement évolué au cours de ces dernières années, qu'elles permettent en particulier de stocker plus de données et de favoriser la parallélisation des calculs sur divers processeurs pour accélérer les calculs, nous chercherons à proposer des algorithmes parallélisables afin de tirer profit de telles architectures.

Ce travail a donné naissance à deux collaborations. La première s'est déroulée dans le cadre du projet ANR OPTIMED (« Algorithmes d'optimisation décomposés pour les problèmes d'imagerie médicale de grande taille ») entre le Laboratoire d'Informatique Gaspard Monge (LIGM) et le Service Hospitalier Frédéric Joliot (SHFJ). La collaboration avec C. Comtat du SHFJ a permis d'apréhender les problématiques rencontrées en TEP dynamique et d'appliquer les méthodes proposées dans les chapitres 3 et 4 sur des données simulées et réelles. Les résultats obtenus sont présentés dans le chapitre 7. Un exemple de données réelles est présenté sur la figure 1.1 ; elle présente les données spatio-temporelles acquises et l'activité reconstruite par la méthode actuellement utilisée en routine clinique.

La seconde collaboration avec L. Briceños et P. L. Combettes du Laboratoire Jacques-Louis Lions (également partenaire d'OPTIMED) a eu pour objectif de résoudre des problèmes inverses multicomposantes. L'objectif était d'étendre l'utilisation des algorithmes proximaux à un cadre multivarié (donc de grande taille) et ainsi de résoudre des problèmes inverses variés comme ceux se posant en stéréoscopie, en imagerie couleur ou dans le cadre de la décomposition en composantes de texture et géométrique.

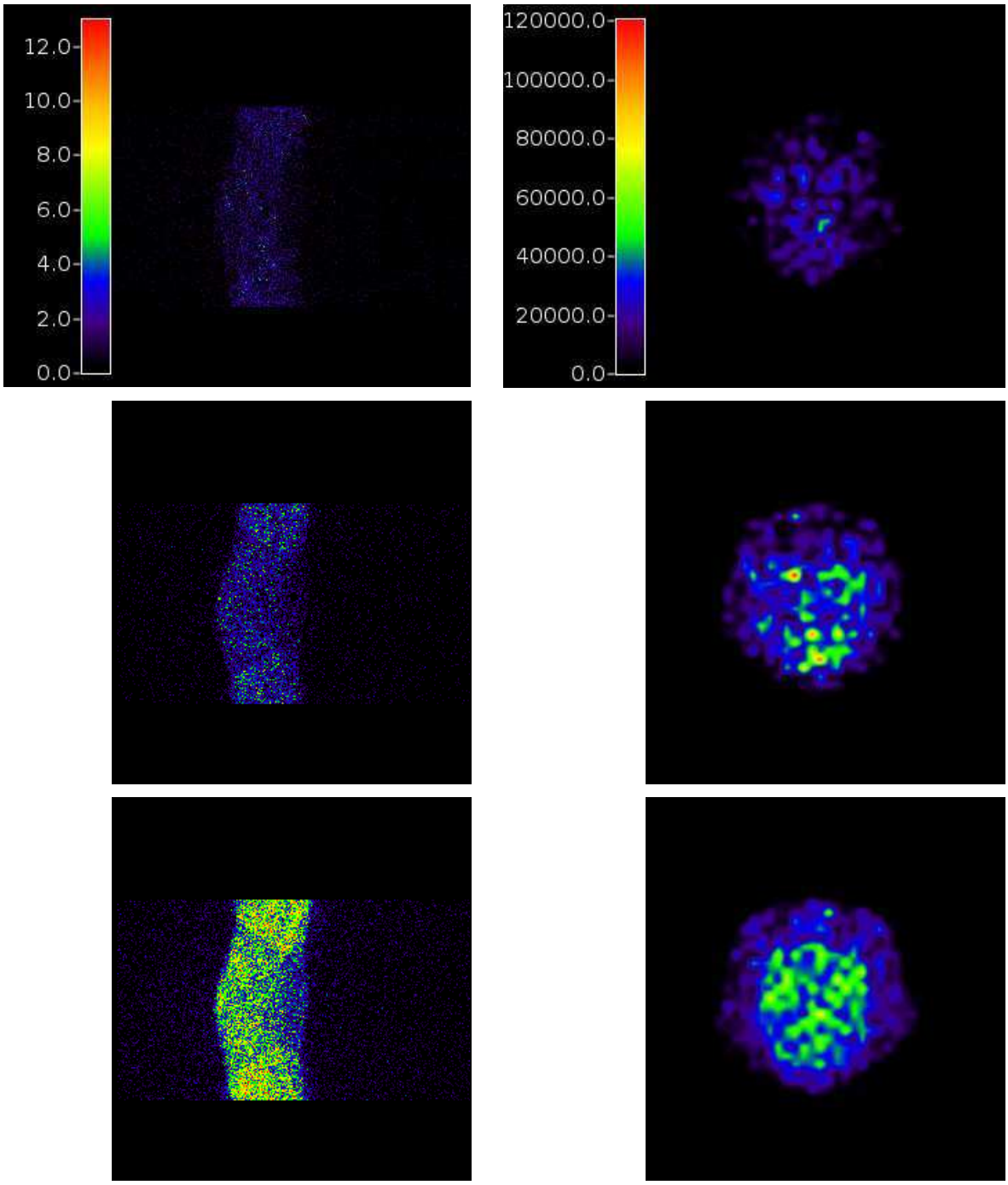


FIGURE 1.1 – Coupes 2D de données dynamiques acquises (gauche) et de l'activité reconstruite (en Bq/cm^3) par la méthode actuellement utilisée en routine clinique (droite) sur un examen [^{18}F]-FDG pour différentes trames temporelles.

1.2 Vue d'ensemble du travail effectué

1.2.1 État de l'art

Dans le deuxième chapitre, nous présenterons un rapide état de l'art des problèmes inverses en imagerie. Nous y détaillerons les méthodes permettant d'aborder ces problématiques. Nous verrons que leur résolution peut passer par la minimisation d'un critère convexe composé de plusieurs fonctions. Ces fonctionnelles sont choisies à partir d'informations *a priori* sur la nature du bruit perturbant les images, sur l'opérateur linéaire de dégradation (opérateur de projection en TEP) ou encore sur la nature des images. Nous présenterons les grandes classes d'algorithmes d'optimisation convexe, leurs avantages et leurs inconvénients respectifs. Nous listerons enfin les méthodes utilisées dans le cas particulier où la dégradation est un bruit de Poisson.

1.2.2 Algorithmes imbriqués et extension quadratique

Les algorithmes proximaux comme l'algorithme explicite-implicite ou l'algorithme de Douglas-Rachford ont montré leur intérêt pour la résolution de problèmes inverses. Ces algorithmes permettent de minimiser un critère composé de deux fonctions convexes. Il en résulte que des problèmes de restauration/reconstruction en présence de bruit gaussien sont résolus efficacement avec l'algorithme explicite-implicite, alors que l'algorithme de Douglas-Rachford permet de gérer des problèmes de débruitage en présence de bruit non-nécessairement additif gaussien.

Afin d'utiliser le formalisme proximal pour la résolution de problèmes de restauration ou de reconstruction en présence de bruit de Poisson, nous combinerons deux outils : (i) des algorithmes proximaux imbriqués qui permettent de minimiser un critère composé d'une somme de trois fonctions et pour lesquels nous fourniront les preuves de convergence associées ; (ii) une extension quadratique permettant d'approximer l'antilog-vraisemblance poissonienne.

1.2.3 Algorithme proximal parallèle et régularisation hybride

L'algorithme de Douglas-Rachford formulé dans un espace produit conduit à l'algorithme parallèle proximal (PPXA). Cet algorithme permet de minimiser des critères convexes composés d'une somme de plus de deux fonctions. Cet algorithme, ses preuves de convergence et son application à des problématiques de restauration d'images en présence de bruit gaussien ont été proposés dans [Combettes, Pesquet, 2008].

Son application à des problématiques de restauration d'images en présence d'un bruit qui ne serait pas additif gaussien n'est pas immédiate. Le problème provient du calcul de l'opérateur proximal, associé au terme d'attache aux données, qui ne prend pas de forme explicite. Nous proposerons dans ce chapitre de décomposer le terme d'attache aux don-

nées en une somme de fonctions dont l'opérateur proximal se calcule explicitement. Ce type de décomposition nous permettra en particulier de gérer efficacement des problèmes de déconvolution et de super-résolution en présence de bruit non-additif gaussien. De plus, nous proposerons d'utiliser une régularisation, que nous nommerons hybride, composée d'un terme de variation totale et d'un terme favorisant la parcimonie sur les coefficients de trames d'ondelettes du signal recherché. Ce type de régularisation a été précédemment utilisé pour résoudre des problèmes de déconvolution en présence de bruit gaussien. Nous verrons dans ce chapitre son influence lorsque le bruit est poissonien.

1.2.4 Formulations à l'analyse et à la synthèse

Dans les méthodes précédemment citées pour résoudre des problèmes inverses, nous faisons appel à des représentations sur des trames. L'utilisation de tels outils permet de formuler le critère de deux façons : la formulation à l'analyse et la formulation à la synthèse. Ces deux formulations sont équivalentes sous certaines hypothèses que nous clarifierons dans ce paragraphe. De plus, nous verrons dans ce chapitre que les algorithmes proximaux s'appliquent, d'une part, plus facilement à des critères formulés à la synthèse et, d'autre part, pour des trames dites ajustées. Nous proposerons des solutions pour utiliser l'algorithme explicite-implicite et une version étendue de l'algorithme PPXA, nommée PPXA+, pour les deux formulations, dans le cas de certaines trames non-ajustées.

1.2.5 Algorithmes proximaux multicomposantes

De nombreux problèmes notamment dans les domaines de l'analyse de données à valeurs complexes, la stéréoscopie, l'imagerie couleur ou la décomposition d'images en composantes de géométrie et de texture sont des problématiques de traitement d'images multicomposantes. Ce chapitre s'intéressera à la formulation multicomposante des algorithmes proximaux et aux preuves de convergence associées. Nous proposerons également diverses applications à la résolution de tels problèmes inverses.

1.2.6 Reconstruction dynamique d'images TEP

Pour résoudre des problèmes de déconvolution en présence de bruit de Poisson nous avons d'une part proposé l'utilisation d'algorithmes imbriqués combinés avec une extension quadratique et, d'autre part, l'utilisation de PPXA pour minimiser un critère incluant une régularisation hybride. Dans ce chapitre, après une brève introduction à la reconstruction de données TEP dynamiques, nous adapterons les méthodes proposées dans le cadre de la déconvolution au problème de reconstruction d'images TEP dynamiques. Pour évaluer les performances de nos méthodes nous avons créé un fantôme adapté à nos problèmes. Les résultats obtenus sur des données dynamiques réelles seront également présentés.

1.3 Publications

1.3.1 Articles de revus acceptés ou publiés

– C. Chaux, J.-C. Pesquet et N. Pustelnik, “Nested iterative algorithms for convex constrained image recovery problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 730-762, June 2009.

1.3.2 Articles soumis

– N. Pustelnik, C. Chaux et J.-C. Pesquet, “Parallel ProXimal Algorithm for image restoration using hybrid regularization,” soumis à IEEE Trans. Image Process., 2010.
– L. Briceño-Arias, P. L. Combettes, J.-C. Pesquet et N. Pustelnik, “Proximal algorithms for multicomponent image processing,” soumis à J. Math. Imag. Vis., 2010.

1.3.3 Articles de conférences invités

– N. Pustelnik, C. Chaux et J.-C. Pesquet , “Hybrid regularization for data restoration in the presence of Poisson noise,” *European Signal processing Conference (EUSIPCO)*, 5 pp., Glasgow, Scotland, 24-28 Août 2009.
– C. Chaux, J.-C. Pesquet et N. Pustelnik, “Frame-based proximal algorithms for Poisson data recovery,” *SIAM Conference on Imaging Science*, Chicago, Illinois, 12-14 Avril 2010.

1.3.4 Autres articles de conférences

– N. Pustelnik, C. Chaux et J.-C. Pesquet, “A constrained forward-backward algorithm for image recovery problems,” *European Signal processing Conference (EUSIPCO)*, 5 pp., Lausanne, Suisse, 25-29 Août 2008.
– N. Pustelnik, C. Chaux et J.-C. Pesquet, “A wavelet-based quadratic extension method for image deconvolution in the presence of Poisson noise,” *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 4 pp., Taipei, Taiwan , 19-24 Avril 2009.
– L. Chaâri, N. Pustelnik, C. Chaux et J.-C. Pesquet, “Solving inverse problems with overcomplete transforms and convex optimization techniques,” *SPIE Wavelets XIII*, vol. 7446, 14 pp., San Diego, Californie, USA , 2-6 Août 2009.
– N. Pustelnik, C. Chaux et J.-C. Pesquet, “Extension des algorithmes imbriqués pour la résolution de problèmes d’optimisation convexe en imagerie,” GRETSI , 4 pp., Dijon, France, 8-11 Septembre 2009.

- N. Pustelnik, C. Chaux, J.-C. Pesquet, F. C. Sureau, E. Dusch et C. Comtat, “Adapted Convex Optimization Algorithm for Wavelet-Based Dynamic PET Reconstruction,” *International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Fully 3D)*, 4 pp., Bejing, Chine, 5- 10 Septembre 2009.
- N. Pustelnik, J.-C. Pesquet et C. Chaux , “Proximal methods for image restoration using a class of non-tight frame representations,” *European Signal processing Conference (EUSIPCO)*, 5 pp., Aalborg, Danemark, 23-27 Août 2010.
- L. M. Briceño-Arias, P. L. Combettes, J.-C. Pesquet, and N. Pustelnik, “Proximal method for geometry and texture image decomposition,” *IEEE International Conference on Image Processing (ICIP)*, Honk Kong, 26-29 Septembre 2010.
- N. Pustelnik, C. Chaux, J.-C. Pesquet, and C. Comtat, “Parallel Algorithm and Hybrid Regularization for Dynamic PET Reconstruction,” *IEEE Medical Imaging Conference (MIC)*, Knoxville, Tennessee, Oct. 30 - Nov. 6 2010.

1.3.5 Autres présentations

- *CEA – Service Hospitalier Frédéric Joliot*, “Restauration/reconstruction d’images de grande taille. Application à la Tomographie par Emission de Positrons,” 23 juin 2009.
- *Ecole d’été de Peyresq*, “Résolution de problèmes inverses à l’aide de transformées redondantes et de techniques d’optimisation convexe,” 23 juillet 2009.
- *GdR MOA – Conférence 2009*, “Algorithme Parallèle ProXimal pour la restauration d’images,” 20 octobre 2009.
- *Séminaire SCAM – Université de Créteil Paris-Est*, “Ondelettes et optimisation convexe pour la restauration d’images,” 10 décembre 2009.
- *GdR ISIS – Journée Modélisation mathématique des textures*, “Méthodes proximales pour la décomposition d’images en géométrie et texture,” 24 mars 2010.
- *CEA – Service Hospitalier Frédéric Joliot*,“Optimisation convexe et régularisation hybride. Application à la Tomographie par Emission de Positrons dynamique,” 18 mai 2010.

CHAPITRE 2

Introduction aux problèmes inverses et à l'optimisation convexe

2.1 Problèmes inverses

2.1.1 Quelques exemples

En traitement d'images, on emploie la terminologie de problème inverse (opposée à la notion de problème direct) dans de nombreuses applications. La résolution d'un problème inverse consiste à retrouver un signal (ou une image) le plus proche possible d'un « signal considéré comme référence », par le biais des observations acquises. En raison du mode d'acquisition et de l'imperfection des détecteurs, ces observations sont la plupart du temps dégradées par un opérateur linéaire et/ou bruitées. Plusieurs exemples de modalités s'inscrivent dans ce cadre, comme la tomographie par émission de positrons (opérateur de projection + bruit de Poisson), la microscopie à champ large (opérateur de convolution + bruit gaussien), la microscopie confocale (opérateur de convolution + bruit de Poisson), l'échantillonnage comprimé plus connu sous le nom de « compressed sensing » (opérateur associé à une matrice parcimonieuse + bruit gaussien) ou l'imagerie satellitaire (opérateur de convolution + bruit gaussien).

Une formulation mathématique du problème direct est la suivante,

$$z = \mathcal{D}_\alpha(A\overline{y}) \tag{2.1}$$

où $z = (z_j)_{1\leq j\leq M} \in \mathbb{R}^M$ et $\overline{y} = (\overline{y}_i)_{1\leq i\leq N} \in \mathbb{R}^N$ sont des représentations vectorielles des observations de taille $M = M_1 \times M_2$ et de l'image originale de taille $N = N_1 \times N_2$. $A = (A_{j,i})_{1\leq j\leq M, 1\leq i\leq N} \in \mathbb{R}^{M\times N}$ est la matrice associée à l'opérateur linéaire de dégradation et $\mathcal{D}_\alpha : \mathbb{R}^M \mapsto \mathbb{R}^M$ représente une dégradation par un bruit de paramètre $\alpha > 0$.

La résolution du problème inverse associé à (2.1) nous conduira à une solution $\widehat{y}(z) \in \mathbb{R}^N$ qui par abus de notation sera notée, dans la plupart des cas, $\widehat{y}$. Dans ce manuscrit, nous supposerons connues la matrice associée à l'opérateur linéaire et la statistique du bruit.

Le type de dégradation présent en TEP est illustré figure 2.1. Ce modèle de dégradation se base sur un opérateur de projection où M est sensiblement plus petit que N et où le

bruit présent est un bruit de Poisson (dont la variance dépend de l'intensité). Notons que ce type de bruit est associé à des processus de comptage ce qui explique pourquoi il est présent en TEP et, plus généralement, dans l'ensemble des applications pour lesquelles les observations représentent le nombre de photons reçus par un capteur dans un intervalle de temps donné. Dans le modèle (2.1), $\mathcal{D}_\alpha$ modélise l'effet du bruit de Poisson et α le paramètre d'échelle associé. Plus α est petit, plus la dégradation est importante comme on peut le remarquer sur les figures 2.1(b) et 2.1(c). En TEP, les observations z forment un sinogramme et représentent le nombre de paires de photons détectées par une ligne (ou par un tube) de réponse formée par deux détecteurs. Cette application sera détaillée dans le chapitre 7.

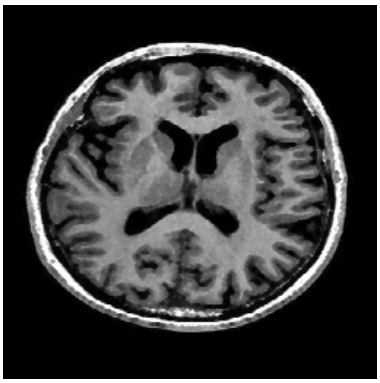
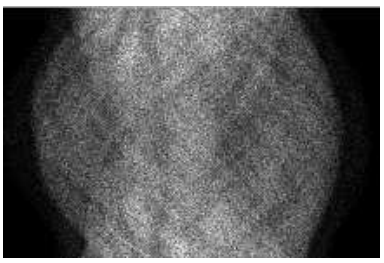
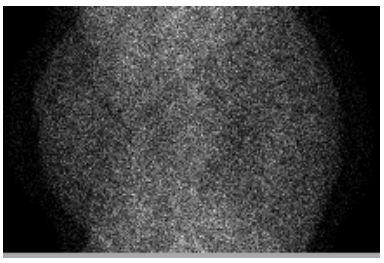


Image originale $\overline{y} \in \mathbb{R}^{256\times256}$
(a)



Sinogramme $z \in \mathbb{R}^{144\times288}$ avec $\alpha = 0.1$
(b)

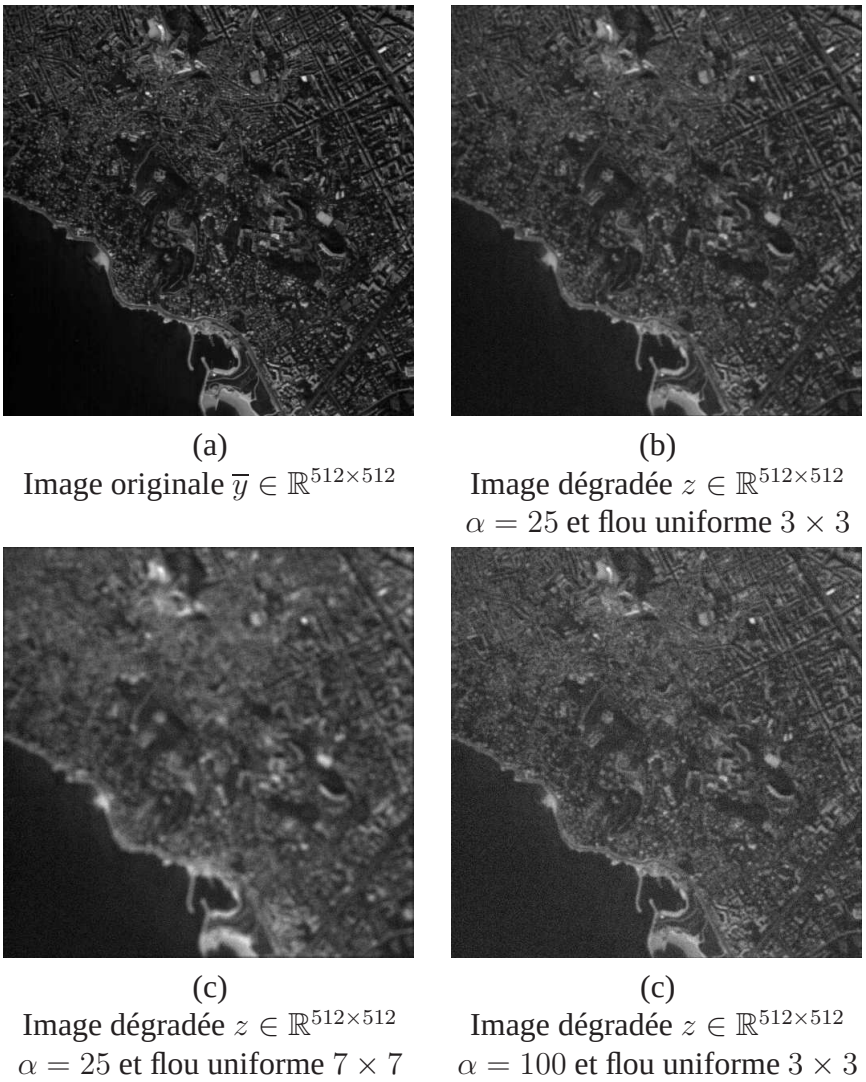


Sinogramme $z \in \mathbb{R}^{144\times288}$ avec $\alpha = 0.01$
(c)

FIGURE 2.1 – Dégradation en TEP.

Pour d'autres modalités, comme c'est le cas en imagerie satellitaire, le modèle de dégradation peut s'écrire $z = Ay + b$. Dans ce cas particulier, le nombre d'observations est égal au nombre de pixels de l'image ($M = N$), $A \in \mathbb{R}^{N\times N}$ est un opérateur de convolution possédant une structure circulante bloc-circulante et $b \in \mathbb{R}^M$ est une réalisation d'un vecteur aléatoire suivant une loi normale de moyenne nulle et de variance $\sigma^2 = \alpha$. Un exemple d'image satellitaire dégradée par un flou uniforme et un bruit blanc gaussien

de variance α est illustré sur la figure 2.2. Différentes valeurs pour la taille du noyau du flou et pour la variance du bruit sont considérées. On remarque que plus la taille du flou est importante, plus les détails sont lissés et, que plus la variance du bruit blanc gaussien est élevée, plus l'image est irrégulière.



(a)
Image originale $\overline{y} \in \mathbb{R}^{512\times512}$

(b)
Image dégradée $z \in \mathbb{R}^{512\times512}$
$\alpha = 25$ et flou uniforme 3×3

(c)
Image dégradée $z \in \mathbb{R}^{512\times512}$
$\alpha = 25$ et flou uniforme 7×7

(c)
Image dégradée $z \in \mathbb{R}^{512\times512}$
$\alpha = 100$ et flou uniforme 3×3

FIGURE 2.2 – Dégradation en imagerie satellitaire.

En pratique, même dans des cas relativement classiques, comme une convolution et l'addition d'un bruit gaussien, obtenir une image restaurée de bonne qualité ne sera pas toujours chose aisée. Cela correspond à des notions de « problèmes mal posés ou/et mal conditionnés » que nous détaillons maintenant.

2.1.2 Problème mal posé, mal conditionné et régularisation

Si $A \in \mathbb{R}^{M\times N}$ représente la matrice associée à l'opérateur de dégradation linéaire, la résolution d'une équation du type $z = Ay$ est dite bien posée si la solution $\widehat{y}$ vérifie les conditions de Hadamard [Hadamard, 1902] c'est à dire :

(i) existence d'une solution *i.e.* $\mathrm{Im}\ A = \mathbb{R}^M$,

(ii) unicité de la solution *i.e.* $\ker A = \{0\}$,

(iii) stabilité de la solution par rapport aux données *i.e.* $\forall (z, z') \in \left(\mathbb{R}^M\right)^2$, $\|z - z'\| \to 0 \ \Rightarrow\ \|\widehat{y}(z) - \widehat{y}(z')\| \to 0$.

La condition d'existence signifie que tout élément $z \in \mathbb{R}^M$ est une image d'un élément $y \in \mathbb{R}^N$. La condition d'unicité implique que l'ensemble des solutions de l'équation $Ay = 0$, où $y \in \mathbb{R}^N$, se réduit au singleton nul. La condition de stabilité signifie qu'une petite perturbation sur l'image dégradée engendre une faible variation sur l'image reconstruite.

En pratique, plusieurs situations se présentent. Dans le cas où $r = M = N$ désigne le rang de la matrice A, une solution existe toujours car $\mathrm{Im}\ A = \mathbb{R}^N$; de plus, cette solution est unique car A est injectif et le problème est alors bien défini. Néanmoins A peut être mal conditionnée. Si $(\lambda_i)_{1\leq i\leq \min(N,M)}$ représentent les valeurs singulières de A rangées par ordre décroissant, le mauvais conditionnement de la matrice A se traduit par un rapport (de conditionnement) $\lambda_1/\lambda_{\min(N,M)}$ très grand. Ce mauvais conditionnement apparaît par exemple lorsque A modélise un opérateur de convolution dont la réponse fréquentielle s'annule à certaines fréquences (typiquement un filtre passe bas).

Pour illustrer cette notion de mauvais conditionnement et les mauvais résultats de reconstruction induits, plaçons nous dans le cas $z = Ay + b$ où b désigne le bruit. Dans le cas où A est carrée inversible, l'image reconstruite peut être obtenue par filtrage inverse, c'est à dire

$$\widehat{y} = \widehat{y}(z) = A^{-1}z. \tag{2.2}$$

Il en résulte que

$$\widehat{y} = A^{-1}(Ay + b) = y + A^{-1}b \tag{2.3}$$

Dans le cas où A est mal conditionnée, $A^{-1}b$ va amplifier le bruit ce qui conduira à une image irrégulière. Lorsque A n'est pas inversible, on peut utiliser l'approche des moindres carrés également connue sous le nom d'inverse généralisée. Elle consiste à minimiser la distance euclidienne entre Ay et les observations z, ce qui nous conduit au problème de minimisation suivant,

$$\widehat{y} \in \underset{y\in\mathbb{R}^N}{\mathrm{Argmin}}\ \|Ay - z\|^2.$$

Dans le cas où A n'est pas injective, on choisit souvent la solution $\widehat{y}$ de norme minimale. Cette approche conduit au même désagrément que le filtrage inverse, à savoir l'irrégularité de la solution.

Dans le cas où $r = M < N$, on parle de problème sous-déterminé et lorsque $r = N < M$ le problème est sur-déterminé. L'unicité n'est alors plus assurée mais elle peut être rétablie en introduisant un terme de régularisation [Tikhonov, 1963].

Finalement, quelles que soient les dimensions de A, la volonté de stabiliser la solution ou d'assurer l'unicité de celle-ci nous conduit à formuler le problème comme suit,

$$\widehat{y} \in \underset{y \in \mathbb{R}^N}{\operatorname{Argmin}} \, \|Ay - z\|^2 + \chi\phi(y), \tag{2.4}$$

où $\phi \colon \mathbb{R}^N \mapsto \,]-\infty, +\infty]$ représente le terme de régularisation et $\chi > 0$ le paramètre de régularisation permettant d'ajuster le compromis entre la proximité aux observations et le degré de régularité.

2.2 Résolution de problèmes inverses

Nous venons de rappeler les raisons pour lesquelles la résolution d'un problème inverse passe souvent par la minimisation d'un critère. De façon à obtenir une image restaurée la plus proche possible de l'image considérée comme référence, ce paragraphe vise à présenter les principales fonctionnelles pouvant constituer le critère. Le choix de ces fonctions passe par (i) la connaissance *a priori* du bruit et de l'opérateur linéaire ayant contaminés les données et (ii) la connaissance *a priori* de l'image considérée comme référence (image constante par morceaux, image de texture,$\cdots$).

2.2.1 Interprétation bayésienne

Afin de mieux appréhender le critère à minimiser pour résoudre efficacement un problème inverse général modélisé par l'équation (2.1), il est intéressant de faire appel à une formulation bayésienne. On suppose alors que y et z sont des réalisations des vecteurs aléatoires Y et Z. L'image inconnue est notée y et elle peut être calculée à l'aide de l'estimateur du Maximum A Posteriori (MAP). Cet estimateur consiste, dans le cas de figure considéré, à trouver la valeur de y qui maximise la distribution de probabilité *a posteriori* sur les pixels de l'image, notée $\mu_{Y|Z=z}(y)$. Cependant, nous n'avons pas accès à un modèle précis pour cette loi. Une solution réside dans l'utilisation de la loi de Bayes qui exprime la loi *a posteriori* en fonction de la vraisemblance $\mu_{Z|Y=y}(z)$, de la loi *a priori* notée $\mu_Y(y)$ et de la loi marginale $\mu_Z(z)$. On souhaite alors trouver

$$\begin{aligned} \widehat{y} \in \underset{y \in \mathbb{R}^N}{\operatorname{Argmax}} \, \mu_{Y|Z=z}(y) &\Leftrightarrow \widehat{y} \in \underset{y \in \mathbb{R}^N}{\operatorname{Argmax}} \, \mu_{Z|Y=y}(z) \frac{\mu_Y(y)}{\mu_Z(z)} \\ &\Leftrightarrow \widehat{y} \in \underset{y \in \mathbb{R}^N}{\operatorname{Argmax}} \, \mu_{Z|Y=y}(z) \mu_Y(y). \end{aligned} \tag{2.5}$$

En se basant sur la monotonicité de la fonction logarithme, (2.5) peut s'écrire

$$\widehat{y} \in \underset{y \in \mathbb{R}^N}{\operatorname{Argmin}} \big\{ -\log \mu_{Z|Y=y}(z) - \log \mu_Y(y) \big\}. \tag{2.6}$$

Le premier terme de l'équation (2.6) correspond au terme d'attache aux données z et dépend du modèle de dégradation. Le second terme quant à lui, correspond à un terme d'*a priori* sur l'image originale $\overline{y}$. En terme de fonctionnelles, si $\psi \colon \mathbb{R}^N \mapsto \,]-\infty, +\infty]$ et $\phi \colon \mathbb{R}^N \mapsto \,]-\infty, +\infty]$ désignent respectivement, le terme d'attache aux données et le terme d'*a priori*, le critère à minimiser s'écrit sous la forme :

$$\widehat{y} \in \underset{y \in \mathbb{R}^N}{\operatorname{Argmin}} \, \psi(y) + \widetilde{\chi}\phi(y) \tag{2.7}$$

avec $\widetilde{\chi} > 0$.

Nous allons maintenant donner quelques formes répandues pour le terme d'attache aux données ψ et pour le terme d'*a priori* ϕ. Nous verrons en particulier que le choix de ψ sera dicté par le type de bruit alors que celui de ϕ sera guidé par les caractéristiques des données à estimer.

2.2.2 Terme d'attache aux données

Lorsque $\mathcal{D}_\alpha$ modélise un bruit blanc gaussien de variance $\alpha = \sigma^2$, la vraisemblance $\mu_{Z|Y=y}(z)$ est une gaussienne, *i.e.*

$$\mu_{Z|Y=y}(z) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}} \exp^{-\frac{\|Ay-z\|^2}{2\sigma^2}} \tag{2.8}$$

ce qui conduit à un terme quadratique pour le terme d'attache aux données lorsque l'on prend l'anti-logarithme de $\mu_{Z|Y=y}(z)$:

$$-\log \mu_{Z|Y=y}(z) \propto \frac{1}{2\sigma^2}\|Ay - z\|^2 = \frac{1}{2\sigma^2}\psi(y). \tag{2.9}$$

Il est clair que lorsque l'on combine (2.7) et (2.9) on se ramène au problème de minimisation (2.4) en posant $\chi = 2\sigma^2\widetilde{\chi}$. En suivant un raisonnement similaire lorsque $\mathcal{D}_\alpha$ modélise un bruit de dégradation poissonienne de paramètre d'échelle α et en rappelant que la vraisemblance poissonienne vaut

$$\mu_{Z|Y=y}(z = (z_i)_{1 \leq i \leq M}) = \prod_{i=1}^{M} \frac{\exp\big(-\alpha(Ay)_i\big)}{z_i!} (\alpha(Ay)_i)^{z_i}, \tag{2.10}$$

il apparaît que le terme d'attache aux données est lié à la divergence de Kullback-Leibler généralisée [Titterington, 1987], notée $D_{\mathrm{KLG}}(z,\cdot)$, et s'exprime par

$$-\log \mu_{Z|Y=y}(z=(z_i)_{1\leq i\leq M}) \propto \sum_{i=1}^{M} -z_i \log(\alpha(Ay)_i) + \alpha(Ay)_i = D_{\mathrm{KLG}}(z, \alpha Ay). \tag{2.11}$$

Remarque 2.1 *Il faut à ce stade remarquer qu'une statistique poissonienne sera beaucoup plus difficile à prendre en compte qu'une statistique gaussienne car l'attache aux données n'est pas finie pour toutes les valeurs de y et n'a pas une dérivée seconde bornée, ce qui nous le verrons, conduira à une première limitation sur les méthodes à envisager pour minimiser le critère* (2.7) *dans le cas poissonien.*

2.2.3 Terme de régularisation

Nous allons introduire dans ce paragraphe quelques grandes classes de régulariseurs aussi bien formulés dans le domaine image que dans un domaine transformé.

2.2.3-a) Dans le domaine image

• Un des premiers types de régularisation est celui formulé par Tikhonov [Tikhonov, 1963] qui est de la forme $\phi(y) = \|\Lambda y\|_2^2$ avec $\Lambda \in \mathbb{R}^{N\times N}$. Les choix de Λ les plus fréquents dans la littérature sont :

- $\Lambda = \mathrm{Id}$ qui favorise les solutions de faible norme (où Id désigne la matrice identité).
- Λ associée à un filtre laplacien discret qui favorise une image avec un laplacien faible c'est à dire une image dont l'intensité lumineuse variera doucement.

• Une autre approche consiste à choisir $\phi(y) = \iota_C(y)$ avec $C = \bigcap_{i=1}^{c} C_i$ où $(C_i)_{i\in\{1,\ldots,c\}}$ représentent un ensemble de $c > 0$ contraintes et où ι_C désigne la fonction indicatrice, précisée plus loin par la définition 2.3. Les contraintes sont par exemple choisies de façon à assurer que la différence entre deux pixels voisins (horizontaux, verticaux et diagonaux) soit inférieure à un certain seuil $\epsilon > 0$ *i.e.*

$$\begin{aligned}
&\forall (i,j) \in \{2,\ldots,N_1\}\times\{1,\ldots,N_2\}, && |y_{i,j} - y_{i-1,j}| < \epsilon,\\
&\forall (i,j) \in \{1,\ldots,N_1\}\times\{2,\ldots,N_2\}, && |y_{i,j} - y_{i,j-1}| < \epsilon,\\
&\forall (i,j) \in \{2,\ldots,N_1\}\times\{2,\ldots,N_2\}, && |y_{i,j} - y_{i-1,j-1}| < \epsilon,\\
&\forall (i,j) \in \{2,\ldots,N_1\}\times\{2,\ldots,N_2\}, && |y_{i-1,j} - y_{i,j-1}| < \epsilon.
\end{aligned}$$

lorsqu'on définit $y = (y_{i,j})_{\{1,\ldots,N_1\}\times\{1,\ldots,N_2\}}$. Cette approche possède le désavantage d'avoir un grand nombre de contraintes c à calculer (de l'ordre de $4\ N_1\ N_2$ contraintes).

• La méthode de régularisation dans le domaine image la plus utilisée reste une régularisation de type variation totale introduite par Rudin, Osher et Fatemi (ROF) [Rudin *et al.*, 1992]. On désigne par $\mathbf{\Omega}$ un ensemble ouvert de $\mathbb{R}^2$ et on note $\mathrm{W}^{1,1}(\mathbf{\Omega})$ l'espace de Sobolev qui définit l'espace des fonctions $f \in L^1(\mathbf{\Omega})$ telles que toutes les dérivées au sens des distributions de f, d'ordre inférieur à 1, sont également dans $L^1(\mathbf{\Omega})$, la variation totale est alors définie par,

$$\forall y \in \mathrm{W}^{1,1}(\mathbf{\Omega}), \qquad \mathrm{tv}(y) = \int_{\mathbf{\Omega}} \|\mathrm{D}y(w)\|_2 \, \mathrm{d}w \tag{2.12}$$

où $\mathrm{D}y$ représente le gradient de y. En dimension finie la variation totale trouve de nombreuses définitions suivant le type de discrétisation choisi pour l'opérateur de différentiation. La plus connue est celle proposée par Chambolle [Chambolle, 2004] qui est telle que :

$$\forall y \in \mathbb{R}^{N_1 \times N_2}, \qquad \phi(y) = \mathrm{tv}(y) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \sqrt{\big((\nabla_1\, y)_{i,j}\big)^2 + \big((\nabla_2\, y)_{i,j}\big)^2} \tag{2.13}$$

avec

$$(\nabla_1\, y)_{i,j} = \begin{cases} y_{i+1,j} - y_{i,j} & \text{si } i < N_1 \\ 0 & \text{si } i = N_1 \end{cases} \tag{2.14}$$

et

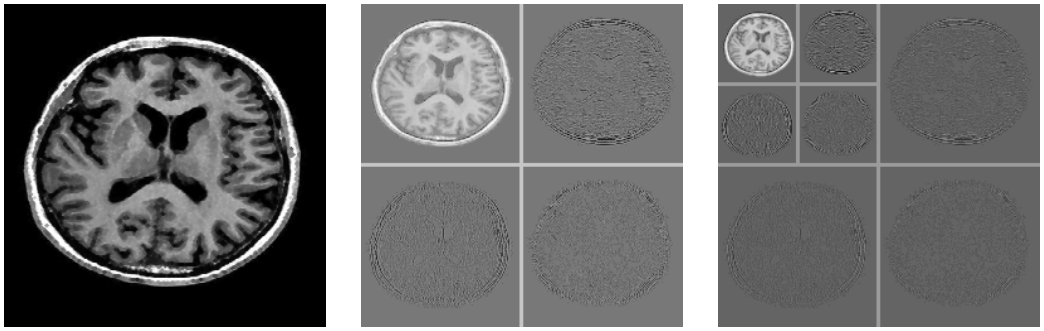
$$(\nabla_2\, y)_{i,j} = \begin{cases} y_{i,j+1} - y_{i,j} & \text{si } j < N_2 \\ 0 & \text{si } j = N_2. \end{cases} \tag{2.15}$$

La variation totale a connu une grande popularité car elle permet d'obtenir des images débruitées de relativement bonne qualité, principalement si l'image originale est de type « bande-dessinée », c'est à dire contenant beaucoup d'aplats. Cependant, pour des images naturelles, cette méthode présente des artefacts donnant un effet « de marche d'escalier » (staircasing effect) [Louchet, 2008].

2.2.3-b) Dans le domaine transformé – introduction des trames

Les transformées en ondelettes permettent de représenter une image « régulière » sur un nombre limité de coefficients d'ondelettes d'amplitude significative, et dans le même temps de répartir équitablement le bruit sur l'ensemble des composantes. Ainsi ce type de transformée est particulièrement bien adaptée à des problèmes de débruitage en utilisant des techniques de seuillage des coefficients [Donoho, 1995; Antoniadis, Fan, 2001] dont les plus connues sont le seuillage dur et le seuillage doux. Cette transformation présente deux types de coefficients : les coefficients de détails (verticaux, horizontaux et diagonaux) et les coefficients d'approximations. Les premiers sont des coefficients modélisant les hautes fréquences de l'image et les seconds, les basses fréquences. Ils

peuvent être obtenus par filtrage des lignes et colonnes intercalés avec des opérations de décimation (bancs de filtres [Mallat, 1997]) ou par l'utilisation de schémas de lifting [Sweldens, 1996]. L'analyse multirésolution consiste à itérer cette décomposition sur plusieurs niveaux. Sur les figures 2.3(b) et (c) sont représentées les transformées en ondelettes de l'image 2.3(a) sur un niveau de résolution et sur 2 niveaux de résolution. Pour plus de détails sur ces transformées, aussi appelées bases d'ondelettes orthonormales (ou bi-orthogonales), ainsi que sur leurs diverses applications, une excellente référence est le livre de S. Mallat [Mallat, 1997].



(a) (b) (c)

FIGURE 2.3 – (a) Image médicale, (b) transformée en ondelettes sur 1 niveau et (c) transformée en ondelettes sur 2 niveaux de résolution.

Plus récemment de nouvelles études ont porté sur le choix de représentations redondantes, appelées trames, permettant une meilleure description de l'image en favorisant la parcimonie des coefficients construits (peu de coefficients significatifs). Dans la suite $\mathbb{K}$ désigne un sous-ensemble non vide de $\mathbb{N}$. $\mathcal{G}$ désigne un espace de Hilbert séparable muni d'un produit scalaire noté $\langle \cdot, \cdot \rangle$ et d'une norme notée $\| \cdot \| = \sqrt{\langle \cdot, \cdot \rangle}$. Une famille $(e_k)_{k \in \mathbb{K}}$ dans $\mathcal{G}$ constitue une trame s'il existe deux constantes $\underline{\nu}$ et $\overline{\nu}$ dans $]0, +\infty[$ telles que

$$(\forall y \in \mathcal{G}) \;\; \underline{\nu}\|y\|^2 \leq \sum_{k \in \mathbb{K}} |\langle y, e_k \rangle|^2 \leq \overline{\nu}\|y\|^2. \tag{2.16}$$

L'opérateur de trame associé est un opérateur linéaire injectif borné,

$$F \colon \mathcal{G} \to \ell^2(\mathbb{K}) \colon y \mapsto (\langle y, e_k \rangle)_{k \in \mathbb{K}}, \tag{2.17}$$

son adjoint, nommé l'opérateur de synthèse de trame est un opérateur linéaire surjectif borné

$$F^* \colon \ell^2(\mathbb{K}) \to \mathcal{G} \colon (\xi_k)_{k \in \mathbb{K}} \mapsto \sum_{k \in \mathbb{K}} \xi_k e_k. \tag{2.18}$$

Remarquons qu'en dimension finie, la borne supérieure dans (2.16) est toujours satisfaite. De plus, lorsque $\underline{\nu} = \overline{\nu} = \nu$ dans (2.16), $(e_k)_{k\in\mathbb{K}}$ est appelée une trame ajustée. Dans ce cas,

$$F^* \circ F = \nu \mathrm{Id}. \tag{2.19}$$

Un exemple simple de trame ajustée est une union de ν bases orthonormales. D'autre part, une union de décompositions réelles (*resp.* complexes) en arbre dual est l'union de deux (*resp.* quatre) bases d'ondelettes orthonormales [Chaux *et al.*, 2006]. Les curvelets [Candès, Donoho, 2002; Fadili, Starck, 2009] constitutent un autre exemple de trame ajustée. De manière historique, les trames de Gabor [Daubechies, 1992] ont joué un rôle important dans de nombreux problèmes inverses. Sous certaines conditions, les contourlets [Do, Vetterli, 2005] constituent également des trames ajustées. Lorsque $F^{-1} = F^*$, une base orthonormale est obtenue. Des constructions et détails supplémentaires sur la théorie des trames peuvent être trouvés dans [Han, Larson, 2000].

Dans un tel cadre d'étude, le modèle de dégradation (2.1) devient,

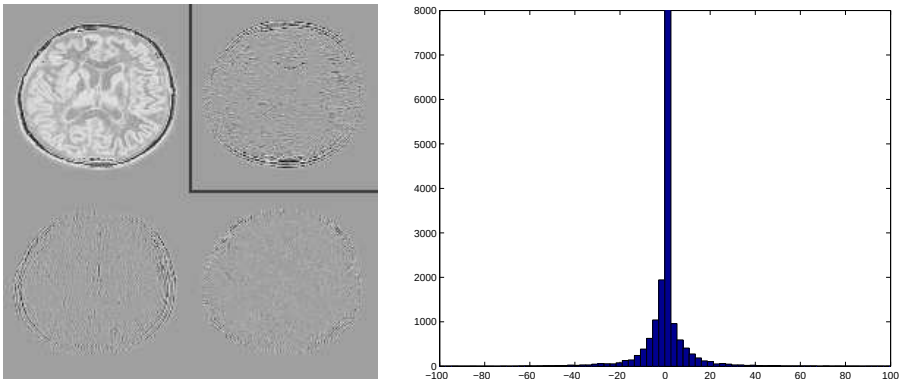
$$z = \mathcal{D}_\alpha(AF^*\overline{x}) \tag{2.20}$$

où $\overline{x} \in \mathbb{R}^K$ représente les coefficients de trame associés à l'image de référence $\overline{y}$ tels que $\overline{y} = F^*\overline{x}$ (notons que $\mathbb{K} = \{1, \ldots, K\}$, $\mathcal{G} = \mathbb{R}^N$ et $K \geq N$). Dans cette configuration, notre objectif est de retrouver des coefficients de trames $\widehat{x}$ les plus proches possible de $\overline{x}$ à partir des observations z. L'approche la plus classique consiste à minimiser un critère de la forme

$$\widehat{x} \in \underset{x\in\mathbb{R}^K}{\mathrm{Argmin}}\ \psi(F^*x) + \chi\widetilde{\phi}(x)$$

où $\chi > 0$, ψ est le terme d'attache aux données et $\widetilde{\phi}\colon \mathbb{R}^K \mapsto\ ]-\infty, +\infty]$ est le terme de régularisation. L'image reconstruite est alors $\widehat{y} = F^*\widehat{x}$.

Le choix de $\widetilde{\phi} = \|\cdot\|_1$ est motivé statistiquement par la modélisation des coefficients de trames par une loi laplacienne i.i.d. (*cf.* figure 2.4) et par le constat que l'anti-logarithme de la densité de probabilité associée conduit à une norme ℓ_1.



(a) (b)

FIGURE 2.4 – (a) Figure 2.3(b) et (b) histogramme associé à la sous-bande de détails horizontaux.

Remarque 2.2

- *Remarquons qu'un lien peut être établi entre une norme ℓ_1 appliquée sur le gradient de l'image (variation totale) et sur les coefficients d'ondelettes non-décimées par un filtre de Haar [Selesnick, Figueiredo, 2009].*
- *Cette approche fait intervenir une norme ℓ_1 et donc un critère convexe mais non différentiable. Nous verrons dans le paragraphe suivant que minimiser un critère non différentiable peut avoir un impact sur la vitesse de convergence des algorithmes mis en oeuvre.*

Dans le cadre d'un bruit gaussien [Donoho, 1995; Do, Vetterli, 2005] et également en présence d'une convolution [Daubechies *et al.*, 2004], l'utilisation de trames est avantageuse en termes de performance. Ces conclusions sont mises en évidence particulièrement pour des images naturelles, images pour lesquelles la variation totale [Figueiredo *et al.*, 2006] se révèle moins adaptée que l'utilisation de trames.

2.3 Optimisation convexe

Comme nous venons de le voir, la résolution d'un problème inverse passe souvent par la minimisation d'un critère. Dans ce manuscrit nous nous limiterons à des fonctions objectifs convexes permettant de résoudre une large classe de problèmes inverses. Dans ce paragraphe, nous commençons par rappeler des définitions et théorèmes d'analyse convexe bien connus [Zălinescu, 2002] qui nous seront indispensables dans le reste du manuscrit. Nous abordons par exemple les notions d'ensemble convexe et de projection sur un convexe, utiles lorsqu'il s'agit d'inclure dans notre critère des contraintes telles que la positivité des données. Nous nous attardons également sur des notions comme la coercivité et la stricte convexité permettant d'assurer l'existence et l'unicité du minimiseur.

Nous présentons ensuite les principales classes d'algorithmes permettant de minimiser des critères différentiables et non-différentiables. Nous faisons la différence entre ces deux types de critères, car la non-différentiabilité est une difficulté supplémentaire dans l'obtention de méthodes compétitives en terme de convergence. Dans la suite, $\mathcal{H}$ désigne un espace de Hilbert séparable muni d'un produit scalaire noté $\langle \cdot, \cdot \rangle$ et d'une norme notée $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$.

2.3.1 Quelques rappels d'analyse convexe

Définition 2.3 (Fonction indicatrice d'un ensemble) *Soit $C \subset \mathcal{H}$ un ensemble. La fonction indicatrice de cet ensemble est notée ι_C et se définit par :*

$$(\forall u \in \mathcal{H}) \quad \iota_C \colon u \mapsto \begin{cases} 0, & \textit{si} \;\; u \in C; \\ +\infty, & \textit{si} \;\; u \notin C. \end{cases}$$

Définition 2.4 (Distance à un ensemble) *Soit $C \subset \mathcal{H}$ un ensemble non vide. La distance à cet ensemble est notée d_C et se définit par*

$$\begin{aligned} d_C \colon \mathcal{H} &\to [0, +\infty[\\ u &\mapsto \inf_{v \in C} \|u - v\|. \end{aligned} \tag{2.21}$$

Définition 2.5 (Ensemble convexe) *Soit $C \subset \mathcal{H}$ un ensemble. Cet ensemble est dit convexe si*

$$\big(\forall (u, v) \in C^2\big) \; \big(\forall \lambda \in]0, 1[\big) \quad \lambda u + (1 - \lambda) v \in C.$$

Theorème 2.6 (Projection sur un convexe) *Soit $C \subset \mathcal{H}$ un ensemble convexe fermé non vide et soit $u \in \mathcal{H}$. Il existe alors un point unique noté $P_C u \in C$ tel que $d_C(u) = \|u - P_C u\|$. On appelle $P_C u$ la projection de u sur C.*

Définition 2.7 (Intérieur et intérieur relatif d'un convexe) *Soit $C \subset \mathcal{H}$ un ensemble convexe non vide. L'intérieur du convexe C est noté $\operatorname{int} C$ et se définit par*

$$\operatorname{int} C = \{u \in C \mid \exists \rho > 0, \quad \mathcal{B}(u; \rho) \subset C\},$$

où $\mathcal{B}(u; \rho)$ désigne une boule de rayon ρ centrée en u. L'intérieur relatif de C est noté $\operatorname{ri} C$ et se définit par

$$\operatorname{ri} C = \{u \in C \mid \exists \rho > 0, \quad \mathcal{B}(u; \rho) \cap \operatorname{aff} C \subset C\},$$

où aff désigne l'enveloppe affine de C.

On peut remarquer que $\operatorname{int} C \subset \operatorname{ri} C \subset C$.

Définition 2.8 (Domaine d'une fonction) *Soit $f : \mathcal{H} \to \,]-\infty, +\infty]$. Le domaine de la fonction f est l'ensemble noté $\operatorname{dom} f$ tel que*

$$\operatorname{dom} f = \{u \in \mathcal{H} \mid f(u) < +\infty\}.$$

On dit que f est propre si $\operatorname{dom} f \neq \varnothing$ (i.e. elle n'est pas identiquement égale à l'infini).

Définition 2.9 (Fonction convexe) *Une fonction $f : \mathcal{H} \to \,]-\infty, +\infty]$ est dite convexe si elle vérifie :*

$$\big(\forall (u,v) \in \mathcal{H}^2\big)\ \big(\forall \lambda \in]0,1[\big) \quad f(\lambda u + (1-\lambda)v) \leq \lambda f(u) + (1-\lambda) f(v). \tag{2.22}$$

La fonction f est strictement convexe si l'inégalité (2.22) *est stricte dès que $(u,v) \in (\operatorname{dom} f)^2$ et $u \neq v$.*

Définition 2.10 (Fonction s.c.i.) *Soit $f : \mathcal{H} \to \,]-\infty, +\infty]$ et soit $u \in \mathcal{H}$. La fonction f est semi-continue inférieurement si*

$$\big(\forall u_0 \in \mathcal{H}\big) \quad \liminf_{u \to u_0} f(u) \geq f(u_0).$$

Définition 2.11 (Classe $\Gamma_0(\mathcal{H})$) *$\Gamma_0(\mathcal{H})$ représente la classe des fonctions $f : \mathcal{H} \to \,]-\infty, +\infty]$ convexes, semi-continues inférieurement et propres.*

Définition 2.12 (Coercivité) *Soit $f : \mathcal{H} \to \,]-\infty, +\infty]$ et soit $u \in \mathcal{H}$. La fonction f est coercive si $\lim_{\|u\| \to +\infty} f(u) = +\infty$.*

Définition 2.13 (Sous-différentiel et sous-gradient) *Soit $f : \mathcal{H} \to \,]-\infty, +\infty]$ une fonction propre. Le sous-différentiel de f est noté ∂f et se définit par*

$$(\forall u \in \mathcal{H}) \quad \partial f(u) = \{w \in \mathcal{H} \mid (\forall v \in \mathcal{H}) \quad \underbrace{\big\langle v - u, w \big\rangle + f(u)}_{I(v,u,w)} \leq f(v)\}. \tag{2.23}$$

Un élément de $\partial f(u)$ est un sous-gradient de f en u.

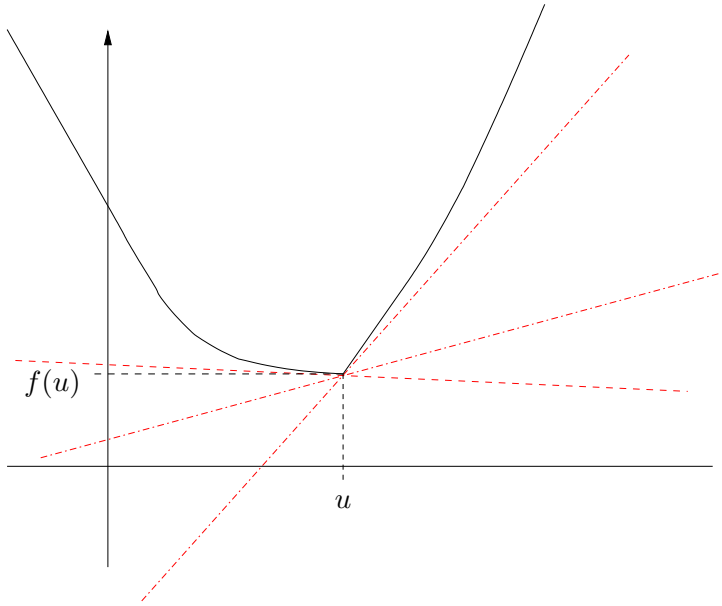


FIGURE 2.5 – Représentation graphique de $I(\cdot, u, w)$ pour trois différentes valeurs de w.

Sur la figure 2.5, la fonction affine $I(\cdot, u, w)$ définie dans l'équation (2.23) est représentée pour trois valeurs de w par les lignes discontinues rouges.

Définition 2.14 (Fonction Gâteaux-différentiable) *On dit qu'une fonction $f : \mathcal{H} \to]-\infty, +\infty]$ est Gâteaux-différentiable en $u \in \operatorname{dom} f$ si et seulement si la dérivée directionnelle*

$$f'(u; v) = \lim_{\lambda \to 0+} \frac{f(u + \lambda v) - f(u)}{\lambda}$$

existe pour toutes les directions $v \in \mathcal{H}$ et si l'application $v \mapsto f'(u; v)$ est linéaire continue.

Theorème 2.15 *Si f est convexe et Gâteaux-différentiable en $u \in \operatorname{dom} f$, alors il existe un unique sous-gradient appelé gradient de f en u noté $\nabla f(u)$. Dans ce cas $\partial f(u) = \{\nabla f(u)\}$.*

Définition 2.16 (Fonction de gradient Lipschitz) *On dit qu'une fonction $f : C \to]-\infty, +\infty]$ est de gradient β-Lipschitz avec $\beta \in]0, +\infty[$ si*

$$\big(\forall (u, v) \in C^2\big) \quad \|\nabla f(u) - \nabla f(v)\| \leq \beta \|u - v\|.$$

Définition 2.17 (Épigraphe) *L'épigraphe d'une fonction* $f\colon \mathcal{H} \to \,]-\infty,+\infty]$ *est noté* $\operatorname{epi} f$ *et se définit par*

$$\operatorname{epi} f = \big\{(u,\eta) \in \mathcal{H} \times \mathbb{R} \mid f(u) \leq \eta\big\}.$$

Définition 2.18 (Section inférieure) *La section inférieure de la fonction* $f\colon \mathcal{H} \to \,]-\infty,+\infty]$ *à la hauteur* $\eta \in \mathbb{R}$ *est notée* $\operatorname{lev}_{\leq\eta} f$ *et se définit par*

$$\operatorname{lev}_{\leq\eta} f = \big\{u \in \mathcal{H} \mid f(u) \leq \eta\big\}.$$

Définition 2.19 [Rockafellar, Wets, 2004, Définition 7.1] *On dira que la suite* $(f_\ell)_{\ell\in\mathbb{N}}$ *de fonctions définies sur* $\mathbb{R}^X$ *épi-converge vers la fonction* $f \in \Gamma_0(\mathbb{R}^X)$ *si les deux fonctions ayant pour épigraphe la limite supérieure et la limite inférieure de la suite des ensembles* $\operatorname{epi} f_\ell$ *coïncident et sont égales à* f.

Proposition 2.20 [Rockafellar, Wets, 2004, Proposition 7.4] *Soit* $(f_\ell)_{\ell\in\mathbb{N}}$ *une suite de fonctions s.c.i. définies sur* $\mathbb{R}^X$. *Si la suite* $(f_\ell)_{\ell\in\mathbb{N}}$ *est croissante (*$f_\ell \leq f_{\ell+1}$*), la suite épi-converge vers sa limite ponctuelle.*

2.3.2 Résultats d'analyse convexe

Proposition 2.21 *Soit* f *une fonction propre convexe et* $u \in \mathcal{H}$ *un minimiseur local de* f. u *est alors un minimiseur global de* f.

Proposition 2.22 [Ekeland, Témam, 1999, Chapitre 3, Proposition 1.2] *Soit* $f \in \Gamma_0(\mathcal{H})$ *et* $C \subset \mathcal{H}$ *un convexe fermé non vide.* f *admet un minimiseur sur* C *dans les deux cas suivants :*

(i) C *est borné.*

(ii) f *est coercive.*

Si, de plus, f *est strictement convexe sur* C, *la solution est unique.*

Définition 2.23 (Convergence faible) *Une suite $(u_n)_{n\in\mathbb{N}}$ d'éléments de $\mathcal{H}$ converge faiblement vers un élément $u \in \mathcal{H}$ lorsque :*

$$\forall v \in \mathcal{H}, \quad \lim_{n\to+\infty} \langle v, u_n \rangle = \langle v, u \rangle.$$

Définition 2.24 (Convergence forte) *Une suite $(u_n)_{n\in\mathbb{N}}$ d'éléments de $\mathcal{H}$ converge fortement vers un élément $u \in \mathcal{H}$ lorsque :*

$$\lim_{n\to+\infty} \|u_n - u\| = 0.$$

La convergence forte implique la convergence faible. En dimension finie, les convergences faible et forte sont équivalentes.

Définition 2.25 (taux de convergence linéaire) *Soit $\widehat{u} \in \mathcal{H}$ la solution du problème d'optimisation. On parle de taux de convergence linéaire (ou taux de convergence géométrique) lorsque la méthode d'optimisation génère une suite d'estimées $(u_n)_{n\in\mathbb{N}}$ qui satisfont :*

$$(\forall n \in \mathbb{N}) \quad \|u_n - \widehat{u}\| \leq \tau^n \|u_0 - \widehat{u}\|$$

où $\tau \in]0, 1[$ définit le taux de convergence.

Theorème 2.26 [Rockafellar, Wets, 2004, Théorème 7.33] *Soient f et $(f_\ell)_{\ell\in\mathbb{N}}$ des fonctions s.c.i et propres de $\mathbb{R}^N$. On suppose que la suite $(f_\ell)_{\ell\in\mathbb{N}}$ est telle que l'union de leur section inférieure à la hauteur $\eta \in \mathbb{R}$ soit bornée, i.e. pour tout $\eta \in \mathbb{R}$, $\bigcup_{\ell\in\mathbb{N}} \mathrm{lev}_{\leq\eta}\, f_\ell$ est bornée. De plus, on suppose que $(f_\ell)_{\ell\in\mathbb{N}}$ épi-converge vers sa limite ponctuelle f. Si f possède un unique minimiseur noté $\widehat{u}$, toute suite $(\widehat{u}_\ell)_{\ell\in\mathbb{N}}$ de minimiseurs de $(f_\ell)_{\ell\in\mathbb{N}}$ converge vers $\widehat{u}$.*

2.3.3 Algorithmes

Ce paragraphe n'a pas la prétention de lister l'ensemble des algorithmes permettant de minimiser des fonctionnelles convexes mais plutôt de rappeler certaines grandes classes d'algorithmes. Dans la suite de ce paragraphe, le problème de minimisation convexe considéré est de la forme :

$$\widehat{u} \in \underset{u\in\mathcal{H}}{\mathrm{Argmin}} \sum_{j=1}^{J} f_j(u) \tag{2.24}$$

où pour tout $j \in \{1, \ldots, J\}$, $f_j \in \Gamma_0(\mathcal{H})$. Pour résoudre ce type de problèmes, il faut faire appel à des algorithmes itératifs générant une suite $(u_n)_{n\in\mathbb{N}}$ qui converge (au moins faiblement) vers une solution du problème (2.24). La solution est unique lorsque le critère est strictement convexe et dans ce cas

$$\widehat{u} = \underset{u\in\mathcal{H}}{\arg\min} \sum_{j=1}^{J} f_j(u) \tag{2.25}$$

2.3.3-a) Algorithmes permettant de minimiser des critères différentiables

Dans ce paragraphe, nous considérons le cas particulier de (2.24) où $J = 1$ et f_1 est une fonction convexe différentiable de gradient β_1-Lipschitz. Nous rappelons les algorithmes de base permettant de résoudre une telle problématique.

• Les méthodes de gradient sont basées sur le fait que pour tout $u \in \mathcal{H}$, $\nabla f_1(u)$ correspond à la direction possédant le taux de croissance le plus élevé de f_1 en u. Par conséquent, $-\nabla f_1(u)$ correspond au taux de décroissance le plus grand ce qui indique un bon choix de direction pour trouver le minimiseur de f_1. Soit $u_0 \in \mathcal{H}$, l'initialisation de l'algorithme, alors les itérations $n \in \mathbb{N}$ construisent une suite $(u_n)_{n\in\mathbb{N}}$ telle que

$$u_{n+1} = u_n - \gamma_n \nabla f_1(u_n),$$

où $(\gamma_n)_{n\in\mathbb{N}}$ représente la suite des pas de l'algorithme. La difficulté réside dans le choix du pas permettant d'approcher le minimum le plus rapidement possible. La méthode de *plus grande pente* choisit le pas de l'algorithme permettant de maximiser la taux de décroissance de la fonction objectif à chaque itération en calculant $\gamma_n = \arg\min_{\gamma>0} f_1(u_n - \gamma\nabla f_1(u_n))$ qui est calculé le plus souvent par des approches dichotomiques. Parmi les méthodes de pas adaptatif on peut également citer la méthode de descente de gradient avec recherche linéaire d'Armijo où le pas γ_n est admissible s'il vérifie

$$f_1\big(u_n - \gamma_n \nabla f_1(u_n)\big) \leq f_1(u_n) - \epsilon\gamma_n \|\nabla f_1(u_n)\|,$$

avec $\epsilon \in]0, 1[$. Il y a également la stratégie de [Barzilai, Borwein, 1988] qui fait intervenir l'inverse du Hessien de f_1. L'utilisation d'un pas adaptatif permet d'accroître les performances d'une descente de gradient à pas constant. Il faut néanmoins rester conscient que les méthodes à pas adaptatifs sont coûteuses à chaque itération de l'algorithme. Remarquons tout de même que des preuves de convergence faible vers $\widehat{u}$ sont garanties si $\gamma_n \in]0, 2/\beta_1[$. D'autre part, lorsque $f_1 = \|A \cdot - z\|^2$, l'algorithme de gradient résultant est connu sous le nom d'algorithme de Landweber [Landweber, 1951].

• Les méthodes de Newton ne se contentent pas d'utiliser la dérivée première mais s'appuient également sur la dérivée seconde. L'idée de cet algorithme consiste à construire successivement une approximation quadratique de la fonction objectif f_1 en un point u_n ajustée à partir de la dérivée première et seconde, puis on trouve le minimiseur de cette fonction quadratique plutôt que de minimiser f_1 directement. Le minimiseur est alors utilisé comme nouveau point de construction d'une approximation. La procédure est ensuite répétée jusqu'à la convergence de l'algorithme. L'approximation quadratique peut être construite en utilisant le développement en série de Taylor de la fonction f_1 au point courant u_n, en négligeant les termes d'ordre supérieur ou égal à trois. Les itérations sont alors de la forme :

$$u_{n+1} = \arg\min_{u\in\mathcal{H}} \underbrace{f_1(u_n) + \nabla f_1(u_n)^\top (u-u_n) + \frac{1}{2}(u-u_n)^\top H(u_n)(u-u_n)}_{I(u,u_n)} \tag{2.26}$$

où H désigne le Hessien de f_1 en u_n. Remarquons que si H désigne une matrice définie positive telle que pour tout $v \in \mathcal{H}$, la fonction $I(\cdot, v)$, définie dans l'équation (2.26), soit une approximation majorante de f_1 en v alors on parle d'algorithme de majoration-minimisation (MM) [Hunter, Lange, 2004]. Lorsque H désigne une approximation du Hessien, on parle de méthode de quasi-Newton. Cette approche permet généralement d'améliorer la vitesse de convergence.

Davantage de détails sur ces méthodes gérant des critères différentiables sont présentés dans [Chouzenoux *et al.*, 2009]. En particulier, des méthodes à barrières sont étudiées car elles permettent de gérer des fonctions différentiables à dérivée seconde non bornée (comme la divergence de Kullback-Leibler).

Remarque 2.27 *Nous avons présenté ces algorithmes dans le cas $J = 1$. Néanmoins ils peuvent être utilisés pour minimiser un critère composé des diverses fonctionnelles. Cela requiert que chaque fonctionnelle soit différentiable et souvent de gradient Lipschitz de façon à obtenir des garanties de convergence.*

2.3.3-b) Algorithmes permettant de minimiser des critères non différentiables

Commençons par formaliser un cas particulier du problème (2.24) lorsque $J = 1$ et $f_1 = \iota_C$, cela conduit à :

Problème 2.28 *Soit $C = \bigcap_{i=1}^{c} C_i$, où $(C_i)_{i\in\{1,\dots,c\}}$ est une famille d'ensembles convexes fermés non vides de $\mathcal{H}$.*

$$\textit{Trouver} \qquad \widehat{u} \in C.$$

Le problème 2.28 est connu sous le nom d'approche ensembliste. C est appelé ensemble admissible lorsque $C \neq \varnothing$ et la solution est admissible lorsqu'elle vérifie l'ensemble des contraintes $(C_i)_{i\in\{1,\dots,c\}}$. Ce formalisme est très utilisé en traitement d'images pour des applications diverses (se référer à [Youla, Webb, 1982], [Capricelli, 2008, Chapitre 3] et aux références associées). La formulation des contraintes est issue de la connaissance *a priori* des données à reconstruire et également des observations. Notons que des exemples de contraintes ont été donnés dans le paragraphe 2.2.3-a) pour la résolution de problèmes inverses. Le problème 2.28 peut être résolu par l'algorithme séquentiel POCS (Projection Onto Convex Sets) [Bregman, 1965] qui consiste à initialiser $u_0 \in \mathcal{H}$ et pour tout $n \in \mathbb{N}$, calculer

$$u_{n+1} = P_{i(n)} u_n$$

où $i(n)$ représente un balayage cyclique tel que $i(n) = (n \mod c) + 1$.
La convergence faible de cet algorithme a été démontrée dans [Bregman, 1965]. Remarquons qu'une version étendue introduisant un paramètre de relaxation a été proposée dans [Gurin *et al.*, 1967], article dans lequel la convergence forte a également été démontrée. Des variantes de l'algorithme POCS telles que ART (Algebraic Reconstruction Technique) [Gordon *et al.*, 1970] ou PPM (Parallel Projection Method) [Combettes, 1994; Iusem, De Pierro, 2006] ont été proposées pour effectuer ces projections en parallèle et permettre ainsi d'améliorer les performances de POCS ne pouvant traiter qu'une contrainte à chaque itération. Ces algorithmes sont utilisés pour résoudre des problèmes inverses [Trussell, Civanlar, 1984] mais peuvent parfois, comme dans certaines applications en traitement d'images, faire appel à un nombre de contraintes très élevé. Une méthode parallèle semble donc beaucoup plus adaptée à ce type de problèmes avec une possible implantation sur une architecture multi-cœur ou GPU (Graphics Processing Unit). Néanmoins, il se peut que les complexités de calcul des projections pour les différentes contraintes ne soient pas les mêmes et un algorithme hybride séquentiel/parallèle permet alor d'optimiser la convergence. Ces algorithmes hybrides portent le nom d'algorithmes par blocs. Pour introduire un de ces algorithmes nous allons nous placer dans un cadre un peu plus large que la formulation du problème 2.28.

Problème 2.29 *Soit* $C = \bigcap_{i=1}^{c} C_i \neq \varnothing$, *où* $(C_i)_{i\in\{1,\dots,c\}}$ *est une famille d'ensembles convexes fermés de* $\mathcal{H}$. *Soit* $f_1 \in \Gamma_0(\mathcal{H})$.

$$\textit{Trouver} \qquad \widehat{u} \in C \quad \textit{tel que} \quad f_1(\widehat{u}) = \inf f_1(C).$$

Ce problème est un cas particulier du problème (2.24) lorsque $J = 2$ et $f_2 = \iota_C$ et il devient équivalent au problème 2.28 lorsque $f_1 = 0$.

L'algorithme itératif par blocs BISGA (Block Iterative Surrogate Gradient Algorithm) proposé dans [Combettes, 2003] est présenté plus bas (algorithme 1) dans le contexte suivant :

Hypothèse 2.30 *Soit $f_1 \in \Gamma_0(\mathcal{H})$ une fonction différentiable et strictement convexe sur $C \subset \mathcal{H}$. On suppose qu'il existe $w \in S \subset \mathcal{H}$ tel que $f_1(w) < +\infty$ et tel que $S = \{v \in \mathcal{H} \mid f_1(v) \leq f_1(w)\}$ est borné (hypothèses assurant l'existence d'un unique minimiseur). De plus, les contraintes sont modélisées par*

$$\big(\forall i \in \{1, \ldots, c\}\big) \qquad C_i = \{u \in \mathcal{H} \mid \varphi_i(u) \leq \delta_i\} \tag{2.27}$$

où pour tout $i \in \{1, \ldots, c\}$, φ_i est une fonction convexe et continue de $\mathcal{H}$ dans $\mathbb{R}$ et δ_i un paramètre réel (les ensembles de contraintes ainsi définis sont des sections inférieures des fonctions $(\varphi_i)_{1 \leq i \leq c}$).

Fixer $\epsilon \in]0, 1/c[$ et $u_0 \in \mathcal{H}$.

Pour $n \in \mathbb{N}$

1- Prendre un ensemble d'indices $I_n \subset \{1, \ldots, c\}$ non vide.

2- Soit $t_{i,n}$ un sous-gradient de φ_i en u_n.

3- Pour $i \in I_n$, $p_{i,n} = \begin{cases} u_n + \frac{\delta_i - \varphi_i(u_n)}{\|t_{i,n}\|} t_{i,n}, & \text{si} \quad \varphi_i(u_n) > \delta_i; \\ u_n, & \text{si} \quad \varphi_i(u_n) \leq \delta_i. \end{cases}$

4- On fixe la suite $(\omega_{i,n})_{i \in I_n}$ variant dans $]\epsilon, 1]$ et telle que $\sum_{i \in I_n} \omega_{i,n} = 1$.

5- $\lambda_n = \begin{cases} \frac{\sum_{i \in I_n} \omega_{i,n} \|p_{i,n} - u_n\|^2}{\|\sum_{i \in I_n} \omega_{i,n} p_{i,n} - u_n\|^2}, & \text{si} \quad u_n \in \bigcap_{i \in I_n} C_i; \\ 1, & \text{sinon.} \end{cases}$

6- $u_n = \arg \min_{u \in D_n \cap E_n} f_1(u)$ avec $\begin{cases} D_n = \{v \in \mathcal{H} \mid \langle u_n - v, \nabla f_1(u_n) \rangle \leq 0\}, \\ E_n = \{v \in \mathcal{H} \mid \langle v - v_n, u_n - v_n \rangle \leq 0\}. \end{cases}$

Algorithme 1: Algorithme BISGA.

L'algorithme BISGA converge vers l'unique solution du problème 2.29 lorsqu'il existe un nombre d'itérations $L > 0$ pour lequel, quelque soit l'itération courante $n \in \mathbb{N}$, $\bigcup_{j=n}^{n+L-1} I_j = \{1, \ldots, c\}$. Cette hypothèse implique que chaque ensemble C_i soit utilisé au moins une fois au cours des L itérations suivantes. La notion d'algorithmes par blocs provient des suites $(I_n)_{n \in \mathbb{N}}$ désignant un groupe de contraintes. L'étape 3 de l'algorithme correspond à une projection sous-différentielle de u_n sur chaque $(C_i)_{i \in I_n}$. Enfin, il est intéressant de remarquer que dans le cas où f_1 est un terme quadratique, l'étape 6 prend une forme explicite. On laissera les lecteurs se reporter à [Combettes, 2003; Miled *et al.*, 2009] pour plus de détails sur cet algorithme, ses simplifications dans le cas quadratique et également son utilisation dans le cas de la stéréoscopie.

Le problème 2.29 nous permet également d'introduire l'algorithme de gradient projeté [Bertsekas, Tsitsiklis, 1997, Chap. 3., Sect.3.3.2][Alber *et al.*, 1998] qui est un algorithme avec un bon taux de convergence lorsque la fonction est de gradient β_1-Lipschitz. Soit $u_0 \in C$, la suite $(u_n)_{n>0}$ est construite de la façon suivante :

$$u_{n+1} = u_n + \lambda_n \Big(P_C \big(u_n - \gamma_n (\nabla f_1(u_n) - u_n) \big) \Big).$$

L'algorithme converge faiblement vers une solution du problème 2.29 lorsque l'on choisit, pour tout $n \in \mathbb{N}$, le paramètre de relaxation $\lambda_n \in]0,1]$ et le pas de l'algorithme $\gamma_n \in]0, 2/\beta_1[$. De la même manière que pour la simple descente de gradient, des stratégies de recherche de pas sont envisageables mais requièrent d'arrêter la recherche linéaire lorsque l'on sort du convexe C. Ce type d'algorithme a récemment été utilisé pour résoudre le problème de déconvolution en présence de bruit gaussien avec une régularisation par variation totale [Chambolle, 2004]. Dans ce travail, Chambolle introduit la variation totale sous forme de contrainte, telle que $C = \{u \in \mathcal{H} \mid \mathrm{tv}(u) \leq \delta\}$ avec $\delta \in]0, +\infty[$, et non sous forme lagrangienne comme nous l'avions fait dans le paragraphe précédent, à savoir,

$$\min_{u \in \mathcal{H}} f_1(u) + \chi \mathrm{tv}(u) \qquad \text{avec } \chi > 0.$$

Le dernier algorithme que nous allons mentionner dans ce paragraphe est l'algorithme de sous-gradient projeté de Polyak [Polyak, 1969]. Cet algorithme permet de résoudre le problème 2.29 dans une formulation générale, ne nécessitant pas d'hypothèse de différentiabilité sur f_1. L'algorithme est initialisé par une valeur $u_0 \in C$ et, pour tout $n \in \mathbb{N}$, t_n désigne un sous-gradient de f_1 en u_n. Les itérations sont obtenues pour tout $n \in \mathbb{N}$ par

$$u_{n+1} = P_C \Big(u_n - \sigma_n \frac{t_n}{\|t_n\|} \Big),$$

où, pour avoir des garanties de convergence, $(\sigma_n)_{n \geq 0}$ est une suite de valeurs strictement positives de somme infinie ($\sum_n \sigma_n = +\infty$) et de carré sommable ($\sum_n \sigma_n^2 < +\infty$). Ces hypothèses impliquent un pas inversement proportionnel au nombre d'itérations, ce qui va se traduire par une convergence lente de l'algorithme. Cet algorithme permet donc de traiter des fonctionnelles convexes non-différentiables de type norme ℓ_1 mais au prix d'une convergence lente.

2.3.4 Conclusion sur les algorithmes

Divers algorithmes permettant de minimiser des critères différentiables ou non-différentiables viennent d'être présentés. Nous récapitulons dans le tableau 2.1 l'ensemble des

avantages et inconvénients de chaque méthode. Dans ce tableau, le fait de gérer des fonctions de gradient non-Lipschitz mais différentiables sur leur domaine concerne directement la divergence de Kullback-Leibler. La non-différentiabilité est liée à la minimisation d'un critère comprenant la norme ℓ_1 (par exemple comme terme de régularisation). La gestion des contraintes permet d'envisager des contraintes comme la positivité des pixels de l'image reconstruite, ou encore s'assurer que la solution appartienne à la boule tv, ce qui reviendrait à formuler le critère (2.7) sous la forme :

$$\min_{y \in \mathbb{R}^N} \psi(y) \qquad \text{s.c.} \quad \phi(y) \leq \delta. \tag{2.28}$$

Un autre élément de comparaison pour ces différents algorithmes est leur possible implantation en parallèle de façon à améliorer leur temps de calcul (important quand les données sont de grandes tailles). Le dernier élément considéré est ainsi la vitesse de convergence.

Le tableau 2.1 montre qu'il est difficile de gérer efficacement un critère incluant simultanément une fonction qui n'est pas de gradient Lipschitz et une norme ℓ_1 avec les méthodes précédemment citées. Lorsqu'il s'agit alors de prendre en compte des données dégradées par un bruit de Poisson, une solution efficace sera de faire appel aux algorithmes proximaux, méthodes permettant de minimiser un critère de la forme (2.24) sous la seule contrainte que, pour tout $i \in \{1, \ldots, J\}$, on soit capable de calculer explicitement la solution de

$$\arg\min_{u \in \mathcal{H}} \frac{1}{2}\|u - v\|^2 + f_i(u),$$

connue sous le nom d'opérateur proximal de la fonction f_i au point $v \in \mathcal{H}$. La notion d'opérateur proximal sera détaillée dans le chapitre 3.

2.4 Bruit de Poisson : quelle méthode adopter ?

La littérature associée aux applications telles que la TEP, l'imagerie SPECT (Single Photon Emission Computed Tomography), l'astronomie gamma, la microscopie confocale ou encore l'imagerie optique, permet d'avoir un large aperçu des méthodes de reconstruction en présence de bruit de Poisson. Nous allons présenter dans ce paragraphe, les principales méthodes utilisées à savoir les approches basées sur le maximum de vraisemblance [Richardson, 1972; Lucy, 1974; Shepp, Vardi, 1982; Lange, Carson, 1984], les approches MAP lorsque le terme d'attache aux données est quadratique [Fessler, Hero, 1995; Bouman, Sauer, 1996; Verhaeghe *et al.*, 2008], et enfin certains travaux permettant de stabiliser la variance des observations tels que les transformées de Anscombe et de Haar-Fisz [Anscombe, 1948; Bar-Lev, Enis, 1988; Fryźlewicz, Nason, 2001; Zhang *et al.*, 2008].

Remarque 2.31 *Dans cette thèse notre intérêt portera sur les approches variationnelles. Il faut néanmoins être conscient que d'autres approches ont été développées dans le cas*

	Gradient	Newton	POCS	Gradient projeté	BISGA	Polyak
Critère différentiable et de gradient Lipschitz	X	X	-	X	X (et strict. convexe)	X
Critère différentiable mais PAS de gradient Lipschitz	-	-	-	-	X (et strict. convexe)	X
Critère non diffé-rentiable	-	-	-	-	X (Si exprimé sous forme de containte)	X
Contraintes avec projecteur explicite	-	-	X	X	X (contrainte de la forme (2.27))	X
Parallé-lisable	-	-	X (variantes parallèles comme ART ou PPM)	-	X	-
Vitesse	Assez lent mais possibilité d'acc. avec méthodes d'adaptation de pas.	Rapide	Vitesse dépendante du nombre de contraintes	Similaire méthode de gradient	Parfois moins rapide que d'autres méthodes avec projecteur explicite	Très lent à cause du choix du pas permettant d'assurer la conv.

TABLE 2.1 – Tentative de comparaison des principaux algorithmes d'optimisation convexe utilisés dans les problèmes d'imagerie.

d'un bruit additif gaussien et plus récemment dans un cadre poissonien. Parmi ces méthodes ont peut citer les approches de type « moyennes non-locales » (Non-local means) [Buades et al.*, 2005; Deledalle* et al.*, 2010] ou encore les approches basées sur l'équivalent du principe de Stein [Stein, 1981] dans le cas du bruit Poisson [Hudson, 1978; Tsui, Press, 1982; Luisier* et al.*, 2010]. Ces méthodes sont actuellement efficaces pour des problèmes de débruitage mais lorsqu'un opérateur de dégradation linéaire rentre en jeu l'extension de ces travaux n'est pas triviale.*

2.4.1 Maximum de vraisemblance

Une première approche, connue sous le nom d'algorithme de Richardson-Lucy dans des applications en microscopie ou en astronomie [Richardson, 1972; Lucy, 1974] et sous le nom d'algorithme EM-ML (Expectation-Maximisation Maximum Likelihood) en TEP [Dempster *et al.*, 1977; Shepp, Vardi, 1982; Lange, Carson, 1984], consiste à maximiser la vraisemblance poissonienne. Le raisonnement pour obtenir les itérations de l'algorithme est le suivant. On cherche l'estimée de $y = (y_i)_{1\leq i\leq N}$ la plus probable notée $\widehat{y}_{MV}$. Ceci revient à résoudre

$$\widehat{y}_{MV} = \arg\max_{y\in\mathbb{R}^N} \log \mu_{Z|Y=y}((z_j)_{1\leq j\leq M})$$

La loi de Poisson est donnée par :

$$\mu_{Z|Y=y}((z_j)_{1\leq j\leq M}) = \prod_{j=1}^{M} \frac{\exp\big(-\alpha\sum_{i=1}^{N} A_{j,i}y_i\big)}{z_j!}(\alpha\sum_{i=1}^{N} A_{j,i}y_i)^{z_j}, \tag{2.29}$$

où $(A_{j,i})_{1\leq j\leq M, 1\leq i\leq N}$ et $(z_j)_{1\leq j\leq M}$ sont définis par l'équation (2.1). Par conséquent,

$$\log \mu_{Z|Y=y}((z_j)_{1\leq j\leq M}) = \sum_{j=1}^{M}\Big\{-\alpha\sum_{i=1}^{N} A_{j,i}y_i + z_j \log\big(\alpha\sum_{i=1}^{N} A_{j,i}y_i\big)\Big\} + \text{constante}.$$

La maximisation de la vraisemblance peut être effectuée par l'algorithme EM. Des explications détaillées sont présentées dans [Shepp, Vardi, 1982, Annexe I]. Les itérations résultantes de l'algorithme EM-ML sont, pour tout $n \in \mathbb{N}$,

$$\big(\forall j \in \{1,\ldots,N\}\big) \qquad y_{j,n+1} = \frac{y_{j,n}}{\alpha\sum_{i=1}^{M} A_{ij}} \sum_{i=1}^{M} A_{ij} \frac{z_i}{\sum_{j=1}^{N} A_{ij}y_j}.$$

La suite $(y_n)_{n\in\mathbb{N}}$ définie précédemment converge vers le maximum de la log vraisemblance poissonienne. Cependant, comme aucun terme de régularisation n'est pris en compte, l'image $\widehat{y}_{MV}$ est une estimée instable, dominée par le bruit. Cependant, un avantage de cette approche provient de la possibilité d'assurer la positivité de la solution. En effet,

cet algorithme est un algorithme multiplicatif avec un facteur multiplicatif positif, ce qui assure que si la première estimée est positive, les suivantes le seront également. Pour plus de détails sur cet algorithme on pourra se référer à [Shepp, Vardi, 1982; Lange, Carson, 1984]. Cet algorithme est néanmoins connu pour posséder des taux de convergence très faibles. Pour cela, des versions accélérées telles que OSEM (Ordered Subsets Expectation Maximization) [Hudson, Larkin, 1994] ont été proposées. L'algorithme OSEM considère à chaque itération un sous-ensemble des données observées. La méthode la plus utilisée consiste à prendre des sous-ensembles consécutifs adaptés. Le raisonnement est similaire à l'implantation de ART proposée dans [Herman, Meyer, 1993].

2.4.2 Approche MAP : traiter le bruit comme s'il sagissait d'un bruit gaussien

L'approche EM-ML permet d'obtenir des résultats corrects mais très éloignés de l'optimum recherché. Pour améliorer ses performances, des approches EM-MAP ont été envisagées mais l'étape de Maximisation de l'EM est alors lourde en temps de calcul. Dans la littérature on peut par contre trouver des approches MAP où le terme d'attache aux données est approximé par une fonction quadratique. On peut se reporter aux nombreux travaux effectués dans cette direction à savoir [Fessler, Hero, 1995; Bouman, Sauer, 1996; Verhaeghe *et al.*, 2008] et les références associées. Généralement, cette approximation conduit à des résultats corrects lorsque les données sont peu dégradées par le bruit de Poisson [Fessler, 1995]. Dans ces approches, le terme de régularisation est souvent une pénalisation quadratique appliquée sur chaque pixel et son voisinage.

Pour envisager des régularisations plus performantes telles que la variation totale ou une norme ℓ_1 appliquée sur des coefficients de trame, il faut faire appel à des techniques d'optimisation convexe plus récentes : les approches proximales [Combettes, Wajs, 2005] ont montré leur efficacité pour des problèmes de restauration en présence de bruit gaussien. Pour traiter des données dégradées par un opérateur linéaire et un bruit de Poisson, il peut être envisagé d'utiliser une approximation quadratique de l'antilog-vraisemblance poissonienne de façon à se ramener à des critères de la forme :

$$\min_{y \in \mathbb{R}^N} \|Ay - z\|^2 + \chi \mathrm{tv}(y), \tag{2.30}$$

ou

$$\min_{x \in \mathbb{R}^K} \|AF^* x - z\|^2 + \chi \|x\|_1, \tag{2.31}$$

2.4.3 Approche MAP : transformées stabilisatrices de variance

• Approximation de Anscombe

Une approche proposée par Donoho dans [Donoho, 1993] consiste à effectuer un prétraitement sur les observations en utilisant la transformée de Anscombe [Anscombe, 1948;

Mäkitalo, Foi, 2010],

$$\mathcal{A}(z) = 2\sqrt{z + \frac{3}{8}}.$$

Ce pré-traitement permet à la statistique du bruit de se rapprocher d'une loi normale centrée de variance 1. Ce constat est illustré par les figures 2.6 et 2.7 pour des paramètres d'échelles fixés à $\alpha = 0.1$ et $\alpha = 0.02$. On remarque que plus la valeur de α est faible, moins la transformation est efficace. Après la transformation, la suite du traitement consiste à considérer les données comme des données gaussiennes.

• Transformée de Haar-Fisz

L'objectif de la transformée de Haar-Fisz [Fryźlewicz, Nason, 2001] est identique à celle de Anscombe. Elle consiste à stabiliser la variance et implique de cette façon que le bruit se rapproche d'un bruit gaussien. La transformée de Haar-Fisz notée $\mathcal{F}$ est obtenue de la façon suivante :

(i) Effectuer une transformée en ondelettes avec des filtres non normalisés de Haar sur l'image $y \in \mathbb{R}^{N_1 \times N_2}$ sur J niveaux, tels que $N_1 = N_2 = 2^J$. A chaque niveau de décomposition $i \in \{1, \ldots, J\}$, modifier les coefficients de détails de la façon suivante :

$$x_o^i = \begin{cases} 0 & \text{si} \quad x_{app.}^i = 0, \\ x_o^i / \sqrt{x_{app.}^i} & \text{sinon} \end{cases}$$

où o représente les coefficients de détails horizontaux, verticaux ou diagonaux.

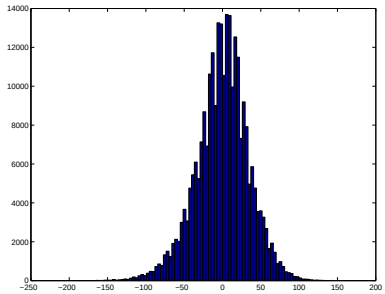
(ii) Effectuer la transformée en ondelettes de Haar inverse sur J niveaux à partir des coefficients transformés à l'étape précédente.

Sur les figures 2.6 et 2.7, sont représentés les histogrammes de bruit pour les différentes transformations. Il apparaît clairement que la transformée de Haar-Fisz fournit une bonne approximation d'une loi normale, et cela quelque soit le niveau de bruit.

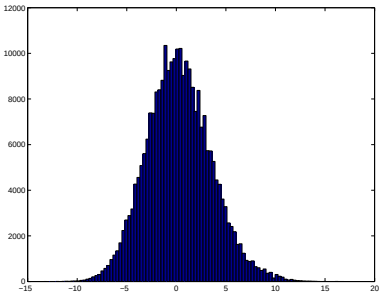
2.4.4 Approches proximales

Grâce à des outils récents d'optimisation convexe que nous présenterons dès le chapitre suivant, des algorithmes comme l'algorithme de Douglas-Rachford [Combettes, Pesquet, 2007a] ou l'algorithme proximal parallèle [Combettes, Pesquet, 2008] permettent, en particulier, de résoudre le problème de débruitage en formulant un critère convexe composé d'une divergence de Kullback-Leibler et d'une régularisation de type norme ℓ_1 sur des coefficients de trame, à savoir,

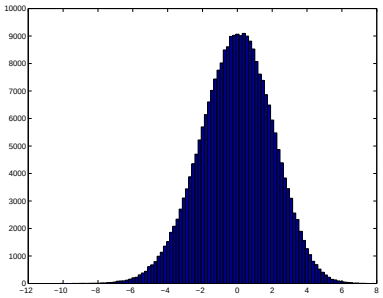
$$\widehat{x} \in \underset{x \in \mathbb{R}^K}{\text{Argmin}}\ D_{\text{KLG}}(z, F^\top x) + \chi \|x\|_1, \tag{2.32}$$



Histogramme de $\overline{y} - z/\alpha$

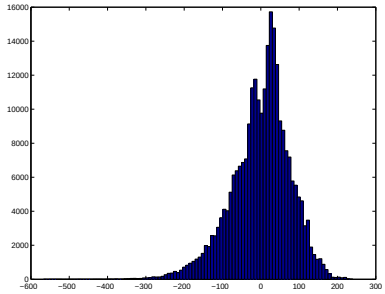


Histogramme de $\mathcal{A}(\overline{y}) - \mathcal{A}(z/\alpha)$

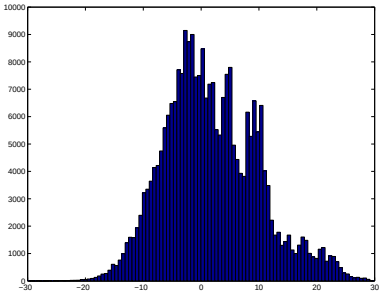


Histogramme de $\mathcal{F}(\overline{y}) - \mathcal{F}(z/\alpha)$

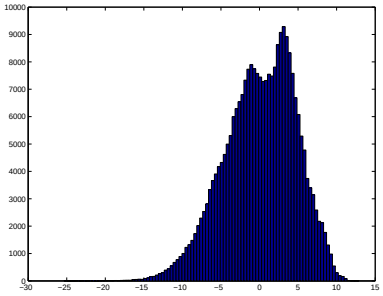
FIGURE 2.6 – Comparaison des histogrammes de bruit pour $\alpha = 0.1$ lorsque $\overline{y}$ correspond à l'image de Lena.



Histogramme de $\overline{y} - z/\alpha$



Histogramme de $\mathcal{A}(\overline{y}) - \mathcal{A}(z/\alpha)$



Histogramme de $\mathcal{F}(\overline{y}) - \mathcal{F}(z/\alpha)$

FIGURE 2.7 – Comparaison des histogrammes de bruit pour $\alpha = 0.02$ lorsque $\overline{y}$ correspond à l'image de Lena.

ou d'une divergence de Kullback-Leibler avec une régularisation par variation totale, c'est à dire,

$$\widehat{y} \in \underset{y \in \mathbb{R}^N}{\text{Argmin}}\ D_{\text{KLG}}(z, y) + \chi \text{tv}(y). \tag{2.33}$$

L'image débruitée par la minimisation du critère (2.32) est $\widehat{y} = F^\top \widehat{x}$. A ce stade, nous nous contentons de montrer quelques résultats de débruitage obtenus avec les algorithmes proximaux, sans en détailler les itérations. Nous comparons ces résultats aux méthodes stabilisatrices de variance et à l'approximation quadratique. Pour ces approches nous minimisons,

$$\widehat{x} \in \underset{x \in \mathbb{R}^K}{\text{Argmin}}\ \|F^\top x - \mathcal{T}(z)\|_2^2 + \chi \|x\|_1, \tag{2.34}$$

ou

$$\widehat{v} = \underset{v \in \mathbb{R}^N}{\arg\min}\ \|v - \mathcal{T}(z)\|_2^2 + \chi \text{tv}(v). \tag{2.35}$$

La notation $\mathcal{T}$ permet d'englober les deux approches stabilisatrices de variance et l'approximation quadratique. En effet, nous avons $\mathcal{T}(z) = \mathcal{A}(z)$ pour la transformée de Anscombe, $\mathcal{T}(z) = \mathcal{F}(z)$ pour la transformée de Haar Fisz et $\mathcal{T}(z) = z$ dans le cas de l'approximation quadratique. Dans notre exemple, F représente une transformée en ondelettes avec des filtres de symmlets de longueur 6 [Mallat, 1997]. D'une part, la solution de l'équation (2.34) revient à effectuer un seuillage doux des coefficients d'ondelettes de $\mathcal{T}(z)$ [Daubechies *et al.*, 2004]. D'autre part, la solution de (2.35) est obtenue en appliquant l'approche de Chambolle sur $\mathcal{T}(z)$. Les images débruitées par le critère (2.34) (resp. du critère (2.35)) sont obtenues par $\widehat{y} = \mathcal{T}^{-1}(F^\top \widehat{x})$ $\big($resp. $\widehat{y} = \mathcal{T}^{-1}(\widehat{v})\big)$.

Pour chaque approche, le paramètre de régularisation χ a été choisi manuellement pour obtenir le plus grand rapport signal sur bruit (SNR). Dans l'ensemble du manuscrit, le SNR mesurera l'erreur entre une image y et l'image originale $\overline{y}$. Il est défini par

$$\text{SNR}(y, \overline{y}) = 20 \log_{10} \frac{\|\overline{y}\|}{\|y - \overline{y}\|}. \tag{2.36}$$

Une autre mesure qui sera fréquemment utilisée est la mesure de similarité de structures (SSIM), introduite dans [Wang, Bovik, 2009]. Ses valeurs sont comprises entre -1 et 1 (le maximum étant obtenu pour deux images identiques). Le programme Matlab est disponible sur la page web de Z. Wang [1].

Les résultats obtenus par les différentes approches sont présentés sur les figures 2.8 et 2.9 où nous pouvons remarquer la nette amélioration des résultats tant en terme de SNR, de SSIM ou visuellement lorsque la divergence de Kullback-Leibler est considérée. Cet exemple vise essentiellement à évaluer la qualité de restauration pour les différents choix de terme d'attache aux données. Cependant, nous pouvons noter que, pour la régularisation par norme ℓ_1 sur les coefficients de trames, de meilleurs résultats auraient pu

1. http ://www.ece.uwaterloo.ca/∼z70wang/research

être obtenus en remplaçant la transformée en ondelettes choisie par des trames telles que des contourlets, des bandelets, des transformées en arbre dual. Notons que pour obtenir de bons résultats en utilisant des trames, le coefficient de régularisation doit être adapté à chaque sous-bande. Lorsqu'un opérateur linéaire est pris en compte, les algorithmes proximaux ne peuvent être utilisés directement lorsque l'attache aux données est une divergence de Kullback-Leibler, cela nous conduira à deux études différentes dans les chapitres 3 et 4.

Remarque 2.32 *Notons que de récents travaux basés sur d'autres approches que les méthodes proximales se sont intéressés à la minimisation d'un critère composé de la divergence de Kullback-Leibler et de la variation totale [Dey* et al.*, 2006; Chan, Chen, 2007; Bardsley, Luttman, 2009].*

2.5 Conclusion

Ce présent chapitre nous a permis d'introduire la problématique et les approches existantes pour l'aborder. Elle permet de dégager trois points importants :

• Nous avons vu que de nombreux algorithmes permettent de résoudre des problèmes de débruitage, restauration et reconstruction en présence de bruit additif gaussien, grâce à un terme d'attache aux données quadratique (fonction de gradient Lipschitz).

• Nous avons montré que pour des données dégradées par un bruit de Poisson impliquant un terme d'attache aux données lié à la divergence de Kullback-Leibler, la propriété de gradient Lipschitz requise pour la convergence de nombreux algorithmes n'est pas vérifiée. Il faut néanmoins noter que la fonctionnelle reste deux fois différentiable sur son domaine.

• Une limitation supplémentaire provient du choix du terme de régularisation. En effet, un terme qui semble adéquat en pratique est une norme ℓ_1 appliquée sur des coefficients de trames. Cependant, dans les algorithmes cités, ceux permettant de traiter des fonctionnelles non-différentiables sont généralement très lents. Une solution consiste à approximer la norme ℓ_1 par une fonction différentiable telle qu'une fonction de Huber [Durand, Nikolova, 2007; Carlavan *et al.*, 2009] mais le choix d'une bonne approximation jouera sur la vitesse de convergence de l'algorithme et la qualité des résultats. Dans la suite nous présenterons des algorithmes proximaux qui constituent des méthodes efficaces permettant de minimiser des critères non-différentiables. Ces algorithmes requièrent néanmoins certaines adaptations pour être appliqués dans le cas d'un bruit poissonien, en particulier lorsque la dégradation comporte également un opérateur linéaire. Enfin, nous avons remarqué que, malgré les bons résultats produits par la régularisation sur des trames d'ondelettes ou une régularisation par variation totale, les résultats obtenus présentent des artefacts. Un point que nous aborderons dans le chapitre 4 est la nécessité de trouver d'autres termes de régularisation permettant d'améliorer la qualité de restauration/reconstruction. Nous présenterons dans ce but, la notion de régularisation hybride.

Image originale

Image dégradée

Solution de (2.34) avec $\mathcal{T} = \mathrm{Id}$
SNR = 18.3 dB et SSIM = 0.61

Solution de (2.34) avec $\mathcal{T} = \mathcal{A}$
SNR = 19.1 dB et SSIM = 0.66

Solution de (2.34) avec $\mathcal{T} = \mathcal{F}$
SNR = 17.8 dB et SSIM = 0.56

Solution de (2.35) avec $\mathcal{T} = \mathrm{Id}$
SNR = 20.3 dB et SSIM = 0.70

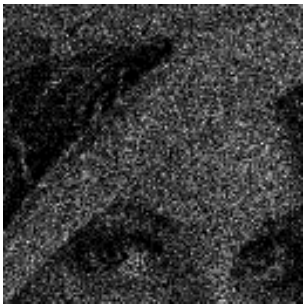
Solution de (2.35) avec $\mathcal{T} = \mathcal{A}$
SNR = 21.0 dB et SSIM = 0.75

Solution de (2.35) avec $\mathcal{T} = \mathcal{F}$
SNR = 19.9 dB et SSIM = 0.67

Solution de (2.33)
SNR = 21.3 dB et SSIM = 0.76

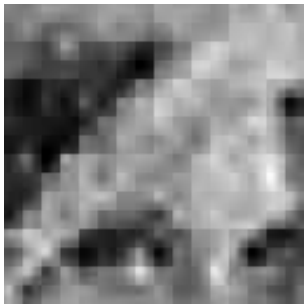
Solution de (2.33)
SNR = 19.7 dB et SSIM = 0.69

FIGURE 2.8 – Comparaison des différentes méthodes de débruitage en présence de bruit de Poisson pour une dégradation avec un paramètre d'échelle fixé à $\alpha = 0.1$.



Solution de (2.34) avec $\mathcal{T} = \mathrm{Id}$
SNR = 15.8 dB et SSIM = 0.46

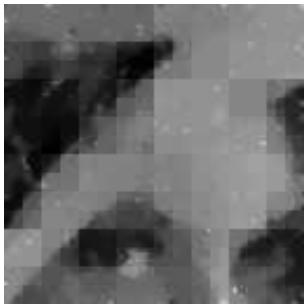
Solution de (2.34) avec $\mathcal{T} = \mathcal{A}$
SNR = 14.5 dB et SSIM = 0.47



Solution de (2.34) avec $\mathcal{T} = \mathcal{F}$
SNR = 14.9 dB et SSIM = 0.40

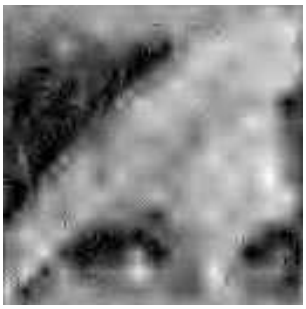
Solution de (2.35) avec $\mathcal{T} = \mathrm{Id}$
SNR = 17.0 dB et SSIM = 0.55

Solution de (2.35) avec $\mathcal{T} = \mathcal{A}$
SNR = 15.7 dB et SSIM = 0.59



Solution de (2.35) avec $\mathcal{T} = \mathcal{F}$
SNR = 16.3 dB et SSIM = 0.50

Solution de (2.33)
SNR = 17.9 dB et SSIM = 0.62



Solution de (2.33)
SNR = 16.4 dB et SSIM = 0.51

FIGURE 2.9 – Comparaison des différentes méthodes de débruitage en présence de bruit de Poisson pour une dégradation avec un paramètre d'échelle fixé à $\alpha = 0.02$.

CHAPITRE 3

Imbrication d'algorithmes proximaux et extension quadratique

3.1 Motivations

Depuis le début des années 2000, de nombreux travaux ont porté sur la minimisation d'un critère convexe composé d'une somme de deux fonctions non-nécessairement différentiables [Bect *et al.*, 2004; Figueiredo, Nowak, 2003; Daubechies *et al.*, 2004; Combettes, Wajs, 2005; Combettes, Pesquet, 2007a]. On peut par exemple citer, l'algorithme EM (Expectation-Maximization) introduit dans [Figueiredo, Nowak, 2003], l'algorithme Landweber seuillé [Bect *et al.*, 2004; Daubechies *et al.*, 2004] également connu sous le nom d'algorithme explicite-implicite dans une forme plus générale [Combettes, Wajs, 2005] et l'algorithme de Douglas-Rachford [Combettes, Pesquet, 2007a]. Ces deux derniers font appel à la notion d'opérateur proximal introduit par Moreau dans [Moreau, 1965] et possèdent des garanties de convergence. Nous définirons dans un premier temps cet opérateur et certaines de ses propriétés qui nous seront utiles dans ce chapitre ainsi que dans le reste du manuscrit. Notre intérêt portera ensuite sur les algorithmes proximaux (explicite-implicite et de Douglas-Rachford) et les propriétés de convergence associées. Le calcul de l'opérateur proximal d'une somme de deux fonctions convexes fera l'objet d'une étude dans le paragraphe suivant. Nous présenterons ensuite la classe des algorithmes proximaux imbriqués que nous avons proposée et leurs propriétés de convergence. Ces méthodes permettent de résoudre une classe de problèmes plus large, à savoir la minimisation d'une somme de trois fonctions. La formulation de cette classe de problèmes est la suivante.

Problème 3.1 *Soient f, g et h des fonctions appartenant à la classe $\Gamma_0(\mathcal{H})$ où g est une fonction de gradient β-Lipschitz avec $\beta > 0$. On souhaite trouver*

$$\widehat{u} \in \underset{u \in \mathcal{H}}{\operatorname{Argmin}}\ f(u) + g(u) + h(u).$$

Le troisième fonctionnelle permet par exemple d'inclure dans le critère, en plus du terme d'attache aux données et d'un terme de régularisation, des contraintes additionnelles telles qu'une contrainte de positivité des données ou une contrainte sur la dynamique des pixels. Dans le paragraphe 3.6, cette approche nous permettra entre autres de résoudre

un problème inverse où l'image est dégradée par un opérateur linéaire et un bruit non-nécessairement additif gaussien, comme un bruit de Poisson. Dans les exemples étudiés dans le paragraphe 3.6, h se réduira à la fonction indicatrice d'un ensemble convexe ce qui nous amènera à considérer un cas particulier du problème 3.1, à savoir :

Problème 3.2 *Soient f et g deux fonctions appartenant à la classe $\Gamma_0(\mathcal{H})$ où g est une fonction de gradient β-Lipschitz avec $\beta > 0$. Soit $C \subset \mathcal{H}$ un ensemble convexe fermé non vide. On souhaite trouver*

$$\widehat{u} \in \underset{u \in \mathcal{H}}{\operatorname{Argmin}}\ f(u) + g(u) + \iota_C(u).$$

3.2 Opérateur proximal

3.2.1 Définition

L'opérateur proximal est associé à une fonction $f \in \Gamma_0(\mathcal{H})$ en un point $u \in \mathcal{H}$ et se note $\operatorname{prox}_f u$. Il correspond à l'unique point qui minimise $f + \|\cdot - u\|^2$, *i.e.*

$$\operatorname{prox}_f \,:\, \mathcal{H} \to \mathcal{H} \,:\, u \mapsto \arg\min_{v \in \mathcal{H}} \frac{1}{2}\|v - u\|^2 + f(v). \tag{3.1}$$

Cet opérateur généralise la notion de projection sur un ensemble convexe fermé non vide $C \subset \mathcal{H}$ telle que $\operatorname{prox}_{\iota_C} = P_C$. De nombreuses formes explicites d'opérateurs proximaux sont répertoriées dans de récents travaux [Combettes, Wajs, 2005; Chaux *et al.*, 2007; Combettes, Pesquet, 2007a; Combettes, Pesquet, 2010]. Une forme bien connue de cet opérateur proximal est celle associée à une norme ℓ_1 (antilog vraisembance d'une loi de Laplace), qui est rappelée ci-dessous.

Exemple 3.3 *Soit $\chi \in]0, +\infty[$ et soit $\varphi = \chi|\cdot|$. On a alors, pour tout $\xi \in \mathbb{R}$,*

$$\operatorname{prox}_\varphi \xi = \begin{cases} \xi - \chi & \textit{si} \quad \xi > \chi \\ 0 & \textit{si} \quad \xi \in [-\chi, \chi] \\ \xi + \chi & \textit{si} \quad \xi < -\chi \end{cases}$$

Cet opérateur proximal correspond à un seuillage doux, c'est à dire, pour tout $\xi \in \mathbb{R}$, $\operatorname{prox}_\varphi \xi = \operatorname{soft}_{[-\chi,\chi]}\xi = \operatorname{sign}(\xi)\max\{|\xi| - \chi, 0\}$.

D'autres formes explicites sont données pour des distributions gaussienne généralisée, exponentielle, gamma, chi, uniforme et, plus généralement, pour des densités de probabilité log-concaves [Chaux *et al.*, 2007]. Dans la suite de ce chapitre, seules les propriétés de cet opérateur qui sont nécessaires à la bonne compréhension du manuscrit seront rappelées.

3.2.2 Quelques propriétés

Dans ce paragraphe nous listons une série de propriétés liées aux opérateurs proximaux. Ces propriétés vont du calcul d'un opérateur proximal pour une fonction composée avec un opérateur linéaire, utile dans des problèmes de restauration d'images, à la propriété de contraction de l'opérateur proximal, nécessaire dans les preuves de convergence des algorithmes proximaux. Nous nous attarderons également sur le calcul de l'opérateur proximal d'une somme de deux fonctions dont l'une est la fonction indicatrice d'un ensemble convexe.

• **Points fixes** [Chaux *et al.*, 2007, lemme 2.3(i)] :
Soit $f \in \Gamma_0(\mathcal{H})$, pour tout $u \in \mathcal{H}$, on a alors

$$u \in \operatorname{Argmin} f \Leftrightarrow \operatorname{prox}_f u = u. \tag{3.2}$$

• **Perturbation quadratique** [Combettes, Wajs, 2005, lemme 2.6(i)] :
Soit $f = h + \vartheta\|\cdot\|^2/2 + \chi\langle\cdot, \widetilde{v}\rangle + \varepsilon$ avec $h \in \Gamma_0(\mathcal{H})$, $\widetilde{v} \in \mathcal{H}$, $\vartheta \in [0, +\infty[$ et $(\chi, \varepsilon) \in \mathbb{R}^2$. Pour tout $u \in \mathcal{H}$, on a alors

$$\operatorname{prox}_f u = \operatorname{prox}_{h/(\vartheta+1)}\big((u - \chi\widetilde{v})/(\vartheta + 1)\big). \tag{3.3}$$

• **Translation** [Combettes, Wajs, 2005, lemme 2.6(ii)] :
Soit $f = h(\cdot - \widetilde{v})$ avec $h \in \Gamma_0(\mathcal{H})$ et $\widetilde{v} \in \mathcal{H}$. Pour tout $u \in \mathcal{H}$, on a alors

$$\operatorname{prox}_f u = \widetilde{v} + \operatorname{prox}_h(u - \widetilde{v}). \tag{3.4}$$

• **Changement d'échelle** [Combettes, Wajs, 2005, lemme 2.6(iii)] :
Soit $f = h(./\chi)$ avec $h \in \Gamma_0(\mathcal{H})$ et $\chi \in \mathbb{R} \smallsetminus 0$. Pour tout $u \in \mathcal{H}$, on a alors

$$\operatorname{prox}_f u = \chi \operatorname{prox}_{h/\chi^2}(u/\chi). \tag{3.5}$$

• **Contraction et « moyenne α »** [Combettes, Wajs, 2005, Définitions 1.1 et 2.3] :
Nous rappelons ici les définitions et quelques propriétés d'un opérateur contractant, d'un opérateur de « moyenne α », d'un opérateur fermement contractant et strictement contractant.

Définition 3.4 *Un opérateur $L : \mathcal{H} \to \mathcal{H}$ est* ***contractant*** *si*

$$(\forall (u,v) \in \mathcal{H}^2) \quad \|Lu - Lv\| \leq \|u - v\|.$$

Définition 3.5 *L est dit de* ***« moyenne α »*** *(α-averaged) si $L = (1-\alpha)\mathrm{Id} + \alpha R$ pour un certain opérateur contractant $R : \mathcal{H} \to \mathcal{H}$. La classe des opérateurs de « moyenne α » est notée $\mathcal{A}(\alpha)$.*

Lemme 3.6 [Combettes, 2004, lemme 2.3] *Soient $L : \mathcal{H} \to \mathcal{H}$ et $\vartheta \in \ ]0, +\infty[$ tels que $\vartheta L \in \mathcal{A}\big(\frac{1}{2}\big)$, et soit $\gamma \in]0, 2\vartheta[$. On a alors $\mathrm{Id} - \gamma L \in \mathcal{A}\big(\frac{\gamma}{2\vartheta}\big)$.*

Lemme 3.7 [Combettes, 2004, lemme 2.1(i) et 2.1(ii)] *Soient $L : \mathcal{H} \to \mathcal{H}$ et $\alpha \in]0, 1[$. Les propriétés suivantes sont alors équivalentes.*

(i) $L \in \mathcal{A}(\alpha)$.

(ii) $(\forall (u,v) \in \mathcal{H}^2)\ \|Lu - Lv\|^2 \leq \|u - v\|^2 - \frac{1-\alpha}{\alpha}\|(\mathrm{Id} - L)u - (\mathrm{Id} - L)v\|^2$.

Définition 3.8 *L'opérateur L est* ***fermement contractant*** *s'il satisfait l'une des conditions équivalentes suivantes :*

(i) $(\forall (u,v) \in \mathcal{H}^2)\ \|Lu - Lv\|^2 \leq \big\langle Lu - Lv, u - v\big\rangle$.

(ii) $(\forall (u,v) \in \mathcal{H}^2)\ \|Lu - Lv\|^2 \leq \|u - v\|^2 - \|(\mathrm{Id} - L)u - (\mathrm{Id} - L)v\|^2$.

On peut remarquer que l'opérateur L est fermement contractant si et seulement si $L \in \mathcal{A}\big(\frac{1}{2}\big)$.

Lemme 3.9 [Combettes, Wajs, 2005, lemme 2.4] *Soit $f \in \Gamma_0(\mathcal{H})$. prox_f et $\mathrm{Id} - \mathrm{prox}_f$ sont alors fermement contractant.*

Définition 3.10 *Un opérateur linéaire $L : \mathcal{H} \to \mathcal{H}$ est* ***strictement contractant*** *de constante $\vartheta \in]0, 1[$ si*

$$(\forall (u,v) \in \mathcal{H}^2) \qquad \|Lu - Lv\| \leq \vartheta\|u - v\|.$$

Proposition 3.11 *Soit f une fonction* ***fortement convexe*** *de module $\vartheta \in \ ]0, +\infty[$, i.e. $f = h + \vartheta \ \|\cdot\|^2/2$ où $h \in \Gamma_0(\mathcal{H})$, alors :*

(i) $(\forall (u,v) \in \mathcal{H}^2)\ \big\langle \mathrm{prox}_f u - \mathrm{prox}_f v, u - v\big\rangle \geq (1+\vartheta)\|\mathrm{prox}_f u - \mathrm{prox}_f v\|^2$

(ii) prox_f *est strictement contractant de constante* $(1+\vartheta)^{-1}$.

<u>Preuve</u> :
La fonction $h/(1+\vartheta)$ appartient à $\Gamma_0(\mathcal{H})$ ce qui implique, d'après le lemme 3.9, que $\mathrm{prox}_{(1+\vartheta)^{-1}h}$ est fermement contractant , i.e.

$$(\forall (u,v) \in \mathcal{H}^2) \qquad \|\mathrm{prox}_{\frac{h}{1+\vartheta}} u - \mathrm{prox}_{\frac{h}{1+\vartheta}} v\|^2 \leq \big\langle \mathrm{prox}_{\frac{h}{1+\vartheta}} u - \mathrm{prox}_{\frac{h}{1+\vartheta}} v, u - v\big\rangle.$$

Par conséquent, en utilisant (3.3) avec $\chi = \varepsilon = 0$, nous obtenons

$$\begin{aligned}(\forall (u,v) \in \mathcal{H}^2) \qquad & \big\langle \operatorname{prox}_f u - \operatorname{prox}_f v, u - v \big\rangle \\ =&(1+\vartheta)\left\langle \operatorname{prox}_{\frac{h}{1+\vartheta}}\left(\frac{u}{1+\vartheta}\right) - \operatorname{prox}_{\frac{h}{1+\vartheta}}\left(\frac{v}{1+\vartheta}\right), \frac{u}{1+\vartheta} - \frac{v}{1+\vartheta}\right\rangle \\ \geq&(1+\vartheta)\left\|\operatorname{prox}_{\frac{h}{1+\vartheta}}\left(\frac{u}{1+\vartheta}\right) - \operatorname{prox}_{\frac{h}{1+\vartheta}}\left(\frac{v}{1+\vartheta}\right)\right\|^2 \\ =&(1+\vartheta)\|\operatorname{prox}_f u - \operatorname{prox}_f v\|^2.\end{aligned}$$

La propriété (ii) peut alors être déduite en appliquant l'inégalité de Cauchy-Schwarz :

$$\begin{aligned}(\forall (u,v) \in \mathcal{H}^2) \qquad (1+\vartheta)\|\operatorname{prox}_f u - \operatorname{prox}_f v\|^2 &\leq \big\langle \operatorname{prox}_f u - \operatorname{prox}_f v, u - v \big\rangle \\ &\leq \|\operatorname{prox}_f u - \operatorname{prox}_f v\|\|u - v\|.\square\end{aligned}$$

• **Composition** :
La propriété suivante concerne le calcul de l'opérateur proximal d'une fonction f de $\Gamma_0(\mathcal{H})$ composée avec un opérateur linéaire borné L vérifiant $L \circ L^* = \chi \operatorname{Id}$ pour $\chi > 0$. On reconnaît ainsi la propriété caractéristique de l'opérateur de synthèse d'une trame ajustée qui a été introduite dans le chapitre 2 [équation (2.19)].

Proposition 3.12 [Combettes, Pesquet, 2007a, proposition 11] *Soit $\mathcal{G}$ un espace de Hilbert réel, soit f une fonction de $\Gamma_0(\mathcal{G})$ et soit $L\colon \mathcal{H} \to \mathcal{G}$ un opérateur linéaire borné. Supposons que la composition de L et L^* satisfait $L \circ L^* = \chi \operatorname{Id}$, pour $\chi \in \,]0, +\infty[$, on a alors : $f \circ L \in \Gamma_0(\mathcal{H})$ et*

$$\operatorname{prox}_{f \circ L} = \operatorname{Id} + \chi^{-1} L^* \circ (\operatorname{prox}_{\chi f} - \operatorname{Id}) \circ L. \tag{3.6}$$

Notons qu'une généralisation de cette proposition sera présentée dans le chapitre 4 [proposition 4.3].

• **Formule de décomposition** :
Nous rappelons ici une proposition utile au calcul de l'opérateur proximal d'une fonction séparable.

Proposition 3.13 [Chaux *et al.*, 2007, proposition 2.10] *Soit $\mathbb{I}$ un sous-ensemble non vide de $\mathbb{N}$, soit $(o_i)_{i\in\mathbb{I}}$ une base orthonormale de $\mathcal{H}$ et soit*

$$f \,:\, \mathcal{H} \to \,]-\infty, +\infty]\,;\, u \mapsto \sum_{i\in\mathbb{I}} \varphi_i(\langle u|o_i\rangle).$$

Si $\mathbb{I}$ est fini, ou s'il existe un sous-ensemble $\mathbb{J}$ de $\mathbb{I}$ tel que :

(i) $\mathbb{I} \smallsetminus \mathbb{J}$ *est fini,*
(ii) $(\forall i \in \mathbb{J})$ $\varphi_i \geq 0$,
(iii) il existe une suite $(\xi_i)_{i\in\mathbb{J}}$ *dans* $\mathbb{R}$ *telle que* $\sum_{i\in\mathbb{J}}|\xi_i|^2 < +\infty$, $\sum_{i\in\mathbb{J}}|\mathrm{prox}_{\varphi_i}\xi_i|^2 < +\infty$ *et* $\sum_{i\in\mathbb{J}}\varphi_i(\xi_i) < +\infty$ *;*

alors $f \in \Gamma_0(\mathcal{H})$ *et*

$$(\forall u \in \mathcal{H}) \qquad \mathrm{prox}_f u = \sum_{i\in\mathbb{I}} \big(\mathrm{prox}_{\varphi_i}\langle u|o_i\rangle\big) o_i.$$

- **Somme d'une fonction de $\Gamma_0(\mathcal{H})$ et d'une fonction indicatrice** :

Dans le paragraphe 3.4, notre attention portera sur le calcul de l'opérateur proximal de la somme d'une fonction indicatrice et d'une fonction de $\Gamma_0(\mathcal{H})$. Nous rappelons ici, que sous certaines limitations ce calcul est direct.

Proposition 3.14 [Combettes, Pesquet, 2007a, proposition 12] *Soit* $f \in \Gamma_0(\mathcal{H})$ *et soit* C *un sous-ensemble convexe de* $\mathcal{H}$ *tel que* $C \cap \mathrm{dom}\, f \neq \varnothing$, *alors :*

(i) $(\forall u \in \mathcal{H}),\ \mathrm{prox}_f u \in C \Rightarrow \mathrm{prox}_{\iota_C+f} u = \mathrm{prox}_f u$.

(ii) *Supposons que* $\mathcal{H} = \mathbb{R}$, *on a alors*

$$\mathrm{prox}_{\iota_C+f} = P_C \circ \mathrm{prox}_f. \tag{3.7}$$

Nous nous proposons de généraliser cette proposition de la manière suivante.

Proposition 3.15 *Soit* $\mathbb{I}$ *un sous-ensemble non vide de* $\mathbb{N}$, *soit* $(o_i)_{i\in\mathbb{I}}$ *une base orthonormale de* $\mathcal{H}$ *et* $(\varphi_i)_{i\in\mathbb{I}}$ *une famille de fonctions de* $\Gamma_0(\mathbb{R})$.
Soit f *une* ***fonction séparable*** *telle que,*

$$f\colon \mathcal{H} \to\ ]-\infty,+\infty] : u \mapsto \sum_{i\in\mathbb{I}} \varphi_i(\langle u, o_i\rangle). \tag{3.8}$$

Soit C un ***convexe séparable*** *tel que,*

$$C = \bigcap_{i\in\mathbb{I}} \{u \in \mathcal{H} \ \mid\ \langle u, o_i\rangle \in C_i\} \tag{3.9}$$

où les $(C_i)_{i\in\mathbb{I}}$ *sont des intervalles fermés de* $\mathbb{R}$ *tels que* $(\forall i \in \mathbb{I})$ $C_i \cap \mathrm{dom}\,\varphi_i \neq \varnothing$.
Si $\mathbb{I}$ *est fini ou s'il existe un sous-ensemble* $\mathbb{J}$ *de* $\mathbb{I}$ *tel que :*

(i) $\mathbb{I} \smallsetminus \mathbb{J}$ *soit fini ;*

(ii) $(\forall i \in \mathbb{J})\ \varphi_i \geq \varphi_i(0) = 0$ *et* $0 \in C_i$ *;*

alors,

$$(\forall u \in \mathcal{H}) \quad \operatorname{prox}_{f+\iota_C} u = \sum_{i \in \mathbb{I}} \pi_i o_i \tag{3.10}$$

où

$$\pi_i = \begin{cases} \inf C_i & \textit{si } \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle < \inf C_i \\ \sup C_i & \textit{si } \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle > \sup C_i \\ \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle & \textit{sinon.} \end{cases} \tag{3.11}$$

<u>Preuve</u> :
Cette proposition se déduit en partie de la proposition 3.13. Assurons-nous que les hypothèses de la proposition 3.13 sont vérifiées. Les formes prises par f et C, nous permettent d'écrire,

$$(\forall u \in \mathcal{H}) \qquad \big(f + \iota_C\big)(u) = \sum_{i \in \mathbb{I}} (\varphi_i + \iota_{C_i})(\langle u, o_i\rangle).$$

De plus, pour tout $i \in \mathbb{I}$, $\varphi_i \in \Gamma_0(\mathbb{R})$ et C_i représente un intervalle fermé de $\mathbb{R}$ dont l'intersection avec $\operatorname{dom}\varphi_i$ est non vide. Par conséquent $\varphi_i + \iota_{C_i}$ est propre et donc $\varphi_i + \iota_{C_i} \in \Gamma_0(\mathbb{R})$. Dans le cas où $\mathbb{I}$ n'est pas un ensemble fini, en considérant l'hypothèse (ii), nous pouvons écrire, pour tout $i \in \mathbb{J}$, $\varphi_i + \iota_{C_i} \geq (\varphi_i + \iota_{C_i})(0) = 0$. Puis, pour tout $i \in \mathbb{J}$, en choisissant $\xi_i = 0$ nous avons $\sum_{i \in \mathbb{J}} |\xi_i|^2 = 0 < +\infty$; d'après les hypothèses (i) et (ii), nous pouvons écrire $\sum_{i \in \mathbb{J}} |(\varphi_i + \iota_{C_i})(\xi_i)|^2 = 0 < +\infty$ et enfin grâce à l'équation (3.2) et l'hypothèse (ii), nous déduisons que $\sum_{i \in \mathbb{J}} |\operatorname{prox}_{\varphi_i + \iota_{C_i}} \xi_i|^2 = 0 < +\infty$. Par conséquent,

$$(\forall u \in \mathcal{H}) \qquad \operatorname{prox}_{f+\iota_C}(u) = \sum_{i \in \mathbb{I}} \big(\operatorname{prox}_{\varphi_i + \iota_{C_i}}\langle u, o_i\rangle\big)\ o_i. \tag{3.12}$$

D'autre part, pour tout $i \in \mathbb{I}$, C_i est un intervalle fermé de $\mathbb{R}$ tel que $C_i \cap \operatorname{dom}\varphi_i \neq \varnothing$. Grâce à la proposition 3.14 (ii), nous pouvons finalement écrire :

$$\begin{aligned} \operatorname{prox}_{\varphi_i + \iota_{C_i}}\langle u, o_i\rangle =& (P_{C_i} \circ \operatorname{prox}_{\varphi_i})(\langle u, o_i\rangle) \\ =& \begin{cases} \inf C_i, & \text{si } \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle < \inf C_i \\ \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle, & \text{si } \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle \in C_i \\ \sup C_i, & \text{si } \operatorname{prox}_{\varphi_i}\langle u, o_i\rangle > \sup C_i. \end{cases} \end{aligned} \tag{3.13}$$

Pour finir, si l'on combine (3.12) et (3.13) nous obtenons les équations (3.10) et (3.11).□

Il faut remarquer que cette proposition est uniquement valide lorsque f **et** C **sont séparables dans la même base orthonormale**. Lorsque ces hypothèses restrictives ne sont plus respectées, le résultat de cette proposition n'a plus lieu d'être vérifié. Pour illustrer cette remarque nous donnons les deux contre-exemples suivants.

Exemple 3.16 *Soit $\mathcal{H} = \mathbb{R}^2$ et soit f une fonction définie pour tout vecteur u de $\mathbb{R}^2$ par $f(u) = \frac{1}{2}u^\top \Lambda\, u$ avec $\Lambda = \begin{pmatrix} 1 & \Lambda_{1,2} \\ \Lambda_{1,2} & \Lambda_{2,2} \end{pmatrix}$ où $\Lambda_{2,2} \geq 0$ et $|\Lambda_{1,2}| \leq \Lambda_{2,2}^{1/2}$. Soit $C = [-1,1]^2$. Cet ensemble convexe est séparable dans la base canonique de $\mathbb{R}^2$. Prenons maintenant $u = 2(\Lambda_{1,2}, 1 + \Lambda_{2,2})^\top$. Après quelques calculs nous obtenons que :*

- $P_C(\mathrm{prox}_f u) = (0,1)^\top$
- $\mathrm{prox}_{\iota_C + f} u = (\pi, 1)^\top$ *où*

$$\pi = \begin{cases} \frac{\Lambda_{1,2}}{2} & \textit{si } \Lambda_{1,2} \in [-2,2] \\ 1 & \textit{si } \Lambda_{1,2} > 2 \\ -1 & \textit{si } \Lambda_{1,2} < -2. \end{cases} \tag{3.14}$$

Dans le cas où $\Lambda_{1,2} \neq 0$, f est alors non séparable et la proposition 3.15 n'est pas vérifiée.

Preuve :
Soit $p = \mathrm{prox}_f u$ et $q = \mathrm{prox}_{f+\iota_C} u$ où $u \in \mathcal{H}$. Soit $h \in \Gamma_0(\mathcal{H})$ une fonction définie, pour tout $v \in \mathcal{H}$, par $h(v) = \frac{1}{2}\|v - u\|^2 + f(v)$. Nous en déduisons que

$$p = (1 + \Lambda)^{-1} u, \tag{3.15}$$

est le minimiseur de h sur $\mathcal{H}$, tandis que q est le minimiseur de h sur C. Nous pouvons alors écrire, pour tout $v \in \mathcal{H}$,

$$\begin{aligned} h(v) &= \frac{1}{2}\Big(\|v\|^2 - 2v^\top u + \|u\|^2\Big) + \frac{1}{2} v^\top \Lambda v \\ &= \frac{1}{2}\Big(v^\top(\Lambda + \mathrm{Id})v + 2v^\top u\Big) + \frac{1}{2}\|u\|^2 \\ &= \frac{1}{2}\Big(v^\top(\Lambda + \mathrm{Id})v + 2v^\top u + u^\top(\mathrm{Id} + \Lambda)^{-1}u\Big) + \frac{1}{2}\|u\|^2 - u^\top(\mathrm{Id} + \Lambda)^{-1}u \\ &= \frac{1}{2}(v - (1+\Lambda)^{-1}u)^\top(\mathrm{Id} + \Lambda)(v - (1+\Lambda)^{-1}u) + \frac{1}{2}\|u\|^2 - u^\top(\mathrm{Id} + \Lambda)^{-1}u \\ &= \overline{h}(v) + g_u, \end{aligned}$$

avec $\overline{h}(v) = \frac{1}{2}(v - p)^\top(\mathrm{Id} + \Lambda)(v - p)$ et g_u une fonction de u. q minimise alors $\overline{h}$ sur C.

Dans l'exemple, nous avons choisi $u = 2(\Lambda_{1,2}, 1 + \Lambda_{2,2})^\top$, qui conduit à $p = (0,2)^\top$. De plus, nous avons choisi $C = [-1,1]^2$, on a alors $P_C(p) = (0,1)^\top$.

Soit $\overline{q} = (\pi, 1)^\top$. Pour montrer que $q = \overline{q}$, nous devons vérifier que $\overline{q}$ minimise $\overline{h}$ sur C. Une condition nécessaire et suffisante pour le montrer [Hiriart-Urruty, Lemaréchal, 1996, p. 293, théorème 1.1.1] est que

$$(\forall v \in C) \qquad \big(\nabla \overline{h}(\overline{q})\big)^\top (v - \overline{q}) \geq 0,$$

où $\nabla \overline{h}(\overline{q}) = (\mathrm{Id} + \Lambda)(\overline{q} - p)$ est le gradient de $\overline{h}$ au point $\overline{q}$. Cela est équivalent à prouver que

$$(\forall (v_1, v_2)^\top \in C) \qquad (2\pi - \Lambda_{1,2})(v_1 - \pi) + (\Lambda_{1,2}\pi - \Lambda_{2,2} - 1)(v_2 - 1) \geq 0. \quad (3.16)$$

Trois cas doivent être considérés.

(i) Le cas où $\Lambda_{1,2} < -2$ implique, lorsque $(v_1, v_2)^\top \in C$, $v_1 \geq -1 = \pi$ et $v_2 \leq 1$. De plus, nous avons $2\pi - \Lambda_{1,2} = -2 - \Lambda_{1,2} > 0$ et $\Lambda_{2,2} - \Lambda_{1,2}^2 \geq 0$, ce qui nous permet d'écrire $\Lambda_{1,2}\pi - \Lambda_{2,2} - 1 \leq -\Lambda_{1,2}^2 - \Lambda_{1,2} - 1 < 0$. L'équation (3.16) est donc vérifiée.

(ii) Le cas où $\Lambda_{1,2} > 2$ fait intervenir des arguments similaires au cas précédent.

(iii) Le cas $\Lambda_{1,2} \in [-2, 2]$ implique $2\pi - \Lambda_{1,2} = 0$ et

$$\Lambda_{1,2}\pi - \Lambda_{2,2} - 1 = \frac{\Lambda_{1,2}^2}{2} - \Lambda_{2,2} - 1 \leq -\frac{\Lambda_{1,2}^2}{2} - 1 \leq 0.$$

L'équation (3.16) est donc satisfaite.

Cela conduit à l'expression de l'exemple 3.16. □

Exemple 3.17 *Soit $\mathcal{H} = \mathbb{R}^2$. Considérons cette fois-ci la fonction séparable $\overline{f}$ définie par $(\forall u = (u_1, u_2)^\top \in \mathbb{R}^2)$ $\overline{f}(u) = (1 + \Lambda_{1,2})(u_1)^2 + (1 - \Lambda_{1,2})(u_2)^2$ où $0 < |\Lambda_{1,2}| \leq 1$. Soit $\overline{C}$ un convexe non séparable défini par*

$$\overline{C} = \{u = (u_1, u_2)^\top \in \mathbb{R}^2 \ \mid \ \max(|u_1 - u_2|, |u_1 + u_2|) \leq \sqrt{2}\}.$$

Dans ce cas, la propriété (3.7) *n'est également pas vérifiée.*

Preuve :
On définit la matrice de rotation

$$R = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}. \quad (3.17)$$

La fonction $\overline{f}$ peut être exprimée comme

$$(\forall u \in \mathbb{R}^2) \qquad \overline{f}(u) = f(Ru),$$

où $f(u) = \frac{1}{2}u^\top \Lambda u$ avec

$$\Lambda = \begin{pmatrix} 1 & \Lambda_{1,2} \\ \Lambda_{1,2} & 1 \end{pmatrix}.$$

De plus,

$$\overline{C} = \{u \in \mathbb{R}^2 \ \mid \ Rx \in [-1,1]^2\} = R^\top [-1,1]^2.$$

On définit $C = R\overline{C}$. Notons que $C = [-1,1]^2$ est l'ensemble convexe séparable considéré dans l'exemple 3.16, alors que f apparaît comme un cas particulier de la classe des fonctions quadratiques considérée dans ce même exemple (en posant $\Lambda_{2,2} = 1$).
L'opérateur proximal de $\overline{f}$ est alors

$$\begin{aligned}
(\forall u \in \mathcal{H}) \qquad \operatorname{prox}_{\overline{f}} u &= \arg\min_{v \in \mathcal{H}} \frac{1}{2}\|v-u\|^2 + \overline{f}(v) \\
&= \arg\min_{v \in \mathcal{H}} \frac{1}{2}\|v-u\|^2 + f(Rv) \\
&= R^\top \arg\min_{\overline{v} \in \mathcal{H}} \frac{1}{2}\|R^\top \overline{v} - u\|^2 + f(\overline{v}) \\
&= R^\top \arg\min_{\overline{v} \in \mathcal{H}} \frac{1}{2}\|R^\top (\overline{v} - Ru)\|^2 + f(\overline{v}) \\
&= R^\top \arg\min_{\overline{v} \in \mathcal{H}} \frac{1}{2}\|\overline{v} - Ru\|^2 + f(\overline{v}) \\
&= R^\top \operatorname{prox}_f(Ru)
\end{aligned}$$

Par des calculs similaires on en déduit que

$$(\forall u \in \mathcal{H}) \qquad P_{\overline{C}}(\operatorname{prox}_{\overline{f}} u) = R^\top P_C\big(\operatorname{prox}_f(Ru)\big)$$

et

$$(\forall u \in \mathcal{H}) \qquad \operatorname{prox}_{\overline{f}+\iota_{\overline{C}}} u = R^\top \operatorname{prox}_{f+\iota_C}(Ru).$$

Donc, si $u = 2R^\top(\Lambda_{1,2}, 2)^\top$, nous déduisons de l'exemple 3.16 que $P_{\overline{C}}(\operatorname{prox}_{\overline{f}} u) = \frac{1}{\sqrt{2}}(1,0)^\top$ et $\operatorname{prox}_{\overline{f}+\iota_{\overline{C}}} u = \frac{1}{\sqrt{2}}(\pi+1, 1-\pi)^\top$, où l'expression de π est donnée par (3.14). Nous pouvons conclure que $P_{\overline{C}}(\operatorname{prox}_{\overline{f}}) \neq \operatorname{prox}_{\overline{f}+\iota_{\overline{C}}} u$. □

Pour résumer, si nous prenons une fonction de $\Gamma_0(\mathcal{H})$ et un ensemble convexe fermé non vide de $\mathcal{H}$, il est possible d'avoir une forme explicite de l'opérateur proximal de la somme de f et de l'indicatrice du convexe. Nous verrons que, dans le cadre de nos travaux, la fonction f et le convexe sont rarement séparables dans une même base orthonormale, et qu'il faut alors faire appel à des algorithmes itératifs.

3.3 Algorithmes pour minimiser une somme de deux fonctions convexes

Après cette brève introduction sur les opérateurs proximaux, revenons au problème de minimisation d'un critère convexe non différentiable composé d'une somme de deux fonctions où l'une d'entre elles peut être ou non de gradient β-Lipschitz. Dans les paragraphes suivants, deux algorithmes permettant de résoudre efficacement ce problème seront présentés. Le premier, nommé algorithme explicite-implicite, permet de résoudre ce problème dans le cas où l'une des deux fonctions est différentiable de gradient β-Lipschitz et il requiert le calcul de l'opérateur proximal de la deuxième fonction. Le second algorithme requiert le calcul des opérateurs proximaux pour les deux fonctions considérées. Pour faciliter la lecture, les fonctions associées à l'algorithme explicite-implicite seront notées f_1 et f_2, celles intervenant dans l'algorithme de Douglas-Rachford seront désignées par g_1 et g_2.

3.3.1 Algorithme explicite-implicite (FB : forward-backward)

Dans le cas particulier où la somme d'un terme quadratique et d'une norme ℓ_1 est considérée, cet algorithme est aussi connu sous le nom de Landweber seuillé. Il a été introduit sous une première forme dans [Bect *et al.*, 2004; Figueiredo, Nowak, 2003; Daubechies *et al.*, 2004] pour résoudre des problèmes inverses en présence de bruit gaussien. Cet algorithme a été généralisé par Combettes et Wajs dans [Combettes, Wajs, 2005] à une somme de deux fonctions appartenant à une classe plus large. Son emploi pour des décompositions sur des trames a été détaillé dans [Chaux *et al.*, 2007]. Notre attention portera dans un premier temps sur cette dernière version nommée algorithme explicite-implicite et aux propriétés de convergences associées, puis nous détaillerons quelques algorithmes dérivés permettant, entre autres, d'accroître la vitesse de convergence.

3.3.1-a) Problème considéré

Problème 3.18 *Soient f_1 et f_2 deux fonctions de $\Gamma_0(\mathcal{H})$ telles que $\operatorname{Argmin} f_1 + f_2 \neq \varnothing$ où f_2 est une fonction différentiable sur $\mathcal{H}$, de gradient β-Lipschitz. On souhaite trouver*

$$\widetilde{u} \in \underset{u \in \mathcal{H}}{\operatorname{Argmin}} f_1(u) + f_2(u).$$

3.3.1-b) Algorithme [Combettes, Wajs, 2005] et résultats de convergence

Soit $u_0 \in \mathcal{H}$ la valeur initiale. L'algorithme construit la suite $(u_n)_{n\in\mathbb{N}}$ en prenant, pour tout $n \in \mathbb{N}$,

$$u_{n+1} = u_n + \lambda_n\big(\mathrm{prox}_{\gamma_n f_1}(u_n - \gamma_n \nabla f_2(u_n) + b_n) + a_n - u_n\big) \tag{3.18}$$

où $\gamma_n > 0$ désigne le « pas » de l'algorithme et $\lambda_n > 0$ le paramètre de relaxation. Les suites $(a_n)_{n\geq 0}$ et $(b_n)_{n\geq 0}$ correspondent aux erreurs potentiellement introduites dans le calcul de l'opérateur proximal et du gradient. Il faut remarquer que ces erreurs ne sont pas fixées par l'utilisateur lors de l'implantation de l'algorithme.

• La **convergence faible** de $(u_n)_{n\in\mathbb{N}}$ vers une solution du problème 3.18 a été démontrée dans [Combettes, Wajs, 2005] lorsque les hypothèses suivantes sont vérifiées :

Hypothèse 3.19

(i) $0 < \underline{\gamma} \leq \overline{\gamma} < 2\beta^{-1}$ *où* $\underline{\gamma} = \inf_{n\in\mathbb{N}} \gamma_n$ *et* $\overline{\gamma} = \sup_{n\in\mathbb{N}} \gamma_n$,

(ii) $(\forall n \in \mathbb{N})\ 0 < \underline{\lambda} \leq \lambda_n \leq 1$,

(iii) $\sum_{n\in\mathbb{N}} \|a_n\| < +\infty$ *et* $\sum_{n\in\mathbb{N}} \|b_n\| < +\infty$.

• La **convergence forte** de cet algorithme est démontrée dans [Combettes, Wajs, 2005, Theorème 3.4] et [Combettes, Pesquet, 2007b, Theorème 4.5] sous certaines hypothèses supplémentaires.

• Nous énonçons ci-dessous un résultat de **convergence linéaire** lorsque f_1 est fortement convexe. Il faut remarquer que d'autres résultats de convergence linéaire sont disponibles dans la littérature, par exemple dans [Chen, Rockafellar, 1997] dans le cadre d'un algorithme explicite-implicite plus étendu (non restreint au calcul de l'opérateur proximal pour l'étape « implicite » et à la descente de gradient pour l'étape « explicite »). On peut également se référer au travail plus récent de [Bredies, Lorenz, 2008]. Il faut néanmoins remarquer que ces deux approches ne tiennent pas compte du paramètre de relaxation λ.

Lemme 3.20 *Considérons les hypothèses 3.19(i) et (ii) combinées aux hypothèses du problème 3.18. Si* f_1 ***est une fonction fortement convexe de module*** ϑ*, alors l'algorithme explicite-implicite* (3.18)*, pour* $a_n \equiv b_n \equiv 0$*, converge linéairement vers une unique solution* $\widetilde{u}$ *du problème 3.18. De manière plus précise, nous avons :*

$$(\forall n \in \mathbb{N}) \qquad \|u_n - \widetilde{u}\| \leq \left(1 - \frac{\underline{\lambda}\underline{\gamma}\vartheta}{1 + \underline{\gamma}\vartheta}\right)^n \|u_0 - \widetilde{u}\|. \tag{3.19}$$

Preuve :
Puisque $\operatorname{Argmin} f_1 + f_2 \neq \varnothing$ et f_1 est une fonction fortement convexe (donc strictement convexe), il existe un unique minimiseur $\widetilde{u}$ de $f_1 + f_2$. $\widetilde{u}$ est alors un point fixe de l'algorithme implicite-explicite (3.18). Par conséquent, quand $a_n \equiv b_n \equiv 0$, pour tout $n \in \mathbb{N}$,

$$u_{n+1} - \widetilde{u} = (1-\lambda_n)(u_n - \widetilde{u}) + \lambda_n \big(\operatorname{prox}_{\gamma_n f_1}(u_n - \gamma_n \nabla f_2(u_n)) - \operatorname{prox}_{\gamma_n f_1}(\widetilde{u} - \gamma_n \nabla f_2(\widetilde{u}))\big)$$

ce qui implique, par passage à la norme, l'inégalité suivante :

$$\begin{aligned}\|u_{n+1} - \widetilde{u}\| \leq (1-\lambda_n)\|u_n - \widetilde{u}\| \\ + \lambda_n \|\operatorname{prox}_{\gamma_n f_1}(u_n - \gamma_n \nabla f_2(u_n)) - \operatorname{prox}_{\gamma_n f_1}(\widetilde{u} - \gamma_n \nabla f_2(\widetilde{u}))\|.\end{aligned}$$

Nous avons considéré f_1 comme fortement convexe de module ϑ, $\gamma_n f_1$ est alors fortement convexe de module $\gamma_n \vartheta$. Nous pouvons déduire de la proposition 3.11(ii) que $\operatorname{prox}_{\gamma_n f_1}$ est une contraction stricte de constante $(1 + \gamma_n \vartheta)^{-1}$. Par conséquent, nous avons

$$\|u_{n+1} - \widetilde{u}\| \leq (1-\lambda_n)\|u_n - \widetilde{u}\| + \frac{\lambda_n}{1+\gamma_n \vartheta}\|u_n - \gamma_n \nabla f_2(u_n) - \widetilde{u} + \gamma_n \nabla f_2(\widetilde{u})\|.$$

Par le biais de l'hypothèse 3.19(i), on peut également écrire,

$$\|u_{n+1} - \widetilde{u}\| \leq (1-\lambda_n)\|u_n - \widetilde{u}\| + \frac{\lambda_n}{1+\underline{\gamma}\vartheta}\|u_n - \gamma_n \nabla f_2(u_n) - \widetilde{u} + \gamma_n \nabla f_2(\widetilde{u})\|.$$

D'autre part, comme f_2 est une fonction convexe différentiable de gradient continu β-Lipschitz avec $\beta > 0$, nous pouvons déduire du théorème de Baillon-Haddad [Baillon, Haddad, 1977], que $\nabla f_2/\beta$ est fermement contractant. D'après les lemmes 3.6 et 3.7, sachant que $\nabla f_2/\beta$ est fermement contractant (donc appartient à $\mathcal{A}\big(\frac{1}{2}\big)$) et $\gamma_n \in]0, 2/\beta[$, alors $\mathrm{Id} - \gamma_n \nabla f_2 \in \mathcal{A}\big(\frac{\gamma_n \beta}{2}\big)$ et, ainsi, est contractant. Cela implique

$$\|u_n - \gamma_n \nabla f_2(u_n) - \widetilde{u} + \gamma_n \nabla f_2(\widetilde{u})\| \leq \|u_n - \widetilde{u}\|$$

et par conséquent,

$$\|u_{n+1} - \widetilde{u}\| \leq \Big(1 - \frac{\lambda_n \underline{\gamma}\vartheta}{1+\underline{\gamma}\vartheta}\Big)\|u_n - \widetilde{u}\| \leq \Big(1 - \frac{\underline{\lambda}\underline{\gamma}\vartheta}{1+\underline{\gamma}\vartheta}\Big)\|u_n - \widetilde{u}\|,$$

ce qui conduit à l'équation (3.19) et démontre la convergence linéaire. □

3.3.1-c) Versions dérivées

Comme nous l'avons déjà mentionné, les premières méthodes qui permettent de résoudre des problèmes inverses en présence de bruit gaussien et avec un terme de régularisation favorisant la parcimonie sont connus sous le nom de Landweber seuillé, ou encore algorithme IST (Iterative Shrinkage/Thresholding Algorithm) [Bect *et al.*, 2004; Figueiredo, Nowak, 2003; Daubechies *et al.*, 2004]. Il s'agit d'un cas particulier de l'algorithme explicite-implicite dans le cas où, pour tout $v \in \mathcal{H}$ et $\chi > 0$, $f_2 = \|L \cdot -v\|_2^2$, $f_1 = \chi\|\cdot\|_1$, $\lambda_n \equiv 1$, $a_n \equiv b_n \equiv 0$ et $\gamma_n \in]0, \beta^{-1}[$. Soit $u_0 \in \mathcal{H}$, l'algorithme IST construit la suite $(u_n)_{n\geq 1}$ en prenant pour tout $n \in \mathbb{N}$:

$$u_{n+1} = \mathrm{prox}_{f_1}(u_n - L^\top(Lu_n - v)),$$

où prox_{f_1} est détaillé dans l'exemple 3.3. La preuve de la convergence forte vers le minimiseur de $f_1 + f_2$ si $\|L^\top L\| \leq 1$ a été démontrée dans [Daubechies *et al.*, 2004], cette preuve est étendue au cas où f_1 est une norme ℓ_p avec $p \in [1, 2]$. On peut évidemment remarquer que cette version est plus contraignante que l'algorithme explicite-implicite dans le sens où elle n'autorise pas d'erreurs numériques dans le calcul du gradient et de l'opérateur proximal. De plus, la dynamique de γ_n qui garantit la convergence est plus restrictive.

Plus récemment, des algorithmes dérivés de IST qui permettent d'accroître sa vitesse de convergence ont vu le jour. Nous allons détailler l'algorithme TwIST [Bioucas-Dias, Figueiredo, 2007], l'algorithme FISTA [Beck, Teboulle, 2009] et l'algorithme de Nesterov [Nesterov, 1983].

• Algorithme TwIST (Two-step Iterative Shrinkage/Thresholding) :
Cet algorithme proposé dans [Bioucas-Dias, Figueiredo, 2007] est développé en dimension finie. On initialise l'algorithme par $u_0 \in \mathcal{H}$ et $u_1 = \mathrm{prox}_{f_1}(u_0 - L^\top(Lu_0 - v))$ puis, pour tout $n \geq 1$,

$$u_{n+1} = (1-\varepsilon)u_{n-1} + (\varepsilon - \kappa)u_n + \kappa\big(\mathrm{prox}_{\gamma_n f_1}(u_n - \gamma_n L^\top(Lu_n - v))\big)$$

où $\mathrm{prox}_{\gamma_n f_1}$ est détaillé dans l'exemple 3.3. Les preuves de convergence de TwIST requièrent des hypothèses supplémentaires à savoir L injectif (ce qui en pratique élimine un grand nombre de problèmes inverses), $\varepsilon \in]0, 2[$, $\kappa \in]0, 2\varepsilon/\|L^\top L\|[$ et $\gamma_n \in]0, 1/\|L^\top L\|[$. Dans le cas où L est non-inversible, les auteurs proposent une seconde version de TwIST nommée MTwIST pour Monotonic TwIST. Cette approche garantit la convergence de la fonction objectif mais pas celle des itérées. Cependant les auteurs affirment qu'en pratique dans de nombreuses expériences qu'ils ont réalisées, MTwIST converge toujours et de façon plus rapide que l'algorithme IST.

• FISTA (Fast Iterative Shrinkage/Thresholding Algorithm) [Beck, Teboulle, 2009] :
L'algorithme est initialisé par $v_1 = u_0 \in \mathcal{H}$ et $t_1 = 1$ puis on construit la suite $(u_n)_{n\geq 1}$ en prenant pour tout $n \geq 1$,

$$\begin{aligned} u_n &= \mathrm{prox}_{\gamma_n f_1}(v_n - \gamma_n \nabla f_2(v_n)) \\ t_{n+1} &= \frac{1+\sqrt{1+4t_n^2}}{2} \\ v_{n+1} &= u_n + \Big(\frac{t_n - 1}{t_{n+1}}\Big)(u_n - u_{n-1}) \end{aligned}$$

avec, pour tout $n \in \mathbb{N}$, $\gamma_n \in]0, \beta^{-1}[$. Les auteurs de cette version accélérée de ISTA démontrent un taux de convergence en $O(1/n^2)$ contrairement à ISTA pour lequel le taux est en $O(1/n)$ où n est le nombre d'itérations. Il faut remarquer que la preuve de convergence relative à cet algorithme concerne la fonction objectif, c'est à dire la suite $\big(f_1(u_n) + f_2(u_n)\big)_{n\in\mathbb{N}}$, et non pas les itérées $(u_n)_{n\in\mathbb{N}}$. Il n'y a donc en général aucune garantie que l'algorithme converge vers le minimiseur de $f_1 + f_2$.

• Algorithme de Nesterov
Contrairement à l'algorithme ISTA où la direction de descente se base sur le gradient de f_2 à l'itération courante u_n, cette version accélérée se base sur une direction de descente définie comme un élément du sous-espace vectoriel engendré par l'ensemble des gradients de f_2 aux itérations précédentes, *i.e.* $\{\nabla f_2(u_0), \ldots, \nabla f_2(u_n)\}$. L'algorithme est initialisé par $\varepsilon_0 = 0$, $\rho = 0$ et $u_0 \in \mathcal{H}$ puis pour tout $n \geq 1$,

$$\begin{aligned} \kappa_n &= \frac{1}{\beta} + \sqrt{\frac{1}{\beta^2} + \frac{2}{\beta}\varepsilon_n} \\ v_{n+1} &= \frac{\varepsilon_n}{\varepsilon_n + \kappa_n} u_n + \frac{\kappa_n}{\varepsilon_n + \kappa_n} \mathrm{prox}_{\varepsilon_n f_1}(u_n - w_n) \\ u_{n+1} &= \mathrm{prox}_{f_1/\beta}(v_n - \frac{1}{\beta}\nabla f_2(v_n)) \\ w_{n+1} &= w_n + \kappa_n \nabla f_2(u_{n+1}) \\ \varepsilon_{n+1} &= \varepsilon_n + \kappa_n \end{aligned}$$

Cet algorithme assure une décroissance optimale de la suite $\big(f_1(u_n)+f_2(u_n)\big)_{n\in\mathbb{N}}$. Cependant, tout comme FISTA, il n'y a pas de résultats de convergence sur les itérées. Pour plus de détails sur cet algorithme, se référer à [Nesterov, 2009; Weiss *et al.*, 2009].

Dans l'ensemble des méthodes citées précédemment, l'astuce réside dans l'utilisation de plusieurs itérations précédentes pour calculer l'itérée courante. Remarquons que dans

de nombreuses applications à des problèmes inverses [Bioucas-Dias, Figueiredo, 2007; Beck, Teboulle, 2009; Weiss *et al.*, 2009], la réduction du temps de calcul en utilisant ces versions accélérées plutôt que ISTA est mise en évidence. Cependant, il faut prêter attention au fait que la preuve de convergence de l'algorithme explicite-implicite permet de faire varier le « pas » γ_n de l'algorithme dans $]0, 2\beta^{-1}[$, contrairement aux preuves de TwIST et FISTA pour lesquelles le « pas » a une marge de manœuvre plus limitée : $\gamma_n \in]0, \beta^{-1}[$. Ce paramètre influe fortement sur la vitesse de convergence. Il faut donc être attentif lors de comparaisons entre ISTA et les versions accélérées que γ_n ait été choisi de la façon la mieux adaptée.

3.3.2 Algorithme de Douglas-Rachford (DR)

L'algorithme de Douglas-Rachford a été introduit dans [Douglas, Rachford, 1956] dans l'optique de résoudre des problèmes linéaires. Il a ensuite été analysé par Lions et Mercier dans [Lions, Mercier, 1979] pour résoudre des problèmes non linéaires et a été étendu dans [Eckstein, Bertekas, 1992; Combettes, 2004]. Dans [Combettes, Pesquet, 2007a], les auteurs ont proposé l'application de cet algorithme pour le traitement du signal et des images en adoptant un formalisme proximal. On pourra également se référer à [Steidl, Teuber, 2010].

3.3.2-a) Problématique

Pour cet algorithme, nous allons relâcher la contrainte de différentiabilité au sens de Lipschitz formulée dans le paragraphe précédent. Nous considérons le problème de minimisation suivant :

Problème 3.21 *Soient g_1 et g_2 deux fonctions de $\Gamma_0(\mathcal{H})$ telles que $\operatorname{Argmin} g_1 + g_2 \neq \varnothing$. On suppose que l'une de ces trois hypothèses est vérifiée :*

(i) $\operatorname{int}(\operatorname{dom} g_1) \cap \operatorname{dom} g_2 \neq \varnothing$ *;*

(ii) $\operatorname{int}(\operatorname{dom} g_2) \cap \operatorname{dom} g_1 \neq \varnothing$ *;*

(iii) *$\mathcal{H}$ est de dimension finie et* $\operatorname{ri}(\operatorname{dom} g_1) \cap \operatorname{ri}(\operatorname{dom} g_2) \neq \varnothing$.

On souhaite trouver

$$\widetilde{v} \in \operatorname*{Argmin}_{v \in \mathcal{H}} g_1(v) + g_2(v).$$

3.3.2-b) Algorithme [Combettes, Pesquet, 2007a] et résultats de convergence

Soit $v_0 \in \mathcal{H}$ la valeur initiale. L'algorithme construit la suite $(v_m)_{m\geq 0}$ telle que pour tout $m \in \mathbb{N}$,

$$\begin{cases} v_{m+\frac{1}{2}} = \mathrm{prox}_{\kappa g_2} v_m + b_m \\ v_{m+1} = v_m + \tau_m \big(\mathrm{prox}_{\kappa g_1}(2v_{m+\frac{1}{2}} - v_m) + a_m - v_{m+\frac{1}{2}}\big) \end{cases} \tag{3.20}$$

où $\kappa > 0$ désigne le « pas » de l'algorithme, $(\tau_m)_{m\in\mathbb{N}}$ est une suite de réels positifs associée au paramètre de relaxation, et $(a_m)_{m\in\mathbb{N}}$ (resp. $(b_m)_{m\in\mathbb{N}}$) est une suite correspondant aux erreurs produites lors du calcul de l'opérateur proximal de κg_1 (resp. κg_2). De façon similaire à l'algorithme explicite-implicite, ces erreurs ne sont pas fixées par l'utilisateur lors de l'implantation.

• La **convergence faible**[1] de $(v_m)_{m\in\mathbb{N}}$ vers $v \in \mathcal{H}$, telle que $\widetilde{v} = \mathrm{prox}_{\kappa g_2} v$ est une solution du problème 3.21, est garantie sous les hypothèses suivantes :

Hypothèse 3.22

(i) $(\forall m \in \mathbb{N})\ \tau_m \in]0, 2[$ *et* $\sum_{m\in\mathbb{N}} \tau_m(2 - \tau_m) = +\infty$,

(ii) $\sum_{m\in\mathbb{N}} \tau_m(\|a_m\| + \|b_m\|) < +\infty$.

• **La convergence forte** de cet algorithme a été démontrée ci-dessous lorsque g_2 est une fonction fortement convexe. Notons que dans un travail récent [Combettes, 2009], la convergence forte a été démontrée dans le cadre plus général des opérateurs monotones.

Proposition 3.23 *Supposons que les hypothèses du problème 3.21 soient vérifiées. Soit g_2 une fonction fortement convexe, et faisons l'hypothèse que $\inf_{m\in\mathbb{N}} \tau_m > 0$, $\sup_{m\in\mathbb{N}} \tau_m \leq 2$ et $a_m \equiv b_m \equiv 0$. L'algorithme de Douglas-Rachford* (3.20) *génère une suite $(v_{m+1/2})_{m\in\mathbb{N}}$ qui converge fortement vers l'unique solution du problème 3.21.*

Preuve :
Notons rprox l'opérateur défini pour tout $f \in \Gamma_0(\mathcal{H})$ par $\mathrm{rprox}_f = 2\mathrm{prox}_f - \mathrm{Id}$. Soient $a_m \equiv b_m \equiv 0$, les itérations de l'algorithme de Douglas-Rachford (3.20) peuvent alors s'écrire $v_{m+1} = S_m v_m$ où

$$S_m = \tau_m \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2}) + \mathrm{Id} - \tau_m \mathrm{prox}_{\kappa g_2}. \tag{3.21}$$

1. La convergence faible de la suite $(v_{m+\frac{1}{2}})_{m\in\mathbb{N}}$ a aussi récemment été démontrée dans [Svaiter, 2010].

Pour tout $(w, w') \in \mathcal{H}^2$, nous avons,

$$\begin{aligned}&\|S_m w - S_m w'\|^2 = \tau_m^2 \|\mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w) - \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w')\|^2 \\ &+ 2\tau_m \left\langle \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w) - \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w'), w - \tau_m \mathrm{prox}_{\kappa g_2} w - w' + \tau_m \mathrm{prox}_{\kappa g_2} w' \right\rangle \\ &+ \|w - \tau_m \mathrm{prox}_{\kappa g_2} w - w' + \tau_m \mathrm{prox}_{\kappa g_2} w'\|^2.\end{aligned} \tag{3.22}$$

Comme $g_1 \in \Gamma_0(\mathcal{H})$ et $\kappa > 0$, $\kappa g_1 \in \Gamma_0(\mathcal{H})$. En utilisant le lemme 3.9, on en déduit que $\mathrm{prox}_{\kappa g_1}$ est fermement contractant. L'expression (3.22) peut alors être bornée supérieurement par :

$$\begin{aligned}&\|S_m w - S_m w'\|^2 \\ &\leq \tau_m^2 \left\langle \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w) - \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w'), \mathrm{rprox}_{\kappa g_2} w - \mathrm{rprox}_{\kappa g_2} w' \right\rangle \\ &+2\tau_m \left\langle \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w) - \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w'), w - \tau_m \mathrm{prox}_{\kappa g_2} w - w' + \tau_m \mathrm{prox}_{\kappa g_2} w' \right\rangle \\ &+ \|w - \tau_m \mathrm{prox}_{\kappa g_2} w - w' + \tau_m \mathrm{prox}_{\kappa g_2} w'\|^2,\end{aligned}$$

ce qui nous donne après simplifications :

$$\begin{aligned}\|S_m w - S_m w'\|^2 \leq \tau_m (2 - \tau_m) &\left\langle \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w) - \mathrm{prox}_{\kappa g_1}(\mathrm{rprox}_{\kappa g_2} w'), w - w' \right\rangle \\ &+ \|w - \tau_m \mathrm{prox}_{\kappa g_2} w - w' + \tau_m \mathrm{prox}_{\kappa g_2} w'\|^2.\end{aligned}$$

En utilisant ensuite la définition de S_m donnée par l'équation (3.21), nous obtenons après quelques calculs,

$$\begin{aligned}\|S_m w - S_m w'\|^2 \leq (2 - \tau_m) &\langle S_m w - S_m w', w - w' \rangle + (\tau_m - 1)\|w - w'\|^2 \\ &- \tau_m^2 \big(\left\langle \mathrm{prox}_{\kappa g_2} w - \mathrm{prox}_{\kappa g_2} w', w - w' \right\rangle - \|\mathrm{prox}_{\kappa g_2} w - \mathrm{prox}_{\kappa g_2} w'\|^2 \big).\end{aligned} \tag{3.23}$$

Comme κg_2 est fortement convexe de module $\kappa\vartheta$, la proposition 3.11(i) implique l'inégalité suivante :

$$\left\langle \mathrm{prox}_{\kappa g_2} w - \mathrm{prox}_{\kappa g_2} w', w - w' \right\rangle \geq (\kappa\vartheta + 1)\|\mathrm{prox}_{\kappa g_2} w - \mathrm{prox}_{\kappa g_2} w'\|^2,$$

ce qui, combiné avec (3.23), donne

$$\begin{aligned}\|S_m w - S_m w'\|^2 + \kappa\vartheta\tau_m^2 &\|\mathrm{prox}_{\kappa g_2} w - \mathrm{prox}_{\kappa g_2} w'\|^2 \\ &\leq (2 - \tau_m) \langle S_m w - S_m w', w - w' \rangle + (\tau_m - 1)\|w - w'\|^2.\end{aligned} \tag{3.24}$$

D'autre part, soit $\widetilde{v}$ l'unique minimiseur de $g_1 + g_2$. On a alors $\widetilde{v} = \mathrm{prox}_{\kappa g_2} v$ où v est un point fixe de S_m. Par conséquent en choisissant $w = v_m$ et $w' = v$ dans (3.24), nous en déduisons que

$$\begin{aligned}\|v_{m+1} - v\|^2 + \kappa\vartheta\tau_m^2 &\|v_{m+\frac{1}{2}} - \widetilde{v}\|^2 \\ &\leq (2 - \tau_m) \langle v_{m+1} - v, v_m - v \rangle + (\tau_m - 1)\|v_m - v\|^2.\end{aligned} \tag{3.25}$$

Puis, en utilisant le fait que :

$$2\left\langle v_{m+1}-v, v_m-v\right\rangle = \|v_{m+1}-v\|^2 + \|v_m-v\|^2 - \|v_{m+1}-v_m\|^2 \tag{3.26}$$

l'équation (3.25) peut être écrite comme

$$\tau_m\|v_{m+1}-v\|^2 + (2-\tau_m)\|v_{m+1}-v_m\|^2 + 2\kappa\vartheta\tau_m^2\|v_{m+\frac{1}{2}}-\widetilde{v}\|^2 \leq \tau_m\|v_m-v\|^2. \tag{3.27}$$

En considérant l'hypothèse $\sup_{m\in\mathbb{N}}\tau_m \leq 2$, le terme $(2-\tau_m)\,\|v_{m+1}-v_m\|^2$ est positif et le terme de gauche de l'inégalité (3.27) peut être borné inférieurement de la manière suivante :

$$\tau_m\|v_{m+1}-v\|^2 + 2\kappa\vartheta\tau_m^2\|v_{m+\frac{1}{2}}-\widetilde{v}\|^2 \leq \tau_m\|v_m-v\|^2.$$

Finalement, en faisant appel à l'hypothèse $\underline{\tau} = \inf_{m\in\mathbb{N}}\tau_m > 0$, nous obtenons :

$$\|v_{m+1}-v\|^2 + 2\kappa\vartheta\underline{\tau}\|v_{m+\frac{1}{2}}-\widetilde{v}\|^2 \leq \|v_m-v\|^2. \tag{3.28}$$

Par conséquent $\|v_{m+1}-v\|^2 \leq \|v_m-v\|^2$ et la suite $(\|v_m-v\|)_{m\in\mathbb{N}}$ est décroissante. Il existe $c \in\ ]0,+\infty[$ tel que $\lim_{m\to+\infty}\|v_m-v\| = c$. En utilisant (3.28), nous pouvons conclure que $\lim_{m\to+\infty} v_{m+\frac{1}{2}} = \widetilde{v}$, ce qui démontre la convergence forte de $(v_{m+1/2})_{m\in\mathbb{N}}$ vers l'unique minimiseur de g_1+g_2. □

3.4 Opérateur proximal d'une somme de deux fonctions

Nous proposons dans ce paragraphe différentes méthodes pour calculer l'opérateur proximal d'une somme de deux fonctions. Nous étudierons les deux configurations possibles présentées ci-dessous.

Problème 3.24 *Soit $\varepsilon > 0$ et soient g et h deux fonctions appartenant à la classe $\Gamma_0(\mathcal{H})$ où g est une fonction de gradient β-Lipschitz avec $\beta > 0$. On souhaite calculer*

$$\operatorname{prox}_{\varepsilon(g+h)} w = \underset{u\in\mathcal{H}}{\operatorname{argmin}}\ \frac{1}{2}\|u-w\|^2 + \varepsilon g(u) + \varepsilon h(u).$$

Une forme plus générale de ce problème est :

Problème 3.25 *Soit $\varepsilon > 0$ et soient f et h deux fonctions appartenant à la classe $\Gamma_0(\mathcal{H})$. On souhaite calculer*

$$\operatorname{prox}_{\varepsilon(f+h)} w = \underset{u\in\mathcal{H}}{\operatorname{argmin}}\ \frac{1}{2}\|u-w\|^2 + \varepsilon f(u) + \varepsilon h(u).$$

Les deux premières solutions proposées consistent à utiliser les algorithmes explicite-implicite et de Douglas-Rachford en faisant des associations astucieuses des trois termes définissant les problèmes de minimisation 3.24 et 3.25. La troisième solution consiste à utiliser un algorithme de type Dykstra permettant directement de calculer l'opérateur proximal d'une somme de deux fonctions.

3.4.1 Algorithme explicite-implicite

L'algorithme explicite-implicite nous permet de résoudre le problème 3.24. En se reportant au paragraphe 3.3.1, si l'on pose $f_1 = \frac{1}{2}\|\cdot - w\|^2 + \varepsilon h$ et $f_2 = \varepsilon g$, il apparaît immédiat de calculer l'opérateur proximal de $\gamma_n f_1$ avec $\gamma_n \in \left]0, +\infty\right[$ c'est à dire l'opérateur proximal de $\frac{\gamma_n}{2}\|\cdot\|^2 - \gamma_n\langle\cdot, w\rangle + \frac{\gamma_n}{2}\|w\|^2 + \gamma_n\varepsilon h$. Il se déduit directement de la propriété (3.3) :

$$(\forall u \in \mathcal{H}) \qquad \operatorname{prox}_{\gamma_n f_1} u = \operatorname{prox}_{\frac{\gamma_n \varepsilon h}{\gamma_n+1}}\Big(\frac{u + \gamma_n w}{\gamma_n + 1}\Big). \tag{3.29}$$

Si g est une fonction de gradient continu β-Lipschitz et que l'on choisit $a_n \equiv b_n \equiv 0$ dans l'algorithme (3.18), nous obtenons, $\forall n \in \mathbb{N}$

$$u_{n+1} = u_n + \lambda_n \left(\operatorname{prox}_{\frac{\gamma_n \varepsilon h}{\gamma_n+1}}\Big(\frac{u_n - \gamma_n(\varepsilon\nabla g(u_n) - w)}{\gamma_n + 1}\Big) - u_n\right) \tag{3.30}$$

avec

$$0 < \underline{\gamma} \leq \gamma_n \leq \overline{\gamma} < 2\varepsilon^{-1}\beta^{-1}. \tag{3.31}$$

L'algorithme ainsi proposé possède les propriétés suivantes :

Proposition 3.26 *Supposons que la condition* (3.31) *et l'hypothèse 3.19(ii) soient vérifiées, l'algorithme* (3.30) *possède les caractéristiques suivantes,*

(i) *pour tout* $n \in \mathbb{N}$,

$$\|u_n - \operatorname{prox}_{\varepsilon(g+h)} w\| \leq \rho^n \|u_0 - \operatorname{prox}_{\varepsilon(g+h)} w\| \qquad \textit{où} \qquad \rho = 1 - \frac{\underline{\lambda}\underline{\gamma}}{1 + \underline{\gamma}}\,; \tag{3.32}$$

(ii) *en choisissant* $h = \iota_C$ *et* $u_0 = \operatorname{prox}_{\varepsilon g} w$, *nous obtenons :*

$$\operatorname{prox}_{\varepsilon g} w \in C \quad \Rightarrow \quad (\forall n \in \mathbb{N}) \;\; u_n = \operatorname{prox}_{\iota_C + \varepsilon g} w. \tag{3.33}$$

Preuve :
(i) : Comme f_1 est une fonction fortement convexe de module $\vartheta = 1$, (3.32) est obtenue en invoquant le lemme 3.20.
(ii) : Si $u_0 = \operatorname{prox}_{\varepsilon g} w \in C$ et $h = \iota_C$, alors le point de la proposition 3.26(i) implique que

$$(\forall n \in \mathbb{N}) \qquad \|u_n - \operatorname{prox}_{\iota_C + \varepsilon g} w\| \leq \Big(1 - \frac{\underline{\lambda}\underline{\gamma}}{1 + \underline{\gamma}}\Big)^n \|\operatorname{prox}_{\varepsilon g} w - \operatorname{prox}_{\iota_C + \varepsilon g} w\| = 0 \tag{3.34}$$

où la proposition 3.14(ii) a été utilisée dans la dernière inégalité. Ceci montre que (3.33) est satisfaite. □

Remarque 3.27

(i) *La proposition 3.26(i) montre que* $(u_n)_{n\in\mathbb{N}}$ *converge linéairement vers* $\operatorname{prox}_{\varepsilon(h+g)}w$. *Bien que l'équation précédente fournisse une borne supérieure, elle suggère de choisir* λ_n *et* γ_n *aussi grands que possible (i.e.* $\lambda_n \equiv 1$ *et* γ_n *proche de* $2\varepsilon^{-1}\beta^{-1}$*) afin d'optimiser la vitesse de convergence. Ceci a été confirmé dans nos simulations.*

(ii) *La proposition 3.26(ii) est une propriété intéressante dans le cas particulier où* h *est une fonction indicatrice. Elle se base sur le fait que la proposition 3.14(i) permet d'écrire que pour* $\operatorname{prox}_{\iota_C+\varepsilon g}w \in C$, $\operatorname{prox}_{\iota_C+\varepsilon g}w$ *prend une forme triviale. Ainsi cette manière de calculer* $\operatorname{prox}_{\iota_C+\varepsilon g}$ *permet de converger rapidement. Ici, la convergence est effectivement garantie en seulement une itération. Remarquons cependant que* $\operatorname{prox}_{\varepsilon g}w$ *n'est pas toujours évident à calculer ; cela dépend de la forme de la fonction* g.

(iii) *Une alternative pour le calcul de* $\operatorname{prox}_{\varepsilon(h+g)}w$ *consisterait à choisir* $f_1 = h$ *et* $f_2 = \frac{1}{2}\|\cdot - w\|^2 + \varepsilon g$. *Dans ce cas, les itérations de l'algorithme explicite-implicite deviennent :*

$$(\forall n \in \mathbb{N}) \qquad u_{n+1} = u_n + \lambda_n\big(\operatorname{prox}_{\varepsilon h}(u_n - \gamma_n(\varepsilon\nabla g(u_n) + u_n - w)) - u_n\big)$$

avec $0 < \underline{\gamma} \leq \overline{\gamma} < 2(\varepsilon\beta+1)^{-1}$. *Si de plus* h *se réduit à une fonction indicatrice d'un sous-ensemble convexe fermé non-vide de* $\mathcal{H}$, *les itérations sont les suivantes,*

$$(\forall n \in \mathbb{N}) \qquad u_{n+1} = u_n + \lambda_n\big(P_C(u_n - \gamma_n(\varepsilon\nabla g(u_n) + u_n - v)) - u_n\big)$$

et il faut alors remarquer que l'algorithme ainsi proposé se réduit à l'algorithme de gradient projeté, présenté dans la section 2.3.3-b) du chapitre 2. Lors de nos simulations nous avons pu constater que la convergence était plus lente qu'en choisissant (3.30), *ce qui peut vraisemblablement s'expliquer par le fait que* $\operatorname{prox}_{\gamma_n f_1}$ *n'est alors plus strictement contractant avec ce nouveau choix de* f_1.

3.4.2 Algorithme de Douglas-Rachford

L'algorithme de Douglas-Rachford introduit dans le paragraphe 3.3.2 peut être utilisé pour résoudre le problème 3.25 (à savoir le calcul de $\operatorname{prox}_{\varepsilon(f+h)}$) en choisissant par exemple $g_1 = \varepsilon f$ et $g_2 = \frac{1}{2}\|\cdot - w\|^2 + \varepsilon h$, puis en vérifiant que l'une des trois hypothèses liées au problème 3.21 est valide. Dans le présent contexte, ces hypothèses sont :

Hypothèse 3.28

(i) $\operatorname{int}(\operatorname{dom} f) \cap \operatorname{dom} h \neq \varnothing$.

(ii) $\operatorname{int}(\operatorname{dom} h) \cap \operatorname{dom} f \neq \varnothing$.

(iii) *$\mathcal{H}$ est de dimension finie et* $\operatorname{ri}(\operatorname{dom} f) \cap \operatorname{ri}(\operatorname{dom} h) \neq \varnothing$.

Nous proposons alors d'utiliser l'algorithme de Douglas-Rachford (3.20) avec $a_m \equiv b_m \equiv 0$, pour calculer l'opérateur proximal désiré. Dans notre cas nous avons,

$$\operatorname{prox}_{\kappa g_1} v = \operatorname{prox}_{\kappa\varepsilon f} v$$

et, de la même manière que pour l'équation (3.29),

$$\operatorname{prox}_{\kappa g_2} v = \operatorname{prox}_{\frac{\kappa\varepsilon h}{\kappa+1}}\Big(\frac{v+\kappa w}{1+\kappa}\Big).$$

Soit $v_0 \in \mathcal{H}$ la valeur d'initialisation de l'algorithme. Les itérations résultantes sont, pour tout $m \in \mathbb{N}$,

$$\begin{cases} v_{m+\frac{1}{2}} = \operatorname{prox}_{\frac{\kappa\varepsilon h}{\kappa+1}}\Big(\dfrac{v_m+\kappa w}{1+\kappa}\Big) \\ v_{m+1} = v_m + \tau_m\big(\operatorname{prox}_{\kappa\varepsilon f}(2v_{m+\frac{1}{2}} - v_m) - v_{m+\frac{1}{2}}\big). \end{cases} \tag{3.35}$$

Voici quelques propriétés associées :

Proposition 3.29 *Supposons que l'hypothèse 3.28(i), 3.28(ii) ou 3.28(iii) soit vérifiée. Considérons l'algorithme* (3.35) *avec* $\inf_{m\in\mathbb{N}} \tau_m > 0$ *et* $\sup_{m\in\mathbb{N}} \tau_m \leq 2$. *On a alors :*

(i) *la suite* $(v_{m+\frac{1}{2}})_{m\in\mathbb{N}}$ *converge fortement vers* $\operatorname{prox}_{\varepsilon(f+h)} w$ *;*

(ii) *en prenant* $h = \iota_C$, $\kappa = 1$ *et* $v_0 = 2\operatorname{prox}_{\kappa\varepsilon f} w - w$, *nous obtenons :*

$$\operatorname{prox}_{\kappa\varepsilon f} w \in C \quad \Rightarrow \quad (\forall m \in \mathbb{N}) \;\; v_{m+\frac{1}{2}} = \operatorname{prox}_{\iota_C + \kappa\varepsilon f} w. \tag{3.36}$$

<u>Preuve</u> :
(i) : Comme g_2 est fortement convexe de module 1, le résultat est obtenu en utilisant la proposition 3.23.
(ii) : En considérant la première itération de l'algorithme (3.35) lorsque $m = 0$, nous avons $v_{\frac{1}{2}} = \operatorname{prox}_{\kappa\varepsilon f} w$ et $v_1 = v_0$. Par conséquent, $(\forall m \in \mathbb{N})\; v_{m+\frac{1}{2}} = \operatorname{prox}_{\kappa\varepsilon f} w$, qui est aussi égal à $\operatorname{prox}_{\iota_C+\kappa\varepsilon f} w$ selon la proposition 3.14(i). $\square$

Remarque 3.30

(i) *Comme précédemment noté dans la remarque 3.27(ii), la proposition 3.29(ii) correspond à une propriété souhaitable, car elle montre que l'algorithme converge en une itération quand* $\mathrm{prox}_{\varepsilon_n f} x \in C$*, ce qui apparaît cohérent avec la proposition 3.14(i).*

(ii) *Un autre choix peut être envisagé pour* g_1 *et* g_2*, en posant* $g_1 = \frac{1}{2}\|\cdot - w\|^2 + \varepsilon f$ *et* $g_2 = \varepsilon h$*. Cependant, la convergence forte de* $(v_{m+1/2})_{m\in\mathbb{N}}$ *en vertue de la proposition 3.23 est seulement garantie dans le cas précédemment étudié. La propriété 3.29(ii) n'apparaît que dans le premier cas quand* $f = \iota_C$ *(quand* $\kappa = 1$ *et* $v_0 = w$*). Le cas* $h = \iota_C$ *est étudié dans [Dupé et al., 2009], où de bons résultats expérimentaux sont néanmoins obtenus.*

3.4.3 Algorithme de type Dykstra

La dernière méthode pour résoudre le problème 3.25 consiste à utiliser l'algorithme de type Dykstra [Bauschke, Combettes, 2008] permettant de calculer l'opérateur proximal d'une somme de deux fonctions. Soient $r_0 = w \in \mathcal{H}$ et $p_0 = q_0 = 0$. Les suites $(r_m)_{m\geq 1}$ et $(s_m)_{m\geq 1}$ sont construites à partir de la routine suivante : $\forall m \in \mathbb{N}$,

$$\begin{aligned} s_m &= \mathrm{prox}_{\varepsilon h}(r_m + p_m) \\ p_{m+1} &= r_m + p_m - s_m \\ r_{m+1} &= \mathrm{prox}_{\varepsilon f}(s_m + q_m) \\ q_{m+1} &= s_m + q_m - r_{m+1}. \end{aligned}$$

Pour un tel algorithme, il est prouvé que les suites $(r_m)_{m\in\mathbb{N}}$ et $(s_m)_{m\in\mathbb{N}}$ convergent fortement vers $\mathrm{prox}_{\varepsilon(f+h)}(w)$ [Bauschke, Combettes, 2008, théorème 3.3] sous l'hypothèse faible que $\mathrm{dom}\, f \cap \mathrm{dom}\, h \neq \varnothing$.

3.5 Algorithmes imbriqués

Nous venons de présenter différentes approches permettant, d'une part, de minimiser une somme de deux fonctions de $\Gamma_0(\mathcal{H})$ et d'autre part, de calculer l'opérateur proximal associé à cette somme de fonctions. Nous disposons maintenant d'outils permettant de résoudre le problème 3.1 (et par conséquent le problème 3.2), pour lequel nous proposerons trois méthodes différentes. Deux d'entre elles correspondent à une combinaison des algorithmes explicite-implicite (FB) et de Douglas-Rachford (DR) et la troisième consiste à associer l'algorithme explicite-implicte avec l'algorithme de type Dykstra. Le tableau 3.1, rappelle les notations choisies pour les paramètres et fonctionnelles associées aux algorithmes explicite-implicite et de Douglas-Rachford.

Algorithme	Itérations	Suite générée par l'algorithme	Pas	Param. de relaxation	Fonctionnelles
FB	n	$(u_n)_{n\in\mathbb{N}}$	γ_n	λ_n	f_1 et f_2 où f_2 est de gradient β-Lipschitz
DR	m	$(v_m)_{m\in\mathbb{N}}$	κ	τ_m	g_1 et g_2

TABLE 3.1 – Paramètres des algorithmes FB et DR.

3.5.1 Première méthode proposée : DR[FB]

La première méthode consiste à appliquer l'algorithme de Douglas-Rachford comme il a été décrit dans le paragraphe 3.3.2, avec $g_1 = f$ et $g_2 = g + h$. En se reportant aux itérations de l'algorithme (3.20), il est nécessaire de calculer $\mathrm{prox}_{\kappa g_1} = \mathrm{prox}_{\kappa f}$ et $\mathrm{prox}_{\kappa g_2} = \mathrm{prox}_{\kappa(g+h)}$, où $\kappa > 0$. La principale difficulté réside dans le calcul du second opérateur proximal qui se ramène à un problème du type 3.24 en posant $\varepsilon = \kappa$. Comme proposé dans le paragraphe 3.4.1, nous pouvons utiliser l'algorithme explicite-implicite pour ce calcul. L'algorithme résultant imbrique des itérations de l'algorithme explicite-implicite (FB) dans celles de l'algorithme de Douglas-Rachford (DR), d'où le nom de DR[FB]. L'algorithme est détaillé ci-après.

<u>Initialisation</u>
① Soit $\underline{\gamma} \in]0, 2\kappa^{-1}\beta^{-1}[$, $\underline{\lambda} \in]0, 1]$ et $\kappa \in\ ]0, +\infty[$. Choisir $(\tau_m)_{m\in\mathbb{N}}$ satisfaisant l'hypothèse 3.22(i).
② Soit $m = 0$, $v_0 = v_{-1/2} \in \mathcal{H}$.

<u>Itérations</u>
③ Soit $u_{m,0} = v_{m-1/2}$.
④ Pour $n = 0, \ldots, N_m - 1$
a) Choisir $\gamma_{m,n} \in [\underline{\gamma}, 2\kappa^{-1}\beta^{-1}[$ et $\lambda_{m,n} \in [\underline{\lambda}, 1]$
b) Calculer

$$u_{m,n+1} = u_{m,n} + \lambda_{m,n}\left(\mathrm{prox}_{\frac{\gamma_{m,n}\kappa h}{\gamma_{m,n}+1}}\left(\frac{u_{m,n}-\gamma_{m,n}(\kappa\nabla g(u_{m,n})-v_m)}{\gamma_{m,n}+1}\right) - u_{m,n}\right).$$

⑤ Soit $v_{m+\frac{1}{2}} = u_{m,N_m}$.
⑥ Soit $v_{m+1} = v_m + \tau_m\big(\mathrm{prox}_{\kappa f}(2v_{m+\frac{1}{2}} - v_m) - v_{m+\frac{1}{2}}\big)$.
⑦ Incrémenter m $(m \leftarrow m+1)$ et retourner à l'étape ③.

Algorithme 2: Forme générale de l'algorithme DR[FB]

L'étape ① consiste à initialiser les paramètres de l'algorithme DR[FB]. Les étapes ② et ③ correspondent aux initialisations respectives des itérées des algorithmes DR et FB. À l'itération $m \geq 0$ de l'algorithme DR, l'étape ④ consiste à effectuer au plus $N_m \geq 1$ itérations de l'algorithme FB, dans lequel il est possible de faire varier le « pas » $(\gamma_{m,n})_{n\geq 0}$ ainsi que le paramètre de relaxation $(\lambda_{m,n})_{n\geq 0}$. Finalement, les étapes ⑤ et ⑥ correspondent aux itérations de l'algorithme DR. Ici, le terme d'erreur a_m lié au calcul de $\operatorname{prox}_{\kappa f}$ est choisi égal à zéro. Par contre, le nombre fini d'itérations N_m calculées à l'étape ④, peut induire une erreur $b_m = v_{m+1/2} - \operatorname{prox}_{\kappa(g+h)} v_m$ à l'étape ⑤. L'initialisation proposée à l'étape ③ permet de garantir la propriété suivante :

Proposition 3.31 *Supposons que le problème 3.1 possède une solution et que l'une des hypothèses :*

(i) $\operatorname{int}(\operatorname{dom} f) \cap \operatorname{dom} g \cap \operatorname{dom} h \neq \varnothing$,

(ii) $\operatorname{int} \operatorname{dom} g \cap \operatorname{int} \operatorname{dom} h \cap \operatorname{dom} f \neq \varnothing$,

(iii) $\mathcal{H}$ *est de dimension finie et* $\operatorname{ri}(\operatorname{dom} f) \cap \operatorname{ri}(\operatorname{dom} g) \cap \operatorname{ri}(\operatorname{dom} h) \neq \varnothing$ *;*

soit vérifiée. Soit ρ *de valeur donnée par* (3.32) *et soit* $\xi > 0$. *Si* $\inf(g+h) > -\infty$ *et, pour tout* $m \in \mathbb{N}$, *l'entier positif* N_m *est choisi tel que*

$$\rho^{N_m}\sqrt{2\kappa}\big(g(v_0) + h(z_0) - \inf(g+h)\big)^{1/2} \leq \xi \quad \textit{si } m = 0, \tag{3.37a}$$

$$\rho^{N_m - 1}\big(1 + \xi^{-1}\rho^{1-m}\|v_m - v_{m-1}\|\big) \leq 1 \qquad \textit{si } m > 0. \tag{3.37b}$$

$(v_m)_{m\in\mathbb{N}}$ *converge alors faiblement vers* $\widehat{v} \in \mathcal{H}$, *telle que* $\widehat{u} = \operatorname{prox}_{\kappa(g+h)}\widehat{v}$ *est une solution du problème 3.1.*

<u>Preuve</u> :
Selon la proposition 3.26(i), pour tout $m \in \mathbb{N}$,

$$(\forall n \in \{0, \ldots, N_m - 1\}) \qquad \|u_{m,n} - \operatorname{prox}_{\kappa(g+h)} v_m\| \leq \rho^n \|u_{m,0} - \operatorname{prox}_{\kappa(g+h)} v_m\|$$

et, par conséquent

$$\|b_m\| = \|v_{m+1/2} - \operatorname{prox}_{\kappa(g+h)} v_m\| \leq \rho^{N_m}\|v_{m-1/2} - \operatorname{prox}_{\kappa(g+h)} v_m\|. \tag{3.38}$$

Les équations (3.37a) et (3.37b) permettent de garantir que

$$\|b_m\| \leq \rho^m \xi. \tag{3.39}$$

– Si $m = 0$, nous déduisons de (3.38) que

$$\|b_0\| \leq \rho^{N_0}\|v_0 - \operatorname{prox}_{\kappa(g+h)} v_0\|. \tag{3.40}$$

En utilisant la définition de l'opérateur proximal, nous obtenons

$$\begin{aligned}(\forall u \in \mathcal{H}) \quad & \frac{1}{2}\|v_0 - u\|^2 + \kappa\,\big(g(u) + h(u)\big) \\ & \geq \frac{1}{2}\|v_0 - \mathrm{prox}_{\kappa(g+h)} v_0\|^2 + \kappa\,\Big(g\big(\mathrm{prox}_{\kappa(g+h)} v_0\big) + h\big(\mathrm{prox}_{\kappa(g+h)} v_0\big)\Big) \\ & \geq \frac{1}{2}\|v_0 - \mathrm{prox}_{\kappa(g+h)} v_0\|^2 + \kappa\,\inf(g+h)\end{aligned}$$

et, comme $v_0 \in \mathcal{H}$,

$$\|v_0 - \mathrm{prox}_{\kappa(g+h)} v_0\|^2 \leq 2\kappa\big(g(v_0) + h(v_0) - \inf(g+h)\big).$$

En combinant, l'inégalité précédente avec les équations (3.40) et (3.37a), nous concluons que $\|b_0\| \leq \xi$.

– Nous montrons maintenant que (3.39) est vérifiée pour $m > 0$, en supposant que $\|b_{m-1}\| \leq \rho^{m-1}\xi$. L'équation (3.38), nous permet d'écrire

$$\begin{aligned}\|b_m\| &\leq \rho^{N_m}\big(\|v_{m-1/2} - \mathrm{prox}_{\kappa(g+h)} v_{m-1} + \mathrm{prox}_{\kappa(g+h)} v_{m-1} - \mathrm{prox}_{\kappa(g+h)} v_m\|\big) \\ &\leq \rho^{N_m}\big(\|b_{m-1}\| + \|\mathrm{prox}_{\kappa(g+h)} v_{m-1} - \mathrm{prox}_{\kappa(g+h)} v_m\|\big) \\ &\leq \rho^{N_m}\big(\|b_{m-1}\| + \|v_{m-1} - v_m\|\big)\end{aligned}$$

où la contractivité de $\mathrm{prox}_{\kappa(g+h)}$ a été utilisée dans la dernière inégalité. Nous en déduisons que

$$\|b_m\| \leq \rho^{N_m}\big(\rho^{m-1}\xi + \|v_{m-1} - v_m\|\big),$$

qui, combinée à (3.37b), mène à (3.39).

L'équation (3.39) permet d'assurer que l'hypothèse 3.22(ii) est satisfaite puisque

$$\sum_{m\in\mathbb{N}} \tau_m(\|a_m\| + \|b_m\|) \leq 2\xi(1-\rho)^{-1}.$$

Les conditions de convergence faible de l'algorithme de Douglas-Rachford sont rassemblées. □

Les équations (3.37a) et (3.37b) constituent plus une garantie théorique de la convergence pour l'algorithme proposé qu'une ligne de conduite pour le choix de N_m. Dans nos résultats expérimentaux, ces conditions fournissent, en effet, des valeurs pessimistes du nombre d'itérations de l'algorithme explicite-implicite à appliquer à l'étape ④.

Dans le cas où h est une fonction indicatrice, l'algorithme DR[FB] permet de résoudre le problème 3.2, il prend alors la forme donnée par l'algorithme 3.

<u>Initialisation</u>
① Soit $\underline{\gamma} \in]0, 2\kappa^{-1}\beta^{-1}[$, $\underline{\lambda} \in]0, 1]$ et $\kappa \in\ ]0, +\infty[$. Choisir $(\tau_m)_{m\in\mathbb{N}}$ satisfaisant les hypothèses 3.22(i).
② Soit $m = 0$, $v_0 = v_{-1/2} \in C$.

<u>Itérations</u>
③ Soit $u_{m,0} = v_{m-1/2}$.
④ Pour $n = 0, \ldots, N_m - 1$
a) Choisir $\gamma_{m,n} \in [\underline{\gamma}, 2\kappa^{-1}\beta^{-1}[$ et $\lambda_{m,n} \in [\underline{\lambda}, 1]$
b) Calculer $u_{m,n+1} = u_{m,n} + \lambda_{m,n}\left(P_C\left(\frac{u_{m,n}-\gamma_{m,n}(\kappa\nabla g(u_{m,n})-v_m)}{1+\gamma_{m,n}}\right) - u_{m,n}\right)$.
⑤ Soit $v_{m+\frac{1}{2}} = u_{m,N_m}$.
⑥ Soit $v_{m+1} = v_m + \tau_m\big(\mathrm{prox}_{\kappa f}(2v_{m+\frac{1}{2}} - v_m) - v_{m+\frac{1}{2}}\big)$.
⑦ Incrémenter m $(m \leftarrow m+1)$ et retourner à l'étape ③.

Algorithme 3: Forme de l'algorithme DR[FB] avec $h = \iota_C$

Il faut remarquer que l'algorithme FB n'a pas été initialisé, à l'étape ③, comme il l'avait été suggéré à la proposition 3.26(ii). Comme nous l'avons déjà mentionné, le calcul de $\mathrm{prox}_{\kappa g} v_m$ est généralement coûteux. Cependant, l'initialisation proposée à l'étape ③ permet de garantir la propriété suivante, qui complète la proposition 3.31 :

Proposition 3.32 *Supposons que le problème 3.2 possède une solution et également que l'une des hypothèses :*

- $\mathrm{int}\,(\mathrm{dom}\, f) \cap \mathrm{dom}\, g \cap C \neq \varnothing$,
- $\mathrm{int}\,\mathrm{dom}\, g \cap \mathrm{int}\, C \cap \mathrm{dom}\, f \neq \varnothing$,
- $\mathcal{H}$ *est de dimension finie et* $\mathrm{ri}\,(\mathrm{dom}\, f) \cap \mathrm{ri}\,\mathrm{dom}\, g \cap \mathrm{ri}\, C \neq \varnothing$ *;*

soit vérifiée. Soit N_m un entier positif choisi de façon à vérifier

$$\rho^{N_m}\sqrt{2\kappa}\big(g(v_0) - \inf g(C)\big)^{1/2} \leq \xi \qquad \textit{si } m = 0 \tag{3.41}$$

$$\rho^{N_m-1}\big(1 + \xi^{-1}\rho^{1-m}\|v_m - v_{m-1}\|\big) \leq 1 \qquad \textit{si } m > 0 \tag{3.42}$$

alors, $(v_m)_{m\in\mathbb{N}}$ converge faiblement vers $\widehat{v} \in \mathcal{H}$ tel que $\widehat{u} = \mathrm{prox}_{\iota_C+\kappa g}\widehat{v}$ est une solution du problème 3.2. Pour tout $m \in \mathbb{N}$, $(u_{m,n})_{0\leq n<N_m}$ (et donc, $v_{m+1/2}$) appartient à C.

<u>Preuve</u> :
La première partie de la propriété se déduit directement de la proposition 3.31. La deuxième partie de la propriété peut être simplement démontrée en utilisant le fait que $u_{0,0} =$

$v_{-1/2} \in C$ et que $u_{m,n+1}$ est une combinaison convexe de $u_{m,n}$ et d'un élément de $\mathcal{H}$ projeté sur C . $\square$

Comme conséquence de la proposition 3.32, à l'étape ④b), le gradient de g est seulement évalué sur un ensemble ouvert $O_C \subset \mathcal{H}$ tel que $C \subset O_C$. Cela signifie que l'hypothèse de Lipschitz continuité sur le gradient de g est seulement requise sur O_C. Cependant le théorème de Baillon-Haddad utilisé dans les preuves de convergence de l'algorithme explicite-implicite nécessite, en plus de la propriété de gradient Lipschitz, que la fonction g soit deux fois Fréchet différentiable sur O_C [Bauschke, Combettes, 2010, théorème 3.3]. Par conséquent, l'algorithme DR[FB] peut être appliqué pour résoudre le problème suivant :

Problème 3.33 *Soit O_C un ensemble ouvert de $\mathcal{H}$ et $C \subset O_C$ un ensemble convexe non vide de $\mathcal{H}$. Soient f et g deux fonctions de $\Gamma_0(\mathcal{H})$, où g est deux fois Fréchet différentiable sur O_C avec un gradient β-Lipschitz continu pour $\beta \in \,]0, +\infty[$.*

$$\textit{Trouver} \qquad \min_{u \in C} f(u) + g(u).$$

Il faut remarquer que dans le problème précédent, la fonction g ne nécessite pas d'être finie.

3.5.2 Seconde méthode proposée : FB[DR]

Pour cette méthode, une association différente des fonctions du problème 3.1 est considérée. On pose $f_1 = f + h$ et $f_2 = g$. De cette manière, f_2 est de gradient β-Lipschitz et nous pouvons appliquer l'algorithme explicite-implicite présenté dans le paragraphe 3.3.1. Cela requiert cependant de calculer $\mathrm{prox}_{\gamma_n f_1} = \mathrm{prox}_{\gamma_n (f+h)}$, ce qui se ramène à un problème du type 3.25 en posant $\varepsilon = \gamma_n$ et qui peut ainsi être résolu à l'aide des itérations de l'algorithme de Douglas-Rachford comme cela est détaillé dans le paragraphe 3.4.2. La forme complète de ce second algorithme imbriqué est détaillée par l'algorithme 4.

Les étapes ② et ④ réalisent l'initialisation des itérées des algorithme FB et DR. Les étapes ③ et ⑥ correspondent aux itérations de l'algorithme FB. Nous pouvons remarquer que l'étape ⑤ consiste à faire au plus $M_n \geq 1$ itérations de l'algorithme DR décrites dans le paragraphe 3.4.2. L'algorithme DR est initialisé en accord avec la proposition 3.29(ii). On note $m_n < M_n$ le nombre d'itérations pour lequel l'algorithme DR s'arrête. Les termes d'erreur introduits à l'étape ⑥ sont $a_n \equiv z_{n,m_n+\frac{1}{2}} - \mathrm{prox}_{\iota_C + \gamma_n f} u_n$ et $b_n \equiv 0$. Les propriétés de l'algorithme FB[DR] sont décrites ci-dessous.

<u>Initialisation</u>
① Choisir une suite $(\gamma_n)_{n\in\mathbb{N}}$ et $(\lambda_n)_{n\in\mathbb{N}}$ satisfaisant les hypothèses 3.19(i) et (ii). Soit $\underline{\tau} \in]0,2]$.
② Soit $n = 0$, $u_0 \in \mathcal{H}$.

<u>Itérations</u>
③ Soit $u'_n = u_n - \gamma_n \nabla g(u_n)$.
④ Soit $v_{n,0} = 2\mathrm{prox}_{\gamma_n f} u'_n - u'_n$.
⑤ Pour $m = 0, \ldots, M_n - 1$
- a) Calculer $v_{n,m+\frac{1}{2}} = \mathrm{prox}_{\gamma_n h}\Big(\dfrac{v_{n,m} + u'_n}{2}\Big)$.
- b) Choisir $\tau_{n,m} \in [\underline{\tau}, 2]$.
- c) Calculer $v_{n,m+1} = v_{n,m} + \tau_{n,m}\big(\mathrm{prox}_{\gamma_n f}(2v_{n,m+\frac{1}{2}} - v_{n,m}) - v_{n,m+\frac{1}{2}}\big)$.
- d) Si $v_{n,m+1} = v_{n,m}$, alors aller à l'étape ⑥.

⑥ Soit $u_{n+1} = u_n + \lambda_n\big(v_{n,m+\frac{1}{2}} - u_n\big)$.
⑦ Incrémenter n $(n \leftarrow n+1)$ et aller à l'étape ③.

Algorithme 4: Forme générale de l'algorithme FB[DR]

Proposition 3.34 *Supposons que le problème 3.1 possède une solution et que l'une des hypothèses*

- $\mathrm{int}\,(\mathrm{dom}\, f) \cap (\mathrm{dom}\, h) \neq \varnothing$,
- $\mathrm{int}\,(\mathrm{dom}\, h) \cap (\mathrm{dom}\, f) \neq \varnothing$,
- *$\mathcal{H}$ est de dimension finie et* $\mathrm{ri}\,(\mathrm{dom}\, f) \cap \mathrm{ri}\,(\mathrm{dom}\, h) \neq \varnothing$ *;*

soit vérifiée.

(i) *S'il existe une suite d'entiers positifs $(\overline{M}_n)_{n\in\mathbb{N}}$ telle que, pour tout $n \in \mathbb{N}$, $M_n \geq \overline{M}_n$, alors $(u_n)_{n\in\mathbb{N}}$ converge faiblement vers une solution du problème 3.1.*

(ii) *Si $h = \iota_C$ et si, à l'étape ② de l'algorithme FB[DR], $u_0 \in C$, alors la suite $(u_n)_{n\geq 0}$ appartient à C et converge faiblement vers une solution du problème 3.2.*

<u>Preuve</u> :
(i) : Soit $\rho \in]0,1[$. Soit $n \in \mathbb{N}$ et soit $(v_{n,m})_{m\in\mathbb{N}}$ la suite définie par les étapes successives ⑤a), b) et c). En invoquant la proposition 3.29(i), nous savons que $(v_{n,m+\frac{1}{2}})_{m\in\mathbb{N}}$ converge fortement vers $\mathrm{prox}_{\gamma_n(f+h)} u'_n$. Par conséquent, il existe une suite $\overline{M}_n \geq 1$ telle que

$$(\forall m \in \mathbb{N}) \qquad m \geq \overline{M}_n - 1 \quad \Rightarrow \quad \|v_{n,m+\frac{1}{2}} - \mathrm{prox}_{\gamma_n(f+h)} u'_n\| \leq \rho^n.$$

Si $M_n \geq \overline{M}_n$, nous en déduisons que

$$\|a_n\| = \|v_{n,m_n+\frac{1}{2}} - \mathrm{prox}_{\gamma_n(f+h)} u'_n\| \leq \rho^n.$$

L'algorithme s'arrête lorsque $m_n = M_n - 1$ ou si la condition de l'étape ⑤d) est satisfaite (dans ce cas v_{n,m_n} est un point fixe à l'étape ⑤c) et $v_{n,m_n+\frac{1}{2}} = \mathrm{prox}_{\iota_C+\gamma_n f} u'_n$). Nous obtenons par conséquent que $\sum_{n\in\mathbb{N}} \|a_n\| < +\infty$. Les conditions pour la convergence faible de l'algorithme FB sont vérifiées.
(ii) : A l'étape ⑤c), dans le cas particulier où $h = \iota_C$, nous obtenons, pour tout $n \in \mathbb{N}$, que $(v_{n,m+\frac{1}{2}})_m$ appartient à C. Comme u_{n+1} est une combinaison convexe de u_n et $v_{n,m+\frac{1}{2}}$, si de plus nous choisissons u_0 appartenant à C, nous en déduisons facilement que pour tout $n \geq 1$, $u_n \in C$. □

La proposition 3.34(i) garantit qu'en choisissant M_n suffisament grand, l'algorithme permet de résoudre le problème 3.1 et par conséquent le problème 3.2 en posant $h = \iota_C$. Bien que ce résultat puisse apparaître imprécis, il a été observé dans nos simulations que de petites valeurs de M_n sont suffisantes pour assurer la convergence de l'algorithme.

3.5.3 Troisième méthode proposée : FB[Dyk]

On associe de la même façon que précédemment les fonctions f, g et h constituant le problème 3.1 à savoir $f_1 = f + h$ et $f_2 = g$. De la même manière que pour le paragraphe précédent, f_2 est de gradient β-Lipschitz. Nous pouvons alors appliquer l'algorithme explicite-implicite présenté dans le paragraphe 3.3.1. Il reste le calcul de $\mathrm{prox}_{\gamma_n f_1} = \mathrm{prox}_{\gamma_n(f+h)}$, qui peut être obtenu en utilisant l'algorithme de type Dykstra, présenté dans le paragraphe 3.4.3, en posant $\epsilon = \gamma_n$. L'algorithme FB[Dyk] résultant est donné par l'algorithme 5.

Proposition 3.35

(i) *Supposons que le problème 3.1 possède une solution et que l'hypothèse* $\mathrm{dom}\, f \cap \mathrm{dom}\, h \neq \varnothing$ *soit vérifiée. S'il existe une suite d'entiers positifs* $(\overline{M}_n)_{n\in\mathbb{N}}$ *telle que, pour tout* $n \in \mathbb{N}$, $M_n \geq \overline{M}_n$, *la suite* $(u_n)_{n\in\mathbb{N}}$ *converge faiblement vers une solution du problème 3.1.*

(ii) *Si* $h = \iota_C$ *et* $u_0 \in C$ *alors* $(u_n)_{n\in\mathbb{N}}$ *appartient à* C *et converge faiblement vers une solution du problème 3.2 sous l'hypothèse qu'une solution existe.*

Preuve :
La démonstration résulte des mêmes arguments que pour l'algorithme FB[DR]. La preuve de convergence est essentiellement basée sur la convergence forte de l'algorithme de type Dykstra. □

Initialisation
① Choisir $u_0 \in \mathcal{H}$ et $n = 0$.
② Choisir $\lambda_n \in]0, 1]$ et $\gamma_n \in]0, 2\beta^{-1}[$.

Itérations
③ Poser $r_{n,0} = u_n - \gamma_n \nabla g(u_n)$.
④ Poser $p_{n,0} = q_{n,0} = 0$.
⑤ Pour $m = 0, \ldots, M_n$
a) $s_{n,m} = \mathrm{prox}_{\gamma_n f}(r_{n,m} + p_{n,m})$
b) $p_{n,m+1} = r_{n,m} + p_{n,m} - s_{n,m}$
c) $r_{n,m+1} = \mathrm{prox}_{\gamma_n h}(s_{n,m} + q_{n,m})$
d) $q_{n,m+1} = s_{n,m} + q_{n,m} - r_{n,m+1}$
e) Si $r_{n,m+1} = r_{n,m}$, alors aller à l'étape ⑥.
⑥ Faire $u_{n+1} = u_n + \lambda_n \big(r_{n,m+1} - u_n\big)$.
⑦ Incrémenter n ($n \leftarrow n + 1$) et aller en ③.

Algorithme 5: Forme générale de l'algorithme FB[Dyk]

Une conséquence directe de la proposition 3.35(ii) (resp. proposition 3.34(ii)) est que le gradient de g est seulement évalué sur C à l'étape ③ de l'algorithme 5 (resp. étape ③ de l'algorithme 4). Les algorithmes FB[Dyk] et FB[DR] permettent, tout comme l'algorithme DR[FB], de résoudre le problème 3.33 et ainsi de considérer une fonction g pouvant être non finie. Dans le paragraphe suivant, nous formulerons des critères correspondant au problème 3.33.

3.6 Résolution de problèmes de restauration d'images

Comme nous l'avions mentionné dans l'introduction de ce chapitre, la partie précédente présente des algorithmes imbriqués utiles pour résoudre des problèmes de restauration d'images. En effet, dans le cas d'une image dégradée par un bruit blanc gaussien, g peut modéliser le terme d'attache aux données et prendre la forme d'un terme quadratique, qui est une fonction de gradient Lipschitz. f et h correspondent dans ce cas de figure à des termes de régularisation. Cette configuration requiert l'utilisation des algorithmes imbriqués. Cependant, dans un cadre de restauration ne se restreignant pas à un bruit blanc gaussien de variance fixe, par exemple un bruit gaussien dont la variance dépend du signal ou un bruit de type Poisson, la fonction peut soit être de gradient Lipschitz avec un β grand soit ne pas être de gradient Lipschitz. Dans ces configurations, les algorithmes imbriqués proposés soit convergent très lentement vers la solution du problème 3.1 soit n'offrent

aucune garantie de convergence et ceci même en dimension finie (le « pas » γ_n de l'algorithme explicite-implicite étant inversement proportionnel à β – cf. hypothèse 3.19(i)).

Remarquons que dans ce paragraphe, nous nous limiterons au cas où $h = \iota_C$. Nous y comparerons dans un premier temps les différents algorithmes imbriqués sur un exemple classique de déconvolution d'images en présence de bruit blanc gaussien. Puis, dans un second temps, nous considérons une classe de fonctions g deux fois différentiables mais ne vérifiant pas l'hypothèse de Lipschitz continuité du gradient. Pour cette classe de fonctions, nous proposons une extension quadratique permettant de contourner le problème de Lipschitz différentiabilité. Nous en étudions le comportement sur deux exemples de restauration : le premier concerne le cas d'un bruit gaussien avec une variance dépendant du signal et le second traite le cas d'un bruit de type Poisson.

Des résultats additionnels dans le cas où $h \neq \iota_C$ sont fournis dans [Pustelnik *et al.*, 2009a].

3.6.1 Modélisation du problème

Notre but consiste à restaurer une image $\overline{y}$ appartenant à un espace de Hilbert séparable $\mathcal{G}$ à partir d'observations dégradées $z \in \mathcal{G}$. Ici, des images numériques de taille $N_1 \times N_2$ sont considérées et $\mathcal{G} = \mathbb{R}^N$ avec $N = N_1 N_2$. Soit A un opérateur linéaire de $\mathcal{G}$ dans $\mathcal{G}$ modélisant un processus de dégradation linéaire (dans cette partie nous considérerons qu'il s'agit d'un opérateur de convolution). L'image $\overline{u} = A\overline{y}$ (resp. z) est une réalisation d'un vecteur aléatoire de valeurs réelles $\overline{U} = (\overline{U}_i)_{1\leq i\leq N}$ (resp. $Z = (Z_i)_{1\leq i\leq N}$). L'image $\overline{U}$ est, de plus, contaminée par un bruit. Conditionnellement à $\overline{U} = (u_i)_{1\leq i\leq N} \in \mathcal{G}$, le vecteur aléatoire Z est supposé avoir des composantes indépendantes, qui sont soit discrètes avec pour probabilité conditionnelle la fonction de masse $(\mu_{Z_i|\overline{U}_i=u_i})_{1\leq i\leq N}$, soit absolument continues avec les densités conditionnelles qui sont également notées $(\mu_{Z_i|\overline{U}_i=u_i})_{1\leq i\leq N}$. Dans ce paragraphe, nous nous intéresserons aux distributions de probabilités telles que :

$$(\forall i \in \{1,\ldots,N\})(\forall \upsilon \in \mathbb{R}) \qquad \mu_{Z_i|\overline{U}_i=\upsilon}(z_i) \propto \exp\big(-\psi_i(\upsilon)\big), \tag{3.43}$$

où $(\psi_i)_{1\leq i\leq N}$ sont des fonctions de $\Gamma_0(\mathbb{R})$, deux fois continuement différentiables sur $\mathbb{R}$ et telles que :

$$(\forall i \in \{1,\ldots,N\}) \quad \inf \psi_i > -\infty. \tag{3.44}$$

De plus, un modèle probabiliste *a priori* simple et efficace de l'image $\overline{y}$ est utilisé en considérant les coefficients de décomposition de cette image sur une trame [Daubechies, 1992; Han, Larson, 2000]. L'espace des coefficients de trame est un espace euclidien $\mathcal{H} = \mathbb{R}^K$ ($K \geq N$). Nous utilisons une représentation linéaire de la forme :

$$\overline{y} = F^*\overline{x}$$

où $F^* : \mathcal{H} \to \mathcal{G}$ est l'opérateur de synthèse de trame (F^* est surjectif). Nous supposons alors que le vecteur $\overline{x}$ des coefficients de trame est une réalisation d'un vecteur aléatoire $\overline{X}$

dont les composantes sont indépendantes. Chaque composante $\overline{X}_k$ avec $k \in \{1, \ldots, K\}$ de $\overline{X}$, possède une densité de probabilité $\exp(-\phi_k(\cdot))/\int_{-\infty}^{+\infty} \exp(-\phi_k(\xi))\, d\xi$ où ϕ_k est une fonction finie de $\Gamma_0(\mathbb{R})$.

Enfin, nous supposons que nous avons une information *a priori* sur $\overline{x}$ qui peut être exprimée par le fait que $\overline{x}$ appartient à un ensemble convexe fermé C de $\mathcal{H}$. La contrainte C devra satisfaire :

$$(AC^*) \cap \operatorname{dom} \Psi \neq \varnothing \tag{3.45}$$

où

$$C^* = F^*C = \left\{ F^*x \;\middle|\; x \in C \right\}$$

et

$$\Big(\forall u = \big(u_i\big)_{1 \leq i \leq N} \in \mathcal{G}\Big) \qquad \Psi(u) = \sum_{i=1}^{N} \psi_i\big(u_i\big).$$

Sous ces hypothèses, on peut montrer (voir [Chaux *et al.*, 2007]) qu'une estimation au sens du *Maximum A Posteriori* (MAP) des coefficients de trame $\overline{x}$ peut être obtenue à partir des observations $z = \big(z_i\big)_{1 \leq i \leq N}$ en minimisant sur un espace de Hilbert $\mathcal{H}$ la fonctionnelle $f + g + \iota_{\mathrm{C}}$ où

$$\Big(\forall x = \big(x_k\big)_{1 \leq k \leq K} \in \mathcal{H}\Big) \qquad f(x) = \sum_{k=1}^{K} \phi_k\big(x_k\big) \tag{3.46}$$

et

$$g = \Psi \circ A \circ F^*. \tag{3.47}$$

Cette formulation pour f, g et $h = \iota_C$, nous permet d'introduire la proposition suivante.

Proposition 3.36 *Sous la condition* (3.45)*, si nous considérons f et g définies respectivement par* (3.46) *et* (3.47)*,*

(i) *f et g appartiennent à la classe $\Gamma_0(\mathcal{H})$;*

(ii) *si f est coercive ou $\operatorname{dom} g \cap C$ est borné, alors la minimisation de $f + g + \iota_{\mathrm{C}}$ admet une solution. De plus, si f est strictement convexe sur $\operatorname{dom} g \cap C$, la solution du problème 3.2 est unique.*

<u>Preuve</u> :
(i) : Comme les fonctions $(\phi_k)_{1 \leq k \leq K}$ sont des fonctions finies de $\Gamma_0(\mathbb{R})$, f est une fonction finie de $\Gamma_0(\mathcal{H})$. De la même manière, comme les fonctions $(\psi_i)_{1 \leq i \leq N}$ sont dans $\Gamma_0(\mathbb{R})$, Ψ appartient à $\Gamma_0(\mathcal{G})$. De plus, en utilisant (3.45), nous avons $\operatorname{Im}(AF^*) \cap \operatorname{dom} \Psi \neq \varnothing$. Nous en déduisons que $\operatorname{dom} g \neq \varnothing$ et, par conséquent, $g \in \Gamma_0(\mathcal{H})$.

(ii) : Comme $\operatorname{dom} f = \mathcal{H}$ et (3.45) montre que $\operatorname{dom} g \cap C = \operatorname{dom}(\Psi \circ A \circ F^*) \cap C \neq \varnothing$, alors $\operatorname{dom} f \cap \operatorname{dom} g \cap C \neq \varnothing$. D'autre part, comme f et g sont dans $\Gamma_0(\mathcal{H})$, nous déduisons que $f + g + \iota_{\mathrm{C}}$ appartient à la classe $\Gamma_0(\mathcal{H})$.
Supposons maintenant que f est coercive. La condition (3.44) implique que $\inf \Psi(\mathcal{G}) > -\infty$ et, par conséquent, $\inf g(\mathcal{H}) \geq \inf \Psi(\mathcal{G}) > -\infty$. On en déduit que $f + g + \iota_{\mathrm{C}} \geq f + \inf g(\mathcal{H}) + \iota_{\mathrm{C}}$ est coercive.
Lorsque $\operatorname{dom} g \cap C$ est borné, $f + g + \iota_{\mathrm{C}}$ est également coercive. L'existence d'une solution au problème de minimisation se déduit des résultats classiques de l'analyse convexe (rappelés par la proposition 2.22 dans le chapitre 2).
Quand f est strictement convexe sur $\operatorname{dom} g \cap C$, l'unicité de la solution est induite par le fait que $f + g + \iota_{\mathrm{C}}$ est strictement convexe. □

Remarque 3.37 *La fonction f est coercive (resp. strictement convexe) si et seulement si les fonctions $(\phi_k)_{1 \leq k \leq N}$ sont coercives [Chaux* et al.*, 2007, Prop. 3.3(iii)(c)] (resp. strictement convexes).*

3.6.2 Comparaison expérimentale des algorithmes imbriqués sur un exemple de restauration en présence de bruit gaussien

Nous considérons le problème de restauration d'une image $\overline{y} \in \mathbb{R}^N$ (figure 3.3(a)) dégradée par un opérateur de convolution A (ici un flou uniforme de taille 3×3) et un bruit blanc gaussien de variance $\sigma^2 = 30^2$. Nous souhaitons trouver une image $\widehat{y} \in \mathbb{R}^N$ aussi proche que possible de l'image $\overline{y}$, en se basant sur les observations (figure 3.3(b)) notée $z \in \mathcal{G}$ avec $\mathcal{G} = \mathbb{R}^N$. La fonctionnelle que nous proposons de minimiser est la suivante :

$$\widehat{x} = \underset{x \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{2\sigma^2} \|AF^*x - z\|_2^2 + f(x) + \iota_{C^*}(F^*x). \tag{3.48}$$

où, $F^* : \mathcal{H} \to \mathcal{G}$ est l'opérateur linéaire de synthèse de trame avec $\mathcal{H} = \mathbb{R}^K$ et $K \geq N$. Cela conduit à des coefficients de trame que nous noterons $x = (x_k)_{1 \leq k \leq K}$. Nous utilisons une version de la représentation en arbre dual [Chaux *et al.*, 2006] associée à des filtres symmlets de longueur 6, conduisant à une trame ajustée ($F^* \circ F = \nu \operatorname{Id}$ avec $\nu = 2$). Dans la formulation (3.48), pour tout $i \in \{1, \ldots, N\}$, $\psi_i = (\cdot - z)^2$ est une fonction de $\Gamma_0(\mathbb{R})$ telle que $\inf \psi_i = 0 > -\infty$. La fonction f correspond à une régularisation dans le domaine transformé $\mathcal{H}$ telle que $f(x) = \sum_{k=1}^K \chi_k |x_k| + \omega_k |x_k|^{p_k}$ où $\chi_k > 0$, $\omega_k > 0$ et $p_k \in \{4/3, 3/2, 2\}$ sont adaptés par sous-bandes. Le calcul de l'opérateur proximal de f se déduit de l'exemple 3.38 et de la proposition 3.39 rappelés ci-dessous.

Exemple 3.38 [Chaux *et al.*, 2007] *Soient* $\chi > 0$, $p \in [1, +\infty[$. *On considère la fonction* $\varphi\colon \mathbb{R} \to \,]-\infty, +\infty] : \eta \mapsto \chi|\xi|^p$. *Pour tout* $\xi \in \mathbb{R}$, $\operatorname{prox}_\varphi \xi$ *est alors donné par,*

$$\begin{cases} \operatorname{sign}(\xi)\max\{|\xi| - \chi, 0\} & \textit{si } p = 1 \\ \xi + \frac{4\chi}{3\,.\,2^{1/3}}\big((\epsilon - \xi)^{1/3} - (\epsilon + \xi)^{1/3}\big) & \\ \qquad \textit{où} \quad \epsilon = \sqrt{\xi^2 + 256\chi^3/729} & \textit{si } p = \frac{4}{3} \\ \xi + \frac{9\chi^2 \operatorname{sign}(\xi)}{8}\Big(1 - \sqrt{1 + \frac{16|\xi|}{9\chi^2}}\Big) & \textit{si } p = \frac{3}{2} \\ \frac{\xi}{1+2\chi} & \textit{si } p = 2 \\ \operatorname{sign}(\xi)\frac{\sqrt{1+12\chi|\xi|}-1}{6\chi} & \textit{si } p = 3 \\ \big(\frac{\epsilon+\xi}{8\chi}\big)^{1/3} - \big(\frac{\epsilon-\xi}{8\chi}\big)^{1/3} & \\ \qquad \textit{où} \quad \epsilon = \sqrt{\xi^2 + 1/(27\chi)} & \textit{si } p = 4 \end{cases} \tag{3.49}$$

Proposition 3.39 [Combettes, Pesquet, 2007b] *Soit* $\chi > 0$, $\omega > 0$, $p \in]1, +\infty[$, *soit* $\varphi\colon \mathbb{R} \to \,]-\infty, +\infty] : \xi \mapsto \chi|\xi|$ *et soit* $\phi\colon \mathbb{R} \to \,]-\infty, +\infty] : \xi \mapsto \omega|\xi|^p$.
On a alors, pour tout $\xi \in \mathbb{R}$,

$$\operatorname{prox}_{\varphi+\phi}\xi = \operatorname{prox}_\phi \circ \big(\operatorname{prox}_\varphi \xi\big).$$

Enfin, $h = \iota_{C^*} \circ F^*$ introduit une contrainte dure sur l'appartenance à un convexe $C^* \subset \mathcal{G}$. Dans notre exemple, en accord avec la dynamique de l'image, nous avons choisi : $C^* = [0, 255]^N$. La projection sur le convexe C ($P_C = \operatorname{prox}_{\iota_{C^*} \circ F^*}$) est calculée en utilisant la proposition 3.12 avec $L = F^*$ et $\chi = \nu$. L'existence et l'unicité de la solution de (3.48) est assurée par le fait que la fonction f est coercive et strictement convexe (cf. proposition 3.36). L'image restaurée $\widehat{y}$ est liée à l'unique minimiseur de (3.48) par la relation $\widehat{y} = F^*\widehat{x}$.

Nous avons donc testé et comparé les trois algorithmes proposés dans le paragraphe précédent, en terme de vitesse de convergence. Deux types de convergence sont évaluées :

- la convergence de la fonctionnelle $f + g$ est représentée figure 3.1,
- la convergence des itérées, à savoir l'évolution de $\|x_n - \widehat{x}\|$ est représentée figure 3.2, où $\widehat{x}$ représente l'unique minimiseur de $f + g + \iota_C$ (en accord avec le choix de f et la proposition 3.36(ii)).

Pour chaque configuration, nous présentons l'évolution du critère en fonction des itérations et en fonction du temps CPU. Pour une comparaison juste, nous initialisons les algorithmes FB[DR] et FB[Dyk] par $u_0 = P_C(Fz)$. Pour l'algorithme DR[FB], on l'initialise par $v_0 = P_C(Fz)$. D'autre part, les paramètres κ, τ_m, λ_n et γ_n, des différents algorithmes, sont choisis afin d'optimiser la vitesse de convergence. Pour l'algorithme DR[FB], nous avons fixé $\kappa = 60$, $\tau_m \equiv 1$, $\lambda_{m,n} \equiv 1$ et $\gamma_{m,n} \equiv 0.99/(\kappa\sigma^2)$. La convergence de l'algorithme est assurée par la proposition 3.32 car l'hypothèse 3.28(i) est vérifiée. Cependant

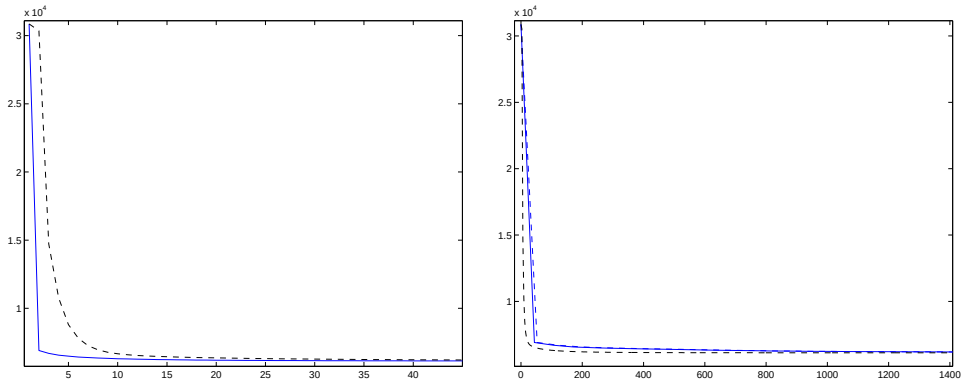
pour augmenter la vitesse de convergence, les règles empiriques suivantes ont été fixées pour choisir le nombre N_m des itérations de l'algorithme FB (cf. (3.41) et (3.42)) :

$$N_m = \inf \left\{ n \in \mathbb{N}^* \;\middle|\; \|u_{m,n} - u_{m,n-1}\| \leq \eta \right\}, \tag{3.50}$$

avec $\eta = 10^{-4}$. Pour les algorithmes FB[DR] ou FB[Dyk], les paramètres liés à l'algorithme FB sont choisis comme suit : $\lambda_n \equiv 1$ et $\gamma_n \equiv 0.995/\sigma^2$. Le paramètre de relaxation de la boucle interne de l'algorithme DR est fixé $\tau_{n,m} \equiv 1$. L'algorithme de type Dykstra ne nécessite pas l'ajustement de paramètres additionnels. La convergence des algorithmes est assurée par les propositions 3.34 et 3.35 (i). le nombre M_n d'itérations de l'algorithme DR et de l'algorithme de type Dykstra a été fixé comme suit :

$$M_n = \inf \left\{ m \in \mathbb{N}^* \;\middle|\; \|v_{n,m} - v_{n,m-1}\| \leq \eta \right\} \tag{3.51}$$

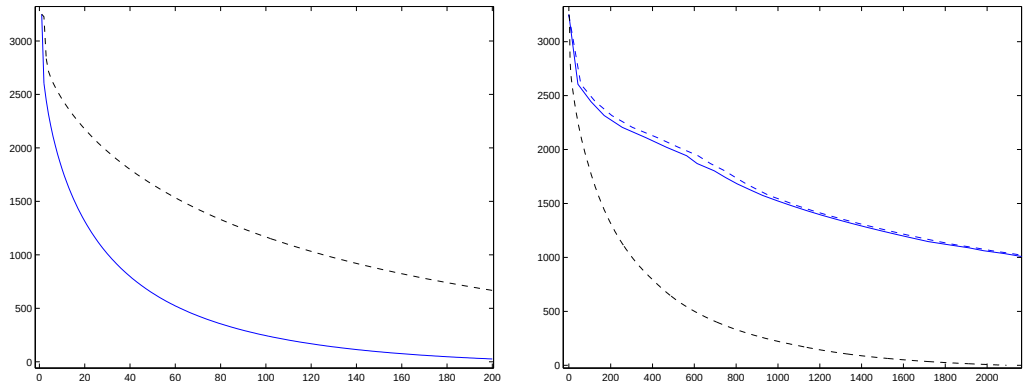
avec la même valeur de η que pour l'algorithme DR[FB].



(a) (b)

FIGURE 3.1 – Evolution de $f + g$ en fonction (a) des itérations et (b) du temps CPU en secondes (Intel Core 2, 2.66Ghz). FB[DR] (ligne continue bleue), FB[Dyk] (ligne discontinue bleue) et DR[FB] (ligne discontinue noire).

Sur la figure 3.1(a), nous constatons que l'algorithme FB[DR] converge bien plus rapidement que l'algorithme DR[FB]. La figure 3.1(b) modère néanmoins nos propos dans le sens où chaque itération de FB sera plus coûteuse que celle de DR, cela étant dû à une convergence plus lente de l'algorithme de DR ou de type Dykstra qui pénalise l'algorithme FB à chaque itération. Reste à comparer l'algorithme FB[DR] et FB[Dyk]. Remarquons, qu'aucune courbe ne correspond à FB[Dyk] sur la figure 3.1(a) car elle se serait superposée à celle de FB[DR]. En effet, l'élément qui diffère entre FB[DR] et FB[Dyk] est le nombre d'itérations internes. La figure 3.1(b), montre l'évolution du critère en fonction des itérations et il résulte de nos expérimentations que l'agorithme FB[DR] converge, dans la meilleure des configurations, aussi rapidement que l'algorithme FB[Dyk], cela



(a) (b)

FIGURE 3.2 – Valeur de $\|x_n - \widehat{x}\|$ en fonction (a) des itérations et (b) du temps CPU en secondes (Intel Core 2, 2.66Ghz). FB[DR] (ligne continue bleue), FB[Dyk] (ligne discontinue bleue) et DR[FB] (ligne discontinue noire).

dépendant des paramètres choisis. Pour les figures 3.2(a) et (b) les conclusions sont identiques.

Remarque 3.40

(i) *Les algorithmes FB[·] requièrent un nombre d'itérations de leur boucle interne bien plus important que l'algorithme DR[FB] pour arriver à une tolérance du même ordre de grandeur pour le calcul de* $\operatorname{prox}_{f+\iota_C}$ *ou* $\operatorname{prox}_{g+\iota_C}$ *(environ 400 itérations pour DR ou Dyk et seulement une vingtaine pour FB pour une tolérance* $\eta = 10^{-4}$*).*

(ii) *L'algorithme explicite-implicite requiert le calcul de* A *et* A^* *à chaque itération. Par conséquent pour certaines applications autres que de la déconvolution, il peut être pénalisant d'utiliser l'algorithme DR[FB].*

(iii) *Entre les algorithmes FB[DR] et FB[Dyk], le choix se porterait de prime abord sur l'algorithme FB[Dyk] possédant des conditions de qualifications moins restrictives et permettant d'obtenir des taux de convergence au moins identiques à FB[DR] et d'une manière générale plus avantageux. Cependant, l'algorithme FB[Dyk] ne possède pas une propriété du style de celle de la proposition 3.29(ii) permettant à la boucle interne de converger en une itération lorsque* $\operatorname{prox}_{\gamma_n f} \in C$*.*

D'autre part, pour mettre en évidence l'intérêt de la fonction h proposée en (3.48), nous examinons l'effet de la restauration avec et sans la contrainte sur les figures 3.3(c) et (d). On peut constater que la présence de la fonction h permet une meilleure restauration de l'image. L'erreur entre l'image restaurée y (après 100 itérations de DR[FB]) et l'image

originale $\overline{y}$ est évaluée par le rapport signal sur bruit (SNR) dont la définition est donnée par l'équation (2.36) dans le chapitre 2.

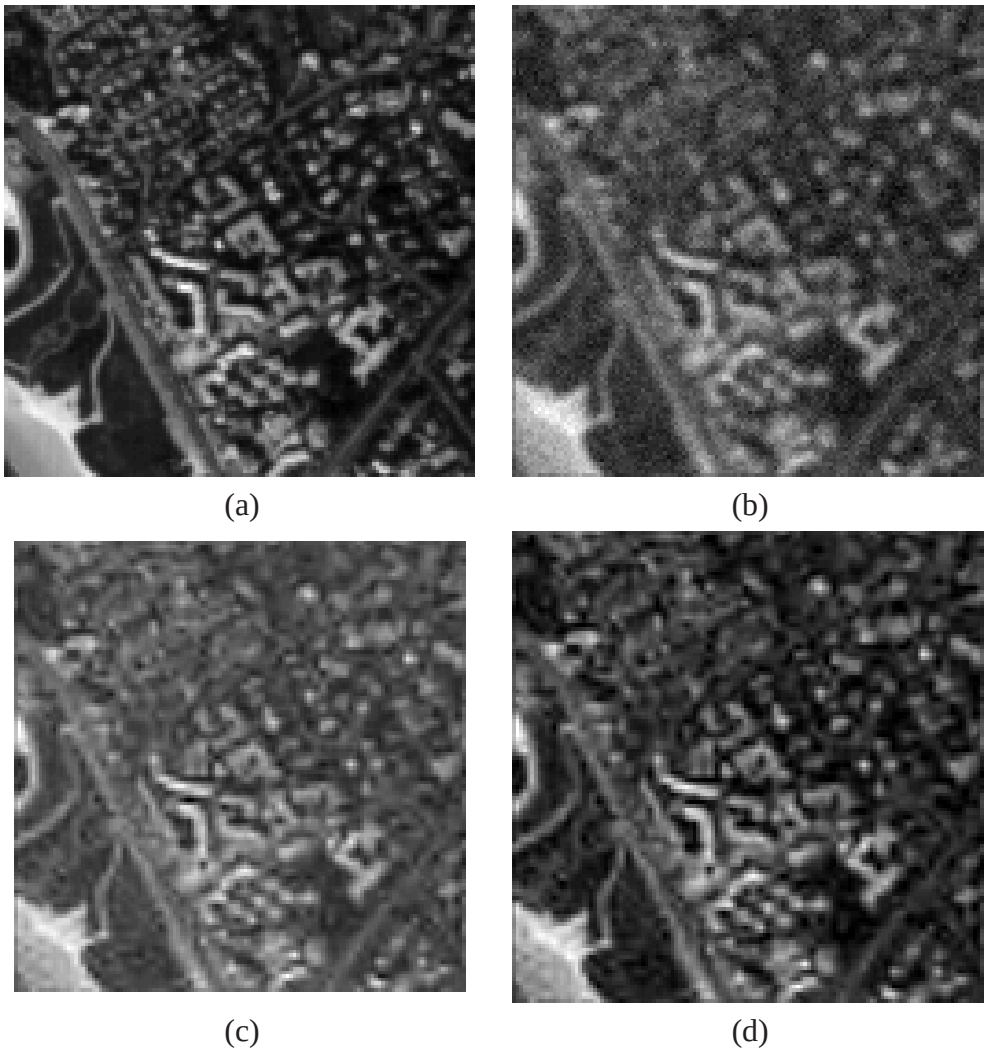


FIGURE 3.3 – Résultats sur une image satellitaire. (a) Originale, (b) dégradée (SNR = 13.4 dB), (c) restaurée sans la contrainte (SNR = 14.8 dB) et (d) restaurée avec la contrainte (SNR = 15.0 dB).

3.6.3 Approximation inférieure du terme d'attache aux données pour la résolution d'une classe plus large de problèmes de restauration

Dans ce paragraphe, de façon à pouvoir gérer des fonctionnelles g non finies, nous effectuons sur les fonctions $(\psi_i)_{1\leq i\leq N}$ des hypothèses plus restrictives que celles proposées dans le paragraphe 3.6.1. Ces hypothèses sont énoncées ci-après. Dans cette partie, nous posons $\mathcal{G} = \mathcal{H} = \mathbb{R}^N$.

Hypothèse 3.41 *Il existe un sous-ensemble non vide $\mathbb{I}$ de $\{1,\ldots,N\}$ et une constante $\delta \in \mathbb{R}$ telle que, pour tout $i \in \{1,\ldots,N\}$,*

(i) $\operatorname{dom}\psi_i =]\delta,+\infty[$ *si* $i \in \mathbb{I}$ *et,* $\operatorname{dom}\psi_i = [\delta,+\infty[$ *si* $i \notin \mathbb{I}$*;*

(ii) *si $i \in \mathbb{I}$, la fonction ψ_i est deux fois continuement différentiable sur $]\delta,+\infty[$ telle que $\inf_{\upsilon\in]\delta,+\infty[}\psi_i(\upsilon) > -\infty$ et*

$$\lim_{\substack{\upsilon\to\delta\\ \upsilon>\delta}} \psi_i(\upsilon) = +\infty.$$

Sa dérivée seconde ψ_i'' est décroissante et satisfait

$$\lim_{\upsilon\to+\infty} \psi_i''(\upsilon) = 0;$$

(iii) *si $i \notin \mathbb{I}$, alors il existe $\alpha_i \in [0,+\infty[$ tel que, pour tout $\upsilon \in [\delta,+\infty[$, $\psi_i(\upsilon) = \alpha_i\upsilon$.*

En se servant des hypothèses 3.41(ii) et (iii), il est clair que les fonctions $(\psi_i)_{1\leq i\leq N}$ sont convexes puisque, pour tout $i \in \mathbb{I}$, et pour tout $\upsilon \in]\delta,+\infty[$, $\psi_i''(\upsilon) \geq 0$. De plus, pour tout $i \in \mathbb{I}$, l'hypothèse 3.41(ii) nous permet d'écrire

$$\lim_{\substack{\upsilon\to\delta\\ \upsilon>\delta}} \psi_i''(\upsilon) = +\infty. \tag{3.52}$$

Les fonctions ψ_i sont semi-continues inférieurement puisque, pour tout $i \in \{1,\ldots,N\}$, on a $\liminf_{\upsilon\to\delta}\psi_i(\upsilon) \geq \psi_i(\delta)$. Des exemples de telles fonctions seront proposées dans les paragraphes 3.6.3-a) et 3.6.3-b).

Si nous étudions maintenant la Lipschitz continuité du gradient de la fonction g définie par l'équation (3.47). Il en résulte que cette propriété est violée dès que Ψ est non finie. De plus, en considérant (3.52), le gradient de g n'est pas non plus garanti d'être Lipschitz continu sur $\operatorname{int}(\operatorname{dom} g)$. Pour contourner ce problème, nous pouvons remarquer

que, grâce à l'hypothèse 3.41(ii) et à l'équation (3.52), pour tout $i \in \mathbb{I}$, il existe une fonction décroissante $\upsilon_i : \ ]0,+\infty[\to]\delta,+\infty[$ telle que $\lim_{\theta\to+\infty} \upsilon_i(\theta) = \delta$ et

$$(\forall \theta \in \ ]0,+\infty[)(\forall \upsilon \in]\delta,+\infty[) \qquad 0 \leq \psi_i''(\upsilon) \leq \theta \Leftrightarrow \upsilon \geq \upsilon_i(\theta). \tag{3.53}$$

Des valeurs de $\upsilon_i(\theta)$ sont représentées sur la figure 3.4.

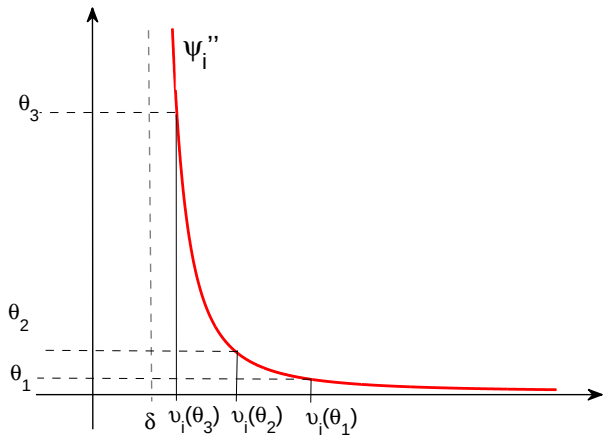


FIGURE 3.4 – Illustration de $\upsilon_i(\theta)$ pour $\theta_1 < \theta_2 < \theta_3$.

Considérons maintenant la fonction

$$g_\theta = \Psi_\theta \circ A \circ F^* \qquad \text{avec} \qquad \theta \in \ ]0,+\infty[, \tag{3.54}$$

où

$$\Big(\forall u = \big(u_i\big)_{1\leq i\leq N} \in \mathbb{R}^N\Big) \qquad \Psi_\theta(u) = \sum_{i=1}^{N} \psi_{\theta,i}\big(u_i\big),$$

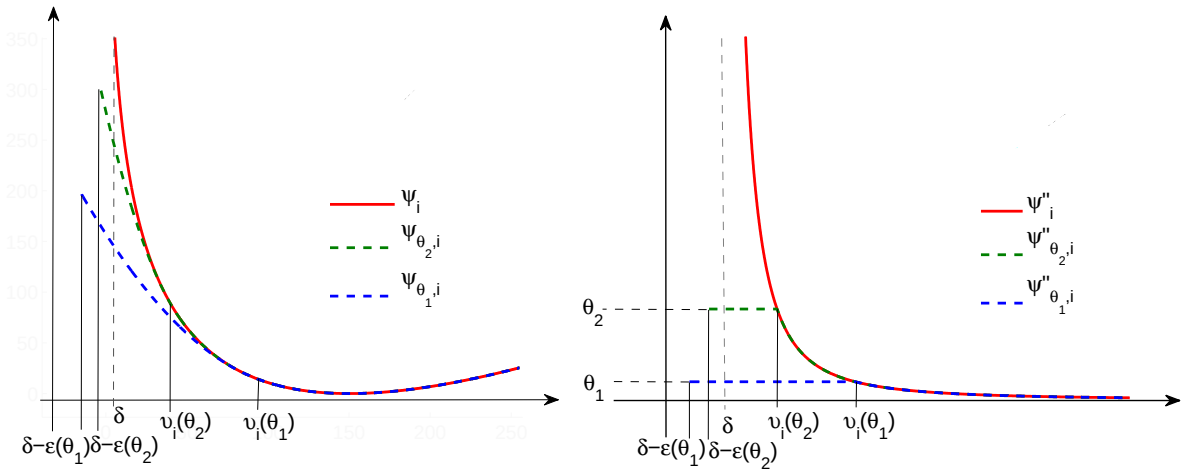
et les fonctions $(\psi_{\theta,i})_{1\leq i\leq N}$ sont choisies telles que

$$(\forall \upsilon \in \mathbb{R}) \qquad \psi_{\theta,i}(\upsilon) = \begin{cases} \dfrac{\theta}{2}\upsilon^2 + \zeta_{i,1}(\theta)\,\upsilon + \zeta_{i,0}(\theta) & \text{si } i \in \mathbb{I} \text{ et } \delta - \epsilon(\theta) \leq \upsilon < \upsilon_i(\theta) \\ \alpha_i \upsilon & \text{si } i \notin \mathbb{I} \text{ et } \delta - \epsilon(\theta) \leq \upsilon < \delta \\ \psi_i(\upsilon) & \text{sinon.} \end{cases} \tag{3.55}$$

De plus, $\epsilon : \ ]0,+\infty[\to \ ]0,+\infty[$ est une fonction décroissante et,

$$(\forall i \in \mathbb{I}) \qquad \begin{aligned} \zeta_{i,0}(\theta) &= \psi_i\big(\upsilon_i(\theta)\big) - \upsilon_i(\theta)\psi_i'\big(\upsilon_i(\theta)\big) + \frac{\theta}{2}\big(\upsilon_i(\theta)\big)^2 \\ \zeta_{i,1}(\theta) &= \psi_i'\big(\upsilon_i(\theta)\big) - \theta\upsilon_i(\theta). \end{aligned}$$

Pour tout $i \in \mathbb{I}$, les constantes $\zeta_{i,0}(\theta)$ et $\zeta_{i,1}(\theta)$ ont été déterminées pour garantir la continuité de $\psi_{\theta,i}$ et celle de ses dérivées première et seconde sur $\mathbb{R}$. Un exemple de fonction ψ_i, de ses approximations quadratiques ($\psi_{\theta_1,i}$ et $\psi_{\theta_2,i}$ avec $\theta_1 < \theta_2$) et des dérivées secondes associées, est présenté figure 3.5.



Fonction ψ_i et ses approximations $\psi_{\theta_1,i}$ et $\psi_{\theta_2,i}$

Dérivée seconde ψ_i'' et ses approximations $\psi_{\theta_1,i}''$ et $\psi_{\theta_2,i}''$

FIGURE 3.5 – Représentation graphique de l'extension quadratique pour $\delta = 0$.

Du cadre théorique que nous venons de définir, nous déduisons les résultats suivants.

Proposition 3.42 *Supposons que l'hypothèse 3.41 et la condition* (3.45) *soient vérifiées, alors*

(i) $(\forall \theta \in \,]0, +\infty[)$, *la fonction* g_θ *définie par l'équation* (3.54) *appartient à la classe* $\Gamma_0(\mathbb{R}^K)$.

(ii) $\left(\forall (\theta_1, \theta_2) \in \,]0, +\infty[^2\right)$, $\theta_1 < \theta_2 \Rightarrow g_{\theta_1} \leq g_{\theta_2} \leq g$.

(iii) *Pour tout* $\theta \in \,]0, +\infty[$, *si* $AF^*C \subset]\delta - \epsilon(\theta), +\infty[^N$, g_θ *est deux fois différentiable sur un ensemble ouvert* O_θ *de* $\mathbb{R}^N$ *tel que* $C \subset O_\theta$ *et possède un gradient Lipschitz continu sur* O_θ *avec une constante* $\beta_\theta = \theta \|AF^*\|^2 \leq \theta \overline{\nu} \|A\|^2$.

(iv) *Pour tout* $\theta \in \,]0, +\infty[$, *si* f *est coercive ou si* $\operatorname{dom} g_\theta \cap C$ *est borné, alors la minimisation de* $f + g_\theta + \iota_C$ *admet une solution. De plus, si* f *est strictement convexe sur* $\operatorname{dom} g_\theta \cap C$, *alors* $f + g_\theta + \iota_C$ *possède un unique minimiseur* $\widehat{x}_\theta$.

(v) *Supposons que*

(a) $\lim_{\theta\to+\infty}\epsilon(\theta)=0$,

(b) $AF^*C\subset[\delta,+\infty[^N$,

(c) *f est coercive ou C est borné,*

(d) *f est strictement convexe sur C.*

Il existe alors $\overline{\theta}\in\ ]0,+\infty[$ tel que, pour tout $\theta\in[\overline{\theta},+\infty[$, $\widehat{x}_\theta$ est le minimiseur de $f+g+\iota_{\mathrm{C}}$.

Preuve :
(i) $\Psi_\theta\in\Gamma_0(\mathbb{R}^N)$ car pour tout $i\in\{1,\ldots,N\}$ et pour tout $\upsilon\in]\delta-\epsilon(\theta),+\infty[$, $\psi''_{\theta,i}(\upsilon)\geq 0$ et $\operatorname{dom}\Psi_\theta=[\delta-\epsilon(\theta),+\infty[^N$. De plus, $\operatorname{dom}\Psi_\theta\cap\operatorname{Im}(AF^*)\supset\operatorname{dom}\Psi\cap\operatorname{Im}(AF^*)\neq\varnothing$. On a alors $g_\theta\in\Gamma_0(\mathbb{R}^K)$.
(ii) Nous déduisons de (3.53) et (3.55) que, pour tout $i\in\mathbb{I}$,

$$(\forall\upsilon\in]\delta,\upsilon_i(\theta_2)[)\qquad \psi''_i(\upsilon)>\psi''_{\theta_2,i}(\upsilon)=\theta_2.$$

Par conséquent, $\psi'_i-\psi'_{\theta_2,i}$ est une fonction strictement croissante sur $]\delta,\upsilon_i(\theta_2)]$ et

$$(\forall\upsilon\in]\delta,\upsilon_i(\theta_2)[)\qquad \psi'_i(\upsilon)-\psi'_{\theta_2,i}(\upsilon)<\psi'_i\big(\upsilon_i(\theta_2)\big)-\psi'_{\theta_2,i}\big(\upsilon_i(\theta_2)\big)=0,$$

ce qui permet d'écrire que $\psi_i-\psi_{\theta_2,i}$ est strictement décroissante sur $]\delta,\upsilon_i(\theta_2)]$ et

$$(\forall\upsilon\in]\delta,\upsilon_i(\theta_2)[)\qquad \psi_i(\upsilon)-\psi_{\theta_2,i}(\upsilon)>\psi_i\big(\upsilon_i(\theta_2)\big)-\psi_{\theta_2,i}\big(\upsilon_i(\theta_2)\big)=0.$$

De plus,

$$\begin{cases}\psi_i(\upsilon)=+\infty & \text{si } (i\in\mathbb{I}\text{ et }\upsilon\leq\delta)\text{ ou }(i\notin\mathbb{I}\text{ et }\upsilon<\delta),\\ \psi_i(\upsilon)=\psi_{\theta_2,i}(\upsilon) & \text{si }\big(i\in\mathbb{I}\text{ et }\upsilon\geq\upsilon_i(\theta_2)\big)\text{ ou }(i\notin\mathbb{I}\text{ et }\upsilon\geq\delta).\end{cases}$$

Nous en déduisons que, pour tout $i\in\{1,\ldots,N\}$, $\psi_i\geq\psi_{\theta_2,i}$ et, par conséquent g est approximée inférieurement par g_{θ_2}.
De façon similaire, nous avons, pour tout $i\in\mathbb{I}$,

$$\begin{array}{ll}(\forall\upsilon\in[\upsilon_i(\theta_1),+\infty[) & \psi_{\theta_2,i}(\upsilon)=\psi_i(\upsilon)=\psi_{\theta_1,i}(\upsilon)\\ (\forall\upsilon\in]\delta-\epsilon(\theta_2),\upsilon_i(\theta_1)[) & \psi''_{\theta_2,i}(\upsilon)>\theta_1=\psi''_{\theta_1,i}(\upsilon)\\ \Rightarrow\quad(\forall\upsilon\in]\delta-\epsilon(\theta_2),\upsilon_i(\theta_1)[) & \psi'_{\theta_2,i}(\upsilon)<\psi'_{\theta_1,i}(\upsilon)\\ \Rightarrow\quad(\forall\upsilon\in[\delta-\epsilon(\theta_2),\upsilon_i(\theta_1)[) & \psi_{\theta_2,i}(\upsilon)>\psi_{\theta_1,i}(\upsilon)\end{array}$$

et

$$(\forall i\notin\mathbb{I})(\forall\upsilon\in[\delta-\epsilon(\theta_2),+\infty[)\qquad \psi_{\theta_2,i}(\upsilon)=\psi_{\theta_1,i}(\upsilon).$$

De plus,

$$(\forall i\in\{1,\ldots,N\})(\forall\upsilon\in]-\infty,\delta-\epsilon(\theta_2)[)\qquad \psi_{\theta_2,i}(\upsilon)=+\infty\geq\psi_{\theta_1,i}(\upsilon).$$

Ceci montre que $\Psi_{\theta_2} \geq \Psi_{\theta_1}$ et par conséquent, $g_{\theta_2} \geq g_{\theta_1}$.

(iii) : Comme nous l'avons mentionné plus haut, $\operatorname{dom} \Psi_\theta = [\delta - \epsilon(\theta), +\infty[^N$. Soit

$$O_\theta = (AF^*)^{-1}(]\delta - \epsilon(\theta), +\infty[^N) = \big\{x \in \mathbb{R}^K \;\big|\; AF^*x \in]\delta - \epsilon(\theta), +\infty[^N\big\}.$$

Comme O_θ est un ensemble ouvert et $AF^*C \subset]\delta - \epsilon(\theta), +\infty[^N$, nous en déduisons que $C \subset O_\theta$. De plus, la fonction g_θ est deux fois différentiable sur O_θ et d'après [Ekeland, Témam, 1999, Chap. 1, Prop. 5.7] son gradient est,

$$(\forall x \in O_\theta) \qquad \nabla g_\theta(x) = FA^*\big(\nabla \Psi_\theta(AF^*x)\big) \tag{3.56}$$

où

$$\big(\forall u = (u_i)_{1\leq i\leq N} \in]\delta - \epsilon(\theta), +\infty[^N\big) \qquad \nabla \Psi_\theta(u) = \big(\psi'_{\theta,i}(u_i)\big)_{1\leq i\leq N}.$$

Nous pouvons alors écrire,

$$\big(\forall u = (u_i)_{1\leq i\leq N} \in]\delta - \epsilon(\theta), +\infty[^N\big)\big(\forall v = (v_i)_{1\leq i\leq N} \in]\delta - \epsilon(\theta), +\infty[^N\big)$$
$$\|\nabla \Psi_\theta(u) - \nabla \Psi_\theta(v)\| = \sqrt{\sum_{i=1}^N \big(\psi'_{\theta,i}(u_i) - \psi'_{\theta,i}(v_i)\big)^2}.$$

En utilisant le théorème des accroissements finis,

$$\begin{aligned}(\forall i \in \{1, \ldots, N\}) \quad \big|\psi'_{\theta,i}(u_i) - \psi'_{\theta,i}(v_i)\big| &\leq |u_i - v_i| \sup_{\xi \in]\delta-\epsilon(\theta),+\infty[} |\psi''_{\theta,i}(\xi)| \\ &\leq \theta |u_i - v_i|,\end{aligned}$$

nous obtenons

$$\big(\forall u \in]\delta - \epsilon(\theta), +\infty[^N\big)\big(\forall v \in]\delta - \epsilon(\theta), +\infty[^N\big) \qquad \|\nabla \Psi_\theta(u) - \nabla \Psi_\theta(v)\| \leq \theta \|u - v\|$$

que nous combinons avec (3.56) pour conclure que

$$\big(\forall (x, x') \in O_\theta^2\big) \qquad \|\nabla g_\theta(x) - \nabla g_\theta(x')\| \leq \theta \|AF^*\|^2 \|x - x'\|.$$

et $\|AF^*\|^2 \leq \|F\|^2 \|A\|^2 \leq \overline{\nu} \|A\|^2$.

(iv) : La preuve est similaire à celle de la proposition 3.36(ii). Par le biais des hypothèses 3.41(ii),

$$(\forall i \in \mathbb{I}) \qquad \inf_{\upsilon \in]\delta, +\infty[} \psi_{\theta,i}(\upsilon) > -\infty$$

et de l'hypothèse 3.41(iii),

$$(\forall i \notin \mathbb{I}) \qquad \inf_{\upsilon \in [\delta, +\infty[} \psi_i(\upsilon) = \alpha_i \delta > -\infty,$$

nous obtenons $\inf \Psi_\theta(\mathbb{R}^N) > -\infty$.

(v) : Dans la suite, nous utiliserons la notation : $h = f + g + \iota_{\mathrm{C}}$ et, pour tout $\theta \in \,]0, +\infty[$, $h_\theta = f + g_\theta + \iota_{\mathrm{C}}$.

Soit $(\theta_\ell)_{\ell\in\mathbb{N}}$ une suite croissante de $]0, +\infty[$ telle que $\lim_{\ell\to+\infty} \theta_\ell = +\infty$. Comme conséquence des propositions 3.42(i) et (ii), $(h_{\theta_\ell})_{\ell\in\mathbb{N}}$ est une suite croissante de fonctions de $\Gamma_0(\mathbb{R}^K)$. Nous déduisons de [Rockafellar, Wets, 2004, proposition 7.4(d)], rappelée par la proposition 2.20 dans le chapitre 2, que $(h_{\theta_\ell})_{\ell\in\mathbb{N}}$ epi-converge vers sa limite ponctuelle (« point-wise »). En utilisant conjointement l'équation (3.55) et le fait que, pour tout $i \in \mathbb{I}$, $\lim_{\theta\to+\infty} \upsilon_i(\theta) = \delta$ et $\lim_{\theta\to+\infty} \epsilon(\theta) = 0$, nous voyons que sa limite ponctuelle est h.

Pour tout $\ell \in \mathbb{N}$, on désigne par $\widehat{x}_{\theta_\ell}$ un minimiseur de h_{θ_ℓ}. D'après les hypothèses (v)(c), pour tout $\ell \in \mathbb{N}$, h_{θ_ℓ} est coercive puisque $C \cap \operatorname{dom} g_{\theta_\ell} = C$ est borné. De manière équivalente, on peut dire que ses sections inférieures, notés

$$\operatorname{lev}_{\leq\eta} h_{\theta_\ell} = \left\{x \in \mathbb{R}^K \;\middle|\; h_{\theta_\ell}(x) \leq \eta\right\}$$

avec $\eta \in \mathbb{R}$, sont également bornés. Comme $(h_{\theta_\ell})_{\ell\in\mathbb{N}}$ est une suite de fonctions croissantes, on en déduit que $\cup_{\ell\in\mathbb{N}}\operatorname{lev}_{\leq\eta} h_{\theta_\ell} = \operatorname{lev}_{\leq\eta} h_{\theta_0}$ est borné. Les fonctions $(h_{\theta_\ell})_{\ell\in\mathbb{N}}$ et h sont semi-continues inférieurement et propres. En combinant, l'hypothèse (v)(d), assurant la stricte convexité de f sur $(C \cap \operatorname{dom} g) \subset (C \cap \operatorname{dom} g_{\theta_\ell}) = C$, avec les propositions 3.36(ii) et 3.42(iv), nous en déduisons l'unicité du minimiseur $\widehat{x}$ de h. Puis, en faisant appel à [Rockafellar, Wets, 2004, théorème 7.33], rappelé par le théorème 2.26 dans le chapitre 2, nous montrons que la suite $(\widehat{x}_{\theta_\ell})_{\ell\in\mathbb{N}}$ converge vers un minimiseur $\widehat{x}$ de h.

On note, pour tout $x \in \mathbb{R}^K$ et $i \in \{1, \ldots, N\}$, $(AF^*x)_i$ la i-ème composante du vecteur AF^*x. Comme $\widehat{x} \in \operatorname{dom} h$, pour tout $i \in \mathbb{I}$, $(AF^*\widehat{x})_i \in \operatorname{dom} \psi_i =]\delta, +\infty[$. Puisque

$$\lim_{\ell\to+\infty} \widehat{x}_{\theta_\ell} = \widehat{x},$$

pour tout $i \in \mathbb{I}$, pour tout $\eta \in \,]0, +\infty[$, il existe $\ell_{\eta,i} \in \mathbb{N}$ telle que

$$\begin{aligned}(\forall \ell \in \mathbb{N}) \quad \ell \geq \ell_{\eta,i} \Rightarrow & |(AF^*\widehat{x}_{\theta_\ell})_i - (AF^*\widehat{x})_i| < \eta \\ \Rightarrow & (AF^*\widehat{x}_{\theta_\ell})_i > \min_{i\in\mathbb{I}}(AF^*\widehat{x})_i - \eta.\end{aligned}$$

En choisissant $\eta = \dfrac{\min_{i\in\mathbb{I}}(AF^*\widehat{x})_i - \delta}{2} > 0$ et $\ell_\eta = \max_{i\in\mathbb{I}} \ell_{\eta,i}$, nous en déduisons que

$$(\forall \ell \in \mathbb{N}) \quad \ell \geq \ell_\eta \Rightarrow (AF^*\widehat{x}_{\theta_\ell})_i \geq \underline{\upsilon} \tag{3.57}$$

où $\underline{\upsilon} = \dfrac{\delta + \min_{i\in\mathbb{I}}(AF^*\widehat{x})_i}{2} > \delta$. De plus, puisque

$$\lim_{\ell\to+\infty} \theta_\ell = +\infty \Rightarrow \lim_{\ell\to+\infty} \max_{i\in\mathbb{I}} \upsilon_i(\theta_\ell) = \delta,$$

il existe $\overline{\ell} \geq \ell_\eta$ telle que $(\forall i \in \mathbb{I})\ \upsilon_i(\theta_{\overline{\ell}}) \leq \underline{\upsilon}$. L'équation (3.55) implique que $(\forall i \in \mathbb{I})$ $(\forall \upsilon \in [\underline{\upsilon}, +\infty[)$, $\psi_{\theta_{\overline{\ell}},i}(\upsilon) = \psi_i(\upsilon)$.
Si, de plus, on définit l'ensemble

$$D = \big\{x \in \operatorname{dom} g \;\big|\; (\forall i \in \mathbb{I})\ (AF^*x)_i \in [\underline{\upsilon}, +\infty[\big\},$$

nous en déduisons que, pour tout $x \in D$, $h_{\theta_{\overline{\ell}}}(x) = h(x)$. De plus, selon l'hypothèse (v)(b), pour tout $\ell \in \mathbb{N}$, si $i \notin \mathbb{I}$,

$$(AF^*\widehat{x}_{\theta_{\overline{\ell}}})^{(i)} \in [\delta, +\infty[. \tag{3.58}$$

En combinant, les équations (3.57) et (3.58), $\widehat{x}_{\theta_{\overline{\ell}}}$ et $\widehat{x}$ appartiennent à D. Par conséquent, comme

$$\widehat{x}_{\theta_{\overline{\ell}}} = \arg\min_{x \in \mathbb{R}^K} h_{\theta_{\overline{\ell}}}(x),$$

nous avons : $h(\widehat{x}_{\theta_{\overline{\ell}}}) = h_{\theta_{\overline{\ell}}}(\widehat{x}_{\theta_{\overline{\ell}}}) \leq h_{\theta_{\overline{\ell}}}(\widehat{x}) = h(\widehat{x})$, ce qui montre que $\widehat{x}_{\theta_{\overline{\ell}}} = \widehat{x}$.
En considérant maintenant que $\theta \in [\theta_{\overline{\ell}}, +\infty[$, par le biais de (ii) nous obtenons : $h_{\theta_{\overline{\ell}}} \leq h_\theta \leq h$. Puisque, $h(\widehat{x}) = h_{\theta_{\overline{\ell}}}(\widehat{x}) \leq h_\theta(\widehat{x}) \leq h(\widehat{x})$, il en résulte que $h_\theta(\widehat{x}) = h(\widehat{x})$ et

$$(\forall x \in \mathbb{R}^K) \qquad h_\theta(x) \geq h_{\theta_{\overline{\ell}}}(x) \geq h_{\theta_{\overline{\ell}}}(\widehat{x}) = h(\widehat{x}).$$

Ceci nous permet de conclure que $\widehat{x}_\theta = \widehat{x}$ dès que $\theta \geq \theta_{\overline{\ell}} = \overline{\theta}$. □

Remarque 3.43

(i) *La proposition 3.42(ii) nous permet d'affirmer que g_θ (resp. $f + g_\theta + \iota_{\mathrm{C}}$), avec $\theta > 0$, constitue une approximation inférieure de g (resp. $f + g + \iota_{\mathrm{C}}$), qui s'affine lorsque θ croît.*

(ii) *Comme le montre la proposition 3.42(iii), le rôle principal du paramètre θ est de contrôler la constante de Lipschitz du gradient de l'approximation de g.*

(iii) *La proposition 3.42(v) indique que le paramètre θ permet de contrôler la distance entre l'approximation $\widehat{x}_\theta$ et le minimiseur original du critère MAP $\widehat{x}$. Cette approximation devient parfaite lorsque la valeur de θ est supérieure à une certaine valeur $\overline{\theta}$.*

Sous les hypothèses 3.42(iii), la minimisation de $f + g_\theta + \iota_{\mathrm{C}}$ avec $\theta \in\ ]0, +\infty[$ est un problème du type problème 3.33. Par conséquent, les propositions 3.32, 3.34 et 3.35 montrent que, sous l'hypothèse que f soit coercive ou C borné, les algorithmes DR[FB], FB[DR] et FB[Dyk] peuvent être appliqués dans ce contexte.

De plus, la proposition 3.42(v) suggère qu'en choisissant θ suffisamment grand, la solution du critère MAP original peut être trouvée. Cependant, selon la proposition 3.42(iii), une grande valeur θ induit une grande valeur de la constante β_θ, ce qui induit une petite valeur pour le « pas » de l'algorithme explicite-implicite, et donc une vitesse de convergence lente. En pratique, le choix de θ résulte d'un compromis que nous illustrons dans les résultats expérimentaux qui suivent.

3.6.3-a) Premier exemple : déconvolution en présence d'un bruit Gaussien de variance dépendant du signal

• **Modèle** :
Nous voulons restaurer une image $\overline{y} \in [0, +\infty[^N$ dégradée par un opérateur linéaire A : $\mathbb{R}^N \to \mathbb{R}^N$ et un bruit additif $w \in \mathbb{R}^N$, à partir des observations

$$z = A\overline{y} + w = \overline{u} + w.$$

De plus, on suppose que la matrice associée à l'opérateur linéaire A ne possède pas d'éléments négatifs et, $w = (w_i)_{1 \leq i \leq N}$ est une réalisation d'un bruit gaussien $W = (W_i)_{1 \leq i \leq N}$ de moyenne nulle. La variance de chaque variable aléatoire W_i avec $i \in \{1, \ldots, N\}$ dépend du signal et elle est égale à $\sigma_i^2(\overline{u}_i)$ où

$$(\forall \upsilon \in]0, +\infty[) \qquad \sigma_i^2(\upsilon) = \frac{\upsilon}{2\alpha_i}$$

avec $\alpha_i \in \ ]0, +\infty[$. Les fonctions $(\psi_i)_{1 \leq i \leq N}$, définies par (3.43), sont,

$$(\forall \upsilon \in \mathbb{R}) \qquad \psi_i(\upsilon) = \begin{cases} \dfrac{\alpha_i \big(\upsilon - z_i\big)^2}{\upsilon} & \text{si } z_i \neq 0 \text{ et } \upsilon \in \ ]0, +\infty[, \\ \alpha_i \upsilon & \text{si } z_i = 0 \text{ et } \upsilon \in [0, +\infty[, \\ +\infty & \text{sinon.} \end{cases}$$

En supposant que $z \neq 0$, l'hypothèse 3.41 est donc satisfaite avec $\delta = 0$ et $\mathbb{I} = \big\{ i \in \{1, \ldots, N\} \bigm| z_i \neq 0 \big\}$ puisque, pour tout $i \in \mathbb{I}$,

$$(\forall \upsilon \in \ ]0, +\infty[) \qquad \begin{aligned} \psi_i'(\upsilon) &= \alpha_i \frac{\upsilon^2 - z_i^2}{\upsilon^2} \\ \psi_i''(\upsilon) &= \frac{2\alpha_i z_i^2}{\upsilon^3}. \end{aligned}$$

Nous déduisons de (3.53) que, pour tout $i \in \mathbb{I}$,

$$(\forall \theta \in \ ]0, +\infty[) \qquad \upsilon_i(\theta) = \Big(\frac{2\alpha_i z_i^2}{\theta} \Big)^{1/3}.$$

• **Résultats expérimentaux** :
Dans nos simulations, A représente un flou uniforme de taille 3×3 ou de taille 7×7 avec $\|A\| = 1$. L'image satellitaire $\overline{y}$ de taille 512×512 ($N = 512^2$) est présentée figure 3.6(a). Elle a été dégradée par A et un bruit additif dépendant du signal suivant le modèle décrit précédemment avec pour tout $i \in \{1, \ldots, N\}$, $\alpha_i \equiv 1$ ou $\alpha_i \equiv 5$. L'image dégradée z présentée figure 3.6(b) correspond à une image dégradée par un flou uniforme dont le noyau du filtre est de taille 7×7 et $\alpha_i \equiv 1$.

Dans cet exemple, une représentation sur une trame ajustée de redondance 2 a été utilisée. Il s'agit d'une représentation en arbre dual [Chaux *et al.*, 2006] ($\underline{\nu} = \overline{\nu} = 2$, $K = 2N$) avec des filtres de symmlets de longueur 6 [Daubechies, 1992]. Les fonctions potentielles ϕ_k sont prises de la forme $\chi_k|\,.\,| + \omega_k|\,.\,|^{p_k}$ où $(\chi_k, \omega_k) \in \,]0, +\infty[^2$ et $p_k \in \{4/3, 3/2, 2\}$ et sont adaptées à chaque sous-bande. Ces paramètres ont été déterminés par une approche du maximum de vraisemblance. Par conséquent, la fonction f définie par (3.46) est coercive et strictement convexe (voir remarque 3.37).

Une contrainte sur la solution est introduite afin de prendre en compte l'intervalle des valeurs admissibles dans l'image en choisissant

$$C^* = [0, 255]^N. \tag{3.59}$$

Grâce à la forme de l'opérateur A et de la contrainte C^*, la condition (3.45) est satisfaite. La proposition 3.36 garantit donc une unique solution fournissant l'estimateur $\widehat{x}$ du critère MAP. Selon la proposition 3.42(iv), pour tout $\theta \in \,]0, +\infty[$, un unique minimiseur $\widehat{x}_\theta$ de $f + g_\theta + \iota_C$ existe et permet d'approcher $\widehat{x}$ comme expliqué dans la proposition 3.42(v).

Pour tout $\theta \in \,]0, +\infty[$, $AC^* = C^* \subset]-\epsilon(\theta), +\infty[^N$, et la proposition 3.42(iii) montre que g_θ a un gradient Lipschitz continu sur C. De plus g_θ est deux fois différentiable ce qui nous ramène au problème 3.33 pouvant être résolu par les algorithmes DR[FB], FB[DR] et FB[Dyk]. Chacun de ces algorithmes converge vers une solution notée $\widehat{x}_\theta$.

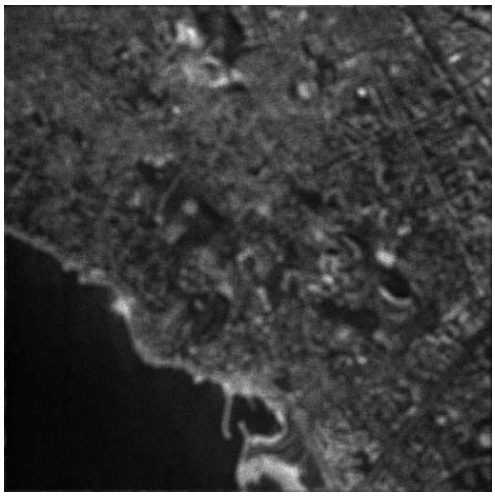
Les résultats sont générés avec l'algorithme DR[FB], le nombre d'itérations de l'algorithme explicite-implicite étant donné par (3.50) avec $\eta = 10^{-4}$. L'algorithme a été initialisé en prenant $v_0 = P_C(Fz)$ et nous avons choisi $\gamma_{m,n} \equiv 1.99/(\kappa\theta)$, $\kappa = 60$ et $\lambda_{m,n} \equiv \tau_m \equiv 1$. Il faut noter que les algorithmes FB[DR] ou FB[Dyk] conduisent aux mêmes résultats pour des temps de calculs plus importants. Pour évaluer la performance de notre algorithme en termes de qualité de restauration, nous utilisons le rapport signal sur bruit (SNR). Le tableau 3.3 montre des valeurs de SNR obtenues.

		Flou 3×3				Flou 7×7			
$\alpha_i = 1$	θ	0.025	0.05	5	7	0.025	0.05	5	7
	SNR	13.9	16.3	**16.8**	16.8	10.9	11.9	**12.1**	12.1
$\alpha_i = 5$	θ	0.15	0.25	10	12	0.15	0.25	10	12
	SNR	15.9	18.0	**18.8**	18.8	12.6	13.3	**13.7**	13.7

TABLE 3.2 – SNR pour l'image satellitaire.

Comme il l'a été suggéré dans la proposition 3.42(v), lorsque la valeur de θ augmente, l'image est mieux restaurée. L'efficacité de l'approche proposée peut également être évaluée visuellement sur la figure 3.6(c) montrant l'image restaurée quand A est la matrice associée à un flou uniforme de taille 7×7, $\alpha_i \equiv 1$ et $\theta = 0.05$. Nous pouvons observer que cette approche permet de retrouver la plupart des détails qui ne sont pas perceptibles à cause de la présence de flou et de bruit.

(a)



(b)

(c)

FIGURE 3.6 – Résultats pour une image satellitaire de la ville de Marseille. (a) Image originale, (b) image dégradée, (c) image restaurée en utilisant des DTT.

Finalement la figure 3.7 illustre l'influence du choix du paramètre θ quand l'algorithme DR[FB] est utilisé pour un flou de taille 7×7 et $\alpha_i \equiv 5$. Comme attendu, plus

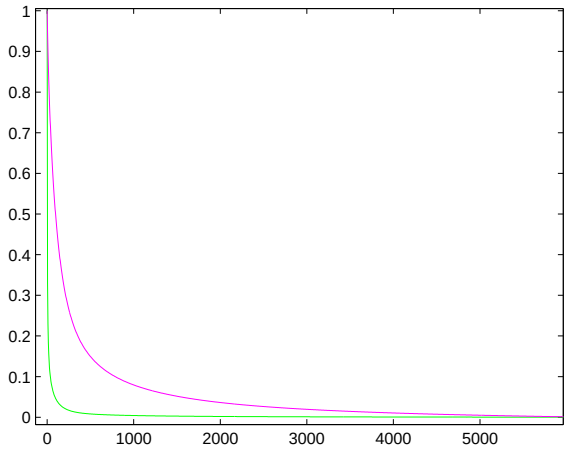


FIGURE 3.7 – Critère MAP normalisé (pour $\theta = 0.15$ en vert et $\theta = 10$ en magenta) versus le temps de calcul (en secondes) (Intel Core 2, 2.66 GHz).

la valeur de θ est grande et plus la vitesse de convergence de l'algorithme est lente. Un ajustement doit par conséquent être effectué : la valeur de θ doit être choisie suffisamment grande pour assurer une bonne qualité de restauration mais elle ne doit pas être trop grande de manière à conserver une convergence rapide.

3.6.3-b) Deuxième exemple : gestion d'une fonction à la dérivée seconde non bornée - déconvolution en présence de bruit de Poisson

• **Modèle** :
Dans ce deuxième scénario, nous voulons restaurer une image $\overline{y} \in [0, +\infty[^N$ qui est dégradée par un opérateur linéaire $A : \mathbb{R}^N \to \mathbb{R}^N$, avec des valeurs supposées positives et qui sont dégradées par un bruit de Poisson (pouvant être non homogène). L'image observée $z = (z_i)_{1 \leq i \leq N} \in \mathbb{N}^N$ suit une distribution de Poisson avec une loi de probabilité donnée par

$$(\forall i \in \{1, \ldots, N\})(\forall \upsilon \in [0, +\infty[) \qquad \mu_{Z_i|\overline{U}_i=\upsilon}(z_i) = \frac{(\alpha_i \upsilon)^{z_i}}{z_i!} \exp\big(-\alpha_i \upsilon\big) \tag{3.60}$$

où $(\alpha_i)_{1\leq i\leq N} \in \ ]0,+\infty[^N$ sont des paramètres d'échelle. En utilisant (3.43) et (3.60), pour tout $i \in \{1,\ldots,N\}$, nous avons,

$$(\forall \upsilon \in \mathbb{R}) \qquad \psi_i(\upsilon) = \begin{cases} \alpha_i\upsilon - z_i + z_i \ln\left(\dfrac{z_i}{\alpha_i\upsilon}\right) & \text{si } z_i > 0 \text{ et } \upsilon \in \ ]0,+\infty[, \\ \alpha_i\upsilon & \text{si } z_i = 0 \text{ et } \upsilon \in [0,+\infty[, \\ +\infty & \text{sinon.} \end{cases} \tag{3.61}$$

Comme les fonctions $(\psi_i)_{1\leq i\leq N}$ sont définies à des constantes près, ces constantes ont été choisies dans (4.77) pour obtenir l'expression du terme de la divergence de Kullback-Leibler généralisée (cf. [Byrne, 1993]).
Dans ce contexte, lorsque $z \neq 0$, l'hypothèse 3.41 est vérifiée quand $\delta = 0$ et $\mathbb{I} = \left\{i \in \{1,\ldots,N\} \;\middle|\; z_i > 0\right\}$ puisque, pour tout $i \in \mathbb{I}$,

$$(\forall \upsilon \in \ ]0,+\infty[) \qquad \begin{aligned} \psi_i'(\upsilon) &= \alpha_i - \frac{z_i}{\upsilon} \\ \psi_i''(\upsilon) &= \frac{z_i}{\upsilon^2}. \end{aligned}$$

Nous déduisons de (3.53) que, pour tout $i \in \mathbb{I}$,

$$(\forall \theta \in \ ]0,+\infty[) \qquad \upsilon_i(\theta) = \sqrt{\frac{z_i}{\theta}}.$$

Remarque 3.44 *Il peut être intéressant de comparer l'extension proposée avec l'approche développée dans [Dupé* et al.*, 2009]. L'utilisation de la transformée de Anscombe [Anscombe, 1948], faite dans [Dupé* et al.*, 2009], permet d'approximer l'antilog-vraisemblance poissonienne ψ_i par*

$$(\forall \upsilon \in \mathbb{R}) \qquad \widetilde{\psi}_i(\upsilon) = \begin{cases} \frac{1}{2}\left(2\sqrt{\alpha_i\upsilon + \frac{3}{8}} - 2\sqrt{z^{(i)} + \frac{3}{8}}\right)^2 & \textit{si } \upsilon \in [0,+\infty[\\ +\infty & \textit{sinon.} \end{cases} \tag{3.62}$$

L'extension quadratique proposée est illustrée par la figure 3.8 où une comparaison graphique avec l'approximation de Anscombe est menée.

• **Résultats expérimentaux** :
Ici, A représente la matrice associée à un flou uniforme de taille 5×5 avec $\|A\| = 1$. $\overline{y}$ correspond à une coupe cérébrale de taille 256×256 ($N = 256^2$). Elle est présentée sur la figure 3.9(a). Cette image est dégradée par A et par un bruit de Poisson suivant le modèle décrit dans le paragraphe précédent pour différents niveaux d'intensité. L'image dégradée z est présentée figure 3.9(b) quand $\alpha_i \equiv 0.01$.

Une base orthonormale d'ondelettes constituée de filtres de symmlets de longueur 6 a été adoptée ($\underline{\nu} = \overline{\nu} = 1$, $K = N$). Les fonctions potentielles ϕ_k sont choisies de la même

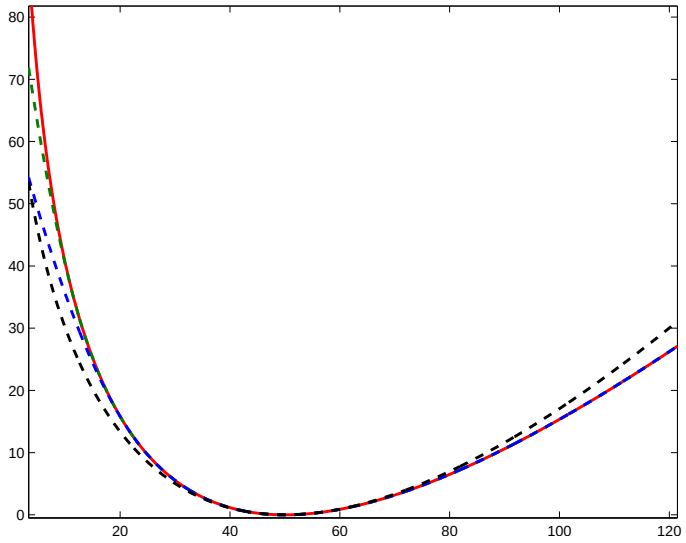


FIGURE 3.8 – Graphe de la fonction ψ_i (ligne continue rouge) quand $\delta = 0$, $\alpha_i = 1$, $z_i = 50$. Son approximation quadratique $\psi_{\theta,i}$ avec $\theta = 0.3$ (ligne discontinue verte) et $\theta = 0.1$ (ligne discontinue bleue) pour $\epsilon(\theta) = 10^{-16}$ et son approximation de Anscombe $\widetilde{\psi}_i$ (ligne discontinue noire).

forme que dans les exemples précédents et la fonction f est par conséquent coercive et strictement convexe.

La contrainte imposée sur la solution est donnée par l'équation (3.59). Comme $AC^* = C^*$, la proposition 3.42(iv) garantit qu'il existe un unique minimiseur de $f + g_\theta + \iota_C$, qui a été calculé avec l'algorithme DR[FB]. L'algorithme a été initialisé en prenant $v_0 = P_C(Fz)$ et nous avons choisi $\gamma_{m,n} \equiv 1.99/(\kappa\theta)$, $\kappa = 60$ et $\lambda_{m,n} \equiv \tau_m \equiv 1$. Le nombre d'itérations de l'algorithme explicite-implicite est donné par (3.50) avec $\eta = 10^{-4}$. Notons que la vitesse de convergence pourrait être accélérée un utilisant une méthode de pas adaptatif telle que la recherche d'Armijo-Goldstein [Tseng, 2000; Dupé *et al.*, 2009]. Cependant, le temps de calcul pour déterminer le « pas » optimal devrait être pris en compte.

Pour évaluer la performance de notre algorithme, nous utilisons le SNR. Le tableau 3.3 montre des valeurs de SNR obtenues pour différentes valeurs de α_i et θ. Conformément aux prévisions de la proposition 3.42(v), à partir de certaines valeurs de θ dépendantes de α_i, la valeur optimale est obtenue.

Nous comparons également les résultats fournis par différentes approches. La première est un EM régularisé (également connue sous le nom de SMART) [Byrne, 1993; Lange *et al.*, 1987] où l'antilog-vraisemblance est pénalisée par un terme de divergence de Kullback-Leibler entre l'image désirée et une image de référence. Le paramètre de régularisation a été ajusté manuellement pour maximiser le SNR et, l'image de référence est une image dont les pixels prennent la valeur moyenne de l'image dégradée. La seconde

approche est basée sur l'approximation de Anscombe [Dupé *et al.*, 2009] que nous avons introduite dans la remarque 3.44. Dans l'optique d'effectuer des comparaisons justes, la même représentation en ondelettes, les mêmes fonctions de régularisation $(\phi_k)_{1\leq k\leq K}$ et le même convexe C sont utilisés dans cette exemple. Il peut être observé que l'approche que nous proposons donne de bons résultats. Cependant, pour de grandes intensités ($\alpha_i \geq 0.1$), la méthode utilisant l'approximation de Anscombe conduit à des résultats équivalents. Les images restaurées sont présentées sur la figure 3.9, pour $\alpha_i \equiv 0.01$, $\theta \equiv 0.001$ et après 3000 itérations. Malgré une forte dégradation de l'image originale, nous pouvons remarquer que notre méthode permet de reconstruire les motifs principaux.

α_i	EM régularisé	Anscombe	Extension quadratique				
			$\theta = 0.001$	$\theta = 0.005$	$\theta = 0.1$	$\theta = 1$	$\theta = 5$
0.01	6.47	8.24	**9.75**	9.75	9.75	9.75	9.75
0.05	9.01	11.5	11.7	**11.9**	11.9	11.9	11.9
0.1	10.1	12.4	12.0	**12.5**	12.5	12.5	12.5
1	13.8	**15.1**	0	10.1	13.7	**15.1**	15.1

TABLE 3.3 – SNR pour l'image médicale.

3.7 Conclusion

Ce chapitre décrit deux types de travaux effectués durant la thèse. D'une part, nous avons introduit des algorithmes imbriqués en donnant leur preuve de convergence et en les comparant. D'autre part, nous nous sommes intéressés à l'utilisation de ces algorithmes pour la résolution de problématiques de restauration d'images, en particulier la déconvolution d'images en présence de bruit de Poisson. Ce problème nécessite de définir un cadre théorique rigoureux où une extension quadratique est utilisée pour s'assurer de la convergence des algorithmes imbriqués. Les résultats obtenus montrent une amélioration de la restauration par rapport à l'état de l'art.

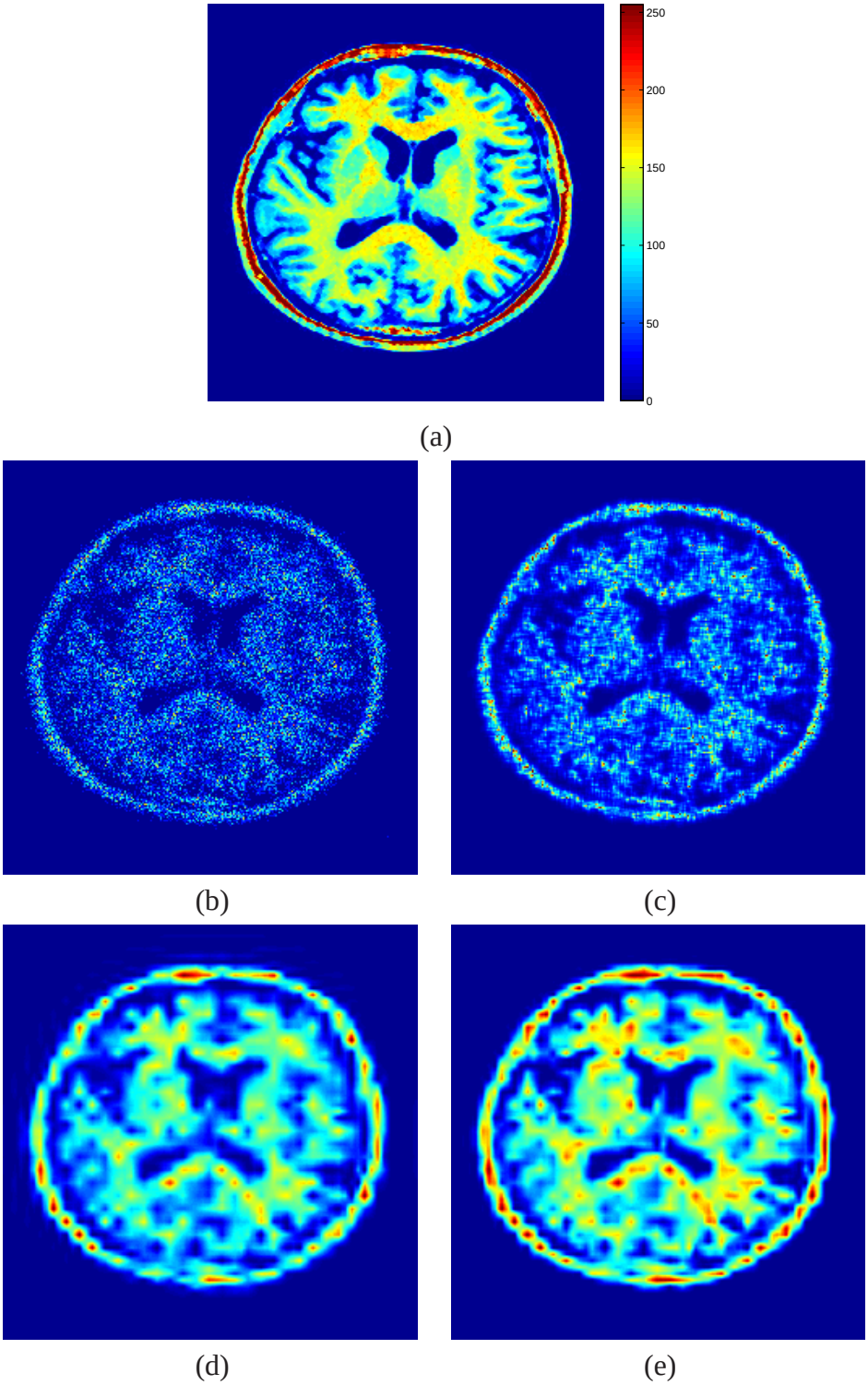


FIGURE 3.9 – Résultats sur une image médicale. (a) Originale, (b) dégradée, (c) restaurée avec EM, (d) restaurée par transformée de Anscombe et (e) restaurée par extension quadratique.

CHAPITRE 4

Algorithme proximal parallèle et régularisation hybride pour la résolution de problèmes inverses en présence de bruit non-additif

4.1 Motivations

Comme nous l'avons rappelé dans le chapitre 2, les méthodes d'optimisation convexe ont montré leur efficacité pour la résolution de problèmes inverses. Nous disposons d'une part, des algorithmes de type POCS et des algorithmes par blocs [Combettes, 2003], qui permettent, entre autre, d'inclure des contraintes de variation totale. D'autre part, certaines méthodes de débruitage sont fondées sur les transformées en ondelettes [Mallat, 1997], et plus généralement sur des trames [Candès, Donoho, 2002; Le Pennec, Mallat, 2005; Selesnick *et al.*, 2005; Chaux *et al.*, 2006]. Dans le chapitre 3, nous avons rappelé que les algorithmes appartenant à la classe des algorithmes explicite-implicite [Daubechies *et al.*, 2004; Figueiredo, Nowak, 2003; Bect *et al.*, 2004; Combettes, Wajs, 2005] permettent de minimiser une somme de deux fonctions appartenant à la classe $\Gamma_0(\mathcal{H})$ sous condition qu'une des deux fonctions soit de gradient Lipschitz sur $\mathcal{H}$. Ces méthodes sont efficaces lorsqu'il s'agit de minimiser un critère composé d'un terme de fidélité aux données lisse, par exemple une fonction quadratique, et un *a priori* non lisse, tel qu'une norme ℓ_1, permettant de favoriser la parcimonie dans les coefficients de trame [Bioucas-Dias, Figueiredo, 2007; Beck, Teboulle, 2009]. Ces méthodes sont donc adaptées à la restauration d'images dégradées par une convolution et du bruit gaussien. Dans le chapitre 3, nous avons également présenté l'algorithme de Douglas-Rachford [Lions, Mercier, 1979; Douglas, Rachford, 1956; Eckstein, Bertekas, 1992], proposé dans [Combettes, Pesquet, 2007a], qui permet de résoudre des problèmes de restauration d'images en relachant la condition de Lipschitz différentiabilité imposée par l'algorithme explicite-implicite. En contrepartie la connaissance explicite des opérateurs proximaux de chaque fonction est requise. Cet algorithme est étendu à la minimisation d'une somme finie de fonctions convexes dans [Combettes, Pesquet, 2008]. Il est fondé sur le calcul de l'opérateur proximal associé à chacune des fonctions. L'un des principaux avantages de cet algorithme, appelé « algorithme proximal parallèle » (PPXA : Parallel ProXimal Algorithm), est sa structure

parallèle qui permet une implantation simple, sur une architecture multicoeurs. PPXA est bien adapté aux problèmes de déconvolution en présence de bruit additif gaussien, où l'opérateur proximal associé au terme de fidélité aux données prend une forme explicite [Combettes, Pesquet, 2008]. Dans ce type d'application, cet algorithme permet d'inclure facilement des contraintes supplémentaires dans le critère, et ainsi d'obtenir de meilleurs résultats en déconvolution. Pour minimiser une somme de deux fonctions incluant un terme quadratique, une autre classe d'algorithme parallèle a été récemment proposé par Fornasier *et al.* dans [Fornasier, 2007; Fornasier, Schönlieb, 2009]. Quand le bruit est non-nécessairement additif gaussien et pour une classe plus large d'opérateurs linéaires, l'opérateur proximal associé au terme de fidélité aux données ne possède pas de forme explicite, ce qui conduit à une limitation dans l'utilisation de PPXA ; de plus, les algorithmes dans [Fornasier, 2007] et [Fornasier, Schönlieb, 2009] ne sont pas applicables. Par conséquent, d'autre solutions doivent être proposées.

Pour une image dégradée par un opérateur linéaire et du bruit de Poisson, une première solution est de considérer la transformée de Anscombe [Dupé *et al.*, 2009]. Une seconde est l'approximation quadratique du terme d'attache aux données poissonien, proposée dans le chapitre 3. Les deux approches font appel à l'utilisation d'algorithmes proximaux imbriqués [Dupé *et al.*, 2009; Chaux *et al.*, 2009]. L'utilisation de tels algorithmes peut apparaître limitée pour deux raisons principales : la parallélisation des itérations semble difficile, et le nombre de fonctions à minimiser est en pratique limité à trois. Plus récemment, des approches liées aux techniques de « lagrangien augmenté » [Hestenes, 1969; Fortin, Glowinski, 1983] ont été considérées dans [Goldstein, Osher, 2008; Setzer *et al.*, 2010; Afonso *et al.*, 2010]. Ces méthodes sont bien adaptées quand l'opérateur linéaire est un opérateur de convolution, en utilisant des techniques de diagonalisation dans le domaine de Fourier. Pour une classe plus large d'opérateurs linéaires, un système linéaire de grande taille doit être résolu numériquement à chaque itération de l'algorithme.

L'objectif du présent chapitre est de proposer une adaptation de PPXA pour minimiser un critère plus général, applicable à une large classe de problèmes de restauration, tels que des données dégradées par un opérateur de convolution ou un opérateur de convolution décimée, en présence d'un bruit non-nécessairement additif gaussien (bruit de Poisson, bruit de Laplace,. . .). Une convolution décimée se rencontre en particulier dans des problèmes de super-résolution. Pour parvenir à cet objectif, lorsque l'opérateur proximal ne peut pas être calculé explicitement, nous montrerons qu'une approche par décomposition permettra de contourner cette difficulté. Ceci est la principale contribution de ce chapitre.

Les travaux de déconvolution en présence de bruit gaussien présentés dans [Bect *et al.*, 2004; Combettes, Pesquet, 2008; Bioucas-Dias, Figueiredo, 2008; Wen *et al.*, 2008], introduisent une régularisation hybride composée d'un terme de variation totale et d'un terme favorisant la parcimonie, permettant de faire bénéficier la solution des propriétés de chaque terme de régularisation. Nous considèrerons ce type de régularisation hybride dans notre cadre général de restauration, en étudiant différentes formes discrètes de variation totale et en utilisant des techniques de décomposition pour calculer les opérateurs

proximaux associés.

Ce chapitre présentera tout d'abord, une forme générale de critère que nous chercherons à minimiser. Dans le paragraphe 4.3, seront présentées des propriétés liées à l'opérateur proximal, complémentaires de celles décrites dans le paragraphe 3.2 du chapitre 3. Nous présenterons également dans ce paragraphe des techniques pour calculer l'opérateur proximal associé :

– au terme de fidélité aux données dans un contexte général de restauration,
– à diverses versions de la variation totale discrétisée.

Dans le paragraphe 4.4, nous proposons une version accélérée de l'algorihme PPXA qui permet de résoudre efficacement des problèmes de restauration d'images utilisant des trames. Le paragraphe 4.5 illustre les résultats théoriques développés dans ce chapitre dans le cadre des problèmes de déconvolution ou de super-résolution en présence de bruit de Poisson, lorsqu'un terme de régularisation hybride est utilisé.

4.2 Formulation du problème

Dans ce chapitre, nous nous intéressons à la résolution du problème d'optimisation convexe suivant :

$$\text{Trouver} \quad \widehat{x} \in \underset{x \in \mathbb{R}^K}{\text{Argmin}} \quad \sum_{j=1}^{J} f_j(x). \tag{4.1}$$

$x = (x_k)_{1 \leq k \leq K} \in \mathbb{R}^K$ représente les coefficients d'une trame ajustée, dont l'opérateur de synthèse de trame est noté $F^\top \in \mathbb{R}^{N \times K}$ (avec $K \geq N$). Les fonctions $(f_j)_{1 \leq j \leq J}$ appartiennent à la classe $\Gamma_0(\mathbb{R}^K)$. L'image restaurée est alors notée par $\widehat{y} = F^\top \widehat{x} \in \mathbb{R}^N$.

Un cas particulier bien connu de l'équation (4.1) est celui où $J = 2$; le problème de minimisation se réduit alors à la minimisation d'une somme de deux fonctions, qui dans un cadre Bayésien, peuvent être interprétées comme un terme d'attache aux données f_1, lié au bruit, et un terme f_2, lié à un modèle probabiliste *a priori* sur les coefficients de trame (cf. paragraphe 2.2.3-b) du chapitre 2). Dans ce chapitre nous nous intéressons plus particulièrement au cas où $J > 2$. Cette configuration permet par exemple :

– d'imposer des contraintes additionnelles sur l'image cible (contrainte sur la dynamique, contrainte de support,. . .),
– d'introduire un terme de régularisation hybride (par exemple : $\ell_1 + \mathrm{tv}$).

Le fait de considérer une représentation sur une trame, nous conduit ainsi à reformuler le problème d'optimisation convexe (4.1) comme :

$$\text{Trouver} \quad \widehat{x} \in \underset{x \in \mathbb{R}^K}{\text{Argmin}} \quad \sum_{j=1}^{S} g_j(F^\top x) + \sum_{j=S+1}^{J} f_j(x), \tag{4.2}$$

où les fonctions $(g_j)_{1\leq j\leq S}$ appartiennent à la classe $\Gamma_0(\mathbb{R}^N)$ et $(f_j)_{S+1\leq j\leq J}$ sont des fonctions de $\Gamma_0(\mathbb{R}^K)$, associées respectivement aux pixels de l'image ou aux coefficients de trame. Les termes associés à $j \in \{1,\ldots,S\}$ sont directement liés aux valeurs des pixels et peuvent être, entre autre, un terme de fidélité aux données ou une contrainte sur la dynamique des pixels. Les fonctions d'indices $j \in \{S+1,\ldots,J\}$, définies sur les coefficients de trame, sont souvent choisies à partir de modèles probabilistes classiques. Ils peuvent par exemple correspondre à l'anti-log vraisemblance de variables indépendantes suivant des distributions gaussiennes généralisées [Do, Vetterli, 2002].

4.3 Formes explicites de quelques opérateurs proximaux

4.3.1 Opérateurs proximaux de fonctions usuelles

Nous commençons par rappeler le calcul de l'opérateur proximal de la norme euclidienne en dimension 2, qui nous sera utile pour le calcul de l'opérateur proximal de la variation totale.

Exemple 4.1 [Combettes, Pesquet, 2008] *Soit $\mu > 0$, et soit*

$$\varphi\colon \mathbb{R}^2 \to \mathbb{R}\colon (\xi_1,\xi_2) \mapsto \mu\sqrt{|\xi_1|^2+|\xi_2|^2}.$$

On a alors, pour tout $(\xi_1,\xi_2) \in \mathbb{R}^2$,

$$\operatorname{prox}_\varphi(\xi_1,\xi_2) = \begin{cases} \left(1-\frac{\mu}{\sqrt{|\xi_1|^2+|\xi_2|^2}}\right)(\xi_1,\xi_2), & \textit{si}\ \ \sqrt{|\xi_1|^2+|\xi_2|^2} > \mu; \\ (0,0), & \textit{sinon}. \end{cases} \tag{4.3}$$

Dans la partie expérimentale, les données seront dégradées par un bruit de Poisson. Il nous sera donc utile de connaître l'opérateur proximal associé à la fonction gamma, cette dernière étant liée à l'antilog vraisemblance d'une loi poisonnienne.

Exemple 4.2 [Combettes, Pesquet, 2007a] *Soit $\alpha > 0$, et soit*

$$\varphi\colon \mathbb{R} \to\ ]-\infty,+\infty] : \xi \mapsto \begin{cases} -\chi\ln(\xi)+\alpha\xi, & \textit{si}\ \ \chi>0\ \ \textit{et}\ \ \xi>0, \\ \alpha\xi, & \textit{si}\ \ \chi=0\ \ \textit{et}\ \ \xi\geq 0, \\ +\infty, & \textit{sinon}. \end{cases} \tag{4.4}$$

On a alors, pour tout $\xi \in \mathbb{R}$,

$$\operatorname{prox}_\varphi \xi = \frac{\xi-\alpha+\sqrt{|\xi-\alpha|^2+4\chi}}{2}. \tag{4.5}$$

4.3.2 Opérateur proximal impliquant un opérateur linéaire

Dans ce paragraphe, notre intérêt porte sur le calcul de l'opérateur proximal associé à une fonction

$$g = \Psi \circ A \tag{4.6}$$

où $A \in \mathbb{R}^{M\times N}$ et

$$\Psi\colon \mathbb{R}^M \to \left]-\infty,+\infty\right] : (u_m)_{1\leq m\leq M} \mapsto \sum_{m=1}^{M} \psi_m(u_m) \tag{4.7}$$

avec pour tout $m \in \{1,\ldots,M\}$, $\psi_m \in \Gamma_0(\mathbb{R})$. Comme l'énonce la proposition 4.3 qui suit, l'opérateur proximal de cette fonction peut avoir une forme explicite uniquement dans des cas spécifiques. Remarquons que cette proposition constitue une généralisation de [Combettes, Pesquet, 2007a, proposition 11] pour les fonctions convexes séparables.

Proposition 4.3 *Soient $\mathcal{H}$ et $\mathcal{G}$, deux espaces de Hilbert, réels, séparables et soit $(o_m)_{m\in\mathbb{K}\subset\mathbb{N}}$ une base orthonormale de $\mathcal{G}$. Considérons une fonction Φ, telle que,*

$$(\forall u \in \mathcal{G}) \qquad \Phi(u) = \sum_{m\in\mathbb{K}} \varphi_m(\langle u, o_m\rangle), \tag{4.8}$$

où $(\varphi_m)_{m\in\mathbb{K}}$ sont des fonctions de $\Gamma_0(\mathbb{R})$. Supposons que $\mathbb{K}$ soit fini ou qu'il existe un sous-ensemble $\mathbb{L}$ de $\mathbb{K}$ tel que $\mathbb{K}\smallsetminus\mathbb{L}$ soit fini et $(\forall m \in \mathbb{L})\ \varphi_m \geq \varphi_m(0) = 0$.
Soit $L\colon \mathcal{H} \to \mathcal{G}$ un opérateur linéaire borné tel que la composition de L avec L^* est un isomorphisme qui est diagonalisable par $(o_m)_{m\in\mathbb{K}}$, i.e.

$$(\forall m \in \mathbb{K}) \qquad \underbrace{L\circ L^*}_{D} o_m = d_m o_m \tag{4.9}$$

où $(d_m)_{m\in\mathbb{K}}$ est une suite de réels positifs.
On a alors $\Phi \circ L \in \Gamma_0(\mathcal{H})$ et

$$\operatorname{prox}_{\Phi\circ L} = \operatorname{Id} + L^* \circ D^{-1} \circ (\operatorname{prox}_{D\Phi} - \operatorname{Id}) \circ L \tag{4.10}$$

où Id *est l'opérateur identité et $D\Phi$ est la fonction définie par*

$$(\forall u \in \mathcal{G}) \qquad D\Phi(u) = \sum_{m\in\mathbb{K}} d_m \varphi_m(\langle u, o_m\rangle). \tag{4.11}$$

Preuve :
$D = L \circ L^*$ est bijectif, donc L est surjectif et puisque $\operatorname{dom}\Phi \neq \varnothing$, on en déduit que $\operatorname{dom}(\Phi\circ L) \neq \varnothing$. Cela nous permet de conclure que $h = \Phi\circ L$ est une fonction de $\Gamma_0(\mathcal{H})$.

Nous cherchons ensuite à calculer l'opérateur proximal de h. Pour cela, commençons par écrire sa définition, à savoir, pour tout $w \in \mathcal{H}$,

$$\operatorname{prox}_h w = \arg\min_{v \in \mathcal{H}} \frac{1}{2}\|v - w\|^2 + \Phi(Lv). \tag{4.12}$$

Comme L est surjectif, $\operatorname{Im} L = \mathcal{G}$ est fermé. Il est alors possible d'écrire tout vecteur $v \in \mathcal{H}$ comme une somme d'un élément L^*t dans $\operatorname{Im} L^*$ et $v_\perp \in (\operatorname{Im} L^*)^\perp = \ker L$. De plus,

$$Lv = LL^*t = Dt.$$

De façon similaire, nous pouvons écrire pour tout vecteur $w \in \mathcal{H}$, $w = L^*u + w_\perp$ où $u \in \mathcal{G}$ et $w_\perp \in \ker L$. Dans ces conditions, $\operatorname{prox}_h w$ peut être connu en déterminant

$$\begin{aligned} &\arg\min_{(t,v_\perp)\in\mathcal{G}\times\mathcal{H}} \frac{1}{2}\|L^*t + v_\perp - L^*u - w_\perp\|^2 + \Phi(Dt) \\ &= \arg\min_{(t,v_\perp)\in\mathcal{G}\times\mathcal{H}} \frac{1}{2}\|L^*(t-u)\|^2 + \frac{1}{2}\|v_\perp - w_\perp\|^2 + \Phi(Dt). \end{aligned} \tag{4.13}$$

Si nous posons,

$$v_\perp = w_\perp = w - L^*u, \tag{4.14}$$

il reste alors à trouver

$$\arg\min_{t\in\mathcal{G}} \frac{1}{2}\|L^*(t-u)\|^2 + \Phi(Dt) = \arg\min_{t\in\mathcal{G}} \frac{1}{2}\left\langle (t-u), D(t-u)\right\rangle + \Phi(Dt). \tag{4.15}$$

En utilisant la séparabilité de Φ, l'équation (4.15) s'écrit

$$\arg\min_{t\in\mathcal{G}} \frac{1}{2}\sum_{m\in\mathbb{K}} d_m(\langle t, o_m\rangle - \langle u, o_m\rangle)^2 + \varphi_m(d_m\langle t, o_m\rangle). \tag{4.16}$$

Nous déduisons alors de la propriété du changement d'échelle (rappelée dans l'équation (3.5) du chapitre 3) que

$$(\forall m \in \mathbb{K}) \qquad \langle t, o_m\rangle = \operatorname{prox}_{\frac{1}{d_m}\varphi_m(d_m\cdot)}(\langle u, o_m\rangle) = \frac{1}{d_m}\operatorname{prox}_{d_m\varphi_m}(d_m\langle u, o_m\rangle), \tag{4.17}$$

ce qui, selon [Chaux *et al.*, 2007, proposition 2.10] (rappelée par la proposition 3.13 du chapitre 3), donne

$$t = D^{-1}\sum_{m\in\mathbb{K}} \operatorname{prox}_{d_m\varphi_m}(d_m\langle u, o_m\rangle)\, o_m = D^{-1}\operatorname{prox}_{D\Phi}(Du). \tag{4.18}$$

En combinant les équations (4.14) et (4.18) nous obtenons

$$v = w + L^*\big(D^{-1}\operatorname{prox}_{D\Phi}(Du) - u\big).$$

De plus, puisque L^*u est la projection de w sur $\operatorname{Im} L^*$, $u = (LL^*)^{-1}Lw = D^{-1}Lw$ et l'équation (7.19) en découle. $\square$

La fonction Ψ définie dans (4.7) est séparable dans la base canonique de $\mathbb{R}^M$. Cependant, pour un opérateur convolutif arbitraire $L = A$, l'égalité (4.9) n'est généralement pas vérifiée. Néanmoins, nous allons voir que **la fonction g, définie par l'équation** (4.6)**, peut être décomposée en une somme de fonctions pour lesquelles les opérateurs proximaux peuvent être calculés explicitement**. Pour cela, nous supposons que

$$(\mathbb{I}_i)_{1\leq i\leq I} \text{ est une partition de } \{1,\ldots,M\} \tag{4.19}$$

composée d'ensembles non vides. Pour tout $i \in \{1,\ldots,I\}$, on note M_i le nombre d'éléments dans $\mathbb{I}_i$ $\big(\sum_{i=1}^I M_i = M\big)$ et

$$\Upsilon_i \,:\, \mathbb{R}^{M_i} \to \,]0,+\infty[\,:\, (u_m)_{m\in\mathbb{I}_i} \mapsto \sum_{m\in\mathbb{I}_i} \psi_m(u_m).$$

Si, pour tout $i \in \{1,\ldots,M\}$, $\xi_i \in \mathbb{R}^N$ représente le vecteur associé à la i-ème ligne de A, nous pouvons écrire $g = \sum_{i=1}^I \Upsilon_i \circ A_i$ où A_i est un opérateur linéaire de $\mathbb{R}^N$ dans $\mathbb{R}^{M_i}$ associé à la matrice

$$\begin{bmatrix} \xi_{m_1}^\top \\ \vdots \\ \xi_{m_{M_i}}^\top \end{bmatrix} \tag{4.20}$$

et $\mathbb{I}_i = \{m_1,\ldots,m_{M_i}\}$. On supposera par la suite que la famille $(\xi_m)_{m\in\mathbb{I}_i}$ vérifie l'hypothèse suivante.

Hypothèse 4.4 *Pour tout $i \in \{1,\ldots,I\}$, $(\xi_m)_{m\in\mathbb{I}_i}$ est une famille de vecteurs orthogonaux non nuls.*

Sous l'hypothèse 4.4, la fonction g peut être décomposée comme une somme de I fonctions $(\Upsilon_i \circ A_i)_{1\leq i\leq I}$ où, pour tout $i \in \{1,\ldots,I\}$, $D_i = A_i \circ A_i^*$ est associé à une matrice diagonale inversible $\operatorname{Diag}(\Delta_{i,1},\ldots,\Delta_{i,M_i})$. D'après la proposition 4.3, nous obtenons

$$\operatorname{prox}_{\Upsilon_i\circ A_i} = \operatorname{Id} + A_i^* \circ D_i^{-1} \circ (\operatorname{prox}_{D_i\Upsilon_i} - \operatorname{Id}) \circ A_i. \tag{4.21}$$

Remarque 4.5

(i) *Notons que l'hypothèse 4.4 est évidemment satisfaite quand $I = M$, c'est à dire lorsque, pour tout $i \in \{1,\ldots,I\}$, $\mathbb{I}_i$ se réduit à un singleton.*

(ii) *L'application de A_i ou A_i^* se réduit à des opérations usuelles de traitement du signal. En effet, quand A est un opérateur de convolution, l'application de A_i se réduit à deux étapes : une convolution avec une réponse impulsionnelle du filtre de dégradation et une décimation pour les positions sélectionnées ($m \in \mathbb{I}_i$). L'application de A_i^* se décompose également en deux étapes : une étape d'interpolation (en insérant des zéros entre les valeurs des données d'indices $m \in \mathbb{I}_i$) suivi d'une convolution avec le filtre de réponse fréquentielle conjuguée.*

L'idée fondamentale derrière l'introduction de la partition $(\mathbb{I}_i)_{1\leq i\leq I}$ définie par (4.19), est de former des groupes de lignes de la matrice A ne se recouvrant pas, donc orthogonales entre elles, de façon à calculer les opérateurs proximaux correspondants. Dans le cas d'une convolution, il s'agit de grouper des translatées du noyau de convolution. Afin de réduire le nombre d'opérateurs proximaux à calculer, une solution consiste à trouver le plus petit entier I tel que, pour tout $i \in \{1, \ldots, I\}$, $(\xi_m)_{m\in\mathbb{I}_i}$ reste une famille orthogonale.

Nous allons considérer successivement le cas de problèmes de déconvolution 1D et 2D, où N représente la taille du signal original (resp. $N = N_1 \times N_2$ celle de l'image originale) alors que M désigne la taille du signal dégradé (resp. $M = M_1 \times M_2$ désigne celle de l'image dégradée). Différentes configurations concernant l'impact des effets de bords sur l'opérateur de convolution sont étudiées, à savoir,

- lorsqu'il n'y a pas d'effets de bords,
- lors d'une convolution avec « zero padding »,
- lors d'une convolution périodique.

Le cas de la convolution décimée d'un facteur $d \in \mathbb{N}^*$, avec une gestion des bords par « zero padding », est également présentée.

Dans la suite, Q désigne la taille du noyau du filtre 1D (resp. $Q = Q_1 \times Q_2$ celle d'un filtre 2D) et la suite $(\theta_q)_{0\leq q<Q}$ (resp. $(\theta_{q_1,q_2})_{0\leq q_1<Q_1,0\leq q_2<Q_2}$) correspond aux valeurs du filtre.

• **Modèle convolutif unidimensionnel sans effets de bords**.
Nous avons typiquement la structure Tœplitz suivante :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \theta_{Q-1} & \ldots & \theta_1 & \theta_0 & 0 & \ldots & 0 \\ 0 & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & 0 \\ 0 & \ldots & 0 & \theta_{Q-1} & \ldots & \theta_1 & \theta_0 \end{bmatrix} \tag{4.22}$$

où $M = N - Q + 1 \geq Q$.
En vue de satisfaire l'hypothèse 4.4, nous pouvons choisir $I = Q$ et, pour tout $i \in \{1, \ldots, I\}$,

$$\mathbb{I}_i = \big\{ m \in \{1, \ldots, M\} \;\big|\; m = i \mod I \big\}. \tag{4.23}$$

Par conséquent, nous avons pour tout $i \in \{1, \ldots, I\}$,

$$\Delta_{i,1} = \ldots = \Delta_{i,M_i} = \sum_{q=0}^{Q-1} |\theta_q|^2. \tag{4.24}$$

Dans cette configuration, g peut être décomposée comme une somme de Q fonctions, dont les opérateurs proximaux peuvent être facilement calculés.

• **Modèle convolutif unidimensionnel avec une gestion des bords par « zero-padding »**.
On considère la matrice Tœplitz suivante :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \theta_0 & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ \theta_1 & \theta_0 & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & \vdots \\ \theta_{Q-1} & & \ddots & \ddots & \ddots & & \vdots \\ 0 & \ddots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \theta_{Q-1} & \ldots & \theta_1 & \theta_0 \end{bmatrix} \tag{4.25}$$

où $M = N \geq 2Q - 1$. Dans ce cas, I peut être choisi égal à Q et les ensembles d'indices $(\mathbb{I}_i)_{1 \leq i \leq I}$ sont toujours donnés par (4.23). Enfin, contrairement à l'exemple précédent, les paramètres diagonaux ne sont pas tous égaux. Pour tout $i \in \{1, \ldots, I\}$, nous avons,

$$\Delta_{i,1} = \sum_{q=0}^{i-1} |\theta_q|^2 \qquad \text{et} \qquad \Delta_{i,2} = \ldots = \Delta_{i,M_i} = \sum_{q=0}^{Q-1} |\theta_q|^2. \tag{4.26}$$

• **Modèle de convolution $d-$décimée avec « zero-padding »**.
On considère la matrice de $\mathbb{R}^{M \times N}$ avec $N = Md \geq 2Q$ telle que :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \theta_{d-1} & \ldots & \theta_0 & 0 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & 0 \\ \theta_{2d-1} & \ldots & \theta_{d-1} & \ldots & \theta_0 & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ \vdots & \ddots & & \ddots & & \ddots & \ddots & & & & \vdots \\ & & \ddots & & \ddots & & \ddots & \ddots & & & \vdots \\ 0 & & & \ddots & & \ddots & & \ddots & \ddots & & \vdots \\ \vdots & \ddots & & & \ddots & & \ddots & & \ddots & \ddots & \vdots \\ \vdots & & \ddots & & & \ddots & & \ddots & & \ddots & 0 \\ 0 & \ldots & \ldots & 0 & \theta_{Q-1} & \ldots & \theta_{2d-1} & \ldots & \theta_{d-1} & \ldots & \theta_0 \end{bmatrix}. \tag{4.27}$$

Dans le but de satisfaire l'hypothèse 4.4, nous choisissons $I = \lceil \frac{Q}{d} \rceil$ et les ensembles d'indices $(\mathbb{I}_i)_{1 \leq i \leq I}$ sont encore donnés par (4.23). Enfin, pour tout $i \in \{1, \ldots, I\}$, nous posons,

$$\Delta_{i,1} = \sum_{q=0}^{\min(i \times d, Q)-1} |\theta_q|^2 \qquad \text{et} \qquad \Delta_{i,2} = \ldots = \Delta_{i,M_i} = \sum_{q=0}^{Q-1} |\theta_q|^2. \tag{4.28}$$

Notons que lorsque $d \geq Q$, $(\xi_m)_{m \in \{1,\ldots,M\}}$ est une famille orthogonale, et dans ce cas $I = 1$.

- **Modèle de convolution périodique unidimensionnel**.

Dans cette configuration, une matrice ayant une structure circulante [Golub, Van Loan, 1996] est impliquée :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \theta_0 & 0 & \ldots & 0 & \theta_{Q-1} & \ldots & \theta_1 \\ \theta_1 & \theta_0 & \ddots & & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & & \ddots & \theta_{Q-1} \\ \theta_{Q-1} & & & \ddots & \ddots & & 0 \\ 0 & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & 0 \\ 0 & \ldots & 0 & \theta_{Q-1} & \ldots & \theta_1 & \theta_0 \end{bmatrix} \tag{4.29}$$

avec $M = N \geq Q$. En vue de satisfaire l'hypothèse 4.4, nous choisissons $I = \min\{i \geq Q \mid M = i \mod Q\}$ et, pour tout $i \in \{1, \ldots, I\}$,

$$\mathbb{I}_i = \begin{cases} \{i\} & \text{si } i \leq Q-1 \\ \{m \in \{Q, \ldots, M\} \mid m = i \mod Q\} & \text{sinon.} \end{cases} \tag{4.30}$$

Les paramètres diagonaux sont donnés par (4.24).

Un autre choix a été fait dans [Pustelnik *et al.*, 2009b] en choisissant $I = \min\{i \geq Q \mid M = 0 \mod i\}$ et en procédant comme dans (4.23) et (4.24). Cette solution peut être préférée, quand la valeur résultante de I est petite, pour sa simplicité d'implantation.

Remarque 4.6 *Dans les exemples précédents (les exemples non-décimés étant un cas particulier où $d = 1$), la complexité de calcul pour appliquer chaque opérateur A_i ou A_i^* est $O(Md)$. De plus, nous avons environ Q/d opérateurs proximaux $\mathrm{prox}_{\Upsilon_i \circ A_i}$ à calculer. En supposant une complexité en $O(M_i)$ pour calculer $\mathrm{prox}_{D_i \Upsilon_i}$, la complexité globale est en $O(M(2Q+1))$. Si nous considérons maintenant le cas où $I = M$, la complexité de*

calcul de A_i ou A_i^* est $O(Q)$, et nous avons N opérateurs proximaux $\mathrm{prox}_{\Upsilon_i \circ A_i}$ à calculer. Cela implique un temps de calcul global du même ordre de grandeur que précédemment. Il n'y a donc pas d'avantage clair, en terme de complexité, à limiter le nombre d'opérateurs proximaux à calculer. Cela nous permet néanmoins de réduire la mémoire requise (gain d'un facteur Md/Q pour le stockage des résultats des opérateurs proximaux).

• **Modèles convolutifs bidimensionnel sans effets de bords**.
Nous avons typiquement la structure Tœplitz-block Tœplitz définie comme suit :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \Theta_{Q_1-1} & \dots & \Theta_1 & \Theta_0 & [0] & \dots & [0] \\ [0] & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & [0] \\ [0] & \dots & [0] & \Theta_{Q_1-1} & \dots & \Theta_1 & \Theta_0 \end{bmatrix} \tag{4.31}$$

où
* $M_1 = N_1 - Q_1 + 1 \geq Q_1$,
* $M_2 = N_2 - Q_2 + 1 \geq Q_2$,
* $[0]$ est la matrice nulle de taille $M_2 \times N_2$,

et pour tout $q_1 \in \{0, \dots, Q_1 - 1\}$,

$$\Theta_{q_1} = \begin{bmatrix} \theta_{q_1,Q_2-1} & \dots & \theta_{q_1,1} & \theta_{q_1,0} & 0 & \dots & 0 \\ 0 & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & 0 \\ 0 & \dots & 0 & \theta_{q_1,Q_2-1} & \dots & \theta_{q_1,1} & \theta_{q_1,0} \end{bmatrix} \in \mathbb{R}^{M_2 \times N_2}. \tag{4.32}$$

Dans le but de satisfaire l'hypothèse 4.4, nous pouvons choisir $I = I_1 I_2$ où $I_1 = Q_1$, $I_2 = Q_2$, et pour tout $(i_1, i_2) \in \{1, \dots, Q_1\} \times \{1, \dots, Q_2\}$, $i = i_2 + Q_2(i_1 - 1)$ définir

$$\mathbb{I}_i = \{m_2 + M_2(m_1 - 1) \mid (m_1, m_2) \in \{1, \dots, M_1\} \times \{1, \dots, M_2\} \\ \text{et } m_1 = i_1 \mod I_1, m_2 = i_2 \mod I_2\} \tag{4.33}$$

avec, pour tout $i \in \{1, \dots, I\}$

$$\Delta_{i,1} = \dots = \Delta_{i,M_i} = \sum_{q_1=0}^{Q_1-1} \sum_{q_2=0}^{Q_2-1} |\theta_{q_1,q_2}|^2. \tag{4.34}$$

Dans ce cas, g peut être décomposée comme une somme de $Q_1 Q_2$ fonctions dont chaque opérateur proximal peut facilement être calculé.

• **Modèle convolutifs à deux dimensions avec « zero-padding »**.
Nous considérons la matrice Tœplitz-block Tœplitz définie par :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \Theta_0 & [0] & \dots & \dots & \dots & \dots & [0] \\ \Theta_1 & \Theta_0 & \ddots & & & & \vdots \\ \vdots & & \ddots & \ddots & & & \vdots \\ \Theta_{Q_1-1} & & & \Theta_0 & \ddots & & \vdots \\ [0] & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & [0] \\ [0] & \dots & [0] & \Theta_{Q_1-1} & \dots & \Theta_1 & \Theta_0 \end{bmatrix} \tag{4.35}$$

où

* $M_1 = N_1 \geq Q_1$,
* $M_2 = N_2 \geq Q_2$,
* $[0]$ est la matrice nulle de taille $N_2 \times N_2$,

et pour tout $q_1 \in \{0, \dots, Q_1 - 1\}$,

$$\Theta_{q_1} = \begin{bmatrix} \theta_{q_1,0} & 0 & \dots & \dots & \dots & \dots & 0 \\ \theta_{q_1,1} & \theta_{q_1,0} & 0 & & & & 0 \\ \vdots & & \ddots & \ddots & & & 0 \\ \theta_{q_1,Q_2-1} & & & \theta_{q_1,0} & \ddots & & 0 \\ 0 & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & 0 \\ 0 & \dots & 0 & \theta_{q_1,Q_2-1} & \dots & \theta_{q_1,1} & \theta_{q_1,0} \end{bmatrix} \in \mathbb{R}^{N_2 \times N_2}. \tag{4.36}$$

Dans ce cas, I peut être choisi égal à $Q_1 Q_2$ et l'ensemble d'indices $(\mathbb{I}_i)_{1 \leq i \leq I}$ est toujours donné par (4.33). Mais contrairement à l'exemple précédent, les paramètres diagonaux ne sont pas tous égaux. Nous avons dans ce cas de figure, pour tout $i \in \{1, \dots, I\}$,

$$\Delta_{i, m_2 + M_2(m_1 - 1)} = \sum_{q_1=0}^{\min(m_1, Q_1)-1} \sum_{q_2=0}^{\min(m_2, Q_2)-1} |\theta_{q_1,q_2}|^2. \tag{4.37}$$

• **Modèle de convolution $d-$décimée avec « zero-padding » bidimensionnel**.
Nous obtenons la matrice de $\mathbb{R}^{M_1 M_2 \times N_1 N_2}$ avec $N_1 = M_1 d_1 \geq 2Q_1$ et $N_2 = M_2 d_2 \geq 2Q_2$

telle que :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \Theta_{d_1-1} & \dots & \Theta_0 & [0] & \dots & \dots & \dots & \dots & \dots & \dots & [0] \\ \Theta_{2d_1-1} & \dots & \Theta_{d_1-1} & \dots & \Theta_0 & [0] & \dots & \dots & \dots & \dots & [0] \\ \vdots & \ddots & & \ddots & & \ddots & \ddots & & & & \vdots \\ & & \ddots & & \ddots & & \ddots & \ddots & & & \vdots \\ [0] & & & \ddots & & \ddots & & \ddots & \ddots & & \vdots \\ \vdots & \ddots & & & \ddots & & \ddots & & \ddots & \ddots & \vdots \\ \vdots & & \ddots & & & \ddots & & \ddots & & \ddots & [0] \\ [0] & \dots & \dots & [0] & \Theta_{Q_1-1} & \dots & \Theta_{2d_1-1} & \dots & \Theta_{d_1-1} & \dots & \Theta_0 \end{bmatrix} \tag{4.38}$$

où

* $d_1 M_1 = N_1 \geq 2Q_1$,
* $d_2 M_2 = N_2 \geq 2Q_2$,
* $[0]$ est la matrice nulle de taille $M_2 \times N_2$,

et pour tout $q_1 \in \{0, \dots, Q_1 - 1\}$

$$\Theta_{q_1} = \begin{bmatrix} \theta_{q_1,d_2-1} & \dots & \theta_{q_1,0} & 0 & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ \theta_{q_1,2d_2-1} & \dots & \theta_{q_1,d_2-1} & \dots & \theta_{q_1,0} & 0 & \dots & \dots & \dots & \dots & 0 \\ \vdots & \ddots & & \ddots & & \ddots & \ddots & & & & \vdots \\ & & \ddots & & \ddots & & \ddots & \ddots & & & \vdots \\ 0 & & & \ddots & & \ddots & & \ddots & \ddots & & \vdots \\ \vdots & \ddots & & & \ddots & & \ddots & & \ddots & \ddots & \vdots \\ \vdots & & \ddots & & & \ddots & & \ddots & & \ddots & 0 \\ 0 & \dots & \dots & 0 & \theta_{q_1,Q_2-1} & \dots & \theta_{q_1,2d_2-1} & \dots & \theta_{q_1,d_2-1} & \dots & \theta_{q_1,0} \end{bmatrix}. \tag{4.39}$$

Dans le but de satisfaire l'hypothèse 4.4, nous choisissons $I = I_1 I_2$ avec $I_1 = \lceil \frac{Q_1}{d_1} \rceil$ et $I_2 = \lceil \frac{Q_2}{d_2} \rceil$. Les ensembles d'indices $(\mathbb{I}_i)_{1 \leq i \leq I}$ sont toujours donnés par (4.33) et pour tout $i \in \{1, \dots, I\}$,

$$\Delta_{i,1} = \sum_{q_1=0}^{\min(i \times d_1, Q_1)-1} \sum_{q_2=0}^{\min(i \times d_2, Q_2)-1} |\theta_{q_1,q_2}|^2 \tag{4.40}$$

et

$$\Delta_{i,2} = \dots = \Delta_{i,M_i} = \sum_{q_1=0}^{Q_1-1} \sum_{q_2=0}^{Q_2-1} |\theta_{q_1,q_2}|^2. \tag{4.41}$$

Notons que lorsque $d_1 \geq Q_1$ et $d_2 \geq Q_2$, $(\xi_m)_{m\in\{1,\dots,M\}}$ est une famille orthogonale, et dans ce cas $I = 1$.

• **Modèles convolutifs périodiques à deux dimensions**.
Dans cette configuration, la matrice A possède une structure circulante-bloc circulante [Golub, Van Loan, 1996] telle que :

$$\begin{bmatrix} \xi_1^\top \\ \vdots \\ \xi_M^\top \end{bmatrix} = \begin{bmatrix} \Theta_0 & [0] & \dots & [0] & \Theta_{Q_1-1} & \dots & \Theta_1 \\ \Theta_1 & \Theta_0 & \ddots & & & \ddots & \vdots \\ \vdots & & \ddots & \ddots & & & \Theta_{Q_1-1} \\ \Theta_{Q_1-1} & & & \ddots & \ddots & & [0] \\ [0] & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & [0] \\ [0] & \dots & [0] & \Theta_{Q_1-1} & \dots & \Theta_1 & \Theta_0 \end{bmatrix} \tag{4.42}$$

où

* $M_1 = N_1 \geq Q_1$,
* $M_2 = N_2 \geq Q_2$,
* $[0]$ est la matrice nulle de taille $N_2 \times N_2$,

et pour tout $q_1 \in \{0, \dots, Q_1 - 1\}$

$$\Theta_{q_1} = \begin{bmatrix} \theta_{q_1,0} & 0 & \dots & 0 & \theta_{q_1,Q_2-1} & \dots & \Theta_{q_1,1} \\ \theta_{q_1,1} & \theta_{q_1,0} & 0 & & & \ddots & \vdots \\ \vdots & & \ddots & \ddots & & & \theta_{q_1,Q_2-1} \\ \theta_{q_1,Q_2-1} & & & \ddots & \ddots & & 0 \\ 0 & \ddots & & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \ddots & 0 \\ 0 & \dots & 0 & \theta_{q_1,Q_2-1} & \dots & \theta_{q_1,1} & \theta_{q_1,0} \end{bmatrix} \in \mathbb{R}^{N_2\times N_2}. \tag{4.43}$$

Dans l'optique de satisfaire l'hypothèse 4.4, nous définissons

$$I_1 = \min\{i_1 \geq Q_1 \mid M_1 = i_1 \mod Q_1\} \tag{4.44}$$

et

$$I_2 = \min\{i_2 \geq Q_2 \mid M_2 = i_2 \mod Q_2\}. \tag{4.45}$$

De plus, nous posons $I = I_1 I_2$ et pour tout $(i_1, i_2) \in \{1, \ldots, I_1\} \times \{1, \ldots, I_2\}$, nous prenons $i = i_2 + I_2(i_1 - 1)$ et définissons

$$\mathbb{I}_i = \begin{cases} \{i_2 + M_2(i_1 - 1)\} & \text{si } i_1 < Q_1 \text{ et } i_2 < Q_2 \\ \{m_2 + M_2(i_1 - 1) \mid m_2 \in \{1, \ldots, M_2\} \quad \text{et} \quad m_2 = i_2 \mod Q_2\} & \text{si } i_1 < Q_1 \text{ et } i_2 \geq Q_2 \\ \{i_2 + M_2(m_1 - 1) \mid m_1 \in \{1, \ldots, M_1\} \quad \text{et} \quad m_1 = i_1 \mod Q_1\} & \text{si } i_1 \geq Q_1 \text{ et } i_2 < Q_2 \\ \{m_2 + M_2(m_1 - 1) \mid (m_1, m_2) \in \{1, \ldots, M_1\} \times \{1, \ldots, M_2\}, & \\ \qquad m_1 = i_1 \mod Q_1 \quad \text{et} \quad m_2 = i_2 \mod Q_2\} & \text{sinon .} \end{cases} \tag{4.46}$$

Les paramètres de la diagonale sont donnés par (4.34).
De la même manière que nous l'avions mentionné en 1D, un autre choix possible pour I_1 et I_2 est envisageable [Pustelnik *et al.*, 2009b]. Il s'agit de prendre $I_1 = \min\{i_1 \geq Q_1 \mid M_1 = 0 \mod i_1\}$ et $I_2 = \min\{i_2 \geq Q_2 \mid M_2 = 0 \mod i_2\}$, et de procéder comme pour (4.33) et (4.34).

Remarque 4.7 *La remarque sur la complexité, formulée dans le cas 1D, reste valable dans le cas bidimensionnel.*

4.3.3 Formes discrètes de la variation totale et opérateurs proximaux associés

La variation totale, initialement introduite par Rudin, Osher et Fatemi dans [Rudin *et al.*, 1992] représente une mesure de régularité puissante en restauration d'images pour retrouver des zones de l'image constantes par morceaux avec des frontières prononcées [Rudin, Osher, 1994; Malgouyres, 2002; Aujol *et al.*, 2006; Weiss *et al.*, 2009]. Différentes versions de la variation totale discrète peuvent être trouvées dans la littérature [Rudin *et al.*, 1992; Chambolle, 2004; Combettes, Pesquet, 2008]. Notre objectif est de considérer des versions discrètes de la variation totale pour lesquelles il est possible de décomposer le terme de variation totale en une somme de fonctions dont l'opérateur proximal associé possède une forme explicite.

Dans cette partie, nous considérons une forme de variation totale, définie pour une image numérique

$$y = (y_{n_1,n_2})_{0 \leq n_1 < N_1, 0 \leq n_2 < N_2} \in \mathbb{R}^{N_1 \times N_2},$$

qui s'exprime par

$$\mathrm{tv}(y) = \sum_{n_1=0}^{N_1-P_1} \sum_{n_2=0}^{N_2-P_2} \rho_{\mathrm{tv}}\big((y_h)_{n_1,n_2}, (y_v)_{n_1,n_2}\big). \tag{4.47}$$

La fonction ρ_{tv} appartient à la classe $\Gamma_0(\mathbb{R}^2)$, et les éléments y_h et y_v sont deux gradients discrets calculés dans des directions orthogonales grâce à des filtres de réponse impulsionnelle finie de taille $P_1 \times P_2$. Plus précisément, dans l'expression ci-dessus, nous avons

$$(y_h)_{n_1,n_2} = \mathrm{tr}(H^\top Y_{n_1,n_2}) \tag{4.48}$$

$$(y_v)_{n_1,n_2} = \mathrm{tr}(V^\top Y_{n_1,n_2}) \tag{4.49}$$

où $H \in \mathbb{R}^{P_1 \times P_2}$ et $V \in \mathbb{R}^{P_1 \times P_2}$ sont les matrices du noyau du filtre, et pour tout $(n_1, n_2) \in \{0, \ldots, N_1 - P_1\} \times \{0, \ldots, N_2 - P_2\}$,

$$Y_{n_1,n_2} = (y_{n_1+p_1,n_2+p_2})_{0 \leq p_1 < P_1, 0 \leq p_2 < P_2}$$

représente un bloc de $P_1 \times P_2$ pixels voisins. Comme l'opérateur proximal associé à la variation totale, définie par l'équation (4.47), ne possède pas une expression simple en général, nous proposons de décomposer l'équation (4.47) comme suit,

$$(\forall y \in \mathbb{R}^{N_1 \times N_2}) \qquad \mathrm{tv}(y) = \sum_{p_1=0}^{P_1-1} \sum_{p_2=0}^{P_2-1} \mathrm{tv}_{p_1,p_2}(y) \tag{4.50}$$

où, pour tout $p_1 \in \{0, \ldots, P_1 - 1\}$ et $p_2 \in \{0, \ldots, P_2 - 1\}$,

$$\mathrm{tv}_{p_1,p_2}(y) = \sum_{n_1=0}^{\lfloor \frac{N_1-p_1}{P_1} \rfloor - 1} \sum_{n_2=0}^{\lfloor \frac{N_2-p_2}{P_2} \rfloor - 1} \rho_{\mathrm{tv}}\big((y_h)^{p_1,p_2}_{n_1,n_2}, (y_v)^{p_1,p_2}_{n_1,n_2}\big) \tag{4.51}$$

La notation $(\cdot)^{p_1,p_2}_{n_1,n_2} = (\cdot)_{P_1 n_1 + p_1, P_2 n_2 + p_2}$ a été utilisée.

La figure 4.1 (ci-dessous) représente pour deux configurations de (p_1, p_2), l'ensemble des blocs $Y^{p_1,p_2}_{n_1,n_2}$ mis en jeu dans l'équation (4.51). Une forme explicite de l'opérateur proximal associé à la fonction (4.51) peut être déduite de la proposition suivante.

Proposition 4.8 *Supposons que* $\mathrm{tr}(HV^\top) = 0$ *et* $\|H\|_{\mathrm{F}}^2 = \|V\|_{\mathrm{F}}^2 = 1$ *où* $\|\cdot\|_{\mathrm{F}}$ *représente la norme de Frobenius. Pour tout* $y = (y_{n_1,n_2})_{0 \leq n_1 < N_1, 0 \leq n_2 < N_2} \in \mathbb{R}^{N_1 \times N_2}$ *et* $\mu > 0$*, nous avons, pour tout* $(p_1, p_2) \in \{0, \ldots, P_1 - 1\} \times \{0, \ldots, P_2 - 1\}$,

$$\mathrm{prox}_{\mu \mathrm{tv}_{p_1,p_2}} y = (\pi_{n_1,n_2})_{0 \leq n_1 < N_1, 0 \leq n_2 < N_2} \tag{4.52}$$

où, pour tout $(n_1, n_2) \in \{0, \ldots, \lfloor \frac{N_1-p_1}{P_1} \rfloor - 1\} \times \{0, \ldots, \lfloor \frac{N_2-p_2}{P_2} \rfloor - 1\}$,

$$\begin{aligned}(\pi_{P_1 n_1 + p_1 + p_1', P_2 n_2 + p_2 + p_2'})_{0 \leq p_1' < P_1, 0 \leq p_2' < P_2} \\ = (\beta^{p_1,p_2}_{n_1,n_2} - h^{p_1,p_2}_{n_1,n_2}) H + (\kappa^{p_1,p_2}_{n_1,n_2} - \upsilon^{p_1,p_2}_{n_1,n_2}) V + Y^{p_1,p_2}_{n_1,n_2}\end{aligned} \tag{4.53}$$

avec

$$h_{n_1,n_2}^{p_1,p_2} = \mathrm{tr}(H^\top Y_{n_1,n_2}^{p_1,p_2}), \tag{4.54}$$

$$v_{n_1,n_2}^{p_1,p_2} = \mathrm{tr}(V^\top Y_{n_1,n_2}^{p_1,p_2}), \tag{4.55}$$

$$(\beta_{n_1,n_2}^{p_1,p_2}, \kappa_{n_1,n_2}^{p_1,p_2}) = \mathrm{prox}_{\mu\, \rho_{\mathrm{tv}}(\cdot,\cdot)}(h_{n_1,n_2}^{p_1,p_2}, v_{n_1,n_2}^{p_1,p_2}). \tag{4.56}$$

et, pour tout $(n_1, n_2) \in \{0, \ldots, N_1 - 1\} \times \{0, \ldots, N_2 - 1\}$,

$$\pi_{n_1,n_2} = y_{n_1,n_2} \qquad \textit{si } n_1 < p_1 \textit{ ou } n_2 < p_2 \textit{ ou } n_1 \geq P_1 \lfloor \tfrac{N_1 - p_1}{P_1} \rfloor \textit{ ou } n_2 \geq P_2 \lfloor \tfrac{N_2 - p_2}{P_2} \rfloor. \tag{4.57}$$

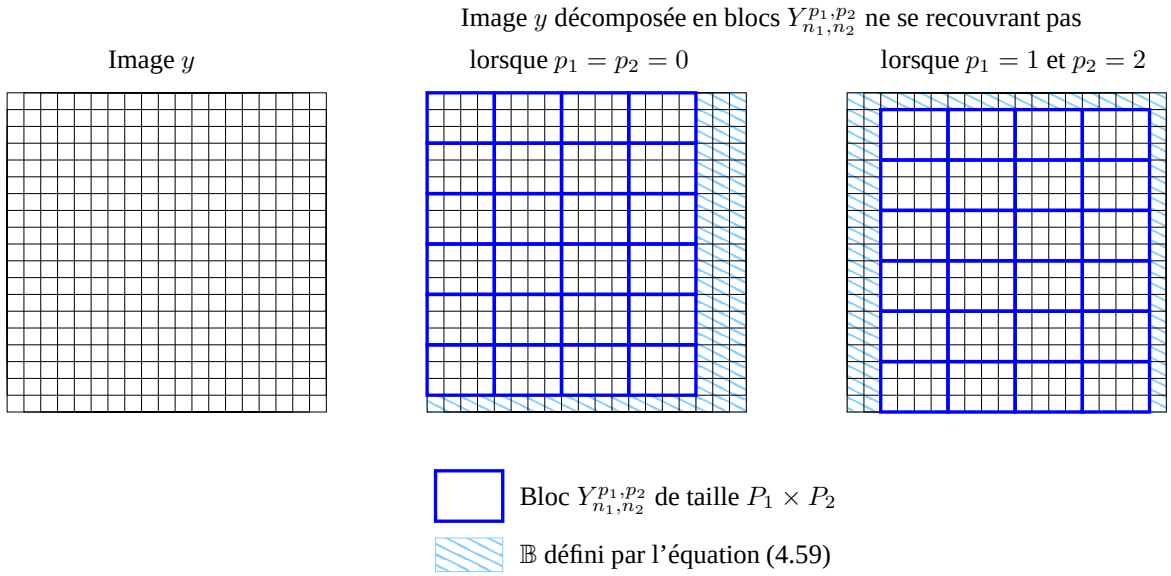


FIGURE 4.1 – Décomposition de l'image qui permet de calculer l'opérateur proximal de la variation totale.

Preuve :
En utilisant la définition d'un opérateur proximal, pour tout $y \in \mathbb{R}^{N_1 \times N_2}$,

$$\pi = \mathrm{prox}_{\mu \mathrm{tv}_{p_1,p_2}} y$$

minimise

$$\frac{1}{2}\|\pi - y\|^2 + \mu \mathrm{tv}_{p_1,p_2}(\pi) \tag{4.58}$$

Si l'on pose, pour tout $(n_1, n_2) \in \{0, \ldots, \lfloor \frac{N_1 - p_1}{P_1} \rfloor - 1\} \times \{0, \ldots, \lfloor \frac{N_2 - p_2}{P_2} \rfloor - 1\}$,

$$\Pi_{n_1,n_2}^{p_1,p_2} = (\pi_{P_1 n_1 + p_1 + p_1', P_2 n_2 + p_2 + p_2'})_{0 \leq p_1' < P_1, 0 \leq p_2' < P_2}$$

et

$$\mathbb{B} = \{(n_1, n_2) \in \mathbb{N}^2 \mid \ 0 \leq n_1 < p_1 \text{ ou } 0 \leq n_2 < p_2 \\ \text{ou } P_1 \lfloor \frac{N_1 - p_1}{P_1} \rfloor \leq n_1 < N_1 \text{ ou } P_2 \lfloor \frac{N_2 - p_2}{P_2} \rfloor \leq n_2 < N_2\}, \tag{4.59}$$

l'équation (4.58) s'écrit également

$$\sum_{(n_1,n_2)\in\mathbb{B}} \frac{1}{2}(\pi_{n_1,n_2} - y_{n_1,n_2})^2 + \sum_{n_1=0}^{\lfloor \frac{N_1-p_1}{P_1} \rfloor - 1} \sum_{n_2=0}^{\lfloor \frac{N_2-p_2}{P_2} \rfloor - 1} \frac{1}{2} \|\Pi_{n_1,n_2}^{p_1,p_2} - Y_{n_1,n_2}^{p_1,p_2}\|_{\mathrm{F}}^2 \\ + \mu\ \rho_{\mathrm{tv}}\big(\mathrm{tr}(H^\top \Pi_{n_1,n_2}^{p_1,p_2}), \mathrm{tr}(V^\top \Pi_{n_1,n_2}^{p_1,p_2})\big). \tag{4.60}$$

Il est clair que l'équation (4.57) est vérifiée lorsque les variables π_{n_1,n_2} pour $(n_1, n_2) \in \mathbb{B}$ ne sont pas des éléments des matrices $\big(\Pi_{n_1,n_2}^{p_1,p_2}\big)_{(n_1,n_2)\in\{0,\ldots,\lfloor \frac{N_1-p_1}{P_1} \rfloor - 1\}\times\{0,\ldots,\lfloor \frac{N_2-p_2}{P_2} \rfloor - 1\}}$. De plus, puisque $\mathrm{tr}(H^\top V) = 0$ et $\|H\|_{\mathrm{F}}^2 = \|V\|_{\mathrm{F}}^2 = 1$, les matrices $\Pi_{n_1,n_2}^{p_1,p_2}$ et $Y_{n_1,n_2}^{p_1,p_2}$ peuvent être décomposées en composantes orthogonales comme suit :

$$\begin{cases} \Pi_{n_1,n_2}^{p_1,p_2} = \beta_{n_1,n_2}^{p_1,p_2} H + \kappa_{n_1,n_2}^{p_1,p_2} V + (\Pi_{n_1,n_2}^{p_1,p_2})^\perp \\ Y_{n_1,n_2}^{p_1,p_2} = h_{n_1,n_2}^{p_1,p_2} H + \upsilon_{n_1,n_2}^{p_1,p_2} V + (Y_{n_1,n_2}^{p_1,p_2})^\perp \end{cases}$$

où

$$\begin{aligned} \beta_{n_1,n_2}^{p_1,p_2} &= \mathrm{tr}(H^\top \Pi_{n_1,n_2}^{p_1,p_2}), \\ \kappa_{n_1,n_2}^{p_1,p_2} &= \mathrm{tr}(V^\top \Pi_{n_1,n_2}^{p_1,p_2}), \\ (\Pi_{n_1,n_2}^{p_1,p_2})^\perp &= \Pi_{n_1,n_2}^{p_1,p_2} - \beta_{n_1,n_2}^{p_1,p_2} H - \kappa_{n_1,n_2}^{p_1,p_2} V, \\ (Y_{n_1,n_2}^{p_1,p_2})^\perp &= Y_{n_1,n_2}^{p_1,p_2} - h_{n_1,n_2}^{p_1,p_2} H - \upsilon_{n_1,n_2}^{p_1,p_2} V, \end{aligned}$$

et $(h_{n_1,n_2}^{p_1,p_2}, \upsilon_{n_1,n_2}^{p_1,p_2})$ est donné par (4.55). Après quelques simplifications, l'équation (4.60) s'écrit, pour tout $n_1 \in \{0, \ldots, \lfloor \frac{N_1-p_1}{P_1} \rfloor - 1\}$ et $n_2 \in \{0, \ldots, \lfloor \frac{N_2-p_2}{P_2} \rfloor - 1\}$,

$$\frac{1}{2} \|\Pi_{n_1,n_2}^{p_1,p_2} - Y_{n_1,n_2}^{p_1,p_2}\|_{\mathrm{F}}^2 + \mu\ \rho_{\mathrm{tv}}\big(\mathrm{tr}(H^\top \Pi_{n_1,n_2}^{p_1,p_2}), \mathrm{tr}(V^\top \Pi_{n_1,n_2}^{p_1,p_2})\big),$$

et, de façon identique,

$$\frac{1}{2} \|(\Pi_{n_1,n_2}^{p_1,p_2})^\perp - (Y_{n_1,n_2}^{p_1,p_2})^\perp\|_{\mathrm{F}}^2 + \frac{1}{2}(\beta_{n_1,n_2}^{p_1,p_2} - h_{n_1,n_2}^{p_1,p_2})^2 \\ + \frac{1}{2}(\kappa_{n_1,n_2}^{p_1,p_2} - \upsilon_{n_1,n_2}^{p_1,p_2})^2 + \mu\ \rho_{\mathrm{tv}}\big(\beta_{n_1,n_2}^{p_1,p_2}, \kappa_{n_1,n_2}^{p_1,p_2}\big).$$

Cela montre que (4.56) est satisfaite et que $(\Pi_{n_1,n_2}^{p_1,p_2})^\perp = (Y_{n_1,n_2}^{p_1,p_2})^\perp$. On en déduit directement l'équation (4.53). $\square$

Le résultat de la proposition 4.8 signifie que, pour une valeur donnée de $(p_1, p_2) \in \{0, \ldots, P_1 - 1\} \times \{0, \ldots, P_2 - 1\}$, l'image est décomposée en blocs ne se recouvrant pas (c.à.d n'ayant pas de pixel commun). Ces blocs sont notés

$$Y_{n_1,n_2}^{p_1,p_2} = (y_{P_1 n_1 + p_1 + p_1', P_2 n_2 + p_2 + p_2'})_{0 \le p_1' < P_1, 0 \le p_2' < P_2}$$

et sont composés de $P_1 \times P_2$ pixels.

La figure 4.1 illustre la construction des blocs et les effets de bords introduits pour différentes valeurs de p_1 et p_2. L'équation (4.53) fournit alors l'expression de l'opérateur proximal associé à chacun de ces blocs, tandis que l'équation (4.57) traite les effets de bord.

Remarque 4.9 *Le résultat ci-dessus offre un certain degré de liberté dans la définition de la variation totale discrète, tant pour le choix des fonctions ρ_{tv} que pour le choix des filtres de gradients H et V. En voici quelques exemples.*

- *Deux choix classiques pour la fonction ρ_{tv} [Rudin et al., 1992] sont les suivants :*
  - (i) *Si $\rho_{\mathrm{tv}} \colon (\xi_1, \xi_2) \mapsto |\xi_1| + |\xi_2|$ alors, une forme anisotrope est obtenue. D'après l'exemple 3.3 du chapitre 2,* (4.56) *se réduit à*

$$\begin{cases} \beta_{n_1,n_2}^{p_1,p_2} = \operatorname{sign}(h_{n_1,n_2}^{p_1,p_2}) \max(|h_{n_1,n_2}^{p_1,p_2}| - \mu, 0) \\ \kappa_{n_1,n_2}^{p_1,p_2} = \operatorname{sign}(v_{n_1,n_2}^{p_1,p_2}) \max(|v_{n_1,n_2}^{p_1,p_2}| - \mu, 0). \end{cases} \tag{4.61}$$

  - (ii) *Si $\rho_{\mathrm{tv}} \colon (\xi_1, \xi_2) \mapsto \sqrt{(\xi_1)^2 + (\xi_2)^2}$, alors on retrouve la forme isotrope usuelle. L'opérateur proximal à déterminer dans* (4.56) *se déduit de l'exemple 4.1.*

- *Certains exemples de matrices de gradient H et V satisfaisant les hypothèses de la proposition 4.8 sont détaillées ci-dessous :*
  - (i) *Les filtres de Roberts tels que $H = \begin{bmatrix} -1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} \end{bmatrix}$ et $V = \begin{bmatrix} 0 & -1/\sqrt{2} \\ 1/\sqrt{2} & 0 \end{bmatrix}$ ont été étudiés dans [Combettes, Pesquet, 2008].*
  - (ii) *Les filtres de différences finies peuvent être utilisés, ils sont tels que $H = V^\top = \begin{bmatrix} 0 & 0 & 0 \\ -1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 0 & 0 & 0 \end{bmatrix}$.*

(iii) *Les filtres de Prewitt satisfont aussi les hypothèses requises. Ils sont définis par* $H = V^\top = \begin{bmatrix} -1/\sqrt{6} & 0 & 1/\sqrt{6} \\ -1/\sqrt{6} & 0 & 1/\sqrt{6} \\ -1/\sqrt{6} & 0 & 1/\sqrt{6} \end{bmatrix}$.

(iv) *Les filtres de Sobel définis par* $H = V^\top = \begin{bmatrix} -1/\sqrt{12} & 0 & 1/\sqrt{12} \\ -2/\sqrt{12} & 0 & 2/\sqrt{12} \\ -1/\sqrt{12} & 0 & 1/\sqrt{12} \end{bmatrix}$ *constituent également un choix possible.*

4.4 Algorithme proximal parallèle (PPXA)

Dans la classe des méthodes d'optimisation convexe, l'algorithme récemment proposé dans [Combettes, Pesquet, 2008], apparaît bien adapté pour résoudre la classe des problèmes fomulés par (4.1). Quand une représentation sur une trame est considérée, le problème peut être formulé par (4.2). Dans cette configuration, lorsque le nombre S de fonctions de $\Gamma_0(\mathbb{R}^N)$ est grand, les opérateurs d'analyse et de synthèse de trame doivent être appliqués de nombreuses fois dans l'algorithme, lors du calcul de l'opérateur proximal de chaque fonction g_i. Cela peut entraîner des temps de calcul longs. Dans cette section, nous rappelons brièvement la formulation générale de l'algorithme proximal parallèle (PPXA), ses propriétés de convergence et ses itérations sont détaillées pour la résolution du problème (4.2). Finalement, nous proposons une version accélérée de PPXA permettant de résoudre efficacement le problème (4.2).

4.4.1 Présentation de l'algorithme

Une formulation équivalente du problème d'optimisation convexe (4.1) est :

$$\text{Trouver} \quad \widehat{x} \in \underset{\substack{x_1 \in \mathbb{R}^K, \ldots, x_J \in \mathbb{R}^K \\ x = x_1 = \ldots = x_J}}{\operatorname{Argmin}} \quad \sum_{j=1}^{J} f_j(x_j). \tag{4.62}$$

Cette formulation a été utilisée dans [Combettes, Pesquet, 2008] pour obtenir l'algorithme proximal parallèle (cf. algorithme 6).

PPXA requiert le choix de constantes réelles comme le pas γ, les pondérations $(\omega_j)_{1 \leq j \leq J}$ et, à chaque itération $\ell \in \mathbb{N}$, le paramètre de relaxation λ_ℓ. Les termes $(a_{j,\ell})_{1 \leq j \leq J}$, correspondant aux erreurs de calcul possibles sur les opérateurs proximaux (par exemple si l'opérateur proximal ne possède pas de forme explicite), soulignent la stabilité numérique de l'algorithme. La suite $(x_\ell)_{\ell \geq 1}$ générée par l'algorithme 6, converge vers une solution du problème (4.62), ou de façon équivalent vers une solution du problème (4.1). La convergence est démontrée dans [Combettes, Pesquet, 2008] sous les hypothèses suivantes.

Initialisation
- Choisir $\gamma \in \,]0,+\infty[$.
- Pour tout $j \in \{1,\ldots,J\}$, choisir $(\omega_j)_{1\leq j\leq J} \in]0,1]^J$ tel que $\sum_{j=1}^J \omega_j = 1$.
- Choisir $(u_{j,0})_{1\leq j\leq J} \in (\mathbb{R}^K)^J$ et $x_0 = \sum_{j=1}^J \omega_j u_{j,0}$.

Pour $\ell = 0,1,\ldots$
- Pour $j = 1,\ldots,J$
  - $p_{j,\ell} = \mathrm{prox}_{\gamma f_j/\omega_j} u_{j,\ell} + a_{j,\ell}$
- $p_\ell = \sum_{j=1}^J \omega_j p_{j,\ell}$
- Choisir $\lambda_\ell \in \,]0,2[$.
- Pour $j = 1,\ldots,J$
  - $u_{j,\ell+1} = u_{j,\ell} + \lambda_\ell\,(2\,p_\ell - x_\ell - p_{j,\ell})$
- $x_{\ell+1} = x_\ell + \lambda_\ell\,(p_\ell - x_\ell)$

Algorithme 6: Forme générale du PPXA.

Hypothèse 4.10

(i) $\lim_{\|x\|\to+\infty} f_1(x) + \ldots + f_J(x) = +\infty$.

(ii) $\cap_{j=1}^J \mathrm{ri}\,\mathrm{dom}\, f_j \neq \varnothing$.

(iii) $(\forall j \in \{1,\ldots,J\})\ \sum_{\ell\in\mathbb{N}} \lambda_\ell\,\|a_{j,\ell}\| < +\infty$.

(iv) $\sum_{\ell\in\mathbb{N}} \lambda_\ell\,(2-\lambda_\ell) = +\infty$.

L'hypothèse 4.10(i) traduit la coercivité de la somme de fonctions à minimiser. Les hypothèses 4.10(i) et 4.10(ii) assurent l'existence d'une solution.

Considérons maintenant le problème (4.2) où une trame ajustée est utilisée. En choisissant, pour tout $j \in \{1,\ldots,S\}$, $f_j = g_j \circ F^\top$ et en invoquant la proposition 4.3 avec $L = F^\top$ et $D = \nu\mathrm{Id}$, les itérations de l'algorithme 6 résultantes sont détaillées dans l'algorithme 7.
Il apparaît que la première boucle peut être coûteuse en terme de complexité de calcul. En effet, il est nécessaire d'appliquer S fois les opérateurs F et $F^\top$ à chaque itération. Nous proposons maintenant une solution pour accélérer ces itérations.

4.4.2 Version accélérée en présence de trames ajustées

La construction de la version accélérée de l'algorithme 7 se base sur les éléments suivants. Si Π_F représente le projecteur sur $\mathrm{Im}\,F$, alors tout élément p de $\mathbb{R}^K$ peut être

Pour $\ell = 0, 1, \ldots$
- Pour $j = 1, \ldots, S$
  - $p_{j,\ell} = u_{j,\ell} + \frac{F}{\nu}\big(\mathrm{prox}_{\nu\gamma g_j/\omega_j}(F^\top u_{j,\ell}) - F^\top u_{j,\ell}\big) + a_{j,\ell}$
- Pour $j = S+1, \ldots, J$
  - $p_{j,\ell} = \mathrm{prox}_{\gamma f_j/\omega_j} u_{j,\ell} + a_{j,\ell}$
- $p_\ell = \sum_{j=1}^{J} \omega_j p_{j,\ell}$
- Choisir $\lambda_\ell \in]0, 2[$
- Pour $j = 1, \ldots, J$
  - $u_{j,\ell+1} = u_{j,\ell} + \lambda_\ell \left(2\, p_\ell - x_\ell - p_{j,\ell}\right)$
- $x_{\ell+1} = x_\ell + \lambda_\ell(p_\ell - x_\ell)$

Algorithme 7: Itérations détaillées du PPXA pour résoudre le problème (4.2).

décomposé comme une somme d'un élément de $\mathrm{Im}\, F$ et $(\mathrm{Im}\, F^\top)^\perp = \ker F$, *i.e.*,

$$p = \Pi_F\, p + p^\perp \tag{4.63}$$

où $p^\perp$ est l'erreur de projection. En combinant le fait qu'il existe un élément $q \in \mathbb{R}^N$ tel que

$$\Pi_F\, p = Fq, \tag{4.64}$$

avec le fait que $F^\top p^\perp = 0$, nous obtenons la relation,

$$q = \frac{1}{\nu} F^\top p. \tag{4.65}$$

De plus, on déduit de (4.63) et (4.65) que

$$p^\perp = p - \frac{1}{\nu} F F^\top p. \tag{4.66}$$

Penchons-nous maintenant sur la première étape de l'algorithme 7 :

$$(\forall j \in \{1, \ldots, S\}) \qquad p_{j,\ell} = u_{j,\ell} + \frac{F}{\nu}(\mathrm{prox}_{\nu\gamma g_j/\omega_j}(F^\top u_{j,\ell}) - F^\top u_{j,\ell}) + a_{j,\ell} \tag{4.67}$$

où $a_{j,\ell}$ est supposé appartenir à $\mathrm{Im}\, F$; en d'autres termes, $a_{j,\ell} = F\widetilde{a}_{j,\ell}$ avec $\widetilde{a}_{j,\ell} \in \mathbb{R}^K$. En définissant $q_{j,\ell} \in \mathbb{R}^N$ similairement à (4.64), cela implique $\Pi_F\, p_{j,\ell} = Fq_{j,\ell}$. Selon (4.65), $q_{j,\ell}$ est tel que

$$q_{j,\ell} = \frac{1}{\nu} F^\top p_{j,\ell}. \tag{4.68}$$

En combinant (4.67) et (4.68),

$$q_{j,\ell} = \frac{1}{\nu}\mathrm{prox}_{\nu\gamma g_j/\omega_j}(v_{j,\ell}) + \widetilde{a}_{j,\ell} \qquad \text{où} \qquad v_{j,\ell} = F^\top u_{j,\ell}. \tag{4.69}$$

De plus, puisque $p_{j,\ell} = Fq_{j,\ell} + p_{j,\ell}^\perp$, le calcul de la variable $p_\ell = \sum_{j=1}^J \omega_j p_{j,\ell}$ dans l'algorithme 7 peut être réécrit

$$p_\ell = F\sum_{j=1}^S \omega_j q_{j,\ell} + \sum_{j=1}^S \omega_j p_{j,\ell}^\perp + \sum_{j=S+1}^J \omega_j p_{j,\ell} \tag{4.70}$$

où, selon (4.66) et (4.67),

$$\begin{aligned} p_{j,\ell}^\perp &= p_{j,\ell} - \frac{1}{\nu}FF^\top p_{j,\ell} \\ &= u_{j,\ell} - \frac{1}{\nu}FF^\top u_{j,\ell} \\ &= u_{j,\ell}^\perp. \end{aligned} \tag{4.71}$$

Dans la nouvelle formulation, les dernières étapes de l'algorithme consistent à mettre à jour $u_{j,\ell}^\perp$ et $v_{j,\ell}$, pour tout $j \in \{1,\ldots,S\}$. Nous proposons de définir

$$r_\ell = 2p_\ell - x_\ell \quad \text{et} \quad \widetilde{r}_\ell = F^\top r_\ell \quad \text{et} \quad r_\ell^\perp = r_\ell - \frac{1}{\nu}F\tilde{r}_\ell$$

ce qui implique

$$v_{j,\ell+1} = v_{j,\ell} + \lambda_\ell\big(\widetilde{r}_\ell - F^\top p_{j,\ell}\big) \qquad \text{et} \qquad u_{j,\ell+1}^\perp = u_{j,\ell}^\perp + \lambda_\ell\big(r_\ell^\perp - p_{j,\ell}^\perp\big).$$

Puis, en utilisant (4.68) et (4.71), ces relations peuvent être simplifiées sous la forme

$$v_{j,\ell+1} = v_{j,\ell} + \lambda_\ell\big(\widetilde{r}_\ell - \nu q_{j,\ell}\big) \quad \text{et} \quad u_{j,\ell+1}^\perp = u_{j,\ell}^\perp + \lambda_\ell\big(r_\ell^\perp - u_{j,\ell}^\perp\big), \tag{4.72}$$

ce qui conduit à l'algorithme 8.

Pour garantir la convergence de l'algorithme proposé sous des conditions similaires à celles de l'algorithme 6, nous faisons l'hypothèse suivante :

Hypothèse 4.11

(i) $\lim_{\|x\|\to+\infty} g_1(F^\top x) + \ldots + g_S(F^\top x) + f_{S+1}(x) + \ldots + f_J(x) = +\infty$.

(ii) $\big(\cap_{j=1}^S \operatorname{ri}\operatorname{dom}(g_j \circ F^\top)\big) \bigcap \big(\cap_{j=S+1}^J \operatorname{ri}\operatorname{dom} f_j\big) \neq \varnothing$.

(iii) $(\forall j \in \{1,\ldots,S\})\ \sum_{\ell\in\mathbb{N}} \lambda_\ell \|\widetilde{a}_{j,\ell}\| < +\infty$
et $(\forall j \in \{S+1,\ldots,J\})\ \sum_{\ell\in\mathbb{N}} \lambda_\ell \|a_{j,\ell}\| < +\infty$.

(iv) $\sum_{\ell\in\mathbb{N}} \lambda_\ell\,(2-\lambda_\ell) = +\infty$.

Initialisation
- Soit $\gamma \in \,]0,+\infty[$.
- Pour tout $j \in \{1,\ldots,J\}$, on choisit $(\omega_j)_{1\leq j\leq J} \in]0,1]^J$ tel que $\sum_{j=1}^J \omega_j = 1$.
- Soit $(u_{j,0})_{1\leq j\leq J} \in (\mathbb{R}^K)^J$ et soit $x_0 = \sum_{j=1}^J \omega_j u_{j,0}$.
- Pour tout $j \in \{1,\ldots,S\}$, soient $v_{j,0} = F^\top u_{j,0}$ et $u_{j,0}^\perp = u_{j,0} - \frac{1}{\nu} F v_{j,0}$.

Pour $\ell = 0,1,\ldots$
- Pour $j = 1,\ldots,S$
  - $q_{j,\ell} = \frac{1}{\nu}\mathrm{prox}_{\nu\gamma g_j/\omega_j} v_{j,\ell} + \widetilde{a}_{j,\ell}$
- Pour $j = S+1,\ldots,J$
  - $p_{j,\ell} = \mathrm{prox}_{\gamma f_j/\omega_j} u_{j,\ell} + a_{j,\ell}$
- $p_\ell = \sum_{j=1}^S \omega_j u_{j,\ell}^\perp + F\sum_{j=1}^S \omega_j q_{j,\ell} + \sum_{j=S+1}^J \omega_j p_{j,\ell}$
- $r_\ell = 2\,p_\ell - x_\ell$
- $\widetilde{r}_\ell = F^\top r_\ell$
- $r_\ell^\perp = r_\ell - \frac{1}{\nu} F \widetilde{r}_\ell$
- Soit $\lambda_\ell \in \,]0,2[$
- Pour $j = 1,\ldots,S$
  - $u_{j,\ell+1}^\perp = u_{j,\ell}^\perp + \lambda_\ell\,(r_\ell^\perp - u_{j,\ell}^\perp)$
  - $v_{j,\ell+1} = v_{j,\ell} + \lambda_\ell\,(\widetilde{r}_\ell - \nu q_{j,\ell})$
- Pour $j = S+1,\ldots,J$
  - $u_{j,\ell+1} = u_{j,\ell} + \lambda_\ell\,(r_\ell - p_{j,\ell})$
- $x_{\ell+1} = x_\ell + \lambda_\ell(p_\ell - x_\ell)$

Algorithme 8: PPXA accéléré.

Nous remarquons finalement que l'hypothèse 4.11 (iii) requiert que l'hypothèse 4.10 (iv) soit satisfaite puisque

$$(\forall j \in \{1,\ldots,S\}) \qquad \sum_{\ell\in\mathbb{N}} \lambda_\ell \,\|a_{j,\ell}\| = \sum_{\ell\in\mathbb{N}} \lambda_\ell \,\|F\widetilde{a}_{j,\ell}\| \leq \|F\| \sum_{\ell\in\mathbb{N}} \lambda_\ell \,\|\widetilde{a}_{j,\ell}\| < +\infty. \tag{4.73}$$

Cela nous permet de transposer les résultats de convergence concernant l'algorithme 7 à l'algorithme 8. L'algorithme 8 converge donc vers une solution du problème (4.2). De plus, cet algorithme requiert seulement 3 applications de F ou $F^\top$ à chaque itération. Un gain par rapport à l'algorithme 7 est obtenu dès que $S \geq 2$. Ce fait sera illustré par le biais de résultats expérimentaux dans le paragraphe 4.5.2-a).

4.5 Application à la restauration d'image

4.5.1 Régularisation hybride

Dans les problèmes de restauration, l'un des termes du critère à minimiser est habituellement un terme d'attache aux données, qui mesure une certaine distance entre l'image dégradée par l'opérateur A et les observations z. Nous supposerons par la suite que cette fonction est de la forme $g = \Psi \circ A$ où $\Psi \in \Gamma_0(\mathbb{R}^M)$. Dans le cas de données dégradées par un bruit blanc gaussien de variance α, un choix usuel pour Ψ est une fonction quadratique telle que

$$(\forall u \in \mathbb{R}^M) \qquad \Psi(u) = \frac{1}{2\alpha}\|u - z\|^2. \tag{4.74}$$

Dans cette configuration, l'opérateur proximal associé à g peut être calculé explicitement en utilisant la propriété suivante :

Proposition 4.12 [Combettes, Pesquet, 2008, Proposition 2.6] *Soient $\mathcal{H}$ et $\mathcal{G}$ deux espaces de Hilbert réels, soit $L\colon \mathcal{H} \to \mathcal{G}$ un opérateur linéaire borné, soit $w \in \mathcal{G}$, soit $\gamma \in]0, +\infty[$ et soit $\varphi = \gamma\|L \cdot - z\|^2$. On a alors $\varphi \in \Gamma_0(\mathcal{H})$ et*

$$(\forall v \in \mathcal{H}) \qquad \operatorname{prox}_\varphi v = (\mathrm{Id} + \gamma L^* L)^{-1}(v - \gamma L^* w).$$

Dans le cas de données dégradées par un bruit indépendant de Poisson, de paramètre d'échelle α, un choix usuel est

$$(\forall u \in \mathbb{R}^M) \qquad \Psi(u) = D_{\mathrm{KLG}}(z, \alpha u) \tag{4.75}$$

où D_{KLG} est la divergence de Kullback-Leibler généralisée [Titterington, 1987; Fessler, 1995; Zheng *et al.*, 2000; Chouzenoux *et al.*, 2009] telle que,

$$\big(\forall u = (u_m)_{1 \le m \le M} \in \mathbb{R}^M\big), \qquad \Psi(u) = \sum_{m=1}^{M} \psi_m(u_m) \tag{4.76}$$

et

$$\psi_m(u_m) = \begin{cases} \alpha u_m - z_m + z_m \ln\left(\dfrac{z_m}{\alpha u_m}\right) & \text{si } z_m > 0 \text{ et } u_m > 0, \\ \alpha u_m & \text{si } z_m = 0 \text{ et } u_m \ge 0, \\ +\infty & \text{sinon.} \end{cases} \tag{4.77}$$

L'opérateur proximal de Ψ peut alors être déduit de l'exemple 4.2 et de la proposition 3.13 du chapitre 3. Cependant, pour un opérateur A quelconque, l'opérateur proximal de la fonction g associée ne possède pas de forme explicite.

Concernant les fonctions de régularisation, un choix classique de penalité est une fonction dans le domaine des trames d'ondelettes pouvant s'écrire,

$$(\forall x = (x_k)_{1\leq k\leq K} \in \mathbb{R}^K) \qquad \Phi(x) = \sum_{k=1}^{K} \phi_k(x_k)$$

où, pour tout $k \in \{1, \ldots, K\}$, ϕ_k est une fonction finie de $\Gamma_0(\mathbb{R})$ telle que

$$\lim_{|x_k|\to+\infty} \phi_k(x_k) = +\infty.$$

Les fonctions $(\phi_k)_{1\leq k\leq K}$ sont généralement des fonctions de puissance dont l'opérateur proximal est donné dans l'exemple 3.38 du chapitre 3 (se référer à [Daubechies *et al.*, 2004; Chaux *et al.*, 2007; Chaari *et al.*, 2010a] pour diverses applications où les fonctions de puissance sont considérées). Le principal problème lors d'une régularisation fondée sur les trames d'ondelettes est la présence d'artefacts visuels (e.g. des artefacts de « ringing »). Certains d'entre eux peuvent être atténués en augmentant la redondance de la représentation. Un autre type de régularisation peut être envisagé en employant une mesure de variation totale [Rudin *et al.*, 1992]. Son principal inconvénient est de générer des artefacts en *effet de marches d'escalier* [Louchet, 2008]. Pour combiner les avantages de ces deux régularisations, nous proposons de considérer le critère suivant :

$$\text{Trouver} \quad \hat{x} \in \underset{x\in\mathbb{R}^K}{\text{Argmin}} \quad \Psi(AF^\top x) + \mu\,\mathrm{tv}(F^\top x) + \iota_C(F^\top x) + \vartheta\,\Phi(x), \tag{4.78}$$

où l'image restaurée est $\hat{y} = F^\top \hat{x}$. Comme précédemment mentionné, Φ correspond à un terme de régularisation opérant dans le domaine des trames d'ondelettes. tv represente un terme de variation totale discrète comme défini par (4.47). Finalement, ι_C est la fonction indicatrice d'un ensemble convexe fermé C de $\mathbb{R}^N$ (par exemple, associé à une contrainte de support ou de dynamique $C = [0, 255]^N$). Ce type de fonction objectif a été récemment étudié dans [Combettes, Pesquet, 2008], cependant l'approche était réduite à l'utilisation d'un terme quadratique de type (4.74) pour le terme d'attache aux données et à une forme spécifique pour le terme de variation totale. Les paramètres réels et positifs ϑ et μ ajustent respectivement, les régularisations dans le domaine transformé et dans le domaine image.

La principale difficulté se posant lorsqu'on applique l'algorithme 6 à notre problème de restauration est qu'il nécessite de calculer les opérateurs proximaux associés à chacun des quatres termes de (4.78). Comme nous l'avons précédemment évoqué, une forme explicite est connue pour les opérateurs proximaux associés à la fonction indicatrice ι_C et à Φ. Cependant, comme expliqué dans le paragraphe 4.3.2 et en supposant que la fonction Ψ soit séparable, le terme d'attache aux données peut être décomposé comme une somme de I fonctions $(\Upsilon_i \circ A_i)_{1\leq i\leq I}$ pour lesquelles les opérateurs proximaux peuvent être calculés explicitement selon (4.21). De façon similaire, en utilisant les résultats du

paragraphe 4.3.3, la fonction tv peut être réécrite comme une somme de P_1P_2 fonctions $(\mathrm{tv}_{p_1,p_2})_{0\leq p_1<P_1,0\leq p_2<P_2}$, l'opérateur proximal de chacune d'entre elles étant donné par la proposition 4.8. L'algorithme 8 peut alors être employé avec

$$S = I + P_1P_2 + 1 \quad \text{et} \quad J = I + P_1P_2 + 2.$$

La convergence de l'algorithme est assurée, car si $\vartheta > 0$, l'hypothèse 4.11 (i) est satisfaite. De plus, l'hypothèse 4.11 (ii) est vérifiée lorsque

$$\big(\cap_{i=1}^{I} A_i^{-1}(\mathrm{ri}\,\mathrm{dom}\,\Upsilon_i)\big) \bigcap \mathrm{ri}\, C \neq \varnothing$$

car, $\mathrm{dom}\,\Phi = \mathbb{R}^K$ et, pour tout $(p_1,p_2) \in \{0,\ldots,P_1-1\}\times\{0,\ldots,P_2-1\}$, $\mathrm{dom}\,\mathrm{tv}_{p_1,p_2} = \mathbb{R}^N$. Cette condition est vérifiée car $]0,+\infty[^M \subset \mathrm{dom}\,\Psi$, $C = [0,255]^N$ et puisque pour tout $i \in \{1,\ldots,I\}$, A_i est supposée positive, à valeurs réelles et composée de lignes étant toutes non nulles (compte tenu de l'hypothèse 4.4).

4.5.2 Résultats expérimentaux pour la résolution de problèmes inverses en présence de bruit de Poisson

Dans nos simulations, nous nous intéresserons dans un premier temps aux performances en terme de temps de convergence de la version accélérée de PPXA. Les algorithmes 7 et 8 sont implantés en choisissant $\gamma = 50$, $\lambda_\ell \equiv 1.6$ et

$$(\forall j \in \{1,\ldots,J\}) \qquad \omega_j = \begin{cases} \frac{1}{4I} & \text{si } 1 \leq j \leq I \\ \frac{1}{4P_1P_2} & \text{si } I+1 \leq j \leq I+P_1P_2 \\ \frac{1}{4} & \text{sinon} \end{cases} \tag{4.79}$$

Les fonctions $(g_j)_{1\leq j\leq I}$ correspondent à la décomposition du terme d'attache aux données. Dans la suite, nous considérons une dégradation par un bruit de Poisson, le terme d'attache aux données sera donc la divergence de Kullback-Leibler généralisée, donnée par (4.75). Les fonctions $(g_j)_{I+1\leq j\leq I+P_1P_2}$ résultent de la décomposition de la fonction tv. Nous utilisons une version trame ajustée de la transformée en arbre dual (DTT) [Chaux *et al.*, 2006] de facteur $\nu = 2$, en utilisant des symmlets de longueur 6 sur 3 niveaux de résolution. Nous choisissons les fonctions de type potentiel de la forme $\phi_k = \chi_k|\cdot|^{p_k}$ où $\chi_k > 0$ et $p_k \in \{1, 4/3, 3/2, 2\}$ pour tout $k \in \{1,\ldots,K\}$. Une comparaison entre les différentes variation totale définies dans le paragraphe 4.4.2 sera effectuée. La discussion suivante portera sur les effets de bord, où deux types de convolution seront considérés : la convolution périodique et la convolution « zero-padding ». L'intérêt de combiner variation totale et régularisation ondelettes sera illustré et des comparaisons avec les méthodes existantes dans le cas poissonien seront effectuées. Dans ce paragraphe, cinq images vont être considérées. Elles sont présentées sur la figure 4.2.

(a)

(b)

(c)

(d)

(e)

FIGURE 4.2 – (a) Image « Sebal » - 128×128 (source : SHFJ), (b) Image « Boat » - 256×256 (source : http ://sipi.usc.edu/database), (c) « Image Peppers » - 256×256 (source : http ://sipi.usc.edu/database), (d) Image « Spot » (source : LIGM)- 256×256 et (e) Image « Marseille » - 512×512 (source : LIGM).

4.5.2-a) Comparaison du taux de convergence entre PPXA et sa version accélérée

Le tableau 4.1 donne le nombre d'itérations et le temps CPU pour l'agorithme dans sa forme originale et sa version accélérée [1]. Le tableau présente des résultats pour différentes tailles d'images (images de « Sebal », « Peppers » et « Marseille ») et différentes tailles de flou. Le critère d'arrêt est fixé sur l'erreur relative de la fonction objectif calculée à l'itération courante et à la précédente (la valeur de la fonction indicatrice n'est pas prise en compte). La tolérance est fixée à 10^{-3}. La dernière ligne du tableau 4.1 illustre le gain en temps CPU lorsque que l'on utilise l'algorithme 8. Sur la figure 4.3, nous avons représenté $\|F^\top(x_n - \hat{x})\|^2$ en fonction du temps CPU, où $(x_n)_{n>0}$ représente la suite générée par l'algorithme 7 ou l'algorithme 8.

Taille image	128×128		256×256		512×512	
Taille flou	3×3	7×7	3×3	7×7	3×3	7×7
Nombre d'itérations	30	50	41	50	50	50
Temps CPU (en sec.)	117.2	633.0	411.7	1298	1458	4514
Temps CPU - version acc. (en sec.)	13.53	29.82	60.59	89.48	263.6	405.0
Gain	8.67	21.2	6.79	14.5	5.53	11.1

TABLE 4.1 – Comparaisons entre PPXA et sa version accélérée.

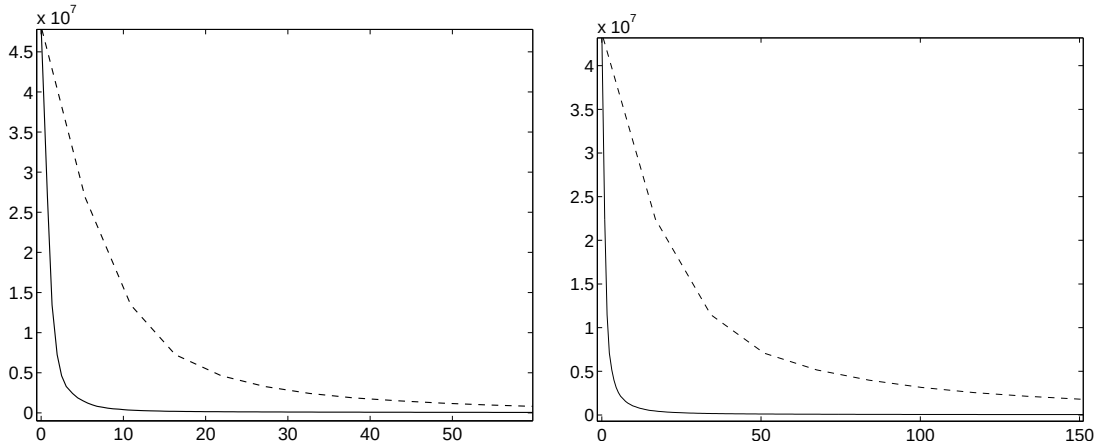


FIGURE 4.3 – Profils de convergence de l'algorithme 7 (ligne discontinue) et de l'algorithme 8 (ligne continue) en fonction du temps de calcul en secondes pour un flou uniforme de taille 3×3 (à gauche) et 7×7 (à droite) pour une image de taille 128×128.

On peut remarquer que plus la taille du noyau de flou est grand, plus le gain est important. Cela provient du fait que plus la taille du flou est importante, plus la valeur de S

1. Les résultats ont été obtenus avec un processeur Intel Core2 6700, 2.66 GHz.

est grande et donc, dans la version de base de l'algorithme, plus le nombre de passages du domaine image au domaine transformé est élevé.

4.5.2-b) Comparaison entre les différentes formes de variation totale

Dans le paragraphe 4.3.3, nous avons introduit l'opérateur proximal associé à la variation totale dans ses formes isotrope et anisotrope pour différents filtres H et V. Nos tests ont porté sur la qualité de restauration des images « Peppers » (figure 4.4 en haut) et « Boat » (figure 4.4 en bas) degradées par un flou uniforme de taille 3×3 et dégradées par un bruit de Poisson de paramètre d'échelle $\alpha = 0.1$. Nous nous proposons de comparer les valeurs de SNR (figure 4.4 courbes de gauche) et de SSIM (figure 4.4 courbes de droite) des images restaurées pour différentes configurations de variation totale présentées dans le paragraphe 4.3.3. Le SNR et le SSIM sont tracés en fonction de μ (le paramètre de régularisation associé à la variation totale), pour une forme donnée de tv (i.e. une fonction isotrope ou anisotrope de la fonction ρ_{tv} avec un filtre spécifique). Une petite valeur du paramètre de régularisation $\vartheta = 0.001$ a été choisie dans le but de mieux illustrer l'influence de chaque forme de tv sur la qualité de restauration.

La figure 4.4 nous permet de conclure que le choix du filtre du gradient et la forme (isotrope/anisotrope) de ρ_{tv} a une influence significative sur la qualité de restauration quand la régularisation ondelette est limitée. En effet, plus le support du filtre est large, plus les valeurs de SNR et SSIM sont faibles. Nous avons pu remarquer dans nos simulations que lorsque le paramètre de régularisation ϑ est plus grand, le choix de la forme de tv a une faible influence sur la qualité de restauration si les paramètres de régularisation sont bien choisis.

4.5.2-c) Effets de bord sur les images restaurées

Ce paragraphe illustre l'influence de la gestion des effets de bords sur la restauration. Plus précisemment, nous dégradons une version étendue de l'image « Boat » avec un flou uniforme de noyau de taille 7×7, et l'image floue résultante est tronquée pour créer une image de taille 256×256. Par conséquent, les valeurs des pixels frontières dépendent de valeurs de pixels qui ne sont plus présents dans l'image dégradée. L'image tronquée est dégradée par un bruit de Poisson de paramètre d'échelle $\alpha = 0.5$. L'objectif est alors de restaurer l'image (qui a été centrée) en utilisant un des modèles de convolution discuté dans le paragraphe 4.3.2, à savoir soit une convolution périodique, soit une convolution « zero-padding ». Des résultats visuels et quantitatifs sont donnés figure 4.5.

Comme on peut le remarquer sur la figure 4.5, le modèle de convolution périodique introduit des artefacts de bord significatifs contrairement à la convolution avec « zero-padding ». Les résultats obtenus en considérant « Marseille », « Peppers » et « Spot » conduisent à la même conclusion. Pour « Sebal », la convolution par « zero-padding »

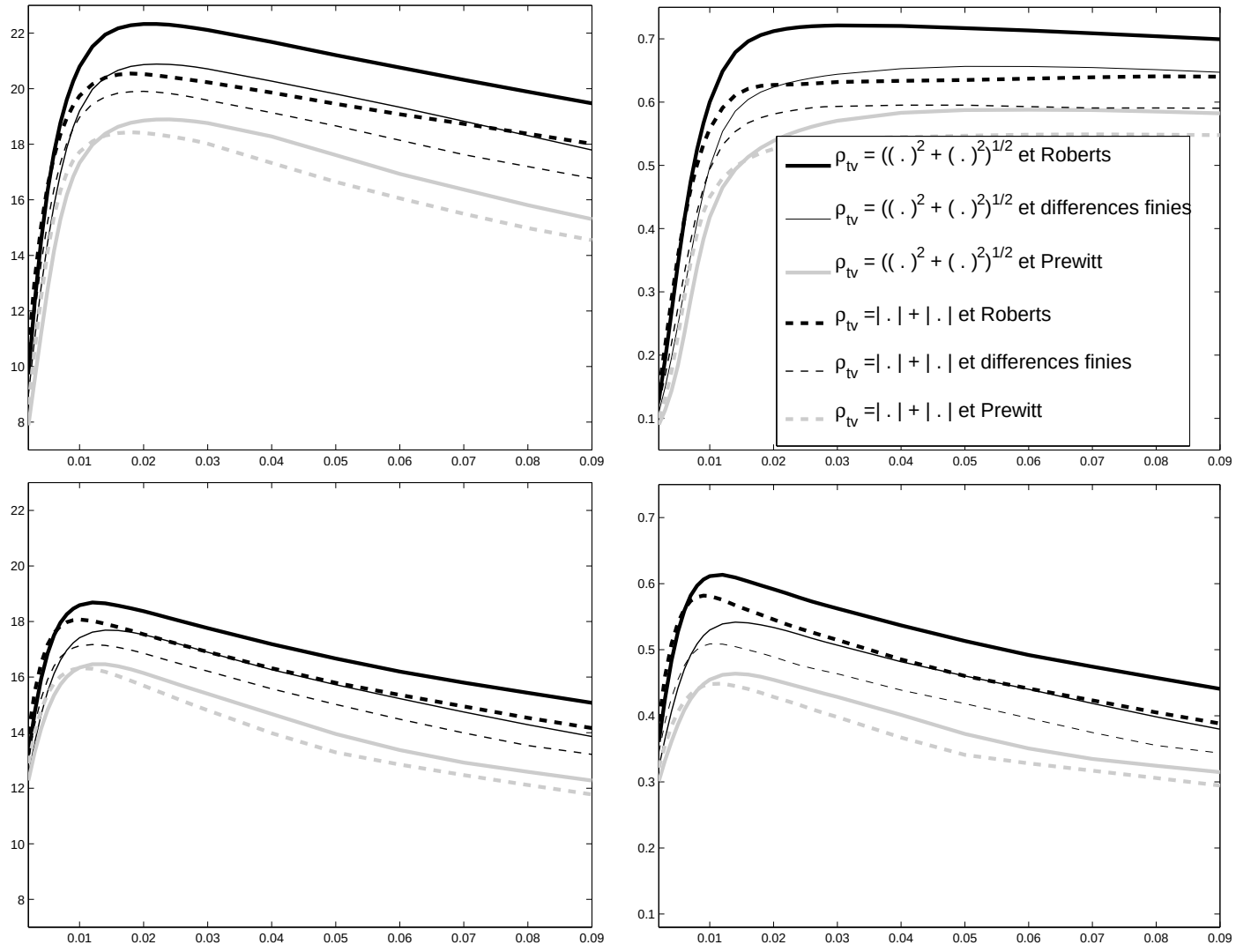


FIGURE 4.4 – SNR (gauche) et SSIM (droite) pour différents termes de variation totale en fonction de μ et pour deux images différentes (« Peppers » en haut et « Boat » en bas).

SNR = 16.9 dB - SSIM = 0.62 SNR = 17.7 dB - SSIM = 0.64

FIGURE 4.5 – Restauration avec gestion des bords périodique (gauche) et gestion des bords « zero-padded » (droite).

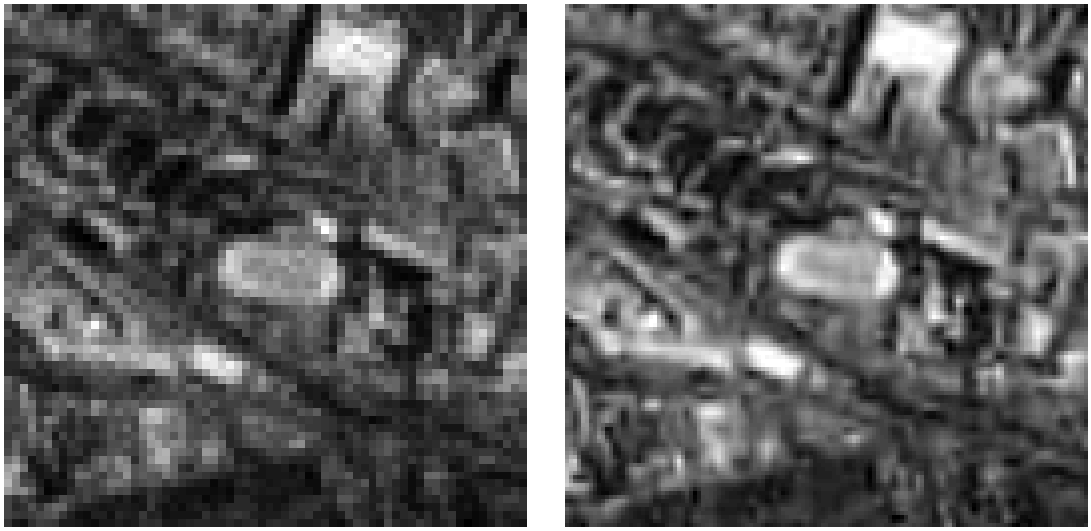
ou « périodique » fournissent des résultats similaires, en raison de la forme spécifique de cette image.

4.5.2-d) Convolution décimée

Nous présentons ici des résultats pour une l'image « Spot » dégradée par un flou uniforme, de noyau de taille $Q = 3 \times 3$, décimé d'un facteur $d_1 = d_2 = 2$. Le paramètre d'échelle du bruit de Poisson est fixé à $\alpha = 1$. En raison de la structure de l'opérateur de dégradation, le terme de fidélité est décomposé en une somme de $I = 4$ fonctions. Les résultats sont présentés figure 4.6 où le bon comportement de notre modèle peut être observé. Remarquons que les approches de type « lagrangien augmenté » utilisées dans [Setzer *et al.*, 2010; Figueiredo, Bioucas-Dias, 2010] sont plus difficilement envisageables dans ce contexte, car la matrice de dégradation n'est plus diagonalisable dans le domaine de Fourier.

4.5.2-e) Influence de chaque terme de régularisation

Dans ce paragraphe, nous allons comparer les résultats de la régularisation hybride avec ceux obtenus en considérant des approches existantes comme les trames d'ondelettes [Figueiredo, Bioucas-Dias, 2010] ou la variation totale [Setzer *et al.*, 2010; Figueiredo, Bioucas-Dias, 2010]. Les méthodes fondées sur une régularisation non hybride peuvent



Dégradée

Restaurée $(\vartheta = 1, \mu = 0.001)$
SNR = 16.1 dB - SSIM = 0.79

FIGURE 4.6 – Résultats de restauration sur un zoom de l'image « spot ».

aussi bien être calculées avec des techniques de lagrangien augmenté [Setzer *et al.*, 2010; Figueiredo, Bioucas-Dias, 2010] ou avec notre approche par décomposition en choisissant $\mu = 0$ (resp. $\vartheta = 0$) pour la régularisation par trame seule (resp. par variation totale seule). Les deux approches conduisent à des temps de calcul similaires. Sur les images présentées figures 4.7, 4.8, 4.9, et 4.10, nous pouvons observer les artefacts introduits par la régularisation avec des trames d'ondelettes, l'effet de « marche d'escalier » que l'on peut observer avec la régularisation par variation totale et également le bénéfice résultant d'une régularisation hybride.

Le tableau 4.2 présente des comparaisons en terme de SNR de notre approche hybride avec les méthodes existantes de restauration d'images en présence de bruit de Poisson, à savoir,

- (1) l'approximation d'Anscombe avec régularisation ℓ_1 sur des coefficients de trame [Dupé *et al.*, 2009],
- (2) l'extension quadratique avec régularisation ℓ_1 sur des coefficients de trame [Chaux *et al.*, 2009],
- (3) la méthode de lagrangien augmenté avec régularisation ℓ_1 sur des coefficients de trame [Figueiredo, Bioucas-Dias, 2010],
- (4) la méthode de lagrangien augmenté avec régularisation tv [Setzer *et al.*, 2010].

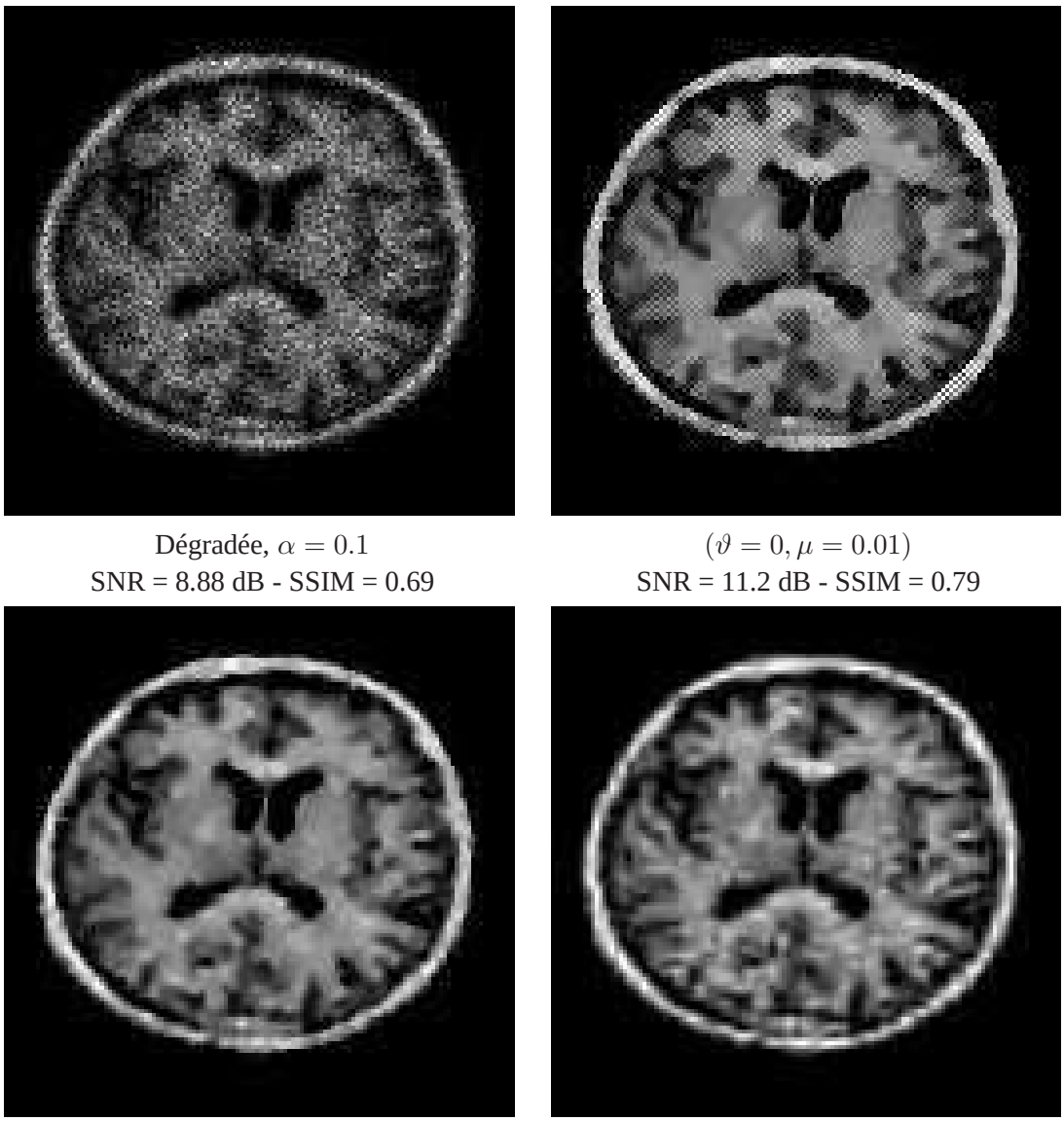
Les tests sont effectués sur les images « Peppers », « Boat » et « Spot » pour différentes valeurs du facteur d'échelle α et différentes tailles de flou.

De façon identique à [Figueiredo, Bioucas-Dias, 2010], le choix des paramètres a été

optimisé par une mesure liée à l'image originale, dans notre cas le SNR. Notons que l'optimisation manuelle des hyperparamètres comme nous le faisons est une pratique courante dans des applications où un ensemble de données tests possède les mêmes caractéristiques (images médicales, images satellitaires,...). Des méthodes automatiques pour l'optimisation d'hyperparamètres peuvent aussi être trouvées dans la littérature telles que les méthodes de validation croisée [Galatsanos, Katsaggelos, 1992], les approches stochastique EM [Delyon *et al.*, 1999], les approches MCMC [Robert, Castella, 2004; Chaari *et al.*, 2010b] ou encore les méthodes de Stein [Ramani *et al.*, 2008]. Ces méthodes automatiques nécessitent souvent de lourds calculs.

4.6 Conclusion

Dans le présent chapitre, nous avons présenté une nouvelle approche, dite par régularisation hybride, qui permet de résoudre un large éventail de problèmes inverses. Les principaux avantages de cette approche sont (i) de pouvoir traiter directement la « vraie » vraisemblance du bruit (i.e. la divergence de Kullback-Leibler généralisée dans le cas du bruit de Poisson) sans utiliser une quelconque appoximation ; (ii) d'utiliser une régularisation sophistiquée, permettant de considérer la parcimonie dans le domaine ondelettes et la variation totale dans le domaine image. De plus, l'algorithme proposé a une structure parallélisable qui en fait un algorithme facilement implantable sur des architectures multicœurs. Des résultats visuels et numériques montrent l'efficacité de l'approche proposée par rapport à l'état de l'art. Nous pouvons noter que, si ce paragraphe est dévolu au cas des opérateurs de convolution et convolution décimée, cette approche peut être étendue à d'autres opérateurs linéaires. Le cas de la tomographie par émission de positrons est une application tout à fait adaptée comme nous le démontrerons dans le chapitre 7.



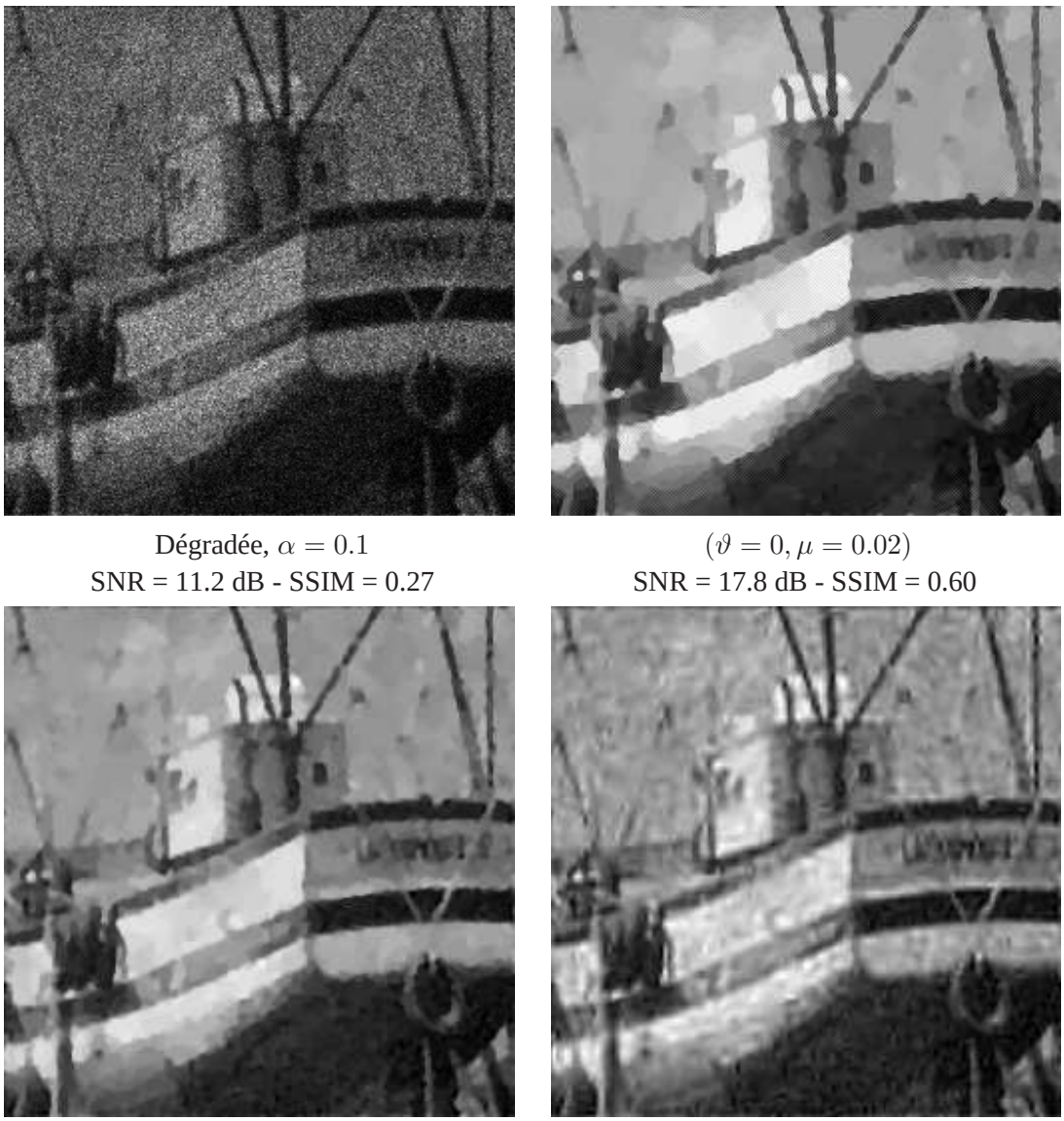
Dégradée, $\alpha = 0.1$
SNR = 8.88 dB - SSIM = 0.69

$(\vartheta = 0, \mu = 0.01)$
SNR = 11.2 dB - SSIM = 0.79

$(\vartheta = 0.09, \mu = 0.006)$
SNR= 12.4 dB - SSIM= 0.85

$(\vartheta = 0.5, \mu = 0)$
SNR= 11.7 dB - SSIM= 0.83

FIGURE 4.7 – Résultats de restauration pour l'image « Sebal ».



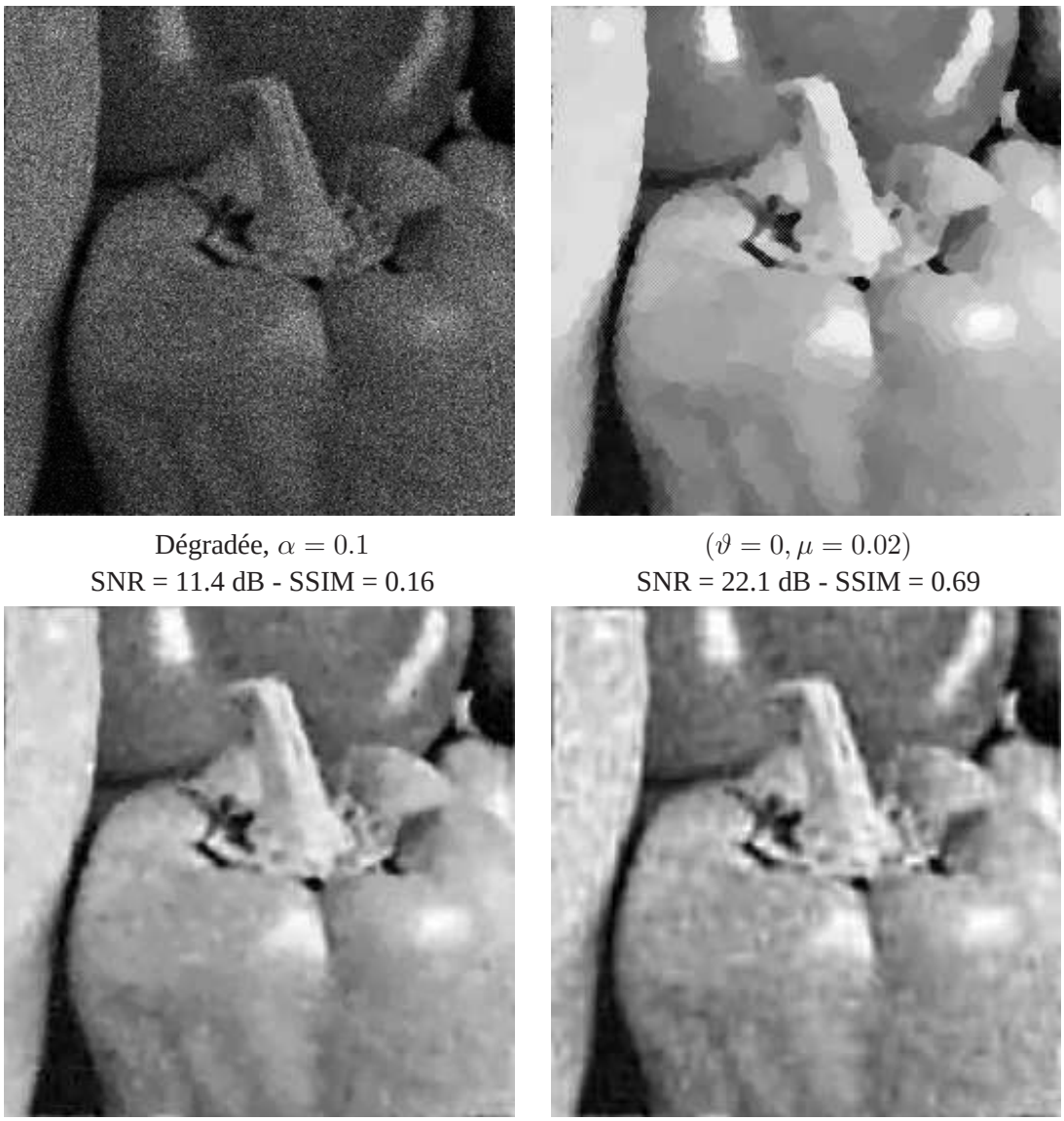
Dégradée, $\alpha = 0.1$
SNR = 11.2 dB - SSIM = 0.27

$(\vartheta = 0, \mu = 0.02)$
SNR = 17.8 dB - SSIM = 0.60

$(\vartheta = 0.06, \mu = 0.011)$
SNR=18.8 dB - SSIM=0.67

$(\vartheta = 0.7, \mu = 0)$
SNR=18.0 dB - SSIM= 0.62

FIGURE 4.8 – Résultats de restauration pour l'image « Boat ».



Dégradée, $\alpha = 0.1$
SNR = 11.4 dB - SSIM = 0.16

$(\vartheta = 0, \mu = 0.02)$
SNR = 22.1 dB - SSIM = 0.69

$(\vartheta = 0.2, \mu = 0.006)$
SNR = 22.7 dB - SSIM = 0.74

$(\vartheta = 0.7, \mu = 0)$
SNR= 21.4 dB - SSIM = 0.68

FIGURE 4.9 – Résultats de restauration pour l'image « Peppers ».

Dégradée, $\alpha = 0.1$
SNR = 7.36 dB - SSIM = 0.42

$(\vartheta = 0, \mu = 0.02)$
SNR = 11.4 dB - SSIM = 0.61

$(\vartheta = 0.09, \mu = 0.011)$
SNR = 12.8 dB - SSIM = 0.70

$(\vartheta = 0.8, \mu = 0)$
SNR= 12.4 dB - SSIM = 0.69

FIGURE 4.10 – Résultats de restauration pour l'image « Marseille ».

« Boat »						
α	Taille flou	(1) : Anscombe	(2) : Ext. quad.	(3) : $D_{\mathrm{KLG}} + \ell_1$	(4) $D_{\mathrm{KLG}} + \mathrm{tv}$	Notre approche
1	3×3	20.9	20.9	20.9	20.9	**22.2**
0.5	3×3	20.2	20.2	20.2	20.0	**21.2**
0.1	3×3	17.6	17.8	17.9	17.8	**18.7**
0.05	3×3	16.4	16.8	16.8	17.0	**17.7**
0.01	3×3	12.3	14.1	14.1	14.6	**14.9**
1	7×7	17.4	17.5	17.5	17.9	**18.5**
0.5	7×7	17.1	17.1	17.1	17.4	**18.0**
0.1	7×7	15.6	15.7	15.7	16.1	**16.4**
0.05	7×7	14.9	15.1	15.1	15.3	**15.9**
0.01	7×7	11.7	13.0	13.0	**13.9**	**13.9**

« Peppers »						
α	Taille flou	(1) : Anscombe	(2) : Ext. quad.	(3) : $D_{\mathrm{KLG}} + \ell_1$	(4) $D_{\mathrm{KLG}} + \mathrm{tv}$	Notre approche
1	3×3	25.1	25.1	25.1	25.3	**26.4**
0.5	3×3	24.2	24.2	24.2	24.4	**25.6**
0.1	3×3	21.2	21.5	21.5	22.2	**23.1**
0.05	3×3	19.4	20.1	20.1	21.2	**21.9**
0.01	3×3	13.9	16.4	16.4	18.6	**19.0**
1	7×7	22.3	22.4	22.4	23.1	**23.6**
0.5	7×7	21.6	21.7	21.7	22.6	**23.2**
0.1	7×7	19.6	19.8	19.8	21.2	**21.7**
0.05	7×7	18.2	18.6	18.6	20.3	**20.7**
0.01	7×7	13.5	15.9	15.9	18.2	**18.4**

« Spot »						
α	Taille flou	(1) : Anscombe	(2) : Ext. quad.	(3) : $D_{\mathrm{KLG}} + \ell_1$	(4) $D_{\mathrm{KLG}} + \mathrm{tv}$	Notre approche
1	3×3	17.4	17.4	17.4	16.2	**17.9**
0.5	3×3	16.5	16.5	16.5	15.3	**17.0**
0.1	3×3	14.1	14.2	14.2	13.0	**14.5**
0.05	3×3	12.7	13.0	13.0	12.1	**13.3**
0.01	3×3	9.45	**10.5**	**10.5**	10.0	**10.5**
1	7×7	13.0	13.0	13.0	12.6	**13.5**
0.5	7×7	12.5	12.5	12.5	12.2	**12.7**
0.1	7×7	11.3	11.3	11.3	11.2	**11.5**
0.05	7×7	10.7	10.8	10.8	10.6	**10.8**
0.01	7×7	8.72	**9.56**	**9.56**	9.45	**9.56**

TABLE 4.2 – Comparaison en terme de SNR (dB) des méthodes de déconvolution en présence de bruit de Poisson pour différentes valeurs de facteur d'échelle α et différentes tailles de flou.

CHAPITRE 5

Comparaison des approches trame à l'analyse et trame à la synthèse dans le cadre des algorithmes proximaux

5.1 Motivations

Dans les chapitres précédents, nous avons vu qu'il pouvait être avantageux que l'un des termes de régularisation s'applique sur les coefficients de trame d'ondelettes. A ce niveau, deux stratégies peuvent être envisagées : le critère peut être minimisé par rapport à l'image (formulation trame à l'analyse) ou par rapport aux coefficients de trame d'ondelettes (approche trame à la synthèse). Plus précisément, si N correspond au nombre de pixels de l'image, $K \geq N$ est le nombre de coefficients de trame et $F \in \mathbb{R}^{K\times N}$ (resp. $F^\top \in \mathbb{R}^{N\times K}$) désigne l'opérateur de trame à l'analyse (resp. à la synthèse), alors la formulation à l'analyse (FA) consiste à trouver :

$$\widehat{y}_{\mathrm{FA}} \in \underset{y\in\mathbb{R}^N}{\mathrm{Argmin}} \sum_{i=1}^{S} h_i(L_i y) + \sum_{j=S+1}^{J} f_j(Fy), \tag{5.1}$$

et la formulation à la synthèse (FS) vise à trouver :

$$\widehat{x}_{\mathrm{FS}} \in \underset{x\in\mathbb{R}^K}{\mathrm{Argmin}} \sum_{i=1}^{S} h_i(L_i F^\top x) + \sum_{j=S+1}^{J} f_j(x). \tag{5.2}$$

On désigne par $\widehat{y}_{\mathrm{FA}}$ et $\widehat{y}_{\mathrm{FS}} = F^\top \widehat{x}_{\mathrm{FS}}$ les images restaurées/reconstruites par les deux formulations. Nous supposons dans la suite que pour tout $i \in \{1,\ldots,S\}$, $L_i \in \mathbb{R}^{M_i\times N}$ est un opérateur linéaire. De plus, pour tout $i \in \{1,\ldots,S\}$, les fonctions h_i appartiennent à la classe $\Gamma_0(\mathbb{R}^{M_i})$ et, pour tout $j \in \{S+1,\ldots,J\}$, les f_j sont des fonctions de $\Gamma_0(\mathbb{R}^K)$. Ces deux formulations, relativement générales, permettent de résoudre efficacement une large classe de problèmes inverses.

Quelques travaux comparant ces deux approches ont récemment été publiés. Dans [Elad *et al.*, 2007], les auteurs considèrent un critère composé d'un terme d'attache aux données quadratique et d'une norme ℓ_p comme « terme *a priori* ». Leur formulation est

un cas particulier de (5.1) et (5.2), avec $J = 2$ et $S = 1$. La matrice $L_1 \in \mathbb{R}^{M\times N}$ est associée à l'opérateur de dégradation, la fonction $h_1 = \|\cdot - z\|^2$ désigne le terme de fidélité aux données $z \in \mathbb{R}^M$ et la fonction $f_2 = \chi\|\cdot\|_p^p$ correspond au terme d'*a priori* dont le paramètre de régularisation est $\chi > 0$. Pour $p = 2$ ou lorsque $F^\top = F^{-1}$, *i.e.* F est une décomposition orthonormale, les auteurs montrent l'équivalence des deux formulations. Plus récemment, une étude expérimentale a été menée dans le cas où $p = 1$ dans [Carlavan *et al.*, 2009]. Pour résoudre numériquement la formulation à la synthèse, les auteurs utilisent un algorithme proposé par Nesterov dans [Nesterov, 2009]. Rappelons que cet algorithme est lié à la classe des algorithmes explicite-implicite, présentée dans le chapitre 3. Concernant la formulation à l'analyse, l'utilisation d'un tel algorithme est limitée par le calcul de l'opérateur proximal de $\|F\cdot\|_1$. En effet, si l'on se reporte à la proposition 4.3 du chapitre 4, cet opérateur proximal ne possède une forme explicite que si le produit de F et $F^\top$ conduit à une matrice diagonale. Dans sa formulation générale, une trame vérifie rarement ce genre de condition. Notons que dans le cas de trames ajustées, par définition nous pouvons écrire $F^\top F = \nu\,\mathrm{Id}$ avec $\nu > 0$, alors qu'il n'existe pas de $\widetilde{\nu} > 0$ tel que $FF^\top = \widetilde{\nu}\,\mathrm{Id}$. Pour contourner cette difficulté, les auteurs approximent la norme ℓ_1 par une fonction de différentiable de Huber. Ils peuvent ainsi utiliser un algorithme de type descente de gradient [Nesterov, 2007] pour minimiser le critère approximé résultant.

Bien que les travaux d'Elad *et al.* et de Carvalan *et al.* laissent à penser que la FA permet d'obtenir des résultats tout aussi performants voir plus performants que la FS, il apparaît souvent plus facile de résoudre numériquement une FS plutôt qu'une FA avec les algorithmes proximaux. En effet, la remarque faite dans le paragraphe précédent sur le calcul de l'opérateur proximal associé à $\|F\cdot\|_1$, s'étend au calcul de $\mathrm{prox}_{f_j\circ F}$. Dans les chapitres 3 et 4, des FS ont été formulées pour cette raison. Lorsqu'on effectue un bref récapitulatif, sur les algorithmes proximaux utilisés et les caractéristiques des trames employées pour résoudre des problèmes inverses associés à une FS, il apparaît que de nombreux algorithmes requièrent la condition de trame ajustée. Par exemple, le problème de débruitage en présence de bruit de Poisson ou de bruit de speckle a été résolu avec l'algorithme de Douglas-Rachford et des trames ajustées dans [Combettes, Pesquet, 2007a]. Dans le cas d'un problème de déconvolution en présence de bruit non-additif gaussien, il est possible de faire appel à l'algorithme proximal parallèle [Combettes, Pesquet, 2008] comme nous l'avons proposé dans le chapitre 4, ou d'utiliser les méthodes de lagrangien augmenté [Afonso *et al.*, 2009; Setzer *et al.*, 2010]. Dans les travaux existants, ces deux classes d'algoritmes sont également limitées à l'utilisation de trames ajustées. Seuls les travaux basés sur l'algorithme explicite-implicite, présentés dans le chapitre 3, permettent de résoudre des problèmes de déconvolution en présence de bruit gaussien par le biais de trames non-nécessairement ajustées [Chaux *et al.*, 2007].

Dans ce chapitre, nous nous fixons deux objectifs. Le premier est de proposer des techniques permettant de résoudre la FA et la FS, sans approximation des critères, par le biais d'un même algorithme proximal. Le second objectif vise à considérer des algorithmes permettant d'utiliser des trames non-ajustées. Pour atteindre ce double objectif, nous utiliserons successivement l'algorithme explicite-implicite [Combettes, Wajs, 2005] et PPXA+ [Pesquet, 2010]. Dans un premier temps, nous détaillerons le formalisme à l'analyse et celui à la synthèse ainsi que les correspondances existantes ; en particulier, nous étendrons les travaux d'Elad *et al.*. Puis, les problèmes (5.1) et (5.2), dans le cas où $J = 2$ et $S = 1$, seront résolus à l'aide de l'algorithme explicite-implicite. Nous utiliserons la dualité de Fenchel-Rockafellar, exploitée dans [Combettes *et al.*, 2009], pour apporter une solution au calcul de $\mathrm{prox}_{f_j \circ F}$. Une série de résultats expérimentaux permettra de comparer FA et FS. Enfin, la section 5.4 se penche sur la résolution des problèmes (5.1) et (5.2), dans leur forme générale, pour une classe particulière de trames non-ajustées. L'intérêt d'utiliser des trames non ajustées et des comparaisons supplémentaires entre FA et FS seront présentées.

5.2 Formulation à l'analyse et formulation à la synthèse

De façon à définir les formulations à l'analyse et à la synthèse de manière intuitive, nous passons par une formulation statistique. Les observations z sont une réalisation d'un vecteur aléatoire à valeurs réelles $Z = (Z_i)_{1 \leq i \leq M}$. Le vecteur de données à restaurer/reconstruire y est une réalisation d'un vecteur aléatoire à valeurs réelles $Y = (Y_i)_{1 \leq i \leq N}$ et le vecteur x des coefficients de trame (tels que $y = F^\top x$) est une réalisation d'un vecteur aléatoire $X = (X_i)_{1 \leq i \leq N}$. Notons que les observations sont obtenues après une dégradation linéaire, notée $L_1 \in \mathbb{R}^{M \times N}$, suivie d'une perturbation aléatoire.

En se basant sur la distribution de probabilité *a posteriori* de Y ou X, les estimateurs envisageables pour retrouver une estimée $\widehat{y}$ de $\overline{y}$ sont la « moyenne *a posteriori* » (MP) et le « maximum *a posteriori* » (MAP). Notre attention portera essentiellement sur l'estimateur MAP. Cependant, remarquons que la MP peut être utilisée lorsqu'il est possible d'intégrer la distribution *a posteriori*. Si tel n'est pas le cas, ce calcul peut être obtenu par le biais de méthodes de Monte Carlo. Cette approche ne sera pas suivie dans ce chapitre, cependant plus de détails peuvent être trouvés dans [Robert, Castella, 2004; Chaari *et al.*, 2010b] et les références associées.

5.2.1 Formulation à l'analyse (FA)

On note $\mu_{Y|Z=z}$ la « distribution *a posteriori* » ; l'approche MAP s'écrit alors :

$$\max_{y \in \mathbb{R}^N} \ \mu_{Y|Z=z}(y). \tag{5.3}$$

En utilisant la formule de Bayes, l'égalité (5.3) se formule :

$$\max_{y\in\mathbb{R}^N} \ \mu_{Z|Y=y}(z)\ \mu_Y(y), \tag{5.4}$$

où la vraisemblance et la « distribution *a priori* » sont respectivement notées $\mu_{Z|Y=y}$ et μ_Y. On écrit la vraisemblance ainsi que la « distribution *a priori* » sous formes exponentielles, *i.e.*

$$\mu_{Z|Y=y}(z) \propto \exp\left(-\widehat{h}_1(L_1 y, z)\right) \tag{5.5}$$

$$\mu_Y(y) \propto \exp\left(-f_2(Fy)\right) \tag{5.6}$$

où $\widehat{h}_1 \colon \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}$ et $f_2 \colon \mathbb{R}^K \to \mathbb{R}$. Nous pouvons remarquer que la « distribution *a priori* » s'applique sur une transformation directe de l'image à retrouver.

La loi *a priori* définit une densité de probabilité si, par exemple,

$$(\forall x \in \mathbb{R}^K) \qquad f_2(x) \geq \rho\|x\|^p \qquad \text{où } \rho > 0 \text{ et } p > 0.$$

En effet, pour un tel choix de fonction f_2, nous pouvons écrire,

$$\int_{\mathbb{R}^N} e^{-f_2(Fy)} dy \leq \int_{\mathbb{R}^N} e^{-\rho\|Fy\|^p} dy \leq \int_{\mathbb{R}^N} e^{-\rho\underline{\nu}^{p/2}\|y\|^p} dy < +\infty.$$

où, $\underline{\nu}$ désigne la constante de trame définie par l'équation (2.16) dans le chapitre 2. Nous déduisons des équations (5.3), (5.5) et (5.6), le problème d'optimisation (estimateur MAP) qui consiste à trouver :

$$\widehat{y}_{\mathrm{FA}} \in \operatorname*{Argmin}_{y\in\mathbb{R}^N} \widehat{h}_1(L_1 y, z) + f_2(Fy), \tag{5.7}$$

Le résultat précédent est un cas particulier de la formulation générale à l'analyse (5.1), lorsque, pour tout $z \in \mathbb{R}^K$, $\widehat{h}_1(\cdot, z) = h_1$ est une fonction convexe, $S = 1$ et $J = 2$.

5.2.2 Formulation à la synthèse (FS)

Lorsqu'on considère la FS, une estimée $\widehat{y}_{\mathrm{FS}}$ de l'image originale $\overline{y}$ sera obtenue en choisissant $\widehat{y}_{\mathrm{FS}} = F^\top \widehat{x}_{\mathrm{FS}}$ où $\widehat{x}_{\mathrm{FS}}$ correspond à l'estimée du critère MAP, qui en utilisant la loi de Bayes s'écrit :

$$\widehat{x}_{\mathrm{FS}} \in \operatorname*{Argmax}_{x\in\mathbb{R}^K} \mu_{Z|X=x}(z)\ \mu_X(x) \tag{5.8}$$

On considère une vraisemblance de forme similaire à la section précédente, à savoir,

$$\mu_{Z|X=x}(z) \propto \exp\left(-\widehat{h}_1(L_1 F^\top x, z)\right) \tag{5.9}$$

où $\widehat{h}_1 \colon \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}$. La « distribution *a priori* », notée μ_X, modélise la représentation du signal dans un dictionnaire et peut se formuler par,

$$\mu_X(x) \propto \exp\left(-f_2(x)\right). \tag{5.10}$$

où $f_2 \colon \mathbb{R}^K \to \mathbb{R}$. Le problème d'optimisation s'écrit alors :

$$\widehat{x}_{\mathrm{FS}} \in \operatorname*{Argmin}_{x \in \mathbb{R}^K} \widetilde{h}_1(L_1 F^\top x, z) + f_2(x). \tag{5.11}$$

Le résultat précédent correspond à un cas particulier de la formulation générale à la synthèse 5.2, lorsque, pour tout $z \in \mathbb{R}^K$, $\widehat{h}_1(\cdot, z) = h_1$ est une fonction convexe, $S = 1$ et $J = 2$.

5.2.3 Comparaison théorique FS versus FA

Notons que pour chaque formulation, le critère pénalisé résultant, (5.7) ou (5.11), est composé de deux termes : un terme d'attache aux données $h_1(L_1 \cdot, z)$ et un terme de pénalisation f_2. Comme mentionné dans le chapitre 4, il peut parfois être judicieux de considérer des régularisations hybrides et des contraintes additionnelles. Cela permet souvent d'améliorer la qualité de restauration/reconstruction. Dans un cadre plus général, les formulations à l'analyse (5.7) et à la synthèse (5.11) nous conduisent aux problèmes présentés dans l'introduction de ce chapitre, à savoir aux équations (5.1) et (5.2). Nous proposons ci-dessous des résultats théoriques sur les équivalences entre FA et FS.

Proposition 5.1 *La formulation à l'analyse* (5.1) *est un cas particulier de la formulation à la synthèse* (5.2).

Preuve :
Par définition d'une trame, $F^\top$ est surjective. Par conséquent, pour tout $y \in \mathbb{R}^N$, il existe un élément x de $\mathbb{R}^K$ tel que $y = F^\top x$. Par conséquent, (5.1) peut s'écrire comme

$$\widehat{y}_{\mathrm{FA}} \in \operatorname*{Argmin}_{y = F^\top x} \sum_{i=1}^{S} h_i(L_i F^\top x) + \sum_{j=S+1}^{J} f_j(F F^\top x),$$

c'est à dire

$$\widehat{y}_{\mathrm{FA}} \in F^\top \operatorname*{Argmin}_{x} \sum_{i=1}^{S} h_i(L_i F^\top x) + \sum_{j=S+1}^{J} \widehat{f}_j(x),$$

avec, pour tout $j \in \{S+1, \ldots, J\}$, $\widehat{f}_j = f_j(F F^\top \cdot)$. □

Proposition 5.2 *Si F désigne l'opérateur d'analyse d'une trame ajustée et $F^\top$ son opérateur de synthèse, tel que $\ker F^\top \neq \{0\}$. Si, pour tout $j \in \{S+1, \ldots, J\}$, f_j s'écrit comme une somme de fonctions $f_{j,1}$ de $\Gamma_0(\mathrm{Im}F)$ et $f_{j,2}$ de $\Gamma_0(\ker F^\top)$, i.e.,*

$$\left(\forall (y, x_\perp) \in \mathbb{R}^N \times \ker F^\top\right) \qquad f_j(Fy + x_\perp) = f_{j,1}(Fy) + f_{j,2}(x_\perp), \tag{5.12}$$

Enfin, si pour tout $j \in \{S+1, \ldots, J\}$, et, pour tout $u \in \ker F^\top$, $f_{j,2}(u) \geq f_{j,2}(0)$. La formulation à la synthèse (5.2) *et la formulation à l'analyse* (5.1) *sont alors équivalentes.*

Preuve :
Supposons que $\ker F^\top \neq \{0\}$ alors $\mathbb{R}^K = \mathrm{Im}F \oplus (\mathrm{Im}F)^\perp$. Par conséquent, tout élément $x \in \mathbb{R}^K$ peut s'écrire comme une somme d'un élément appartenant à l'image de F et un élément appartenant au noyau de $F^\top$ car $\ker F^\top = (\mathrm{Im}F)^\perp$, *i.e.*,

$$\left(\forall x \in \mathbb{R}^K\right) \left(\exists (y, x_\perp) \in \mathbb{R}^N \times \ker F^\top\right) \qquad x = Fy + x_\perp.$$

Par conséquent, $F^\top x = F^\top F y$. L'équation (5.2) peut alors s'écrire,

$$\min_{\substack{y \in \mathbb{R}^N \\ x_\perp \in \ker F^\top}} \sum_{i=1}^{S} h_i(L_i F^\top F y) + \sum_{j=S+1}^{J} f_j(Fy + x_\perp). \tag{5.13}$$

On désigne par F l'opérateur d'analyse d'une trame ajustée telle que $F^\top F = \nu\,\mathrm{Id}$ (avec $\nu > 0$). L'équation (5.13) nous conduit à

$$\min_{\substack{y \in \mathbb{R}^N \\ x_\perp \in \ker F^\top}} \sum_{i=1}^{S} \widehat{h}_i(L_i y) + \sum_{j=S+1}^{J} f_j(Fy + x_\perp), \tag{5.14}$$

avec pour tout $i \in \{1, \ldots, S\}$, $\widehat{h}_i = h_i(\nu \cdot)$.
Enfin, comme pour tout $j \in \{S+1, \ldots, J\}$, f_j est séparable au sens de l'équation (5.12), l'équation (5.14) s'écrit donc,

$$\min_{y \in \mathbb{R}^N} \left\{ \sum_{i=1}^{S} \widehat{h}_i(L_i y) + \sum_{j=S+1}^{J} f_{j,1}(Fy) \right\} + \min_{x_\perp \in \ker F^\perp} \sum_{j=S+1}^{J} f_{j,2}(x_\perp). \tag{5.15}$$

De plus, pour tout $j \in \{S+1, \ldots, J\}$, pour tout $u \in \ker F^\top$, $f_{j,2}(u) \geq f_{j,2}(0)$, et l'équation (5.15) se réduit à

$$\min_{y \in \mathbb{R}^N} \sum_{i=1}^{S} \widehat{h}_i(L_i y) + \sum_{j=S+1}^{J} f_{j,1}(Fy),$$

c'est à dire une formulation à l'analyse. □

Remarque 5.3

- *Un cas particulier de fonction f_j satisfaisant la séparabilité, définie par l'équation* (5.12)*, est le carré de la norme ℓ_2. Ce cas de figure est présenté dans [Elad* et al.*, 2007].*
- *Dans le cas où $\ker F^\top = \{0\}$, la composante $x_\perp$ est nulle. Par conséquent, l'hypothèse de séparabilité n'est plus nécessaire. Ce cas de figure n'apparaît que lorsque $F^\top$ est l'opérateur de synthèse d'une transformation orthonormale.*

Cette partie visait à clarifier les liens existants entre FA et FS. Il apparaît que de toute FA, on peut déduire une FS conduisant au même résultat. Par contre l'inverse n'est pas toujours vérifié. Dans les sections suivantes, nous nous interrogeons sur la façon de résoudre numériquement chacune de ces deux formulations.

5.3 Algorithme explicite-implicite pour des trames non nécessairement ajustées

Considérons les formulations (5.1) et (5.2) lorsque $S = 1$ et $J = 2$. On veut donc trouver

$$\widehat{y}_{\mathrm{FA}} \in \underset{y \in \mathbb{R}^N}{\mathrm{Argmin}}\ h_1(L_1 y) + f_2(Fy), \tag{5.16}$$

et

$$\widehat{x}_{\mathrm{FS}} \in \underset{x \in \mathbb{R}^K}{\mathrm{Argmin}}\ h_1(L_1 F^\top x) + f_2(x). \tag{5.17}$$

On complète les hypothèses formulées dans l'introduction, en supposant que la fonction h_1 est différentiable et de gradient β_1-Lipschitz continu avec $\beta_1 > 0$. L'algorithme explicite-implicite, rappelé dans le paragraphe 3.3.1 (chapitre 3), peut être utilisé pour résoudre (5.16) et (5.17). Dans les paragraphes suivants, nous en rappelons les itérations, nous détaillons les problèmes numériques rencontrés et les solutions proposées.

5.3.1 Algorithme et difficultés rencontrées

Les initialisations et les itérations de l'algorithme explicite-implicite qui conduisent aux solutions de (5.1) et (5.2) sont décrites par les algorithmes 9 et 10. Les étapes ② de chaque algorithme correspondent à des descentes de gradient. Les opérateurs proximaux de $f_2 \circ F$ et f_2 sont calculés aux étapes ③ de chaque algorithme. Enfin, les étapes ④ correspondent aux mises à jour des suites $(y_n)_{n\in\mathbb{N}}$ et $(x_n)_{n\in\mathbb{N}}$. La convergence de la suite $(y_n)_{n\in\mathbb{N}}$ (resp. $(x_n)_{n\in\mathbb{N}}$) vers une solution du problème (5.16) (resp. du problème (5.17)) est assurée quand $\sum_{n=0}^{+\infty} \|a_n\| < +\infty$ [Combettes, Wajs, 2005]. La suite $(a_n)_{n\in\mathbb{N}}$ correspond à la tolérance d'une erreur numérique dans l'évaluation de l'opérateur proximal.

Initialisation
Choisir $y_0 \in \mathbb{R}^N$.

Pour $n \in \mathbb{N}$
① Soit $\gamma_n \in]0, 2/(\beta_1 \|L_1\|^2)[$ et $\lambda_n \in]0, 1]$.
② $y_{n+\frac{1}{2}} = y_n - \gamma_n \nabla(h_1 \circ L_1)(y_n)$
③ $p_n = \mathrm{prox}_{\gamma_n f_2 \circ F}(y_{n+\frac{1}{2}}) + a_n$
④ $y_{n+1} = y_n + \lambda_n \big(p_n - y_n\big)$

$\widehat{y}_{\mathrm{FA}} = \lim\limits_{n \to +\infty} y_n$.

Algorithme 9: Algorithme explicite-implicite pour résoudre (5.16)

Initialisation
Choisir $x_0 \in \mathbb{R}^K$.

Pour $n \in \mathbb{N}$
① Soit $\gamma_n \in]0, 2/(\beta_1 \left\|L_1 F^\top\right\|^2)[$ et $\lambda_n \in]0, 1]$
② $x_{n+\frac{1}{2}} = x_n - \gamma_n \nabla(h_1 \circ L_1 \circ F^\top)(x_n)$
③ $p_n = \mathrm{prox}_{\gamma_n f_2}(x_{n+\frac{1}{2}}) + a_n$
④ $x_{n+1} = x_n + \lambda_n \big(p_n - x_n\big)$

$\widehat{y}_{\mathrm{FS}} = \lim\limits_{n \to +\infty} F^\top x_n$.

Algorithme 10: Algorithme explicite-implicite pour résoudre (5.17)

Pour mettre en oeuvre ces algorithmes, deux principaux problèmes vont se poser. D'une part, comme nous l'avons mentionné dans l'introduction, le calcul de l'opérateur proximal associé à $f_2 \circ F$ dans l'algorithme 9 ne possède en général pas de forme explicite. D'autre part, le choix du pas γ_n des algorithmes dépend des valeurs de $\|L_1\|$ et $\left\|L_1 F^\top\right\|^2$. En effet, pour obtenir de bons taux de convergence, il est souvent judicieux de choisir le pas de l'algorithme au plus proche de sa borne supérieure, d'où l'importance du calcul de ces normes.

5.3.2 Calcul de l'opérateur proximal associé à $f_2 \circ F$

Pour des opérateurs d'analyse de trame F spécifiques, par exemple lorsque $FF^\top = \widetilde{\nu}\,\mathrm{Id}$ (cf. proposition 3.12 du chapitre 2), l'opérateur proximal peut être calculé explicitement. Cette remarque est également valable si $FF^\top = D$ où D désigne une matrice diagonale, et f_2 est une fonction séparable (cf. proposition 4.3 du chapitre 4). En pratique, dans le cas où F est l'opérateur d'analyse d'une base d'ondelettes orthonormales, $F^\top = F^{-1}$ la proposition 3.12 (chapitre 2) peut s'appliquer, néanmoins la majorité des trames ne vérifient pas une telle propriété. Comme nous l'avions mentionné dans l'introduction, le problème se pose même pour une trame ajustée. En effet, cette dernière assure que $F^\top F = \nu\,\mathrm{Id}$, mais pas que $FF^\top = \widetilde{\nu}\,\mathrm{Id}$.

Dans de récents travaux sur la dualité de Fenchel-Moreau-Rockafellar [Combettes *et al.*, 2009; Fadili, Peyré, 2010], les auteurs se sont placés dans un cadre théorique proche de notre problème. Pour mieux s'imprégner de cette notion de dualité et voir comment elle nous servira, nous effectuons ci-dessous un bref récapitulatif des définitions et propriétés qui nous serons utiles par la suite. Nous commençons par rappeler la définition de la conjuguée et de l'enveloppe de Moreau d'une fonction convexe, semi-continue inférieurement et propre.

Définition 5.4 [Rockafellar, 1974] *Soit $f \in \Gamma_0(\mathcal{H})$. La conjuguée de f est la fonction notée f^* appartenant à la classe $\Gamma_0(\mathcal{H})$ définie par*

$$(\forall u \in \mathcal{H}) \qquad f^*(u) = \sup_{v \in \mathcal{H}} \langle v|u\rangle - f(v).$$

Le théorème de Fenchel-Moreau établit que $f^{**} = f$.

Définition 5.5 [Rockafellar, 1974] *Soit $f \in \Gamma_0(\mathcal{H})$. L'enveloppe de Moreau de f est la fonction convexe continue, notée $\widetilde{f}$ et définie par*

$$\widetilde{f}\colon \mathcal{H} \to \mathbb{R}\colon u \mapsto \min_{v \in \mathcal{H}} \frac{1}{2}\|v-u\|^2 + f(v).$$

La dualité de Fenchel-Moreau-Rockafellar introduite dans [Fenchel, 1953; Moreau, 1967; Rockafellar, 1967] permet de relier les solutions de deux problèmes d'optimisation comme ceux présentés par lemme 5.6. Notons l'existence d'une autre dualité, connue en optimisation convexe et récemment utilisée pour la résolution de problèmes inverses par Weiss [Weiss, 2008] : il s'agit de la dualité min-max [Ekeland, Témam, 1999].

Lemme 5.6 [Zălinescu, 2002, Corollaire 2.8.5] *Soit $\varphi \in \Gamma_0(\mathcal{H})$, soit $\psi \in \Gamma_0(\mathcal{G})$ et soit L un opérateur linéaire borné de $\mathcal{H}$ dans $\mathcal{G}$ tel que $0 \in \operatorname{sri}(L(\operatorname{dom}\varphi) - \operatorname{dom}\psi)$. On a alors*

$$\underbrace{\inf_{u\in\mathcal{H}} \varphi(u) + \psi(Lu)}_{\textit{Problème primal}} = \underbrace{-\min_{v\in\mathcal{G}} \varphi^*(-L^*v) + \psi^*(v)}_{\textit{Problème dual}} \tag{5.18}$$

Sous les conditions d'existence liées au lemme précédent, la solution de problème dual est liée à la solution du problème primal, l'une étant l'opposée de l'autre. Ce lemme permet aux auteurs de [Combettes *et al.*, 2009] de lier les deux problèmes d'optimisation qui suivent.

Problème 5.7 (Problème primal) *Soit $\mathcal{H}$ et $\mathcal{G}$ deux espaces de Hilbert réels, soient $w \in \mathcal{H}$, $r \in \mathcal{G}$, $f \in \Gamma_0(\mathcal{H})$, $g \in \Gamma_0(\mathcal{G})$ et soit $L : \mathcal{H} \to \mathcal{G}$ un opérateur linéaire borné non-vide tel que la condition de qualification $r \in \operatorname{sri}\big(L(\operatorname{dom} f) - \operatorname{dom} g\big)$ soit vérifiée. Le problème consiste à trouver*

$$\widehat{u} \in \operatorname*{Argmin}_{u\in\mathcal{H}} f(u) + g(Lu - r) + \frac{1}{2}\|u - w\|^2.$$

Problème 5.8 (Problème dual) *Sous les mêmes hypothèses que le problème 5.7, le problème est de trouver*

$$\widehat{v} \in \operatorname*{Argmin}_{v\in\mathcal{H}} \widetilde{f}^*(w - L^*v) + g^*(v) + \langle v|r\rangle.$$

En utilisant le lemme 5.6 il est possible de relier la solution du problème primal 5.7 à celle du problème dual 5.8.

Proposition 5.9 [Combettes *et al.*, 2009, Proposition 3.4] *Soit $\widehat{v}$ une solution du problème 5.8 et soit*

$$\widehat{u} = \operatorname{prox}_f(w - L^*\widehat{v}).$$

$\widehat{u}$ est alors la solution du problème 5.7.

Cette proposition est particulièrement intéressante lorsque le problème primal est plus complexe à résoudre que le problème dual. Il est alors possible d'obtenir une forme explicite de la solution primale à partir de la solution duale. Ces deux solutions sont liées par la proposition 5.9.

Voyons comment appliquer ce formalisme à notre problème, c'est à dire le calcul de $\mathrm{prox}_{f_2 \circ F}$. La définition de l'opérateur proximal nous conduit au problème primal suivant :

$$\mathrm{prox}_{f_2 \circ F}(y) = \arg\min_{u \in \mathbb{R}^N} \frac{1}{2}\|y - u\|^2 + f_2(Fu). \tag{5.19}$$

L'équation (5.19) est un cas particulier du problème primal 5.7, avec $\mathcal{H} = \mathbb{R}^N$, $\mathcal{G} = \mathbb{R}^K$, $f = 0$, $g = f_2$, $L = F$, $w = y$ et $r = 0$. Le problème dual associé s'écrit donc,

$$\begin{aligned}\min_{v \in \mathbb{R}^K} f_2^*(v) + \widetilde{f}^*(y - F^\top v) &\Leftrightarrow \min_{v \in \mathbb{R}^K} \Big\{ f_2^*(v) + \min_{u \in \mathbb{R}^N} \big\{ \frac{1}{2}\|y - F^\top v - u\|^2 + f^*(u) \big\} \Big\} \\ &\Leftrightarrow \min_{v \in \mathbb{R}^K} f_2^*(v) + \frac{1}{2}\|y - F^\top v\|^2. \end{aligned} \tag{5.20}$$

Cette formulation contient un terme de gradient Lipschitz et l'opérateur proximal de f_2^* prend une forme explicite dès qu'il en est de même pour l'opérateur proximal de f_2. On peut en effet utiliser la propriété reliant l'opérateur proximal d'une fonction à celui de son conjugué. Cette proposition est rappelée ci-dessous.

Proposition 5.10 [Moreau, 1965] *Soit $f \in \Gamma_0(\mathcal{H})$ et soit $\gamma \in\ ]0, +\infty[$. On a alors*

$$\frac{1}{\gamma}\big(\mathrm{Id} - \mathrm{prox}_{\gamma f}\big) = \mathrm{prox}_{f^*/\gamma}(\cdot/\gamma).$$

Il est donc possible d'appliquer l'algorithme explicite-implicite pour trouver un minimiseur de (5.20), puis utiliser la proposition 5.9 pour connaître la valeur de $\mathrm{prox}_{f_2 \circ F}\, y$. L'algorithme résultant est présenté dans [Combettes *et al.*, 2009]. Nous en rappelons les itérations dans l'algorithme 11.

Initialisation
- Choisir $\epsilon \in]0, \min\{1, 1/\overline{\nu}\}[$.
- Soit $v_0 \in \mathbb{R}^K$.

Pour $n \in \mathbb{N}$
- Soit $\gamma_n \in [\epsilon, 2/\overline{\nu} - \epsilon]$ et soit $\lambda_n \in [\epsilon, 1]$.
- $u_n = y - F^\top v_n$
- $v_{n+1} = v_n + \lambda_n \gamma_n \big(F u_n - \mathrm{prox}_{\frac{1}{\gamma_n} f_2}(\frac{v_n}{\gamma_n} + F u_n)\big)$

$\mathrm{prox}_{f_2 \circ F}\, y = \lim_{n \to +\infty} u_n.$

Algorithme 11: Algorithme décomposé Dual-Primal pour le calcul de $\mathrm{prox}_{f_2 \circ F}\, y$

La suite $(v_n)_{n\in\mathbb{N}}$ générée par l'algorithme 11 converge vers une solution $\widehat{v}$ du problème dual (5.20) et $\operatorname{prox}_{f_2\circ F}(y) = y - F^\top\widehat{v}$. Rappelons que $\overline{\nu}$ désigne la constante de trame supérieure (cf. équation (2.16) dans le chapitre 2).

5.3.3 Calcul de la constante de trame

La deuxième difficulté, liée aux algorithmes explicite-implicite 9 et 10, consiste à calculer $\|L_1\|^2$ et $\|L_1F^\top\|^2$. En pratique, il est souvent possible de déterminer la valeur théorique de $\|L_1\|^2$ mais le calcul de $\|L_1F^\top\|^2$ est généralement plus complexe. Nous proposons dans ce paragraphe une méthode itérative permettant d'approcher la valeur de cette norme. Dans un premier temps, détaillons le raisonnement qui nous conduit au choix de la borne supérieure du pas γ_n de l'algorithme 10. Pour cela nous considèrons la fonction objectif (5.17) et posons $f_1 = h_1 \circ L_1 \circ F^\top \in \Gamma_0(\mathbb{R}^K)$. La fonction h_1 est de gradient β_1-Lipschitz avec $\beta_1 > 0$, nous pouvons alors écrire :

$$\big(\forall (u_1,u_2) \in (\mathbb{R}^M)^2\big) \qquad \|\nabla h_1(u_1) - \nabla h_1(u_2)\| \leq \beta_1\|u_1 - u_2\|. \tag{5.21}$$

Le gradient de f_1 est défini par

$$(\forall x \in \mathbb{R}^K) \qquad \nabla f_1(x) = \nabla(h_1 \circ L_1 \circ F^\top)(x) = FL_1^\top(\nabla h_1(L_1F^\top x)), \tag{5.22}$$

ce qui conduit à

$$\big(\forall (x_1,x_2) \in (\mathbb{R}^K)^2\big) \qquad \|\nabla f_1(x_1) - \nabla f_1(x_2)\| \leq \beta_1\|L_1F^\top\|^2\|x_1 - x_2\|. \tag{5.23}$$

On peut en déduire que f_1 est une fonction de gradient Lipschitz de constante $\beta_1\|L_1F^\top\|^2$. Ce résultat combiné au choix théorique du pas de l'algorithme explicite, présenté par l'hypothèse 3.19(i) (chapitre 3), nous conduit à $\gamma_n \in]0, 2/(\beta_1\|L_1F^\top\|^2)[$ pour l'algorithme 10.

Dans la plupart des cas, la norme $\|L_1F^\top\|^2$ ne possède pas une valeur connue *a priori*. Comme le taux de convergence dépend fortement de la borne supérieure du pas γ_n et donc de cette norme, il est important de l'approximer au plus près. Pour obtenir cette borne, nous proposons d'utiliser l'algorithme 12 qui converge vers $\|L_1F^\top\|^2$.

Initialisation
- Choisir aléatoirement $x_0 \in \mathbb{R}^K$.
- Soient $\rho_0 = 1 + 10^{-6}$, $\rho_1 = 1$, $B = L_1 F^\top$ et $n = 1$.

Tant que $\frac{|\rho_n - \rho_{n-1}|}{\rho_n} \geq \epsilon$
- $x_n = B^\top B x_{n-1}$
- $\rho_n = \frac{\|x_n\|}{\|x_{n-1}\|}$

$\|L_1 F^\top\|^2 = \lim_{n \to +\infty} \rho_n$.

Algorithme 12: Calcul de la constante de trame

Le raisonnement qui conduit à l'algorithme 12 est le suivant. On a posé $B = L_1 F^\top$. On effectue la décomposition en valeurs propres de la matrice associée à l'opérateur défini semi-positif $B^\top B$, *i.e.*,

$$B^\top B = U \Lambda U^\top$$

où $\Lambda = \mathrm{Diag}\{\lambda_1, \ldots, \lambda_K\}$ et $U = [u_1, \ldots, u_K] \in \mathbb{R}^{K \times K}$ est une matrice orthogonale. Par définition de la norme, $\|B\|^2 = \|L_1 F^\top\|^2 = \lambda_{i_0}$ où $i_0 \in \arg\max_{1 \leq i \leq K} \lambda_i$. Si x_0 désigne un élément de $\mathbb{R}^K$ qui n'appartient pas à l'espace propre de $B^\top B$, nous pouvons écrire,

$$\frac{\|B^n x_0\|^2}{\|B^{n-1} x_0\|^2} = \frac{\sum_{i=1}^K \lambda_i^n |\langle x_0, u_i \rangle|^2}{\sum_{i=1}^K \lambda_i^{n-1} |\langle x_0, u_i \rangle|^2}, \tag{5.24}$$

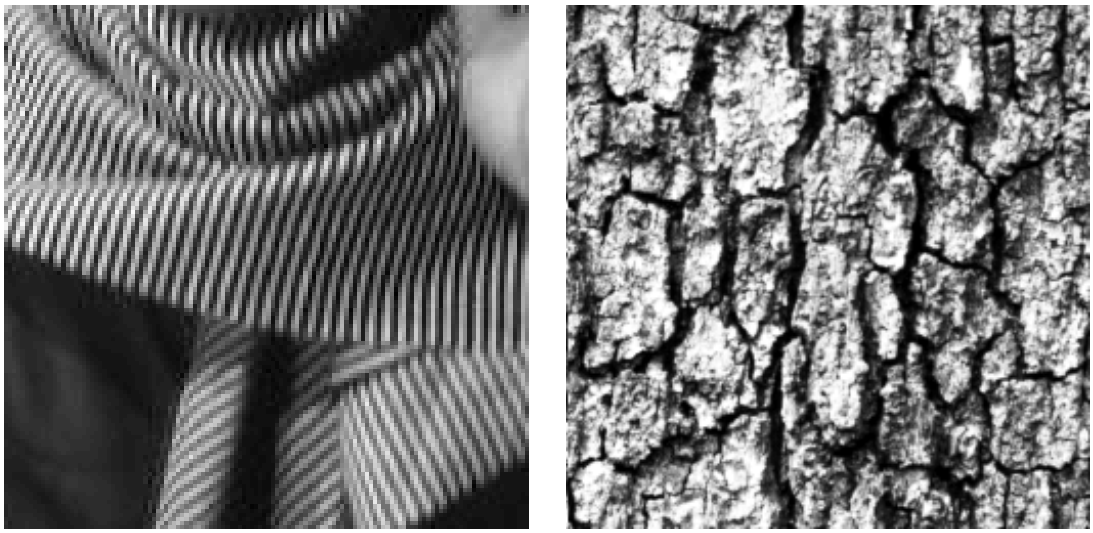
ce qui conduit à

$$\lim_{n \to +\infty} \frac{\|B^n x_0\|^2}{\|B^{n-1} x_0\|^2} = \lambda_{i_0} = \|L_1 F^\top\|^2. \tag{5.25}$$

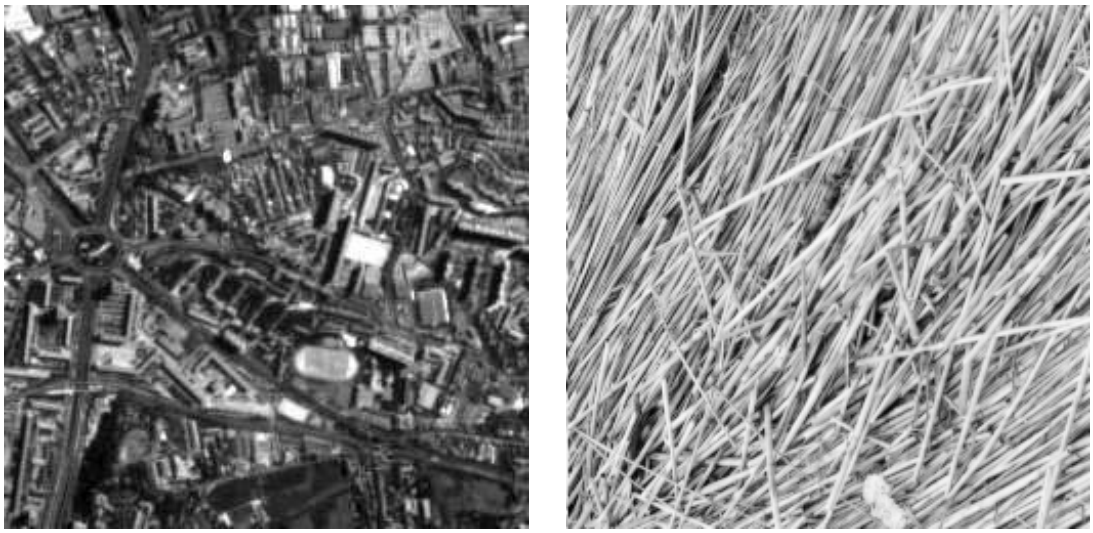
5.3.4 Comparaison de FA et FS sur un exemple de restauration en présence de bruit gaussien

Dans les paragraphes précédents, nous avons présenté des techniques permettant de résoudre numériquement FA et FS à l'aide de l'algorithme explicite-implicite. Pour illustrer et comparer le comportement des deux approches, nous allons considérer des images dégradées par un flou uniforme de taille $Q = 3 \times 3$ ou $Q = 5 \times 5$ (ainsi $A = L_1 \in \mathbb{R}^{N \times N}$ correspond à la matrice de dégradation associée au filtrage uniforme) et par un bruit additif gaussien de variance $\sigma^2 = 25$ ou $\sigma^2 = 100$. Les tests sont conduits sur quatre images standards : deux images de textures (source : http ://sipi.usc.edu/database), une version tronquée de l'image « Barbara » (source : http ://sipi.usc.edu/database) et une image satellitaire SPOT5 (source : LIGM). Les images sont toutes de taille $N = 256 \times 256$

sauf l'image « Barbara » pour laquelle $N = 128 \times 128$. Ces images sont présentées figure 5.1.



(a) Image « Barbara » (b) Image « Bark »



(c) Image « Spot » (d) Image « Straw »

FIGURE 5.1 – Images originales pour la comparaison FA versus FS.

Les critères FA et FS que nous considérons sont les suivants :
– FA :

$$\widehat{y}_{\mathrm{FA}} \in \underset{y \in \mathbb{R}^N}{\mathrm{Argmin}} \frac{1}{2\sigma^2}\|Ay - z\|^2 + \sum_{k=1}^{K} \chi_k |(Fy)_k|, \tag{5.26}$$

avec, pour tout $k \in \{1, \ldots, K\}$, $\chi_k > 0$.

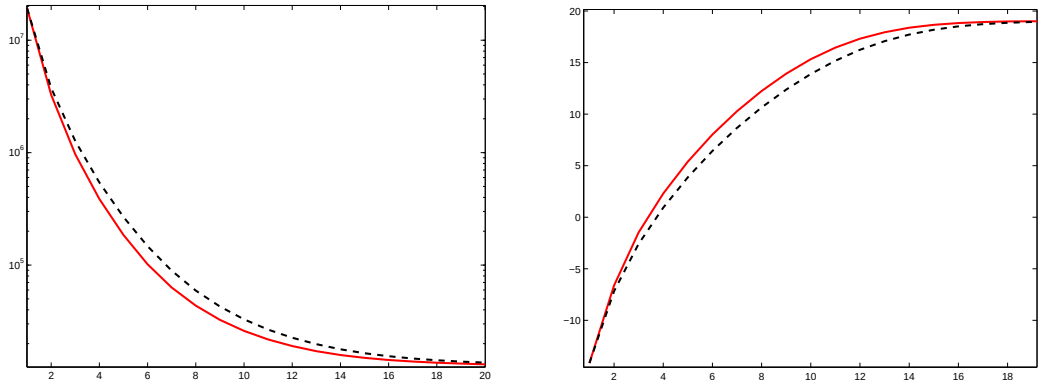
– FS :

$$\widehat{y}_{\mathrm{FS}} = F^\top \widehat{x}_{\mathrm{FS}} \quad \text{avec} \quad \widehat{x}_{\mathrm{FS}} \in \underset{x \in \mathbb{R}^K}{\mathrm{Argmin}} \frac{1}{2\sigma^2} \|AF^\top x - z\|^2 + \sum_{k=1}^{K} \widetilde{\chi}_k |x_k|, \tag{5.27}$$

avec, pour tout $k \in \{1, \ldots, K\}$, $\widetilde{\chi}_k > 0$.

Le terme d'attache aux données est un terme quadratique (antilog-vraisemblance d'une loi normale). Comme *a priori*, nous avons choisi une norme ℓ_1 pour favoriser la parcimonie. Pour chaque formulation, les paramètres χ_k (resp. $\widetilde{\chi}_k$) ont une valeur identique pour l'ensemble des sous-bandes de détails mais prennent une valeur différente (proche de 0) pour les coefficients de la sous-bande d'approximation. Ces paramètres sont choisis manuellement de façon à maximiser le rapport-signal-sur-bruit entre l'image restaurée et l'image originale. Dans nos expériences le comportement de deux trames non-ajustées est évalué : des tranformées GenLOTs [De Queiroz *et al.*, 1996; Gauthier *et al.*, 2009] et des transformées en arbre dual (DTT) [Selesnick *et al.*, 2005; Chaux *et al.*, 2006].

La figure 5.2 illustre l'impact du calcul exact de la norme $\|AF^\top\|$ sur la vitesse de convergence de l'algorithme explicite-implicite dans sa version FS lorsque des GenLOTs sont considérés. Ce calcul permet en effet d'améliorer la vitesse de convergence. Il faut également remarquer que cet algorithme itératif peut être utilisé, plus généralement, pour calculer la norme de l'opérateur de trame quand sa valeur n'est pas explicite.



Évolution du critère (échelle logarithmique) Évolution du SNR

FIGURE 5.2 – Comparaison des profils de convergence lorsque $\|L_1 F^\top\|^2$ est approximé (ligne discontinue) ou non (ligne continue) en fonction des itérations.

Les tableaux 5.1 et 5.2 présentent les valeurs de SNR et SSIM obtenues lorsque la restauration des images est effectuée par FA ou FS. Nous pouvons remarquer que les résultats obtenus tant en SNR qu'en SSIM sont très proches. Il est difficile de mettre en

avant une approche par rapport à l'autre car suivant la trame, le type d'image, le niveau de dégradation et la mesure d'évaluation considérés, les résultats peuvent favoriser soit FA soit FS. Les figures 5.3 et 5.4 présentent deux exemples de restauration avec FA et FS en considérant des DTT et des GenLOTs. On note clairement que la différence visuelle entre les images restaurées n'est pas significative. Nous pouvons pourtant noter que la figure 5.3 présente les images résultant de l'exemple avec la plus grande différence de SNR et SSIM entre FA et FS.

			SNR			SSIM		
			Initial	FA	FS	Initial	FA	FS
$\sigma = 5$	$Q = 3 \times 3$	Barbara	11.9	**16.0**	**16.0**	0.58	**0.81**	**0.81**
		Bark	16.1	**19.1**	18.8	0.82	**0.91**	**0.91**
		Spot5	17.0	**19.8**	19.5	0.83	**0.90**	**0.90**
		Straw	18.8	**20.8**	**20.8**	0.73	0.84	**0.85**
	$Q = 5 \times 5$	Barbara	8.75	**13.9**	**13.9**	0.16	**0.72**	**0.72**
		Bark	12.4	**15.5**	**15.5**	0.57	**0.80**	**0.80**
		Spot5	12.9	**16.2**	15.8	0.57	**0.78**	**0.78**
		Straw	15.8	**17.8**	17.5	0.42	**0.65**	**0.65**
$\sigma = 10$	$Q = 3 \times 3$	Barbara	11.9	**13.4**	13.3	0.51	**0.68**	**0.68**
		Bark	15.5	**17.4**	17.1	0.80	**0.87**	**0.87**
		Spot5	15.7	**17.9**	17.5	0.76	**0.86**	0.85
		Straw	17.9	**19.4**	19.3	0.68	0.77	**0.78**
	$Q = 5 \times 5$	Barbara	8.56	**12.0**	11.8	0.11	**0.57**	**0.57**
		Bark	12.1	**14.4**	14.1	0.55	**0.73**	**0.73**
		Spot5	12.3	**14.7**	14.3	0.52	**0.70**	**0.70**
		Straw	15.3	**16.6**	**16.6**	0.39	0.55	**0.56**

TABLE 5.1 – Evaluation du SNR et du SSIM pour des images dégradées par un flou et un bruit gaussien puis restaurées en utilisant des DTT.

			SNR			SSIM		
			Initial	FA	FS	Initial	FA	FS
$\sigma = 5$	$Q = 3 \times 3$	Barbara	11.9	17.5	**17.6**	0.58	**0.86**	0.85
		Bark	16.1	**19.2**	**19.2**	0.82	**0.92**	**0.92**
		Spot5	17.0	**20.1**	20.0	0.83	**0.91**	**0.91**
		Straw	18.8	**21.3**	21.2	0.73	**0.86**	**0.86**
	$Q = 5 \times 5$	Barbara	8.75	**16.0**	**16.0**	0.16	**0.80**	**0.80**
		Bark	12.4	**15.8**	**15.8**	0.57	**0.81**	**0.81**
		Spot5	12.9	**16.2**	16.0	0.57	**0.79**	0.78
		Straw	15.8	**17.9**	17.8	0.42	**0.68**	0.67
$\sigma = 10$	$Q = 3 \times 3$	Barbara	11.9	**15.2**	15.1	0.51	0.77	0.76
		Bark	15.5	**17.6**	**17.6**	0.80	**0.88**	**0.88**
		Spot5	15.7	**18.1**	17.9	0.76	**0.86**	**0.86**
		Straw	17.9	**19.6**	**19.6**	0.68	**0.79**	**0.79**
	$Q = 5 \times 5$	Barbara	8.56	**14.1**	13.9	8.56	**0.71**	0.70
		Bark	12.1	**14.5**	14.4	12.1	**0.75**	**0.75**
		Spot5	12.3	**14.8**	14.7	12.3	**0.71**	**0.71**
		Straw	15.3	16.8	**16.9**	15.3	**0.57**	**0.57**

TABLE 5.2 – Evaluation du SNR et du SSIM pour des images dégradées par un flou et un bruit gaussien puis restaurées en utilisant des tranformées GenLOTs.

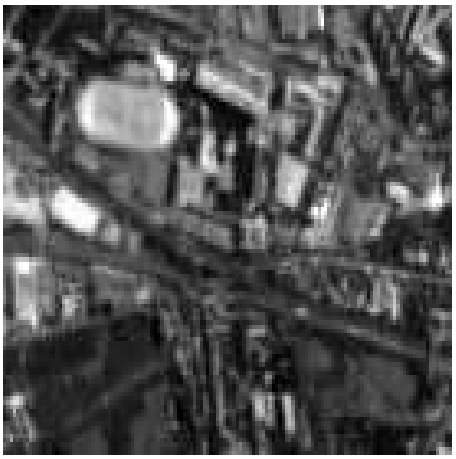


Image originale

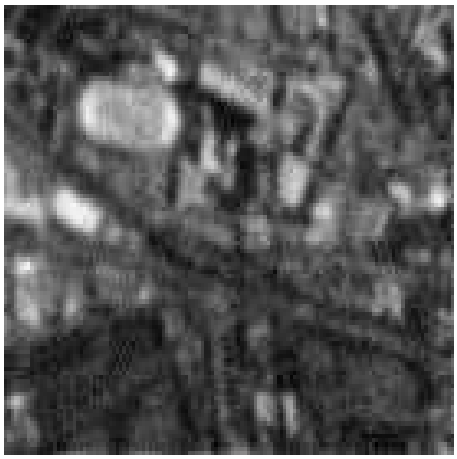


Image dégradée avec $\sigma = 10$ et $Q = 3 \times 3$
$\mathrm{SNR} = 15.7\ \mathrm{dB}$

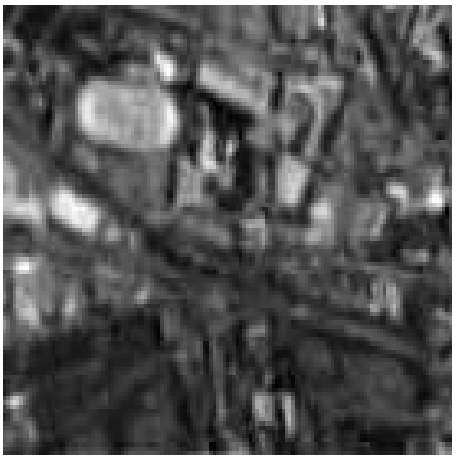


Image restaurée avec FA et DTT
$\mathrm{SNR} = 17.9\ \mathrm{dB}$

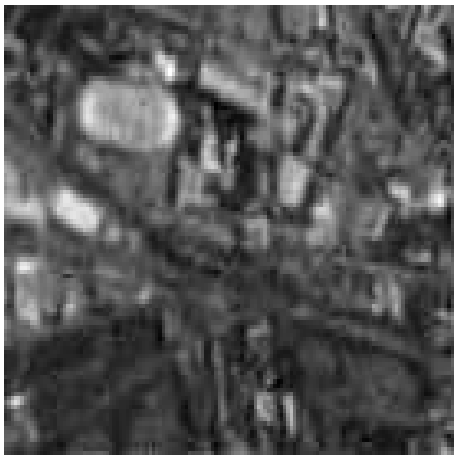


Image restaurée avec FS et DTT
$\mathrm{SNR} = 17.5\ \mathrm{dB}$

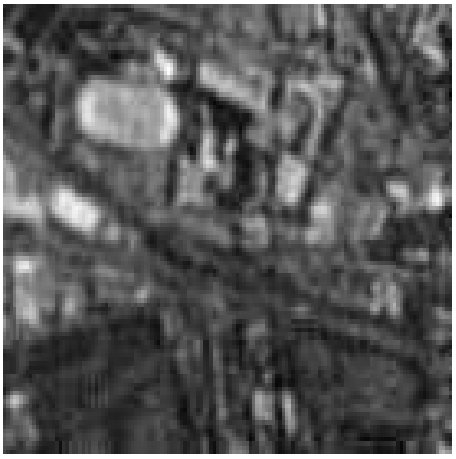


Image restaurée avec FA et GenLOTs
$\mathrm{SNR} = 18.1\ \mathrm{dB}$

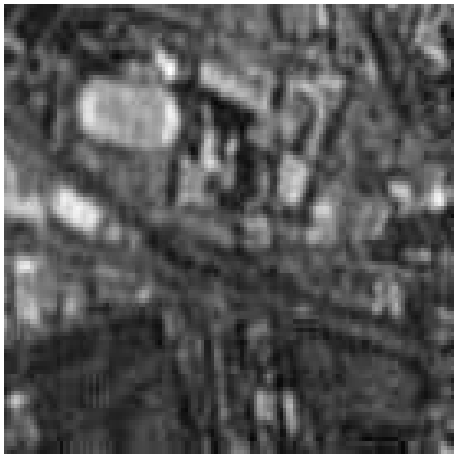


Image restaurée avec FS et GenLOTs
$\mathrm{SNR} = 17.9\ \mathrm{dB}$

FIGURE 5.3 – Restauration de l'image satellitaire avec les approches à l'analyse et à la synthèse en utilisant des DTT et des GenLOTs.

Image originale

Image dégradée avec $\sigma = 5$ et $Q = 3 \times 3$
$\mathrm{SNR} = 15.7\ \mathrm{dB}$

Image restaurée avec FA et DTT
$\mathrm{SNR} = 16.0\ \mathrm{dB}$

Image restaurée avec FS et DTT
$\mathrm{SNR} = 16.0\ \mathrm{dB}$

Image restaurée avec FA et GenLOTs
$\mathrm{SNR} = 17.5\ \mathrm{dB}$

Image restaurée avec FS et GenLOTs
$\mathrm{SNR} = 17.5\ \mathrm{dB}$

FIGURE 5.4 – Restauration de l'image « Barbara » avec les approches à l'analyse et à la synthèse en utilisant des DTT et des GenLOTs.

5.4 PPXA+ pour une classe de trames non ajustées

Dans la section précédente, nous avons proposé une solution pour résoudre FA et FS lorsque le critère impliqué est composé de deux fonctions, dont l'une est de gradient Lipschitz. Pour résoudre les formulations plus générales (5.1) et (5.2), deux solutions sont envisageables. La première permet de minimiser un critère composé d'une somme finie de fonctions. Il s'agit de l'algorithme proximal parallèle (PPXA) [Combettes, Pesquet, 2008], présenté dans le paragraphe 4.4 (chapitre 4). L'une des principales difficulté pour résoudre (5.1) et (5.2) par le biais de cet algorithme, est le calcul de l'opérateur proximal impliquant une fonction et un opérateur linéaire. Ce calcul peut être effectué :

- en utilisant la proposition 4.3 (chapitre 4) sur le calcul de l'opérateur proximal d'une fonction composée avec un opérateur linéaire,
- en décomposant la fonctionnelle en une somme de I fonctions, comme nous l'avons proposé dans la section 4.3.2 (chapitre 4),
- en passant par l'approche duale utilisée dans le paragraphe 5.3.2 de ce présent chapitre.

La seconde solution passe par l'utilisation d'algorithmes de type Lagrangien Augmenté (LA), aussi nommés algorithmes décomposés de Bregman [Afonso *et al.*, 2009; Setzer *et al.*, 2010]. Il est important de remarquer que dans les travaux existants, ces deux solutions sont exclusivement utilisées lorsque les critères font intervenir des trames ajustées.

Dans [Pesquet, 2010], il est montré que PPXA et LA peuvent être rassemblés dans un cadre unifié qui est fondé sur les opérateurs proximaux. L'algorithme résultant est nommé PPXA+. Nous proposons d'étendre l'utilisation de PPXA+ au cas d'une classe de trames non ajustées que nous présentons ci-après.

5.4.1 Un exemple particulier de trame non-ajustées

La notion de trame a été préalablement introduite dans le paragraphe 2.2.3-b) (chapitre 2). Nous présentons ici une sous-classe de trames non-ajustées, telle que,

$$F = U \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_{\mathsf{Q}} \end{bmatrix} \tag{5.28}$$

où, $U \in \mathbb{R}^{K \times \mathsf{Q}N}$ est la matrice associée à l'opérateur d'analyse d'une trame ajustée de constante $\nu_U \in \,]0, +\infty[$, et où, pour tout $q \in \{1, \ldots, \mathsf{Q}\}$, $V_q \in \mathbb{R}^{N \times N}$ désigne un opérateur de préfiltrage. Pour que la condition de trame (2.16), présentée dans le chapitre 2, soit vérifiée, les préfiltres $(V_q)_{1 \leq q \leq \mathsf{Q}}$ de réponses fréquentielles $(\check{v}_q(\omega))_{1 \leq q \leq \mathsf{Q}}$ doivent satisfaire la relation :

$$(\forall \omega \in [-\pi, \pi]^2) \qquad \nu_U \sum_{q=1}^{\mathsf{Q}} |\check{v}_q(\omega)|^2 \geq \nu > 0. \tag{5.29}$$

Il est alors facile de montrer que

$$F^\top F = \nu_U \sum_{q=1}^{\mathsf{Q}} V_q^\top V_q. \tag{5.30}$$

L'exemple le plus simple d'une telle trame est un banc de filtre non-décimé sur Q canaux avec une redondance d'un facteur Q. Afin d'obtenir des représentations à faible redondance, les trames telles que les transformées 2D réelles (resp. complexes) en arbre dual M-bandes ont été proposées dans [Kingsbury, 2001; Chaux *et al.*, 2006]. Les DTT consistent à effectuer deux (resp. quatre) décompositions M-bandes en parallèle (matrices orthogonales U_q). Chacune des décompositions est précédée d'une étape de préfiltrage (matrices V_q) liée à la discrétisation. Finalement, une combinaison orthogonale (matrice orthogonale R) des sous-bandes est appliquée pour assurer la directionalité. Un tel type de trame satisfait (5.28) et (5.29), où

$$U = R \begin{bmatrix} U_1 & 0 & \dots & 0 \\ 0 & U_2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & \dots & U_{\mathsf{Q}} \end{bmatrix}. \tag{5.31}$$

Notons que les DTT *discrètes* contrairement aux DTT continues ne constituent, en général, pas des trames ajustées.

5.4.2 Résolution des deux formulations avec PPXA+

Les initialisations et les itérations, résultantes de l'algorithme PPXA+ [Pesquet, 2010] pour trouver les solutions de (5.1) et (5.2), sont données par les algorithmes 13 et 14. La convergence de la suite $(y_\ell)_{\ell\in\mathbb{N}}$ (resp. $(x_\ell)_{\ell\in\mathbb{N}}$) générée par l'algorithme 13 (resp. algorithme 14) est établie sous les hypothèses suivantes :

Hypothèse 5.11

(i) $\left(\bigcap_{i=1}^{S} \operatorname{ri}\operatorname{dom} h_i \circ L_i F^\top\right) \cap \left(\bigcap_{j=S+1}^{J} \operatorname{ri}\operatorname{dom} f_j\right) \neq \varnothing$

(resp. $\left(\bigcap_{i=1}^{S} \operatorname{ri}\operatorname{dom} h_i \circ L_i\right) \cap \left(\bigcap_{j=S+1}^{J} \operatorname{ri}\operatorname{dom} f_j \circ F\right) \neq \varnothing$*).*

(ii) *Il existe* $\underline{\lambda} \in]0,2[$ *tel que* $(\forall \ell \in \mathbb{N}),\ \underline{\lambda} \leq \lambda_{\ell+1} \leq \lambda_\ell$.

(iii) *(*$\forall i \in \{1,\dots,S\}$*),* $\sum_{\ell\in\mathbb{N}} \|a_{i,\ell}\| < +\infty$ *et*

($\forall j \in \{S+1,\dots,J\}$*),* $\sum_{\ell\in\mathbb{N}} \|b_{j,\ell}\| < +\infty$.

Initialisation

$(\eta_i)_{1\leq i\leq S} \in \,]0,+\infty[^S$, $(\kappa_j)_{S+1\leq j\leq J} \in \,]0,+\infty[^{(J-S)}$

$(v_{i,0})_{1\leq i\leq S} \in (\mathbb{R}^N)^S, (w_{j,0})_{S+1\leq j\leq J} \in (\mathbb{R}^K)^{(J-S)}$

$y_0 = \arg\min_{u\in\mathbb{R}^N} \sum_{i=1}^S \eta_i \|L_i u - v_{i,0}\|^2 + \sum_{j=S+1}^J \kappa_j \|Fu - w_{j,0}\|^2$

Pour $\ell = 0, 1, \ldots$

Pour $i = 1, \ldots, S$

$p_{i,\ell} = \mathrm{prox}_{h_i/\eta_i} v_{i,\ell} + a_{i,\ell}$

Pour $j = S+1, \ldots, J$

$r_{j,\ell} = \mathrm{prox}_{f_j/\kappa_j} w_{j,\ell} + b_{j,\ell}$

$\lambda_\ell \in]0, 2[$

$c_\ell = \arg\min_{u\in\mathbb{R}^N} \sum_{i=1}^S \eta_i \|L_i u - p_{i,\ell}\|^2 + \sum_{j=S+1}^J \kappa_j \|Fu - r_{j,\ell}\|^2$

Pour $i = 1, \ldots, S$

$v_{i,\ell+1} = v_{i,\ell} + \lambda_\ell \big(L_i(2c_\ell - y_\ell) - p_{i,\ell}\big)$

Pour $j = S+1, \ldots, J$

$w_{j,\ell+1} = w_{j,\ell} + \lambda_\ell \big(F(2c_\ell - y_\ell) - r_{j,\ell}\big)$

$y_{\ell+1} = y_\ell + \lambda_\ell (c_\ell - y_\ell)$

$\widehat{y}_{\mathrm{FA}} = \lim_{\ell\to+\infty} y_\ell.$

Algorithme 13: Algorithme PPXA+ pour trouver une solution de (5.1).

Initialisation

$(\eta_i)_{1\leq i\leq S} \in \,]0,+\infty[^S$, $(\kappa_j)_{S+1\leq j\leq J} \in \,]0,+\infty[^{(J-S)}$

$(v_{i,0})_{1\leq i\leq S} \in (\mathbb{R}^N)^S$, $(w_{j,0})_{S+1\leq j\leq J} \in (\mathbb{R}^K)^{(J-S)}$

$x_0 = \arg\min_{u\in\mathbb{R}^K} \sum_{i=1}^S \eta_i \|L_i F^\top u - v_{i,0}\|^2 + \sum_{j=S+1}^J \kappa_j \|u - w_{j,0}\|^2$

Pour $\ell = 0, 1, \ldots$

Pour $i = 1, \ldots, S$

$p_{i,\ell} = \mathrm{prox}_{h_i/\eta_i} v_{i,\ell} + a_{i,\ell}$

Pour $j = S+1, \ldots, J$

$r_{j,\ell} = \mathrm{prox}_{f_j/\kappa_j} w_{j,\ell} + b_{j,\ell}$

$\lambda_\ell \in]0, 2[$

$c_\ell = \arg\min_{u\in\mathbb{R}^K} \sum_{i=1}^S \eta_i \|L_i F^\top u - p_{i,\ell}\|^2 + \sum_{j=S+1}^J \kappa_j \|u - r_{j,\ell}\|^2$

Pour $i = 1, \ldots, S$

$v_{i,\ell+1} = v_{i,\ell} + \lambda_\ell \big(L_i F^\top (2c_\ell - x_\ell) - p_{i,\ell}\big)$

Pour $j = S+1, \ldots, J$

$w_{j,\ell+1} = w_{j,\ell} + \lambda_\ell \big(2c_\ell - x_\ell - r_{j,\ell}\big)$

$x_{\ell+1} = x_\ell + \lambda_\ell (c_\ell - x_\ell)$

$\widehat{y}_{\mathrm{FS}} = \lim_{\ell\to+\infty} F^\top x_\ell.$

Algorithme 14: Algorithme PPXA+ pour trouver une solution de (5.2).

Ces algorithmes font intervenir les suites $(a_{i,\ell})_{1\leq i\leq S}$ et $(b_{j,\ell})_{S+1\leq j\leq J}$ qui correspondent à des erreurs numériques possibles lors du calcul des opérateurs proximaux à l'itération ℓ. Les différents paramètres à ajuster, comme $(\eta_i)_{1\leq i\leq S}$, $(\kappa_j)_{S+1\leq j\leq J}$ et $(\lambda_\ell)_{\ell\in\mathbb{N}}$, permettent une grande flexibilité des algorithmes.

Dans le calcul de la variable intermédiaire $(c_\ell)_{\ell\in\mathbb{N}}$ et les étapes d'initialisation de y_0 et x_0, les algorithmes 14 et 13 requièrent une inversion matricielle de grande dimension dûe à la minimisation d'une somme de fonctions quadratiques. Nous illustrons ce propos sur le calcul de x_0 présent dans l'algorithme 14.

$$x_0 = \arg\min_{u\in\mathbb{R}^K} \sum_{i=1}^{S} \eta_i \|L_i F^\top u - v_{i,0}\|^2 + \sum_{j=S+1}^{J} \kappa_j \|u - w_{j,0}\|^2 \tag{5.32}$$

$$= \Big(\sum_{i=1}^{S} \eta_i F L_i^\top L_i F^\top + \kappa \mathrm{Id}\Big)^{-1} \Big(\sum_{i=1}^{S} \eta_i F L_i^\top v_{i,0} + \sum_{j=S+1}^{J} \kappa_j w_{j,0}\Big). \tag{5.33}$$

où $\kappa = \sum_{j=1}^{(J-S)} \kappa_j$. Pour s'assurer que cette inversion ne soit pas trop coûteuse en calcul, malgré l'utilisation de trames non-ajustées, nous supposons que l'opérateur de trame F est défini par l'équation (5.28). Le lemme d'inversion matricielle nous permet de formuler les égalités qui suivent.

– Pour FA (algorithme 13)

$$\Big(\sum_{i=1}^{S} \eta_i L_i^\top L_i + \kappa F^\top F\Big)^{-1} = \Big(\sum_{i=1}^{S} \eta_i L_i^\top L_i + \kappa \nu_U \sum_{q=1}^{\mathsf{Q}} V_q^\top V_q\Big)^{-1} \tag{5.34}$$

– Pour FS (algorithme 14)

$$\begin{aligned}\kappa\Big(\sum_{i=1}^{S} \eta_i F L_i^\top L_i F^\top + \kappa \mathrm{Id}\Big)^{-1} \\ &= \mathrm{Id} - F\big(\sum_{i=1}^{S} \eta_i L_i^\top L_i\big)\Big(\kappa \mathrm{Id} + F^\top F\big(\sum_{i=1}^{S} \eta_i L_i^\top L_i\big)\Big)^{-1} F^\top, \\ &= \mathrm{Id} - F\big(\sum_{i=1}^{S} \eta_i L_i^\top L_i\big)\Big(\kappa \mathrm{Id} + \big(\nu_U \sum_{q=1}^{\mathsf{Q}} V_q^\top V_q\big)\big(\sum_{i=1}^{S} \eta_i L_i^\top L_i\big)\Big)^{-1} F^\top.\end{aligned} \tag{5.35}$$

Les dernières inversions matricielles dans (5.34) et (5.35) s'effectuent simplement en notant que $(L_i)_{1\leq i\leq S}$ et $(V_q)_{1\leq q\leq \mathsf{Q}}$ peuvent être diagonalisées de façon jointe dans le domaine de Fourier. En effet toute matrice B associée à un opérateur de convolution périodique peut être factorisé comme $B = D^*\Lambda D$ où D représente la transformée de Fourier discrète. D est une matrice unitaire, telle que $D^{-1} = D^*$ et Λ est une matrice diagonale. Par conséquent, si B est inversible, $B^{-1} = (D^*\Lambda D)^{-1} = D^*\Lambda^{-1}D$.

5.4.3 Résultats expérimentaux

Dans cette partie, nous appliquons l'algorithme PPXA+ pour restaurer des images dégradées par un bruit de Poisson avec un paramètre d'échelle $\alpha = 0.8$ et un flou uniforme de taille 3×3. Les critères FA et FS que nous minimisons sont :

$$\min_{y \in \mathbb{R}^N} h_1(Ay) + \iota_C(y) + f_3(Fy) \tag{5.36}$$

et

$$\min_{x \in \mathbb{R}^K} h_1(AF^\top x) + \iota_C(F^\top x) + f_3(x) \tag{5.37}$$

où F représente une DTT 2-bandes [Chaux *et al.*, 2006]. Ce choix de trame est un cas particulier de la sous-classe de trame définie par (5.28). Les DTT [Chaux *et al.*, 2006] sont calculées en utilisant des filtres de symmlets de longueur 6 sur 3 niveaux de résolution. Le problème (5.37) (resp. (5.36)) est un cas particulier du problème (5.2) (resp. (5.1)) où $J = 3$ et $S = 2$.

La fonction h_1 représente le terme d'attache aux données. Ici, elle correspond à la divergence de Kullback-Leibler généralisée qui est bien adaptée au bruit de Poisson.

La fonction $h_2 = \iota_C$ désigne la fonction indicatrice de l'ensemble convexe non vide $C = [0, 255]^N$ de $\mathbb{R}^N$. Cet ensemble modélise une contrainte convexe sur la dynamique des pixels de l'image.

La fonction f_3 correspond au terme de régularisation opérant dans le domaine transformé. Cette fonction est de la forme :

$$f_3 \colon (\eta_k)_{1 \leq k \leq K} \mapsto \sum_{k=1}^{K} \chi_k |\eta_k|. \tag{5.38}$$

Dans nos simulations, les paramètres $(\chi_k)_{1 \leq k \leq K} \in]0, +\infty[^K$ sont choisis proche de 0 pour les valeurs de k correspondant aux sous-bandes d'approximation. Pour les valeurs de k associées aux coefficients de détails, ces paramètres sont égaux et choisis de manière à maximiser le SNR. L'opérateur proximal de f_3 se réduit au seuillage doux présenté par l'exemple 3.3 (chapitre 3).

La matrice $L_1 = A \in \mathbb{R}^{N \times N}$ est associée à l'opérateur de convolution (périodique) 2D et $L_2 = \mathrm{Id}$. Dans notre formulation, les matrices L_1 et $(V_q)_{1 \leq q \leq \mathsf{Q}}$ (reliées à F) peuvent être diagonalisées par une transformée de Fourier discrète 2D, dans l'optique de résoudre efficacement les inversions matricielles dans (5.34) et (5.35).

La figure 5.5 montre une comparaison entre l'utilisation des DTT complexes et d'une version ajustée (où pour tout $q \in \{1, \ldots, 4\}$, $V_q = \mathrm{Id}$). Dans cet exemple de simulation, une version tronquée de l'image de « Barbara » ($N = 128 \times 128$) est considérée en adoptant un critère FS. L'utilisation de DTT non-ajustées incluant des préfiltres permet d'améliorer la qualité des résultats aussi bien visuellement qu'en terme de SNR et de SSIM. Dans cet exemple, nous présentons également les résultats obtenus avec l'algorithme EM-ML (cf. section 2.4.1 du chapitre 3) et ceux obtenus avec l'EM régularisé

où l'antilog-vraisemblance poissonienne est pénalisée par une divergence de Kullback-Leibler entre l'image désirée et une image de référence uniforme.

La figure 5.6 affiche un deuxième exemple de restauration, pour l'image tronquée de « Marseille » ($N = 128 \times 128$). Les DTT non-ajustées sont utilisées et les images obtenues par FA et FS sont présentées. La restauration présente de très légères différences en faveur de FA.

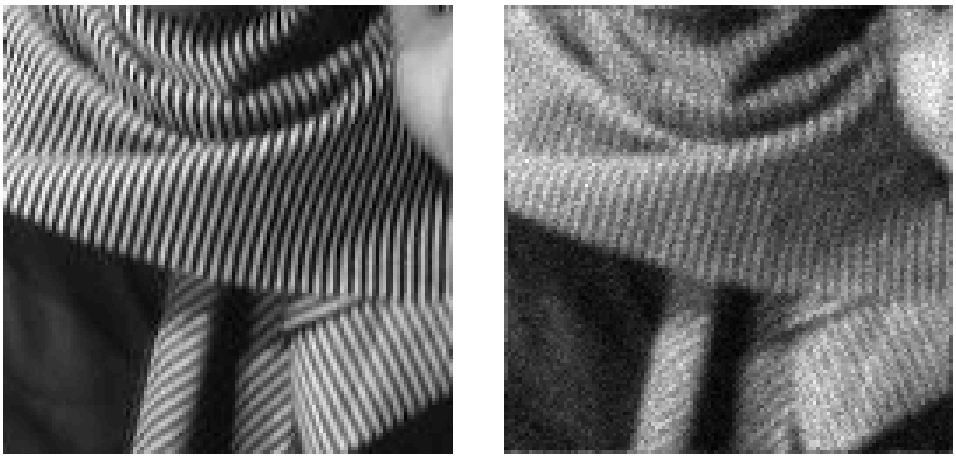
5.5 Conclusion

Cette étude nous permet d'avancer que les algorithmes proximaux peuvent aussi bien être utilisés pour résoudre la FS que la FA. Cependant, certains algorithmes permettront de traiter plus efficacement ces deux formulations que d'autres. En effet, l'algorithme explicite-implicite requiert l'utilisation d'un algorithme imbriqué pour FA ce qui pénalisera le taux de convergence global. Concernant l'algorithme PPXA+, les approches à l'-analyse et à la synthèse ont des complexités équivalentes.

Dans un soucis de simplifier l'implantation des algorithmes proximaux, les trames ajustées ont souvent été préférées aux trames non-ajustées [Combettes, Pesquet, 2007a; Afonso *et al.*, 2009; Combettes, Pesquet, 2008]. Cependant, nos résultats expérimentaux montrent que certaines trames non-ajustées permettent d'obtenir un gain dans la qualité de restauration. Il semble donc intéressant de proposer des méthodes permettant d'utiliser des versions non-ajustées.

Nous avons également constaté que le choix des paramètres de l'algorithme explicite-implicite, en particulier le « pas » de celui-ci pouvait influer sur le taux de convergence. Il paraît donc préférable, dès que cela est possible, de calculer précisément la constante associée à la fonction de gradient Lipschitz.

Les résultats obtenus par FA et FS sont généralement très proches tant visuellement que quantitativement. Nous avons également montré que FA et FS sont théoriquement équivalentes sous certaines conditions. Il semble donc, d'une part, plus judicieux de considérer la formulation la plus simple à résoudre pour un algorithme donné et, d'autre part, intéressant de combiner les régularisations à l'analyse et à la synthèse. Nous pouvons remarquer que la régularisation hybride ($\mathrm{tv}+\ell_1$), proposée dans la section 4.5.1 (chapitre 4), s'inscrit dans cette démarche.



originale

dégradée
SNR = 11.4 dB , SSIM = 0.53

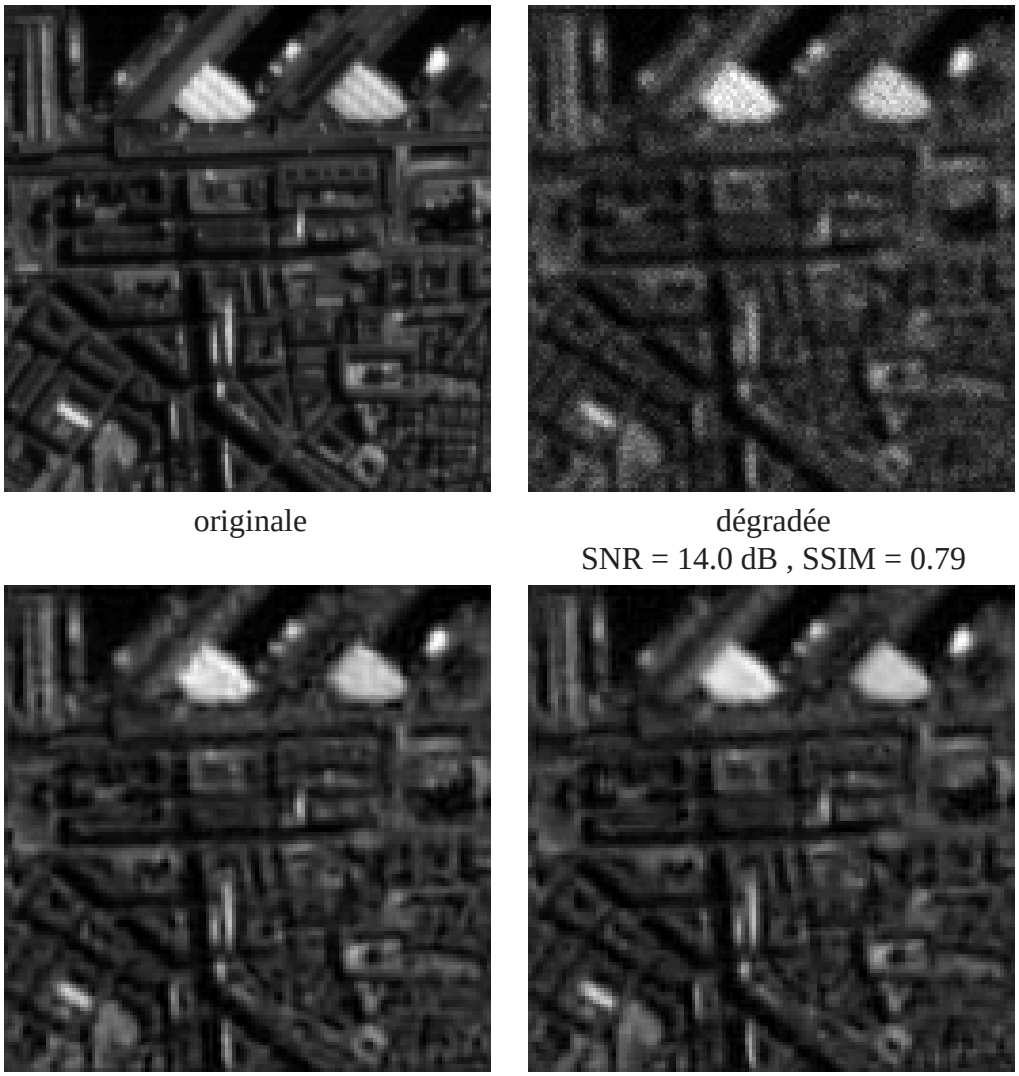
EM-ML
SNR = 2.17 dB , SSIM = 0.26

EM régularisé
SNR = 12.3 dB , SSIM = 0.64

DTT complexes version trame ajustée
SNR = 13.3 dB , SSIM = 0.69

DTT complexes
SNR = 14.2 dB , SSIM = 0.73

FIGURE 5.5 – Versions tronquées de l'image de « Barbara ». Images restaurées avec FS et DTT.



originale

dégradée
SNR = 14.0 dB , SSIM = 0.79

restaurée FS
SNR = 16.3 dB , SSIM = 0.87

restaurée FA
SNR = 16.6 dB , SSIM = 0.88

FIGURE 5.6 – FS versus FA.

CHAPITRE 6

Algorithmes proximaux pour la résolution de problèmes multicomposantes

6.1 Motivations

Dans ce chapitre, nous considérons divers problèmes inverses de traitement du signal, dans lesquels la solution idéale est représentée par m composantes, notées $x_1, \ldots, x_m$, appartenant respectivement à des espaces de Hilbert réels $\mathcal{H}_1, \ldots, \mathcal{H}_m$. De tels problèmes apparaissent dans de nombreux domaines allant de l'imagerie couleur à l'imagerie hyperspectrale en passant par le traitement du signal multi-canal et la décomposition d'images en composantes géométrique et de texture [Anthoine *et al.*, 2006; Aujol *et al.*, 2005; Aujol, Chambolle, 2005; Aujol *et al.*, 2006; Aujol, Chambolle, 2006; Chan *et al.*, 2007; Daubechies, Teschke, 2005; Katsaggelos *et al.*, 1993; Kang, 1998; Tschumperlé, Deriche, 2002; Wang *et al.*, 2008; Wen *et al.*, 2008]. La plupart du temps, les tâches de traitement de signal/image multicomposantes peuvent être formulées comme des problèmes variationnels de la forme,

$$\underset{x_1 \in \mathcal{H}_1, \ldots, x_m \in \mathcal{H}_m}{\text{minimiser}} \quad \Phi(x_1, \ldots, x_m), \tag{6.1}$$

où Φ est une fonction convexe modélisant l'information disponible sur les m composantes, leurs intéractions, et elle peut également prendre en compte le processus d'acquisition des données.

Le problème d'optimisation convexe (6.1) est trop générique pour être résolu directement. Il doit être formulé dans un style plus structuré pour obtenir des solutions numériques efficaces. A cette fin, Φ peut être décomposée comme une somme de p fonctions pouvant être traitées individuellement de façon plus aisée. Cela conduit au modèle suivant, sur lequel notre attention se portera à travers ce chapitre.

Problème 6.1 *Soient $(\mathcal{H}_i)_{1 \leq i \leq m}$ des espaces de Hilbert réels, et soient $(f_k)_{1 \leq k \leq p}$ des fonctions convexes, semi-continues inférieurement et propres de la somme directe des espaces de Hilbert $\mathcal{H}_1 \oplus \cdots \oplus \mathcal{H}_m$ vers $]-\infty, +\infty]$. Le problème consiste à*

$$\underset{x_1 \in \mathcal{H}_1, \ldots, x_m \in \mathcal{H}_m}{\text{minimiser}} \quad \sum_{k=1}^{p} f_k(x_1, \ldots, x_m), \tag{6.2}$$

sous l'hypothèse que des solutions existent.

Dans le cas de problèmes de traitement de signaux ou d'images univariés ($m = 1$), les méthodes proximales ont montré leur efficacité pour résoudre le problème 6.1 ; on pourra se référer aux chapitres précédents et aux références associées pour les travaux de bases et diverses applications. Par conséquent, il est naturel de se demander si ces méthodes peuvent être étendues aux problèmes multivariés. Des travaux initiaux qui s'orientent dans cette direction ont été menés dans [Briceño-Arias, Combettes, 2009] dans le cas particulier où $m = 2$, f_1 est une somme séparable (i.e., $f_1 \colon (x_i)_{1\leq i\leq m} \mapsto \sum_{i=1}^m \varphi_i(x_i)$), et f_2 est différentiable sur $\mathcal{H}_1 \oplus \cdots \oplus \mathcal{H}_m$ avec un gradient Lipschitz continu. Cette formulation couvre également les travaux tels que [Aujol *et al.*, 2005; Aujol, Chambolle, 2005; Aujol *et al.*, 2006; Combettes, Wajs, 2005; Goldburg, Marks II, 1985; Huang *et al.*, 2008; Vese, Osher, 2003; Vese, Osher, 2004; Wen *et al.*, 2008]. L'objectif de ce chapitre est de proposer une formulation générale et de présenter, sous des hypothèses adéquates, plusieurs algorithmes proximaux offrant des garanties de convergence vers une solution du problème 6.1.

Ce chapitre est organisé de la façon suivante. Dans le paragraphe 6.2, nous introduirons les principales notations utilisées dans ce chapitre. De nouveaux résultats pour des opérateurs proximaux multicomposantes seront présentés dans le paragraphe 6.3. Dans le paragraphe 6.4, nous décrirons les algorithmes proximaux qui sont pertinents pour résoudre le problème 6.1. Finalement, nous illustrerons dans le paragraphe 6.5 l'efficacité des algorithmes proposés sur trois exemples d'imagerie multicomposante.

6.2 Notations et rappels

Dans ce chapitre, $\mathcal{H}$, $\mathcal{G}$, et $(\mathcal{H}_i)_{1\leq i\leq m}$ sont des espaces de Hilbert réels. Pour faciliter les notations, les produits scalaires associés sont tous notés $\langle \cdot, \cdot \rangle$, la norme associée $\|\cdot\|$, et les opérateurs identités Id. Nous désignerons par $\boldsymbol{x} = (x_i)_{1\leq i\leq m}$ un élément générique dans $\mathcal{H}_1 \times \cdots \times \mathcal{H}_m$ et par $\boldsymbol{\mathcal{H}}$ la somme directe des espaces de Hilbert $\mathcal{H}_1 \oplus \cdots \oplus \mathcal{H}_m$, i.e. l'espace produit $\mathcal{H}_1 \times \cdots \times \mathcal{H}_m$ muni du produit scalaire

$$(\boldsymbol{x}, \boldsymbol{y}) \mapsto \sum_{i=1}^m \langle x_i, y_i \rangle. \tag{6.3}$$

L'espace des opérateurs bornés de $\mathcal{H}$ vers $\mathcal{G}$ est noté $\mathcal{B}(\mathcal{H}, \mathcal{G})$.

Soient C et D deux sous-ensembles convexes non vides de $\mathcal{H}$. Nous disons que 0 appartient à l'intérieur relatif fort de C, de façon symbolique, $0 \in \operatorname{sri} C$, si $\bigcup_{\lambda>0} \lambda C = \overline{\operatorname{span}}\, C$. En particulier, si nous choisissons $C - D = \big\{u - v \;\big|\; (u, v) \in C \times D\big\}$, l'inclusion $0 \in \operatorname{sri}(C - D)$ se vérifie dans chacun des cas suivants :

- $C - D$ est un sous-ensemble vectoriel.
- $0 \in \operatorname{int}(C - D)$.
- $C \cap \operatorname{int} D \neq \varnothing$.

- $\mathcal{H}$ est de dimension finie et $(\operatorname{ri} C) \cap (\operatorname{ri} D) \neq \varnothing$, où $\operatorname{ri} C$ représente l'intérieur relatif de C.

Des compléments d'informations sur l'analyse convexe peuvent être trouvés dans [Zălinescu, 2002].

L'opérateur proximal d'une fonction $\varphi \in \Gamma_0(\mathcal{H})$ au point $u \in \mathcal{H}$ est caractérisé par l'inégalité variationnelle :

$$(\forall p \in \mathcal{H}) \quad p = \operatorname{prox}_\varphi u \quad \Leftrightarrow \quad (\forall v \in \mathcal{H}) \quad \big\langle v - p, u - p \big\rangle + \varphi(p) \leq \varphi(v). \tag{6.4}$$

Cette dernière nous sera utile pour certaines des démonstrations qui suivent.

6.3 Opérateurs proximaux multicomposantes

Le calcul d'opérateurs proximaux dans l'espace de Hilbert $\boldsymbol{\mathcal{H}}$ joue un rôle fondamental dans ce chapitre. Ci-dessous, nous décrivons des situations importantes dans lesquelles ce calcul est explicite.

Proposition 6.2 *Supposons que, pour tout $i \in \{1, \ldots, m\}$, $(e_{i,k})_{k \in \mathbb{K}}$ est une base orthonormale de $\mathcal{H}_i$. De plus, soient $(\phi_k)_{k \in \mathbb{K}}$ des fonctions dans $\Gamma_0(\mathbb{R}^m)$ et supposons que l'une des hypothèses suivantes est vérifiée.*

(i) *Pour tout $i \in \{1, \ldots, m\}$, $\mathcal{H}_i$ est de dimension infinie et, pour tout $k \in \mathbb{K}$, $\phi_k \geq \phi_k(\mathbf{0}) = 0$.*

(ii) *Pour tout $i \in \{1, \ldots, m\}$, $\mathcal{H}_i$ est de dimension finie.*

Soit

$$\begin{aligned} f \colon \boldsymbol{\mathcal{H}} &\to \left]-\infty, +\infty\right] \\ \boldsymbol{x} &\mapsto \sum_{k \in \mathbb{K}} \phi_k\big(\big\langle x_1, e_{1,k} \big\rangle, \ldots, \big\langle x_m, e_{m,k} \big\rangle\big). \end{aligned} \tag{6.5}$$

On a alors $f \in \Gamma_0(\boldsymbol{\mathcal{H}})$ et, pour tout $\boldsymbol{x} \in \boldsymbol{\mathcal{H}}$,

$$\operatorname{prox}_f \boldsymbol{x} = \left(\sum_{k \in \mathbb{K}} \pi_{1,k} e_{1,k}, \ldots, \sum_{k \in \mathbb{K}} \pi_{m,k} e_{m,k} \right), \tag{6.6}$$

où

$$(\forall k \in \mathbb{K}) \qquad (\pi_{1,k}, \ldots, \pi_{m,k}) = \operatorname{prox}_{\phi_k}\big(\big\langle x_1, e_{1,k} \big\rangle, \ldots, \big\langle x_m, e_{m,k} \big\rangle\big). \tag{6.7}$$

<u>Preuve</u> :
Nous définissons, pour tout $k \in \mathbb{K}$,

$$\begin{aligned} \psi_k \colon \boldsymbol{\mathcal{H}} &\to \left]-\infty, +\infty\right] \\ \boldsymbol{x} &\mapsto \phi_k\big(\big\langle x_1, e_{1,k} \big\rangle, \ldots, \big\langle x_m, e_{m,k} \big\rangle\big). \end{aligned} \tag{6.8}$$

Nos hypothèses impliquent alors que les fonctions $(\psi_k)_{k\in\mathbb{K}}$ sont dans $\Gamma_0(\boldsymbol{\mathcal{H}})$. Sous l'hypothèse (i), en supposant sans perte de généralité que $\mathbb{K}=\mathbb{N}$, nous pouvons écrire $f = \sup_{K\in\mathbb{K}}\sum_{k=0}^{K}\psi_k$. Comme la semi-continuité inférieure et la convexité sont préservées par somme finie et en prenant le supremum, nous pouvons en déduire que f est semi-continue inférieurement et convexe. De plus, puisque $f(\mathbf{0})=0$, nous obtenons $f\in\Gamma_0(\boldsymbol{\mathcal{H}})$. D'autre part, sous l'hypothèse (ii), la somme dans (6.5) est finie et nos hypothèses impliquent directement que $f\in\Gamma_0(\boldsymbol{\mathcal{H}})$.
Maintenant, soit $\boldsymbol{x}\in\boldsymbol{\mathcal{H}}$ et soit $|\cdot|$ la norme associée à l'espace euclidien $\mathbb{R}^m$. Soit

$$(\forall i\in\{1,\ldots,m\})(\forall k\in\mathbb{K})\quad \xi_{i,k}=\big\langle x_i,e_{i,k}\big\rangle. \tag{6.9}$$

De plus, pour tout $i\in\{1,\ldots,m\}$, soit $y_i\in\mathcal{H}_i$ et soit $(\eta_{i,k})_{k\in\mathbb{K}}=\big(\big\langle y_i,e_{i,k}\big\rangle\big)_{k\in\mathbb{K}}$. Nous déduisons de (6.4) et (6.7) que, pour tout $k\in\mathbb{K}$,

$$\sum_{i=1}^{m}(\eta_{i,k}-\pi_{i,k})(\xi_{i,k}-\pi_{i,k})+\phi_k(\pi_{1,k},\ldots,\pi_{m,k})\leq\phi_k(\eta_{1,k},\ldots,\eta_{m,k}). \tag{6.10}$$

Supposons dans un premier temps que l'hypothèse (i) soit vérifiée. Puisque, pour tout $k\in\mathbb{K}$, $\mathbf{0}$ est un minimiseur de ϕ_k, (6.4) vérifie $\operatorname{prox}_{\phi_k}\mathbf{0}=\mathbf{0}$. Par conséquent, en utilisant (6.7), (6.9), le fait que l'opérateur proximal est contractant (se référer au lemme 3.9 dans le chapitre 3), et l'identité de Parseval, nous obtenons

$$\begin{aligned}
\sum_{k\in\mathbb{K}}\sum_{i=1}^{m}|\pi_{i,k}|^2 &= \sum_{k\in\mathbb{K}}|(\pi_{1,k},\ldots,\pi_{m,k})|^2\\
&= \sum_{k\in\mathbb{K}}|\operatorname{prox}_{\phi_k}(\xi_{1,k},\ldots,\xi_{m,k})|^2\\
&= \sum_{k\in\mathbb{K}}|\operatorname{prox}_{\phi_k}(\xi_{1,k},\ldots,\xi_{m,k})-\operatorname{prox}_{\phi_k}\mathbf{0}|^2\\
&\leq \sum_{k\in\mathbb{K}}|(\xi_{1,k},\ldots,\xi_{m,k})-\mathbf{0}|^2\\
&= \sum_{k\in\mathbb{K}}|(\xi_{1,k},\ldots,\xi_{m,k})|^2\\
&= \sum_{k\in\mathbb{K}}\sum_{i=1}^{m}|\xi_{i,k}|^2\\
&= \sum_{i=1}^{m}\|x_i\|^2.
\end{aligned} \tag{6.11}$$

Par conséquent, pour tout $i\in\{1,\ldots,m\}$, $\sum_{k\in\mathbb{K}}|\pi_{i,k}|^2<+\infty$ et nous pouvons définir

$$(\forall i\in\{1,\ldots,m\})\qquad z_i=\sum_{k\in\mathbb{K}}\pi_{i,k}e_{i,k}. \tag{6.12}$$

Nous pouvons également définir cette dernière équation sous l'hypothèse (ii) car dans cette configuration $\mathbb{K}$ est fini. Il reste alors à montrer que $(z_i)_{1\leq i\leq m} = \operatorname{prox}_f(x_1,\ldots,x_m)$. En sommant sur k dans (6.10), cela conduit à

$$\sum_{k\in\mathbb{K}}\sum_{i=1}^{m}(\eta_{i,k}-\pi_{i,k})(\xi_{i,k}-\pi_{i,k}) + \sum_{k\in\mathbb{K}}\phi_k(\pi_{1,k},\ldots,\pi_{m,k}) \leq \sum_{k\in\mathbb{K}}\phi_k(\eta_{1,k},\ldots,\eta_{m,k}) \quad (6.13)$$

et, par conséquent,

$$\sum_{i=1}^{m}\big\langle y_i - z_i, x_i - z_i\big\rangle + g(z_1,\ldots,z_m) \leq g(y_1,\ldots,y_m). \quad (6.14)$$

En utilisant (6.4), la preuve est achevée. □

Proposition 6.3 *Pour tout $j \in \{1,\ldots,q\}$, soit $\mathcal{G}_j$ un espace de Hilbert réel, soit $\varphi_j \in \Gamma_0(\mathcal{G}_j)$, et, pour tout $i \in \{1,\ldots,m\}$, soit $L_{j,i} \in \mathcal{B}(\mathcal{H}_i,\mathcal{G}_j)$. Soit*

$$\begin{aligned} f\colon \boldsymbol{\mathcal{H}} &\to \left]-\infty,+\infty\right] \\ \boldsymbol{x} &\mapsto \sum_{j=1}^{q}\varphi_j\bigg(\sum_{i=1}^{m}L_{j,i}x_i\bigg) \end{aligned} \quad (6.15)$$

et supposons que, pour tout $j \in \{1,\ldots,q\}$, il existe $\alpha_j \in \left]0,+\infty\right[$ tel que

$$(\forall k \in \{1,\ldots,q\}) \qquad \sum_{i=1}^{m}L_{j,i}\circ L_{k,i}^* = \begin{cases}\alpha_j\mathrm{Id}, & \textit{si } j=k;\\ 0, & \textit{sinon}.\end{cases} \quad (6.16)$$

On a alors $f \in \Gamma_0(\boldsymbol{\mathcal{H}})$ et, pour tout $\boldsymbol{x} \in \boldsymbol{\mathcal{H}}$,

$$\operatorname{prox}_f\boldsymbol{x} = (p_1,\ldots,p_m) \quad (6.17)$$

où, pour tout $i \in \{1,\ldots,m\}$,

$$p_i = x_i + \sum_{j=1}^{q}\alpha_j^{-1}L_{j,i}^*\operatorname{prox}_{\alpha_j\varphi_j}\bigg(\sum_{k=1}^{m}L_{j,k}x_k\bigg) - \sum_{j=1}^{q}\alpha_j^{-1}L_{j,i}^*\sum_{k=1}^{m}L_{j,k}x_k. \quad (6.18)$$

<u>Preuve</u> :
Notons par $\boldsymbol{\mathcal{G}}$ l'espace produit $\mathcal{G}_1\times\cdots\times\mathcal{G}_q$ muni du produit scalaire

$$(\boldsymbol{y},\boldsymbol{z}) \mapsto \sum_{j=1}^{q}\alpha_j^{-1}\big\langle y_j, z_j\big\rangle. \quad (6.19)$$

Nous pouvons écrire $f = g \circ \boldsymbol{L}$, où

$$\begin{aligned} g\colon \boldsymbol{\mathcal{G}} &\to \left]-\infty,+\infty\right] \\ \boldsymbol{y} &\mapsto \sum_{j=1}^{q} \varphi_j(y_j) \end{aligned} \tag{6.20}$$

et $\boldsymbol{L} \in \mathcal{B}\left(\boldsymbol{\mathcal{H}}, \boldsymbol{\mathcal{G}}\right)$ est défini par

$$\begin{aligned} \boldsymbol{L}\colon \boldsymbol{\mathcal{H}} &\to \boldsymbol{\mathcal{G}} \\ \boldsymbol{x} &\mapsto \left(\sum_{i=1}^{m} L_{1,i}x_i, \ldots, \sum_{i=1}^{m} L_{q,i}x_i\right), \end{aligned} \tag{6.21}$$

Il résulte de (6.19) que, pour tout, $(\boldsymbol{x}, \boldsymbol{y}) \in \boldsymbol{\mathcal{H}} \times \boldsymbol{\mathcal{G}}$,

$$\begin{aligned} \left\langle \boldsymbol{L}\boldsymbol{x}, \boldsymbol{y}\right\rangle &= \sum_{j=1}^{q} \alpha_j^{-1} \left\langle \sum_{i=1}^{m} L_{j,i}x_i \,\middle|\, y_j \right\rangle \\ &= \sum_{j=1}^{q} \alpha_j^{-1} \sum_{i=1}^{m} \left\langle x_i, L_{j,i}^* y_j \right\rangle \\ &= \sum_{i=1}^{m} \left\langle x_i \,\middle|\, \sum_{j=1}^{q} \alpha_j^{-1} L_{j,i}^* y_j \right\rangle, \end{aligned} \tag{6.22}$$

relation à partir de laquelle nous introduisons l'adjoint de $\boldsymbol{L}$ qui est

$$\begin{aligned} \boldsymbol{L}^*\colon \boldsymbol{\mathcal{G}} &\to \boldsymbol{\mathcal{H}} \\ \boldsymbol{y} &\mapsto \left(\sum_{j=1}^{q} \alpha_j^{-1} L_{j,1}^* y_j, \ldots, \sum_{j=1}^{q} \alpha_j^{-1} L_{j,m}^* y_j\right). \end{aligned} \tag{6.23}$$

Nous obtenons alors à partir de (6.16) que $\boldsymbol{L} \circ \boldsymbol{L}^* = \mathrm{Id}$. La proposition 3.12 (chapitre 3) implique que $f = g \circ \boldsymbol{L} \in \Gamma_0(\boldsymbol{\mathcal{H}})$ et que

$$\mathrm{prox}_{g \circ \boldsymbol{L}} = \mathrm{Id} + \boldsymbol{L}^* \circ (\mathrm{prox}_g - \mathrm{Id}) \circ \boldsymbol{L}. \tag{6.24}$$

De plus, on déduit de (6.19) et (6.20) que, pour tout $\boldsymbol{y} \in \boldsymbol{\mathcal{G}}$,

$$\mathrm{prox}_g \boldsymbol{y} = (\mathrm{prox}_{\alpha_1\varphi_1} y_1, \ldots, \mathrm{prox}_{\alpha_q\varphi_q} y_q). \tag{6.25}$$

En considérant simultanément, (6.21), (6.23), (6.24), et (6.25) cela conduit à (6.17)–(6.18). $\square$

Corollaire 6.4 *Soit $\varphi \in \Gamma_0(\mathcal{G})$ et, pour tout $i \in \{1, \ldots, m\}$, soit $L_i \in \mathcal{B}(\mathcal{H}_i, \mathcal{G})$. Soit*

$$\begin{aligned} f\colon \boldsymbol{\mathcal{H}} &\to \left]-\infty, +\infty\right] \\ \boldsymbol{x} &\mapsto \varphi\Big(\sum_{i=1}^{m} L_i x_i\Big) \end{aligned} \tag{6.26}$$

et supposons qu'il existe $\alpha \in \left]0, +\infty\right[$ tel que

$$\sum_{i=1}^{m} L_i \circ L_i^* = \alpha \mathrm{Id}. \tag{6.27}$$

On a alors $f \in \Gamma_0(\boldsymbol{\mathcal{H}})$ et, pour tout $\boldsymbol{x} \in \boldsymbol{\mathcal{H}}$, $\operatorname{prox}_f \boldsymbol{x} = (p_1, \ldots, p_m)$ où, pour tout $i \in \{1, \ldots, m\}$,

$$p_i = x_i + \alpha^{-1} L_i^* \operatorname{prox}_{\alpha\varphi}\Big(\sum_{k=1}^{m} L_k x_k\Big) - \alpha^{-1} L_i^* \sum_{k=1}^{m} L_k x_k. \tag{6.28}$$

<u>Preuve</u> :
Il suffit de poser $q = 1$ dans la proposition 6.3. □

Proposition 6.5 *Supposons que $\mathcal{G}$ soit de dimension finie K, soient $(\phi_k)_{1 \le k \le K}$ des fonctions de $\Gamma_0(\mathbb{R})$, et soit $(e_k)_{1 \le k \le K}$ une base orthonormale de $\mathcal{G}$. Pour tout $i \in \{1, \ldots, m\}$, soit $L_i \in \mathcal{B}(\mathcal{H}_i, \mathcal{G})$ et supposons qu'il existe $\{\alpha_k\}_{1 \le k \le K} \subset \left]0, +\infty\right[$ tel que*

$$(\forall y \in \mathcal{G}) \quad \sum_{i=1}^{m} L_i L_i^* y = \sum_{k=1}^{K} \alpha_k \langle y, e_k \rangle e_k. \tag{6.29}$$

Soit

$$f\colon \boldsymbol{\mathcal{H}} \to \left]-\infty, +\infty\right] : \boldsymbol{x} \mapsto \sum_{k=1}^{K} \phi_k\Big(\Big\langle \sum_{j=1}^{m} L_j x_j \,\Big|\, e_k \Big\rangle\Big) \tag{6.30}$$

et, pour tout $k \in \{1 \ldots, K\}$,

$$\pi_k = \frac{1}{\alpha_k} \operatorname{prox}_{\alpha_k \phi_k}\Big(\Big\langle \sum_{j=1}^{m} L_j x_j \,\Big|\, e_k \Big\rangle\Big). \tag{6.31}$$

On a alors $f \in \Gamma_0(\boldsymbol{\mathcal{H}})$ et, pour tout $\boldsymbol{x} \in \boldsymbol{\mathcal{H}}$, $\operatorname{prox}_f \boldsymbol{x} = (p_i)_{1 \le i \le m}$ où, pour tout $i \in \{1, \ldots, m\}$,

$$p_i = x_i + L_i^* \sum_{k=1}^{K} \Big(\pi_k - \frac{1}{\alpha_k} \sum_{j=1}^{m} \langle L_j x_j, e_k \rangle\Big) e_k. \tag{6.32}$$

Preuve :
Pour tout $j \in \{1, \ldots, K\}$, soit $\mathcal{G}_j = \mathbb{R}$, $\varphi_j = \phi_j$, et

$$(\forall i \in \{1, \ldots, m\}) \qquad L_{j,i} \colon \mathcal{H}_i \to \mathcal{G}_j \colon x \mapsto \big\langle L_i x, e_j \big\rangle, \tag{6.33}$$

d'où

$$L_{j,i}^* \colon \mathcal{G}_j \to \mathcal{H}_i \colon \xi \mapsto \xi\, L_i^* e_j. \tag{6.34}$$

Pour tout j et k dans $\{1, \ldots, K\}$ et tout $\xi \in \mathbb{R}$, nous déduisons alors de (6.29) que

$$\begin{aligned}
\sum_{i=1}^{m} L_{j,i} L_{k,i}^* \, \xi &= \sum_{i=1}^{m} L_{j,i} \, \xi \, L_i^* e_k \\
&= \xi \sum_{i=1}^{m} \big\langle (L_i L_i^*) e_k, e_j \big\rangle \\
&= \xi \Big\langle \sum_{l=1}^{K} \alpha_l \big\langle e_k, e_l \big\rangle e_l \;\Big|\; e_j \Big\rangle \\
&= \xi \sum_{l=1}^{K} \alpha_l \big\langle e_k, e_l \big\rangle \big\langle e_l, e_j \big\rangle.
\end{aligned} \tag{6.35}$$

Par conséquent, pour tout $j \in \{1, \ldots, K\}$, (6.16) est vérifiée. On en déduit que la proposition 6.3 avec $q = K$ garantit que $f \in \Gamma_0(\boldsymbol{\mathcal{H}})$, et (6.18) se réduit à (6.32). □

6.4 Algorithmes proximaux multicomposantes

Nous présentons plusieurs algorithmes proximaux pour résoudre le problème 6.1 en formulant différentes hypothèses sur les fonctions mises en jeu. La plupart de ces algorithmes sont robustes aux erreurs numériques sur le calcul des points proximaux et des gradients. Pour quantifier le taux d'erreur qui est toléré, nous utiliserons la notation qui suit. Soient deux suites $(\boldsymbol{x}_n)_{n\in\mathbb{N}}$ et $(\boldsymbol{y}_n)_{n\in\mathbb{N}}$ dans $\boldsymbol{\mathcal{H}}$,

$$\Big[(\forall n \in \mathbb{N}) \;\; \boldsymbol{x}_n \approx \boldsymbol{y}_n\Big] \;\Leftrightarrow\; \sum_{n\in\mathbb{N}} \|\boldsymbol{x}_n - \boldsymbol{y}_n\| < +\infty. \tag{6.36}$$

6.4.1 Algorithme explicite-implicite

Problème 6.6 *Dans le problème 6.1, supposons que $p = 2$ et que la fonction f_2 soit différentiable sur $\boldsymbol{\mathcal{H}}$ avec un gradient continu β–Lipschitz pour $\beta \in \,]0, +\infty[$. Le problème est alors de*

$$\underset{x_1 \in \mathcal{H}_1, \ldots, x_m \in \mathcal{H}_m}{\text{minimiser}} \quad f_1(x_1, \ldots, x_m) + f_2(x_1, \ldots, x_m), \tag{6.37}$$

sous les hypothèses qu'une solution existe.

Le cas particulier où f_1 est une somme séparable et f_2 induit un mélange linéaire des variables recherchées a été abordé dans [Briceño-Arias, Combettes, 2009]. Les résultats suivants s'intéressent au cas général ; cela nécessite comme hypothèse que l'opérateur proximal de f_1 soit calculable avec une erreur quantifiable.

Theorème 6.7 *Soient* $(x_{1,n})_{n\in\mathbb{N}}, \ldots, (x_{m,n})_{n\in\mathbb{N}}$ *des suites générées par la routine suivante.*

$$
\begin{array}{l}
\text{Initialisation} \\
\left\lfloor
\begin{array}{l}
\varepsilon \in \left]0, \min\{1, 1/\beta\}\right[\\
\text{Pour } i = 1, \ldots, m \\
\lfloor\ x_{i,0} \in \mathcal{H}_i
\end{array}
\right. \\
\text{Pour } n = 0, 1, \ldots \\
\left\lfloor
\begin{array}{l}
(y_{i,n})_{1\leq i\leq m} \approx \nabla f_2(x_{i,n})_{1\leq i\leq m} \\
\gamma_n \in [\varepsilon, (2/\beta) - \varepsilon] \\
(u_{i,n})_{1\leq i\leq m} \approx \operatorname{prox}_{\gamma_n f_1}(x_{i,n} - \gamma_n y_{i,n})_{1\leq i\leq m} \\
\lambda_n \in [\varepsilon, 1] \\
\text{Pour } i = 1, \ldots, m \\
\lfloor\ x_{i,n+1} = x_{i,n} + \lambda_n(u_{i,n} - x_{i,n})
\end{array}
\right.
\end{array}
$$

Pour tout $i \in \{1, \ldots, m\}$*, la suite* $(x_{i,n})_{n\in\mathbb{N}}$ *converge alors faiblement vers un point* $x_i \in \mathcal{H}_i$*. De plus,* $(x_i)_{1\leq i\leq m}$ *est une solution du problème 6.6.*

Preuve :
On applique [Combettes, Wajs, 2005, théorème 3.4(i)] à $\boldsymbol{\mathcal{H}}$ et on utilise (6.3). □

Remarque 6.8

(i) *Des variantes multicomposantes de l'algorithme explicite-implicite comme les méthodes de premier ordre de type Nesterov [Beck, Teboulle, 2009; Nesterov, 1983; Weiss* et al.*, 2009] peuvent être obtenues avec des reformulations similaires dans* $\boldsymbol{\mathcal{H}}$*. Cependant, pour ces méthodes, la convergence des itérées vers une solution du problème 6.6 n'est pas garantie, même en considérant une formulation en dimension finie.*

(ii) *Les conditions permettant d'obtenir la convergence forte du théorème 6.7 peuvent être déduites de [Combettes, Wajs, 2005, théorème 3.4(iv)].*

6.4.2 Algorithme de Douglas-Rachford

Dans ce paragraphe, nous relâchons l'hypothèse assurant que la fonction f_2 soit une fonction lisse. Cependant nous supposons que l'opérateur proximal de cette fonction est implantable avec une erreur quantifiable.

Problème 6.9 *Dans le problème 6.1, supposons que* $p = 2$ *et que*

$$\mathbf{0} \in \operatorname{sri}(\operatorname{dom} f_1 - \operatorname{dom} f_2). \tag{6.38}$$

Le problème est alors de

$$\underset{x_1 \in \mathcal{H}_1, \ldots, x_m \in \mathcal{H}_m}{\text{minimiser}} \quad f_1(x_1, \ldots, x_m) + f_2(x_1, \ldots, x_m), \tag{6.39}$$

sous l'hypothèse qu'une solution existe.

Theorème 6.10 *Soit* $(x_{1,n})_{n\in\mathbb{N}}$, ..., $(x_{m,n})_{n\in\mathbb{N}}$ *les suites générées par les routines suivantes.*

Initialisation

$\varepsilon \in \left]0, 1\right[$
$\gamma \in \left]0, +\infty\right[$
Pour $i = 1, \ldots, m$
$\quad x_{i,0} \in \mathcal{H}_i$

Pour $n = 0, 1, \ldots$

$(y_{i,n})_{1\leq i\leq m} \approx \operatorname{prox}_{\gamma f_2}(x_{i,n})_{1\leq i\leq m}$
$(u_{i,n})_{1\leq i\leq m} \approx \operatorname{prox}_{\gamma f_1}(2y_{i,n} - x_{i,n})_{1\leq i\leq m}$
$\lambda_n \in [\varepsilon, 2-\varepsilon]$
Pour $i = 1, \ldots, m$
$\quad x_{i,n+1} = x_{i,n} + \lambda_n(u_{i,n} - y_{i,n})$.

Pour tout $i \in \{1, \ldots, m\}$, *la suite* $(x_{i,n})_{n\in\mathbb{N}}$ *converge alors faiblement vers un point* $x_i \in \mathcal{H}_i$ *et* $\operatorname{prox}_{\gamma f_2}(x_1, \ldots, x_m)$ *est une solution du problème 6.9. De plus, si* $\mathcal{H}$ *est de dimension finie, la suite* $((y_{i,n})_{1\leq i\leq m})_{n\in\mathbb{N}}$ *converge vers* $\operatorname{prox}_{\gamma f_2}(x_1, \ldots, x_m)$.

Preuve :
On applique [Combettes, Pesquet, 2007a, théorème 20] dans $\mathcal{H}$ et on utilise (6.3). □

Remarque 6.11

(i) *Les conditions de convergence forte du théorème 6.10 peuvent être déduites de [Combettes, 2009, théorème 2.1(ii)].*

(ii) *Si* $\mathcal{H}$ *est de dimension finie, la condition de qualification* (6.38) *se réduit à*

$$(\operatorname{ri}\operatorname{dom} f_1) \cap (\operatorname{ri}\operatorname{dom} f_2) \neq \varnothing. \tag{6.40}$$

6.4.3 Algorithme proximal parallèle (PPXA)

L'algorithme présenté dans ce paragraphe a pour but de résoudre le problème 6.1 sous des hypothèses techniques minimales. Le coût de l'implantation dépend de la complexité du calcul de chaque opérateur proximal.

Problème 6.12 *Dans le problème 6.1, supposons que*

$$\mathbf{0} \in \operatorname{sri}(\boldsymbol{D} - \operatorname{dom} f_1 \times \cdots \times \operatorname{dom} f_p) \tag{6.41}$$

où $\boldsymbol{D} = \big\{(\boldsymbol{x}, \ldots, \boldsymbol{x}) \;\big|\; \boldsymbol{x} \in \boldsymbol{\mathcal{H}}\big\}$. *Le problème est de*

$$\underset{x_1 \in \mathcal{H}_1, \ldots, x_m \in \mathcal{H}_m}{\text{minimiser}} \quad \sum_{k=1}^{p} f_k(x_1, \ldots, x_m), \tag{6.42}$$

sous l'hypothèse qu'une solution existe.

Dans [Afonso *et al.*, 2009], un cas particulier du problème 6.12 dans des espaces de dimension finie est considéré. Les auteurs utilisent la méthode des directions alternées des multiplicateurs (ADMM : « Alternating Direction Method of Multipliers »). L'algorithme utilisé ci-dessous est une application de PPXA proposé dans [Combettes, Pesquet, 2008].

Theorème 6.13 *Soient* $(x_{1,n})_{n\in\mathbb{N}}, \ldots, (x_{m,n})_{n\in\mathbb{N}}$ *des suites générées par la routine suivante.*

Initialisation

$\varepsilon \in \left]0,1\right[,\ \gamma \in \left]0,+\infty\right[, \{\omega_k\}_{1\leq k\leq p} \subset \left]0,1\right]$ et $\displaystyle\sum_{k=1}^{p} \omega_k = 1$

Pour $i = 1, \ldots, m$

  Pour $k = 1, \ldots, p$

  $y_{i,k,0} \in \mathcal{H}_i$

  $\displaystyle x_{i,0} = \sum_{k=1}^{p} \omega_k y_{i,k,0}$

Pour $n = 0, 1, \ldots$

  Pour $k = 1, \ldots, p$

  $(u_{i,k,n})_{1\leq i\leq m} \approx \operatorname{prox}_{\gamma f_k/\omega_k}(y_{i,k,n})_{1\leq i\leq m}$

  Pour $i = 1, \ldots, m$

  $\displaystyle s_{i,n} = \sum_{k=1}^{p} \omega_k u_{i,k,n}$

  $\lambda_n \in [\varepsilon, 2-\varepsilon]$

  Pour $k = 1, \ldots, p$

  $y_{i,k,n+1} = y_{i,k,n} + \lambda_n(2s_{i,n} - x_{i,n} - u_{i,k,n})$

  $x_{i,n+1} = x_{i,n} + \lambda_n(s_{i,n} - x_{i,n})$

Pour tout $i \in \{1, \ldots, m\}$*, la suite* $(x_{i,n})_{n\in\mathbb{N}}$ *converge alors faiblement vers un point* $x_i \in \mathcal{H}_i$*. De plus,* $(x_1, \ldots, x_m)$ *est une solution du problème 6.12.*

Preuve :
Il suffit d'appliquer [Combettes, Pesquet, 2008, théorème 3.4] dans $\boldsymbol{\mathcal{H}}$ et utiliser (6.3). □

Remarque 6.14 *Supposons que* $\boldsymbol{\mathcal{H}}$ *est de dimension finie et que*

$$\bigcap_{k=1}^{p} \operatorname{ri} \operatorname{dom} f_k \neq \varnothing. \tag{6.43}$$

Il résulte de [Combettes, Pesquet, 2008, proposition 3.6(vi)] que la condition de qualification (6.41) *est satisfaite.*

6.4.4 Algorithme de type Dykstra

Nous considérons des cas particuliers du problème 6.1 pour lesquels f_p est une simple fonction quadratique.

Problème 6.15 *Dans le problème 6.1, supposons que* $p \geq 3$*,*

$$\bigcap_{k=1}^{p-1} \operatorname{dom} f_k \neq \varnothing, \tag{6.44}$$

et que $f_p \colon \boldsymbol{x} \mapsto (p-1)\sum_{i=1}^{m} \|x_i - z_i\|^2/2$*, où* $\boldsymbol{z} \in \boldsymbol{\mathcal{H}}$*. Le problème est alors de*

$$\underset{x_1\in\mathcal{H}_1,\ldots,\, x_m\in\mathcal{H}_m}{\text{minimiser}} \quad \sum_{k=1}^{p-1} f_k(x_1, \ldots, x_m) + \frac{p-1}{2}\sum_{i=1}^{m} \|x_i - z_i\|^2. \tag{6.45}$$

Soit $f = \sum_{k=1}^{p-1} f_k/(p-1)$. Il résulte alors de (6.44) que $f \in \Gamma_0(\boldsymbol{\mathcal{H}})$. Au vu de (6.3), le problème 6.15 admet une solution unique, qui est $\operatorname{prox}_f \boldsymbol{z}$.

Theorème 6.16 *Soient* $(x_{1,n})_{n\in\mathbb{N}}$, ..., $(x_{m,n})_{n\in\mathbb{N}}$ *des suites générées par les routines suivantes.*

$$
\begin{array}{l}
\text{Initialisation} \\
\left\lfloor \begin{array}{l}
\text{Pour } i = 1, \ldots, m \\
\left\lfloor \begin{array}{l}
x_{i,0} = z_i \\
\text{Pour } k = 1, \ldots, p-1 \\
\left\lfloor\, y_{i,k,0} = x_{i,0} \right.
\end{array} \right.
\end{array} \right. \\
\text{Pour } n = 0, 1, \ldots \\
\left\lfloor \begin{array}{l}
\text{Pour } k = 1, \ldots, p-1 \\
\left\lfloor\, (u_{i,k,n})_{1\leq i\leq m} = \operatorname{prox}_{f_k}(y_{i,k,n})_{1\leq i\leq m} \right. \\
\text{Pour } i = 1, \ldots, m \\
\left\lfloor \begin{array}{l}
x_{i,n+1} = \dfrac{1}{p-1}\displaystyle\sum_{k=1}^{p-1} u_{i,k,n} \\
\text{Pour } k = 1, \ldots, p-1 \\
\left\lfloor\, y_{i,k,n+1} = x_{i,n+1} + y_{i,k,n} - u_{i,k,n}. \right.
\end{array} \right.
\end{array} \right.
\end{array}
\tag{6.46}
$$

Pour tout $i \in \{1, \ldots, m\}$, *la suite* $(x_{i,n})_{n\in\mathbb{N}}$ *converge alors fortement vers un point* $x_i \in \mathcal{H}_i$. *De plus,* $(x_1, \ldots, x_m)$ *est une solution du problème 6.15.*

Preuve :
On applique [Combettes, 2009, théorème 4.2] dans $\boldsymbol{\mathcal{H}}$ et on utilise (6.3). □

Remarque 6.17 *Supposons que* (6.44) *est remplacée par la condition plus forte* (6.41) *(appliquée aux fonctions* $(f_k)_{1\leq k\leq p-1}$*). Il résulte alors de [Combettes, 2009, théorème 3.3] que les conclusions du théorème présenté ci-dessus restent valables si les opérateurs proximaux sont implantés de façon approximatives dans* (6.46).

6.5 Applications à la décomposition et à la restauration d'images

Dans ce paragraphe, nous appliquons les algorithmes proposés dans le paragraphe 6.4 aux problèmes de restauration d'images stéréoscopiques, de débruitage d'images multispectrales et aux problèmes de décomposition d'images.

6.5.1 Restauration d'images stéréoscopiques

6.5.1-a) Formulation du problème

Nous considérons le problème qui consiste à restaurer une paire d'images stéréoscopiques, notées $\overline{x}_1 \in \mathbb{R}^N$ et $\overline{x}_2 \in \mathbb{R}^N$, constituées de N pixels et qui correspondent aux vues gauche et droite de la même scène. Pour une valeur donnée du champ de disparité, le processus de compensation de disparité entre les deux images est modélisé par

$$\overline{x}_1 = D\overline{x}_2 + v, \tag{6.47}$$

où $D \in \mathbb{R}^{N\times N}$ correspond à la matrice de compensation de disparité [Scharstein, Szeliski, 2002] et où $v \in \mathbb{R}^N$ désigne le vecteur des erreurs de modélisation. Les observations sont respectivement les versions dégradées

$$z_1 = L_1\overline{x}_1 + w_1 \quad \text{et} \quad z_2 = L_2\overline{x}_2 + w_2 \tag{6.48}$$

de $\overline{x}_1$ et $\overline{x}_2$. Les matrices $L_1 \in \mathbb{R}^{N\times N}$ et $L_2 \in \mathbb{R}^{N\times N}$ modélisent le processus d'acquisition des données, et w_1 et w_2 sont des vecteurs de bruit gaussien avec des composantes indépendantes qui sont respectivement $\mathcal{N}(0,\sigma_1^2)$– et $\mathcal{N}(0,\sigma_2^2)$–distribuées. De plus, nous supposons que les décompositions respectives de $\overline{x}_1$ et $\overline{x}_2$ dans les bases orthonormales $(e_{1,k})_{1\leq k\leq N}$ et $(e_{2,k})_{1\leq k\leq N}$ de $\mathbb{R}^N$ sont parcimonieuses. Pour tout $k \in \{1,\ldots,N\}$, les fonctions $\phi_{1,k} \in \Gamma_0(\mathbb{R})$ et $\phi_{2,k} \in \Gamma_0(\mathbb{R})$ sont utilisées pour promouvoir la parcimonie des décompositions [Combettes, Pesquet, 2007b; Daubechies *et al.*, 2004]. La formulation variationnelle qui suit est cohérente avec les hypothèses et modèles précédents.

Problème 6.18 *Soit $\vartheta \in [0,+\infty[$. L'objectif est de*

$$\underset{x_1\in\mathbb{R}^N,\, x_2\in\mathbb{R}^N}{\text{minimiser}} \quad \sum_{k=1}^{N} \phi_{1,k}(\langle x_1, e_{1,k}\rangle) + \sum_{k=1}^{N} \phi_{2,k}(\langle x_2, e_{2,k}\rangle) + \frac{1}{2\sigma_1^2}\|L_1x_1 - z_1\|^2 + \frac{1}{2\sigma_2^2}\|L_2x_2 - z_2\|^2 + \frac{\vartheta}{2}\|x_1 - Dx_2\|^2. \tag{6.49}$$

Nous pouvons reformuler le problème 6.18 comme un cas particulier du problème 6.1 avec $\mathcal{H}_1 = \mathcal{H}_2 = \mathbb{R}^N$ et $m = 2$ fonctions, qui sont

$$f_1 \colon (x_1, x_2) \mapsto \sum_{k=1}^{N} \phi_{1,k}(\langle x_1, e_{1,k} \rangle) + \sum_{k=1}^{N} \phi_{2,k}(\langle x_2, e_{2,k} \rangle) \tag{6.50}$$

et

$$f_2 \colon (x_1, x_2) \mapsto \frac{1}{2\sigma_1^2}\|L_1 x_1 - z_1\|^2 + \frac{1}{2\sigma_2^2}\|L_2 x_2 - z_2\|^2 + \frac{\vartheta}{2}\|x_1 - D x_2\|^2. \tag{6.51}$$

Proposition 6.19 *Soient x_1 et x_2 des vecteurs arbitraires de $\mathbb{R}^N$. La fonction f_2 est différentiable en (x_1, x_2) et*

$$\nabla f_2(x_1, x_2) = \Big(\frac{1}{\sigma_1^2} L_1^\top (L_1 x_1 - z_1) + \vartheta(x_1 - D x_2), \\ \frac{1}{\sigma_2^2} L_2^\top (L_2 x_2 - z_2) + \vartheta D^\top (D x_2 - x_1)\Big). \tag{6.52}$$

De plus, ∇f_2 est β-Lipschitz continue, avec

$$\beta = \max\{\sigma_1^{-2}\|L_1\|^2, \sigma_2^{-2}\|L_2\|^2\} + \vartheta(1 + \|D\|^2). \tag{6.53}$$

<u>Preuve</u> :
L'expression (6.52) est déduite de calculs directs. On obtient,

$$\boldsymbol{L} = \begin{bmatrix} \sigma_1^{-1} L_1 & [0] \\ [0] & \sigma_2^{-1} L_2 \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{M} = \sqrt{\vartheta} \begin{bmatrix} \mathrm{Id} & -D \\ [0] & [0] \end{bmatrix}. \tag{6.54}$$

En utilisant la notation matricielle, nous pouvons alors écrire

$$\nabla f_2 \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = (\boldsymbol{L}^\top \boldsymbol{L} + \boldsymbol{M}^\top \boldsymbol{M}) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \boldsymbol{L}^\top \begin{bmatrix} \sigma_1^{-1} z_1 \\ \sigma_2^{-1} z_2 \end{bmatrix}. \tag{6.55}$$

Par conséquent, la constante de Lipschitz de ∇f_2 est $\|\boldsymbol{L}^\top \boldsymbol{L} + \boldsymbol{M}^\top \boldsymbol{M}\|$, où $\|\cdot\|$ représente la norme spectrale. Pour obtenir une borne adéquate, nous observons que

$$\begin{aligned} \|\boldsymbol{L}^\top \boldsymbol{L} + \boldsymbol{M}^\top \boldsymbol{M}\| &\leq \|\boldsymbol{L}^\top \boldsymbol{L}\| + \|\boldsymbol{M}^\top \boldsymbol{M}\| \\ &= \|\boldsymbol{L}\|^2 + \|\boldsymbol{M}\|^2 \\ &= \|\boldsymbol{L}\|^2 + \vartheta(1 + \|D\|^2). \end{aligned} \tag{6.56}$$

Maintenant, choisissons $\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^\top$. On a alors

$$\begin{aligned} \|\boldsymbol{L}\boldsymbol{x}\|^2 &= \sigma_1^{-2}\|L_1 x_1\|^2 + \sigma_2^{-2}\|L_2 x_2\|^2 \\ &\leq \sigma_1^{-2}\|L_1\|^2\|x_1\|^2 + \sigma_2^{-2}\|L_2\|^2\|x_2\|^2 \\ &\leq \max\{\sigma_1^{-2}\|L_1\|^2, \sigma_2^{-2}\|L_2\|^2\}\|\boldsymbol{x}\|^2. \end{aligned} \tag{6.57}$$

Par conséquent, $\|\boldsymbol{L}\|^2 \leq \max\{\sigma_1^{-2}\|L_1\|^2, \sigma_2^{-2}\|L_2\|^2\}$ et (6.56) conduit à

$$\|\boldsymbol{L}^\top\boldsymbol{L} + \boldsymbol{M}^\top\boldsymbol{M}\| \leq \beta.$$

□

Au vu de la proposition 6.19, le problème 6.18 peut être résolu à l'aide de l'algorithme explicite-implicte (voir théorème 6.7).

6.5.1-b) Résultats numériques

Les résultats expérimentaux sont présentés sur la figure 6.1 pour des images stéréoscopiques de taille 256×256 ($N = 256^2$). Dans cet exemple, L_1 et L_2 sont des opérateurs de convolution périodique associés à des flous gaussien ayant respectivement des noyaux de tailles 7×7 et 3×3. Ce type de flou a été considéré dans un cadre d'étude similaire dans [Pedone, Heikkilä, 2008]. Un bruit blanc gaussien qui correspond à un BSNR (Blurred Signal-to-Noise-Ratio) de 21.6 dB pour z_1 et 21.8 dB pour z_2 a été ajouté. Les images observées z_i avec $i \in \{1, 2\}$ sont en fait dégradées par un bruit de variance $\sigma_i^2 \equiv 144$ et le BSNR est défini par $10 \log_{10}\left(\|L_i\overline{x}_i\|^2/(N\sigma_i^2)\right)$. De plus, $(e_{1,k})_{1\leq k\leq N}$ et $(e_{2,k})_{1\leq k\leq N}$ désignent des bases d'ondelettes orthogonales de symmlets (longueur 6) sur 2 niveaux de décomposition. Pour les indices $k \in \{1, \ldots, N\}$ qui correspondent aux coefficients de détails, $\phi_{1,k} = \mu_{1,k}|\cdot|$ et $\phi_{2,k} = \mu_{2,k}|\cdot|$, où $\{\mu_{1,k}, \mu_{2,k}\} \subset \,]0, +\infty[$, alors que pour les indices $k \in \{1, \ldots, N\}$ qui désignent les coefficients d'approximation, $\phi_{1,k} = \mu_{1,k}|\cdot|^{3/2}$ et $\phi_{2,k} = \mu_{2,k}|\cdot|^{3/2}$, où $\{\mu_{1,k}, \mu_{2,k}\} \subset \,]0, +\infty[$.

Les opérateurs $(\mathrm{prox}_{\phi_{1,k}})_{1\leq k\leq N}$ et $(\mathrm{prox}_{\phi_{2,k}})_{1\leq k\leq N}$ peuvent être calculés explicitement [Chaux *et al.*, 2007, exemples 4.2 et 4.4]. l'opérateur proximal de f_1 peut alors être déduit de la proposition 6.2, de la séparabilité de cette fonction et de [Combettes, Wajs, 2005, lemmes 2.8 et 2.9]. Les valeurs de $(\mu_{1,k})_{1\leq k\leq N}$ et $(\mu_{2,k})_{1\leq k\leq N}$ sont choisies par une approche de maximum de vraisemblance pour être adaptées à chaque sous-bande. La valeur de ϑ est choisie pour maximiser le rapport-signal-sur-bruit (SNR). Dans nos résultats expérimentaux nous proposons également de comparer les images restaurées en terme de SSIM. La carte de disparité a été estimée en utilisant la méthode décrite dans [Miled *et al.*, 2009]. Notons que l'existence d'une solution du problème 6.18 est assurée par le fait que $f_1 + f_2$ est une fonction coercive de $\Gamma_0(\mathbb{R}^N \oplus \mathbb{R}^N)$ [Combettes, Wajs, 2005, propositions 3.1(i) et 5.15(i)]. Le problème 6.18 est alors un cas particulier du problème 6.6. Dans ce contexte, en choisissant $\lambda_n \equiv 1$, l'algorithme explicite-implicite prend la forme qui suit.

Originale gauche $\overline{x}_1$

Originale droite $\overline{x}_2$

Dégradée gauche z_1
SNR = 12.9 dB – SSIM = 0.39

Dégradée droite z_2
SNR = 18.0 dB – SSIM = 0.56

Restaurée gauche x_1 avec $\vartheta = 0$
SNR = 15.5 dB – SSIM = 0.58

Restaurée droite x_2 avec $\vartheta = 0$
SNR = 19.3 dB – SSIM = 0.73

Restaurée gauche x_1 avec $\vartheta = 1.6 \times 10^{-3}$
SNR = 17.8 dB – SSIM = 0.79

Restaurée droite x_2 avec $\vartheta = 1.6 \times 10^{-3}$
SNR = 19.7 dB – SSIM = 0.83

FIGURE 6.1 – Restauration d'images stéréoscopiques.

Initialisation

$$\sigma_1 = \sigma_2 = 12$$
$$\vartheta = 0 \quad \text{ou} \quad \vartheta = 1.6 \times 10^{-3}$$
$$\gamma = 1.9/\big(\max\{\sigma_1^{-2}\|L_1\|^2, \sigma_2^{-2}\|L_2\|^2\} + \vartheta(1 + \|D\|^2)\big)$$
$$x_{1,0} = z_1$$
$$x_{2,0} = z_2$$

Pour $n = 0, 1, \ldots$

$$y_{1,n} = \sigma_1^{-2} L_1^\top (L_1 x_{1,n} - z_1) + \vartheta(x_{1,n} - D x_{2,n})$$
$$y_{2,n} = \sigma_2^{-2} L_2^\top (L_2 x_{2,n} - z_2) - \vartheta D^\top (x_{1,n} - D x_{2,n})$$
$$x_{1,n+1} = \sum_{k=1}^{N} \big(\mathrm{prox}_{\gamma\phi_{1,k}} \langle x_{1,n} - \gamma y_{1,n}, e_{1,k} \rangle\big) e_{1,k}$$
$$x_{2,n+1} = \sum_{k=1}^{N} \big(\mathrm{prox}_{\gamma\phi_{2,k}} \langle x_{2,n} - \gamma y_{2,n}, e_{2,k} \rangle\big) e_{2,k}$$

Quand $\vartheta = 0$, il n'y a pas de terme de couplage entre les vues de droite et de gauche. Comme on peut l'observer sur la figure 6.1, le terme de couplage conduit à une amélioration significative de la restauration, particulièrement pour l'image la plus dégradée (image en bas à droite).

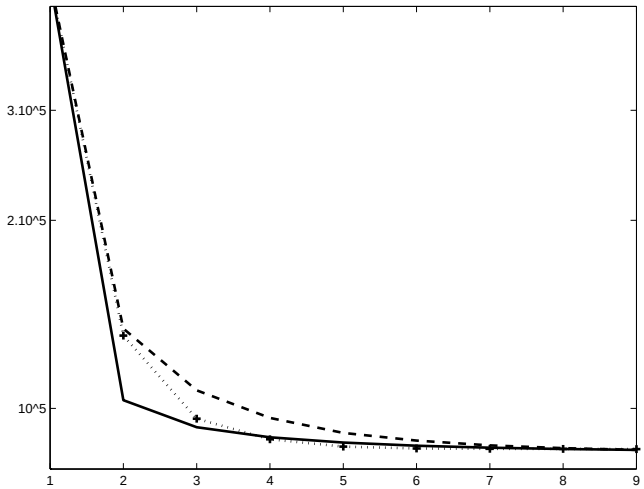


FIGURE 6.2 – Convergence de la fonction objectif du problème 6.18 pour l'algorithme explicite-implicite (ligne continue), l'algorithme de Beck-Teboulle (ligne discontinue) et l'algorithme de Nesterov (ligne en pointillés) en fonction du nombre d'itérations.

Dans le cas où $\vartheta = 1.6 \times 10^{-3}$, nous comparons l'algorithme explicite-implicite, i.e. le théorème 6.7 (implanté avec $\lambda_n \equiv 1$ et $\gamma_n \equiv 1.99/\beta$) aux versions multicomposantes de l'algorithme de Beck-Teboulle [Beck, Teboulle, 2009] et de l'algorithme de Nesterov [Nesterov, 2009]. Contrairement au premier algorithme, les deux autres approches ne garantissent pas théoriquement la convergence des itérées mais sont connues pour fournir des taux de convergence optimaux pour la fonction objectif. Cependant, dans cet exemple, leurs performances sont similaires sur ce point (se référer à la figure 6.2).

6.5.2 Débruitage d'images multispectrales

6.5.2-a) Formulation du problème

Un problème usuel en imagerie multispectrale est de débruiter m images $(\overline{y}_i)_{1\leq i\leq m}$ de $\mathbb{R}^N$ à partir d'observations bruitées $(z_i)_{1\leq i\leq m}$ données par

$$(\forall i \in \{1,\ldots,m\}) \qquad z_i = \overline{y}_i + w_i, \tag{6.58}$$

où $(w_i)_{1\leq i\leq m}$ sont des réalisations mutuellement indépendantes de processus de bruit blanc gaussien à moyenne nulle, de variances respectives $(\sigma_i^2)_{1\leq i\leq m}$. De récentes méthodes de débruitage d'images multispectrales sont décrites dans [Hunt, Kübler, 1984]. Un tutoriel sur le débruitage basé sur des techniques d'ondelettes peut être trouvé dans [Chaux *et al.*, 2010].

Pour résoudre ce problème de débruitage, nous supposons que, pour tout $i \in \{1,\ldots,m\}$, l'image $\overline{y}_i$ satisfait certaines contraintes représentées par un ensemble convexe fermé $C_i \subset \mathbb{R}^N$, et qu'elle admet une décomposition parcimonieuse dans une base orthonormale $(e_{i,k})_{1\leq k\leq N}$ de $\mathbb{R}^N$. De plus, les similarités entre les images sont mises en évidence en pénalisant une distance entre leurs composantes dans une certaine base orthonormale $(b_k)_{1\leq k\leq N}$ de $\mathbb{R}^N$. Ces considérations conduisent au problème variationnel suivant.

$$\underset{y_1\in C_1,\ldots,\, y_m\in C_m}{\text{minimiser}} \sum_{i=1}^{m} \frac{1}{2\sigma_i^2}\|y_i - z_i\|^2 + \sum_{i=1}^{m}\sum_{k=1}^{N} \widetilde{\mu}_{i,k}|\langle y_i, e_{i,k}\rangle| + \sum_{i=1}^{m-1}\sum_{j=i+1}^{m} \widetilde{\vartheta}_{i,j} \sum_{k=1}^{N} |\langle y_i - y_j, b_k\rangle| \tag{6.59}$$

où, pour tout $i \in \{1,\ldots,m\}$, $\{\widetilde{\mu}_{i,k}\}_{1\leq k\leq N} \subset \,]0,+\infty[$ et $\{\widetilde{\vartheta}_{i,j}\}_{i+1\leq j\leq m} \subset \,]0,+\infty[$. Après une remise à l'échelle adéquate des variables, ce problème peut être reformulé comme suit.

Problème 6.20 *Pour tout* $i \in \{1,\ldots,m\}$*, soient* $\{\mu_{i,k}\}_{1\leq k\leq N} \subset \,]0,+\infty[$ *et* $\{\vartheta_{i,j}\}_{i+1\leq j\leq m} \subset \,]0,+\infty[$*. L'objectif est de*

$$\underset{x_1\in\mathbb{R}^N,\ldots,x_m\in\mathbb{R}^N}{\text{minimiser}} \quad \frac{p-1}{2}\sum_{i=1}^{m}\|x_i-\sigma_i^{-1}z_i\|^2+\sum_{i=1}^{m}\sum_{k=1}^{N}\mu_{i,k}\sigma_i|\langle x_i,e_{i,k}\rangle| \\ +\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\vartheta_{i,j}\sum_{k=1}^{N}|\langle \sigma_i x_i-\sigma_j x_j,b_k\rangle|+\sum_{i=1}^{m}\iota_{C_i}(\sigma_i x_i). \quad (6.60)$$

Pour que ce problème entre dans le cadre du problème 6.1, nous définissons

$$J=\big\{(i,j)\in\mathbb{N}^2 \;\big|\; 1\leq i\leq m-1,\; i+1\leq j\leq m\big\} \quad (6.61)$$

et

$$\begin{aligned} \mathrm{i}\colon \quad J &\to \big\{1,\ldots,m(m-1)/2\big\} \\ (i,j) &\mapsto m(i-1)-i(i+1)/2+j. \end{aligned} \quad (6.62)$$

(6.63)

De plus, nous choisissons $p=m(m-1)/2+3$ et

$$\begin{cases} (\forall (i,j)\in J) \quad f_{\mathrm{i}(i,j)}\colon (x_1,\ldots,x_m)\mapsto \vartheta_{i,j}\sum_{k=1}^{N}|\langle \sigma_i x_i-\sigma_j x_j,b_k\rangle| \\ f_{p-2}\colon (x_1,\ldots,x_m)\mapsto \sum_{i=1}^{m}\sum_{k=1}^{N}\mu_{i,k}\sigma_i|\langle x_i,e_{i,k}\rangle| \\ f_{p-1}\colon (x_1,\ldots,x_m)\mapsto \sum_{i=1}^{m}\iota_{C_i}(\sigma_i x_i) \\ f_p\colon (x_1,\ldots,x_m)\mapsto \dfrac{p-1}{2}\sum_{i=1}^{m}\|x_i-\sigma_i^{-1}z_i\|^2. \end{cases} \quad (6.64)$$

Notons que, pour tout $k\in\{1,\ldots,p-2\}$, $\operatorname{dom} f_k=(\mathbb{R}^N)^m$ et $\operatorname{dom} f_{p-1}=\sigma_1^{-1}C_1\times\cdots\times\sigma_m^{-1}C_m$. Par conséquent, puisque les ensembles $(C_i)_{1\leq i\leq m}$ sont non vides, (6.44) est vérifiée et le problème 6.20 peut être résolu avec l'algorithme de type Dykstra présenté par le théorème 6.16, avec $\mathcal{H}_1=\cdots=\mathcal{H}_m=\mathbb{R}^N$. Une forme explicite des opérateurs proximaux des fonctions $(f_k)_{1\leq k\leq m(m-1)/2}$ peut être déduite de la proposition 6.5. Par conséquent, pour tout $(i,j)\in J$, nous pouvons choisir dans cette proposition $\mathcal{H}_1=\cdots=\mathcal{H}_m=\mathcal{G}=\mathbb{R}^N$, puis pour tout $k\in\{1,\ldots,N\}$, $\phi_k=\vartheta_{i,j}|\cdot|$, et enfin définir les matrices

$(L_\ell)_{1\leq\ell\leq m}$ dans $\mathbb{R}^{N\times N}$ comme suit :

$$(\forall \ell \in \{1,\ldots,m\}) \quad L_\ell = \begin{cases} \sigma_\ell \operatorname{Id}, & \text{si } \ \ell = i; \\ -\sigma_\ell \operatorname{Id}, & \text{si } \ \ell = j; \\ 0 & \text{sinon.} \end{cases} \tag{6.65}$$

Finalement, l'opérateur proximal de f_{p-2} peut être déduit de la proposition 6.2 combinée avec la séparabilité de cette fonction, [Combettes, Wajs, 2005, lemme 2.8 du 2.9] et [Chaux *et al.*, 2007, exemple 4.2]. L'opérateur proximal de f_{p-1} est donné par $\operatorname{prox}_{\iota_C} = P_C$.

6.5.2-b) Résultats expérimentaux

La figure 6.4 montre les résultats obtenus sur une image multispectrale de taille 256×256 ($N = 256^2$) pour 3 canaux ($m = 3$) et une dynamique comprise dans l'intervalle $[0, 255]$. Ces images sont dégradées par des bruits blancs gaussien avec des écart-types $\sigma_1 = 11$, $\sigma_2 = 12$, et $\sigma_3 = 13$ (les valeurs correspondantes de SNR sont indiquées sur la figure 6.3). La base $(b_k)_{1\leq k\leq N}$ correspond à une base orthonormale de Haar sur 3 niveaux de décomposition et, pour tout $i \in \{1,\ldots,m\}$, $(e_{i,k})_{1\leq k\leq N}$, représentent des bases orthonormales de symmlet (longueur 6) sur 3 niveaux de décomposition. Les valeurs des paramètres de régularisation $(\mu_{i,k})_{1\leq i\leq 3, 1\leq k\leq N}$ sont adaptés par sous-bande par une approche de maximum de vraisemblance, puis les paramètres de couplage $\vartheta_{1,2}$, $\vartheta_{1,3}$, et $\vartheta_{2,3}$ sont sélectionnés de façon à maximiser le SNR. Pour tout $i \in \{1,\ldots,m\}$, $C_i = [0, 255]^N$ modélise la contrainte sur la dynamique des valeurs des pixels. L'algorithme de type Dykstra résultant est décrit par l'algorithme 15.

On peut observer à partir des images présentées sur la figure 6.4 que l'introduction du terme de couplage a une influence significative sur les performances de débruitage.

Pour ce problème particulier, une méthode alternative est l'algorithme PPXA (voir théorème 6.13) appliqué à la minimisation de la somme de $m(m-1)/2+2$ fonctions f_1, f_2, …, f_{p-2}, et $f_{p-1} + f_p$ est définie par (6.64). L'opérateur proximal de la dernière fonction est donnée par [Combettes, Wajs, 2005, lemme 2.6(i)]. Par conséquent, la condition de qualification (voir (6.43)) se trouve satisfaite car $\operatorname{dom} f_1 = \cdots = \operatorname{dom} f_{p-2} = (\mathbb{R}^N)^m$ et, pour tout $i \in \{1,\ldots,m\}$, $\operatorname{int} C_i = \,]0, 255[$ est un ensemble non vide. Le choix des paramètres de PPXA peut être optimisé empiriquement pour ajuster la vitesse de convergence en choisissant $\lambda_n \equiv 1.3$, $\gamma = 1$, et $\omega_1 = \cdots = \omega_{p-1} = 1/(p-1)$. La figure 6.5 représente l'évolution de $\|\boldsymbol{x}_n - \boldsymbol{x}_\infty\|/\|\boldsymbol{x}_0 - \boldsymbol{x}_\infty\|$ en fonction du temps de calcul, où $(\boldsymbol{x}_n)_{n\in\mathbb{N}} = \big((x_{1,n}, x_{2,n}, x_{3,n})\big)_{n\in\mathbb{N}}$ est une suite générée par l'algorithme et $\boldsymbol{x}_\infty$ est l'unique minimiseur du problème 6.20. Dans nos simulations, 500 itérations ont été utilisées pour obtenir cette solution.

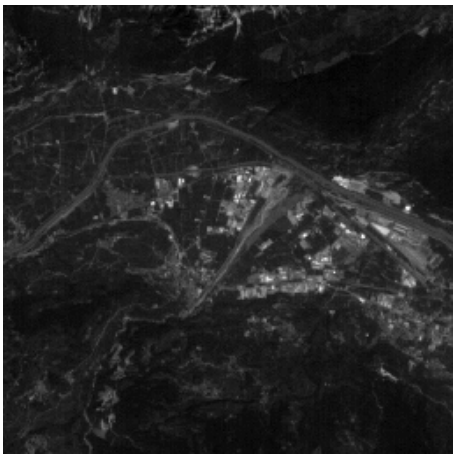


Image originale $\overline{y}_1$

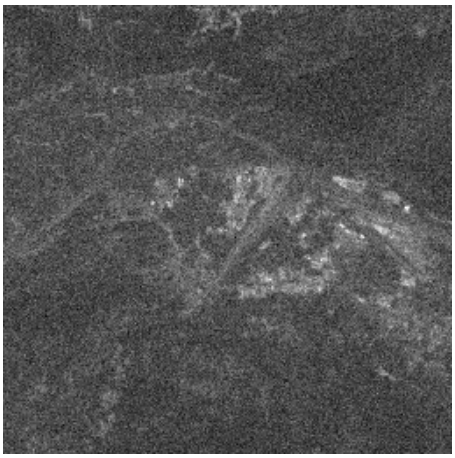


Image dégradée z_1
SNR = 16.2 dB – SSIM = 0.47

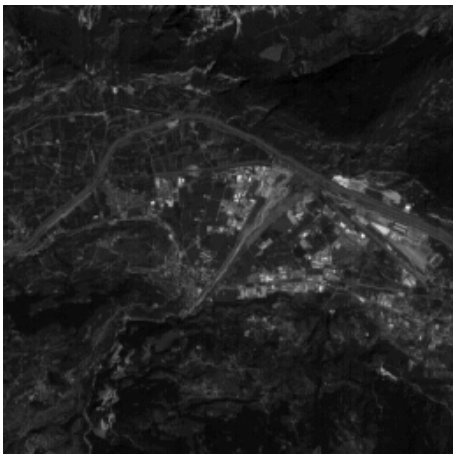


Image originale $\overline{y}_2$

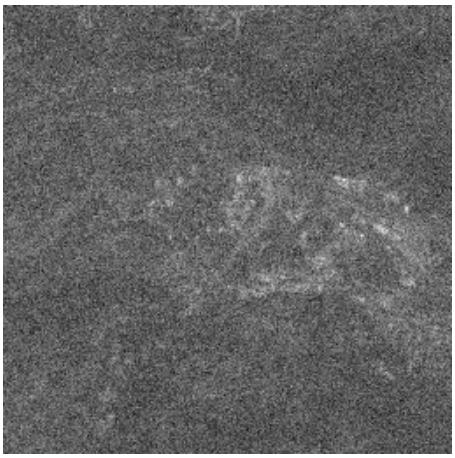


Image dégradée z_2
SNR = 8.28 dB – SSIM = 0.38

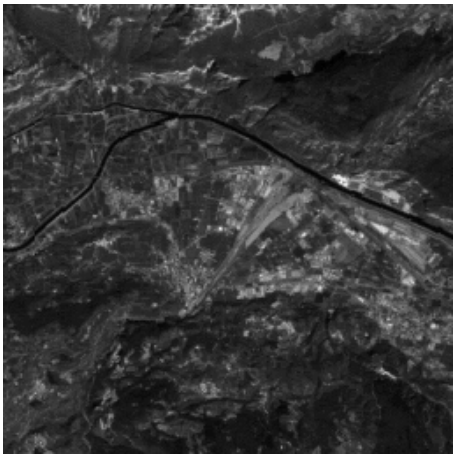


Image originale $\overline{y}_3$

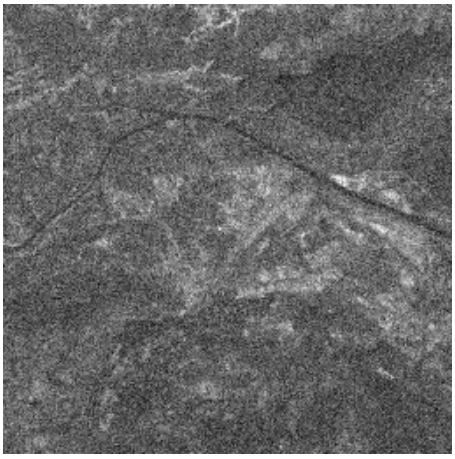


Image dégradée z_3
SNR = 7.08 dB – SSIM = 0.45

FIGURE 6.3 – Débruitage multispectral : images originales et dégradées.

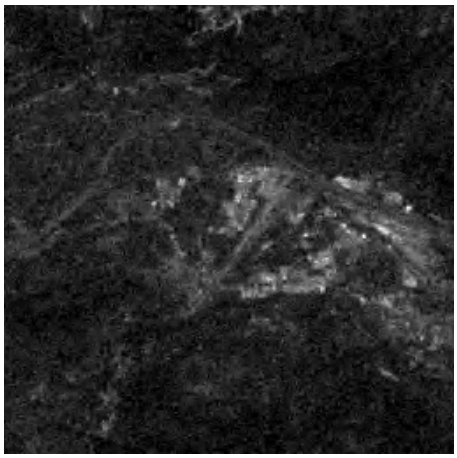


Image restaurée y_1 sans terme de couplage
SNR = 22.3 dB – SSIM = 0.78

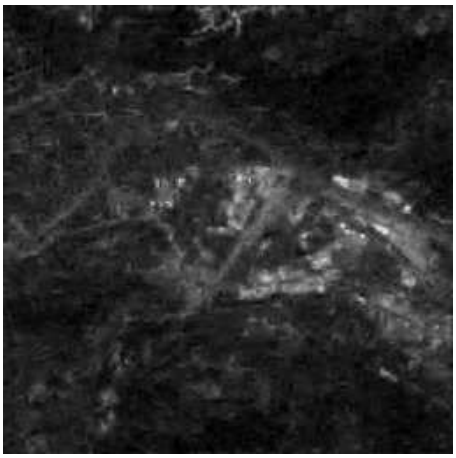


Image restaurée y_1 avec terme de couplage
SNR = 24.2 dB – SSIM = 0.87

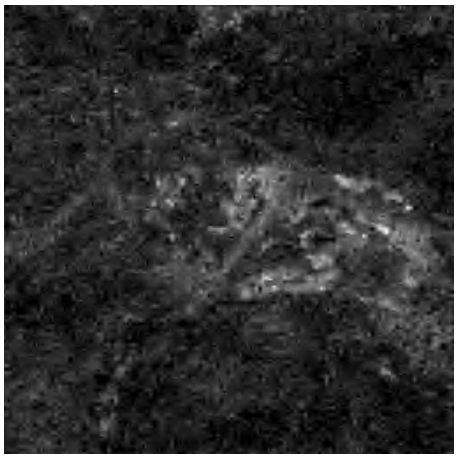


Image restaurée y_2 sans terme de couplage
SNR = 17.4 dB – SSIM = 0.85

Image restaurée y_2 avec terme de couplage
SNR = 19.3 dB – SSIM = 0.91

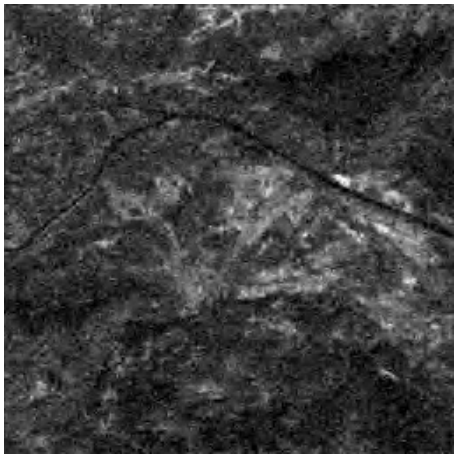


Image restaurée y_3 sans terme de couplage
SNR = 13.2 dB – SSIM = 0.75

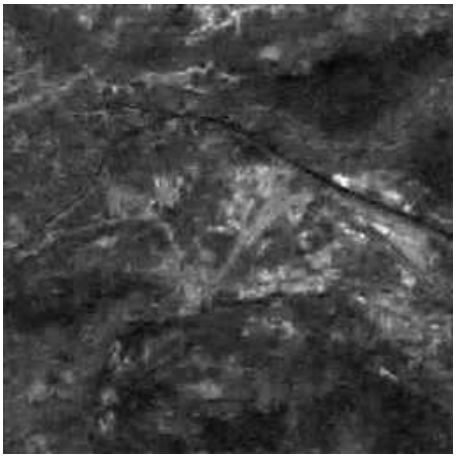


Image restaurée y_3 avec terme de couplage
SNR = 14.7 dB – SSIM = 0.82

FIGURE 6.4 – Débruitage multispectral : images restaurées.

Initialisation

$\sigma_1 = 11\ ;\ \sigma_2 = 12\ ;\ \sigma_3 = 13$

$y_{1,1,0} = \ldots = y_{1,5,0} = x_{1,0} = z_1$

$y_{2,1,0} = \ldots = y_{2,5,0} = x_{2,0} = z_2$

$y_{3,1,0} = \ldots = y_{3,5,0} = x_{3,0} = z_3$

$\alpha_{1,2} = \sigma_1^2 + \sigma_2^2\ ,\ \alpha_{1,3} = \sigma_1^2 + \sigma_3^2\ ,\ \alpha_{2,3} = \sigma_2^2 + \sigma_3^2$

Pour $n = 0, 1, \ldots$

$$u_{1,1,n} = y_{1,1,n} + \alpha_{1,2}^{-1}\sigma_1 \sum_{k=1}^{N} \Big(\mathrm{prox}_{\alpha_{1,2}\vartheta_{1,2}|\cdot|}\big\langle \sigma_1 y_{1,1,n} - \sigma_2 y_{2,1,n}, b_k\big\rangle + \big\langle \sigma_1 y_{1,1,n} - \sigma_2 y_{2,1,n}, b_k\big\rangle\Big) b_k$$

$$u_{2,1,n} = y_{2,1,n} - \alpha_{1,2}^{-1}\sigma_2 \sum_{k=1}^{N} \Big(\mathrm{prox}_{\alpha_{1,2}\vartheta_{1,2}|\cdot|}\big\langle \sigma_1 y_{1,1,n} - \sigma_2 y_{2,1,n}, b_k\big\rangle + \big\langle \sigma_1 y_{1,1,n} - \sigma_2 y_{2,1,n}, b_k\big\rangle\Big) b_k$$

$$u_{3,1,n} = y_{3,1,n}$$

$$u_{1,2,n} = y_{1,2,n} + \alpha_{1,3}^{-1}\sigma_1 \sum_{k=1}^{N} \Big(\mathrm{prox}_{\alpha_{1,3}\vartheta_{1,3}|\cdot|}\big\langle \sigma_1 y_{1,2,n} - \sigma_3 y_{3,2,n}, b_k\big\rangle + \big\langle \sigma_1 y_{1,2,n} - \sigma_3 y_{3,2,n}, b_k\big\rangle\Big) b_k$$

$$u_{2,2,n} = y_{2,2,n}$$

$$u_{3,2,n} = y_{3,2,n} - \alpha_{1,3}^{-1}\sigma_3 \sum_{k=1}^{N} \Big(\mathrm{prox}_{\alpha_{1,3}\vartheta_{1,3}|\cdot|}\big\langle \sigma_1 y_{1,2,n} - \sigma_3 y_{3,2,n}, b_k\big\rangle + \big\langle \sigma_1 y_{1,2,n} - \sigma_3 y_{3,2,n}, b_k\big\rangle\Big) b_k$$

$$u_{1,3,n} = y_{1,3,n}$$

$$u_{2,3,n} = y_{2,3,n} + \alpha_{2,3}^{-1}\sigma_2 \sum_{k=1}^{N} \Big(\mathrm{prox}_{\alpha_{2,3}\vartheta_{2,3}|\cdot|}\big\langle \sigma_2 y_{2,3,n} - \sigma_3 y_{3,3,n}, b_k\big\rangle + \big\langle \sigma_2 y_{2,3,n} - \sigma_3 y_{3,3,n}, b_k\big\rangle\Big) b_k$$

$$u_{3,3,n} = y_{3,3,n} - \alpha_{2,3}^{-1}\sigma_3 \sum_{k=1}^{N} \Big(\mathrm{prox}_{\alpha_{2,3}\vartheta_{2,3}|\cdot|}\big\langle \sigma_2 y_{2,3,n} - \sigma_3 y_{3,3,n}, b_k\big\rangle + \big\langle \sigma_2 y_{2,3,n} - \sigma_3 y_{3,3,n}, b_k\big\rangle\Big) b_k$$

$$u_{1,4,n} = \sum_{k=1}^{N} \big(\mathrm{prox}_{\mu_{1,k}\sigma_1|\cdot|}\big\langle y_{1,4,n}, e_{1,k}\big\rangle\big) e_{1,k}$$

$$u_{2,4,n} = \sum_{k=1}^{N} \big(\mathrm{prox}_{\mu_{2,k}\sigma_2|\cdot|}\big\langle y_{2,4,n}, e_{2,k}\big\rangle\big) e_{2,k}$$

$$u_{3,4,n} = \sum_{k=1}^{N} \big(\mathrm{prox}_{\mu_{3,k}\sigma_3|\cdot|}\big\langle y_{3,4,n}, e_{3,k}\big\rangle\big) e_{3,k}$$

$$u_{1,5,n} = P_{C_1}(\sigma_1\ y_{1,5,n})$$

$$u_{2,5,n} = P_{C_2}(\sigma_2\ y_{2,5,n})$$

$$u_{3,5,n} = P_{C_3}(\sigma_3\ y_{3,5,n})$$

$$x_{1,n+1} = (u_{1,1,n} + u_{1,2,n} + u_{1,3,n} + u_{1,4,n} + u_{1,5,n})/5$$

$$x_{2,n+1} = (u_{2,1,n} + u_{2,2,n} + u_{2,3,n} + u_{2,4,n} + u_{2,5,n})/5$$

$$x_{3,n+1} = (u_{3,1,n} + u_{3,2,n} + u_{3,3,n} + u_{3,4,n} + u_{3,5,n})/5$$

Pour $j = 1, \ldots, 5$

$$y_{1,j,n+1} = x_{1,n+1} + y_{1,j,n} - u_{1,j,n}$$

$$y_{2,j,n+1} = x_{2,n+1} + y_{2,j,n} - u_{2,j,n}$$

$$y_{3,j,n+1} = x_{3,n+1} + y_{3,j,n} - u_{3,j,n}$$

Algorithme 15: Algorithme de débruitage d'images multispectrales.

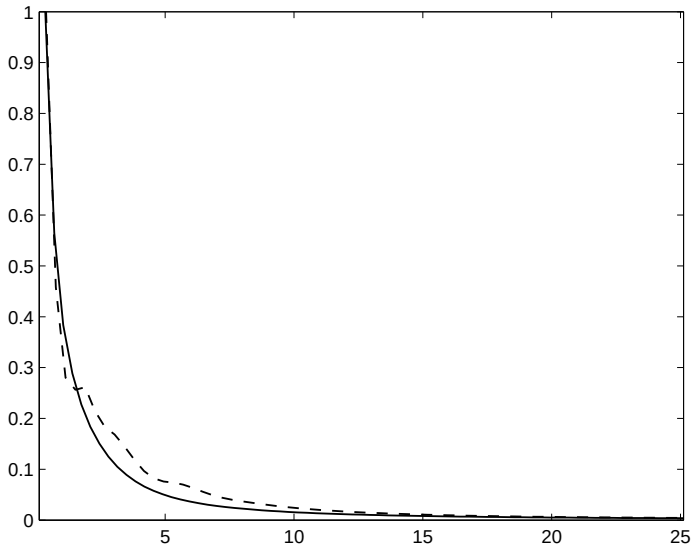


FIGURE 6.5 – Profils de convergence de l'algorithme de type Dykstra (ligne continue) et du PPXA (ligne discontinue) en fonction du temps de calcul en secondes.

6.5.3 Décomposition d'images

Un problème important en traitement d'images est de décomposer une image en structures élémentaires. Dans le contexte du débruitage, cette décomposition a été considérée dans [Rudin *et al.*, 1992] par l'utilisation de la variation totale. Dans [Meyer, 2001], un potentiel différent a été utilisé pour pénaliser de façon plus adéquate les composantes à fortes oscillations. Cependant, la résolution du problème variationnel résultant n'est pas directe. Des méthodes numériques ont été proposées dans [Aujol *et al.*, 2005; Vese, Osher, 2003] ainsi que des résultats expérimentaux dans le cas de débruitage d'images et de problèmes de décomposition en composantes géométrique et de texture. Un autre enjeu est l'extraction de composantes significatives à partir d'une image dégradée par un flou et du bruit. Cette problématique dans le cas particulier d'un bruit additif gaussien est considérée dans [Anthoine *et al.*, 2006; Daubechies, Teschke, 2005]. La méthode développée dans ce paragraphe rend possible cette décomposition en considérant des modèles de bruit généraux (non nécessairement additifs et gaussiens) et des opérateurs de dégradation linéaires arbitraires. Dans la suite nous considérons une simple décomposition en géométrie et texture à partir d'une observation dégradée.

6.5.3-a) Formulation du problème

Dans cette expérience, l'image observée $z \in \mathbb{R}^N$ est obtenue en multipliant l'image originale $\overline{y} \in \mathbb{R}^N$ pour une matrice $A \in \mathbb{R}^{N\times N}$ qui modélise un flou et en dégradant $A\overline{y}$ avec un bruit de Poisson de paramètre d'échelle α. Nous supposons que

$$\begin{aligned} &A \text{ a toutes ses entrées dans } [0,+\infty[\\ &\text{et chacune de ses lignes est non nulle.} \end{aligned} \tag{6.66}$$

Le problème que nous considérons est de décomposer une image $\overline{y} \in \mathbb{R}^N$ en une somme de composantes géométrique et de texture, c'est à dire

$$\overline{y} = R_1(\overline{x}_1) + R_2(\overline{x}_2), \tag{6.67}$$

où $R_1 \colon \mathbb{R}^{N_1} \mapsto \mathbb{R}^N$ et $R_2 \colon \mathbb{R}^{N_2} \mapsto \mathbb{R}^N$ sont des opérateurs connus. Les vecteurs $\overline{x}_1 \in \mathbb{R}^{N_1}$ et $\overline{x}_2 \in \mathbb{R}^{N_2}$ correspondent respectivement aux composantes de géométrie et de texture.

Nous considérons une forme simplifiée de (6.67) employant un mélange linéaire tel que $N_1 = N$, $R_1 \colon x_1 \mapsto x_1$ et $R_2 \colon x_2 \mapsto F^\top x_2$, où $F^\top \in \mathbb{R}^{N\times K}$ est un opérateur de synthèse de trame ajustée. En d'autres termes, l'information qui correspond à la composante de texture se rapporte aux coefficients $\overline{x}_2$ de sa décomposition dans une trame. Rappelons que la condition de trame ajustée implique que

$$F^\top F = \nu \mathrm{Id}, \text{ pour un certain } \nu \in \,]0,+\infty[\,. \tag{6.68}$$

L'image originale est alors décomposée sous la forme $\overline{y} = \overline{x}_1 + F^\top \overline{x}_2$. On sait *a priori* que $\overline{y} \in C_1 \cap C_2$, où

$$C_1 = [0,255]^N \tag{6.69}$$

modélise la contrainte sur la dynamique des pixels de l'image, et

$$C_2 = \left\{ y \in \mathbb{R}^N \;\middle|\; \check{y} = (\eta_k)_{1\leq k\leq N},\ \sum_{k\in\mathbb{I}} |\eta_k|^2 \leq \delta \right\}, \tag{6.70}$$

pour un certain $\delta \in \,]0,+\infty[$, modélise la contrainte d'énergie sur certaines bandes basses-fréquences $\mathbb{I} \subset \{1,\ldots,N\}$ de la transformée de Fourier discrète 2-D de l'image originale, notée $\check{\overline{y}}$. De plus, pour limiter la variation totale de la composante géométrique, le potentiel $x \mapsto \psi(Hx, Vx)$ est utilisé, avec

$$\psi \colon \big((\eta_k)_{1\leq k\leq N}, (\zeta_k)_{1\leq k\leq N}\big) \mapsto \chi \sum_{k=1}^{N} \sqrt{|\eta_k|^2 + |\zeta_k|^2}, \tag{6.71}$$

où $\chi \in \,]0,+\infty[$ et où $H \in \mathbb{R}^{N\times N}$ et $V \in \mathbb{R}^{N\times N}$ sont respectivement des représentations matricielles des opérateurs de dérivations discrètes horizontale et verticale. De plus, pour favoriser la parcimonie dans la trame de la composante de texture de l'image, le potentiel

$$h \colon (\eta_k)_{1\leq k\leq K} \mapsto \sum_{k=1}^{K} \tau_k |\eta_k| \tag{6.72}$$

est introduit, où $\{\tau_k\}_{1\leq k\leq K} \subset \,]0,+\infty[$. Finalement, comme terme d'attache aux données bien adapté au bruit de Poisson, nous utilisons la divergence de Kullback-Leibler généralisée avec un paramètre d'échelle $\alpha \in \,]0,+\infty[$. En définissant $z = (\zeta_k)_{1\leq k\leq N}$, cela conduit à la fonction

$$g\colon (\xi_k)_{1\leq k\leq N} \mapsto \sum_{k=1}^{N} \phi_k(\xi_k), \tag{6.73}$$

où, pour tout $k \in \{1,\ldots,K\}$,

$$\begin{aligned} \phi_k\colon \mathbb{R} &\to \,]-\infty,+\infty] \\ \xi &\mapsto \begin{cases} -\zeta_k \ln(\xi) + \alpha\xi, & \text{si } \zeta_k > 0 \text{ et } \xi > 0; \\ \alpha\xi, & \text{si } \zeta_k = 0 \text{ et } \xi \geq 0; \\ +\infty, & \text{sinon.} \end{cases} \end{aligned} \tag{6.74}$$

Une fois tous ces éléments regroupés, le problème variationnel considéré devient

$$\underset{\substack{x_1\in\mathbb{R}^N,\, x_2\in\mathbb{R}^K \\ x_1+F^\top x_2\in C_1 \\ x_1+F^\top x_2\in C_2}}{\text{minimiser}} \; \psi(Hx_1, Vx_1) + h(x_2) + g(Ax_1 + AF^\top x_2). \tag{6.75}$$

Ce problème est un cas particulier de (6.2) avec $m = 2$, $p = 4$, et

$$\begin{cases} f_1\colon (x_1,x_2) \mapsto \psi(Hx_1, Vx_1) + h(x_2), \\ f_2\colon (x_1,x_2) \mapsto g(Ax_1 + AF^\top x_2), \\ f_3\colon (x_1,x_2) \mapsto \iota_{C_1}(x_1 + F^\top x_2), \\ f_4\colon (x_1,x_2) \mapsto \iota_{C_2}(x_1 + F^\top x_2). \end{cases} \tag{6.76}$$

Cependant, puisque les opérateurs proximaux $(\operatorname{prox}_{f_i})_{1\leq i\leq 4}$ ne sont pas facilement implantables, nous ne pouvons directement appliquer les théorèmes 6.7, 6.10, ou 6.13. Pour contourner ce problème, une stratégie consiste à écrire (6.75) dans une formulation équivalente faisant intervenir des variables supplémentaires.

Une première formulation équivalente à (6.75) est

$$\underset{\substack{x_1,x_2,x_3,x_4,x_5,x_6 \\ x_3=x_1+F^\top x_2 \\ x_3\in C_1\cap C_2 \\ x_4=Ax_3 \\ x_5=Hx_1,\; x_6=Vx_1}}{\text{minimiser}} \; \psi(x_5,x_6) + h(x_2) + g(x_4), \tag{6.77}$$

où nous avons introduit les variables auxilliaires $(x_3,x_4,x_5,x_6) \in \mathbb{R}^N \oplus \mathbb{R}^N \oplus \mathbb{R}^N \oplus \mathbb{R}^N$.

Le problème (6.77) est un cas particulier de (6.2) avec $m = 6$, $p = 3$, et

$$\begin{cases} f_1 \colon (x_1, \ldots, x_6) \mapsto h(x_2) + \iota_{C_1}(x_3) + g(x_4) + \psi(x_5, x_6), \\ f_2 \colon (x_1, \ldots, x_6) \mapsto \iota_{C_2}(x_3), \\ f_3 \colon (x_1, \ldots, x_6) \mapsto \iota_{\{0\}}(x_1 + F^\top x_2 - x_3) + \iota_{\{0\}}(Ax_3 - x_4) + \\ \qquad\qquad\qquad\qquad\qquad\qquad \iota_{\{0\}}(Hx_1 - x_5) + \iota_{\{0\}}(Vx_1 - x_6). \end{cases} \tag{6.78}$$

Dans cette formulation, le rôle de f_3 est d'imposer les contraintes $x_1 + F^\top x_2 = x_3$, $Ax_3 = x_4$, $Hx_1 = x_5$ et $Vx_1 = x_6$. Comme nous l'avons vu précédemment, $\operatorname{prox}_{\iota_{C_1}} = P_{C_1}$ et $\operatorname{prox}_{\iota_{C_2}} = P_{C_2}$. De plus, les opérateurs proximaux de ψ, h, et g peuvent être respectivement obtenus à partir de l'exemple 4.1 (chapitre 4), de l'exemple 3.3 (chapitre 3) et l'exemple 4.2 (chapitre 4). Puisque f_1 est séparable, il en résulte que son opérateur proximal est directement déduit de celui de chacune de ses composantes. Posons maintenant

$$\boldsymbol{L}_1 = \begin{bmatrix} \mathrm{Id} & F^\top & -\mathrm{Id} & [0] & [0] & [0] \\ [0] & [0] & A & -\mathrm{Id} & [0] & [0] \\ H & [0] & [0] & [0] & -\mathrm{Id} & [0] \\ V & [0] & [0] & [0] & [0] & -\mathrm{Id} \end{bmatrix}. \tag{6.79}$$

Il résulte de (6.78) et (6.79) que $f_3 = \iota_{\ker \boldsymbol{L}_1}$, où $\ker \boldsymbol{L}_1 = \big\{ \boldsymbol{x} \in \boldsymbol{\mathcal{H}} \;\big|\; \boldsymbol{L}_1 \boldsymbol{x} = \boldsymbol{0} \big\}$. Puis, en se basant sur le fait que $\operatorname{prox}_{\iota_{\ker \boldsymbol{L}_1}} = P_{\ker \boldsymbol{L}_1}$ et en utilisant [Deutsch, 2001, Chapter 8],

$$\operatorname{prox}_{f_3} = P_{\ker \boldsymbol{L}_1} = \mathrm{Id} - \boldsymbol{L}_1^\top \big(\boldsymbol{L}_1 \boldsymbol{L}_1^\top \big)^{-1} \boldsymbol{L}_1. \tag{6.80}$$

Sous l'hypothèse que les matrices A, H et V sont circulantes avec des blocs circulants, elles sont diagonalisables par transformée de Fourier discrète 2-D. Par conséquent, en combinant (6.80), (6.79) et (6.68) nous en déduisons que $\operatorname{prox}_{f_3}$ se calcule explicitement. De plus, on déduit de (6.78), (6.72), (6.69), (6.73), (6.71), (6.70) et (6.79) que

$$\begin{cases} \operatorname{ri} \operatorname{dom} f_1 & = \mathbb{R}^N \times \mathbb{R}^K \times \operatorname{int} C_1 \times \left]0, +\infty\right[^N \times \mathbb{R}^N \times \mathbb{R}^N \\ \operatorname{ri} \operatorname{dom} f_2 & = \mathbb{R}^N \times \mathbb{R}^K \times \operatorname{int} C_2 \times \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \\ \operatorname{ri} \operatorname{dom} f_3 & = \ker \boldsymbol{L}_1. \end{cases} \tag{6.81}$$

Les conditions de qualification (6.43) sont réduites à

$$\begin{cases} \exists (x_1, \ldots, x_6) \in \ker \boldsymbol{L}_1 \text{ tel que} \\ x_3 \in \operatorname{int} C_1 \cap \operatorname{int} C_2 \text{ et } x_4 \in \left]0, +\infty\right[^N, \end{cases} \tag{6.82}$$

ce qui est équivalent à

$$\begin{cases} \exists (x_1, \ldots, x_6) \in \mathbb{R}^N \times \mathbb{R}^K \times \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^N \\ \text{tel que} \quad x_1 + F^\top x_2 = x_3 \in \operatorname{int} C_1 \cap \operatorname{int} C_2, \\ Ax_3 = x_4 \in \left]0, +\infty\right[^N, Hx_1 = x_5, \text{ et } Vx_1 = x_6. \end{cases} \tag{6.83}$$

Cette condition est satisfaite si

$$A(\operatorname{int}(C_1 \cap C_2)) \cap \,]0,+\infty[^N \neq \varnothing. \tag{6.84}$$

Finalement, soit $y \in A(\operatorname{int}(C_1 \cap C_2)) \cap \,]0,+\infty[^N$. Il existe alors $u \in \operatorname{int}(C_1 \cap C_2)$ tel que $Au = y \in \,]0,+\infty[^N$. Donc, pour tout $x_2 \in \mathbb{R}^K$ si nous choisissons $x_3 = u$, $x_4 = y = Ax_3$, $x_1 = x_3 - F^\top x_2$, $x_5 = Hx_1$ et $Vx_1 = x_6$, nous obtenons que (6.83) est vérifiée. Puisque (6.69) et (6.70) conduisent à $\operatorname{int}(C_1 \cap C_2) \neq \varnothing$, nous déduisons de (6.66) que (6.84) (et du coup (6.43)) est satisfaite. Par conséquent, (6.77) peut être résolu avec PPXA (se référer au théorème 6.13 et à la remarque 6.14).

La seconde formulation équivalente au problème (6.75) est

$$\underset{\substack{x_1,x_2,x_3,x_4,x_5,x_6,x_7 \\ x_3=x_1+F^\top x_2 \\ x_4=Ax_3 \\ x_5=Hx_1,\, x_6=Vx_1 \\ x_7=x_3 \\ x_3\in C_1,\, x_7\in C_2}}{\operatorname{minimiser}} \psi(x_5,x_6) + h(x_2) + g(x_4), \tag{6.85}$$

où $x_7 \in \mathbb{R}^N$ désigne une variable auxiliaire additionnelle. Le problème (6.85) est un cas particulier de (6.2) avec $m = 7$, $p = 2$ et

$$\begin{cases} f_1 \colon (x_1,\ldots,x_7) \mapsto h(x_2) + \iota_{C_1}(x_3) + g(x_4) + \psi(x_5,x_6) + \iota_{C_2}(x_7) \\ f_2 \colon (x_1,\ldots,x_7) \mapsto \iota_{\{0\}}(x_1 + F^\top x_2 - x_3) + \iota_{\{0\}}(Ax_3 - x_4) \\ \qquad\qquad\qquad\qquad + \iota_{\{0\}}(Hx_1 - x_5) + \iota_{\{0\}}(Vx_1 - x_6) + \iota_{\{0\}}(x_3 - x_7). \end{cases} \tag{6.86}$$

Comme précédemment observé, les opérateurs proximaux de ψ, h, g, ι_{C_1} et ι_{C_2} possèdent une forme explicite et il en est donc de même pour $\operatorname{prox}_{f_1}$. De plus, si nous choisissons

$$\boldsymbol{L}_2 = \begin{bmatrix} \mathrm{Id} & F^\top & -\mathrm{Id} & [0] & [0] & [0] & [0] \\ [0] & [0] & A & -\mathrm{Id} & [0] & [0] & [0] \\ H & [0] & [0] & [0] & -\mathrm{Id} & [0] & [0] \\ V & [0] & [0] & [0] & [0] & -\mathrm{Id} & [0] \\ [0] & [0] & \mathrm{Id} & [0] & [0] & [0] & -\mathrm{Id} \end{bmatrix}, \tag{6.87}$$

il peut être déduit de (6.86) que l'opérateur proximal de $f_2 = \iota_{\ker \boldsymbol{L}_2}$ peut être calculé de façon similaire à celui de $\iota_{\ker \boldsymbol{L}_1}$. Nous déduisons de (6.86), (6.72), (6.69), (6.73), (6.71), (6.70) et (6.87) que

$$\begin{cases} \operatorname{ri}\operatorname{dom} f_1 & = \mathbb{R}^N \times \mathbb{R}^K \times \operatorname{int} C_1 \times \,]0,+\infty[^N \times \mathbb{R}^N \times \mathbb{R}^N \times \operatorname{int} C_2 \\ \operatorname{ri}\operatorname{dom} f_2 & = \ker \boldsymbol{L}_2. \end{cases} \tag{6.88}$$

Ainsi, en employant des arguments similaires à ceux proposés plus-haut, (6.40) se réduit à (6.84), qui est alors satisfaite. Cela implique que le critère (6.85) peut être résolu avec l'algorithme de Douglas-Rachford (se référer au théorème 6.10 et à la remarque 6.11(ii)).

6.5.3-b) Résultats expérimentaux

La figure 6.6 montre les résultats de la décomposition en composantes géométrique et de texture pour une image de microscopie électronique de taille 512×512 ($N = 512^2$) dégradée par un flou gaussien de taille 5×5 et du bruit de Poisson ayant comme paramètre d'échelle $\alpha = 0.6$. Le paramètre χ de (6.71) et les paramètres $(\tau_k)_{1 \leq k \leq K}$ de (6.72) sont choisis de manière à maximiser le SNR. La matrice F est une version en trames ajustées de la transformée en arbre dual proposée dans [Chaux *et al.*, 2006] reposant sur des symmlets de longueur 6 sur 3 niveaux de décomposition ($\nu = 2$ and $K = 2N$). Les mêmes matrices de gradient discrétisé H et V que celles utilisées dans [Chambolle, 2004] sont considérées. Pour cette application, nous comparons PPXA et l'algorithme de Douglas-Rachford en terme de convergence. Les paramètres choisis sont les suivants : $\lambda_n \equiv 1$, $\gamma = 100$, et les poids pour PPXA sont $\omega_1 = \omega_2 = \omega_3 = 1/3$. Dans ce contexte, la forme prise par PPXA est la suivante.

Initialisation
- $\gamma = 100$
- $(y_{1,1,0}, \ldots, y_{6,1,0}) = (z, F^\top z, z, z, z, z)$
- $(y_{1,2,0}, \ldots, y_{6,2,0}) = (z, F^\top z, z, z, z, z)$
- $(y_{1,3,0}, \ldots, y_{6,3,0}) = (z, F^\top z, z, z, z, z)$
- Pour $i = 1, \ldots, 6$
  - $x_{i,0} = (y_{i,1,0} + y_{i,2,0} + y_{i,3,0})/3$

Pour $n = 0, 1, \ldots$
- $(u_{1,1,n}, u_{2,1,n}, u_{3,1,n}, u_{4,1,n}) = (y_{1,1,n}, \mathrm{prox}_{3\gamma h}(y_{2,1,n}), P_{C_1}(y_{3,1,n}), \mathrm{prox}_{3\gamma g}(y_{4,1,n}))$
- $(u_{5,1,n}, u_{6,1,n}) = \mathrm{prox}_{3\gamma\psi}(y_{5,1,n}, y_{6,1,n})$
- $(u_{1,2,n}, u_{2,2,n}, u_{3,2,n}, u_{4,2,n}, u_{5,2,n}, u_{6,2,n})$
  $= (y_{1,2,n}, y_{2,2,n}, P_{C_2}(y_{3,2,n}), y_{4,2,n}, y_{5,2,n}, y_{6,2,n})$
- $(u_{1,3,n}, \ldots, u_{6,3,n}) = P_{\ker \boldsymbol{L}_1}(y_{1,3,n}, \ldots, y_{6,3,n})$
- Pour $i = 1, \ldots, 6$
  - $s_{i,n} = (1/3) \sum_{k=1}^{3} u_{i,k,n}$
  - $y_{i,1,n+1} = y_{i,1,n} + 2s_{i,n} - x_{i,n} - u_{i,1,n}$
  - $y_{i,2,n+1} = y_{i,2,n} + 2s_{i,n} - x_{i,n} - u_{i,2,n}$
  - $y_{i,3,n+1} = y_{i,3,n} + 2s_{i,n} - x_{i,n} - u_{i,3,n}$
  - $x_{i,n+1} = x_{i,n} + s_{i,n} - x_{i,n}$

et l'algorithme de Douglas-Rachford s'écrit :

Initialisation
$\gamma = 100$
$(x_{1,0}, \ldots, x_{7,0}) = (z, F^\top z, z, z, z, z, z)$

Pour $n = 0, 1, \ldots$
$(y_{1,n}, y_{2,n}, y_{3,n}, y_{4,n}) = (x_{1,n}, \mathrm{prox}_{\gamma h}(x_{2,n}), P_{C_1}(x_{3,n}), \mathrm{prox}_{\gamma g}(x_{4,n}))$
$(y_{5,n}, y_{6,n}) = \mathrm{prox}_{\gamma\psi}(x_{5,n}, x_{6,n})$
$y_{7,n} = P_{C_2}(x_{7,n})$

$$(u_{1,n}, \ldots, u_{7,n}) = P_{\ker \boldsymbol{L}_2}\Big(2(y_{1,n}, \ldots, y_{7,n}) - (x_{1,n}, \ldots, x_{7,n})\Big)$$

Pour $i = 1, \ldots, 7$
$x_{i,n+1} = x_{i,n} + u_{i,n} - y_{i,n}$

Dans la figure 6.7, la valeur de $\|\boldsymbol{y}_n - \boldsymbol{y}_\infty\|/\|\boldsymbol{y}_0 - \boldsymbol{y}_\infty\|$ pour la suite $(\boldsymbol{y}_n)_{n\in\mathbb{N}} = \big((y_{1,n}, \ldots, y_{7,n})\big)_{n\in\mathbb{N}}$ générée par le théorème 6.10 et $\|\boldsymbol{x}_n - \boldsymbol{x}_\infty\|/\|\boldsymbol{x}_0 - \boldsymbol{x}_\infty\|$ pour la suite $(\boldsymbol{x}_n)_{n\in\mathbb{N}} = \big((x_{1,n}, \ldots, x_{6,n})\big)_{n\in\mathbb{N}}$ générée par le théorème 6.13 (où $\boldsymbol{y}_\infty$ et $\boldsymbol{x}_\infty$ correspondent aux solutions respectives) sont tracées en fonction du temps de calcul en secondes. Dans nos simulations, 1000 itérations sont utilisées pour produire une solution.

Dans le tableau 6.1, nous comparons les images restaurées par la méthode proposée dans ce chapitre (permettant d'extraire les composantes de texture et géométrique mais également d'obtenir une image restaurée) et par la méthode proposée dans le chapitre 4 basée sur la régularisation hybride. Nous avons fixé l'ensemble des paramètres de régularisation de façon à maximiser les SNR des images restaurées. Les tests sont effectuées sur deux images [1], présentées sur la figure 6.8, qui ont été dégradées par un opérateur de convolution (flou uniforme de taille $Q = 3 \times 3$ ou $Q = 5 \times 5$) et un bruit de Poisson de facteur d'échelle $\alpha = 0.5$ ou $\alpha = 0.1$.

1. source : http ://remf.dartmouth.edu/

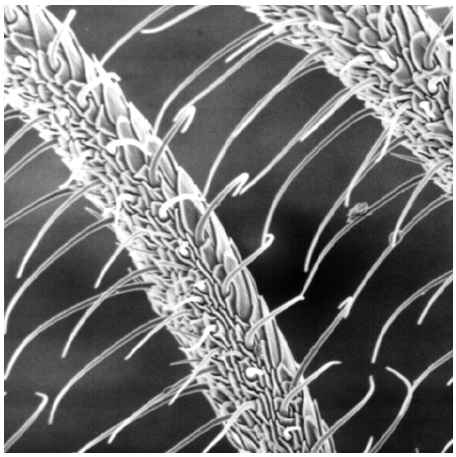


Image originale $\overline{y}$

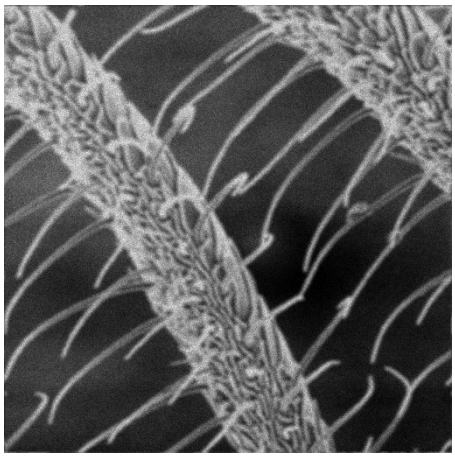
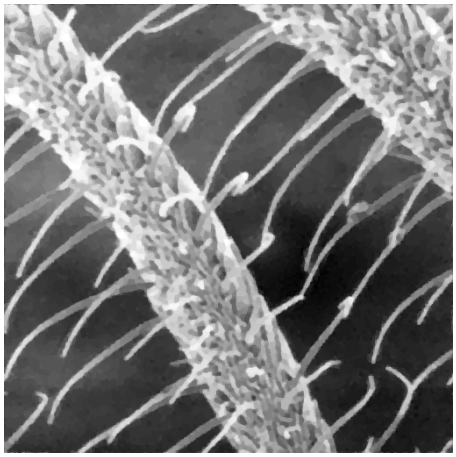
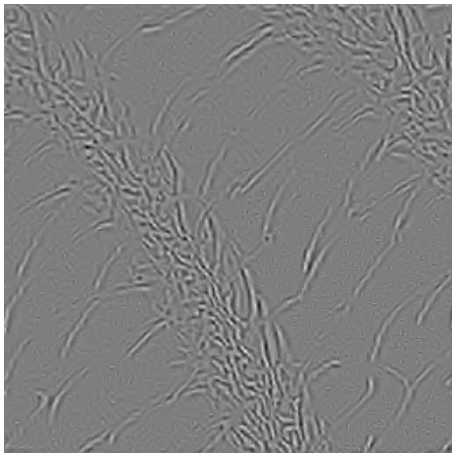


Image dégradée z
SNR = 14.2 dB – SSIM = 0.74.



Composante géométrique x_1.



Composante de texture $F^\top x_2$.

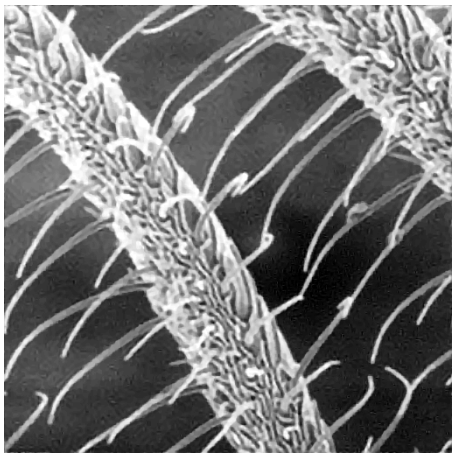


Image restaurée x_3
SNR = 17.7 dB – SSIM = 0.86.

FIGURE 6.6 – Résultats de décomposition et de restauration.

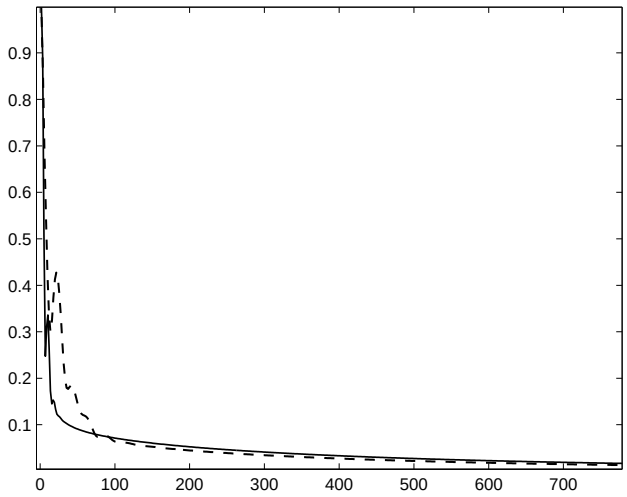


FIGURE 6.7 – Profils de convergence de l'algorithme Douglas-Rachford (ligne continue) et du PPXA (line discontinue) en fonction du temps de calcul en secondes.

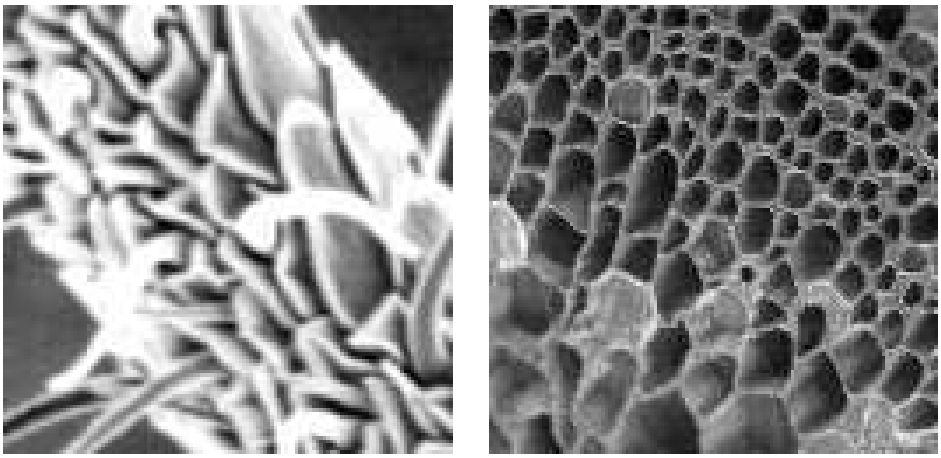


FIGURE 6.8 – Images « Culicoïdae » et « Sunflower » .

		$\alpha = 0.5$		$\alpha = 0.1$	
		$Q = 3 \times 3$	$Q = 5 \times 5$	$Q = 3 \times 3$	$Q = 5 \times 5$
Culicoïdae	PPXA Hybride	21.9	18.9	18.4	16.5
	PPXA Text+Geom	21.6	18.8	18.1	16.4
Sunflower	PPXA Hybride	15.6	13.4	13.5	12.0
	PPXA Text+Geom	15.0	13.2	13.3	12.0

TABLE 6.1 – Comparaison en terme de SNR (en dB) des images restaurées par la méthode proposée dans le chapitre 4, nommée ici « PPXA Hybride », et la méthode présentée dans ce chapitre, nommée « PPXA Text+Geom ».

6.6 Conclusion

Dans ce chapitre, le formalisme proximal a été appliqué au traitement d'images multicomposantes. Des formes explicites de nouveaux opérateurs proximaux dans un espace produit ont été proposées. Nous avons illustré l'intérêt des modèles multicomposantes par le biais de trois applications différentes : la stéréoscopie, l'imagerie multispectrale et la décomposition d'images en composantes géométrique et de texture. Un autre champ d'application pour lequel ces techniques pourraient apparaître utiles est le traitement des images couleurs. Le formalisme proximal ainsi établi peut également être utilisé pour obtenir des algorithmes applicables à des problèmes de traitement du signal et des images complexes en notant qu'un signal complexe est un signal avec $m = 2$ composantes réelles, qui sont respectivement ses parties réelle et imaginaire.

CHAPITRE 7

Algorithmes proximaux pour la reconstruction de données dynamiques en TEP

7.1 Introduction à la TEP

7.1.1 Traceur radioactif

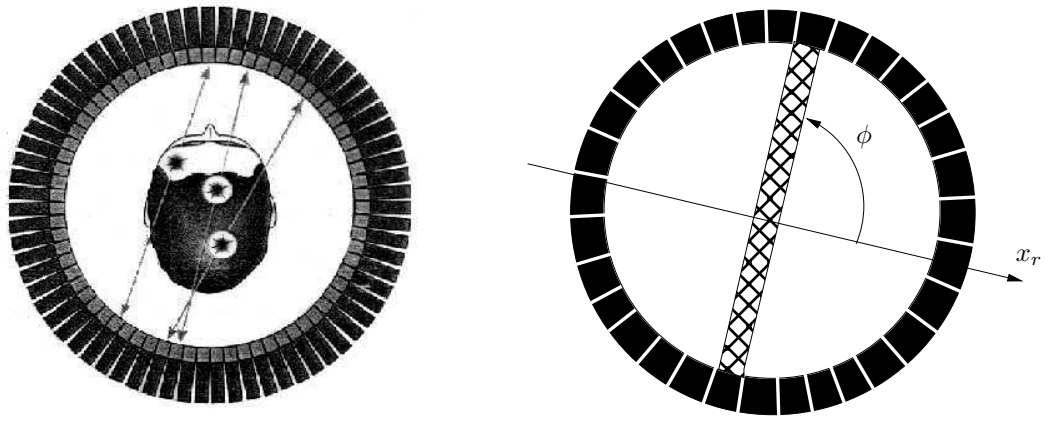
La Tomographie par Emission de Positrons (TEP) est une modalité d'imagerie médicale de médecine nucléaire. Il s'agit d'une imagerie par émission, contrairement à l'imagerie radiologique qui est une imagerie par transmission. La différence fondamentale entre ces deux modalités provient du sens du rayonnement. Dans le cas de la TEP, le rayonnement provient du patient, après l'injection d'un traceur radioactif, alors que la seconde modalité crée un faisceau externe qui traverse le patient.

La TEP permet de mesurer l'activité métabolique d'un organe par le biais d'un traceur radioactif émetteur de positrons. Ce produit est l'association d'une molécule vectrice et d'un marqueur radioactif. La molécule vectrice se fixe sur certaines structures de l'organe cible et le marqueur permet de localiser la position de la molécule dans l'organisme. Un exemple de traceur est le Fluorodesoxyglucose marqué au Fluor 18 ([^{18}F]-FDG). Il se caractérise par un marqueur ^{18}F incorporé dans un analogue du glucose. Ce traceur se fixe sur les tissus ayant un métabolisme de glucose élevé, comme les tissus cancéreux. Dans la molécule, un atome est remplacé par son isotope radioactif qui, en se désintégrant [1] par voie β^+ , émet un positron. Un positron est l'anti-particule associée à l'électron, tous deux possèdent donc la même masse mais des charges électriques opposées (positive pour le positron et négative pour l'électron). Dans la matière, les positrons vont perdre leur énergie cinétique et, au repos, chacun des positrons va s'annihiler avec un électron. La masse de ces deux particules est convertie en énergie sous forme de 2 photons gamma.

1. Transformation d'un proton en neutron permettant à la molécule de passer d'un niveau instable à un niveau stable.

7.1.2 Détection des coïcidences

Les deux photons issus de l'annihilation d'un positron avec un électron sont émis à 180° l'un de l'autre. Voyageant à la vitesse de la lumière, ils vont être détectés par une couronne de détecteurs, située autour du patient, presque simultanément ($\pm$ quelques nanosecondes). Le parcours des photons est illustré sur la figure 7.1(a) pour trois annihilations distinctes. On considère que deux photons proviennent d'une même annihilation s'ils sont détectés pendant un intervalle de temps de l'ordre de quelques nanosecondes, appelé fenêtre de coïncidences. Les deux détecteurs forment un tube de réponse (TOR) illustré figure 7.1(b) qui permet une localisation de la zone d'annihilation (quelque part à l'intérieur du tube). À ce stade, on peut remarquer que pour pour une injection de 2.10^8 Bq (*i.e.* désintégrations par seconde), des centaines de milliards de désintégrations se produisent dans tout l'organisme mais seulement quelques centaines de millions sont détectées. En particulier, des photons émis dans la direction opposée du plan de coupe de la figure 7.1(a) ne seront pas détectés.



(a) (b)

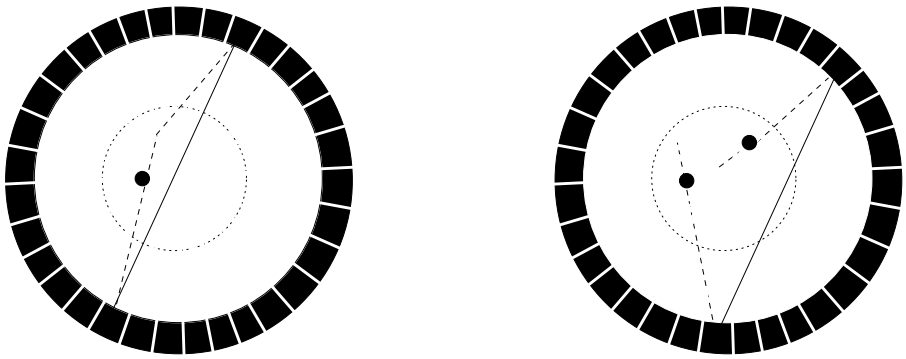
FIGURE 7.1 – (a) Principe de détection des coïncidences et (b) tube de réponse.

En réalité, des dixaines d'anneaux (couronnes de détecteurs) sont empilées. On parle de TEP 3D lorsque l'on ne peut pas décomposer le problème comme un ensemble de problèmes 2D. Dans la suite nous décrivons le fonctionnement de la TEP dans le cas 2D. Deux formats d'enregistrement des détections existent : le format sinogramme ou le mode évènementiel. Le sinogramme est un histogramme représentant le nombre de détections pour chaque tube de réponse (angle azimutal ϕ + position radiale x_r). Le mode évènementiel enregistre individuellement chaque détection en coïncidence, à savoir l'adresse

des deux détecteurs et la datation des deux détections. Dans ce chapitre, notre intérêt portera sur le mode sinogramme.

Lorsque les photons traversent la matière, des perturbations interviennent [Valk *et al.*, 2004]. Il s'agit de phénomènes distincts :

- l'atténuation apparaît lorsque les photons traversent la matière et perdent de l'énergie. Les photons peuvent être absorbés et ne jamais sortir de l'organisme.
- Les coïncidences diffusées, illustrées figure 7.2(a), apparaissent lorsque le passage par la matière modifie le parcours des photons. Leurs trajets n'étant plus rectiligne, un biais peut être introduit et le lieu de l'annihilation ne se situe plus le long du TOR.
- Les coïncidences multiples se produisent dans le cas de figure où plus de deux photons sont détectés dans une fenêtre de coïncidences.
- Les coïncidences fortuites, illustrées figure 7.2(b), se produisent lorsque les photons sont détectés dans la même fenêtre de coïncidences mais proviennent de deux annihilations différentes.



(a) (b)

FIGURE 7.2 – (a) Coïncidences diffusées et (b) coïncidences fortuites.

Les doses d'activité injectées sont très faibles, ce qui se traduit par un faible nombre d'évènements détectés et donc des données très bruitées. Nous allons voir que le bruit présent est un bruit de Poisson. Le rapport signal sur bruit est donc proportionnel à la racine carrée du nombre d'évènements détectés.

7.1.3 Modèle de dégradation spatio-temporel

Sous l'hypothèse que le nombre d'évènements suit une loi de Poisson et que le processus de détection est poissonien, nous pouvons considérer le modèle direct de dégradation suivant :

$$(\forall t \in \{1, \ldots, T\}) \qquad z_t = \mathcal{D}(A\overline{y}_t + f_t + d_t), \tag{7.1}$$

où l'effet du bruit de Poisson est représenté par $\mathcal{D}$. $\overline{y}_t = (\overline{y}_{i,t})_{1 \leq i \leq N}$ représente une paramétrisation finie, composée de N éléments nommés « voxels », de la distribution spatiale de la concentration radioactive pour une trame temporelle t (une partie de cette paramétrisation est illustrée par la figure 7.3). Dans la suite nous nous réfèrerons à cette distribution en parlant d'activité spatiale dans les tissus. Son unité de mesure est Bq/cc. La paramétrisation finie $z_t = (z_{j,t})_{1 \leq j \leq M}$ modélise les données TEP dynamiques et correspond au nombre de coïncidences détectées par chaque tube de réponse durant une trame-temporelle t. Notons que dans notre modèle, $(z_t)_{t \in \{1,\ldots,T\}}$ désigne un ensemble de sinogrammes, chacun composé de M éléments, nommés « bins », (M = nb. d'angles azimutaux$\times$nb. de positions radiales en 2D et M = nb. d'angles azimutaux$\times$nb. de positions radiales$\times$(nb. d'anneaux)^2 en 3D). L'effet des coïncidences diffusées et fortuites est modélisé par des termes additifs notés, respectivement, $(d_t)_{t \in \{1,\ldots,T\}}$ et $(f_t)_{t \in \{1,\ldots,T\}}$. Notons que ces effets seront uniquement pris en compte lors du traitement de données réelles. La matrice $A = (A_{j,i})_{1 \leq j \leq M, 1 \leq i \leq N}$ désigne la matrice système ; elle correspond à l'implantation numérique de l'opération de projection. Chaque élément $A_{j,i}$ représente la probabilité qu'une coïncidence produite au niveau du voxel i soit détectée par le tube de réponse j. Un exemple de calcul de la matrice A est illustré par la figure 7.3. D'autres calculs envisageables sont, par exemple, basés sur la ligne d'intersection ou une mesure d'angle solide [Valk *et al.*, 2004]. Dans le cas de données réelles, la matrice A inclut les facteurs d'atténuation des photons dans la matière.

Dans ce contexte, notre objectif est de trouver une paramétrisation finie $(\widehat{y}_t)_{t \in \{1,\ldots,T\}}$ au plus proche de la paramétrisation originale $(\overline{y}_t)_{t \in \{1,\ldots,T\}}$, en utilisant l'information contenue dans l'ensemble des sinogrammes temporels $(z_t)_{t \in \{1,\ldots,T\}}$.

Parmi les enjeux liés à la reconstruction dynamique TEP, on peut citer l'extraction de paramètres physiologiques. Pour cela il est nécessaire de connaître le modèle compartimental faisant le lien entre la mesure dans les tissus $(\widehat{y}_t)_{t \in \{1,\ldots,T\}}$ et les paramètres physiologiques. Nous présentons dans la suite le modèle existant dans le cas d'un examen [^{18}F]-FDG en imagerie cérébrale.

7.1.4 Principe compartimental dans le cas du [^{18}F]-FDG

L'imagerie TEP du métabolisme du glucose est largement utilisée en oncologie et en neurologie. En oncologie, l'un des objectifs est de localiser les cellules cancereuses et ainsi de procéder au suivi de patients sous traitement oncologiques. En neurologie, la mesure de métabolisme de glucose permet de mettre en évidence un déficit fonctionnel

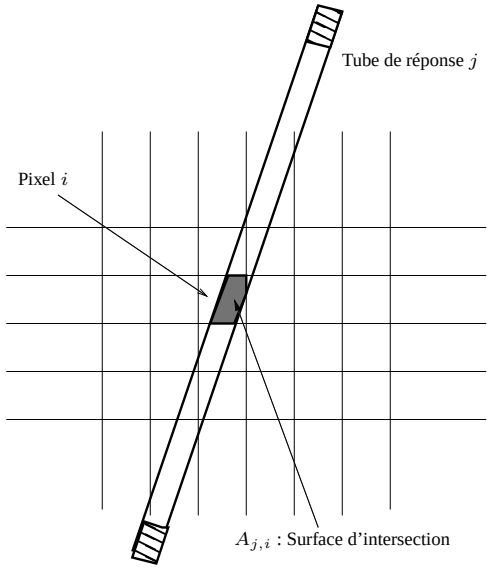


FIGURE 7.3 – Illustration d'un calcul possible des valeurs $A_{j,i}$ de la matrice système en 2D. La grille arrière correspond à une partie de la paramétrisation finie de l'activité.

du cerveau. Dans l'examen [^{18}F]-FDG, le fluorodeoxyglucose (FDG) est substitué au glucose. Nous verrons plus bas, que cet analogue du glucose permet de faciliter l'extraction de cartes paramétriques de la mesure du métabolisme. De façon à mieux appréhender la signification de cette mesure, la circulation du FDG dans le cerveau est décrite par le modèle tri-compartimental développé par Sokoloff, illustré figure 7.4. Les trois compartiments sont :

- le compartiment plasmatique artériel qui modélise le FDG dans le plasma des capillaires du cerveau. On désigne par C_a l'évolution de la concentration de traceur dans le sang artériel (plasma).
- le compartiment tissulaire précurseur qui correspond à l'état du FDG libre dans les tissus. La concentration du traceur dans ce compartiment est notée C_1.
- le compartiment tissulaire qui représente l'état du FDG dans les tissus après phosphorylation par hexokynase (il est noté FDG-6-P). La concentration radiocative dans ce compartiment est notée C_2.

Il faut remarquer qu'au niveau du voxel, on ne distingue pas C_1 de C_2 mais on observe la somme de ces deux concentrations.

Le FDG est transporté entre le plasma et les tissus en se fixant sur des enzymes (GLUT). Une zone de délimitation plasma-tissu, appelée barrière hémato-encéphalique (BBB : Blood Brain Barrier), permet d'éviter de laisser passer les impuretés dans le tissu cérébral. Le taux de transfert de molécules de FDG à travers la BBB est noté K_1. Son unité de mesure correspond à une concentration de molécules de FDG transférée par minute. k_2 est le taux de transfert à travers la BBB dans le sens inverse (tissu cérébral vers plasma).

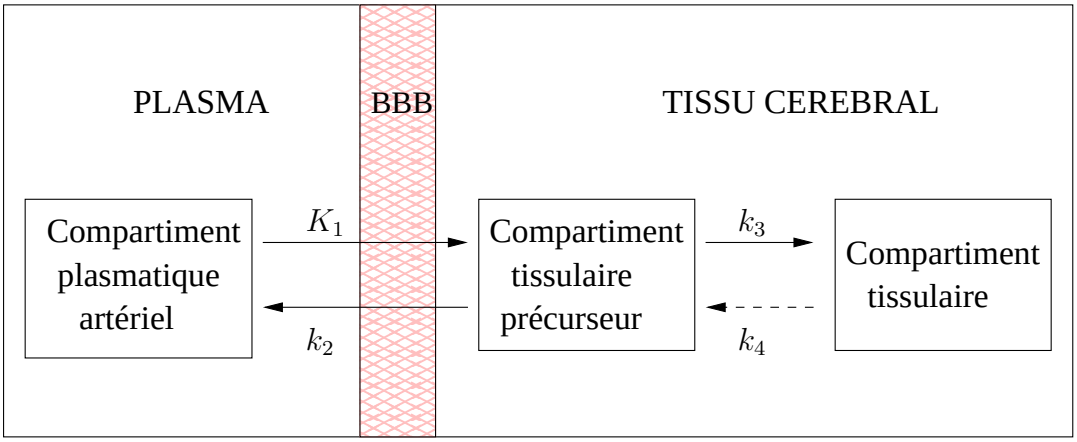


FIGURE 7.4 – Modèle tri-compartimental de Sokoloff

k_3 correspond à la phosphorylation de FDG par hexokynase. k_4 permet de prendre en compte la déphosphorylation. Si l'examen n'exède pas 1 heure post-injection, la déphosphorylation n'est pas observée [Lucignani *et al.*, 1993]. Cette hypothèse d'irréversibilité se traduit par $k_4 = 0$. Il faut noter que contrairement au FDG, le glucose se métabolise il n'y a donc ni accumulation du FDG-6-P dans les tissus ni déphosphorylation. Le modèle compartimental dans le cas du glucose est donc beaucoup plus complexe.

La formulation mathématique du modèle compartimental se base sur des équations différentielles qui font le lien entre les concentrations d'activité dans les différents compartiments et les paramètres physiologiques. Pour chaque compartiment du tissu cérébral, le flux net peut être défini comme la somme de toutes les entrées moins la somme de toutes le sorties. On peut alors écrire :

$$\begin{cases} \frac{dC_1}{dt} & = K_1 C_a - k_3 C_1 - k_2 C_1 + k_4 C_2, \\ \frac{dC_2}{dt} & = k_3 C_1 - k_4 C_2. \end{cases} \tag{7.2}$$

En utilisant l'hypothèse d'irréversibilité ($k_4 = 0$), l'équation (7.2) s'écrit

$$\begin{cases} \frac{dC_1}{dt} & = K_1 C_a - k_3 C_1 - k_2 C_1, \\ \frac{dC_2}{dt} & = k_3 C_1. \end{cases} \tag{7.3}$$

L'activité dans les tissus correspond à la somme de la concentration dans le compartiment tissulaire précurseur C_1 et de la concentration dans le compartiment tissulaire C_2. La connaissance de l'activité dans les tissus passe donc par la résolution des équations différentielles (7.3). L'activité peut alors s'exprimer comme suit (cf. [Dittrich *et al.*, 2008, chapitre 5]) :

$$\forall t > 0 \qquad C(t) = C_a(t) \otimes \Big(\frac{K_1 k_2}{(k_2 + k_3)^2} (1 - \exp(-(k_2 + k_3)t) + \frac{K_1 k_3}{k_2 + k_3} t \Big). \tag{7.4}$$

où $\otimes$ désigne l'opérateur de convolution.

L'approche PATLAK [Gjedde, 1982; Patlak *et al.*, 1983] est une procédure de linéarisation qui permet de décrire le comportement du modèle FDG quand le FDG libre dans les tissus a atteint un état pour lequel le ratio des concentrations C/C_a devient linéaire. La principale hypothèse consiste à supposer que la concentration C_a devient constante. L'équation (7.4) s'écrit alors :

$$\forall t > 0 \qquad C(t) = C_a\Big(\frac{K_1 k_2}{(k_2+k_3)^2}(1-\exp(-(k_2+k_3)t) + \frac{K_1 k_3}{k_2+k_3}t\Big).$$

Cette hypothèse est en réalité approximativement vérifiée car les variations de C_a sont très faibles devant la constante de temps $1/(k_2+k_3)$. D'autre part, on remarque qu'après un certain temps t^* d'examen, le terme exponentiel devient négligeable et le ratio entre l'activité dans les tissus et l'activité dans le sang artériel suit une relation linéaire :

$$\forall t \geq t^* \qquad \frac{C(t)}{C_a} = \frac{K_1 k_2}{(k_2+k_3)^2} + \frac{K_1 k_3}{k_2+k_3}t. \tag{7.5}$$

Une simple analyse graphique nous permet alors de connaître le paramètre

$$K = \frac{K_1 k_3}{k_2+k_3}.$$

En pratique, $C(t)$ correspond à la paramétrisation finie de l'activité reconstruite $\widehat{y}_t$ et C_a est connue car des prélèvements de la concentration de traceur dans le sang artériel sont effectués tout au long de l'examen. Le métabolisme de glucose dans une région du cerveau, noté $r\mathrm{CMRglu}$, correspond à :

$$\mathrm{rCMRglu} = \frac{C_p}{LC}K, \tag{7.6}$$

où C_p désigne la concentration de glucose dans le plasma. La valeur de C_pK donnerait le taux de phosphorylation du glucose si le glucose et le FDG avait le même comportement. Pour prendre en compte cette différence, la constante LC (« lumped constant ») intervient dans l'équation (7.6).

L'intérêt du médecin portera sur les valeurs de ces constantes. Par exemple, comme le $\mathrm{rCMRglu}$ dépend linéairement de K_1, une valeur de rCMRglu faible traduira un mauvais transfert des molécules à travers la BBB. L'enjeu de notre étude consiste à se rapprocher au plus de l'activité mesurée dans les tissus à un instant t pour ensuite obtenir une valeur de $\mathrm{rCMRglu}$ la plus précise possible tant au niveau de la valeur que pour la finesse de la résolution. Étant donné le niveau de bruit important au niveau du voxel en TEP dynamique, on calcule généralement la valeur de $\mathrm{rCMRglu}$ en la moyennant sur un organe. Notre objectif consiste alors à améliorer le rapport signal sur bruit à l'échelle du voxel de façon à déterminer le paramètre d'intérêt au niveau du voxel et nom de l'organe. C'est ce qu'on appelle l'imagerie paramétrique.

7.2 Méthodes de reconstruction en TEP

L'extraction de paramètres physiologiques à l'échelle du voxel est un challenge qui passe par l'élaboration d'une méthode de reconstruction efficace.

En routines cliniques, les méthodes actuelles de reconstruction de données spatio-temporelles TEP consistent à mettre en œuvre une reconstruction analytique ou itérative pour chaque trame temporelle. Les méthodes usuelles sont la rétroprojection filtrée (« Filtered Back Projection » : FBP) et l'algorithme EM-ML. La rétroprojection filtrée est une technique de reconstruction analytique qui consiste à effectuer l'opération inverse de la projection qui se traduit par une étape de filtrage par un filtre rampe puis une rétroprojection du sinogramme filtré. De façon à limiter l'amplification du bruit lors du filtrage par le filtre rampe, on effectue généralement une étape complémentaire de filtrage passe-bas. Le principe de l'algorithme EM-ML est complètement différent ; il permet de minimiser l'antilog-vraisemblance poissonienne par une approche itérative. Rappelons que nous avons présenté cet algorithme dans la section 2.4.1 du chapitre 2. Cette méthode a initialement été proposée en TEP par [Shepp, Vardi, 1982; Lange, Carson, 1984]. Des versions accélérées de cet algorithme ont ensuite été suggérées. La plus connue est OSEM (« Ordered Subsets EM ») proposée par [Hudson, Larkin, 1994] qui consiste à appliquer l'algorithme EM à des sous-ensembles de projections ordonnés d'une façon particulière. Cependant, aucune preuve de convergence de cet algorithme n'existe. Dans [Fessler, Hero, 1994], les auteurs ont proposé la méthode SAGE (« Space-alternating Genralized EM ») qui consiste à mettre à jour séquentiellement des sous-ensembles de pixels de l'image reconstruite. Une autre approche a été proposée dans [Browne, De Pierro, 1996] où les auteurs se sont intéressés à une version modifiée de OSEM possédant des propriétés de convergence vers la solution du maximum de la vraisemblance poissonienne. Lorsque les sous ensembles considérés sont réduits à des singletons, l'algorithme résultant est connu sous le nom de RAMLA [Browne, De Pierro, 1996].

Les approches EM-ML donnent des résultats sensiblement meilleurs que la rétroprojection filtrée en terme de qualité de reconstruction. Cependant la suite générée par l'algorithme EM-ML converge vers une image de mauvaise qualité (cf. paragraphe 2.4.1 du chapitre 2). En effet, l'algorithme EM-ML va sélectionner l'image reconstruite qui s'ajuste au mieux aux observations mais dont les propriétés peuvent être différentes de celles de l'image originale. Rappelons que la reconstruction des données dynamiques TEP est un problème mal-posé et une petite perturbation des données peut induire de très grandes variations dans l'image reconstruite. Pour résoudre le problème de l'instabilité des solutions, différentes méthodes ont été proposées :

- l'approche EM-ML avec un critère d'arrêt [Veklerov, Llacer, 1987],
- la méthode des moindres carrés pénalisée où le critère résultant est quadratique, ce qui revient à supposer implicitement que le bruit présent est gaussien et qui permet ainsi d'utiliser des méthodes classiques d'optimisation [Fessler, 1994],

– la reconstruction SIEVES qui restreint l'espace des solutions admissibles en introduisant des contraintes assurant que les images reconstruites soient lisses. En pratique cette méthode revient à lisser l'image reconstruite par une approche EM-ML [Snyder, Miller, 1985],
– les approches MAP qui permettent l'introduction d'une distribution *a priori* modélisant une connaissance sur le type d'images acceptables [Herman *et al.*, 1979].

L'introduction d'une pénalisation dans le critère a largement permis d'améliorer la qualité de reconstruction [Hebert, Leahy, 1989]. Parmi ces approches MAP, on peut citer les travaux de Sauer et Bouman [Sauer, Bouman, 1993; Bouman, Sauer, 1996] basés, pour l'un, sur l'approche de Gauss-Seidel (initialement utilisée pour résoudre les EDP) et qui consiste à mettre à jour chaque pixel de façon à minimiser une fonction de coût globale, et pour l'autre, sur l'algorithme ICD (« Iterative Coordinate Descent »). Ces deux approches permettent de gérer respectivement un terme d'attache aux données quadratique (incluant une matrice de pondération des projections plus ou moins bruitées) ou une divergence de Kullback-Leibler. La régularisation se base sur des champs de Markov (MRF : « Markov Random Fields ») qui modélisent les interactions spatiales entre les voxels voisins. Toujours par le biais d'une régularisation par MRF, le travail de Fessler [Fessler, 1995] s'intéresse à un terme de fidélité hybride qui utilise l'antilog-vraisemblance poissonienne pour les éléments du sinogramme de faible intensité et des approximations quadratiques ou cubiques pour les observations ayant une intensité élevée. La résolution de ce critère utilise l'algorithme ICD [Bouman, Sauer, 1996] et le terme hybride permet de réduire le temps de calcul sans augmenter le biais.

Plus récemment, des études ont portées sur l'utilisation de modèles multirésolution. Par exemple, des travaux [Turkheimer *et al.*, 1999; Alpert *et al.*, 2006] se sont intéressés au débruitage des images reconstruites en utilisant des ondelettes spatiales 2D ou 3D. Une réduction de bruit allant jusqu'à un facteur de 2dB en terme de SNR a pu être observée dans [Alpert *et al.*, 2006]. Dans [Zhou *et al.*, 2007], les auteurs considèrent un terme d'attache aux données quadratique et ils ont remplacé la pénalisation avec MRF par une pénalisation favorisant la parcimonie des coefficients d'ondelettes (norme ℓ_1). L'image reconstruite est obtenue par un algorithme EM-MAP. Les mêmes auteurs [Zhou *et al.*, 2009] ont adapté l'algorithme RAMLA pour résoudre le problème d'estimation MAP sous la contrainte que le terme de régularisation soit différentiable (algorithme BSREM [De Pierro, Yamagishi, 2001]). Dans l'ensemble des approches mentionnées, la gestion de fonctionnelles non différentiables est souvent synonyme de méthodes coûteuses en temps de calcul.

Dans [Snyder, 1984; Nichols *et al.*, 2002; Kamasac *et al.*, 2005; Reader *et al.*, 2006; Verhaeghe *et al.*, 2008] les auteurs montrent l'avantage de prendre en compte les caractéristiques spatio-temporelles pour améliorer les résultats de reconstruction et l'estimation des cartes paramétriques. Ces méthodes peuvent être divisées en deux classes : les méth-

odes qui estiment directement les cartes paramétriques à partir des observations [Kamasac *et al.*, 2005] et les méthodes qui reconstruisent au préalable, pour chaque trame temporelle, l'activité mesurée dans les tissus puis qui estiment la carte paramétrique à partir de l'activité spatio-temporelle reconstruite [Snyder, 1984; Nichols *et al.*, 2002; Verhaeghe *et al.*, 2008; Sureau *et al.*, 2008; Reader *et al.*, 2006]. Notre travail s'inscrit dans cette deuxième catégorie.

Les algorithmes proximaux proposés dans une littérature récente d'optimisation convexe [Daubechies *et al.*, 2004; Combettes, Pesquet, 2007a; Chaux *et al.*, 2009] permettent de résoudre numériquement une grande classe de problèmes d'estimation de critères MAP. Ces algorithmes ont été, pour la plupart, décrits dans les chapitres 3, 4 et 5. Les principaux avantages de ces algorithmes sont :

– leur capacité à traiter efficacement une large classe de problèmes d'optimisation convexe (critères différentiables ou non-différentiables),
– leurs garanties de convergence des itérées vers la (les) solution(s) du problème considéré.

La première application de ces outils récents d'optimisation convexe en TEP dynamique a été suggérée par Verhaeghe *et al.* [Verhaeghe *et al.*, 2007] en considérant les itérations de l'algorithme explicite-implicite (FB) [Daubechies *et al.*, 2004] pour minimiser un critère composé d'un terme de fidélité quadratique et d'une régularisation par des ondelettes splines. Dans [Sureau *et al.*, 2008], les auteurs effectuent le débruitage des courbes d'activité temporelle (CAT) comme pré-traitement à la reconstruction. L'étape de débruitage est basée sur l'algorithme de Douglas-Rachford (DR) qui permet, rappelons-le, contrairement au FB, de minimiser un critère incluant une fonction n'étant pas de gradient Lipschitz, telle qu'une divergence de Kullback-Leibler généralisée.

Les algorithmes FB et DR ne sont pas directement applicables pour effectuer simultanément la tâche de débruitage et de reconstruction en utilisant un critère non-différentiable, une régularisation basée sur des trames d'ondelettes, et un terme de fidélité aux données adapté au bruit de Poisson. Dans le chapitre 3, nous avons proposé une classe d'algorithmes imbriqués permettant de trouver la solution de tels critères. Nous adapterons l'un de ces algorithmes à notre problème de reconstruction d'images spatio-temporelles TEP. Le fait que la divergence de Kullback-Leibler généralisée ne soit pas de gradient Lipschitz requiert l'utilisation de l'extension quadratique proposée dans la section 3.6.3-b) du paragraphe 3 dans un exemple de déconvolution. Comme nous le démontrerons dans ce chapitre, le cadre théorique requis pour cette extension quadratique peut facilement être étendu au cas de la reconstruction spatio-temporelle TEP. Cette méthode nous laisse également la possibilité d'ajouter une contrainte sur la dynamique de l'activité mesurée dans les tissus pour une trame temporelle donnée. Des tests seront menés sur un fantôme simulant un examen dynamique TEP [^{18}F]-FDG (2D+t). Nous verrons que cette méthode permet d'améliorer la qualité de reconstruction mais qu'elle possède également deux in-

convénients qui sont : un temps de calcul important dû à la boucle interne et la présence d'artefacts introduits par la régularisation ondelettes.

Pour contourner les problèmes rencontrés en combinant la classe des algorithmes imbriqués et l'extension quadratique, nous proposons d'utiliser la méthode proposée dans le chapitre 4. Rappelons que cette méthode permet initialement de résoudre des problèmes de déconvolution ou de super-résolution en présence de bruit non-nécessairement additif gaussien. Dans ce chapitre, nous adaptons cette méthode au problème de la reconstruction dynamique TEP. L'algorithme utilisé est l'algorithme proximal parallèle (PPXA) [Combettes, Pesquet, 2008]. Nous verrons que cette approche est particulièrement bien adaptée à l'imagerie TEP dynamique grâce à sa structure parallèle et à la possibilité de gérer une régularisation hybride telle que la combinaison d'une pénalisation par trames d'ondelettes et une mesure de variation totale [Rudin *et al.*, 1992]. Cette méthode sera dans un premier temps évaluée sur des données simulées puis elle sera appliquée sur des données réelles.

7.3 Extension quadratique et algorithmes proximaux

Dans un cadre stochastique, nous considérons d'une part que l'activité spatio-temporelle dans les tissus $y = (y_{i,t})_{(i,t)\in N_T}$ (avec $N_T = \{1,\dots,N\} \times \{1,\dots,T\}$) est une réalisation d'un vecteur aléatoire à valeurs réelles $Y = (Y_{i,t})_{(i,t)\in N_T}$. Nous supposons que le vecteur aléatoire Y possède des composantes indépendantes qui ont une loi de probabilité notée $\mu_{Y_{i,t}}(y_{i,t})$ et que ces fonctions sont log-concaves. D'autre part, nous supposons que l'activité projetée dans le domaine sinogramme, notée,

$$u = \Big(u_{j,t} = \sum_{i=1}^{N} A_{ji} y_{i,t}\Big)_{(j,t)\in M_T} \qquad \text{avec} \qquad M_T = \{1,\dots,M\} \times \{1,\dots,T\}$$

et les observations $z = (z_{j,t})_{(j,t)\in M_T}$ sont respectivement des réalisations des vecteurs aléatoires $U = (U_{j,t})_{(j,t)\in M_T}$ et $Z = (Z_{j,t})_{(j,t)\in M_T}$. Le vecteur aléatoire Z est supposé avoir des composantes indépendantes discrètes de probabilité conditionnelle $\mu_{Z_{j,t}|U_{j,t}=u_{j,t}}(z_{j,t})$ qui correspond à la vraisemblance poissonienne. Par conséquent, l'activité estimée $\widehat{y}$ par l'approche MAP s'exprime par :

$$\widehat{y} \in \underset{y=(y_{i,t})_{(i,t)\in N_T}\in\mathbb{R}^{NT}}{\operatorname{Argmax}} \prod_{t=1}^{T}\prod_{j=1}^{M} \mu_{Z_{j,t}|U_{j,t}=\sum_{i=1}^{N} A_{ji}y_{i,t}}(z_{j,t}) \prod_{t=1}^{T}\prod_{i=1}^{N} \mu_{Y_{i,t}}(y_{i,t}). \tag{7.7}$$

Nous précisons ci-dessous la définition de la vraisemblance poissonienne pour des données spatio-temporelles,

$$\mu_{Z_{j,t}|U_{j,t}=u_{j,t}}(z_{j,t}) = \frac{(u_{j,t})^{z_{j,t}}}{z_{j,t}!} \exp\big(-u_{j,t}\big).$$

La reformulation du critère (7.7) par passage à l'anti-logarithme, nous amène au problème d'optimisation convexe qui suit :

$$\widehat{y} \in \underset{y \in \mathbb{R}^{NT}}{\operatorname{Argmin}}\ \widetilde{g}(y) + \widetilde{f}(y) \tag{7.8}$$

où $\widetilde{f}$ et $\widetilde{g}$ sont des fonctions convexes de $\mathbb{R}^{NT}$ dans $\mathbb{R}$. La fonction $\widetilde{g}$ désigne le terme de fidélité aux données, il s'agit d'une mesure assurant la proximité de l'activité estimée par rapport aux observations. Comme nous l'avions précisé dans la section 2.2.2 du chapitre 2, cette fonctionnelle correspond à la divergence de Kullback-Leibler généralisée. La principale difficulté réside dans le choix de la fonction $\widetilde{f}$ qui désigne le terme de régularisation. Les chapitres précédents ont montré l'intérêt de régulariser dans le domaine des trames d'ondelettes. Par conséquent, nous reformulons le problème de minimisation (7.8) dans le domaine des trames.

On désigne par $F \in \mathbb{R}^{K \times NT}$ (tel que $K \geq NT$) la matrice associée à l'opérateur d'analyse d'une trame d'ondelettes spatio-temporelles. Dans la suite, nous considèrerons des trames ajustées telle que $F^* \circ F = \nu\,\mathrm{Id}$ avec $\nu \in]0, +\infty[$. On désigne par $x \in \mathbb{R}^K$ les coefficients associés à l'activité $y \in \mathbb{R}^{NT}$. Par conséquent l'activité originale mesurée dans les tissus $\bar{y}$ peut s'écrire :

$$\overline{y} = F^* \overline{x} = (F_t^* \overline{x})_{1 \leq t \leq T}$$

où, pour tout $t \in \{1, \ldots, T\}$, $F_t^* \in \mathbb{R}^{N \times K}$ représente la restriction de l'opérateur de synthèse de trame aux données reconstruites à l'instant t. Notre modèle direct (7.1) devient :

$$(\forall t \in \{1, \ldots, T\}) \qquad z_t = \mathcal{D}(AF_t^* \overline{x} + f_t + d_t), \tag{7.9}$$

où $\overline{x}$ représente le vecteur inconnu des coefficients de trame d'ondelettes associés à l'activité originale dans les tissus. Dans ce contexte, le problème de minimisation (7.8) peut être reformulé comme :

$$\widehat{x} \in \underset{x \in \mathbb{R}^K}{\operatorname{Argmin}}\ g(x) + f(x) \tag{7.10}$$

où l'activité reconstruite vaut $\widehat{y} = F^* \widehat{x}$. Le terme de fidélité aux données est la fonction g telle que :

$$g = \sum_{t=1}^{T} \Psi_t \circ A \circ F_t^*$$

où, pour tout $u_t = (u_{j,t})_{1 \leq j \leq M} \in \mathbb{R}^M$,

$$\Psi_t(u_t) = \sum_{j=1}^{M} \psi_{j,t}(u_{j,t}) \qquad \text{avec} \qquad \psi_{j,t}(u_{j,t}) = -\log \mu_{Z_{j,t}|U_{j,t}=u_{j,t}}(z_{j,t}). \tag{7.11}$$

Pour le terme de régularisation f, des études ont démontré l'avantage d'utiliser une norme ℓ_1 [Verhaeghe *et al.*, 2007]. Cependant, ce type de fonction est convexe mais non strictement convexe, il n'y a donc pas unicité de la solution. Dans [Chaux *et al.*, 2007], les auteurs ont proposé un terme de régularisation composé de la somme d'une norme ℓ_1 et d'une norme ℓ_p avec $p > 1$ de façon à assurer la stricte convexité du terme de régularisation. En choisissant ce type de régularisation, la fonction f est de la forme :

$$(\forall x = (x_k)_{1\leq k\leq K}) \qquad f(x) = \sum_{k=1}^{K} \chi_k |x_k| + \omega_k |x_k|^{p_k}. \tag{7.12}$$

avec, pour tout $k \in \{1, \ldots, K\}$, $\chi_k > 0$, $\omega_k > 0$ et $p_k > 1$. Remarquons à ce stade que les paramètres χ_k, ω_k et p_k peuvent avoir des valeurs différentes pour chaque coefficient. Dans un souci de simplifier l'estimation de ces paramètres, nous nous contenterons de les optimiser pour chaque sous-bande. Nous ajoutons dans le critère une contrainte sur la dynamique des données. On note C un ensemble convexe fermé de $\mathbb{R}^K$ tel que $C^* = F^*C = [0, V_{\max}]^{NT}$ où $V_{\max}$ dépend de la quantité de traceur injecté. Le problème (7.10) se reformule alors plus généralement comme :

$$\widehat{x} = \underset{x\in C}{\operatorname{argmin}}\ g(x) + f(x). \tag{7.13}$$

Le problème (7.13) se rapproche du problème (3.33) formulé dans le chapitre 3 lorsque $\mathcal{H} = \mathbb{R}^K$. Cependant la fonction g n'est pas de gradient Lipschitz. Pour contourner ce problème, et ainsi pouvoir utiliser la classe des algorithmes imbriqués, nous proposons d'adapter l'extension quadratique proposée dans la section 3.6.3 du chapitre 3 au problème de la reconstruction dynamique TEP.

7.3.1 Extension quadratique

Les fonctions $(\psi_{j,t})_{(j,t)\in M_T}$, définies par l'équation (7.11), vérifient l'hypothèse 3.41 présentée dans le chapitre 3, lorsque l'ensemble $\mathbb{I}$ est défini par

$$\mathbb{I} = \{(j,t) \in M_T \mid z_{j,t} > 0\},$$

pour une constante $\delta = 0$ et, pour tout $(j,t) \in M_T$, le paramètre $\alpha_{j,t} \equiv 1$. $\mathbb{I}$ désigne l'ensemble des bins non nuls. De plus, pour tout $(j,t) \in M_T$,

$$(\forall \theta \in\ ]0, +\infty[)\ (\forall \upsilon \in]0, +\infty[) \qquad 0 \leq \psi''_{j,t}(\upsilon) \leq \theta \Leftrightarrow \upsilon \geq \upsilon_{j,t}(\theta) = (z_{j,t}/\theta)^{1/2}.$$

Par conséquent, en se référant à l'inégalité (3.53) présentée dans le chapitre 3, on en déduit qu'il est possible de construire une approximation inférieure de $\psi_{j,t}$ qui est de gradient

θ-Lipschitz. Nous notons cette approximation $\psi_{\theta,j,t}$ et elle se définit, pour tout $(j,t) \in \mathbb{I}$, par

$$(\forall \upsilon \in \mathbb{R}) \qquad \psi_{\theta,j,t}(\upsilon) = \begin{cases} \frac{\theta}{2}\upsilon^2 + \zeta_{j,t,1}(\theta)\ \upsilon + \zeta_{j,t,0}(\theta) & \text{si } -\epsilon(\theta) \leq \upsilon < \upsilon_{j,t}(\theta), \\ \psi_{j,t} & \text{sinon,} \end{cases} \tag{7.14}$$

où la fonction ϵ et les constantes $\zeta_{j,t,0}(\theta)$ et $\zeta_{j,t,1}(\theta)$ ont des définitions similaires à celles formulées dans paragraphe 3.6.3 (chapitre 3). L'équation (7.14) signifie que pour toute valeur supérieure au seuil $\upsilon_{j,t}(\theta)$, la fonction $\psi_{\theta,j,t}$ est choisie égal à l'antilog-vraisemblance poissonienne et pour les valeurs inférieures au seuil, $\psi_{\theta,j,t}$ prend une forme quadratique. Pour tout $(j,t) \notin \mathbb{I}$, c'est à dire pour tout indice $(j,t) \in M_T$ tels que $z_{j,t} = 0$ (bin de sinogramme pour une trame temporelle donnée ayant une valeur nulle),

$$(\forall \upsilon \in \mathbb{R}) \qquad \psi_{\theta,j,t}(\upsilon) = \begin{cases} \upsilon & \text{si } \upsilon \geq -\epsilon(\theta) \\ +\infty & \text{sinon.} \end{cases} \tag{7.15}$$

Pour une valeur de θ donnée, nous définissons une approximation inférieure de la fonction g, notée g_θ, telle que

$$g_\theta = \sum_{t=1}^{T} g_{\theta,t} \qquad \text{avec} \qquad g_{\theta,t} = \Psi_{\theta,t} \circ A \circ F_t^*.$$

Cette approximation est une fonction de $\Gamma_0(\mathbb{R}^K)$ deux fois différentiable sur son domaine ayant un gradient de type $(\nu\|A\|^2\theta)$-Lipschitz sur l'ouvert $(AF^*)^{-1}\big(]0,+\infty[^{NT}\big)$. Le critère résultant est :

$$\widehat{x}_\theta = \underset{x \in C}{\arg\min}\ g_\theta(x) + f(x). \tag{7.16}$$

f correspond à la fonction définie par l'équation (7.12), il s'agit donc d'une fonction coercive et strictement convexe sur $\mathbb{R}^K$. De plus, par définition de A, de $C^* = F^*C = [0, V_{\max}]^{NT}$ et de Ψ_θ, on en déduit que $(AC^*) \cap \operatorname{dom} \Psi \subset [0, V_{\max}]^{NT} \neq \varnothing$ et l'ensemble des hypothèses requises par la proposition 3.42 (chapitre 3) sont rassemblées pour en déduire qu'il existe une valeur $\bar{\theta} \in]0,+\infty[$ telle que, pour tout $\theta \in [\bar{\theta},+\infty[$, le minimiseur $\widehat{x}_\theta$ de (7.16) est également le minimiseur du problème (7.13).

L'équation (7.16) combinée aux hypothèses faites sur g_θ est un cas particulier du problème 3.33 (chapitre 3). La classe des algorithmes imbriqués permet donc d'y apporter une solution numérique. Pour notre problème de reconstruction spatio-temporelle en TEP, nous adaptons l'algorithme itératif imbriqué nommé FB[DR] présenté dans la section 3.5.2 (chapitre 3).

7.3.2 Adaptation du FB[DR] pour la reconstruction TEP dynamique

L'algorithme FB[DR] est basé sur les itérations de l'algorithme implicite-explicite (FB) [Combettes, Wajs, 2005] et il requiert le calcul du gradient de g_θ et la connaissance de l'opérateur proximal de $f + \iota_C$. Rappelons que ce dernier ne possède pas une forme explicite car f et C ne sont pas séparables dans le même domaine (C est séparable dans le domaine image et f l'est dans le domaine des trames). Nous choisissons d'utiliser l'algorithme de DR pour effectuer le calcul de $\mathrm{prox}_{f+\iota_C}$. Ce calcul se traduit par une boucle interne pour chaque itération de FB. Dans la section 3.5 du chapitre 3, nous avions proposé trois algorithmes pour minimiser un critère composé de trois fonctions tel que (7.16), à savoir FB[DR], FB[Dyk] et DR[FB]. Notre choix pour FB[DR] s'explique par un calcul de gradient coûteux en TEP comparé à une application de déconvolution pour laquelle ce calcul peut efficacement être implanté dans le domaine de Fourier. Nous préférons donc éviter d'avoir à calculer le gradient dans la boucle interne, d'où le choix des approches FB[.] plutôt que DR[FB]. Ensuite, la préférence de FB[DR] sur FB[Dyk] est liée à la propriété de convergence en 1 itération (se référer au paragraphe 3.5.2 du chapitre 3) uniquement vérifiée par l'algorithme considéré.

Initialisation

① Soient $\gamma = 1.9/(\nu\|A\|^2\theta)$ et $\lambda = 1$.
② Soit $\tau = 1.5$.
③ Soit $n = 0$ et $x_0 \in C$.

Itérations

④ Calculer $y_n = F^* x_n$.
⑤ Pour chaque trame temporelle $t \in \{1, \ldots, T\}$, calculer $d_{t,n} = A^* \nabla \Psi_{\theta,t}(A y_{t,n})$.
⑥ Soit $p_n = x_n - \gamma F d_n$.
⑦ Soit $v_{n,0} = 2\mathrm{prox}_{\gamma f} p_n - p_n$.
⑧ Pour $m = 0, \ldots, M_n - 1$
  a) Calculer $v_{n,m+\frac{1}{2}} = P_C\Big(\dfrac{v_{n,m} + p_n}{2}\Big)$.
  b) Calculer $v_{n,m+1} = v_{n,m} + \tau\big(\mathrm{prox}_{\gamma f}(2v_{n,m+\frac{1}{2}} - v_{n,m}) - v_{n,m+\frac{1}{2}}\big)$.
  c) Si $v_{n,m+1} = v_{n,m}$, alors aller en ⑨.
⑨ Soit $x_{n+1} = x_n + \lambda\big(v_{n,m+\frac{1}{2}} - x_n\big)$.
⑩ Incrémenter n et aller en ④.

Algorithme 16: Algorithme FB[DR] appliqué à la reconstruction TEP dynamique.

L'algorithme de reconstruction résultant est présenté par l'algorithme 16. Les étapes d'intérêt sont :

- la descente de gradient qui est effectuée dans le domaine spatial. Elle s'effectue trame temporelle par trame temporelle,
- la décomposition sur une trame spatio-temporelle qui permet de tenir compte de l'information temporelle de l'activité cérébrale.

A l'étape ①, on choisit le « pas » γ et le paramètre de relaxation λ associés à l'algorithme implicite-explicite. L'étape ② se résume au choix du paramètre de relaxation de l'algorithme de DR. L'algorithme FB génère une suite $(x^{(n)})_{n\in\mathbb{N}}$. L'étape ③ est l'initialisation de la suite générée par l'algorithme FB ; il s'agit d'un vecteur de coefficients de trame d'ondelettes appartenant à l'ensemble convexe C. L'étape ④ transforme le vecteur de coefficients de trame d'ondelettes généré par l'algorithme dans le domaine image. L'étape suivante désigne le calcul du gradient de g_θ. Précisons que le gradient de g_θ est défini, pour tout $x \in \mathbb{R}^K$, par

$$\nabla g_\theta(x) = F(A^* \nabla\Psi_{\theta,t}(AF_t^* x))_{1\leq t\leq T},$$

où, pour tout élément $u_t = (u_{j,t})_{1\leq j\leq M} \in \mathbb{R}^M$,

$$\nabla\Psi_{\theta,t}(u_t) = (\psi'_{\theta,j,t}(u_{j,t}))_{1\leq j\leq M}, \tag{7.17}$$

$$\psi'_{\theta,j,t}(u_{j,t}) = \begin{cases} 1 - z_{j,t}/u_{j,t} & \text{si } z_{j,t} > 0 \text{ et } u_{j,t} \geq \upsilon_{j,t}(\theta) \\ \theta u_{j,t} + \zeta_{j,t,1}(\theta) & \text{si } z_{j,t} > 0 \text{ et } \epsilon(\theta) \leq u_{j,t} < \upsilon_{j,t}(\theta) \\ 1 & \text{si } z_{j,t} = 0 \text{ et } \epsilon(\theta) \leq u_{j,t}. \end{cases} \tag{7.18}$$

La descente de gradient s'effectue à l'étape ⑥. L'étape ⑧ nous permet de calculer l'opérateur proximal de $\iota_C + f$ au point $p^{(n)}$ à l'aide de l'algorithme de DR. L'initialisation du vecteur $(v^{(\cdot,m)})_{m\in\mathbb{N}}$ est présentée à l'étape ⑦. Cette initialisation permet d'assurer la convergence de l'algorithme de DR en une itération lorsque l'opérateur proximal de f au point $p^{(n)}$ appartient à C. Le calcul de P_C se base sur la proposition 3.12 (chapitre 3), tel que,

$$(\forall x \in \mathbb{R}^K) \qquad P_C(x) = x + \frac{1}{\nu} F(P_{C^*}(F^* x) - F^* x) \tag{7.19}$$

Pour le calcul de $\operatorname{prox}_{\gamma_n f}$, les formes explicites sont données par l'exemple 3.38 et la proposition 3.39 (chapitre 3). Enfin, l'étape ⑧ correspond à l'étape de relaxation de l'algorithme FB.

Si M_n est choisi suffisamment grand, d'après la proposition 3.34 et le fait que $\operatorname{ri}\operatorname{dom} f \cap \operatorname{ri} C = \operatorname{ri} C = (F^*)^{-1}\big(]0, V_{\max}[^{NT}\big) \neq \varnothing$, la suite $(x^{(n)})_{n\in\mathbb{N}}$ générée par l'algorithme FB[DR] converge vers une solution du problème (7.16). Si de plus θ est suffisamment grand, la suite converge vers le minimiseur du problème (7.13).

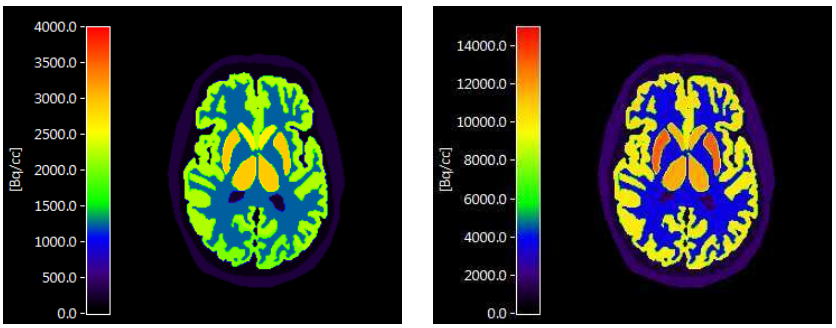
7.3.3 Matériel et résultats expérimentaux sur données simulées

L'impact de notre méthode sur la reconstruction de données dynamiques TEP est illustré sur deux coupes différentes du fantôme *Zubal* [Zubal *et al.*, 1994] incluant deux artères additionnelles. Les choix des coupe 1 et 2 s'expliquent respectivement par le fait que la première coupe est centrée sur le cortex (zone verte sur la figure 7.5 en haut à gauche), les noyaux gris centraux (zone jaune supérieure), le thalamus (zone jaune inférieure) et la matière blanche (zone bleue) ; alors que coupe 2 est centrée sur les artères (zone rouge sur la figure 7.6 en haut à gauche). La restauration de ce type de coupe permet de mesurer directement la fonction d'entrée sur l'image reconstruite. Chaque coupe $2D + t$ est composée d'images de $N = 256 \times 256$ voxels (taille : 1.1mm$\times 1.1$mm). L'activité générée correspond à un examen [^{18}F]-FDG qui est simulé et divisé en $T = 16$ trames temporelles ayant une durée qui varie de 50 secondes pour les premières trames temporelles à 5 minutes pour les dernières. Des sinogrammes de taille $M = 288$ (radial) $\times$ 144 (orientations), avec un échantillonnage radial de 2.247 mm, ont été simulés par projection analytique du fantôme en présence de bruit de Poisson. Les effets d'atténuation et de coïncidences fortuites et diffusées n'ont pas été simulés. Le nombre d'évènements dans la coupe 1 varie de 3 pour la première trame-temporelle à 647162 pour la dernière. La coupe 2 a un nombre d'évènements qui varie de 47 à 331348. Ce nombre de coïcidences est un choix réaliste.

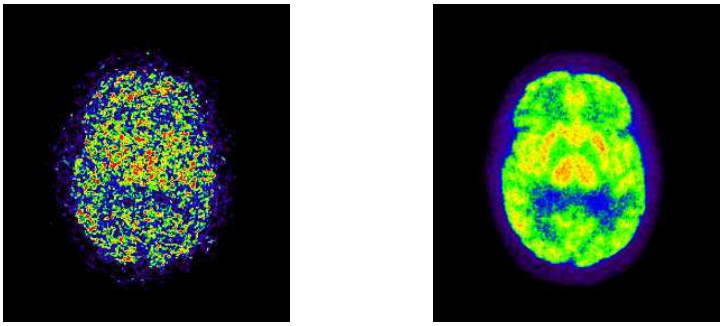
L'algorithme FB[DR] est lancé sur 2000 itérations de 50 s chacune. La boucle interne de DR s'arrête lorsque $\|x^{(n)} - x^{(n-1)}\| \leq 10^{-5}$. Nous avons choisi des ondelettes orthogonales avec des filtres de symmlets de longueur 6 sur 2 niveaux de résolution pour la décomposition spatiale et des ondelettes de Daubechies-6 sur intervalle [Cohen *et al.*, 1993] pour la décomposition temporelle plutôt que des ondelettes de Daubechies classiques. Ce dernier choix est motivé par le faible nombre de trames temporelles. Le paramètre θ est choisi égal à 10^{-4} et les paramètres $(\chi_k)_{1\leq k\leq K}$, $(\omega_k)_{1\leq k\leq K}$ et $(p_k)_{1\leq k\leq K}$, associés au terme de régularisation f, ont été déterminés pour chaque sous bande par une approche de maximum de vraisemblance avec $p_k = \{3/2, 4/3, 2\}$. L'impact du paramètre θ sur le taux de convergence de l'algorithme est présenté sur la figure 7.9.

Nous comparons notre approche avec une approche EM-ML avec critère d'arrêt (stoppée à l'itération donnant la plus faible EQM, ici 10 itérations pour les coupes 1 et 2). Nous présentons également les résultats obtenus avec la méthode SIEVES. La largeur du filtre est fixé de façon à minimiser l'EQM. Le nombre d'itérations dans cette approche est 120 et 100 pour les coupes 1 et 2 respectivement. La largeur à mi-hauteur maximum (FWHM : « Full Width at Half Maximum ») du filtre gaussien est de 4.7mm$\times 4.7$mm.

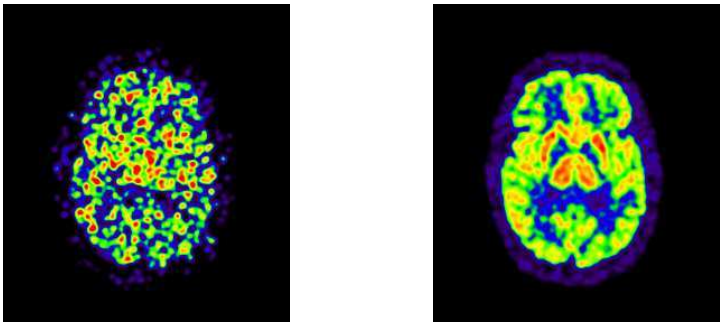
Les figures 7.5 et 7.6 représentent les différents résultats de reconstruction pour les 4ème et 14ème trames-temporelles ($t = 4$ et $t = 14$) de chaque coupe. Les premières trames temporelles étant beaucoup plus bruitées que les dernières, le choix des 4ème et



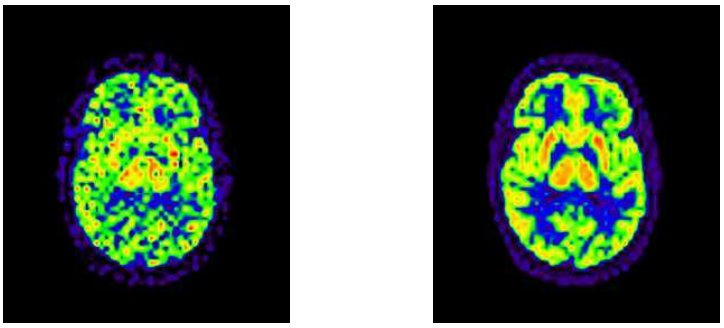
Activité originale pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14



Activité reconstruite par EM-ML avec critère d'arrêt pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14

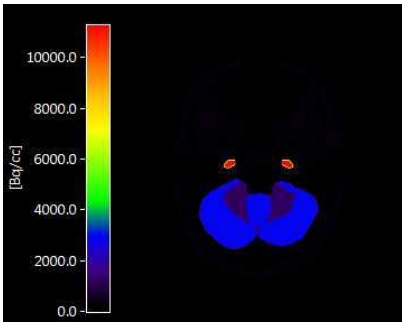


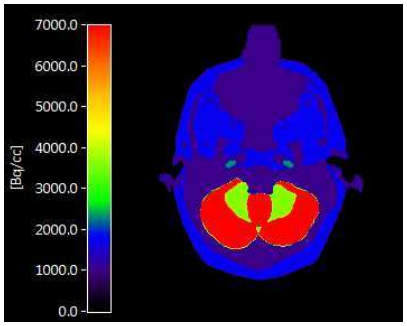
Activité reconstruite par « SIEVES » pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14



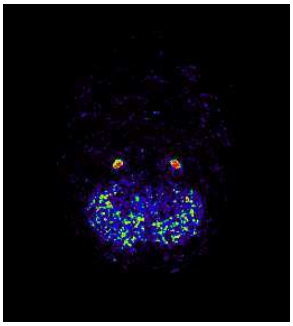
Activité reconstruite par FB[DR] pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14

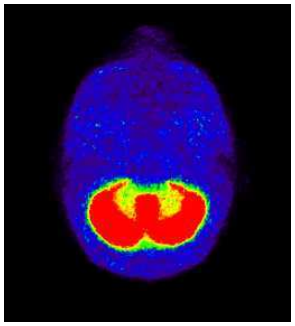
FIGURE 7.5 – Comparaison des activités reconstruites pour la coupe 1 en fonction de la méthode de reconstruction.



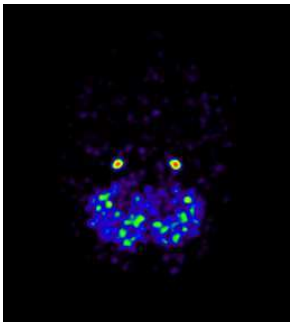


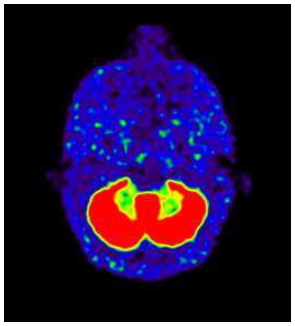
Activité originale pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14



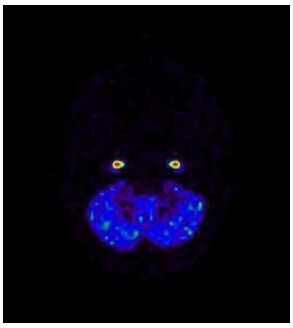


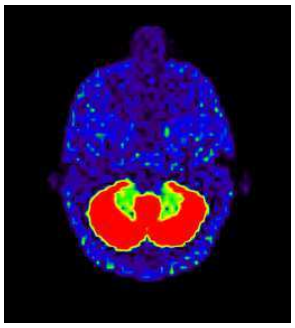
Activité reconstruite par EM-ML avec critère d'arrêt pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14





Activité reconstruite par « SIEVES » pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14





Activité reconstruite par FB[DR] pour : (i) la trame temporelle n°4 (ii) la trame temporelle n°14

FIGURE 7.6 – Comparaison des activités reconstruites pour la coupe 2 en fonction de la méthode de reconstruction.

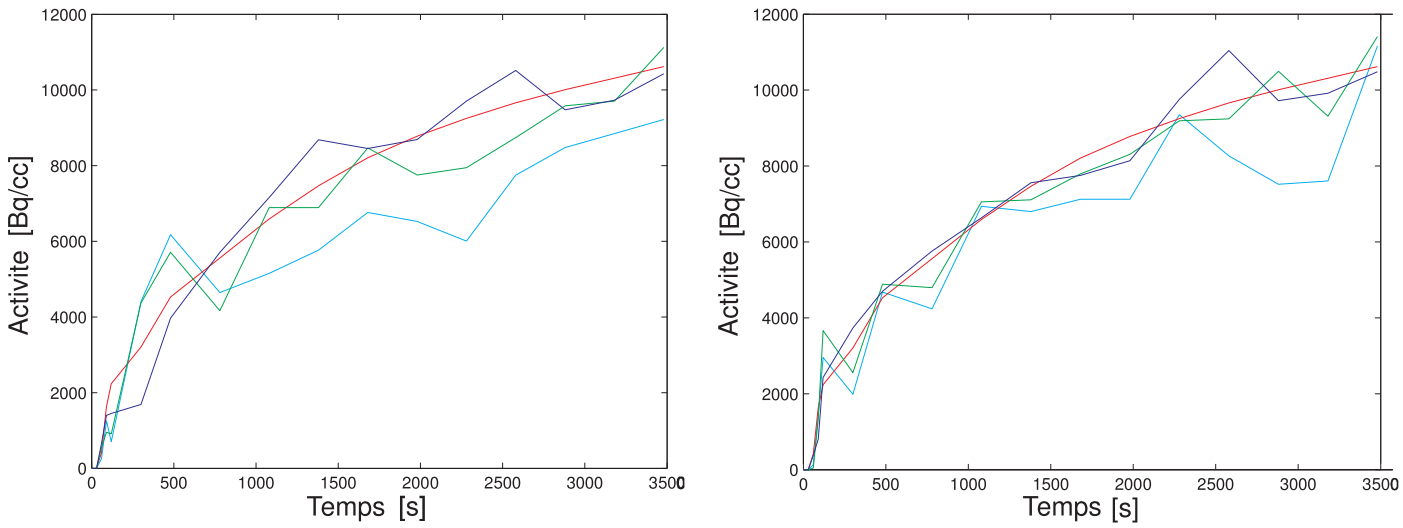


FIGURE 7.7 – CAT extraites de la coupe 1 pour deux voxels voisins. Activité originale (rouge), activité reconstruite avec EM-ML avec critère d'arrêt (cyan), SIEVES (vert) et FB[DR] (bleue).

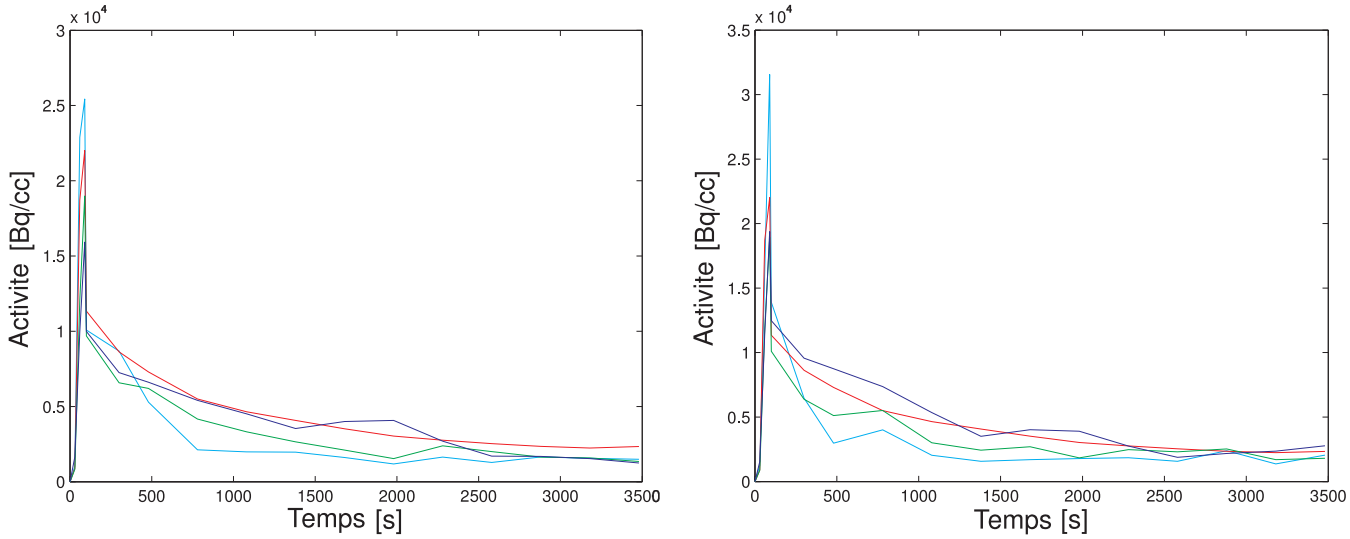


FIGURE 7.8 – CAT extraites de la coupe 2 pour deux voxels voisins. Activité originale (rouge), activité reconstruite avec EM-ML avec critère d'arrêt (cyan), SIEVES (vert) et FB[DR] (bleue).

14ème trames-temporelles permet d'illustrer les qualités de reconstruction pour différents niveaux de bruit au sein d'un même examen. Comme nous pouvons le remarquer, les structures sont bien mieux reconstruites en utilisant l'approche proposée. Le principal avantage de prendre en compte l'aspect temporel peut être observé pour les premières trames temporelles des coupes 1 et 2 lorsque le bruit est relativement important (colonnes de gauche sur les figures 7.5 et 7.6). D'autre part, on peut remarquer sur la figure 7.5 (en bas à droite) que les zones froides sont mieux reconstruites.

Les courbes d'activités temporelles (CAT) sont présentées sur les figures 7.7 et 7.8 pour deux pixels voisins dans le cortex (zone verte sur la figure 7.5 représentant l'activité originale) et dans les artères (zone rouge sur la figure 7.6 représentant l'activité originale). Le tableau 7.1 affiche les différences d'EQM entre les CAT reconstruites et les CAT originales. Dans chaque configuration, notre approche est la plus précise.

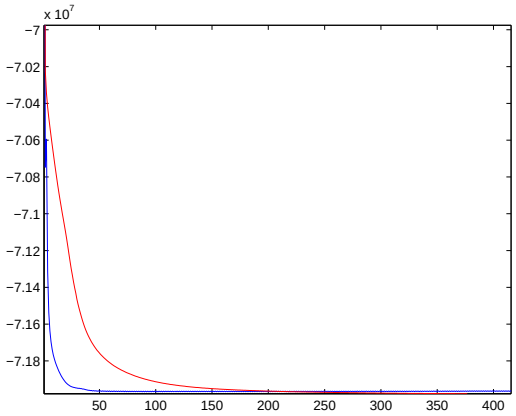


FIGURE 7.9 – Evolution de $f(x^{(n)}) + g_\theta(x^{(n)})$ au cours des itérations pour $\theta = 10^{-4}$ (en rouge) et pour $\theta = 10^{-5}$ (en bleu).

	Fig. 7.7 (gauche)	Fig. 7.7 (droite)	Fig. 7.8 (gauche)	Fig. 7.8 (droite)
EM-ML	21.9e+06	16.4e+06	14.9e+06	15.2e+06
SIEVES	4.09e+06	2.45e+06	7.68e+06	5.88e+06
FBDR	**2.24e+06**	**2.20e+06**	**5.96e+06**	**3.67e+06**

TABLE 7.1 – EQM associées aux CAT présentées sur les figures 7.7 et 7.8.

7.3.4 Conclusion

Nous venons de présenter les résultats obtenus en utilisant l'algorithme FB[DR] pour reconstruire l'activité présente dans les tissus. Cette méthode est flexible et permet de considérer des critères non-différentiables, qui peuvent être intéressants lorsque des trames d'ondelettes sont utilisées. Les résultats obtenus par des simulations en TEP $2D + t$ sont réellement encourageants. Cependant, cette approche est actuellement limitée par ses temps de calculs encore très longs provenant des itérations internes. Pour contourner ce problème, nous avons développé une seconde approche, qui ne nécessite pas de boucle interne et qui est bien adaptée aux gros volumes de données du fait de sa facilité à être implantée en parallèle.

7.4 PPXA et régularisation hybride

L'approche précédente était limitée pour deux raisons principales : (i) un temps de calcul coûteux et (ii) des artefacts pouvant être engendrés par la régularisation sur les coefficients de base d'ondelettes. Pour contourner ces problèmes, nous proposons d'adapter l'algorithme PPXA [Combettes, Pesquet, 2008] au problème de reconstruction dynamique TEP.

7.4.1 Méthode

Nous considérons une approche variationnelle où le critère est similaire au critère (7.13) mais où f correspond à une régularisation hybride $\ell_1 + \mathrm{tv}$, i.e.

$$\hat{x} \in \underset{x \in C \subset \mathbb{R}^K}{\mathrm{Argmin}}\ g(x) + f_1(x) + f_2(x).$$

La fonction g reste donc le terme de fidélité aux données défini par l'équation (7.11). Le terme f_1 sera lié à la régularisation par variation totale, notée tv et définie par l'équation (4.47) (chapitre 4), et s'appliquera sur l'activité dans les tissus de chaque trame-temporelle. La fonction f_2 désigne la régularisation sur les coefficients de trame d'ondelettes spatio-temporelles. Remarquons que cette approche ne requerra pas la stricte convexité de la fonction f_1 pour avoir des garanties de convergence vers un minimiseur global (pas d'extension quadratique). Nous considèrerons donc une simple norme ℓ_1 comme pénalisation. Enfin, de la même manière que pour l'approche précédente, nous ajoutons une contrainte $C^* = F^*C \subset \mathbb{R}^{NT}$ permettant de limiter la dynamique de l'activité. Le problème de minimisation qui en résulte consiste à trouver

$$\widehat{x} \in \underset{x=(x_k)_{\{1,\dots,K\}} \in \mathbb{R}^K}{\mathrm{Argmin}} \sum_{t=1}^{T} \Big(\Psi_t(AF_t^* x) + \vartheta \mathrm{tv}(F_t^* x) \Big) + \sum_{k=1}^{K} \chi_k |x_k| + \iota_C(x), \qquad (7.20)$$

où, pour tout $k \in \{1, \ldots, K\}$, $\chi_k > 0$ et où $\vartheta > 0$ désigne le paramètre de régularisation permettant d'ajuster l'impact de la régularisation par variation totale. L'activité reconstruite est notée $\widehat{y} = F^* \widehat{x}$.

Pour trouver un minimiseur de cette fonctionnelle, un algorithme bien adapté est l'algorithme PPXA [Combettes, Pesquet, 2008]. Les itérations de PPXA pour résoudre (7.20) sont présentées par l'algorithme 17.

Initialisation

Soit $\gamma \in \,]0, +\infty[$ et $\lambda = 1$.
Pour tout $q \in \{1, \ldots, 4\}$, soit $(\omega_q)_{1 \leq q \leq 4} \in]0, 1]^4$ tel que $\sum_{q=1}^{4} \omega_q = 1$.
Soient $\ell = 0$, $(u_{q,0})_{1 \leq q \leq 4} \in (\mathbb{R}^K)^4$ et $x_0 = \sum_{q=1}^{4} \omega_q u_{q,0}$.

Itérations

① Pour $t = 1, \ldots, T$
$\quad p_{1,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_1} \Psi_t (A F_t^* \cdot)} u_{1,\ell}$
② Pour $t = 1, \ldots, T$
$\quad p_{2,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_2} \mathrm{tv}(F_t^* \cdot)} u_{2,\ell}$
③ $p_{3,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_3} \|\cdot\|_1} u_{3,\ell}$
④ $p_{4,\ell} = P_C u_{4,\ell}$

⑤ $p_\ell = \omega_1 p_{1,\ell} + \omega_2 p_{2,\ell} + \omega_3 p_{3,\ell} + \omega_4 p_{4,\ell}$
⑥ Pour $q = 1, \ldots, 4$
$\quad u_{q,\ell+1} = u_{q,\ell} + \lambda \, (2 \, p_\ell - x_\ell - p_{q,\ell})$
⑦ $x_{\ell+1} = x_\ell + \lambda \, (p_\ell - x_\ell)$
⑧ Incrémenter ℓ et aller en ①.

Algorithme 17: PPXA pour minimiser le critère 7.20

La principale difficulté de cet algorithme réside dans les étapes ① à ④ qui consistent à calculer les opérateurs proximaux associés aux différentes fonctions composant le critère. L'étape ③ est une étape de seuillage doux des coefficients de trame. L'étape ④ se calcule par la formulation (7.19). Les deux façons de procéder au calcul de l'opérateur proximal associé à la variation totale sont (i) l'utilisation de l'approche itérative proposée dans [Chambolle, 2004] et (ii) la décomposition du terme de variation totale en une somme de fonctions dont on sait calculer l'opérateur proximal. On se reportera à la section 4.3.3 (chapitre 4) pour plus de détails sur cette dernière façon de procéder. Il en résulte que tv peut être décomposée en une somme de 4 fonctions quand des filtres de Roberts sont considérés. Dans la suite nous adopterons cette deuxième solution et la fonction tv sera

écrite comme une somme de 4 fonctions $(\mathrm{tv}_{p_1,p_2})_{0\leq p_1<2,0\leq p_2<2}$ comme suggéré dans le chapitre 4.

La dernière difficulté réside en l'étape ① qui consiste à calculer l'opérateur proximal de $\Psi_t(AF_t^*\cdot)$. On se réfèrera au paragraphe 4.3.2 (chapitre 4) pour mieux comprendre la limitation de ce calcul dans le cas où A désigne l'opérateur associé à une convolution. On y trouve également une technique de décomposition de $\Psi_t(A\cdot)$ qui permet de se ramener à une somme de fonctions plus simples dont les opérateurs proximaux prennent une forme explicite. Nous adaptons à présent cette technique de décomposition lorsque que A désigne la matrice système en TEP.

Nous supposons que pour tout $r \in \{1,\ldots,R\}$, $\mathbb{I}_r$ est une partition de $\{1,\ldots,M\}$ en ensembles non vides et nous désignons par $A^{(r)} = (A_{j,i})_{j\in\mathbb{I}_r,1\leq i\leq N}$ la matrice constituée de lignes de A sans recouvrement et donc orthogonales entre elles. Contrairement au cadre de la déconvolution, il est difficile de représenter la matrice système en TEP. En pratique, ces sous-ensembles désignent les lignes de la matrice système associées aux bins des TOR parallèles n'ayant pas de pixels communs.

De plus, pour tout $t \in \{1,\ldots,T\}$, nous pouvons écrire chaque Ψ_t comme une somme de R fonctions notées $\Psi_t^{(r)}\colon \mathbb{R}^{\mathrm{Card}\{\mathbb{I}_r\}} \to \,]-\infty,+\infty]$, qui, pour tout $r \in \{1,\ldots,R\}$, sont telles que

$$\Psi_t^{(r)} = \sum_{j\in\mathbb{I}_r} \Psi_{j,t}$$

et qui opèrent sur les bins de sinogrammes appartenant à $\mathbb{I}_r$. Il en résulte que

$$\Psi_t \circ A = \sum_{r=1}^{R} \Psi_t^{(r)} \circ A^{(r)}.$$

Cette décomposition permet d'avoir une forme explicite pour le calcul de l'opérateur proximal associé à $\Psi_t^{(r)} \circ A^{(r)}$.

La méthode qui découle de l'ensemble des remarques précédentes est résumée par l'algorithme 18. La convergence de la suite $(x_\ell)_{\ell\in\mathbb{N}}$ vers un minimiseur de (7.20) est assurée par la coercivité de f_1 et par le fait que

$$\Big(\bigcap_{t=1}^{T}\big(\cap_{r=1}^{R}\left(A^{(r)}\right)^{-1}(\mathrm{ri}\,\mathrm{dom}\,\Psi^{(r)})\big)\Big)\bigcap \mathrm{ri}\,C^* \neq \varnothing.$$

Cette dernière condition est vérifiée car $]0,+\infty[^{\mathrm{Card}\{\mathbb{I}_r\}} \subset \mathrm{dom}\,\Psi^{(r)}$, $C^* = [0,V_{\max}]^N$ et puisque pour tout $r \in \{1,\ldots,R\}$, $A^{(r)}$ est supposée positive, à valeurs réelles et composée de lignes étant toutes non nulles.

Initialisation

On choisit $\gamma = 10$ et $\lambda = 1$.
On pose $\omega_1 = \ldots = \omega_R = \frac{1}{4R}$, $\omega_{R+1} = \ldots = \omega_{R+4} = \frac{1}{16}$ et $\omega_{R+5} = \omega_{R+6} = \frac{1}{4}$.
Soit $\ell = 0$ et, pour tout $q \in \{1, \ldots, R+6\}$, $u_{q,0} = F\ z$.
Soit $x_0 = \sum_{q=1}^{R+6} \omega_q u_{q,0}$.

Itérations

① Pour $t = 1, \ldots, T$

Pour $r = 1, \ldots, R$ $\quad p_{r,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_q}\Psi_t^{(r)} \circ A^{(r)} \circ F_t^*}\big(u_{1,\ell}\big)$

$p_{R+1,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_{R+1}}\mathrm{tv}_{0,0}(F_t^* \cdot)} u_{R+1,\ell}$

$p_{R+2,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_{R+2}}\mathrm{tv}_{0,1}(F_t^* \cdot)} u_{R+2,\ell}$

$p_{R+3,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_{R+3}}\mathrm{tv}_{1,0}(F_t^* \cdot)} u_{R+3,\ell}$

$p_{R+4,t,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_{R+4}}\mathrm{tv}_{1,1}(F_t^* \cdot)} u_{R+4,\ell}$

$p_{R+5,\ell} = \mathrm{prox}_{\frac{\gamma}{\omega_{R+5}}\|\cdot\|_1} u_{R+5,\ell}$

$p_{R+6,\ell} = P_C\big(u_{R+6,\ell}\big)$

② $p_\ell = \sum_{q=1}^{R+6} \omega_q p_{q,\ell}$

③ Pour $q = 1, \ldots, R+6$

$u_{q,\ell+1} = u_{q,\ell} + \lambda\ (2\ p_\ell - x_\ell - p_{q,\ell})$

④ $x_{\ell+1} = x_\ell + \lambda\ (p_\ell - x_\ell)$

⑤ Incrémenter ℓ et aller en ①.

Algorithme 18: PPXA pour minimiser critère 7.20.

7.4.2 Matériel et résultats expérimentaux sur données simulées

La méthode est évaluée sur des données simulées spatio-temporelle (2D+t), basées sur le fantôme cérébral de Zubal incluant une artère additionnelle. Nous utilisons un modèle compartimental avec $k_4 = 0$ pour simuler un examen cérébral [^{18}F]-FDG composé de $T = 16$ trames temporelles qui ont des durées variant de 30 secondes pour les premières trames temporelles à 3 minutes pour les dernières. De plus, de faibles variations de K_i ont été créées dans le cortex et le putamen. Les valeurs des K_i ont été fixées en s'inspirant des valeurs obtenues sur des données réelles dans [Piert *et al.*, 1996]. Sur la figure 7.10, nous avons présenté les trames-temporelles $t = 4$, $t = 8$ et $t = 14$ du fantôme généré.

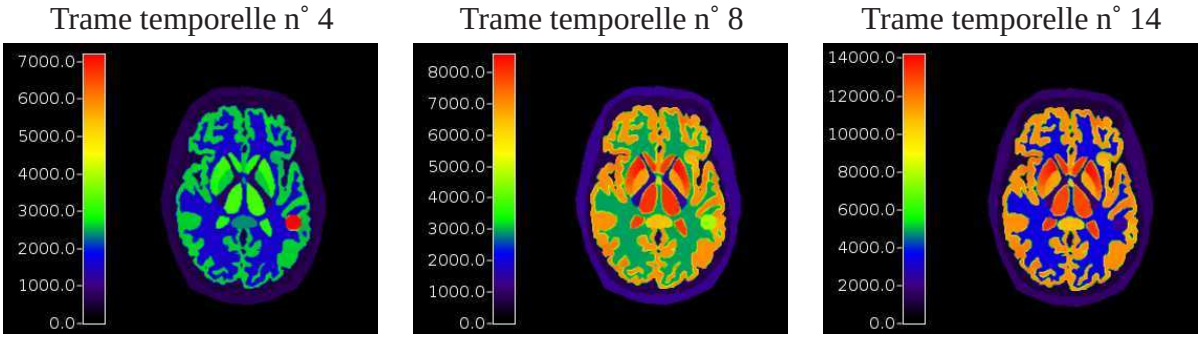


FIGURE 7.10 – Activité dans les tissus pour les trames temporelles $t = 4$, $t = 8$ et $t = 14$ en Bq/cc.

Les variations d'intensité dans le cortex et le putamen sont visibles sur les trames temporelles 8 et 14. Deux sinogrammes (2D+t) de taille 288 (radial) $\times$ 144 (angles) avec un échantillonnage radial de 2.247 mm ont été simulés par projection analytique en présence de bruit de Poisson. Les coincidences atténuées, fortuites et diffusées ne sont pas simulées. Le premier sinogramme (sinogramme 1) créé est assez peu bruité et le nombre d'évènements varie entre 449 pour les premières trames temporelles à 3432199 pour les dernières ; il est présenté sur la figure 7.11. Le second sinogramme (sinogramme 2), présenté figure 7.12, est formé de 36 coïncidences pour la première trame temporelle et de 343589 pour la dernière.

L'algorithme PPXA est implanté en parallèle avec la librairie OpenMP [2] sur une architecture 8 coeurs et nous effectuons 400 itérations de 10 s chacune. La matrice système est décomposée en $R = 432$ sous-ensembles. La contrainte sur la dynamique des données est $C^* = [0, 10^5]^{NT}$ (en Bq/cc). Nous avons choisi une base orthonormale d'ondelettes correspondant à des filtres de symmlets de longueur 6 et des ondelettes de Daubechies sur

2. http ://openmp.org/wp/

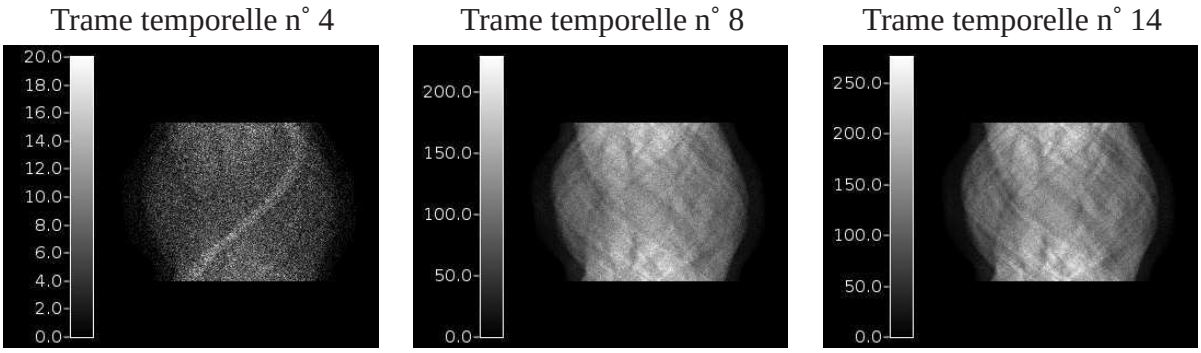


FIGURE 7.11 – sinogramme 1 : sinogramme (2D+t) généré pour les trames temporelles $t = 4$, $t = 8$ et $t = 14$ où l'échelle représente le nombre de coïncidences.

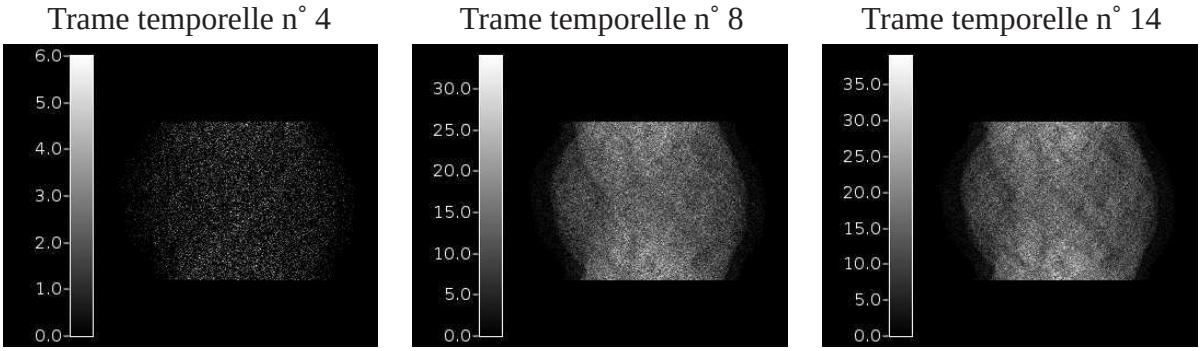


FIGURE 7.12 – sinogramme 2 : sinogramme (2D+t) généré pour les trames temporelles $t = 4$, $t = 8$ et $t = 14$ où l'échelle représente le nombre de coïncidences.

intervalle de longueur 6 en temporel. L'influence du nombre de niveaux de décomposition spatiale et temporelle est présenté sur les figures 7.13 et 7.15. Sur ces figures sont tracées les EQM et SSIM évaluées entre l'activité originale et l'activité reconstruite pour chacune des trames temporelles. Les figures 7.14 et 7.16 présentent l'activité reconstruite pour différentes valeurs du paramètre ϑ (lié à la variation totale). Il apparaît que la variation totale permet de lisser certains des artefacts. Il ressort de nos tests que le choix $\vartheta = 10^{-5}$ ainsi que le choix d'une décomposition sur 2 niveaux en temporel et 3 niveaux spatialement permettent d'obtenir un bon compromis aussi bien en termes d'EQM que de SSIM sur les données reconstruites.

Nous comparons les images reconstruites par notre méthode (PPXA), par EM-ML après 250 itérations, par EM-ML lissé (« SIEVES »). Dans cette dernière, le nombre

d'itérations de EM-ML est fixé à 250 et un noyau de largeur à mi-hauteur est fixé de deux façons différentes. Dans le premier cas, nous cherchons la largeur à mi-hauteur permettant de minimiser l'EQM entre l'image reconstruite et l'image considérée comme référence ; il en résulte une largeur à mi-hauteur de 10 mm $\times$ 10 mm. Nous nommerons cette première approche « SIEVES 1 ». La largeur à mi-hauteur dans la seconde méthode (« SIEVES 2 ») est fixé à 12 mm $\times$ 12 mm et permet d'obtenir un niveau de régularité de l'image identique à notre approche. Deux résultats obtenus avec notre approche sont présentés. Le premier (PPXA 1) estime les paramètres $(\chi_k)_{1\leq k\leq K}$ par maximum de vraisemblance à partir des données originales, alors que la seconde (PPXA 2) estime ces mêmes paramètres à partir de l'image EM-ML obtenue après 20 itérations. Cette deuxième approche permet de se dispenser d'utiliser l'information contenue dans l'image originale, inconnue en réalité. Ce choix de 20 itérations a été effectué après une série de tests que nous avons mené. Les figures 7.17, 7.18 et 7.24 présentent les trames $t = 4$, $t = 8$ et $t = 14$ provenant des images reconstruites par les différentes approches, et ceci à partir du sinogramme 2 (le plus bruité) qui correspond au cas de figure le plus réaliste. Pour les premières trames temporelles, lorsque le bruit est particulièrement élevé, l'algorithme PPXA reconstruit mieux l'activité du cerveau même si des artefacts peuvent être observés (par exemple au niveau du thalamus sur la figure 7.16). Pour confirmer ces résultats, le tableau 7.2 présente les valeurs d'EQM pour différentes zones de l'image, à savoir : le cortex, le putamen, l'artère et la matière blanche ; nous pouvons remarquer que notre approche permet d'obtenir la plus petite EQM dans la plupart des zones d'intérêt considérées.

Les courbes d'activités temporelles (CAT) sont présentées sur les figures 7.20, 7.21, 7.22 et 7.23 pour quatres pixels voisins et pour différentes zones d'intérêt, à savoir le cortex, dans les noyaux gris centraux, dans l'artère additionnelle et dans les artères.

En utilisant le logiciel commercial Pmod [3], nous calculons les images paramétriques à partir des images reconstruites par PPXA, EM-ML et SIEVES. Les résultats sont présentés sur la figure 7.24.

3. http ://www.pmod.com/

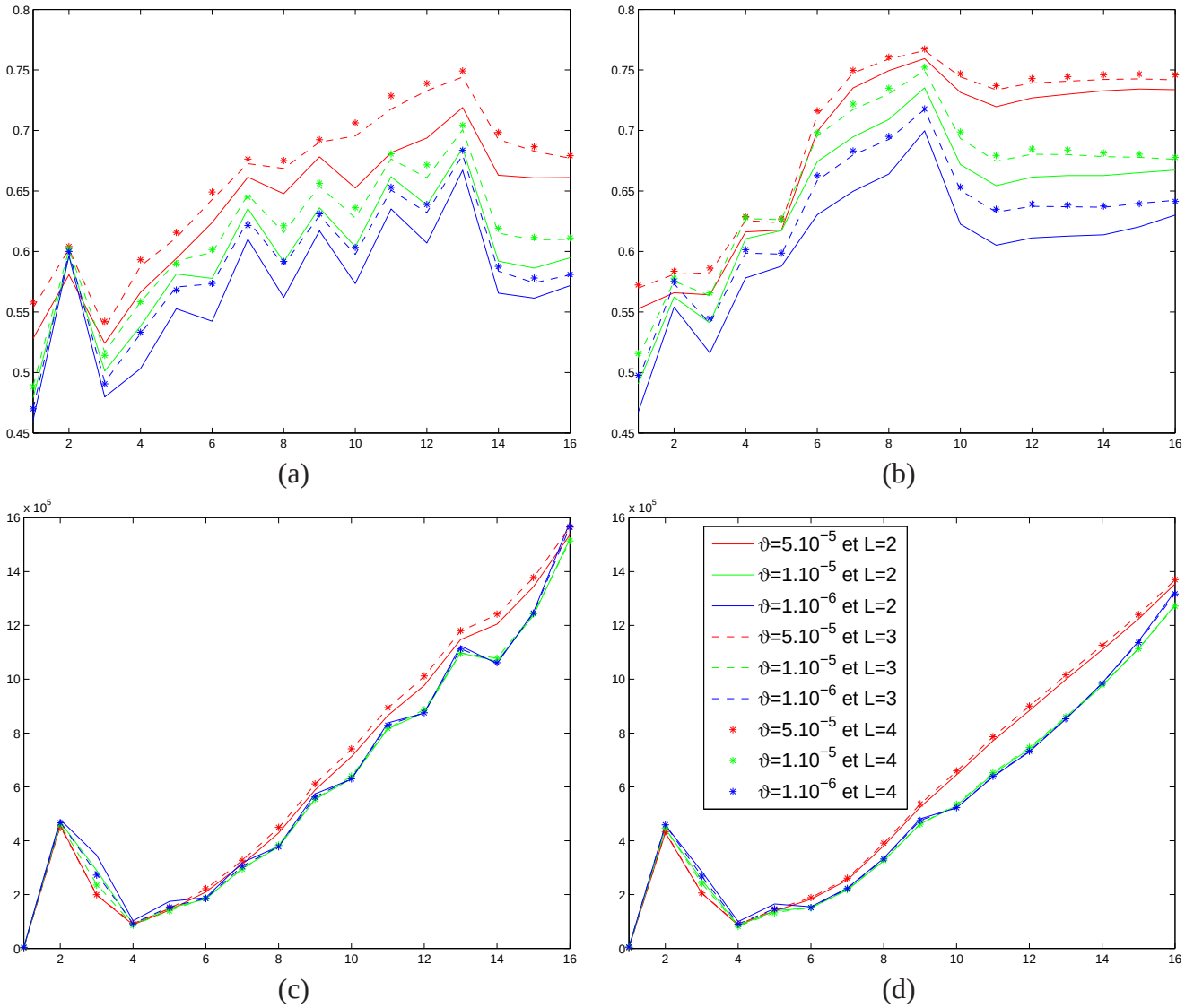


FIGURE 7.13 – Valeurs de SSIM et EQM pour les $T = 16$ trames temporelles pour différentes valeurs de l (nombre de niveaux de décomposition temporels), de L (nombre de niveaux de décomposition spatial) et de ϑ (influence du terme de variation totale) pour des données reconstruites à partir du sinogramme 1. (a) valeurs de SSIM pour $l = 1$, (b) valeurs de SSIM pour $l = 2$, (c) valeurs de SNR pour $l = 1$ et (d) valeurs de SNR pour $l = 2$.

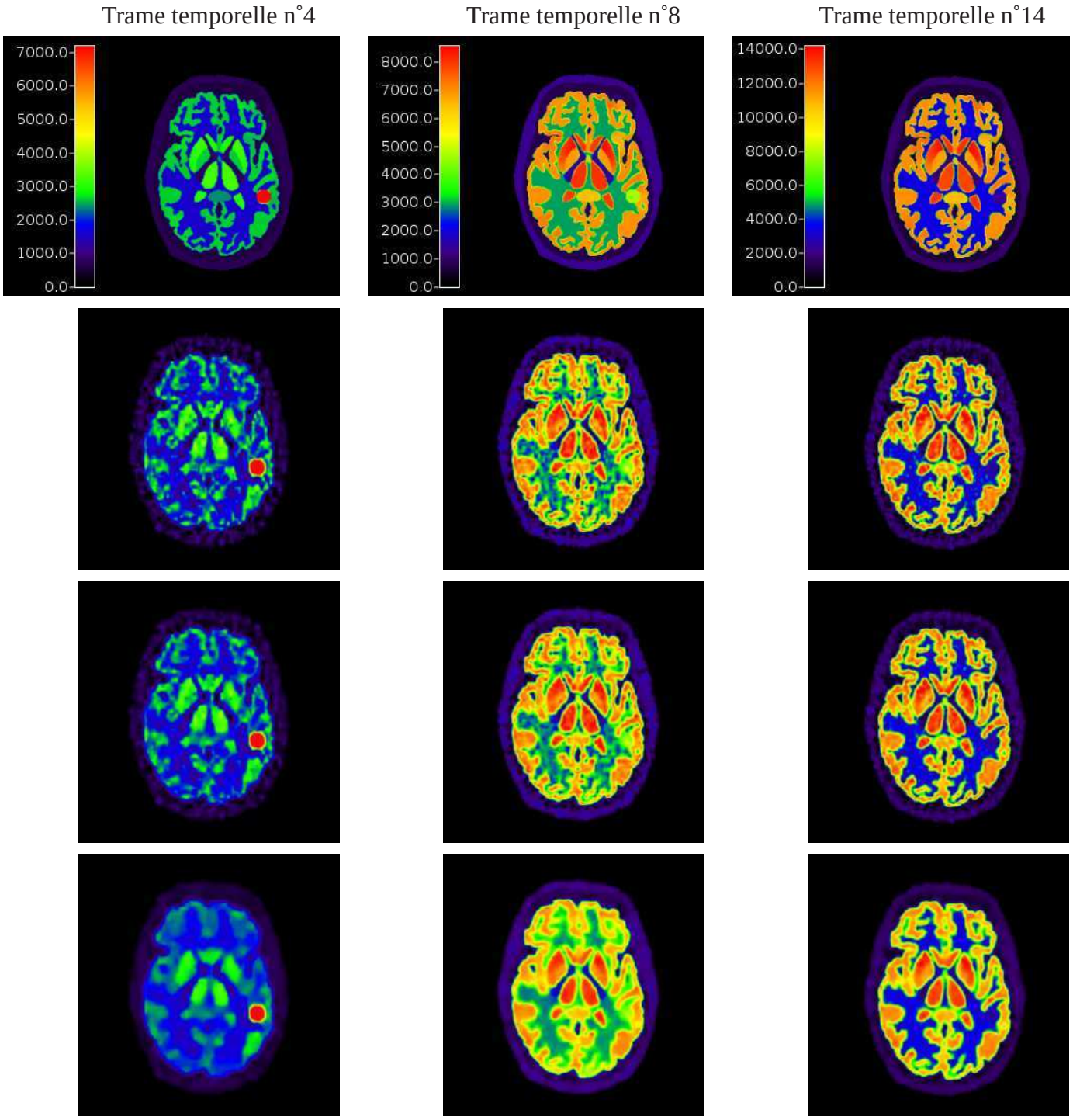


FIGURE 7.14 – Comparaison des activités reconstruites (en Bq/cc) à partir du sinogramme 1 pour les trames temporelles 4, 8, et 14 en fonction du paramètre de régularisation ϑ. La première ligne correspond à l'activité de référence pour chacune de trames considérées, la seconde ligne présente l'activité reconstruite pour $\vartheta = 10^{-5}$, la troisième ligne celle pour $\vartheta = 10^{-4}$ et la dernière considère le cas où $\vartheta = 5 \times 10^{-4}$.

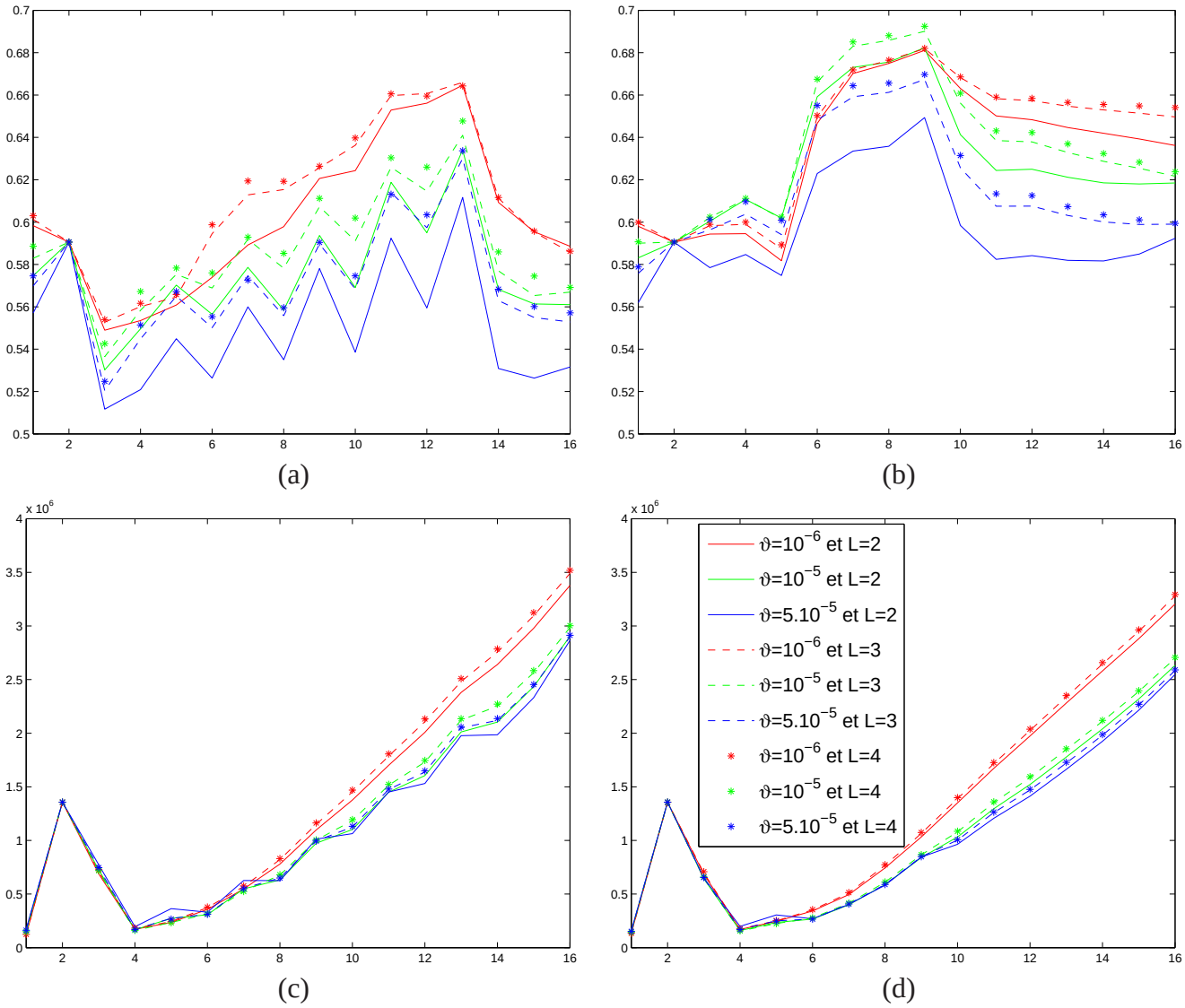


FIGURE 7.15 – Valeurs de SSIM et EQM pour les différentes trames temporelles pour différentes valeurs de l (nombre de niveaux de décomposition temporels), de L (nombre de niveaux de décomposition spatial) et de ϑ (influence du terme de variation totale) pour des données reconstruites à partir du sinogramme 2. (a) valeurs de SSIM pour $l = 1$, (b) valeurs de SSIM pour $l = 2$, (c) valeurs de SNR pour $l = 1$ et (d) valeurs de SNR pour $l = 2$.

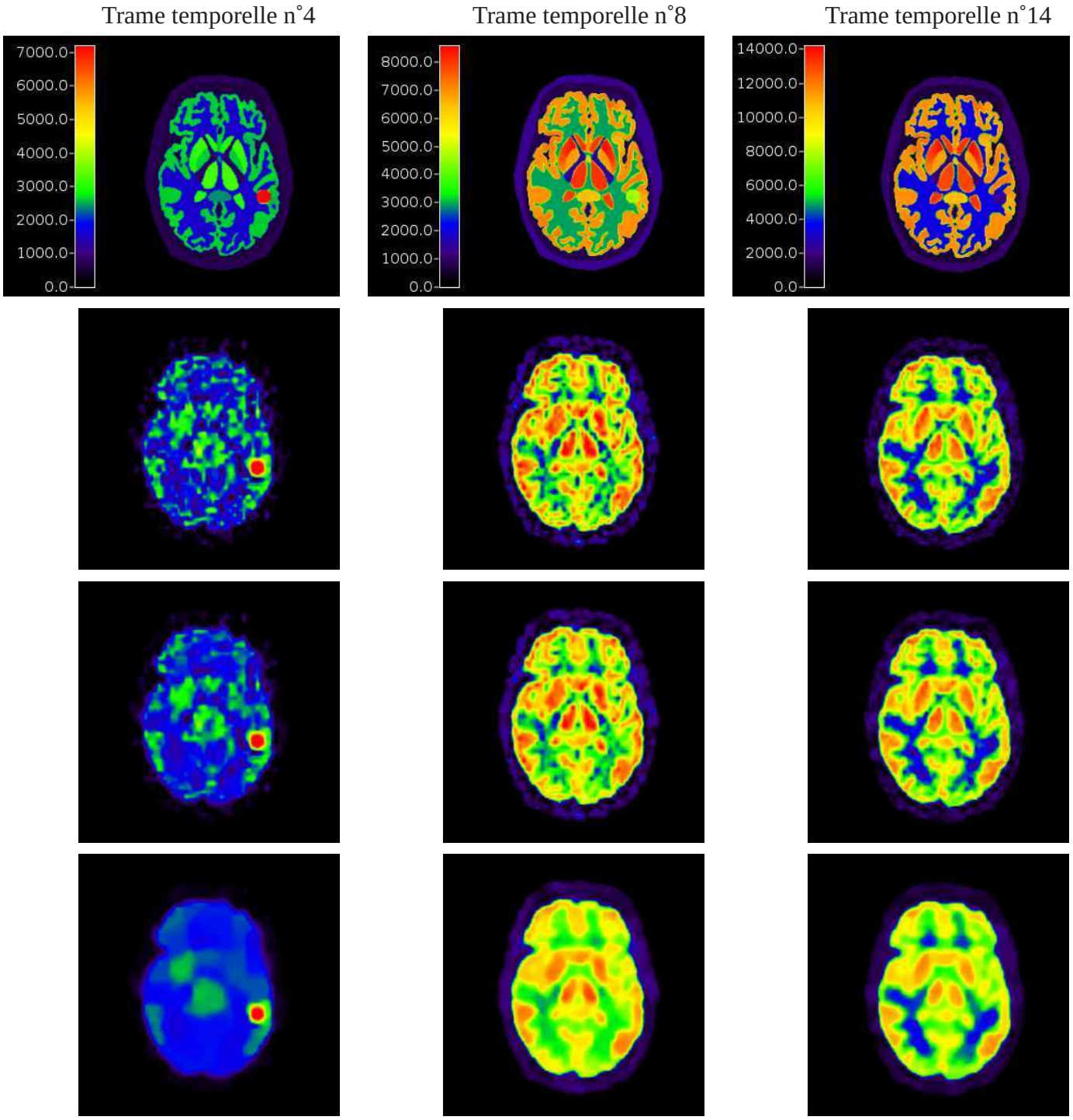
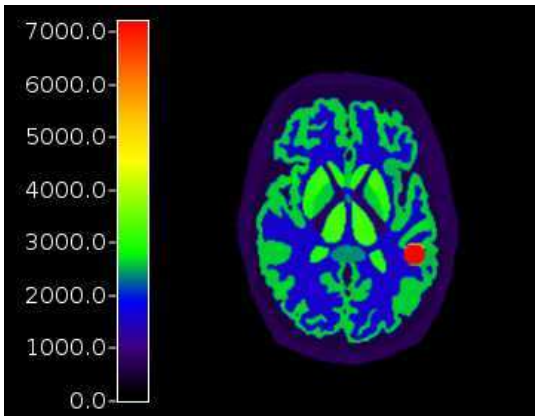
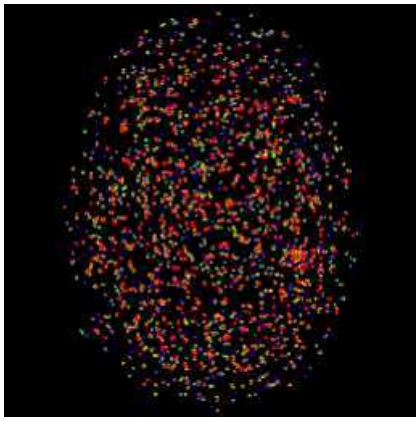


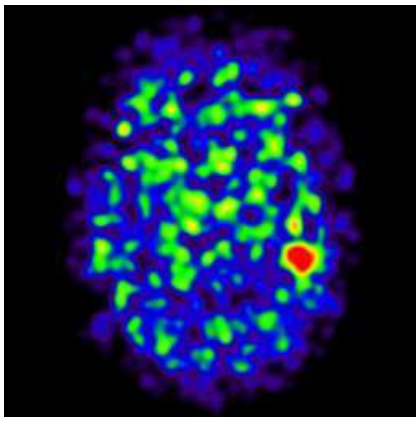
FIGURE 7.16 – Comparaison des activités reconstruites (en Bq/cc) à partir du sinogramme 2 pour les trames temporelles 4, 8, et 14 en fonction du paramètre de régularisation ϑ. La première ligne correspond à l'activité de référence pour chacune de trames considérées, la seconde ligne présente l'activité reconstruite pour $\vartheta = 10^{-5}$, la troisième ligne celle pour $\vartheta = 10^{-4}$ et la dernière considère le cas où $\vartheta = 5 \times 10^{-4}$



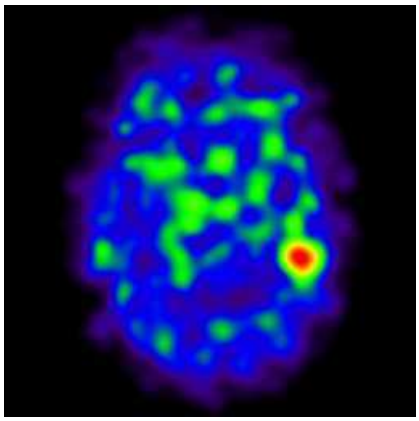
Activité originale



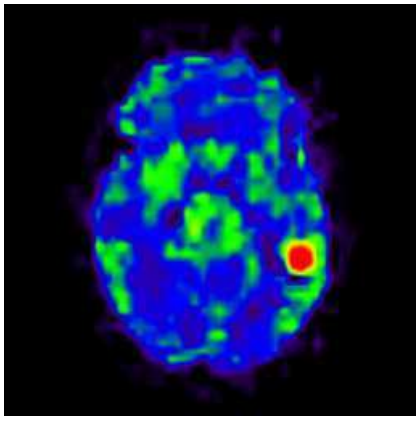
Activité reconstruite par EM-ML



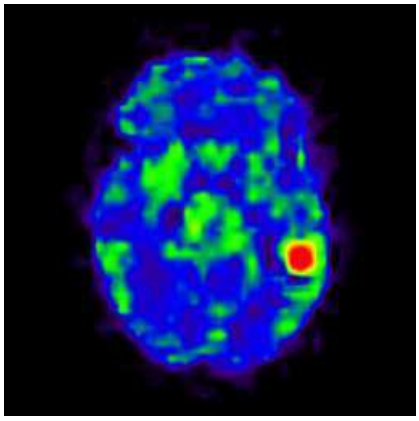
Activité reconstruite avec SIEVES 1



Activité reconstruite avec SIEVES 2

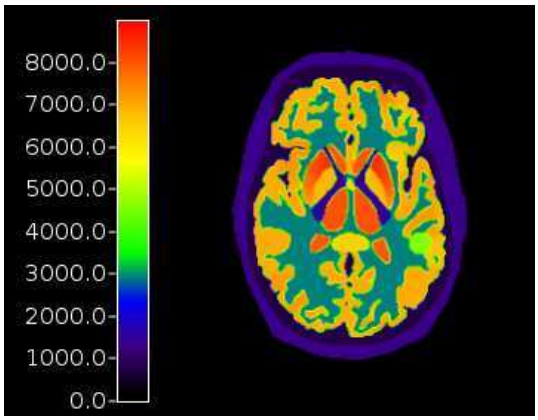


Activité reconstruite avec PPXA 1

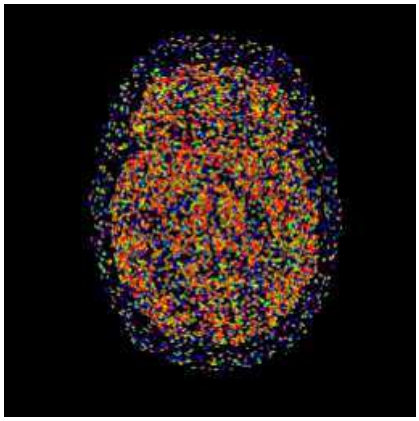


Activité reconstruite avec PPXA 2

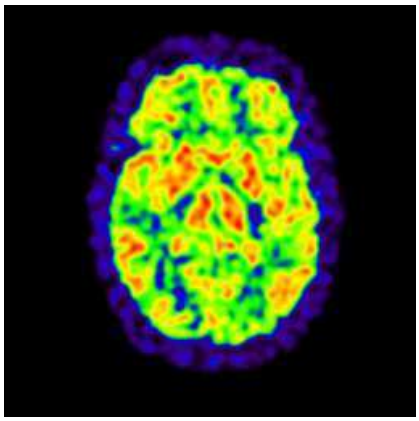
FIGURE 7.17 – Comparaison des activités reconstruites (en Bq/cc) par différentes approches pour la trame temporelle $t = 4$.



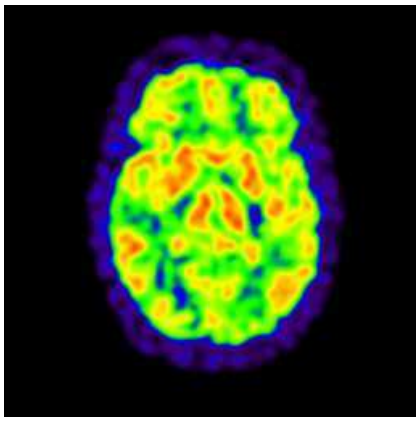
Activité originale



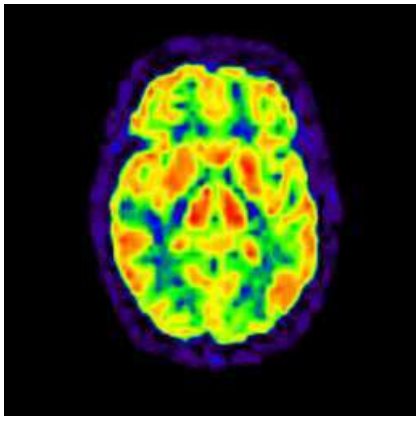
Activité reconstruite par EM-ML



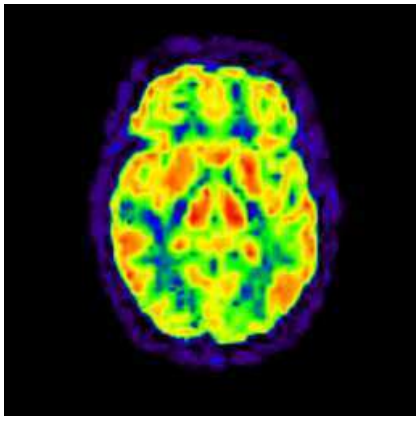
Activité reconstruite avec SIEVES 1



Activité reconstruite avec SIEVES 2

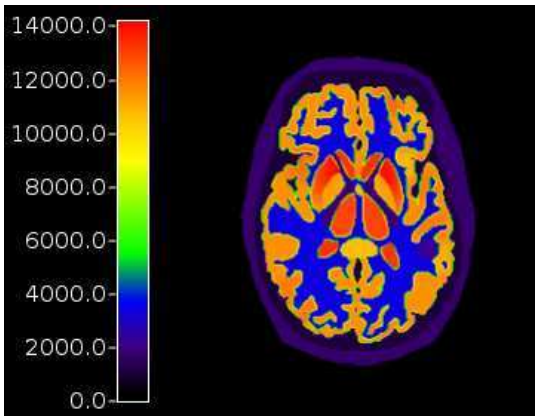


Activité reconstruite avec PPXA 1

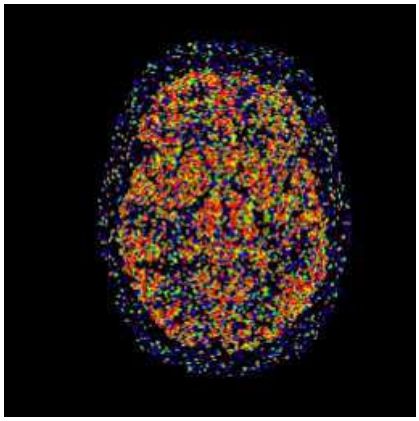


Activité reconstruite avec PPXA 2

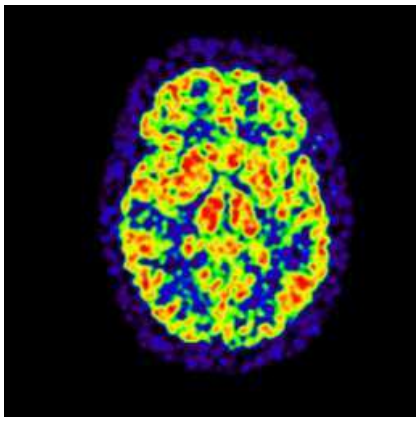
FIGURE 7.18 – Comparaison des activités reconstruites (en Bq/cc) par différentes approches pour la trame temporelle $t = 8$.



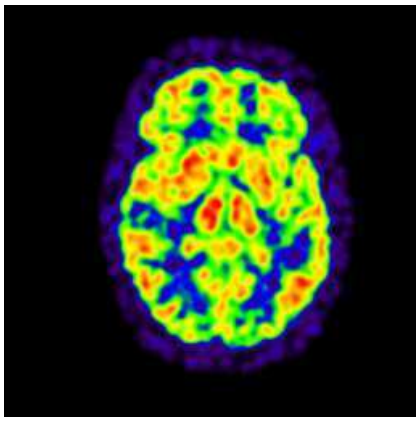
Activité originale



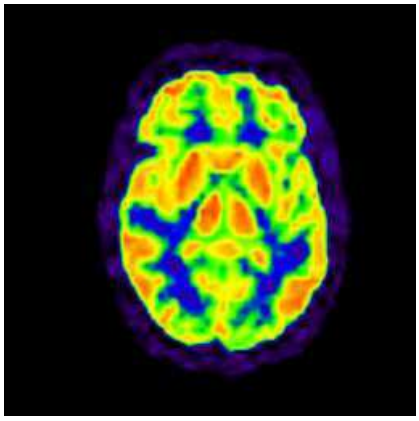
Activité reconstruite par EM-ML



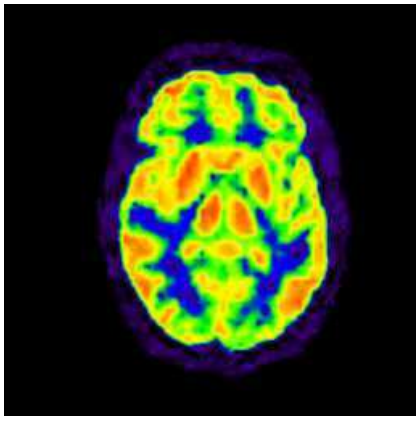
Activité reconstruite avec SIEVES 1



Activité reconstruite avec SIEVES 2



Activité reconstruite avec PPXA 1



Activité reconstruite avec PPXA 2

FIGURE 7.19 – Comparaison des activités reconstruites (en Bq/cc) par différentes approches pour la trame temporelle $t = 14$.

Cortex					
	EM-ML	SIEVES 1	SIEVES 2	PPXA 1	PPXA 2
$t = 4$	6.48×10^7	2.08×10^6	1.55×10^6	$\mathbf{4.92 \times 10^5}$	5.03×10^5
$t = 8$	8.38×10^7	3.23×10^6	3.26×10^6	$\mathbf{2.13 \times 10^6}$	2.42×10^6
$t = 14$	1.69×10^8	1.02×10^7	1.07×10^7	$\mathbf{8.44 \times 10^6}$	8.99×10^6
Thalamus					
	EM-ML	SIEVES 1	SIEVES 2	PPXA 1	PPXA 2
$t = 4$	8.89×10^7	2.60×10^6	1.88×10^6	5.51×10^5	$\mathbf{5.27 \times 10^5}$
$t = 8$	9.54×10^7	3.51×10^6	3.38×10^6	$\mathbf{2.25 \times 10^6}$	2.63×10^6
$t = 14$	2.47×10^8	8.88×10^6	8.64×10^6	$\mathbf{7.73 \times 10^6}$	8.17×10^6
Artère					
	EM-ML	SIEVES 1	SIEVES 2	PPXA 1	PPXA 2
$t = 4$	5.62×10^8	1.43×10^7	$\mathbf{1.41 \times 10^7}$	1.47×10^7	1.48×10^7
$t = 8$	3.85×10^7	4.83×10^5	$\mathbf{3.25 \times 10^5}$	3.96×10^5	4.00×10^5
$t = 14$	5.91×10^7	2.54×10^6	2.53×10^6	$\mathbf{2.46 \times 10^6}$	2.58×10^6
Matière blanche					
	EM-ML	SIEVES 1	SIEVES 2	PPXA 1	PPXA 2
$t = 4$	5.38×10^7	1.51×10^6	1.08×10^6	$\mathbf{3.01 \times 10^5}$	3.14×10^5
$t = 8$	3.93×10^7	1.95×10^6	$\mathbf{1.89 \times 10^6}$	1.92×10^6	1.90×10^6
$t = 14$	6.26×10^7	$\mathbf{6.03 \times 10^6}$	6.35×10^6	6.81×10^6	6.98×10^6

TABLE 7.2 – EQM pour différentes régions d'intérêt.

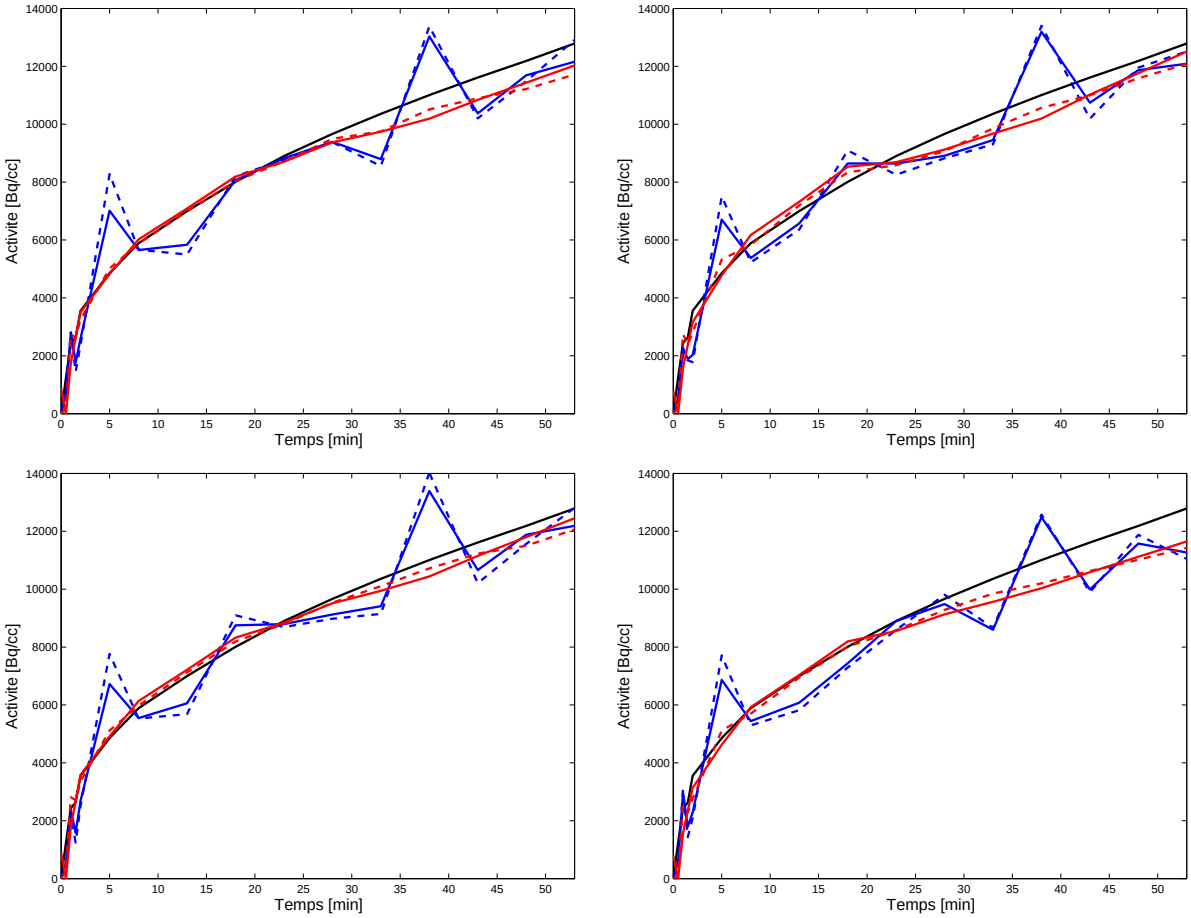


FIGURE 7.20 – CAT extraites dans le cortex pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue).

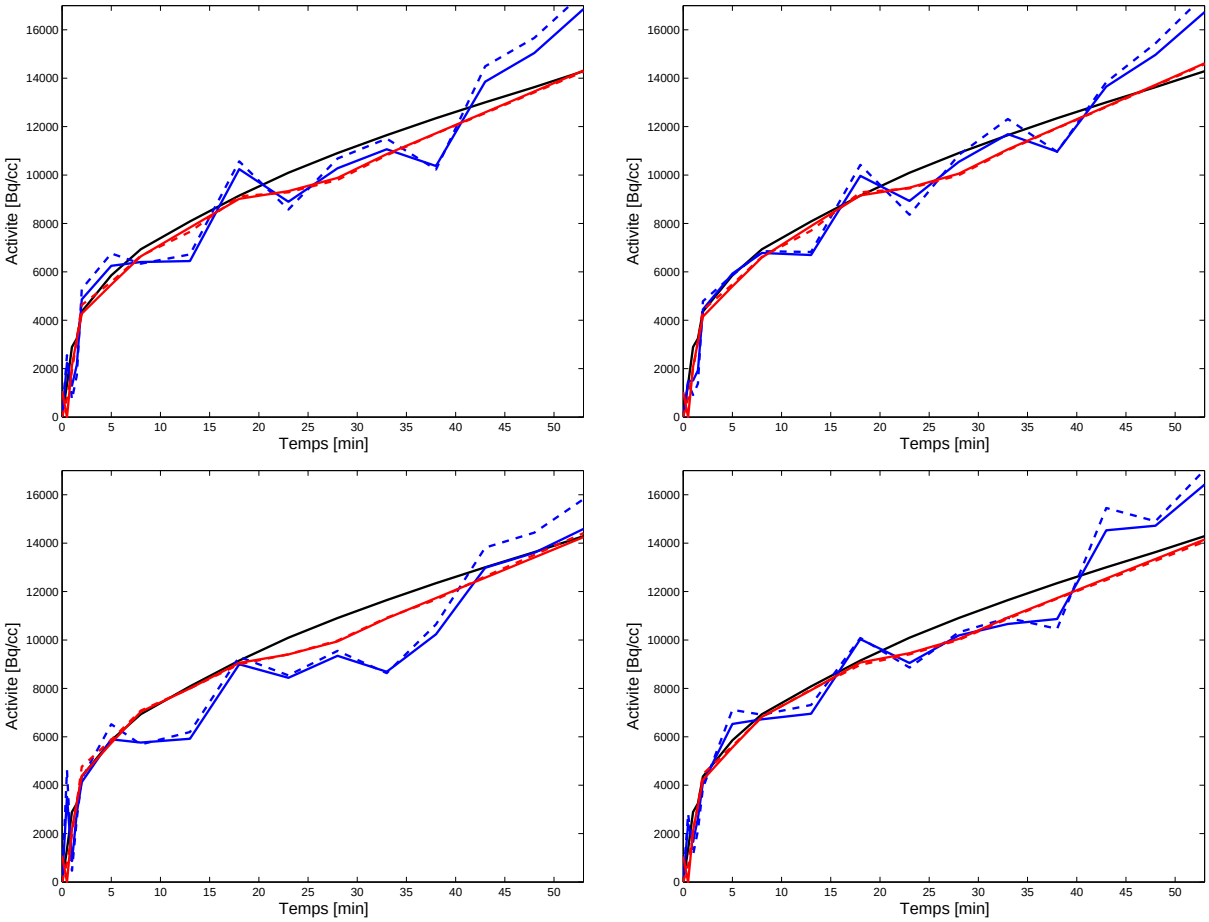


FIGURE 7.21 – CAT dans les noyaux gris centraux pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue)

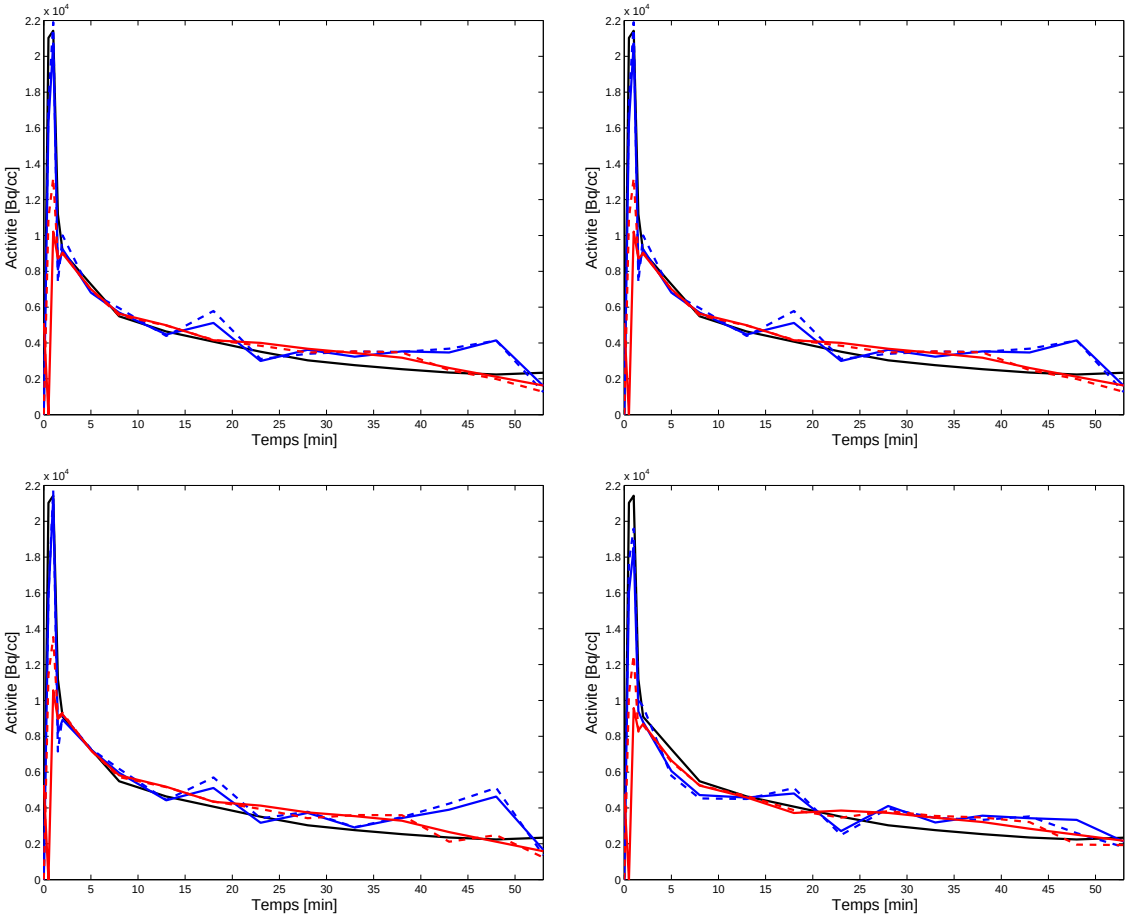


FIGURE 7.22 – CAT dans l'artère ajoutée pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue)

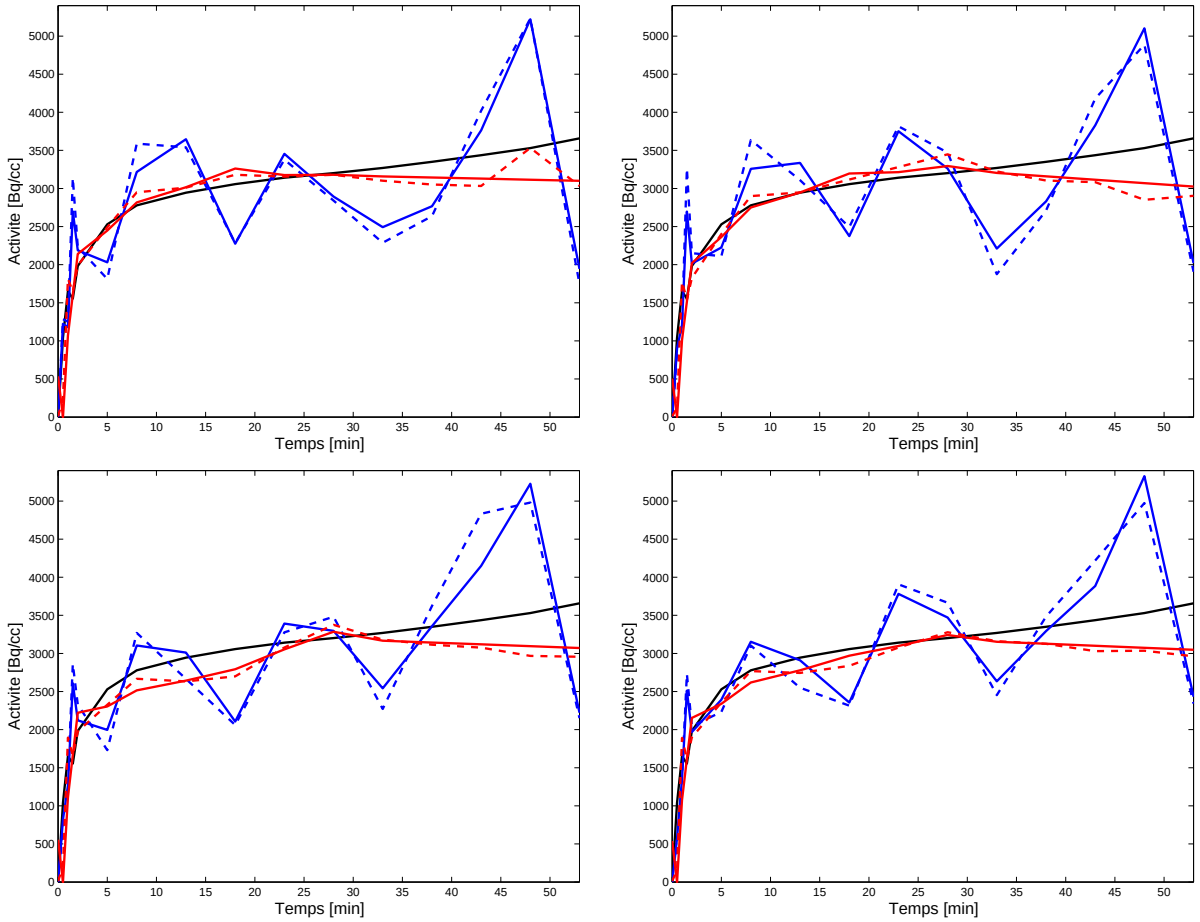
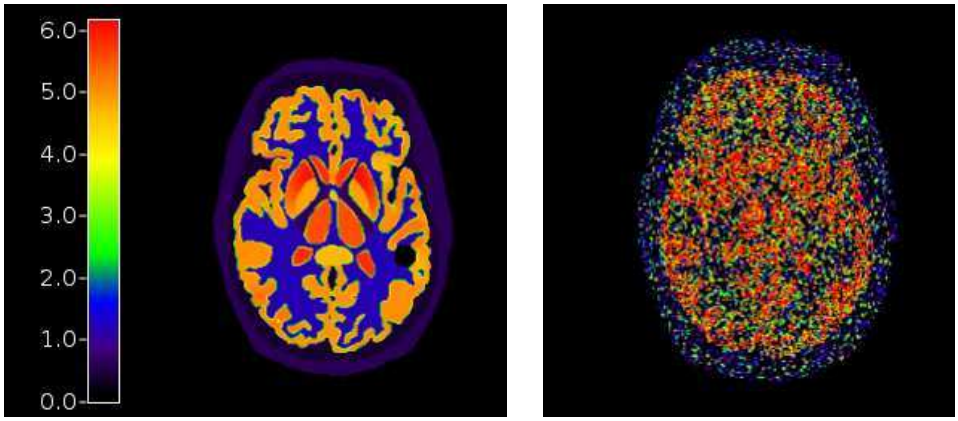
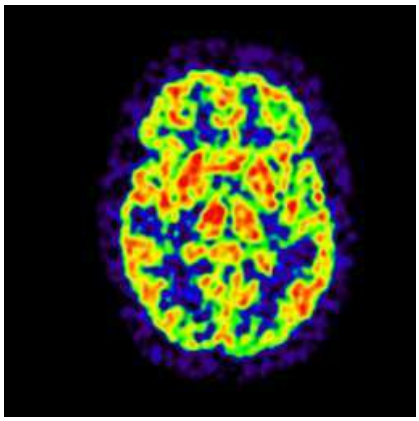


FIGURE 7.23 – CAT dans la matière blanche pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue)

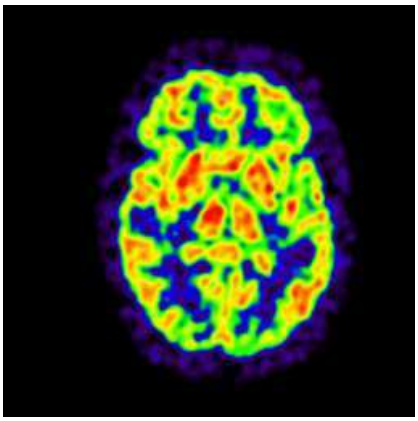


rCMRglu original

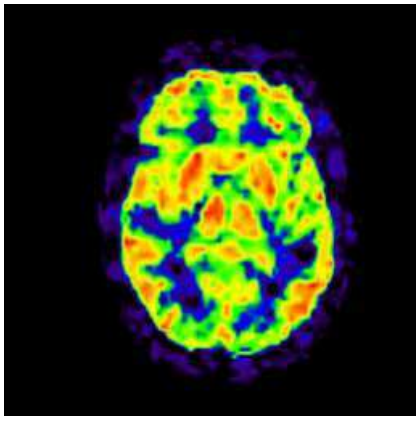
rCMRglu avec EM-ML



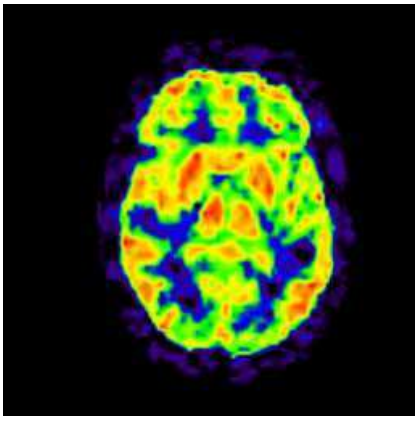
rCMRglu avec SIEVES 1



rCMRglu avec SIEVES 2



rCMRglu avec PPXA 1



rCMRglu avec PPXA 2

FIGURE 7.24 – Cartes paramétriques du taux de métabolisme cérébral régional du glucose (en μmol/min/100g).

7.4.3 Résultat expérimentaux sur données réelles

Nous avons également évalué notre méthode sur des données réelles. Pour cela nous avons traité des données provenant d'un examen [^{18}F]-FDG, où une activité de 284 MBq a été injectée à un babouin de 4.5 kg. L'injection s'est produite sous caméra ce qui signifie que le temps $t = 0$ de l'injection correspond au début de l'acquisition. Une acquisition en mode (2D+t), avec les *septa* [4] sortis, a été réalisée. Les coïncidences vraies et retardées ont été acquises séparément. Nous avons acquis 128 trames d'une durée variant de 10 secondes à 30 secondes. La durée totale de l'examen est de 54 minutes. La transmission a été réalisée avant injection par une source rotative de germanium 57 ; cette procédure permettra de corriger les effets d'atténuation durant la reconstruction. Des prélèvements artériels ont été réalisés afin de connaître la fonction d'entrée et ainsi être capable de mesurer le taux de métabolisme cérébral régional du glucose (rCMRglu).

Nous avons procédé à la reconstruction de ces données réelles par rétroprojection filtrée (logiciel du constructeur à savoir SIEMENS), par la méthode SIEVES et par notre approche (PPXA). Pour chacune d'entre elles, les effets d'atténuations, de normalisation et les coïncidences fortuites sont pris en compte lors de la reconstruction. Cependant, nous ne tenons pas compte des coïncidences diffusées (nous justifions ce choix par le fait qu'il s'agit d'une acquisition 2D+t). La matrice système utilisée dans les reconstructions SIEVES et PPXA modélise uniquement les effets géométriques [5]. Nous présentons sur la figure 7.25 les images obtenues par les différentes méthodes aux instants $t = 4$, $t = 8$ et $t = 14$.

7.4.4 Conclusion

Dans ce chapitre nous avons utilisé successivement les algorithmes imbriqués et PPXA pour résoudre le problème de reconstruction d'images TEP dynamiques. Pour des raisons de temps de calculs nous avons préféré PPXA aux algorithmes imbriqués. Par ailleurs, PPXA nous a permis de minimiser efficacement un critère complexe composé de la divergence de Kullback-Leibler, de la variation totale, d'un terme favorisant la parcimonie et d'une contrainte sur la dynamique. Ainsi nous avons pu valider notre approche par rapport aux méthodes classiquement utilisées en TEP. Les résultats sur les données simulées sont très encourageants. Un objectif à court terme est d'observer le comportement des différents algorithmes pour un découpage temporel plus fin (32 ou 64 trames temporelles). L'implantation de PPXA pour restaurer des données réelles et sa comparaison aux méthodes de référence, en particulier la méthode du constructeur, démontre que notre approche est quantitativement correcte pour des données réelles.

4. Les septa permettent d'éliminer les photons faisant un angle avec le plan de coupe.
5. Nous ne considèrons pas les effets dégradants liés à résolution des détecteurs.

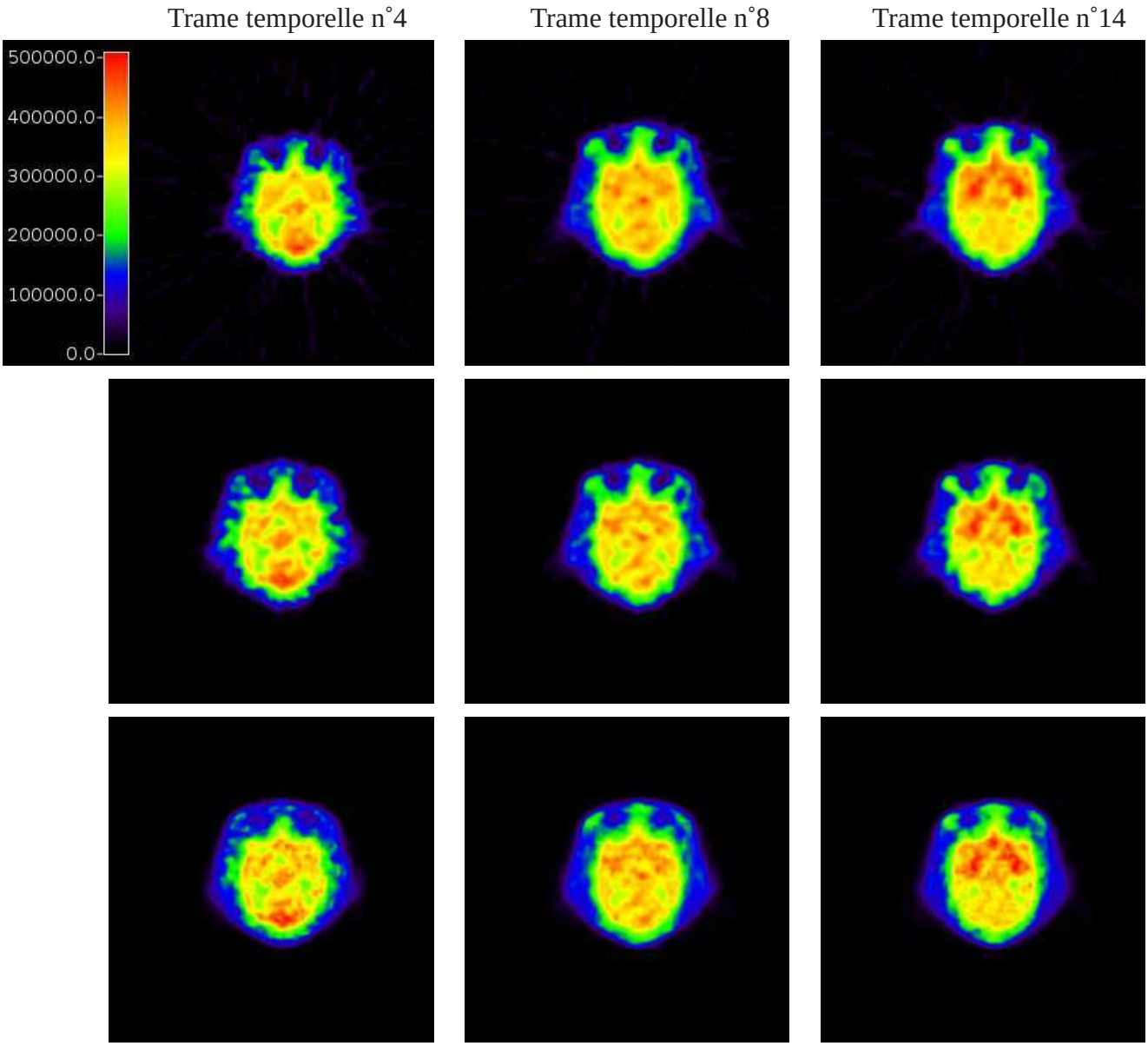


FIGURE 7.25 – Comparaison des activités reconstruites (en Bq/cc) pour la coupe 24 de l'acquisition de données sur babouin. 1ère ligne : SIEMENS, 2ème ligne : SIEVES, 3ème ligne : PPXA.

CHAPITRE 8

Conclusion et perspectives

8.1 Bilan

Cette thèse a conduit au développement de nouvelles méthodes pour la résolution de problèmes inverses de grandes tailles. Ces méthodes offrent une grande flexibilité dans le choix du critère à minimiser, ce qui est un avantage par rapport à certaines méthodes actuellement employées. Par ailleurs, elles apportent des garanties de convergence.

Au début de cette thèse, l'algorithme proximal explicite-implicite et ses déclinaisons [Daubechies *et al.*, 2004; Bect *et al.*, 2004; Combettes, Wajs, 2005; Weiss *et al.*, 2009] commencaient à être populaires au sein de la communauté du traitement d'image pour résoudre des problèmes de déconvolution en présence de bruit gaussien ou des problèmes d'échantillonnage comprimé. En effet, cet algorithme permet de gérer efficacement un critère composé d'un terme quadratique (plus généralement une fonction de gradient Lipschitz) et d'une fonction pouvant être non différentiable (comme une norme ℓ_1, souvent appliquée à des coefficients de trame afin d'en favoriser la parcimonie). Un autre algorithme proximal connu sous le nom d'algorithme de Douglas-Rachford a également été présenté dans [Combettes, Pesquet, 2007a] pour relâcher la contrainte de Lipschitz différentiabilité et ainsi traiter des problèmes inverses comme le débruitage en présence de bruit de Poisson. Cet algorithme a montré son efficacité pour le débruitage des courbes d'activités temporelles en TEP [Sureau *et al.*, 2008]. Contrairement à l'algorithme explicite-implicite, l'algorithme de Douglas-Rachford ne permet pas de gérer aussi facilement des opérateurs linéaires tels qu'un opérateur de convolution ou de projection, ce qui est problématique si l'on veut résoudre efficacement un problème de reconstruction en TEP.

Notre première contribution a donc été de résoudre ce problème en combinant une classe d'algorithmes imbriqués et une extension quadratique permettant de gérer des fonctions deux fois différentiables sur leur domaine, mais pouvant ne pas être de gradient Lipschitz, telle que l'antilog-vraisemblance poissonienne. Pour mener à bien ce travail, nous avons développé une classe d'algorithmes proximaux imbriqués basés sur les algorithmes explicite-implicite, de Douglas-Rachford et de type Dykstra. Les preuves de convergence de cette classe d'algorithmes ont été données. Pour étendre le champ d'application de ces méthodes à des fonctions convexes n'ayant pas de gradient Lipschitz (et ainsi résoudre le problème de restauration/reconstruction en présence de bruit de Poisson), nous avons proposé une approximation inférieure du critère basée sur l'utilisation d'une extension

quadratique. Cette dernière fait intervenir un paramètre θ qui permet d'ajuster la proximité entre l'approximation inférieure et la fonction de départ. Cependant, il faut remarquer que plus ce paramètre est grand, meilleure est l'approximation, au prix d'une diminution de la vitesse de convergence. Nous avons montré, sous certaines hypothèses techniques précisées dans la proposition 3.42, qu'il existe une valeur de θ au delà de laquelle la solution du critère approximé est la même que la solution du critère originel. Le bon comportement de cette approximation comparée aux approches constituant l'état de l'art a également été démontré. L'essentiel des résultats de ce chapitre sont présentés dans [Chaux *et al.*, 2009].

La méthode précédemment proposée souffre de temps de calculs relativement longs, dûs aux sous-itérations, et ce malgré le soin apporté à l'initialisation qui vise à réduire leur nombre. Pour une application à la reconstruction de données TEP dynamique, où les volumes de données sont plus importants que pour des problèmes usuels de déconvolution d'images, cette méthode ne répond pas à l'un des trois objectifs (fiabilité, efficacité et rapidité) que nous nous étions fixés, à savoir la rapidité. Nous avons alors orienté notre travail vers l'utilisation de l'algorithme proximal parallèle [Combettes, Pesquet, 2008] permettant de minimiser une somme finie de fonctions. Cette méthode possède comme principal avantage d'avoir une structure hautement parallélisable mais elle souffre, de la même manière que l'algorithme de Douglas-Rachford (dont elle constitue une extension dans un espace produit), de la difficulté que pose le calcul de l'opérateur proximal d'une fonction composée avec un opérateur linéaire. Nous avons proposé dans le chapitre 4 une décomposition permettant de résoudre ce problème lorsque l'opérateur linéaire est un opérateur de convolution ou un opérateur de convolution décimée (intervenant en super-résolution). Il faut remarquer que ce type d'approche par décomposition permet de résoudre des problèmes différents de ceux résolus avec ADMM (Alternative Direction Method of Multipliers) qui requiert une inversion matricielle lourde à réaliser, notamment dans des approches de type super-résolution ou en TEP. De façon à améliorer la qualité des images restaurées/reconstruites en présence d'un bruit non nécessairement additif gaussien, nous avons inclus une régularisation hybride ($\mathrm{tv}+\ell_1$) dans notre critère . Ce type de régularisation auparavant utilisée en présence de bruit additif gaussien devient d'un intérêt accru pour d'autres types de perturbations.

L'utilisation de trames pour la résolution de problèmes inverses soulève le problème du choix entre formulation à l'analyse et formulation à la synthèse. En effet, l'emploi de trames se traduit par la présence de l'opérateur d'analyse de trame ou de l'opérateur de synthèse de trame dans le critère à minimiser. Une première étude théorique a été menée dans [Elad *et al.*, 2007]. Nous avons étendu ces résultats dans le chapitre 5 en explorant les liens entre formulation à l'analyse et formulation à la synthèse.

Ce chapitre présente également plusieurs techniques qui permettent de gérer efficacement des problèmes pouvant être rencontrés lors de l'utilisation des algorithmes proximaux. Tout d'abord, une alternative au calcul de l'opérateur proximal d'une fonction

composée avec un opérateur linéaire a été présentée. Cette dernière se base sur la dualité de Moreau-Fenchel-Rockafellar [Combettes *et al.*, 2009]. Nous avons aussi décrit un algorithme itératif permettant de calculer la valeur de la norme d'un opérateur linéaire quand celle-ci n'est pas explicite. Enfin, nous avons proposé d'étendre l'utilisation de certains algorithmes proximaux à une classe de trames non-ajustées. Actuellement, la gestion de telles trames pose problèmes avec des algorithmes tels que l'algorithme de Douglas-Rachford ou l'algorithme proximal parallèle.

La régularisation hybride $\mathrm{tv}+\ell_1$ proposée dans le chapitre 4, nous a conduit à un autre type de régularisation qui nous semblait mieux adapté, à savoir une régularisation tv appliquée sur la partie géométrique de l'image et une régularisation favorisant la parcimonie des coefficients de trame sur la composante texturée de l'image. Ce type d'approche nécessite une formulation multicomposante. Nous avons donc proposé dans le chapitre 6 une extension des algorithmes proximaux et de leurs propriétés de convergence à ce cadre d'étude. Nous avons également fourni des formes explicites d'opérateurs proximaux multicomposantes et avons illustré notre propos par diverses applications à des problèmes inverses de grande taille entrant dans ce cadre d'étude.

Notre dernière contribution est l'adaptation des méthodes proposées dans le cadre de la déconvolution en présence de bruit de Poisson au cas de la reconstruction de données dynamiques TEP, qui constituait l'objectif applicatif principal de la thèse. Ce travail s'inscrit dans la suite des travaux de [Sureau *et al.*, 2008] et permet de gérer simultanément débruitage et rétroprojection à l'aide des algorithmes proximaux. Dans le chapitre 7, nous avons donc successivement présenté les résultats obtenus avec les algorithmes imbriqués et PPXA. Nous avons pu en particulier observer l'intérêt d'utiliser une régularisation spatio-temporelle basée sur les ondelettes et d'ajouter un terme de variation totale permettant aux médecins d'obtenir des résultats plus ou moins lissés.

8.2 Quelques perspectives

Dans l'état actuel de son développement, ce travail ouvre un certain nombre de perspectives tant théoriques qu'applicatives.

- **Optimisation du paramètre θ de l'extension quadratique**

Dans le chapitre 3, nous avons vu que le choix du paramètre θ lié à l'extension quadratique permettait d'assurer un compromis entre précision de l'approximation et vitesse de convergence. Dans notre approche, nous choisissons un paramètre θ ne variant pas au cours des itérations mais une perspective intéressante serait d'adapter à chaque itération $n \in \mathbb{N}$

la valeur de ce paramètre, noté alors θ_n, de façon à optimiser le taux de convergence de l'algorithme, par exemple en choisissant un θ grossier pour les premières itérations et en affinant ce choix petit à petit. Un choix automatique pourrait également être envisagé en s'inspirant des méthodes d'adaptation de pas de Barzilai-Borwein [Barzilai, Borwein, 1988].

- **Version accélérée de PPXA**

Les chapitres 4 et 7 montrent l'intérêt d'utiliser l'algorithme proximal parallèle pour résoudre des problèmes inverses comme la restauration ou la reconstruction. Cependant, pour utiliser l'algorithme PPXA en routine clinique en TEP il faudrait être capable d'accélerer cet algorithme. En effet, rappelons que l'EM a été implanté en routine clinique lorsque des versions accélérées du type OSEM ou RAMLA ont été proposées. L'accélération de PPXA peut passer par l'optimisation de paramètres tels que celui de relaxation, mais on pourrait également envisager un raisonnement similaire à RAMLA en considérant à chaque itération des sous-ensembles de pixels ou encore exploiter les liens existants avec les méthodes de minimisation alternée (Gauss-Seidel).

- **Choix du terme de régularisation**

L'utilisation d'une régularisation hybride a permis d'améliorer la qualité de reconstruction dans les chapitres 4, 6 et 7. Il serait donc d'un intérêt certain de réfléchir à des régularisations multiples encore mieux adaptées. Si l'on se réfère à l'équation (5.12) où $x \in \mathbb{R}^K$ représente les coefficients de trame tels que, pour tout $(y, x_\perp) \in \mathbb{R}^N \times \ker F^\top$, $x = Fy + x_\perp$, on pourrait par exemple envisager une régularisation différente sur Fy et $x_\perp$.

Toujours dans l'optique d'obtenir une régularisation plus pertinente, il faut remarquer que les méthodes proximales requièrent la plupart du temps des trames ajustées. Nous avons proposés dans le chapitre 5 un cadre d'étude permettant d'étendre leur application à des trames pouvant être non ajustées. En élargissant les hypothèses formulée dans la section 5.4.1, d'autres trames pourraient être envisagées.

- **Application à l'imagerie radar à synthèse d'ouverture (SAR)**

En imagerie SAR, une des problématiques est d'augmenter la résolution en éliminant simultanément le bruit de Speckle, de type multiplicatif. Les travaux existants visent pour l'essentiel à effectuer l'opération de débruitage seule. L'idée serait donc d'explorer les approches variationnelles développées dans le cas du débruitage et de les adapter au cas où il s'agit d'accroître la résolution et de débruiter l'image conjointement. Pour résoudre le seul problème de débruitage de manière efficace par des approches variationnelles, le terme d'attache aux données peut être choisi de diverses façons. La première consiste à considérer le logarithme des données qui permet de se rapprocher d'un bruit blanc gaussien [Durand *et al.*, 2010] et ainsi de se ramener à un terme d'attache aux données quadratique ; cette approche possède l'inconvénient d'introduire un biais dans l'image

reconstruite. Une seconde méthode a été suggérée dans [Combettes, Pesquet, 2007a] en utilisant un modèle de bruit simplifié. En restant dans un cadre convexe ou en faisant une incursion en territoire non convexe [Aubert, Aujol, 2008], il serait certainement envisageable de proposer d'autres approximations plus performantes.

- **Applications multicomposantes**

Le chapitre sur l'extension des algorithmes proximaux aux problèmes multicomposantes permet d'envisager de nombreuses applications telles que l'extraction de différentes textures. Remarquons que le préalable à ce travail est de bien définir ce qui est sous-entendu par la notion de texture et la variété que peut recouvrir une telle notion. On peut ainsi envisager des régularisations permettant de modéliser différents types de textures. Enfin, on peut songer à des extensions pour la résolution de problèmes inverses en imagerie couleur.

- **Validation de PPXA Hybride pour des données dynamiques TEP**

De façon à pouvoir exploiter un algorithme tel que PPXA pour la résolution de problèmes de reconstruction dynamique TEP en routine clinique, il serait souhaitable :

  – d'accélérer les temps de calcul en s'inspirant, comme évoqué précédemment, de RAMLA et/ou en effectuant une implantation à l'aide de GPU,
  – de valider la procédure de reconstruction auprès de médecins.

Cela nécessite encore un certain travail de mise en œuvre et de validation mais les résultats déjà obtenus nous encouragent à réaliser cet investissement.

Table des figures

1.1 Coupes 2D de données dynamiques acquises (gauche) et de l'activité reconstruite (en Bq/cm^3) par la méthode actuellement utilisée en routine clinique (droite) sur un examen [^{1}8F]-FDG pour différentes trames temporelles. 17

2.1 Dégradation en TEP. 24
2.2 Dégradation en imagerie satellitaire. 25
2.3 (a) Image médicale, (b) transformée en ondelettes sur 1 niveau et (c) transformée en ondelettes sur 2 niveaux de résolution. 31
2.4 (a) Figure 2.3(b) et (b) histogramme associé à la sous-bande de détails horizontaux. 33
2.5 Représentation graphique de $I(\cdot, u, w)$ pour trois différentes valeurs de w. 36
2.6 Comparaison des histogrammes de bruit pour $\alpha = 0.1$ lorsque $\overline{y}$ correspond à l'image de Lena. 49
2.7 Comparaison des histogrammes de bruit pour $\alpha = 0.02$ lorsque $\overline{y}$ correspond à l'image de Lena. 49
2.8 Comparaison des différentes méthodes de débruitage en présence de bruit de Poisson pour une dégradation avec un paramètre d'échelle fixé à $\alpha = 0.1$. 53
2.9 Comparaison des différentes méthodes de débruitage en présence de bruit de Poisson pour une dégradation avec un paramètre d'échelle fixé à $\alpha = 0.02$. 54

3.1 Evolution de $f + g$ en fonction (a) des itérations et (b) du temps CPU en secondes (Intel Core 2, 2.66Ghz). FB[DR] (ligne continue bleue), FB[Dyk] (ligne discontinue bleue) et DR[FB] (ligne discontinue noire). 90
3.2 Valeur de $\|x_n - \widehat{x}\|$ en fonction (a) des itérations et (b) du temps CPU en secondes (Intel Core 2, 2.66Ghz). FB[DR] (ligne continue bleue), FB[Dyk] (ligne discontinue bleue) et DR[FB] (ligne discontinue noire). 91
3.3 Résultats sur une image satellitaire. (a) Originale, (b) dégradée (SNR = 13.4 dB), (c) restaurée sans la contrainte (SNR = 14.8 dB) et (d) restaurée avec la contrainte (SNR = 15.0 dB). 92
3.4 Illustration de $\upsilon_i(\theta)$ pour $\theta_1 < \theta_2 < \theta_3$. 94
3.5 Représentation graphique de l'extension quadratique pour $\delta = 0$. 95
3.6 Résultats pour une image satellitaire de la ville de Marseille. (a) Image originale, (b) image dégradée, (c) image restaurée en utilisant des DTT. . 102
3.7 Critère MAP normalisé (pour $\theta = 0.15$ en vert et $\theta = 10$ en magenta) versus le temps de calcul (en secondes) (Intel Core 2, 2.66 GHz). 103

3.8 Graphe de la fonction ψ_i (ligne continue rouge) quand $\delta = 0$, $\alpha_i = 1$, $z_i = 50$. Son approximation quadratique $\psi_{\theta,i}$ avec $\theta = 0.3$ (ligne discontinue verte) et $\theta = 0.1$ (ligne discontinue bleue) pour $\epsilon(\theta) = 10^{-16}$ et son approximation de Anscombe $\widetilde{\psi}_i$ (ligne discontinue noire). 105
3.9 Résultats sur une image médicale. (a) Originale, (b) dégradée, (c) restaurée avec EM, (d) restaurée par transformée de Anscombe et (e) restaurée par extension quadratique. 107

4.1 Décomposition de l'image qui permet de calculer l'opérateur proximal de la variation totale. 125
4.2 (a) Image « Sebal » - 128×128 (source : SHFJ), (b) Image « Boat » - 256×256 (source : http ://sipi.usc.edu/database), (c) « Image Peppers » - 256×256 (source : http ://sipi.usc.edu/database), (d) Image « Spot » (source : LIGM)- 256×256 et (e) Image « Marseille » - 512×512 (source : LIGM). 136
4.3 Profils de convergence de l'algorithme 7 (ligne discontinue) et de l'algorithme 8 (ligne continue) en fonction du temps de calcul en secondes pour un flou uniforme de taille 3×3 (à gauche) et 7×7 (à droite) pour une image de taille 128×128. 137
4.4 SNR (gauche) et SSIM (droite) pour différents termes de variation totale en fonction de μ et pour deux images différentes (« Peppers » en haut et « Boat » en bas). 139
4.5 Restauration avec gestion des bords périodique (gauche) et gestion des bords « zero-padded » (droite). 140
4.6 Résultats de restauration sur un zoom de l'image « spot ». 141
4.7 Résultats de restauration pour l'image « Sebal ». 143
4.8 Résultats de restauration pour l'image « Boat ». 144
4.9 Résultats de restauration pour l'image « Peppers ». 145
4.10 Résultats de restauration pour l'image « Marseille ». 146

5.1 Images originales pour la comparaison FA versus FS. 162
5.2 Comparaison des profils de convergence lorsque $\|L_1 F^\top\|^2$ est approximé (ligne discontinue) ou non (ligne continue) en fonction des itérations. . . 163
5.3 Restauration de l'image satellitaire avec les approches à l'analyse et à la synthèse en utilisant des DTT et des GenLOTs. 166
5.4 Restauration de l'image « Barbara » avec les approches à l'analyse et à la synthèse en utilisant des DTT et des GenLOTs. 167
5.5 Versions tronquées de l'image de « Barbara ». Images restaurées avec FS et DTT. 175
5.6 FS versus FA. 176

6.1 Restauration d'images stéréoscopiques. 193

6.2 Convergence de la fonction objectif du problème 6.18 pour l'algorithme explicite-implicite (ligne continue), l'algorithme de Beck-Teboulle (ligne discontinue) et l'algorithme de Nesterov (ligne en pointillés) en fonction du nombre d'itérations. 194
6.3 Débruitage multispectral : images originales et dégradées. 198
6.4 Débruitage multispectral : images restaurées. 199
6.5 Profils de convergence de l'algorithme de type Dykstra (ligne continue) et du PPXA (ligne discontinue) en fonction du temps de calcul en secondes. 201
6.6 Résultats de décomposition et de restauration. 208
6.7 Profils de convergence de l'algorithme Douglas-Rachford (ligne continue) et du PPXA (line discontinue) en fonction du temps de calcul en secondes. 209
6.8 Images « Culicoïdae » et « Sunflower » 209

7.1 (a) Principe de détection des coïncidences et (b) tube de réponse. 212
7.2 (a) Coïncidences diffusées et (b) coïncidences fortuites. 213
7.3 Illustration d'un calcul possible des valeurs $A_{j,i}$ de la matrice système en 2D. La grille arrière correspond à une partie de la paramétrisation finie de l'activité. 215
7.4 Modèle tri-compartimental de Sokoloff 216
7.5 Comparaison des activités reconstruites pour la coupe 1 en fonction de la méthode de reconstruction. 228
7.6 Comparaison des activités reconstruites pour la coupe 2 en fonction de la méthode de reconstruction. 229
7.7 CAT extraites de la coupe 1 pour deux voxels voisins. Activité originale (rouge), activité reconstruite avec EM-ML avec critère d'arrêt (cyan), SIEVES (vert) et FB[DR] (bleue). 230
7.8 CAT extraites de la coupe 2 pour deux voxels voisins. Activité originale (rouge), activité reconstruite avec EM-ML avec critère d'arrêt (cyan), SIEVES (vert) et FB[DR] (bleue). 230
7.9 Evolution de $f(x^{(n)}) + g_\theta(x^{(n)})$ au cours des itérations pour $\theta = 10^{-4}$ (en rouge) et pour $\theta = 10^{-5}$ (en bleu). 231
7.10 Activité dans les tissus pour les trames temporelles $t = 4$, $t = 8$ et $t = 14$ en Bq/cc. 236
7.11 sinogramme 1 : sinogramme (2D+t) généré pour les trames temporelles $t = 4$, $t = 8$ et $t = 14$ où l'échelle représente le nombre de coïncidences. . 237
7.12 sinogramme 2 : sinogramme (2D+t) généré pour les trames temporelles $t = 4$, $t = 8$ et $t = 14$ où l'échelle représente le nombre de coïncidences. . 237

7.13 Valeurs de SSIM et EQM pour les $T = 16$ trames temporelles pour différentes valeurs de l (nombre de niveaux de décomposition temporels), de L (nombre de niveaux de décomposition spatial) et de ϑ (influence du terme de variation totale) pour des données reconstruites à partir du sinogramme 1. (a) valeurs de SSIM pour $l = 1$, (b) valeurs de SSIM pour $l = 2$, (c) valeurs de SNR pour $l = 1$ et (d) valeurs de SNR pour $l = 2$. . 239

7.14 Comparaison des activités reconstruites (en Bq/cc) à partir du sinogramme 1 pour les trames temporelles 4, 8, et 14 en fonction du paramètre de régularisation ϑ. La première ligne correspond à l'activité de référence pour chacune de trames considérées, la seconde ligne présente l'activité reconstruite pour $\vartheta = 10^{-5}$, la troisième ligne celle pour $\vartheta = 10^{-4}$ et la dernière considère le cas où $\vartheta = 5 \times 10^{-4}$. 240

7.15 Valeurs de SSIM et EQM pour les différentes trames temporelles pour différentes valeurs de l (nombre de niveaux de décomposition temporels), de L (nombre de niveaux de décomposition spatial) et de ϑ (influence du terme de variation totale) pour des données reconstruites à partir du sinogramme 2. (a) valeurs de SSIM pour $l = 1$, (b) valeurs de SSIM pour $l = 2$, (c) valeurs de SNR pour $l = 1$ et (d) valeurs de SNR pour $l = 2$. . 241

7.16 Comparaison des activités reconstruites (en Bq/cc) à partir du sinogramme 2 pour les trames temporelles 4, 8, et 14 en fonction du paramètre de régularisation ϑ. La première ligne correspond à l'activité de référence pour chacune de trames considérées, la seconde ligne présente l'activité reconstruite pour $\vartheta = 10^{-5}$, la troisième ligne celle pour $\vartheta = 10^{-4}$ et la dernière considère le cas où $\vartheta = 5 \times 10^{-4}$ 242

7.17 Comparaison des activités reconstruites (en Bq/cc) par différentes approches pour la trame temporelle $t = 4$. 243

7.18 Comparaison des activités reconstruites (en Bq/cc) par différentes approches pour la trame temporelle $t = 8$. 244

7.19 Comparaison des activités reconstruites (en Bq/cc) par différentes approches pour la trame temporelle $t = 14$. 245

7.20 CAT extraites dans le cortex pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue). 247

7.21 CAT dans les noyaux gris centraux pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue) 248

7.22 CAT dans l'artère ajoutée pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue) 249
7.23 CAT dans la matière blanche pour 4 pixels voisins. Activité originale (ligne noire), activité reconstruite par SIEVES 1 (ligne bleue continue), activité reconstruite par SIEVES 2 (ligne bleue discontinue), activité reconstruite par PPXA 1 (ligne rouge continue) et activité reconstruite par PPXA 2 (ligne rouge discontinue) 250
7.24 Cartes paramétriques du taux de métabolisme cérébral régional du glucose (en μmol/min/100g). 251
7.25 Comparaison des activités reconstruites (en Bq/cc) pour la coupe 24 de l'acquisition de données sur babouin. 1ère ligne : SIEMENS, 2ème ligne : SIEVES, 3ème ligne : PPXA. 253

Liste des tableaux

2.1 Tentative de comparaison des principaux algorithmes d'optimisation convexe utilisés dans les problèmes d'imagerie. 45

3.1 Paramètres des algorithmes FB et DR. 78
3.2 SNR pour l'image satellitaire. 101
3.3 SNR pour l'image médicale. 106

4.1 Comparaisons entre PPXA et sa version accélérée. 137
4.2 Comparaison en terme de SNR (dB) des méthodes de déconvolution en présence de bruit de Poisson pour différentes valeurs de facteur d'échelle α et différentes tailles de flou. 147

5.1 Evaluation du SNR et du SSIM pour des images dégradées par un flou et un bruit gaussien puis restaurées en utilisant des DTT. 164
5.2 Evaluation du SNR et du SSIM pour des images dégradées par un flou et un bruit gaussien puis restaurées en utilisant des tranformées GenLOTs. . 165

6.1 Comparaison en terme de SNR (en dB) des images restaurées par la méthode proposée dans le chapitre 4, nommée ici « PPXA Hybride », et la méthode présentée dans ce chapitre, nommée « PPXA Text+Geom ». . . 210

7.1 EQM associées aux CAT présentées sur les figures 7.7 et 7.8. 231
7.2 EQM pour différentes régions d'intérêt. 246

Bibliographie

AFONSO, M. V., BIOUCAS-DIAS, J. M., FIGUEIREDO, M. A. T. (2009). An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Trans. Image Process.* In press.

AFONSO, M. V., BIOUCAS-DIAS, J. M., FIGUEIREDO, M. A. T. (2010). Fast image recovery using variable splitting and constrained optimization. *IEEE Trans. Image Process.*, 19(9):2345–2356.

ALBER, Y. I., IUSEM, A. N., SOLODOV, M. V. (1998). On the projected subgradient method for nonsmooth convex optimization in a Hilbert space. *Math. Programm.*, 81(1):23–35.

ALPERT, N. M., REILHAC, A., CHIO, T. C., SELESNICK, I. (2006). Optimization of dynamic measurement of receptor kinetics by wavelet denoising. *Neuroimage*, 30(2): 444–451.

ANSCOMBE, F. J. (1948). The transformation, of Poisson, binomial and negative-binomial data. *Biometrika*, 35(3-4):246–254.

ANTHOINE, S., PIERPAOLI, E., DAUBECHIES, I. (2006). Deux méthodes de déconvolution et séparation simultanées ; application à la reconstruction des amas de galaxies. *Trait. Signal*, 23(5-6):439–447.

ANTONIADIS, A., FAN, J. (2001). Regularization of wavelets approximations. *J. American Statist. Ass.*, 96(455):939–963.

AUBERT, A., AUJOL, J.-F. (2008). A variational approach to remove multiplicative noise. *SIAM J. App. Math.*, 68(4):925–946.

AUJOL, J.-F., AUBERT, G., BLANC-FÉRAUD, L., CHAMBOLLE, A. (2005). Image decomposition into a bounded variation component and an oscillating component. *Int. J. Comp. Vis.*, 22:71–88.

AUJOL, J.-F., CHAMBOLLE, A. (2005). Dual norms and image decomposition models. *Int. J. Comp. Vis.*, 63:85–104.

AUJOL, J.-F., CHAMBOLLE, A. (2006). Color image decomposition and restoration. *J. Vis. Comm. Image Repr.*, 17(4):916–928.

AUJOL, J.-F., GILBOA, G., CHAN, T., OSHER, S. (2006). Structure-texture image decomposition - modeling, algorithms, and parameter selection. *Int. J. Comp. Vis.*, 67(1):111–136.

BAILLON, J. B., HADDAD, G. (1977). Quelques propriétés des opérateurs angle-bornés et n-cycliquement monotones. *Israel Journal of Mathematics*, 26:137–150.

BAR-LEV, S. K., ENIS, P. (1988). On the classical choice of variance stabilizing transformations and an application for a Poisson variate. *Biometrika*, 75(4):803–804.

BARDSLEY, J. M., LUTTMAN, A. (2009). Total varation-penalized Poisson likelihood estimation for ill-posed problem. *Adv. Comput. Math.*, 31(1):35–39.

BARZILAI, J., BORWEIN, J. M. (1988). Two point step size gradient methods. *IMA J. Numer. Anal.*, 8(1):141–148.

BAUSCHKE, H. H., COMBETTES, P. L. (2008). A Dykstra-like algorithm for two monotone operators. *Pacific J. of Optim.*, 4:383–391.

BAUSCHKE, H. H., COMBETTES, P. L. (2010). The Baillon-Haddad theorem revisited. *J. Convex Anal.*, 17(4):781–787.

BECK, A., TEBOULLE, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202.

BECT, J., BLANC-FÉRAUD, L., AUBERT, G., CHAMBOLLE, A. (2004). A ℓ_1-unified variational framework for image restoration. *In* PAJDLA, T., MATAS, J., éditeurs : *Proc. European Conference on Computer Vision*, vol. LNCS 3024, pages 1–13, Prague, Czech Republic. Springer.

BERTSEKAS, D. P., TSITSIKLIS, J. N. (1997). *Parallel and Distributed Computation : Numerical Methods*. Athena Scientific.

BIOUCAS-DIAS, J., FIGUEIREDO, M. A. T. (2008). An iterative algorithm for linear inverse problems with compound regularizers. *In Proc. Int. Conf. Image Process.*, pages 685–688, San Diego, CA, USA.

BIOUCAS-DIAS, J. M., FIGUEIREDO, M. A. T. (2007). A new TwIST : two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Process.*, 16(12):2992–3004.

BOUMAN, C. A., SAUER, K. (1996). A unified approach to statistical tomography using coordinate descent optimization. *IEEE Trans. Image Process.*, 5(3):480–492.

BREDIES, K., LORENZ, D. A. (2008). Linear convergence of iterative soft-thresholding. *J. Fourier Anal. Appl.*, 14(5-6):813–837.

BREGMAN, L. M. (1965). The method of successive projection for a common point of convex sets. *Soviet Math. Dokl.*, 6:688–692.

BRICEÑO-ARIAS, L. M., COMBETTES, P. L. (2009). Convex variational formulation with smooth coupling for multicomponent signal decomposition and recovery. *Numer. Math. Theory Methods Appl.*, 2(4):485–508.

BROWNE, J., DE PIERRO, A. R. (1996). A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography. *IEEE Trans. Med. Imag.*, 15(5): 687–699.

BUADES, A., COLL, B., MOREL, J. (2005). A review of image denoising algorithms, with a new one. *Multiscale Model. and Simul.*, 4(2):490–530.

BYRNE, C. L. (1993). Iterative image reconstruction algorithms based on cross-entropy minimization. *IEEE Trans. Image Process.*, 2(1):96–103.

CANDÈS, E. J., DONOHO, D. L. (2002). Recovering edges in ill-posed inverse problems : Optimality of curvelet frames. *Ann. Stat.*, 30(3):784–842.

CAPRICELLI, T. D. (2008). *Algorithmes de projections convexes généralisés et applications en imagerie médicale*. Spécialité : Mathématiques appliquées, Thèse de l'université Pierre et Marie Curie – Paris VI.

CARLAVAN, M., WEISS, P., BLANC-FÉRAUD, L., ZERUBIA, J. (2009). Algorithme rapide pour la restauration d'image régularisée sur les coefficients d'ondelettes. *In Proc. GRETSI*, Dijon, France.

CHAARI, L., PESQUET, J.-C., CIUCIU, P., BENAZZA-BENYAHIA, A. (2010a). An iterative method for parallel MRI SENSE-based reconstruction in the wavelet domain. *Medical Image Analysis*. In press.

CHAARI, L., PESQUET, J.-C., TOURNERET, J.-Y., CIUCIU, P., BENAZZA-BENYAHIA, A. (2010b). A hierarchical bayesian model for frame representation. *IEEE Trans. Signal Process.* In press.

CHAMBOLLE, A. (2004). An algorithm for total variation minimization and applications. *J. Math. Imag. Vis.*, 20(1-2):89–97.

CHAN, R. H., CHEN, K. (2007). Multilevel algorithms for a Poisson noise removal model with total variation regularization. *Int. J. Comput. Math.*, 84(8):1183–1198.

CHAN, T., ESEDOGLU, S., PARK, F. (2007). Image decomposition combining staircase reduction and texture extraction. *J. Vis. Comm. Image Repr.*, 18(6):464–486.

CHAUX, C., BENAZZA-BENYAHIA, A., PESQUET, J.-C., DUVAL (2010). Wavelet transform for the denoising of multivariate images. pages 203–237. ISTE Ltd and John Wiley & Sons Inc.

CHAUX, C., COMBETTES, P. L., PESQUET, J.-C., WAJS, V. R. (2007). A variational formulation for frame-based inverse problems. *Inverse Probl.*, 23(4):1495–1518.

CHAUX, C., DUVAL, L., PESQUET, J.-C. (2006). Image analysis using a dual-tree M-band wavelet transform. *IEEE Trans. Image Process.*, 15(8):2397–2412.

CHAUX, C., PESQUET, J.-C., PUSTELNIK, N. (2009). Nested iterative algorithms for convex constrained image recovery problems. *SIAM J. Imag. Sc.*, 2(2):730–762.

CHEN, G. H.-G., ROCKAFELLAR, R. T. (1997). Convergence rates in forward-backward splitting. *SIAM J. Opt.*, 7:421–444.

CHOUZENOUX, E., MOUSSAOUI, S., IDIER, J. (2009). A new line search method for barrier functions with strong convergence properties. Tech. report, IRCCyN.

COHEN, A., DAUBECHIES, I., VIAL, P. (1993). Wavelets on interval and fast wavelet transforms. *Appl. and Comp. Harmonic Analysis*, 1(1):54–81.

COMBETTES, P. L. (1994). Inconsistent signal feasibility problems : least-squares solutions in a product space. *IEEE Trans. Signal Process.*, 42(11):2955–2966.

COMBETTES, P. L. (2003). A block-iterative surrogate constraint splitting method for quadratic signal recovery. *IEEE Trans. Signal Process.*, 51(7):1771–1782.

COMBETTES, P. L. (2004). Solving monotone inclusions via compositions of nonexpansive averaged operators. *Optimization*, 53:475–504.

COMBETTES, P. L. (2009). Iterative construction of the resolvent of a sum of maximal monotone operators. *J. Convex Anal.*, 16(4):727–748.

COMBETTES, P. L., DŨNG, D., CÔNG VŨ, B. (2009). Dualization of signal recovery problems. *Set-Valued and Variational Anal.* In press.

COMBETTES, P. L., PESQUET, J.-C. (2007a). A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery. *IEEE J. Selected Topics Signal Process.*, 1(4):564–574.

COMBETTES, P. L., PESQUET, J.-C. (2007b). Proximal thresholding algorithm for minimization over orthonormal bases. *SIAM J. Opt.*, 18(4):1351–1376.

COMBETTES, P. L., PESQUET, J.-C. (2008). A proximal decomposition method for solving convex variational inverse problems. *Inverse Probl.*, 24(6):x+27.

COMBETTES, P. L., PESQUET, J.-C. (2010). Proximal splitting methods in signal processing. *In* BAUSCHKE, H. H., BURACHIK, R., COMBETTES, P. L., ELSER, V., LUKE, D. R., WOLKOWICZ, H., éditeurs : *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer-Verlag, New York.

COMBETTES, P. L., WAJS, V. R. (2005). Signal recovery by proximal forward-backward splitting. *Multiscale Model. and Simul.*, 4(4):1168–1200.

DAUBECHIES, I. (1992). *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, PA.

DAUBECHIES, I., DEFRISE, M., DE MOL, C. (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Applied Math.*, 57(11):1413–1457.

DAUBECHIES, I., TESCHKE, G. (2005). Variational image restoration by means of wavelets : simultaneous decomposition, deblurring and denoising. *Appl. Comp. Harm. Analysis*, 19:1–16.

DE PIERRO, A., YAMAGISHI, M. (2001). Fast EM-like methods for maximum "a posteriori" estimates in emission tomography. *IEEE Trans. Med. Imag.*, 20(4):280–288.

DE QUEIROZ, R. L., NGUYEN, T. Q., RAO, K. R. (1996). The GenLOT : generalized linear-phase lapped orthogonal transform. *IEEE Trans. Signal Process.*, 40:497–507.

DELEDALLE, C.-A., TUPIN, F., DENIS, L. (2010). Poisson NL means : unsupervised non local means for Poisson noise. *In Proc. Int. Conf. Image Process.*, Hong Kong.

DELYON, B., LAVIELLE, M., MOULINES, E. (1999). Convergence of a stochastic approximation version of the EM algorithm. *Ann. Stat.*, 27:94–128.

DEMPSTER, A., LAIRD, N. M., RUBIN, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, 39(1):1–38.

DEUTSCH, F. (2001). *Best Approximation in Inner Product Spaces*. Springer-Verlag, New York.

DEY, N., BLANC-FÉRAUD, L., ZIMMER, C., KAM, Z., ROUX, P., OLIVO-MARIN, J., ZERUBIA, J. (2006). Richardson-Lucy algorithm with total variation regularization for 3D confocal microscope deconvolution. *Microscopy Research Technique*, 69: 260–266.

DITTRICH, S., van den HOOF, J., MAGUIRE, R. P., éditeurs (2008). *PET Pharmacokinetic Course*. IEEE Nuclear Science Symposium and Medical Imaging Conference.

DO, M. N., VETTERLI, M. (2002). Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance. *IEEE Trans. Image Process.*, 11(2): 146–158.

DO, M. N., VETTERLI, M. (2005). The contourlet transform : an efficient directional multiresolution image representation. *IEEE Trans. Image Process.*, 14(12):2091–2106.

DONOHO, D. L. (1993). Nonlinear wavelet methods for recovery of signals, densities, and spectra from indirect and noisy data. *Proceedings of Symposia in Applied Mathematics*, 47:173–205.

DONOHO, D. L. (1995). De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41(3):613–627.

DOUGLAS, J., RACHFORD, H. H. (1956). On the numerical solution of the heat conduction problem in two and three space variables. *Trans. Amer. Math. Soc.*, 82(2):421–439.

DUPÉ, F.-X., FADILI, M. J., STARCK, J.-L. (2009). A proximal iteration for deconvolving Poisson noisy images using sparse representations. *IEEE Trans. Image Process.*, 18(2):310–321.

DURAND, S., FADILI, J., NIKOLOVA, M. (2010). Multiplicative noise removal using L1 fidelity on frame coefficients. *J. Math. Imag. Vis.*, 36(3):201–226.

DURAND, S., NIKOLOVA, M. (2007). Denoising of frame coefficients using ℓ_1 data-fidelity term and edge-preserving regularization. *Multiscale Model. and Simul.*, 6(2): 547–576.

ECKSTEIN, J., BERTEKAS, D. P. (1992). On the Douglas-Rachford splitting methods and the proximal point algorithm for maximal monotone operators. *Math. Programm.*, 55(3):293–318.

EKELAND, I., TÉMAM, R. (1999). *Convex analysis and variational problems*. SIAM, Philadelphia.

ELAD, M., MILANFAR, P., RON, R. (2007). Analysis versus synthesis in signal priors. *Inverse Probl.*, 23(3):947–968.

FADILI, M. J., PEYRÉ, G. (2010). Total variation projection with first order schemes. *IEEE Trans. Image Process.* In press.

FADILI, M. J., STARCK, J.-L. (2009). *Curvelets and Ridgelets , in Encyclopedia of Complexity and Systems Science*, vol. 3. Springer.

FENCHEL, W. (1953). *Convex cones, sets and functions*. Princeton University, Philadelphia, PA.

FESSLER, J. A. (1994). Penalized weighted least-squares image reconstruction for Positron Emission Tomography. *IEEE Trans. Med. Imag.*, 13(2):290–300.

FESSLER, J. A. (1995). Hybrid Poisson/polynomial objective functions for tomographic image reconstruction from transmission scans. *IEEE Trans. Image Process.*, 4(10): 1439–1450.

FESSLER, J. A., HERO, A. O. (1994). Space-alternating generalized expectation maximization algorithm. *IEEE Trans. Signal Process.*, 42(10):2664–2676.

FESSLER, J. A., HERO, A. O. (1995). Penalized maximum-likelihood image reconstruction using space-alternating generalized em algorithms. *IEEE Trans. Image Process.*, 4(10):1417–1429.

FIGUEIREDO, M. A. T., BIOUCAS-DIAS, J. M. (2010). Restoration of Poissonian images using alternating direction optimization. *IEEE Trans. Image Process.* In press.

FIGUEIREDO, M. A. T., BIOUCAS-DIAS, J. M., OLIVEIRA, J. P., NOWAK, R. D. (2006). A new majorization-minimization algorithm and an experimental comparison with wavalet denoising. *In Proc. Int. Conf. Image Process.*, pages 2633–2636, Atlanta, GA.

FIGUEIREDO, M. A. T., NOWAK, R. D. (2003). An EM algorithm for wavelet-based image restoration. *IEEE Trans. Image Process.*, 12(8):906–916.

FORNASIER, M. (2007). Domain decomposition methods for linear inverse problems with sparsity constraints. *Inverse Probl.*, 23(6):2505–2526.

FORNASIER, M., SCHÖNLIEB, C.-B. (2009). Subspace correction methods for total variation and ℓ_1-minimization. *SIAM J. Numer. Anal.*, 47(8):3397–3428.

FORTIN, M., GLOWINSKI, R., éditeurs (1983). *Augmented Lagrangian Methods : Applications to the Numerical Solution of Boundary-Value Problems*. Elsevier Science Ltd, Amsterdam : North-Holland.

FRYŹLEWICZ, P., NASON, G. (2001). Poisson intensity estimation using wavelets and the fisz transformation. Rapport technique, Department of Mathematics, University of Bristol, BRISTOL, UK. Technical Report 01-10.

GALATSANOS, N. P., KATSAGGELOS, A. K. (1992). Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation. *IEEE Trans. Image Process.*, 1(3):322–336.

GAUTHIER, J., DUVAL, L., PESQUET, J.-C. (2009). Optimization of synthesis oversampled complex filter banks. *IEEE Trans. Signal Process.*, 57(10):3827–3843.

GJEDDE, A. (1982). Calculation of cerebral glucose phosphorylation from brain uptake of glucose analogs in vivo : a re-examination. *Brain Res.*, 257(2):237–274.

GOLDBURG, M., MARKS II, R. (1985). Signal synthesis in the presence of an inconsistent set of constraints. *IEEE Trans. Circuits Syst.*, 32(7):647–663.

GOLDSTEIN, D., OSHER, S. (2008). The split bregman method for ℓ_1 regularized problems. Rapport technique, UCLA CAM.

GOLUB, G. H., VAN LOAN, C. F. (1996). *Matrix computations*. The Johns Hopkins University Press ; 3rd edition.

GORDON, R., BENDER, R., HERMAN, G. T. (1970). Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography. *Journal of Theoretical Biology*, 29(3):471–481.

GURIN, L. G., POLYAK, B. T., RAIK, E. V. (1967). Projection methods for finding a common point of convex sets. *Zh. Vychisl. Mat. Mat. Fiz.*, 7(6):1211–1228.

HADAMARD, J. (1902). Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton Univ. Bull*, 13:49–52.

HAN, D., LARSON, D. R. (2000). Frames, bases, and group representations. *Mem. Amer. Math. Soc.*, 147(697):x+94.

HEBERT, T., LEAHY, R. (1989). A generalized EM algorithm for 3-D bayesian reconstruction from Poisson data using Gibbs priors. *IEEE Trans. Med. Imag.*, 8(2):194–202.

HERMAN, G. T., HURWITZ, H., LENT, A., LUNG, H. (1979). On the bayesian approach to image reconstruction. *Info. and Cont.*, 42:60–71.

HERMAN, G. T., MEYER, L. B. (1993). Algebraic reconstruction techniques can be made computationally efficient. *IEEE Trans. Med. Imag.*, 12(3):600–609.

HESTENES, M. R. (1969). Multiplier and gradient methods. *Journal of Opt. Theory and Applications*, 4(5):303–320.

HIRIART-URRUTY, J.-B., LEMARÉCHAL, C. (1996). *Convex analysis and minimization algorithms, Part I : Fundamentals*, vol. 305 *Grundlehren der mathematischen Wissenschaften*. Springer-Verlag, Berlin, Heidelberg, N.Y., 2nd édition.

HUANG, Y., NG, M., WEN, Y.-W. (2008). A fast total variation minimization method for image restoration. *Multiscale Model. and Simul.*, 7(2):774–795.

HUDSON, H., LARKIN, R. (1994). Accelerated image reconstruction using ordered subsets of projection data. *IEEE Trans. Med. Imag.*, 13(4):601–609.

HUDSON, H. M. (1978). A natural identity for exponential families with applications in multiparameter estimation. *Ann. Stat.*, 6(3):473–484.

HUNT, B., KÜBLER, O. (1984). Karhunen-Loeve multispectral image restoration, part i : Theory. *IEEE Trans. Acous., Speech Signal Process.*, 32(3):592–600.

HUNTER, D. R., LANGE, K. (2004). A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37.

IUSEM, A. N., DE PIERRO, A. R. (2006). Convergence results for an accelerated nonlinear Cimmino algorithm. *Numerische Mathematik*, 49(4):367–378.

KAMASAC, M. E., BOUMAN, C. A., MORRIS, E. D., SAUER, K. (2005). Direct reconstruction of kinetic parameter images from dynamic PET data. *IEEE Trans. Med. Imag.*, 24(6):636–650.

KANG, M. (1998). Generalized multichannel image deconvolution approach and its applications. *Opt. Eng.*, 37(11):2953–2964.

KATSAGGELOS, A., LAY, K., GALATSANOS, N. (1993). A general framework for frequency domain multi-channel signal processing. *IEEE Trans. Image Process.*, 2(3):417–420.

KINGSBURY, N. G. (2001). Complex wavelets for shift invariant analysis and filtering of signals. *Appl. Comp. Harm. Analysis*, 10(3):234–253.

LANDWEBER, L. (1951). An iteration formula for fredholm integral equations of the first kind. *Amer. J. Math.*, 73(3):615–624.

LANGE, K., BAHN, M., LITTLE, R. (1987). A theoretical study of some maximum likelihood algorithms for emission and transmission tomography. *IEEE Trans. Med. Imag.*, MI-6:106–114.

LANGE, K., CARSON, R. (1984). EM reconstruction algorithms for emission and transmission tomography. *J. Comput. Assist. Tomo.*, 8(2):306–316.

LE PENNEC, E., MALLAT, S. (2005). Sparse geometric image representations with bandelets. *IEEE Trans. Image Process.*, 14(4):423–438.

LIONS, P. L., MERCIER, B. (1979). Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.*, 16(6):964–979.

LOUCHET, C. (2008). *Modèles variationnels et bayésiens pour le débruitage d'images : de la variation totale vers les moyennes non-locales.* Mathématiques appliquées, Université Paris Descartes.

LUCIGNANI, G., SCHMIDT, K. C., MORESCO, R. M., STRIANO, G., COLOMBO, F., SOKOLOFF, L., FAZIO, F. (1993). Measurement of regional cerebral glucose utilization with fluorine-18-FDG and PET in heterogeneous tissues : Theoretical considerations and practical procedure. *J. Nucl. Med.*, 34(3):360–369.

LUCY, L. B. (1974). An iterative technique for the rectification of observed distributions. *Astronomical Journal*, 79(6):745–754.

LUISIER, F., VONESCH, C., BLU, T., UNSER, M. (2010). Fast interscale wavelet denoising of Poisson-corrupted images. *Signal Process.*, 90(2):415–427.

MÄKITALO, M., FOI, A. (2010). Optimal inversion of the Anscombe transformation in low-count Poisson image denoising. *IEEE Trans. Image Process.* In press.

MALGOUYRES, F. (2002). Mathematical analysis of a model which combines total variation and wavelet for image restoration. *Journal of Information Processes*, 2(1):1–10.

MALLAT, S. (1997). *A wavelet tour of signal processing*. Academic Press, San Diego, USA.

MEYER, Y. (2001). *Oscillating patterns in image processing and in some nonlinear evolution equations*. AMS, Providence, RI.

MILED, W., PESQUET, J.-C., PARENT, M. (2009). A convex optimization approach for depth estimation under illumination variation. *IEEE Trans. Image Process.*, 18(4): 813–830.

MOREAU, J. J. (1965). Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. France*, 93:273–299.

MOREAU, J. J. (1966-1967). Fonctionnelles convexes. *In Séminaire sur les équations aux dérivées partielles II, Collège de France, Paris*.

NESTEROV, Y. E. (1983). A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Math. Dokl.*, 27(2):372–376.

NESTEROV, Y. E. (2007). Gradient methods for minimizing composite objective function. *In CORE Discussion paper 76*.

NESTEROV, Y. E. (2009). Primal-dual subgradient methods for convex problems. *Math. Programm.*, 120(1):221–259.

NICHOLS, T. E., QI, J., ASMA, E., LEAHY, R. M. (2002). Spatiotemporal reconstruction of list-mode PET data. *IEEE Trans. Med. Imag.*, 21(4):396–404.

PATLAK, C. S., BLASBERG, R. G., FENSTERMACHER, J. D. (1983). Graphical evaluation of blood-to-brain transfer constants from multiple-time uptake data. *J. Cereb. Blood Flow Metab.*, 3(1):1–7.

PEDONE, M., HEIKKILÄ, J. (2008). Blur and contrast invariant fast stereo matching. *Lecture Notes in Comput. Sci.*, 5259:883–890.

PESQUET, J.-C. (2010). A parallel inertial proximal optimization method. Preprint.

PIERT, M., KOEPPE, R. A., GIORDANI, B., BERENT, S., KUHL, D. E. (1996). Diminished glucose transport and phosphorylation in Alzheimer's disease determined by dynamic FDG-PET. *J. Nucl. Med.*, 37(2):201–208.

POLYAK, B. T. (1969). Minimization of unsmooth functionals. *USSR Computational Mathematics and Mathematical Physics*, 9(3):14–29.

PUSTELNIK, N., CHAUX, C., PESQUET, J.-C. (2009a). Extension des algorithmes imbriqués pour la résolution de problèmes d'optimisation convexe en imagerie. *In Proc. GRETSI*, pages x+4, Dijon, France.

PUSTELNIK, N., CHAUX, C., PESQUET, J.-C. (2009b). Hybrid regularization for data restoration in the presence of Poisson noise. *In Proc. Eur. Sig. and Image Proc. Conference*, pages x+5, Glasgow, Scotland.

RAMANI, S., BLU, T., UNSER, M. (2008). Monte-Carlo SURE : A black-box optimization of regularization parameters for general denoising algorithms. *IEEE Trans. Image Process.*, 17(9):1540–1554.

READER, A. J., SUREAU, F. C., COMTAT, C., TREBOSSEN, R., BUVAT, I. (2006). Joint estimation of dynamic PET images and temporal basis functionc using fully 4D ML-EM. *Phys. Med. Biol.*, 51(21):5455–5474.

RICHARDSON, W. H. (1972). Bayesian-based iterative method of image restoration. *J. Opt. Soc. America*, 62(1):55–59.

ROBERT, C., CASTELLA, G. (2004). *Monte Carlo statistical methods*. Springer.

ROCKAFELLAR, R. T. (1967). Duality and stability in extremum problems involving convex functions. *Pacific J. of Math.*, 21(1):167–187.

ROCKAFELLAR, R. T. (1974). *Conjugate duality and optimization*. SIAM, Philadelphia, PA.

ROCKAFELLAR, R. T., WETS, R. J.-B. (2004). *Variational analysis*. Springer-Verlag.

RUDIN, L., OSHER, S. (1994). Total variation based image restoration with free local constraints. *In Proc. Int. Conf. Image Process.*, vol. 1, pages 31–35, Austin, Texas.

RUDIN, L., OSHER, S., FATEMI, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268.

SAUER, K., BOUMAN, C. A. (1993). A local update strategy for iterative reconstruction from projections. *IEEE Trans. Signal Process.*, 41(2):534–548.

SCHARSTEIN, D., SZELISKI, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comp. Vis.*, 47(1-3):7–42.

SELESNICK, I. W., BARANIUK, R. G., KINGSBURY, N. G. (2005). The dual-tree complex wavelet transform. *IEEE Signal Process. Mag.*, 22(6):123–151.

SELESNICK, I. W., FIGUEIREDO, M. (2009). Signal restoration with overcomplete wavelet transforms : comparison of analysis and synthesis priors. *In SPIE*, vol. 7446.

SETZER, S., STEIDL, G., TEUBER, T. (2010). Deblurring Poissonian images by split Bregman techniques. *J. Visual Communication and Image Representation*, 21(3): 193–199.

SHEPP, L. A., VARDI, Y. (1982). Maximum likelihood reconstruction in Positron Emission Tomography. *IEEE Trans. Med. Imag.*, 1(2):113–122.

SNYDER, D. L. (1984). Parameter estimation for dynamic studies in emission-tomography systems having list-mode data. *IEEE Trans Nucl. Sci.*, 31(2):925–931.

SNYDER, D. L., MILLER, M. I. (1985). The use of Sieves to stabilize images produced with the EM algorithm for Emission Tomography. *IEEE Trans Nucl. Sci.*, 32:3864–3872.

STEIDL, G., TEUBER, T. (2010). Removing multiplicative noise by Douglas-Rachford splitting methods. *J. Math. Imag. Vis.*, 36(3):168–184.

STEIN, C. (1981). Estimation of the mean of a multivariate normal distribution. *Ann. Stat.*, 9(6):1135–1151.

SUREAU, F. C., PESQUET, J.-C., CHAUX, C., PUSTELNIK, N., READER, A. J., COMTAT, C., TREBOSSEN, R. (2008). Temporal wavelet denoising of PET sinograms and images. Dresden, Germany.

SVAITER, B. F. (2010). Weak convergence on Douglas-Rachford method. http ://www.optimization−online.org/DB_HTML/2010/07/2676.html.

SWELDENS, W. (1996). The lifting scheme : A custom-design construction of biorthogonal wavelets. *Appl. Comp. Harm. Analysis*, 3(2):186–200.

TIKHONOV, A. (1963). Tikhonov regularization of incorrectly posed problems. *Soviet Mathematics Doklady*, 4:1624–1627.

TITTERINGTON, D. (1987). On the iterative image space reconstruction algorithm for ECT. *IEEE Trans. Med. Imag.*, 6(1):52–56.

TRUSSELL, H. J., CIVANLAR, M. R. (1984). The feasible solution in signal restoration. *IEEE Trans. Acous., Speech Signal Process.*, 32(2):201–212.

TSCHUMPERLÉ, D., DERICHE, R. (2002). Diffusion PDEs on vector-valued images. *IEEE Signal Process. Mag.*, 19(5):16–25.

TSENG, P. (2000). A modified forward-backward splitting method for maximal monotone mappings. *SIAM J. Control. & Optim.*, 38(2):431–446.

TSUI, K. W., PRESS, S. J. (1982). Simultaneous estimation of several Poisson parameters under k-normalized squared error loss. *Ann. Stat.*, 10(1):93–100.

TURKHEIMER, F. E., BRETT, M., VISVIKIS, D., CUNNINGHAM, V. J. (1999). Multiresolution analysis of Emission Tomography images in the wavelet domain. *Journal of Cerebral Blood Flow and Metabolism*, 19(11):1189–1208.

VALK, P. E., BAILEY, D. L., TOWNSEND, D. W., MAISEY, M. N., éditeurs (2004). *Positron Emission Tomography. Basic science and clinical pratice*. Springer.

VEKLEROV, E., LLACER, J. (1987). Stopping rule for the MLE algorithm based on statistical hypothesis testing. *IEEE Trans. Med. Imag.*, 6(4):209–226.

VERHAEGHE, J., VAN DE VILLE, D., KHALIDOV, I., D'ASSELER, Y., LEMAHIEU, I., UNSER, M. (2008). Dynamic PET reconstruction using wavelet regularization with adapted basis functions. *IEEE Trans. Med. Imag.*, 27(7):943–959.

VERHAEGHE, J., VAN DE VILLE, D., KHALIDOV, I., UNSER, M., D'ASSELER, Y., LEMAHIEU, I. (2007). Reconstruction of dynamic PET data using spatio-temporal wavelet l_1 regularization. *In EMBS 2007. 29th Annual International Conference of the IEEEC*, pages 6539–6542.

VESE, L., OSHER, S. (2003). Modeling textures with total variation minimization and oscillating patterns in image processing. *J. Sci. Comput.*, 19(1-3):553–572.

VESE, L., OSHER, S. (2004). Image denoising and decomposition with total variation minimization and oscillatory functions. *J. Math. Imag. Vis.*, 20(1-2):7–18.

WANG, Y., YANG, J., YIN, W., ZHANG, Y. (2008). A new alternating minimization algorithm for total variation image reconstruction. *SIAM J. Imaging Sci.*, 1(3):248–272.

WANG, Z., BOVIK, A. C. (2009). Mean squared error : love it or leave it ? *IEEE Signal Process. Mag.*, 26(1):98–117.

WEISS, P. (2008). *Algorithmes rapides d'optimisation convexe. Applications à la restauration d'images et à la détection de changement.* Spécialité : Automatique, traitement du signal et des images, Thèse de doctorat en Sciences de l'université de Nice - Sophia Antipolis.

WEISS, P., BLANC-FÉRAUD, L., AUBERT, G. (2009). Efficient schemes for total variation minimization under constraints in image processing. *SIAM J. Sci. Comput.*, 31(3): 2047–2080.

WEN, Y.-W., NG, M. K., CHING, W.-K. (2008). Iterative algorithms based on decoupling of deblurring and denoising for image restoration. *SIAM J. Sci. Comput.*, 30(5): 2655–2674.

YOULA, D. C., WEBB, H. (1982). Image restoration by the method of projections onto convex sets. Part I - theory. *IEEE Trans. Med. Imag.*, 1(2):81–94.

ZHANG, B. H., FADILI, M. J., STARCK, J. L. (2008). Wavelets, ridgelets, and curvelets for Poisson noise removal. *IEEE Trans. Image Process.*, 17(7):1093–1108.

ZHENG, J., SAQUIB, S. S., SAUER, K., BOUMAN, C. A. (2000). Parallelizable bayesian tomography algorithms with rapid, guaranteed convergence. *IEEE Trans. Image Process.*, 9(10):1745–1759.

ZHOU, J., COATRIEUX, J.-L., BOUSSE, A., SHU, H. Z., LUO, L. M. (2007). A bayesian MAP-EM algorithm for PET image reconstruction using wavelet transform. *IEEE Trans Nucl. Sci.*, 54(5):1660–1669.

ZHOU, J., SENHADJI, L., COATRIEUX, J.-L., LUO, L. (2009). Iterative PET image reconstruction using translation invariant wavelet transform. *IEEE Trans Nucl. Sci.*, 56(1):116–128.

ZĂLINESCU, C. (2002). *Convex analysis in general vector spaces*. World Scientific, River Edge, NJ, USA.

ZUBAL, I. G., HARRELL, C. R., SMITH, E. O., RATTNER, Z., GINDI, G., HOFFER, P. B. (1994). Computerized three-dimensional segmented human anatomy. *Medical Physics*, 21(1):299–302.