



HAL
open science

Algorithmes bayésiens variationnels accélérés et applications aux problèmes inverses de grande taille

Yuling Zheng

► **To cite this version:**

Yuling Zheng. Algorithmes bayésiens variationnels accélérés et applications aux problèmes inverses de grande taille. Traitement des images [eess.IV]. Université Paris Sud - Paris XI, 2014. Français. NNT : 2014PA112354 . tel-01135180

HAL Id: tel-01135180

<https://theses.hal.science/tel-01135180v1>

Submitted on 24 Mar 2015

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-SUD

ÉCOLE DOCTORALE : Sciences et Technologie de l'Information, des
Télécommunications et des Systèmes

Laboratoire des signaux et systèmes

DISCIPLINE : Physique

THÈSE DE DOCTORAT

soutenue le 04/12/2014

par

Yuling ZHENG

**Algorithmes bayésiens variationnels accélérés et
applications aux problèmes inverses de grande taille**

Directeur de thèse : Thomas RODET Professeur de l'ENS Cachan
Encadrant : Aurélia FRAYSSE Maître de conférence de l'Université Paris Sud

Composition du jury :
Président du jury : François GOUDAIL Professeur de l'Institut d'Optique
Rapporteurs : André FERRARI Professeur de l'Université de Nice Sophia Antipolis
 Jérôme IDIER Directeur de Recherche CNRS

Examineur : Jean-Christophe PESQUET Professeur de l'Université de Paris-Est

Remerciements

A l'issue de ces trois dernières années de travail, je voudrais remercier toutes les personnes qui m'ont accompagnée, aidée, soutenue et encouragée.

Tout d'abord, je tiens à remercier mon directeur de thèse, M. Thomas Rodet et ma co-encadrante de thèse, Mme. Aurélia Fraysse, pour la confiance qu'ils m'ont accordée, pour leur enthousiasme, leur patience et leurs précieux conseils. J'ai beaucoup appris à leurs côtés et je leur adresse ma gratitude pour tout cela.

Je remercie également les membres du jury qui ont accepté d'évaluer mon travail de thèse. Je tiens à remercier M. François Goudail pour avoir accepté de présider le jury. Je remercie sincèrement les deux rapporteurs, M. André Ferrari et M. Jérôme Idier. Je leur suis reconnaissante du temps qu'ils ont consacré pour lire ma thèse et les remarques et suggestions qui m'ont permis d'améliorer la qualité de cette thèse. Je remercie également M. Jean-Christophe Pesquet pour avoir accepté d'examiner ce travail.

Un grand merci aux collègues du Groupe Problèmes Inverses (GPI). D'abord, merci à toutes les personnes dans la salle des thésards : Leila, Long, Thomas, Mircea, Caifang, Li, Fangcheng et les anciens doctorants qui sont déjà partis : Sha et Ning. Je n'oublierai jamais les moments que nous avons passé à discuter et à partager les spécialités de différents pays. Je remercie également les autres membres du groupe : Ali, Nicolas, Matthieu, Hélène et François.

Je remercie aussi Alexandre Renaux et Matthieu Kowalski avec qui j'ai eu l'honneur de travailler pendant mon monitorat à l'Université Paris Sud.

J'exprime aussi ma reconnaissance à Maryvonne Giron, notre secrétaire, pour sa gentillesse et sa patience.

Mes remerciements vont également à mes amis qui m'ont accompagnée durant ces années. Merci à Yijing, Jing pour passer du bon temps ensemble à jouer au Badminton. Merci à Xiaoxia, Jin, Yujun, Li, Chuan, ... d'être toujours là. Un remerciement particulier à Li pour m'avoir accompagnée et soutenue tout le temps.

Mes plus profonds remerciements vont à mes parents. Tout au long de mes études en France, ils m'ont toujours soutenue et encouragée, malgré la distance.

A mes parents et mon frère.

Table des matières

Table des figures	IX
Liste des tableaux	1
1 Introduction	3
1.1 Introduction générale	3
1.2 Problème inverse	5
1.2.1 Modèle direct	6
1.2.2 Problèmes inverses	6
1.3 Inférence bayésienne	7
1.3.1 La vraisemblance	8
1.3.2 Loi <i>a priori</i>	9
1.3.3 Estimateurs ponctuels dans l'inférence bayésienne	10
1.3.4 Estimation non-supervisée	11
1.4 Algorithmes de calculs dans l'inférence bayésienne	12
1.4.1 Méthodes pour calculer l'estimateur MAP	13
1.4.2 Méthode d'approximation stochastique	18
2 État de l'art sur les méthodes d'approximation analytique	21
2.1 Introduction	21
2.2 Approximation de Laplace	21
2.3 Approximation bayésienne variationnelle classique	22
2.3.1 Principe de la méthode d'approximation bayésienne variationnelle	23
2.3.2 Choix de la mesure de dissemblance	23
2.3.3 L'algorithme d'approximation bayésienne variationnelle classique	25
2.3.4 Lien entre l'algorithme d'approximation BV et l'algorithme EM	26
2.3.5 Choix de séparation	28

2.4	Approximation bayésienne variationnelle de type gradient	32
2.5	Conclusion	35
3	Méthode d'approximation bayésienne variationnelle de sous-espace	37
3.1	Introduction	37
3.2	Développement de l'algorithme d'approximation BV de sous-espace	38
3.2.1	Méthode itérative de sous-espace dans les espaces de Hilbert	38
3.2.2	Méthode d'approximation BV fondée sur la méthode d'optimisation de sous-espace	40
3.3	Application à un problème inverse en utilisant une loi <i>a priori</i> de Student	43
3.3.1	Formulation bayésienne	43
3.3.2	Applications des méthodes d'approximation BV	44
3.3.3	Les algorithmes de reconstruction non-supervisés	50
3.4	Évaluations expérimentales	54
3.4.1	Descriptions des algorithmes à comparer	54
3.4.2	Configuration de simulation	57
3.4.3	Résultats de simulation dans le cas supervisé	57
3.4.4	Résultats de simulation dans le cas non-supervisé	60
3.5	Conclusion	61
4	Applications aux problèmes inverses en traitement d'images	63
4.1	Introduction	63
4.2	Approche non-supervisée basée sur un <i>a priori</i> de TV	64
4.2.1	Formulation bayésienne	65
4.2.2	Application des méthodes d'approximation BV	66
4.2.3	Des approches de l'état de l'art	72
4.2.4	Simulations	73
4.3	Approche non-supervisée basée sur un <i>a priori</i> à variables cachées de type Geman	82
4.3.1	Formulation bayésienne	83
4.3.2	Loi <i>a priori</i> pour les hyperparamètres	85
4.3.3	Application des méthodes d'approximation BV	86
4.3.4	Simulations	91
4.4	Conclusion	96

5	Applications dans le domaine des ondelettes	99
5.1	Introduction	99
5.2	Énoncé du problème	101
5.3	Approches de l'état de l'art	101
5.3.1	Approche <i>SUREShrink</i>	101
5.3.2	Approche par GSURE <i>Shrinkage</i> itérative	102
5.4	Approche non-supervisée avec un <i>a priori</i> de la famille GSM	104
5.4.1	Formulation bayésienne	104
5.4.2	Application des méthodes d'approximation BV	106
5.4.3	Cas d'un <i>a priori</i> gaussien généralisé	108
5.5	Résultats de simulation	109
5.5.1	Problème de débruitage	110
5.5.2	Problème de déconvolution	113
5.6	Conclusion	118
6	Conclusions et perspectives	121
6.1	Bilan des travaux	121
6.2	Perspectives	123
A	Détails des calculs de la fonction intermédiaire	127
A.1	La dérivation de $q_i^{r_1}$ pour le problème inverse en utilisant la loi <i>a priori</i> de Student	127
A.2	La dérivation de $q_i^{r_1}$ pour le problème inverse en utilisant la loi <i>a priori</i> de TV	129
B	Calculs du pas de l'algorithme	131
B.1	Calculs pour le pas sous-optimal de l'approche BV basé sur le Sous-espace Gradient (BV-SG)	131
B.1.1	Étude des dérivées partielles de la moyenne \mathbf{m}_s et de la variance σ_s^2	132
B.2	Calculs pour le pas sous-optimal de l'approche BV basé sur le sous-espace Mémoire de Gradient (BV-MG)	133
C	Détails de calculs du Chapitre 4	135
C.1	Détails de calculs de la variable auxiliaire λ_i	135
C.2	Calculs de la fonction de partition de la loi <i>a priori</i> de type Geman $Z_{\mathbf{x} \mathbf{b}}$	136

Publications	138
Bibliographie	139

Table des figures

1.1	Problème direct et Problème inverse	5
3.1	Données : sinogramme obtenue avec 32 angles de projections et 95 détecteurs.	57
3.2	Les images sont présentées avec le même niveau de gris inversé. (a) L'image Vrai, (b) FBP avec un filtre rampe, (c) l'échantillonnage de Gibbs, (d) BVClass-SP, (e) BVClass-ST, (f) BV-Grad, (g) BV-SG, (h) BV-MG. . . .	58
3.3	Les images sont présentées avec le même niveau de gris inversé. Les algorithmes supervisés : (a) BV-SG, (b) BV-MG, les algorithmes non-supervisés : (c) NSBV-SG (algorithme 6), (d) NSBV-MG (algorithme 7). . .	61
3.4	Les évolutions des estimations des hyperparamètres. Les lignes en pointillé représentent les valeurs vraies des hyperparamètres. (a) Le logarithme de la variance du bruit ($\sigma_n^2 = 1/\gamma_n$) pour USBV-SG, (b) le logarithme de la variance de l' <i>a priori</i> ($\sigma_p^2 = 1/\gamma_p$) pour USBV-SG, (c) le logarithme de la variance du bruit ($\sigma_n^2 = 1/\gamma_n$) pour USBV-MG, (d) le logarithme de la variance de l' <i>a priori</i> ($\sigma_p^2 = 1/\gamma_p$) pour USBV-MG.	62
4.1	Illustration de la procédure MM dans le cas TV. La courbe en gras représente $ x $ et les trois courbes au-dessus représentent les fonctions majorantes de $ x : \frac{x^2+\lambda}{2\sqrt{\lambda}}$ où $\lambda > 0$	71
4.2	Images à haute résolution : (a) <i>Testpat</i> de dimension 256×256 , (b) <i>Cameraman</i> de dimension 256×256 , (c) <i>Lena</i> de dimension 512×512 , (d) <i>Jetplane</i> de dimension 512×512	75
4.3	Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour <i>Testpat</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	78
4.4	Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour <i>Cameraman</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	79

4.5	Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour <i>Lena</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	80
4.6	Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour <i>Jetplane</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	81
4.7	Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour <i>Testpat</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	94
4.8	Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour <i>Cameraman</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	94
4.9	Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour <i>Lena</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	95
4.10	Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour <i>Lena</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	95
4.11	Image <i>Mire</i> à haute résolution de dimension 256×256	96
4.12	Exemples des images BR (ligne en haut), exemples des images HR obtenues par l'approche BV-MGSR (la deuxième ligne), l'approche BV&MCMC-SR (la troisième ligne) pour <i>Mire</i> dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.	97
5.1	Illustration de la décomposition en ondelettes sur 3 niveaux.	110
5.2	(a) L'image vraie, (b) l'image bruitée, SNR = 20 dB, (c) l'approche <i>SUR-EShrink</i> , (d) l'approche BV-MGSt et (e) l'approche BV-MGGG	112
5.3	Histogrammes des coefficients d'ondelettes de <i>Lena</i> à différente sous-bande. Les lignes de la plus haut à la plus bas correspondent aux échelles de la plus grossière à la plus fine.	113
5.4	(a) L'image vraie, (b) l'image floue, SNR = 40dB, (c) le seuillage itératif, (d) l'approche BV-MGSt et (e) l'approche BV-MGGG.	115

5.5	(a) L'image vraie, (b) l'image floue, SNR = 40dB, (c) le seuillage itératif, (d) l'approche BV-MGSt et (e) l'approche BV-MGGG.	116
5.6	Les histogrammes des coefficients des ondelettes dans l'intervalle $[-0.1, 0.1]$. Première ligne : BV-MGSt, deuxième ligne : BV-MGGG. (a) <i>Lena</i> , (b) <i>Camerman</i>	117
5.7	Première ligne : les résultats obtenus avec l' <i>a priori</i> TV, deuxième ligne : les résultats obtenus par BV-MGGG.	118

Liste des tableaux

3.1	Comparaison de performance en termes de PSNR(dB)/Temps CPU (s). . .	59
3.2	Comparaison de performance en termes de PSNR(dB).	60
4.1	Comparaisons des performances de BVClassSR [BMK11], BV-GradSR [FR14] et notre approche BV-MGSR en termes de nombre d'itérations (Nb) et Temps de Calculs (en Secondes).	76
4.2	Comparaisons des Performances de BVClassSR [BMK11], BV-MGSR et l'approche BV&MCMC-SR en termes de PSNR (dB) et Temps de Calculs (en Secondes).	92
4.3	Comparaisons des Performances de l' <i>a priori</i> TV (BV-MGSR) et l' <i>a priori</i> de type Geman (BV&MCMC-SR) en termes de PSNR (dB).	96
5.1	Les valeurs PSNR des reconstructions obtenues par <i>SUREShrink</i> , BV-MGSt et BV-MGGG	111
5.2	Hyperparamètres γ_p^l estimés par l'approche BV-MGSt. Échelle de la plus grossière à la plus fine : 3 \rightarrow 1.	112
5.3	Hyperparamètres γ_p^l estimés par l'approche BV-MGGG. Échelle de la plus grossière à la plus fine : 3 \rightarrow 1.	113
5.4	Les valeurs PSNR (dB) des reconstructions obtenues par le seuillage itératif, BV-MGSt et BV-MGGG	114
5.5	Hyperparamètres γ_p^l estimés par l'approche BV-MGGG pour <i>Cameraman</i> . Échelle de la plus grossière à la plus fine : 3 \rightarrow 1.	116
5.6	Les valeurs de γ_n estimées par nos approches : BV-MGSt et BV-MGGG . .	117

1

Introduction

1.1 Introduction générale

Durant ces vingt dernières années le développement de la micro-électronique et l'informatique a considérablement augmenté, d'une part les capacités de calcul des ordinateurs et, d'autre part, le nombre de données à traiter. Par exemple, en dix ans de 1992 à 2002, les appareils photos numériques sont passés d'une image $376 \times 284 \approx 10^5$ pixels à une image $4096 \times 2730 \approx 11^6$ pixels. De plus, alors que dans les années 70 on ne s'intéressait qu'au traitement de signal mono-dimensionnel, on s'intéresse maintenant aux séquences d'images « 2D+t » et aux séquences d'image 3D « 3D+t ». On remarque donc une constante augmentation du nombre de données à traiter et du nombre de paramètres à estimer.

Comme la complexité des algorithmes de traitement du signal n'est pas linéaire, l'augmentation régulière de la puissance de calcul des matériels informatiques ne permet pas d'obtenir des résultats dans un temps de calcul raisonnable dans le cas de traitement de données de très grande dimension.

De plus, depuis quelques années l'apparition de grands instruments délivrant des données en continu a rendu indispensable l'accélération des algorithmes de traitement. Par exemple, le LHC au CERN peut transférer chaque seconde jusqu'à 10 gigaoctets de données. Les observatoires spatiaux fournissent des données en continu aux astrophysiciens. Ces données sont aussi de grande dimension et hétérogènes. En effet, les astrophysiciens optimisent le protocole de mesures en fonction de l'objet stellaire qu'ils observent. Le rapport entre les informations contenues dans les données et les informations *a priori* évolue avec le temps. Dans ce cadre, il est primordial de développer des approches non supervisées qui règlent ce compromis de manière automatique. Nous allons donc essayer de développer des approches non-supervisées permettant de résoudre des problèmes de grandes tailles le plus efficacement possible.

Pour cela, nous commençons par introduire la notion de problèmes inverses (déduire

des informations nouvelles sur l'objet d'intérêt à partir des mesures acquises). La plupart des problèmes inverses sont mal-posés. La résolution de ce type de problèmes s'appuie sur l'introduction d'informations *a priori* sur l'objet d'intérêt. Dans cette thèse, nous nous intéressons particulièrement aux problèmes de reconstruction des images. Pour résoudre ce type de problèmes, nous introduisons des informations sur l'image inconnue comme la parcimonie ou la régularité par morceau.

L'objectif principal de cette thèse est donc de développer des approches non-supervisées pour les problèmes de grande taille à l'aide d'informations *a priori* sur l'image inconnue. Pour atteindre cet objectif, nous devons d'abord déterminer des modèles *a priori* pertinents choisis pour l'image inconnue ainsi que pour les hyperparamètres qui règlent le compromis entre la fidélité aux données et la confiance aux informations *a priori*. Des bons modèles devraient décrire correctement les informations qu'on veut apporter et en même temps ne pas trop augmenter la complexité de calculs. Le premier modèle *a priori* introduit dans la littérature est l'*a priori* gaussien [Tik63]. Cet *a priori* a l'avantage d'avoir une expression simple qui permet une mise en œuvre efficace. Néanmoins, il présente l'inconvénient de favoriser les images douces ou proche d'une forme connue, ce qui dégrade les contours des images reconstruites. Pour résoudre ce problème, des *a priori* (des termes de régularisation) variés ont été proposés [GG84, Hub81, CBFAB97, HL89, GM85]. Dans le cadre classique de l'estimation au sens de maximum *a posteriori* (MAP), le problème d'estimation revient à l'optimisation d'un critère convexe ou non, qui nécessite un coût de calcul souvent très important. Les frères Geman [GG84] ont été dans les premiers à proposer une estimation en utilisant des variables auxiliaires que l'on appellera dans la suite des variables cachées afin de simplifier l'optimisation. Nous allons dans le cadre de cette thèse utiliser largement ce modèle à variables cachées afin de simplifier la résolution de notre problème d'inférence. Cette technique est très largement utilisé en maximisation de vraisemblance (algorithme EM) [DLR77, MK07] et dans les approches bayésiennes de type bayésien hiérarchique [CGLS08, Gio08].

Pour développer des approches non-supervisées, dans le cadre bayésien, on introduit également des lois *a priori* pour les hyperparamètres (par exemple, la loi de Jeffreys qui est la plus utilisée), ce qui conduit à une loi *a posteriori* conjointe. Dans ce cas non-supervisé, les lois *a posteriori* conjointes ont des expressions complexes (non-convexes). En conséquence, la plus grande difficulté consiste à trouver une approche suffisamment rapide en terme de temps de calcul pour des problèmes de grande dimension ($> 200\,000$ inconnues). En fait, dans le cas non-supervisé, le calcul de l'estimateur du maximum *a posteriori* conduit à un problème d'optimisation non-convexe qui est délicat. Pour s'affranchir de cette difficulté, beaucoup de travaux récents ont choisi de calculer l'estimateur de la moyenne *a posteriori* à l'aide des approches stochastiques de type Monte Carlo par Chaîne de Markov (MCMC). Néanmoins, ce type d'approches sont connues comme coûteuse en temps de calcul et donc ne peuvent pas s'adapter aux problèmes de très grandes dimensions. Dans le même contexte, afin de calculer l'estimateur de la moyenne *a posteriori*, comme nous le verrons dans le chapitre 2, les approches bayésiennes variationnelles ont de bonnes propriétés en terme de temps de calcul [SW10]. Néanmoins ce n'est pas suffisant pour les problèmes de très grandes tailles. En conséquence, le développement d'approches plus efficaces que les approches bayésiennes variationnelles existantes constitue un objectif principal de cette thèse. Un autre objectif est d'appliquer les approches bayésiennes variationnelles proposées aux problèmes inverses en traitement d'images pour développer des approches non-supervisés efficaces.

Cette thèse est organisée comme suit. Le chapitre introductif détaillera les approches bayésiennes pour les problèmes inverses mal-posés, et plus particulièrement les techniques permettant de calculer des estimateurs ponctuels. Le chapitre 2 traitera de plusieurs méthodes d'approximations analytiques de l'état de l'art : l'approximation de Laplace, l'approximation bayésienne variationnelle classique et l'approximation bayésienne variationnelle de type gradient. Le chapitre 3 sera consacré à la présentation de nos méthodes d'approximation bayésienne variationnelle accélérées, et un exemple d'application sur un problème linéaire simple sera donné pour montrer l'efficacité de nos approches. Le chapitre 4 exposera deux algorithmes bayésien variationnel non-supervisés que nous avons proposés qui utilisent soit un *a priori* de Variation Totale (TV), soit un *a priori* à base de variables cachées utilisant le modèle proposé par Geman et repris par Idier et Giovannelli [Idi01b, CI04, Gio08]. Le chapitre 5 présentera le développement de méthode de mélange de gaussienne dans un cadre général. Une illustration de l'intérêt de cette approche sera faite dans le cadre de problème classique de débruitage et déconvolution en utilisant une approche de type dictionnaire où l'on veut introduire de la parcimonie. Le chapitre 6 sera consacré aux conclusions et aux perspectives.

1.2 Problème inverse

Dans divers domaines d'applications en traitement du signal et des images : imagerie médicale [DMND⁺01, EF03], astronomie [ROGA08], géophysique [LT84], contrôle non destructif [Idi01a, chapitre 9], etc., on se retrouve souvent confronté au problème suivant : l'accès direct à une grandeur physique n'est pas toujours possible et on ne dispose que de *mesures indirectes*. Dans ce cas, on est amené à travailler dans le cadre de problèmes inverses, où le but est de retrouver la grandeur d'intérêt à partir de mesures indirectes.

En pratique, la résolution des problèmes inverses consiste à résoudre successivement un problème direct et un problème inverse. La résolution du problème direct consiste à construire un modèle direct qui met en relation la grandeur d'intérêt et les données observables. La résolution du problème inverse associé vise à déterminer la grandeur d'intérêt à partir des mesures observées, comme illustré par la figure 1.1.

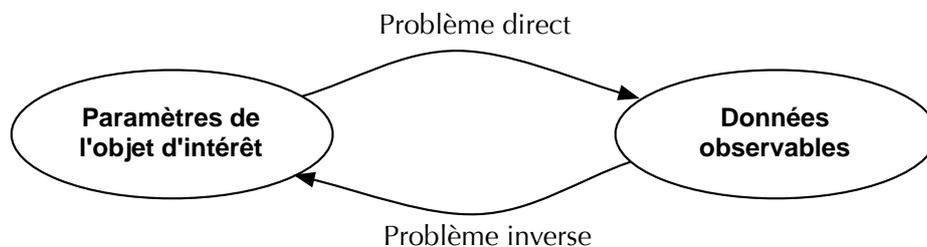


FIGURE 1.1 – Problème direct et Problème inverse

1.2.1 Modèle direct

Le modèle direct est construit, en général, à partir des lois physiques. Par exemple, dans le contexte de la tomographie par rayon X, la loi de Beer-Lambert décrit suffisamment fidèlement les phénomènes physiques, ce qui nous donne la relation entre les données (l'intensité du rayonnement qui a traversé l'objet d'intérêt) et les coefficients d'atténuation de l'objet.

Si on utilise un vecteur $\mathbf{y} \in \mathbb{R}^M$ pour représenter les données et un vecteur $\mathbf{x} \in \mathbb{R}^N$ pour les paramètres de l'objet d'intérêt, nous pouvons introduire un modèle direct comme ci-dessous :

$$\mathbf{y} = G(\mathbf{x}) \oplus \mathbf{n} \quad (1.1)$$

où G est un opérateur décrivant l'interaction entre l'objet d'intérêt et l'instrument de mesure. Néanmoins, $G(\mathbf{x})$ n'est jamais une modélisation parfaite de \mathbf{y} en raison des erreurs de modélisation, des erreurs de discrétisation et des erreurs instrumentales, qui sont incontournables. En conséquence, nous introduisons \mathbf{n} pour décrire l'incertitude liée à ces erreurs. Dans le cas d'un bruit additif, l'opérateur \oplus devient $+$. Concernant l'opérateur G , il peut être linéaire ou non en fonction des problèmes. Néanmoins, nous nous intéressons dans la suite à un opérateur linéaire ainsi qu'à un bruit additif. Dans un cas linéaire, nous remplaçons l'opérateur G par une matrice $\mathbf{A} \in \mathbb{R}^{M \times N}$, ce qui nous donne

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \quad (1.2)$$

où $\mathbf{n} \in \mathbb{R}^M$ représente le bruit de mesure et/ou les incertitudes sur le modèle.

1.2.2 Problèmes inverses

Le principe du problème inverse associé à (1.2) est de construire un estimateur $\hat{\mathbf{x}}$ des paramètres d'intérêt \mathbf{x} à partir de mesures \mathbf{y} [Idi01a]. Dans certains problèmes tels qu'un problème de déconvolution aveugle [CW98] ou un problème de séparation de source aveugle [BAMCM97], la matrice \mathbf{A} n'est aussi pas complètement connue et a besoin d'être estimée en même temps que \mathbf{x} . Mais dans cette thèse, nous nous intéressons seulement aux problèmes non-aveugles où la matrice \mathbf{A} est parfaitement connue.

Même si les problèmes directs sont généralement *bien posés*, les problèmes inverses associés sont cependant *mal-posés*. Un problème est dit bien posé au sens d'Hadamard s'il satisfait les conditions suivantes :

- la solution existe,
- la solution est unique,
- la solution dépend de façon continue des données (i.e. une petite variation dans les données n'introduit pas de grande variation dans la solution).

Un problème est donc dit mal posé au sens d'Hadamard lorsque l'une de ces conditions n'est pas respectée. Malheureusement, la plupart des problèmes inverses rencontrés en traitement du signal et des images sont mal-posés. Dans le cas linéaire décrit par (1.2), ce fait vient de la matrice \mathbf{A} qui est non-inversible ou mal-conditionnée.

Pour s'affranchir de la difficulté posée par la matrice non-inversible, une approche naturelle pour résoudre (1.2) consiste à utiliser la méthode des *moindres carrés* pour

choisir un estimateur $\hat{\mathbf{x}}_{MC}$. Cet estimateur est calculé en minimisant la norme L^2 du résidu du modèle, ce qui nous permet d'écrire

$$\hat{\mathbf{x}}_{MC} = \arg \min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 \quad (1.3)$$

On peut facilement montrer que la minimisation de cette norme L^2 conduit à la résolution de l'équation suivante :

$$\mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{y}. \quad (1.4)$$

Sous condition que $\mathbf{A}^T \mathbf{A}$ soit inversible, la solution des moindres carrés est unique et donnée par

$$\hat{\mathbf{x}}_{MC} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y}. \quad (1.5)$$

Lorsque la matrice $\mathbf{A}^T \mathbf{A}$ n'est pas inversible, la solution n'est plus unique (quand le problème est sous-déterminé par exemple, i.e. $M < N$). Dans ce cas, la *solution généralisée* peut être établie en choisissant une solution de norme euclidienne minimale parmi toutes les solutions vérifiant $\mathbf{y} = \mathbf{A}\mathbf{x}$.

Malgré le fait qu'il est possible de se ramener à une solution unique avec la *solution généralisée*, celle-ci n'est pas nécessairement stable, i.e., pas robuste du bruit. En fait, la méthode des moindres carrés n'exploite que des informations apportées par les données ce qui ne sont, en général, pas suffisantes pour avoir une solution unique ou robuste au bruit. Pour s'affranchir de cette difficulté, d'autres méthodes d'inversions sont développées en introduisant des informations supplémentaires concernant le paramètre d'intérêt \mathbf{x} [Idi01a]. En général, les méthodes d'inversion pour résoudre les problèmes inverses mal-posés peuvent être divisées en deux catégories : les approches pénalisées et les approches statistiques. Néanmoins, nous nous intéressons dans cette thèse aux approches statistiques, plus particulièrement, aux approches bayésiennes. Notons qu'un grand nombre de solutions pénalisées peuvent être interprétées dans un cadre bayésien.

1.3 Inférence bayésienne

L'inférence bayésienne est basée sur la modélisation probabiliste du vecteur des paramètres inconnus \mathbf{x} et du vecteur du bruit \mathbf{n} . Par conséquent, les informations apportées par les mesures ainsi que les connaissances *a priori* sur l'objet cherché sont toutes modélisées sous forme de densités de probabilité. L'inférence bayésienne s'appuie principalement sur la règle de Bayes qui nous permet d'obtenir une loi *a posteriori* de paramètres inconnus \mathbf{x} sachant les mesures \mathbf{y} . Cette loi résume toute l'information disponible sur les paramètres inconnus. En utilisant la règle de Bayes, la loi *a posteriori* est donnée par

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})} \quad (1.6)$$

où

- $p(\mathbf{y}|\mathbf{x})$ est la *vraisemblance* qui rend compte des incertitudes sur les données observées. Cette probabilité est obtenue à partir du modèle direct et la modélisation probabiliste du bruit,
- $p(\mathbf{x})$ est la loi *a priori* sur les paramètres d'intérêt, qui représente nos connaissances *a priori* sur l'objet avant de faire la mesure,

1.3.1 - La vraisemblance

- $p(\mathbf{y}) = \int p(\mathbf{y}|\mathbf{x})p(\mathbf{x})d\mathbf{x}$ est la loi marginale qui assure la normalisation de la loi *a posteriori*. Elle est donc souvent appelée constante de normalisation. Par ailleurs, elle est connue aussi sous le nom de *fonction de partition* dans la littérature de physique statistique et sous le nom d'*évidence du modèle* lorsqu'elle est utilisée pour la sélection de modèle. Cette loi marginale n'est généralement pas explicitement connue. Son obtention nécessite de calculer l'intégrale ci-dessus, ce qui est, en général, difficile.

À partir de (1.6), nous pouvons voir que la loi *a posteriori* est proportionnelle au produit de la vraisemblance et de la loi *a priori*. Nous allons donner tout d'abord plus de détails sur ces deux distributions et ensuite la manière d'inférer sur les paramètres à partir de la loi *a posteriori*, notamment les estimateurs ponctuels usuels, les difficultés rencontrées pendant les calculs ainsi que les techniques utilisées pour s'en affranchir.

1.3.1 La vraisemblance

La vraisemblance se déduit de la modélisation du problème direct et d'une modélisation statistique de la composante de bruit \mathbf{n} . En traitement d'images, l'hypothèse du bruit gaussien et du bruit poissonnien [LCA07] sont couramment utilisées. Dans ce travail, on n'étudiera que le bruit gaussien qui nous permettra de faire une interprétation statistique de l'approche des *moindres carrés*.

Supposons que le bruit est un bruit gaussien de moyenne nulle et de matrice de covariance définie positive Σ . Selon l'hypothèse du bruit et le modèle direct (1.2), la loi conditionnelle de \mathbf{y} sachant \mathbf{x} , qui est appelée la *vraisemblance*, est de la forme suivante :

$$p(\mathbf{y}|\mathbf{x}, \Sigma) = (2\pi)^{-M/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax})\right). \quad (1.7)$$

Si l'on suppose en plus que les composantes du vecteur de bruit sont indépendantes et identiquement distribuées (*i.i.d.*), la matrice de covariance Σ devient diagonale et ses composantes diagonales sont donc égales à une même valeur, notée ici σ_n^2 . En conséquence, la vraisemblance devient

$$p(\mathbf{y}|\mathbf{x}, \sigma_n^2) \propto \exp\left(-\frac{\|\mathbf{y} - \mathbf{Ax}\|^2}{2\sigma_n^2}\right) \quad (1.8)$$

- **Lien avec la méthode des moindres carrés**

Un estimateur ponctuel qu'on peut construire avec seulement la vraisemblance est le maximum de vraisemblance (MV) qui est défini par

$$\begin{aligned} \hat{\mathbf{x}}_{MV} &= \arg \max_{\mathbf{x}} \{p(\mathbf{y}|\mathbf{x})\} \\ &= \arg \min_{\mathbf{x}} \{-\ln p(\mathbf{y}|\mathbf{x})\} \\ &= \arg \min_{\mathbf{x}} J_{MV}(\mathbf{x}) \end{aligned} \quad (1.9)$$

où $J_{MV}(\mathbf{x}) = -\ln p(\mathbf{y}|\mathbf{x})$.

Lorsque la vraisemblance est donnée par (1.7),

$$J_{MV}(\mathbf{x}) = \frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax}). \quad (1.10)$$

Dans ce cas, $J_{MV}(\mathbf{x})$ correspond au critère de la méthode des moindres carrés pondérés, noté ici $J_0(\mathbf{x})$, qui est aussi appelé le terme d'adéquation aux données dans le cadre des approches pénalisées. Lorsque la matrice de covariance est proportionnelle à la matrice identité : $\Sigma = \sigma_n^2 \mathbf{I}$, i.e. la vraisemblance est donnée par (1.8), le critère $J_{MV}(\mathbf{x})$ correspond au critère des moindres carrés (voir (1.3)). Celui ci montre le lien décrit par

$$J_{MV}(\mathbf{x}) = -\ln p(\mathbf{y}|\mathbf{x}) \leftrightarrow J_0(\mathbf{x}). \quad (1.11)$$

1.3.2 Loi *a priori*

Dans le cadre bayésien, l'information *a priori* sur l'objet cherché prend aussi la forme d'une probabilité :

$$p(\mathbf{x}) \propto \exp(-\Phi(\mathbf{x})). \quad (1.12)$$

où $\Phi : \mathbb{R}^N \rightarrow \mathbb{R}$ une fonction dont les propriétés restent à définir. Le choix de la loi *a priori* est un problème non-trivial et lié au problème à résoudre. Dans la littérature, de nombreuses possibilités ont été proposées. La loi *a priori* gaussienne a été premièrement introduite et largement utilisée [Tik63, Dem89]. Néanmoins, sa capacité à reconstruire des composantes non-linéaires, tels que l'emplacement et la grandeur des discontinuités, est limitée. Pour pouvoir décrire les propriétés non-linéaires de l'objet à chercher, les champs aléatoires de Markov composés [JW91], les champs aléatoires de Markov avec des fonctions de potentiel non-quadratiques telles que la valeur absolue (une loi de Laplace) [CDS01], les potentiels dit L_2 - L_1 quadratique autour de l'origine puis linéaire [Hub81], et les distributions à queue lourde, par exemple, la loi de Student, ont été aussi utilisés. Par ailleurs, dans diverses applications du traitement d'image ou traitement du signal, une caractéristique présente souvent dans notre objet d'intérêt est la régularité par morceau. Pour cela, un *a priori* de la variation totale [ROF92] est souvent utilisé.

Enfin, de nombreux modèles introduisant des variables cachées ont été étudiés dans la littérature. Leurs bonnes propriétés ont fait que nous les avons utilisés dans la suite de cette thèse.

- **Les modèles à variables cachées**

Comme son nom le suggère, les modèles à variables cachées s'appuient sur l'introduction des variables cachées, notées ici \mathbf{z} , dans la modélisation probabiliste des connaissances *a priori* sur notre objet d'intérêt \mathbf{x} . Ils consistent à définir une loi conditionnelle de l'objet d'intérêt \mathbf{x} sachant les variables cachées \mathbf{z} qui en fonction de leur valeur modifient la probabilité et donc l'information sur l'objet. Un avantage de ce type de modèles est d'améliorer la précision de modélisation en utilisant une loi de forme simple pour \mathbf{x} sachant \mathbf{z} , la loi gaussienne, par exemple. Un exemple d'un tel modèle est le modèle de mélange de gaussiennes fini ou infini qui est largement utilisé dans diverses applications. Par exemple, le modèle de mélange de gaussiennes fini où les variables cachées représentent des poids de chaque composante gaussienne a été utilisé dans la reconnaissance du locuteur [RQD00], dans la séparation de sources [SD00, SD04], en segmentation [GRG06], etc. Le modèle de mélange de gaussiennes infini fournit un ensemble des distributions à queue lourde telles que la distribution de Student, gaussienne généralisée et les distributions α -stables qui sont beaucoup utilisées pour introduire des informations de parcimonie [CGLS08, PSWS03, ATB03, BD06].

1.3.3 Estimateurs ponctuels dans l'inférence bayésienne

En pratique, on est très souvent intéressé par un estimateur ponctuel. La loi *a posteriori* $p(\mathbf{x}|\mathbf{y})$ permet la construction d'un certain nombre d'estimateurs aux caractéristiques diverses, parmi lesquels l'estimateur du maximum *a posteriori* (MAP) et l'estimateur de l'espérance *a posteriori* (EAP) sont les plus utilisés.

- L'estimateur du maximum *a posteriori* (MAP) s'obtient en calculant le mode de la loi *a posteriori* :

$$\hat{\mathbf{x}}_{MAP} = \arg \max_{\mathbf{x}} \{p(\mathbf{x}|\mathbf{y})\} \quad (1.13)$$

Par conséquent, cet estimateur est le point le plus probable au sens de la loi *a posteriori*.

- L'estimateur de l'espérance *a posteriori* (EAP) calcule l'espérance de la loi *a posteriori* :

$$\hat{\mathbf{x}}_{EAP} = \int \mathbf{x}p(\mathbf{x}|\mathbf{y})\mathbf{d}\mathbf{x}. \quad (1.14)$$

On peut montrer que c'est l'estimateur qui minimise l'erreur quadratique moyenne [Idi01a].

Ici, avec la définition de l'estimateur MAP, nous pouvons construire un lien entre cet estimateur et l'estimateur des approches pénalisées dans le cadre déterministe.

- **Lien avec les approches pénalisées**

Selon la définition de l'estimateur MAP (1.13) et la formule de la loi *a posteriori* (1.6), nous pouvons facilement déduire ce qui suit :

$$\begin{aligned} \hat{\mathbf{x}}_{MAP} &= \arg \min_{\mathbf{x}} \{-\ln p(\mathbf{x}|\mathbf{y})\} \\ &= \arg \min_{\mathbf{x}} \{-\ln p(\mathbf{y}|\mathbf{x}) - \ln p(\mathbf{x})\}. \end{aligned} \quad (1.15)$$

Nous avons montré précédemment un lien entre $-\ln p(\mathbf{y}|\mathbf{x})$ et le terme d'adéquation aux données (voir (1.11)). De façon analogue, on peut associer une loi *a priori* à un terme de régularisation $\mathcal{R}(\mathbf{x})$ pour que la loi de probabilité ci-dessous ait effectivement un sens :

$$p(\mathbf{x}) = \frac{1}{Z_2} \exp(-\alpha\mathcal{R}(\mathbf{x})) \quad (1.16)$$

où Z_2 est la constante de normalisation si elle existe et α est un paramètre d'échelle. Par conséquent, on peut construire le lien :

$$-\ln p(\mathbf{x}) \leftrightarrow \mathcal{R}(\mathbf{x}) \quad (1.17)$$

Sous réserve que l'on puisse établir des liens (1.11) et (1.16), nous pouvons associer le critère à minimiser pour calculer l'estimateur MAP (voir (1.15)) au critère des approches pénalisées tel que

$$-\ln p(\mathbf{x}|\mathbf{y}) \leftrightarrow J_0(\mathbf{x}) + \lambda\mathcal{R}(\mathbf{x}) \quad (1.18)$$

avec λ le paramètre de régularisation qui contrôle le compromis entre l'attache aux données et les informations *a priori*.

Dans (1.13) et (1.14), on peut voir que l'obtention de l'estimateur MAP nécessite une méthode d'optimisation pour le critère donné par $J_{MAP}(\mathbf{x}) = -\ln p(\mathbf{x}|\mathbf{y})$ et le calcul de l'estimateur EAP nécessite une méthode permettant de calculer une intégrale. En pratique, les calculs de ces estimateurs ne sont généralement pas faciles. L'expression analytique de ces estimateurs peut être obtenue pour des problèmes simples, par exemple, des problèmes avec un *a posteriori* gaussien.

Nous considérons ici un bruit blanc gaussien de moyenne nulle et de variance σ_n^2 qui nous permet d'obtenir une vraisemblance donnée par (1.8). Par ailleurs, nous considérons une loi *a priori* gaussienne donnée par

$$p(\mathbf{x}|\sigma_p^2) \propto \exp\left(-\frac{\|\mathbf{\Gamma}\mathbf{x}\|^2}{2\sigma_p^2}\right) \quad (1.19)$$

où σ_p^2 est la variance de l'*a priori* et $\mathbf{\Gamma}$ est une matrice qui influence les informations apportées dans le problème considéré. Par exemple, dans le cas où $\mathbf{\Gamma}$ est la matrice identité, on introduit des informations que la norme euclidienne de l'objet d'intérêt \mathbf{x} est petite. Par ailleurs, dans certains cas, les opérateurs de différence finie sont utilisés pour apporter des informations que l'objet est lisse.

Dans ce cas, en utilisant la règle de Bayes, nous pouvons obtenir la loi *a posteriori* qui s'écrit comme :

$$p(\mathbf{x}|\mathbf{y}, \sigma_n^2, \sigma_p^2) \propto \exp\left(-\frac{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2\sigma_n^2} - \frac{\|\mathbf{\Gamma}\mathbf{x}\|^2}{2\sigma_p^2}\right), \quad (1.20)$$

qui est aussi une distribution gaussienne. Dans ce cas, l'estimateur MAP et l'estimateur du EAP sont confondus.

$$\begin{aligned} \hat{\mathbf{x}}_{MAP, EAP} &= \arg \min\{-\ln p(\mathbf{x}|\mathbf{y}, \sigma_n^2, \sigma_p^2)\} \\ &= \arg \min\{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + \lambda\|\mathbf{\Gamma}\mathbf{x}\|^2\} \end{aligned} \quad (1.21)$$

où $\lambda = \sigma_n^2/\sigma_p^2$, qui peut être identifié comme le paramètre de régularisation. Ce problème d'optimisation admet une solution analytique donnée par

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A} + \lambda \mathbf{\Gamma}^T \mathbf{\Gamma})^{-1} \mathbf{A}^T \mathbf{y}. \quad (1.22)$$

Nous pouvons voir dans l'équation (1.22) que l'estimateur $\hat{\mathbf{x}}$ est linéaire vis-à-vis du vecteur de mesures \mathbf{y} . Par conséquent, sa capacité à reconstruire des composantes non-linéaires est limitée. Dans ce cas, les *a priori* non-gaussiens sont utilisés pour surmonter cette limitation.

Néanmoins, pour des problèmes non-gaussiens, les solutions explicites n'existent pas toujours. Des algorithmes plus compliqués sont nécessaires pour trouver l'estimateur MAP ou l'estimateur EAP, comme on le verra dans la partie 1.4.

1.3.4 Estimation non-supervisée

Comme on a vu dans la partie précédente, l'estimateur de \mathbf{x} (voir (1.22), par exemple) dépend des valeurs des hyperparamètres (σ_n^2 et σ_p^2 dans l'exemple gaussien précédent) choisies. Historiquement, ses valeurs sont choisies manuellement et telles approches sont dites *supervisées*. Néanmoins, cette façon de choisir les hyperparamètres s'avère peu pratique

1.3.4 - Estimation non-supervisée

et peu efficace lorsque le nombre d'hyperparamètres devient important (> 3). Le cadre bayésien permet de résoudre ce problème en définissant des approches non-supervisées où les hyperparamètres sont déterminés de façon automatique. Dans le cadre bayésien, les deux approches les plus répandues pour la détermination des hyperparamètres sont le *maximum de vraisemblance marginale* et l'approche *bayésienne hiérarchique*.

Dans la suite, nous utilisons θ_1 (σ_n^2 dans le cas gaussien) pour représenter le paramètre qui définit la vraisemblance et θ_2 (σ_p^2 dans le cas gaussien) pour le paramètre de la loi *a priori*. De plus, $\theta = (\theta_1, \theta_2)$ est l'ensemble des hyperparamètres.

- 1) **Maximum de vraisemblance marginale** L'estimation au sens du maximum de vraisemblance marginale de θ se déduit de la loi marginale de \mathbf{y} qui peut être calculée à partir de la loi jointe en marginalisant par rapport à \mathbf{x} , c'est à dire,

$$\begin{aligned}\hat{\theta} &= \arg \max_{\theta} p(\mathbf{y}|\theta) \\ &= \arg \max_{\theta} \int p(\mathbf{y}, \mathbf{x}|\theta) d\mathbf{x}\end{aligned}\tag{1.23}$$

Néanmoins, il est souvent lourd voire impossible à calculer lorsque la loi *a posteriori* n'est pas gaussienne.

- 2) **L'approche bayésienne hiérarchique** L'approche bayésienne hiérarchique ou entièrement bayésienne [Mol94], [HBJ+97] considère que θ suit une loi *a priori* $p(\theta)$, ce qui nous permet d'estimer θ de la même façon que \mathbf{x} . La loi *a posteriori* jointe de \mathbf{x} et θ s'exprime alors :

$$p(\mathbf{x}, \theta|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x}, \theta_1)p(\mathbf{x}|\theta_2)p(\theta)}{p(\mathbf{y})}.\tag{1.24}$$

Nous pouvons ensuite construire les estimateurs tels que le maximum *a posteriori* ou l'espérance *a posteriori* à partir de la loi *a posteriori* jointe. Cette approche bayésienne hiérarchique est beaucoup utilisée dans cette thèse. Cependant, cette loi est en général complexe et justement connue à une constante multiplicative près. Par conséquent, elle ne peut être utilisée directement. Par exemple, le calcul d'un estimateur du MAP conduit en général à un problème d'optimisation non convexe difficile à résoudre. L'estimateur EAP n'est pas directement calculable car on ne connaît pas la constante de normalisation. Dans ce cas, le défi principal est de récupérer cette loi *a posteriori* à l'aide des méthodes d'approximation comme on va l'introduire dans la suite.

1.4 Algorithmes de calculs dans l'inférence bayésienne

Dans les cas où les distributions *a priori* de \mathbf{x} ne sont pas gaussiennes, les lois *a posteriori* obtenues ne sont pas gaussiennes non plus. En conséquence, l'estimateur MAP et l'estimateur EAP n'ont plus une forme simple et c'est plus difficile voire impossible d'obtenir des expressions analytiques de ces estimateurs. Par conséquent, les méthodes d'optimisation ou les techniques d'intégration sont nécessaires pour trouver ces estimateurs. La suite de cette partie est donc consacrée à introduire plusieurs méthodes pour calculer ces estimateurs.

1.4.1 Méthodes pour calculer l'estimateur MAP

Sous l'hypothèse que le bruit est gaussien, le calcul de l'estimateur MAP revient à minimiser le critère suivant :

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax}) + \Phi(\mathbf{x}). \quad (1.25)$$

Lorsque la loi *a priori* n'est pas gaussienne, $J(\mathbf{x})$ n'est pas quadratique. Dans ce cas, le minimiseur du critère $J(\mathbf{x})$ peut s'obtenir rarement de façon analytique et les techniques d'optimisation itératives sont généralement utilisées pour approcher son minimiseur. En fait, même avec un *a priori* quadratique, lorsque la taille des données devient très grande, le minimiseur analytique (voir (1.22)) n'est plus calculable et la solution ne peut être approchée par une méthode itérative. Des exemples des méthodes itératives couramment utilisées sont la méthode du gradient, la méthode du gradient conjugué, la méthode de Newton ou une méthode plus récente, la méthode d'optimisation de sous-espace. Ces exemples sont tous des méthodes d'optimisation locales permettant de converger vers les minimiseurs locaux du problème. Lorsque le critère est convexe, il est unimodal. Dans ce cas, les méthodes d'optimisation locales permettent de trouver le minimum global du critère. Dans le cas contraire, si le critère est non-convexe, sa minimisation nécessite une approche d'optimisation globale telle que le recuit simulé proposée par Kirkpatrick *et al.* [KGV⁺83], voir aussi [GR92], ou les algorithmes génétiques [Hol92].

Dans le même contexte des problèmes d'optimisation non-quadratiques voir non-convexes, il existe également des algorithmes *semi-quadratiques* proposés par Geman *et al.* [GR92] [GY95] dont le principe consiste à construire un *critère augmenté* qui dépend à la fois de la variable \mathbf{x} et d'une variable auxiliaire \mathbf{b} à l'aide des outils de dualité. Ce nouveau critère possède les mêmes extréma que le critère original avec une structure remarquable : ce nouveau critère est quadratique par rapport à \mathbf{x} et son optimisation vis-à-vis de la variable auxiliaire est explicite. Ceci permet d'obtenir des estimateurs non-linéaires en utilisant les algorithmes d'optimisation standards.

Algorithmes semi-quadratiques (SQ)

Le critère augmenté des algorithmes SQ sera noté dans la suite par $K(\mathbf{x}, \mathbf{b})$ et satisfait la relation :

$$J(\mathbf{x}) = \min_{\mathbf{b} \in B} K(\mathbf{x}, \mathbf{b}) \quad (1.26)$$

pour un ensemble B à préciser. Avec (1.26), nous pouvons voir que

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} J(\mathbf{x}) \Leftrightarrow \exists \hat{\mathbf{b}}; (\hat{\mathbf{x}}, \hat{\mathbf{b}}) = \arg \min_{\mathbf{x}, \mathbf{b}} K(\mathbf{x}, \mathbf{b}). \quad (1.27)$$

Par conséquent, la minimisation de $K(\mathbf{x}, \mathbf{b})$ vis-à-vis de \mathbf{x} et de \mathbf{b} nous permet d'obtenir un minimiseur de $J(\mathbf{x})$. L'intérêt est que le critère augmenté a une structure spéciale qui est plus facile à optimiser. Pour construire le critère augmenté $K(\mathbf{x}, \mathbf{b})$, [GR92] et [GY95] ont proposé deux structures semi-quadratiques différentes.

Supposons que

$$\Phi(\mathbf{x}) = \sum_j \phi(u_j) \quad \text{où } u_j = \mathbf{v}_j^T \mathbf{x} - \epsilon_j \quad (1.28)$$

$$\mathbf{v}_j \in \mathbb{R}^N, \epsilon_j \in \mathbb{R},$$

1.4.1 - Méthodes pour calculer l'estimateur MAP

les deux structures sont respectivement de la forme suivante :

$$K_{GR}(\mathbf{x}, \mathbf{b}) = \frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax}) + \sum_j \left[\frac{1}{2} b_j (\mathbf{v}_j^T \mathbf{x} - \epsilon_j)^2 + \psi(b_j) \right] \quad (1.29)$$

$$K_{GY}(\mathbf{x}, \mathbf{b}) = \frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax}) + \frac{1}{\alpha} \sum_j \left[\frac{1}{2} (\mathbf{v}_j^T \mathbf{x} - \epsilon_j - b_j)^2 + \zeta_\alpha(b_j) \right] \quad (1.30)$$

où les fonctions ψ et ζ_α sont choisies pour satisfaire (1.26). Les critères augmentés K_{GR} et K_{GY} sont dits *semi-quadratiques*, comme ils sont des fonctions quadratiques de \mathbf{x} lorsque les variables $\mathbf{b} = (b_j)$ sont fixées. Néanmoins, pour une fonction potentiel ϕ , les fonctions ψ et ζ_α n'existent pas toujours. Il a été montré dans [Idi01b] que les fonctions ψ et ζ_α peuvent être bien définies par les relations de dualité convexe lorsque ϕ satisfait certaines conditions, qui sont détaillées dans les propositions suivantes :

Proposition 1. (GR) : *Considérons les fonctions ϕ qui satisfont les hypothèses suivantes*

- 1) ϕ est paire,
- 2) $\phi(\sqrt{\cdot})$ est concave sur \mathbb{R}_+ ,
- 3) ϕ est continue en zéro et C^1 sur $\mathbb{R} \setminus \{0\}$,

et considérons $\psi(b) = \sup_{x \in \mathbb{R}} (\phi(x) - bx^2/2)$. Alors,

$$\begin{aligned} \inf_{b \in \mathbb{R}_+} (bx^2/2 + \psi(b)) &= \phi(x) \\ \arg \min_{b \in \mathbb{R}_+} (bx^2/2 + \psi(b)) &= \phi'(x)/x. \end{aligned} \quad (1.31)$$

De (1.31), il est facile de déduire (1.26) pour $K = K_{GR}$.

Proposition 2. (GY) : *Considérons les fonctions ϕ telles que $(\cdot)^2/2 - \alpha\phi$ est une fonction convexe pour $\alpha > 0$, de sorte que $\zeta_\alpha(b) = \sup_{x \in \mathbb{R}} (\alpha\phi(x) - (x - b)^2/2)$ soit bien définie (voir [GY95] pour la preuve). Alors,*

$$\begin{aligned} \inf_{b \in \mathbb{R}} ((x - b)^2/2 + \zeta_\alpha(b)) / \alpha &= \phi(x) \\ \arg \min_{b \in \mathbb{R}} ((x - b)^2/2 + \zeta_\alpha(b)) &= x - \alpha\phi'(x) \end{aligned} \quad (1.32)$$

où la deuxième équation tient si ϕ est différentiable.

Grâce aux avantages structurels de K_{GR} et K_{GY} , on préfère les minimiser plutôt que minimiser J . Les minimisations de K_{GR} et K_{GY} sont effectuées en alternant les deux étapes suivantes : la minimisation des critères par rapport à \mathbf{x} en fixant \mathbf{b} et la minimisation des critères par rapport à \mathbf{b} en fixant \mathbf{x} . Ces deux minimisations sont simples car l'un est un critère quadratique et l'autre un critère séparable. Pour calculer le minimiseur, [GR92] [GY95] ont proposé d'utiliser la méthode du recuit simulé fondée sur le principe de l'échantillonnage de Gibbs alterné. L'avantage de cette méthode est qu'elle permet de trouver le minimum global d'un critère donné, même un critère non-convexe. Néanmoins, cette méthode est généralement coûteuse en temps de calcul. Par conséquent, de nombreux travaux tels que [CBFAB94, BI96, VO98] s'intéressent aux algorithmes déterministes car on peut obtenir des équations de mise à jour explicites pour les deux sous-problèmes correspondant respectivement à la minimisation par rapport à \mathbf{x} et à la minimisation par rapport à \mathbf{b} . En général, les algorithmes déterministes sont plus rapides que la méthode du recuit simulé, mais ils ne peuvent pas garantir le minimum global sauf dans le cas où les critères augmentés sont convexes (voir [Idi01b] pour la convexité des critères augmentés).

Lien entre les algorithmes semi-quadratiques et l'algorithme Espérance-Maximisation (EM)

Les algorithmes SQ sont des outils performants pour l'optimisation d'un critère non-quadratique. Ce type d'algorithmes a été proposé dans le cadre déterministe. Néanmoins, une interprétation bayésienne a été donnée dans [CI04] qui dit qu'il existe un lien entre les algorithmes SQ et l'algorithme EM. Ce lien est construit à l'aide des modèles *a priori* à variables cachées.

Un avantage des modèles à variables cachées est qu'ils nous permettent d'obtenir des lois *a priori* ayant les propriétés désirées et en même temps d'avoir des formulations plus simples lorsqu'on calcule les estimateurs ponctuels. Un exemple de tels modèles *a priori* est le mélange de gaussiennes, qui nous sert ici pour donner le lien entre les algorithmes SQ et l'algorithme EM.

Nous avons considéré une loi *a priori*

$$p(\mathbf{x}) \propto \exp\left(-\sum_j \phi(u_j)\right), \quad (1.33)$$

où $u_j = \mathbf{v}_j^T \mathbf{x} - \epsilon_j$. Cette loi est propre lorsque la matrice $\mathbf{V} = (\mathbf{v}_j)$ est de rang N et $\int_{\mathbb{R}} \exp(-\phi(u)) du < \infty$. Cette condition est supposée dans la suite.

Pour construire le lien entre les algorithmes SQ et l'algorithme EM, nous introduisons une variable aléatoire scalaire U dont la densité de probabilité est $p_U(u) \propto \exp(-\phi(u))$. En particulier, nous supposons que U est distribuée selon une loi de mélange de gaussiennes.

Mélange de gaussiennes Nous définissons ici deux types de mélange : un mélange de gaussiennes par changement d'échelle (GSM) et un mélange de gaussiennes par changement de positions (GPM).

GSM Dans le cas d'un GSM, nous supposons que

- 1) $U = G/\sqrt{B}$ avec G et B indépendantes,
- 2) la variable $G \sim \mathcal{N}(0, 1)$ et la variable $B > 0$.

Dans ce cas là, on peut facilement obtenir

$$p_U(u_j) = \int_{\mathbb{R}^+} p_G(u_j \sqrt{b_j}) \sqrt{b_j} p_B(b_j) db_j \quad (1.34)$$

où p_G et p_B représentent la densité de probabilité de G et B , respectivement.

A l'aide de p_U , nous pouvons réécrire (1.33) comme suit

$$\begin{aligned} p(\mathbf{x}) &\propto \prod_j p_U(u_j) \\ &\propto \prod_j \int_{\mathbb{R}^+} p_G(u_j \sqrt{b_j}) \sqrt{b_j} p_B(b_j) db_j. \end{aligned} \quad (1.35)$$

1.4.1 - Méthodes pour calculer l'estimateur MAP

Notons le vecteur des variables cachées $\mathbf{b} = (b_j)$. Dans le cas d'un GSM, p_G est la densité d'une loi gaussienne. Dans ce cas, à partir de (1.35) et la relation entre u_j et \mathbf{x} , on peut déduire que la loi conditionnelle de \mathbf{x} sachant \mathbf{b} est donnée par

$$\begin{aligned} p(\mathbf{x}|\mathbf{b}) &= \mathcal{N}(\mathbf{m}_x, \mathbf{P}_x^{-1}) \quad \text{avec} \quad \mathbf{P}_x = \mathbf{V}\text{Diag}(\mathbf{b})\mathbf{V}^T \\ &\quad \text{et} \quad \mathbf{m}_x = \mathbf{P}_x^{-1}\mathbf{V}\text{Diag}(\mathbf{b})\boldsymbol{\epsilon}. \end{aligned} \quad (1.36)$$

où $\text{Diag}(\mathbf{b})$ représente une matrice diagonale dont les éléments diagonaux sont donnés par le vecteur \mathbf{b} .

GPM De façon analogue, dans le cas d'un GPM, nous supposons que

- 1) $U = \sigma G + B$ avec $\sigma > 0$ et G, B indépendantes,
- 2) la variable $G \sim \mathcal{N}(0, 1)$.

Dans ce cas là, la densité p_U satisfait

$$p_U(u_j) = \frac{1}{\sigma} \int_{\mathbb{R}} p_G\left(\frac{u_j - b_j}{\sigma}\right) p_B(b_j) db_j. \quad (1.37)$$

Comme dans le cas GSM, nous pouvons réécrire (1.33) comme suit

$$\begin{aligned} p(\mathbf{x}) &\propto \prod_j p_U(u_j) \\ &\propto \prod_j \frac{1}{\sigma} \int p_G\left(\frac{u_j - b_j}{\sigma}\right) p_B(b_j) db_j. \end{aligned} \quad (1.38)$$

Dans le cas d'un GPM, p_G est aussi la densité d'une loi gaussienne. Nous pouvons donc obtenir

$$\begin{aligned} p(\mathbf{x}|\mathbf{b}) &= \mathcal{N}(\mathbf{m}_x, \mathbf{P}_x^{-1}) \quad \text{avec} \quad \mathbf{P}_x = \sigma^{-2}\mathbf{V}\mathbf{V}^T \\ &\quad \text{et} \quad \mathbf{m}_x = (\mathbf{V}\mathbf{V}^T)^{-1}\mathbf{V}(\boldsymbol{\epsilon} + \mathbf{b}). \end{aligned} \quad (1.39)$$

L'algorithme Espérance-Maximisation (EM) Notre modèle probabiliste dépend de variables cachées non-observables \mathbf{b} . Dans ce cas, l'algorithme Espérance-Maximisation (EM) est un algorithme bien connu pour trouver l'estimateur du maximum de vraisemblance des paramètres [DLR77], et par extension, l'estimateur du maximum *a posteriori* [LH87, MK07, BD06] en considérant que les variables cachées comme les *données manquantes*. Cet algorithme permet d'augmenter la vraisemblance ou la probabilité *a posteriori* à chaque itération. Par ailleurs, cet algorithme peut être considéré comme une sous-classe de l'algorithme MM (Majoration-Minimisation ou Minoration-Maximisation) [HL04]. Néanmoins, il ont des inconvénients :

- il ne peut pas garantir de donner l'estimateur du maximum de vraisemblance ou l'estimateur du maximum *a posteriori* lorsque le critère à optimiser est multimodal,
- la convergence peut être lente. En conséquence, des variantes ont été proposées pour accélérer la vitesse de convergence, tels que ceux qui utilisant la méthode du gradient conjugué [JJ93] ou la méthode du Newton [Lan95, JJ97],

- l'algorithme EM est utile pour la famille de distributions exponentielles. Mais lorsque les distributions ne sont pas dans la famille exponentielle, l'algorithme EM peut rencontrer des difficultés à obtenir des expressions analytiques. Pour remédier à ce problème, des variantes telles que l'algorithme EM stochastique (SEM) [CD85] et l'algorithme Monte-Carlo EM (MCEM) [WT90] peuvent être utilisées. Ces variantes permettent aussi de réduire le risque de tomber dans un maximum local.

Calcul de l'estimateur MAP Pour notre problème avec un *a priori* mélange de gaussiennes, nous calculons l'estimateur MAP de \mathbf{x} qui est le maximiseur de la loi *a posteriori* $p(\mathbf{x}|\mathbf{y})$. Pour ce faire, nous utilisons l'algorithme EM qui alterne deux étapes :

- une étape d'évaluation de l'espérance (E) qui calcule

$$\begin{aligned}
 Q(\mathbf{x}|\hat{\mathbf{x}}^k) &= \mathbb{E}_{p(\mathbf{b}|\hat{\mathbf{x}}^k, \mathbf{y})} [\ln\{p(\mathbf{y}, \mathbf{b}|\mathbf{x})\} + \ln\{p(\mathbf{x})\}] \\
 &= \int \ln\{p(\mathbf{y}, \mathbf{b}|\mathbf{x})\} p(\mathbf{b}|\hat{\mathbf{x}}^k, \mathbf{y}) d\mathbf{b} + \ln\{p(\mathbf{x})\} \\
 &= \int \ln\{p(\mathbf{y}, \mathbf{b}, \mathbf{x})\} p(\mathbf{b}|\hat{\mathbf{x}}^k, \mathbf{y}) d\mathbf{b} \\
 &= \ln\{p(\mathbf{y}|\mathbf{x})\} + \int \ln\{p(\mathbf{x}|\mathbf{b})\} p(\mathbf{b}|\hat{\mathbf{x}}^k, \mathbf{y}) d\mathbf{b} + C
 \end{aligned} \tag{1.40}$$

où $\hat{\mathbf{x}}^k$ est l'estimation obtenue à l'itération k et C est une constante qui ne dépende pas de \mathbf{x} ,

- une étape de maximisation (M), où on maximise l'espérance trouvée à l'étape (E),

$$\begin{aligned}
 \hat{\mathbf{x}}^{k+1} &= \arg \max_{\mathbf{x}} Q(\mathbf{x}|\hat{\mathbf{x}}^k) \\
 &= \arg \min_{\mathbf{x}} \left\{ -\ln p(\mathbf{y}|\mathbf{x}) - \int \ln\{p(\mathbf{x}|\mathbf{b})\} p(\mathbf{b}|\hat{\mathbf{x}}^k, \mathbf{y}) d\mathbf{b} \right\} \\
 &= \arg \min_{\mathbf{x}} \left\{ \frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax}) \right. \\
 &\quad \left. + \frac{1}{2} \int (\mathbf{x} - \mathbf{m}_{\mathbf{x}})^T \mathbf{P}_{\mathbf{x}}(\mathbf{x} - \mathbf{m}_{\mathbf{x}}) p(\mathbf{b}|\hat{\mathbf{x}}^k, \mathbf{y}) d\mathbf{b} \right\}
 \end{aligned} \tag{1.41}$$

qui est obtenu en utilisant le fait que $p(\mathbf{y}|\mathbf{x})$ est une distribution gaussienne multi-dimensionnelle et dans les modèles *a priori* de mélange de gaussiennes, $p(\mathbf{x}|\mathbf{b})$ est aussi une distribution gaussienne.

Dans le cas d'un GSM, d'après (1.36), nous pouvons facilement obtenir

$$\begin{aligned}
 (\mathbf{x} - \mathbf{m}_{\mathbf{x}})^T \mathbf{P}_{\mathbf{x}}(\mathbf{x} - \mathbf{m}_{\mathbf{x}}) &= (\mathbf{V}^T \mathbf{x} - \boldsymbol{\epsilon})^T \text{Diag}(\mathbf{b})(\mathbf{V}^T \mathbf{x} - \boldsymbol{\epsilon}) \\
 &= \sum_j b_j (\mathbf{v}_j^T \mathbf{x} - \epsilon_j)^2.
 \end{aligned} \tag{1.42}$$

Dans ce cas,

$$\hat{\mathbf{x}}^{k+1} = \arg \min_{\mathbf{x}} \left\{ \frac{1}{2}(\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1}(\mathbf{y} - \mathbf{Ax}) + \sum_j \frac{1}{2} \langle b_j \rangle (\mathbf{v}_j^T \mathbf{x} - \epsilon_j)^2 \right\} \tag{1.43}$$

où $\langle b_j \rangle = \int b_j p(b_j|\hat{\mathbf{x}}^k, \mathbf{y}) db_j$. Les calculs dans [CI04] montrent que cette espérance est donnée par

$$\langle b_j \rangle = \phi'(u_j^k)/u_j^k \quad \text{avec} \quad u_j^k = \mathbf{v}_j^T \hat{\mathbf{x}}^k - \epsilon_j. \tag{1.44}$$

1.4.2 - Méthode d'approximation stochastique

En comparant ce résultat avec (1.29) et (1.31), nous trouvons que l'utilisation de EM nous permet de retrouver l'algorithme SQ de type GR avec

- l'étape d'espérance cohérente avec l'étape de maximisation par rapport aux variables auxiliaires de l'algorithme SQ,
- l'étape de maximisation cohérente avec l'étape de maximisation par rapport aux variables d'intérêt de l'algorithme SQ.

Suivant des mêmes étapes de calculs, dans le cas d'un GPM, nous pouvons obtenir,

$$\hat{\mathbf{x}}^{k+1} = \arg \min_{\mathbf{x}} \left\{ \frac{1}{2} (\mathbf{y} - \mathbf{Ax})^T \Sigma^{-1} (\mathbf{y} - \mathbf{Ax}) + \frac{1}{\sigma^2} \sum_j \frac{1}{2} (\mathbf{v}_j^T \mathbf{x} - \epsilon_j - \langle b_j \rangle)^2 \right\} \quad (1.45)$$

où $\langle b_j \rangle$ est montré dans [CI04] d'être égale à

$$\langle b_j \rangle = u_j^k - \sigma^2 \phi'(u_j^k) \quad \text{avec} \quad u_j^k = \mathbf{v}_j^T \hat{\mathbf{x}}^k - \epsilon_j. \quad (1.46)$$

En comparant ce résultat avec (1.30) et (1.32), nous trouvons que l'utilisation de EM nous permet de retrouver l'algorithme SQ de type GY.

En conclusion, sous réserve que la loi *a priori* puisse être représentée par mélange de gaussiennes par changement d'échelle (ou par changement de position), les algorithmes SQ de type Geman & Reynolds (respectivement Geman & Yang) sont des cas particuliers de l'algorithme Espérance-Maximisation (EM).

1.4.2 Méthode d'approximation stochastique

Dans les cas où l'estimateur MAP est difficile à calculer, un autre choix possible est de calculer l'estimateur EAP. Néanmoins, le calcul direct de cet estimateur est généralement impossible car la loi *a posteriori* est souvent connue à une constante multiplicative près. Pour résoudre ce problème, il existe deux types de méthodes :

- 1) les méthodes stochastiques de type Monte Carlo par Chaîne de Markov (MCMC),
- 2) les méthodes d'approximation analytique, notamment l'approximation bayésienne variationnelle.

Nous allons introduire brièvement les méthodes MCMC dans cette partie. Les méthodes d'approximation analytique, plus particulièrement l'approximation bayésienne variationnelle qui est la principale préoccupation de cette thèse, vont être présentées dans le chapitre suivant.

L'idée centrale des méthodes MCMC [RC00] est de fournir une approximation numérique de la distribution ciblée, notée par p dans la suivante, à l'aide d'un ensemble d'échantillons. Les caractéristiques statistiques des paramètres d'intérêt, e.g. la moyenne, la variance, peuvent donc être estimées en utilisant des moments empiriques des échantillons. Dans la littérature, de nombreux algorithmes MCMC ont été développés. Dans le cadre bayésien, les deux algorithmes MCMC les plus utilisés sont l'algorithme de Metropolis-Hastings et l'échantillonnage de Gibbs.

- 1) **L'algorithme de Metropolis-Hastings** Les approches Metropolis-Hastings ont l'avantage de ne demander qu'une connaissance limitée de la loi ciblée ce qui permet une grande liberté dans leur mise en œuvre. Elles suivent les étapes suivantes :
 - (a) Choisir une valeur initiale \mathbf{w}^0

- (b) Étant donné \mathbf{w}^k , générer \mathbf{w}^* à l'aide d'une *loi instrumentale* $g(\cdot|\mathbf{w}^k)$
(c) On accepte ou rejette cette valeur de \mathbf{w}^* à l'aide d'une procédure d'acceptation-rejet :

$$\mathbf{w}^{k+1} = \begin{cases} \mathbf{w}^* & \text{avec probabilité } \rho(\mathbf{w}^k, \mathbf{w}^*) \\ \mathbf{w}^k & \text{avec probabilité } 1 - \rho(\mathbf{w}^k, \mathbf{w}^*) \end{cases} \quad (1.47)$$

où

$$\rho(\mathbf{w}^k, \mathbf{w}^*) = \min \left(\frac{p(\mathbf{w}^*) g(\mathbf{w}^k|\mathbf{w}^*)}{p(\mathbf{w}^k) g(\mathbf{w}^*|\mathbf{w}^k)}, 1 \right) \quad (1.48)$$

(d) Répéter les étapes (b), (c) pour obtenir les échantillons utilisés pour l'inférence. L'algorithme de Metropolis-Hastings décrit ci-dessus est un algorithme « universel », c'est-à-dire que, théoriquement, il permet de simuler n'importe quelle loi p en utilisant une loi instrumentale quelconque. Néanmoins, pour avoir des algorithmes aussi utilisables en pratique, on implémente généralement l'algorithme de Metropolis-Hastings en utilisant des g spécifiques. Un cas particulier courant de l'algorithme est celui où g dépend de \mathbf{w}^k et est symétrique, i.e. $g(\mathbf{w}^*|\mathbf{w}^k) = g(\mathbf{w}^k|\mathbf{w}^*)$. Dans ce cas, l'algorithme se déplace de \mathbf{w}^k à \mathbf{w}^* et il est connu comme l'algorithme de Metropolis-Hastings à marche aléatoire. Un autre cas aussi connu est celui où g est indépendante de \mathbf{w}^k , i.e., $g(\mathbf{w}^*|\mathbf{w}^k) = g(\mathbf{w}^*)$. Dans ce cas, l'algorithme est connu comme l'algorithme de Metropolis-Hastings indépendant.

Les étapes de l'algorithme de Metropolis-Hastings montrent qu'il ne dépend que du rapport $p(\mathbf{w}^*)/p(\mathbf{w}^k)$ et peut être alors utilisé pour simuler les distributions connues à une constante multiplicative près.

En ce qui concerne la convergence, l'algorithme de Metropolis-Hastings génère une chaîne de Markov dont la loi stationnaire est la loi désirée p (voir [RC00, chapter 4]), c'est-à-dire que la chaîne de Markov converge asymptotiquement vers p . Néanmoins, bien que la chaîne de Markov converge finalement vers la distribution désirée, les échantillons initiaux peuvent suivre une distribution très différente, en particulier si le point de départ se trouve dans une région à faible densité. En conséquence, un temps de chauffe (« burn in ») est généralement nécessaire, où un nombre important d'échantillons initiaux sont jetés.

- 2) **L'échantillonnage de Gibbs** L'échantillonnage de Gibbs peut être utilisé pour générer les échantillons d'une distribution jointe $p(\mathbf{w})$ si les distributions conditionnelles de chaque composante sont simulables, i.e. si $\mathbf{w} = (\mathbf{w}_1, \dots, \mathbf{w}_P)$, les distributions conditionnelles incluent $p(\mathbf{w}_1|\mathbf{w}_2, \dots, \mathbf{w}_P)$, $p(\mathbf{w}_2|\mathbf{w}_1, \mathbf{w}_3, \dots, \mathbf{w}_P)$, \dots , $p(\mathbf{w}_P|\mathbf{w}_1, \dots, \mathbf{w}_{P-1})$. L'algorithme d'échantillonnage de Gibbs associé à cette décomposition suit les étapes suivantes :

- (a) Choisir une valeur initiale \mathbf{w}^0
(b) Étant données \mathbf{w}^k , simuler

$$\begin{aligned} \mathbf{w}_1^{k+1} &\sim p(\mathbf{w}_1|\mathbf{w}_2^k, \dots, \mathbf{w}_P^k) \\ \mathbf{w}_2^{k+1} &\sim p(\mathbf{w}_2|\mathbf{w}_1^{k+1}, \mathbf{w}_3^k, \dots, \mathbf{w}_P^k) \\ &\dots \\ \mathbf{w}_P^{k+1} &\sim p(\mathbf{w}_P|\mathbf{w}_1^{k+1}, \dots, \mathbf{w}_{P-1}^{k+1}) \end{aligned} \quad (1.49)$$

- (c) Répéter jusqu'à obtenir les échantillons utilisés pour l'inférence des paramètres (un nombre d'échantillons initiaux sont aussi jetés).

1.4.2 - Méthode d'approximation stochastique

En fait, l'algorithme de Gibbs peut s'exprimer comme cas particulier d'algorithme de Metropolis-Hastings, ou plus exactement une combinaison d'algorithmes de Metropolis-Hastings. Mais comme mentionné dans [RC00], il a des motivations méthodologiques et historiques fondamentalement différentes.

Les méthodes MCMC permettent de générer les échantillons d'une distribution pas exactement connue (connue à une constante près), qui peuvent être utilisés pour les inférences de paramètres d'intérêt. Néanmoins, les méthodes MCMC sont fondées sur des propriétés asymptotiques de convergence et sont généralement coûteuses en temps de calcul. Elles sont mal adaptées aux problèmes de grande dimension car en dimension importante, la loi instrumentale doit être très proche de la loi ciblée pour éviter un taux de rejet important. En plus, c'est difficile de générer des échantillons pour une loi de matrice de covariance quelconque. Donc, pour contourner ces problèmes, une autre méthode est d'utiliser des approximations analytiques de la loi désirée, qui vont être introduites dans le chapitre suivant.

2

État de l'art sur les méthodes d'approximation analytique

2.1 Introduction

Les méthodes MCMC permettent de générer des approximations à précision arbitraire. Néanmoins, elles nécessitent un temps de calcul long. Pour s'affranchir de ce problème et pour d'autres applications comme le choix du modèle, les approximations analytiques ont été introduites comme des compromis entre la précision de l'approximation et la complexité du calcul. Ce chapitre est consacré à la présentation des méthodes d'approximation analytique, notamment des méthodes d'approximation bayésienne variationnelle, pour approcher la fonction densité de probabilité (loi *a posteriori* compliquée).

Dans la suite, nous utilisons \mathbf{w} pour représenter un vecteur de taille J regroupant toutes les variables à estimer, par exemple, $\mathbf{w} = \mathbf{x}$ pour les problèmes estimant seulement des paramètres d'intérêt, et $\mathbf{w} = (\mathbf{x}, \boldsymbol{\theta})$ lorsqu'on estime conjointement les hyperparamètres. La loi *a posteriori* à approcher est représentée par $p(\cdot|\mathbf{y})$ et son approximation analytique est donnée par q . Il existe plusieurs méthodes d'approximation analytique permettant de déterminer q . Dans ce chapitre, nous rappelons trois de ces méthodes, deux méthodes couramment utilisées et une méthode récemment proposée :

- la méthode d'approximation de Laplace,
- la méthode d'approximation bayésienne variationnelle classique,
- la méthode d'approximation bayésienne variationnelle de type gradient.

2.2 Approximation de Laplace

La méthode d'approximation de Laplace [KR95] est bien connue dans le contexte de construction des critères pour la sélection de modèles. Par exemple, le critère Baye-

CHAPITRE 2. ÉTAT DE L'ART SUR LES MÉTHODES D'APPROXIMATION ANALYTIQUE

sian Information Criterion (BIC) est issu de l'approximation de Laplace. Cette méthode d'approximation s'appuie sur une approximation locale de la loi *a posteriori* autour de l'estimation ponctuelle du maximum *a posteriori* (MAP) en utilisant une distribution gaussienne. Supposons que $\hat{\mathbf{w}}$ est l'estimation MAP, cette approximation gaussienne peut être donnée par

$$q^{Lap}(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\hat{\mathbf{w}}, \mathbf{H}^{-1}) \quad (2.1)$$

où la matrice \mathbf{H} est l'opposé de la matrice hessienne des dérivées secondes partielles du logarithme de la loi jointe $p(\mathbf{w}, \mathbf{y})$ par rapport à \mathbf{w} en $\hat{\mathbf{w}}$:

$$\mathbf{H} = - \left[\frac{\partial^2 \ln p(\mathbf{w}, \mathbf{y})}{\partial w_i \partial w_j} \right]_{\mathbf{w}=\hat{\mathbf{w}}} \quad \forall i, j = 1, \dots, N. \quad (2.2)$$

La méthode d'approximation de Laplace donne directement la formule explicite de la loi approchante. Néanmoins elle a présenté plusieurs inconvénients. Premièrement, sa détermination nécessite de déterminer tout d'abord l'estimateur du maximum *a posteriori* et la matrice hessienne \mathbf{H} qui n'est pas facile dans les cas où la loi *a posteriori* est de forme compliquée. Deuxièmement, l'approximation de Laplace est incapable d'aborder des distributions multimodales. Troisièmement, elle fixe la forme de la loi approchante à une loi gaussienne. Néanmoins, la loi gaussienne n'est pas toujours une bonne approximation.

Dans le contexte d'approcher une loi compliquée, un autre type de méthode est la méthode d'approximation bayésienne variationnelle qui vise à déterminer des lois approchantes de forme libre. Grâce à sa liberté de choix de la forme de la loi approchante, l'approximation bayésienne variationnelle a plus de chance de donner des approximations plus proches que celle obtenue par l'approximation de Laplace. Les méthodes d'approximation bayésienne variationnelle sont présentées dans la suite.

2.3 Approximation bayésienne variationnelle classique

La méthode d'approximation bayésienne variationnelle (BV) provient dans la physique statistique dans le domaine de la théorie du champ moyen [Par88]. En physique statistique, des fonctions de probabilité de grande dimension et leur simplification ont été une préoccupation des physiciens depuis longtemps. Les physiciens considèrent souvent un système de plusieurs particules qui interagissent les unes avec les autres. Dans ce cas, la fonction de probabilité associée à ce système est de forme compliquée et sa fonction de partition n'est pas explicite ce qui rend l'étude de ce système difficile. Pour s'affranchir de cette difficulté, l'approximation de champ moyen remplace toutes les interactions à une seule particule par une interaction moyenne, qui simplifie le système de plusieurs particules en interaction à un système de plusieurs particules sans interactions mutuelles. La fonction de probabilité associée à ce système sans interactions est une fonction séparable qui est plus facile à étudier. Une telle fonction séparable optimale peut être choisie en minimisant une mesure de dissemblance entre la probabilité originale compliquée et la fonction de probabilité séparable. L'approximation bayésienne variationnelle est un exemple de telles approximations en prenant la divergence de *Kullback-Leibler* comme la mesure de dissemblance. L'approximation de champ moyen est une généralisation de l'approximation BV en exploitant les différentes fonctions de dissemblance.

La méthode d'approximation BV a été introduite dans le cadre bayésien par MacKay [Mac95]. Depuis cela, elle a été largement utilisée dans diverses applications telles que l'apprentissage des modèles graphiques [JGJS99], restauration d'images [LG04, CGLS08], la séparation de la source [Cho02] et la super-résolution [BMK11]. Nous donnons dans la suite une présentation de cette méthode.

2.3.1 Principe de la méthode d'approximation bayésienne variationnelle

L'idée centrale de la méthode d'approximation bayésienne variationnelle (BV) est d'approcher la loi *a posteriori* ciblée $p(\cdot|\mathbf{y})$ par une loi séparable q^{opt} dont la forme est librement choisie, ce qui est différent de l'approximation de Laplace qui fixe la loi approchante à une loi gaussienne. Cette loi approchante doit être la plus proche possible de la loi ciblée au sens où elle doit minimiser une mesure de dissemblance entre une loi approchante quelconque q et la loi $p(\cdot|\mathbf{y})$, notée dans la suite par $\Delta(q||p(\cdot|\mathbf{y}))$. Cette mesure de dissemblance satisfait

- $\Delta(q||p(\cdot|\mathbf{y})) \geq 0$ et 0 n'est atteint que pour $q = p(\cdot|\mathbf{y})$,
- $\Delta(q||p(\cdot|\mathbf{y}))$ est convexe.

Définissons un espace Ω qui est un espace de densités de probabilité séparables,

$$\Omega = \left\{ q : \text{densité de probabilité et } q(\mathbf{w}) = \prod_{i=1}^P q_i(\mathbf{w}_i) \text{ avec } \mathbf{w} = (\mathbf{w}_1, \dots, \mathbf{w}_P) \right\}, \quad (2.3)$$

où $(\mathbf{w}_i)_{i=1, \dots, P}$ représentent les sous-ensembles disjoints de \mathbf{w} avec P un entier entre 1 et J . Plus la valeur de P est grande, plus la séparabilité est forte. Le problème de recherche de la loi approchante optimale peut être considéré comme un problème d'optimisation fonctionnelle dans l'espace Ω :

$$q^{opt} = \arg \min_{q \in \Omega} \Delta(q||p(\cdot|\mathbf{y})). \quad (2.4)$$

2.3.2 Choix de la mesure de dissemblance

En ce qui concerne la mesure de dissemblance $\Delta(q||p(\cdot|\mathbf{y}))$, un choix naturel est la divergence de *Kullback-Leibler* (\mathcal{KL}) qui est une mesure de la différence entre deux densités de probabilité. La divergence \mathcal{KL} de q à $p(\cdot|\mathbf{y})$ est définie par

$$\mathcal{KL}[q||p(\cdot|\mathbf{y})] = \int_{\mathbb{R}^J} q(\mathbf{w}) \ln \frac{q(\mathbf{w})}{p(\mathbf{w}|\mathbf{y})} d\mathbf{w}. \quad (2.5)$$

Cette divergence a les propriétés suivantes :

- $\mathcal{KL}[q||p(\cdot|\mathbf{y})] \geq 0$, $\forall q, p(\cdot|\mathbf{y})$ et elle s'annule lorsque les deux distributions sont identiques,
- elle est asymétrique, c'est à dire $\mathcal{KL}[q||p(\cdot|\mathbf{y})] \neq \mathcal{KL}[p(\cdot|\mathbf{y})||q]$,
- elle est convexe [CT12].

En raison de l'asymétrie de la divergence \mathcal{KL} , deux possibilités existent pour la mesure en prenant les deux ordres possibles des arguments de la divergence \mathcal{KL} :

2.3.2 - Choix de la mesure de dissemblance

- La première est la divergence \mathcal{KL} de la loi *a posteriori* à la loi approchante $\mathcal{KL}[p(\cdot|\mathbf{y})||q]$. Cette divergence est dite optimale car
 - D'un point de vue bayésien, la minimisation de cette divergence donne une approximation de risque minimale [Ber79, ŠQ06]. Dans la suite, on utilise donc \mathcal{KL}_{RM} pour représenter cette divergence.
 - En supposant que la loi approchante est séparable : $q(\mathbf{w}) = \prod_{i=1}^P q_i(\mathbf{w}_i)$, la minimisation de \mathcal{KL}_{RM} donne la solution suivante :

$$\begin{aligned}
 q_i^{opt}(\mathbf{w}_i) &= \arg \min_{q_i} \mathcal{KL}[p(\mathbf{w}|\mathbf{y})||q(\mathbf{w})] \\
 &= \arg \min_{q_i} \int p(\mathbf{w}_i, \mathbf{w}_{/i}|\mathbf{y}) \ln \frac{1}{q_i(\mathbf{w}_i)} d\mathbf{w}_i d\mathbf{w}_{/i} \\
 &= \arg \min_{q_i} \mathcal{KL}[p(\mathbf{w}_i|\mathbf{y})||q_i(\mathbf{w}_i)] \\
 &= p(\mathbf{w}_i|\mathbf{y}), \tag{2.6}
 \end{aligned}$$

où $\mathbf{w}_{/i}$ représente le complémentaire de \mathbf{w}_i d'un ensemble de variables \mathbf{w} . À partir de (2.6), nous pouvons voir que l'approximation optimale de la loi *a posteriori* au sens de minimiser \mathcal{KL}_{RM} est un produit des lois marginales. Dans ce cas, l'espérance de la loi approchante est égale à l'espérance de la vraie loi *a posteriori*. Ce résultat est attrayant mais cette loi approchante n'est pas utilisable en pratique car, en général, on ne connaît pas les lois marginales $(p(\mathbf{w}_i|\mathbf{y}))_{i=1,\dots,P}$.

- La deuxième possibilité est la divergence \mathcal{KL} de la loi approchante à la loi *a posteriori* $\mathcal{KL}[q||p(\cdot|\mathbf{y})]$ qui est notée par \mathcal{KL}_{BV} dans la suite. Les méthodes d'approximation bayésienne variationnelle utilisent cette divergence comme la mesure de la dissemblance car elle permet d'obtenir des solutions analytiques plus faciles à obtenir, comme on le verra dans la partie 2.3.3.

En pratique, il est intéressant de pouvoir estimer la convergence de l'algorithme. Néanmoins, la divergence \mathcal{KL}_{BV} n'est pas numériquement calculable car elle dépend de la loi *a posteriori* qui n'est pas connue dans notre cas. En effet, généralement, on connaît analytiquement la loi jointe $p(\mathbf{w}, \mathbf{y})$ et pas la loi *a posteriori* car on ne connaît pas explicitement la fonction de partition. Mais on peut développer l'expression de la divergence comme suite :

$$\begin{aligned}
 \mathcal{KL}[q||p(\cdot|\mathbf{y})] &= \int_{\mathbb{R}^J} q(\mathbf{w}) \ln \frac{q(\mathbf{w})}{p(\mathbf{w}|\mathbf{y})} d\mathbf{w} \\
 &= \int_{\mathbb{R}^J} q(\mathbf{w}) \ln \frac{q(\mathbf{w})p(\mathbf{y})}{p(\mathbf{w}, \mathbf{y})} d\mathbf{w} \\
 &= \ln p(\mathbf{y}) - \int_{\mathbb{R}^J} q(\mathbf{w}) \ln \frac{p(\mathbf{w}, \mathbf{y})}{q(\mathbf{w})} d\mathbf{w} \\
 &= \ln p(\mathbf{y}) - \mathcal{F}(q) \tag{2.7}
 \end{aligned}$$

Nous rappelons que $p(\mathbf{y})$ est l'évidence du modèle et

$$\mathcal{F}(q) = \int_{\mathbb{R}^J} q(\mathbf{w}) \ln \frac{p(\mathbf{w}, \mathbf{y})}{q(\mathbf{w})} d\mathbf{w} \tag{2.8}$$

est appelée l'énergie libre négative. Ici, la log-évidence $\ln p(\mathbf{y})$ peut être vue comme une constante par rapport à q . En conséquence, la minimisation de la divergence \mathcal{KL}_{BV} est

équivalente à la maximisation de l'énergie libre négative $\mathcal{F}(q)$. Nous voyons que cette énergie dépend de la loi jointe $p(\mathbf{y}, \mathbf{w})$ qui peut être aisément obtenue par le produit de la vraisemblance et de la loi *a priori*. Par conséquent, en pratique, nous utilisons souvent l'énergie libre négative $\mathcal{F}(q)$ comme une alternative de la divergence \mathcal{KL}_{BV} .

Par ailleurs, d'après (2.7), on peut faire une autre remarque. Comme la divergence \mathcal{KL} est positive, on a $\ln p(\mathbf{y}) \geq \mathcal{F}(q)$. En conséquence, l'énergie libre négative est une borne inférieure de la log-évidence. Lorsque l'algorithme BV converge, la divergence \mathcal{KL} est quasiment nulle. L'énergie libre négative à convergence peut être donc utilisée comme une approximation de la log-évidence utile par exemple dans les problèmes de sélection de modèle.

2.3.3 L'algorithme d'approximation bayésienne variationnelle classique

En prenant \mathcal{KL}_{BV} comme la mesure de dissemblance, la loi approchante optimale est déterminée en résolvant le problème d'optimisation suivant :

$$q^{opt} = \arg \min_{q \in \Omega} \mathcal{KL}[q||p(\cdot|\mathbf{y})], \quad (2.9)$$

qui est, comme montré précédemment, équivalent au problème d'optimisation donné par

$$q^{opt} = \arg \max_{q \in \Omega} \mathcal{F}(q), \quad (2.10)$$

avec \mathcal{F} l'énergie libre négative définie dans (2.8). De plus, \mathcal{F} est une fonctionnelle concave.

Ce problème d'optimisation admet une solution analytique qui est donnée par $q^{opt}(\mathbf{w}) = \prod_{i=1}^P q_i(\mathbf{w}_i)$ avec

$$q_i(\mathbf{w}_i) = K_i \exp \left(\langle \ln p(\mathbf{y}, \mathbf{w}) \rangle_{\prod_{j \neq i} q_j(\mathbf{w}_j)} \right), \quad \forall i = 1, \dots, P \quad (2.11)$$

où $\langle \cdot \rangle_q = \mathbb{E}_q[\cdot]$ et K_i représente la constante de normalisation. Cette solution analytique a été trouvée de plusieurs manières différentes. Par exemple, Šmídl et Quinn [ŠQ06] donnent une dérivation de cette solution en utilisant seulement des calculs probabilistes simples et les propriétés de la divergence \mathcal{KL} indiquées ci-dessus, Miskin a obtenu la même solution dans [Mis00] en annulant la différentielle de la divergence \mathcal{KL} par rapport à la loi approchante. Les approches d'approximation bayésienne variationnelle classiques sont basées sur cette solution analytique donnée par (2.11). Néanmoins, nous pouvons voir de (2.11) que chaque composante séparable q_i dépend de toutes les autres composantes q_j avec j différent de i . Par conséquent, on ne peut pas obtenir une expression explicite pour q^{opt} sauf dans des cas extrêmement simples. En pratique, ce problème est généralement abordé en utilisant un algorithme alterné, qui met à jour une composante séparable en fixant les autres à chaque étape, pour approcher la solution optimale de façon itérative. Finalement, on peut résumer l'algorithme d'approximation bayésienne variationnelle classique par l'Algorithme 1.

Dans l'Algorithme 1, les composantes de q sont mises à jour dans l'ordre de q_1 à q_P , mais en pratique, il n'y a pas de contrainte sur l'ordre des mises à jour et on peut choisir un ordre qui convient le mieux au problème abordé.

2.3.4 - Lien entre l'algorithme d'approximation BV et l'algorithme EM

Algorithm 1 Algorithme d'approximation Bayésienne Variationnelle Classique (BV-Class)

1. Initialiser ($q^0 \in \Omega$)
2. Mettre à jour $q_1^{k+1}(\mathbf{w}_1)$
3. Mettre à jour $q_2^{k+1}(\mathbf{w}_2)$
4. ...
5. Mettre à jour $q_P^{k+1}(\mathbf{w}_P)$ avec

$$q_i^{k+1}(\mathbf{w}_i) = K_i \exp \left(\langle \ln p(\mathbf{y}, \mathbf{w}) \rangle_{(\prod_{j=1}^{i-1} q_j^{k+1}(\mathbf{w}_j)) (\prod_{j=i+1}^P q_j^k(\mathbf{w}_j))} \right), \forall i = 1, \dots, P. \quad (2.12)$$

6. Retourner à l'étape 2 jusqu'à convergence
-

L'importance des lois *a priori* conjuguées Théoriquement, l'algorithme 1 peut être utilisé pour approcher n'importe quelle loi *a posteriori*. Néanmoins, en pratique, le choix des lois *a priori* conjuguées joue un rôle très important dans le développement des algorithmes bayésiens variationnels implémentables numériquement car les lois *a priori* conjuguées assurent que la loi *a posteriori* appartient à la même famille paramétrique que la loi *a priori*. En conséquence, les lois approchantes obtenues par (2.11) se trouvent dans la même famille et l'optimisation des lois est ramenée à mettre à jour des paramètres, ce qui est beaucoup plus simple, et surtout beaucoup plus simple à implémenter sur un ordinateur (pas besoin de calcul formel).

2.3.4 Lien entre l'algorithme d'approximation BV et l'algorithme EM

Une propriété très intéressante de cette approche d'approximation BV est que l'on peut considérer que c'est une généralisation de l'algorithme Espérance-Maximisation (EM). C'est ce que nous allons montrer dans la suite de cette partie (Des preuves équivalentes se trouvent dans [BG02] et [ŠQ06]).

L'algorithme EM est typiquement utilisé lorsque le modèle implique des variables latentes non-observables. Nous supposons donc que $\mathbf{w} = (\mathbf{x}, \mathbf{z})$ où \mathbf{x} représente le vecteur des paramètres d'intérêt et \mathbf{z} est le vecteur des variables latentes. Nous nous intéressons ici à un estimateur du maximum *a posteriori* marginal pour le paramètre d'intérêt \mathbf{x} qui est mathématiquement exprimé par

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) \quad (2.13)$$

$$\text{où } p(\mathbf{x}|\mathbf{y}) = \int p(\mathbf{x}, \mathbf{z}|\mathbf{y}) d\mathbf{z} \propto \int p(\mathbf{x}, \mathbf{z}, \mathbf{y}) d\mathbf{z} \quad (2.14)$$

En général, on peut facilement obtenir l'expression explicite de la loi jointe $p(\mathbf{x}, \mathbf{z}, \mathbf{y})$ mais il est difficile voire impossible de calculer l'intégrale de $p(\mathbf{x}, \mathbf{z}, \mathbf{y})$ par rapport à \mathbf{z} qui est donnée par (2.14). Par conséquent, l'estimateur du maximum *a posteriori* marginal (voir (2.13)) ne peut être obtenu directement. Dans ce cas, l'algorithme EM est un algorithme couramment utilisé pour s'affranchir de cette difficulté.

L'utilisation de l'algorithme EM

L'algorithme EM pour calculer l'estimateur du MAP marginal consiste en deux étapes : une étape d'évaluation de l'espérance conditionnelle et une étape de maximisation, où l'on trouve une nouvelle estimation en maximisant l'espérance obtenue, voir partie 1.4.1 du chapitre 1 pour plus de détails sur l'algorithme EM. Ces deux étapes sont données par

- Étape d'évaluation de l'*Espérance*

$$Q(\mathbf{x}|\hat{\mathbf{x}}^k) = \mathbb{E}_{p(\mathbf{z}|\hat{\mathbf{x}}^k, \mathbf{y})} [\ln p(\mathbf{y}, \mathbf{z}|\mathbf{x})] + \ln p(\mathbf{x}), \quad (2.15)$$

- Étape de *Maximisation*

$$\hat{\mathbf{x}}^{k+1} = \arg \max_{\mathbf{x}} Q(\mathbf{x}|\hat{\mathbf{x}}^k), \quad (2.16)$$

qui sont itérées jusqu'à la convergence.

La dérivation de EM via l'approximation bayésienne variationnelle

Le principe de l'approximation BV consiste à chercher une loi séparable pour approcher la loi *a posteriori* jointe $p(\mathbf{x}, \mathbf{z}|\mathbf{y})$. La loi approchante optimale est obtenue en résolvant le problème d'optimisation défini par (2.9) ou (2.10). Dans ce cas, les formes des composantes de la loi approchante peuvent être librement choisies. Néanmoins, on peut aussi fixer en avance la forme de la loi approchante. Cette contrainte supplémentaire sur les formes des lois nous permet d'obtenir des solutions sous-optimales du problème (2.9) ou (2.10), ce qui conduit à l'approche d'approximation BV restreinte qui a été présentée dans [ŠQ06, chapter 3]. L'approche d'approximation BV restreinte est donnée par

Corollaire 1. (*Approche d'approximation Bayésienne Variationnelle Restreinte*). Soit $p(\mathbf{w}|\mathbf{y})$ la loi *a posteriori* du paramètre $\mathbf{w} = (\mathbf{w}_1, \mathbf{w}_2)$ où nous considérons une division en deux de \mathbf{w} . Soit $q(\mathbf{w}) = q_1(\mathbf{w}_1)\bar{q}_2(\mathbf{w}_2|\hat{\beta})$, dont q_1 est de forme libre tandis que \bar{q}_2 est une distribution de **forme fixée** paramétrisée par $\hat{\beta}$, l'approximation séparable de $p(\mathbf{w}|\mathbf{y})$. Alors le minimum de la divergence \mathcal{KL}_{BV} est atteint pour

$$q_1(\mathbf{w}_1) \propto \exp \left(\langle \ln p(\mathbf{y}, \mathbf{w}) \rangle_{\bar{q}_2(\mathbf{w}_2|\hat{\beta})} \right) \quad (2.17)$$

$$\hat{\beta} = \arg \min_{\beta} \mathcal{KL} [q_1(\mathbf{w}_1)\bar{q}_2(\mathbf{w}_2|\beta) || p(\mathbf{w}|\mathbf{y})] \quad (2.18)$$

Ici, nous avons $\mathbf{w} = (\mathbf{x}, \mathbf{z})$ et nous supposons que la loi approchante de la loi *a posteriori* $p(\mathbf{x}, \mathbf{z}|\mathbf{y})$ est notée par $q(\mathbf{x}, \mathbf{z})$ et qu'elle satisfait

$$q(\mathbf{x}, \mathbf{z}) = q_1(\mathbf{z})\bar{q}_2(\mathbf{x}), \text{ avec } \bar{q}_2(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{m}) \quad (2.19)$$

où δ représente la distribution de Dirac.

Selon le Corollaire 1, la minimisation de la divergence \mathcal{KL}_{BV} par rapport à q_1 et \mathbf{m}

2.3.5 - Choix de séparation

nous donne :

$$q_1^{k+1}(\mathbf{z}) = K_1 \exp \left(\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{\delta(\mathbf{x}-\mathbf{m}^k)} \right) = p(\mathbf{z}|\mathbf{m}^k, \mathbf{y}), \quad (2.20)$$

$$\begin{aligned} \mathbf{m}^{k+1} &= \arg \min_{\mathbf{m}} \mathcal{KL} [p(\mathbf{z}|\mathbf{m}^k, \mathbf{y})\delta(\mathbf{x}-\mathbf{m})||p(\mathbf{x}, \mathbf{z}|\mathbf{y})] \\ &= \arg \max_{\mathbf{m}} \left\{ \langle \ln p(\mathbf{m}, \mathbf{z}, \mathbf{y}) \rangle_{p(\mathbf{z}|\mathbf{m}^k, \mathbf{y})} \right\} \\ &= \arg \max_{\mathbf{m}} \left\{ \mathbb{E}_{p(\mathbf{z}|\mathbf{m}^k, \mathbf{y})} [\ln p(\mathbf{y}, \mathbf{z}|\mathbf{m})] + \ln p(\mathbf{m}) \right\}, \end{aligned} \quad (2.21)$$

ce qui nous permet de retrouver la même formule que (2.16).

2.3.5 Choix de séparation

L'approximation BV impose la séparabilité entre les sous-ensembles de paramètres $(\mathbf{w}_i)_{i=1, \dots, P}$. On néglige alors les liens statistiques entre ces paramètres. En général, plus le degré de séparabilité utilisé est grand, plus la loi approchante est éloignée de la loi cible.

En pratique, il n'y a pas de règles pour le choix de la séparabilité et on peut choisir la séparabilité selon ses envies. Nous divisons dans la suite les hypothèses de séparabilité en deux catégories :

- La séparation partielle où $P < J$. Dans ce cas, on choisit de garder les corrélations entre une partie des paramètres. La séparation partielle a été largement utilisée dans la littérature, par exemple, Babacan *et al.* [BMK11] supposent uniquement la séparabilité entre les paramètres d'intérêt et les hyperparamètres. Dans le modèle du Chantas *et al.* [CGLS08] où des variables cachées apparaissent, une séparabilité entre les paramètres d'intérêt, les variables cachées et les hyperparamètres a donc été considérée. Dans ces cas, les corrélations internes des paramètres d'intérêt, des hyperparamètres et des variables cachées sont préservées. Néanmoins, un inconvénient lié à la séparation partielle est qu'on obtient alors des matrices pour chaque groupe de variables (par exemple, la matrice de covariance dans un cas gaussien), qui ont besoin d'être stockées et inversées, comme on le verra dans la suite par un exemple gaussien. Dans les problèmes de grande dimension, ce type de calculs est très coûteux.
- La séparation totale où $P = J$. Sous cette hypothèse, la séparabilité est imposée entre toutes les composantes de \mathbf{w} . Cette hypothèse est forte et fait perdre les corrélations entre les composantes mais cela permet de faciliter le calcul des lois approchantes.

Nous prenons ici un exemple simple – un problème gaussien (voir partie 1.3) pour illustrer la différence entre les deux choix de séparabilité. Considérons le modèle linéaire défini par (1.2) dans le début de cette thèse :

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \quad (2.22)$$

Comme donné dans la partie 1.3, nous considérons un bruit blanc gaussien qui nous permet d'obtenir une vraisemblance de forme (1.8), et nous introduisons une loi *a priori* conjuguée – une loi gaussienne (1.19) sur \mathbf{x} . En ce qui concerne les hyperparamètres, nous faisons une reparamétrisation en introduisant les paramètres de précision $\gamma_n = 1/\sigma_n^2$ et

$\gamma_p = 1/\sigma_p^2$. Nous introduisons ensuite des lois *a priori* conjuguées pour ces paramètres de précision γ_n et γ_p – des lois Gamma :

$$p(\gamma_n) = \mathcal{G}(\gamma_n | \tilde{a}_n, \tilde{b}_n) \quad (2.23)$$

$$p(\gamma_p) = \mathcal{G}(\gamma_p | \tilde{a}_p, \tilde{b}_p) \quad (2.24)$$

où pour $a > 0$ et $b > 0$

$$\mathcal{G}(z | a, b) = \frac{b^a}{\Gamma(a)} z^{a-1} \exp(-bz) \quad (2.25)$$

En ce qui concerne les hyperparamètres, en général, nous ne possédons pas d'information *a priori*. Nous préférons donc prendre des lois *a priori* non-informatives, c'est à dire les lois avec un impact minimal sur la loi *a posteriori*. Ce type d'*a priori* peut être obtenu en donnant zéro comme paramètres des distributions Gamma : $\tilde{a}_n = 0$, $\tilde{b}_n = 0$ et $\tilde{a}_p = 0$, $\tilde{b}_p = 0$, ce qui conduit à l'*a priori* impropre de Jeffreys.

En conséquence, nous pouvons obtenir une loi *a posteriori* suivante :

$$\begin{aligned} p(\mathbf{x}, \gamma_n, \gamma_p | \mathbf{y}) &\propto p(\mathbf{y} | \mathbf{x}, \gamma_n) p(\mathbf{x} | \gamma_p) p(\gamma_n) p(\gamma_p) \\ &\propto \gamma_n^{M/2} \exp\left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2}\right] \gamma_p^{N/2} \exp\left[-\frac{\gamma_p \|\mathbf{\Gamma}\mathbf{x}\|^2}{2}\right] \\ &\quad \times \frac{\tilde{b}_n^{\tilde{a}_n}}{\Gamma(\tilde{a}_n)} \gamma_n^{\tilde{a}_n-1} \exp(-\tilde{b}_n \gamma_n) \frac{\tilde{b}_p^{\tilde{a}_p}}{\Gamma(\tilde{a}_p)} \gamma_p^{\tilde{a}_p-1} \exp(-\tilde{b}_p \gamma_p) \end{aligned} \quad (2.26)$$

Nous approchons cette loi *a posteriori* par la méthode d'approximation BV en prenant l'hypothèse de séparabilité partielle et l'hypothèse de séparabilité totale.

Approximation bayésienne variationnelle en prenant l'hypothèse de séparabilité partielle

La première étape pour appliquer l'algorithme d'approximation BV est de choisir la forme de séparation dans la loi approchante. Ici, nous supposons que

$$q(\mathbf{x}, \gamma_n, \gamma_p) = q_{\mathbf{x}}(\mathbf{x}) q_{\gamma_n}(\gamma_n) q_{\gamma_p}(\gamma_p). \quad (2.27)$$

Nous supposons la séparabilité entre γ_n et γ_p car la loi *a posteriori* $p(\mathbf{x}, \gamma_n, \gamma_p | \mathbf{y})$ est séparable entre eux.

Selon l'Algorithme 1, la loi approchante q peut être obtenue en alternant l'optimisation de $q_{\mathbf{x}}$, q_{γ_n} et q_{γ_p} .

L'optimisation de $q_{\mathbf{x}}$ En utilisant (2.11), nous pouvons obtenir

$$\begin{aligned} q_{\mathbf{x}}^{k+1}(\mathbf{x}) &\propto \exp\left(\langle \ln p(\mathbf{y}, \mathbf{x}, \gamma_n, \gamma_p) \rangle_{q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p)}\right), \\ &\propto \exp\left(-\frac{\langle \gamma_n \rangle^k}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 - \frac{\langle \gamma_p \rangle^k}{2} \|\mathbf{\Gamma}\mathbf{x}\|^2\right). \end{aligned} \quad (2.28)$$

2.3.5 - Choix de séparation

où $\langle \gamma_n \rangle^k = \mathbb{E}_{q_{\gamma_n}^k} [\gamma_n]$ et $\langle \gamma_p \rangle^k = \mathbb{E}_{q_{\gamma_p}^k} [\gamma_p]$. Cette distribution peut être identifiée comme une distribution gaussienne dont la matrice de covariance Σ^{k+1} et la moyenne \mathbf{m}^{k+1} sont données respectivement par

$$(\Sigma^{k+1})^{-1} = \langle \gamma_n \rangle^k \mathbf{A}^T \mathbf{A} + \langle \gamma_p \rangle^k \mathbf{\Gamma}^T \mathbf{\Gamma}. \quad (2.29)$$

$$\mathbf{m}^{k+1} = \Sigma^{k+1} [\langle \gamma_n \rangle^k \mathbf{A}^T \mathbf{y}], \quad (2.30)$$

Nous pouvons voir ici que l'obtention de la moyenne et de la matrice de covariance de la loi approchante nécessite d'inverser la matrice donnée par (2.29). L'inversion des matrices est une opération coûteuse voire impossible lorsque la taille des problèmes devient très importante. Pour contourner des inversions de matrices dans le calcul de \mathbf{m}^{k+1} , une approche proposée dans [BMK11] est d'utiliser une méthode itérative, la méthode du gradient conjugué, pour approcher \mathbf{m}^{k+1} de façon itérative, c'est à dire résoudre l'équation linéaire ci-dessous :

$$(\Sigma^{k+1})^{-1} \mathbf{m}^{k+1} = \langle \gamma_n \rangle^k \mathbf{A}^T \mathbf{y}. \quad (2.31)$$

Néanmoins, le problème reste toujours dans la détermination de la matrice de covariance. Dans [BMK11], la matrice de covariance est approchée par une matrice diagonale dont les composantes sont égales aux inverses des composantes diagonales de la matrice $(\Sigma^{k+1})^{-1}$. Notons que cette approximation peut conduire à une forte erreur.

L'optimisation de q_{γ_n} La loi approchante q_{γ_n} est aussi déterminée en utilisant (2.11).

$$q_{\gamma_n}^{k+1}(\gamma_n) \propto \gamma_n^{M/2 + \tilde{a}_n - 1} \exp \left[-\gamma_n \left(\tilde{b}_n + \frac{1}{2} \mathbb{E}_{q_{\mathbf{x}}^{k+1}} (\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2) \right) \right], \quad (2.32)$$

qui est une distribution Gamma dont les paramètres sont déterminés par

$$a_n^{k+1} = M/2 + \tilde{a}_n \quad (2.33)$$

$$\begin{aligned} b_n^{k+1} &= \tilde{b}_n + \frac{1}{2} \mathbb{E}_{q_{\mathbf{x}}^{k+1}} (\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2) \\ &= \tilde{b}_n + \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{m}^{k+1}\|^2 + \frac{1}{2} \text{trace} (\Sigma^{k+1} \mathbf{A}^T \mathbf{A}) \end{aligned} \quad (2.34)$$

où on peut voir que l'expression explicite de Σ^{k+1} est nécessaire.

L'optimisation de q_{γ_p} Une autre fois, en utilisant (2.11),

$$q_{\gamma_p}^{k+1}(\gamma_p) \propto \gamma_p^{N/2 + \tilde{a}_p - 1} \exp \left[-\gamma_p \left(\tilde{b}_p + \frac{1}{2} \mathbb{E}_{q_{\mathbf{x}}^{k+1}} (\|\mathbf{\Gamma}\mathbf{x}\|^2) \right) \right] \quad (2.35)$$

Cette distribution est identifiée à une loi Gamma avec des paramètres calculés par

$$a_p^{k+1} = N/2 + \tilde{a}_p \quad (2.36)$$

$$\begin{aligned} b_p^{k+1} &= \tilde{b}_p + \frac{1}{2} \mathbb{E}_{q_{\mathbf{x}}^{k+1}} (\|\mathbf{\Gamma}\mathbf{x}\|^2) \\ &= \tilde{b}_p + \frac{1}{2} \|\mathbf{\Gamma}\mathbf{m}^{k+1}\|^2 + \frac{1}{2} \text{trace} (\Sigma^{k+1} \mathbf{\Gamma}^T \mathbf{\Gamma}) \end{aligned} \quad (2.37)$$

qui dépend aussi de la matrice de covariance Σ^{k+1} .

Approximation bayésienne variationnelle en prenant l'hypothèse de séparabilité totale

Par rapport à la partie précédente, nous supposons ici que la loi approchante est totalement séparable, c'est à dire

$$q(\mathbf{x}, \gamma_n, \gamma_p) = \prod_{i=1}^N q_i(x_i) q_{\gamma_n}(\gamma_n) q_{\gamma_p}(\gamma_p). \quad (2.38)$$

Dans ce cas, la loi approchante q est obtenue en alternant

- l'optimisation de q_1
- l'optimisation de q_2
- ...
- l'optimisation de q_N
- l'optimisation de q_{γ_n}
- l'optimisation de q_{γ_p} .

L'optimisation des lois approchantes des hyperparamètres q_{γ_n} et q_{γ_p} est identique à la précédente. Nous donnons donc seulement les calculs pour l'optimisation de $(q_i)_{i=1, \dots, N}$.

L'optimisation de $(q_i)_{i=1, \dots, N}$ Les calculs se font en utilisant (2.11).

$$\begin{aligned} q_i^{k+1}(x_i) &\propto \exp \left(\langle \ln p(\mathbf{y}, \mathbf{x}, \gamma_n, \gamma_p) \rangle_{\prod_{j < i} q_j^{k+1}(x_j) \prod_{j > i} q_j^k(x_j) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p)} \right), \\ &\propto \exp \left[- \frac{\langle \gamma_n \rangle^k}{2} (x_i^2 \text{diag}(\mathbf{A}^T \mathbf{A})_i - 2x_i (\mathbf{A}^T \mathbf{y})_i + 2x_i (\mathbf{A}^T \mathbf{A} \tilde{\mathbf{m}})_i - 2x_i \text{diag}(\mathbf{A}^T \mathbf{A})_i (\tilde{\mathbf{m}})_i) \right. \\ &\quad \left. - \frac{\langle \gamma_p \rangle^k}{2} (x_i^2 \text{diag}(\mathbf{\Gamma}^T \mathbf{\Gamma})_i + 2x_i (\mathbf{\Gamma}^T \mathbf{\Gamma} \tilde{\mathbf{m}})_i - 2x_i \text{diag}(\mathbf{\Gamma}^T \mathbf{\Gamma})_i (\tilde{\mathbf{m}})_i) \right]. \end{aligned} \quad (2.39)$$

où $\text{diag}(\mathbf{M})$ représente un vecteur dont les éléments sont les éléments diagonaux de la matrice \mathbf{M} et $\tilde{\mathbf{m}} = (m_1^{k+1}, \dots, m_{i-1}^{k+1}, m_i^k, \dots, m_N^k)^T$ est un vecteur dépendant de l'indice i .

Cette distribution est encore une distribution gaussienne dont la moyenne et la variance sont données par

$$\begin{aligned} m_i^{k+1} &= (\sigma_i^2)^{k+1} \left[\langle \gamma_n \rangle^k ((\mathbf{A}^T \mathbf{y})_i - (\mathbf{A}^T \mathbf{A} \tilde{\mathbf{m}})_i + \text{diag}(\mathbf{A}^T \mathbf{A})_i (\tilde{\mathbf{m}})_i) \right. \\ &\quad \left. - \langle \gamma_p \rangle^k ((\mathbf{\Gamma}^T \mathbf{\Gamma} \tilde{\mathbf{m}})_i - \text{diag}(\mathbf{\Gamma}^T \mathbf{\Gamma})_i (\tilde{\mathbf{m}})_i) \right], \end{aligned} \quad (2.40)$$

$$(\sigma_i^2)^{k+1} = [\langle \gamma_n \rangle^k \text{diag}(\mathbf{A}^T \mathbf{A})_i + \langle \gamma_p \rangle^k \text{diag}(\mathbf{\Gamma}^T \mathbf{\Gamma})_i]^{-1}. \quad (2.41)$$

Nous pouvons voir que ici, on n'a plus besoin d'inverser des matrices. Néanmoins, cette séparation totale conduit à un algorithme alterné – l'algorithme de descente par coordonnées dans un espace de Hilbert, qui n'est pas suffisamment efficace pour des problèmes de grande dimension.

En résumé, même si les lois *a priori* conjuguées sont utilisées, un verrou à l'utilisation de l'algorithme d'approximation BV classique pour résoudre des problèmes de grande dimension c'est soit la matrice trop coûteuse voire impossible à inverser (lorsque l'hypothèse de séparabilité partielle est utilisée), soit le temps de calculs qui est encore trop

2.3.5 - Choix de séparation

long (lorsque la séparabilité totale est utilisée). Par conséquent, le développement des algorithmes d'approximation BV accélérés est nécessaire. C'est ce qui constitue l'objet principal de cette thèse.

2.4 Approximation bayésienne variationnelle de type gradient

Comme indiqué dans la partie 2.3, les approches d'approximation BV classiques souffrent d'une vitesse de convergence faible et s'avèrent peu pratiques pour traiter les données de grande taille. Pour obtenir les approches BV plus rapides, une méthode différente a été récemment proposée dans [FR14]. Cette méthode s'appuie sur une adaptation de la méthode du gradient dans un espace de Hilbert vers un espace de densités de probabilité impliqué dans le cadre bayésien variationnel.

Comme détaillé dans la partie 2.3, les approches d'approximation BV classiques consistent en deux étapes : premièrement obtenir une solution analytique mais non-explicite de notre problème d'optimisation fonctionnelle (2.9) ou (2.10) en utilisant (2.11), ensuite approcher cette solution non-explicite à l'aide des approches itératives. Au lieu de suivre ces deux étapes, Fraysse *et al.* ont proposé de chercher directement la solution approchée de notre problème d'optimisation fonctionnelle de façon itérative grâce à un algorithme de type gradient. Cette méthode possède des équations de mise à jour explicites pour toutes les composantes séparables qui permettent une mise à jour en parallèle, qui amène une accélération significative par rapport aux approches d'approximation BV classiques.

L'approximation BV a pour but de trouver une distribution séparable optimale q^{opt} pour approcher la loi *a posteriori* $p(\cdot|\mathbf{y})$ en minimisant la divergence \mathcal{KL}_{BV} entre eux, qui est équivalent à maximiser l'énergie libre négative $\mathcal{F}(q)$ (voir la partie 2.3.2 ou [Cho02]) rappelée ici comme

$$\begin{aligned}\mathcal{F}(q) &= \int_{\mathbb{R}^J} q(\mathbf{w}) \ln \frac{p(\mathbf{y}, \mathbf{w})}{q(\mathbf{w})} d\mathbf{w} \\ &= \int_{\mathbb{R}^J} q(\mathbf{w}) \ln p(\mathbf{y}, \mathbf{w}) d\mathbf{w} - \int_{\mathbb{R}^J} q(\mathbf{w}) \ln q(\mathbf{w}) d\mathbf{w}\end{aligned}\quad (2.42)$$

$$= \langle \ln p(\mathbf{y}, \cdot) \rangle_{q(\mathbf{w})} + \mathcal{H}(q)\quad (2.43)$$

où $\mathcal{H}(q) = - \int_{\mathbb{R}^J} q(\mathbf{w}) \ln q(\mathbf{w}) d\mathbf{w}$ est l'entropie de q .

Nous rappelons que Ω est un espace des densités de probabilité séparables,

$$\Omega = \left\{ q : \text{densité de probabilité et } q(\mathbf{w}) = \prod_{i=1}^P q_i(\mathbf{w}_i) \text{ avec } \mathbf{w} = (\mathbf{w}_1, \dots, \mathbf{w}_P) \right\}.\quad (2.44)$$

Nous considérons dans cette partie le problème bayésien variationnel donné par (2.10) et rappelé comme :

$$q^{opt} = \arg \max_{q \in \Omega} \mathcal{F}(q),\quad (2.45)$$

qui est un problème d'optimisation fonctionnelle par rapport à $q = \prod_{i=1}^P q_i$. Comme montré dans [FR14], il existe un problème équivalent à (2.45) dans un espace de mesures de probabilité séparables $\mathcal{A} = \bigotimes_{i=1}^P \mathcal{A}_i$, le produit cartésien des \mathcal{A}_i , qui est défini par

$$\mathcal{A}_i = \{\mu_i : \text{mesure de probabilité} \\ \text{et } \mu_i(d\mathbf{w}_i) = q_i(\mathbf{w}_i)d\mathbf{w}_i \text{ avec } q_i \text{ une densité de probabilité}\}.$$

L'espace \mathcal{A} peut être considéré comme un sous-ensemble de l'espace des mesures de Radon signés, \mathcal{M} , doté de la norme de la variation totale, qui est un espace de Banach. Par conséquent, l'optimisation de (2.45) est équivalente à la résolution du problème d'optimisation suivant :

$$\mu^{opt} = \arg \max_{\mu \in \mathcal{A}} F(\mu), \quad (2.46)$$

où la fonctionnelle $F : \mathcal{M} \rightarrow \mathbb{R}$ satisfait que $\forall \mu \in \mathcal{A}$ avec une densité q , $F(\mu) = \mathcal{F}(q)$. Ce problème d'optimisation peut être considéré comme un problème d'optimisation sous-contraintes dans un espace de Banach de dimension infinie \mathcal{M} .

Comme \mathcal{M} est un espace de Banach, on peut calculer la différentielle de Gâteaux de F qui s'écrit comme

$$\forall \nu \in \mathcal{M}, \quad \partial F_\mu(\nu) = \lim_{t \rightarrow 0} \frac{F(\mu + t\nu) - F(\mu)}{t}. \quad (2.47)$$

Dans certains cas, comme dans ce qui suit, on peut trouver une fonction continue et bornée supérieurement $\partial f : \mathcal{M} \times \mathbb{R}^J \rightarrow \mathbb{R}$ de sorte que la différentielle de Gâteaux est donnée par

$$\forall \nu \in \mathcal{M}, \quad \partial F_\mu(\nu) = \int_{\mathbb{R}^J} \partial f(\mu, \mathbf{w}) d\nu(\mathbf{w}). \quad (2.48)$$

Dans le cas où $\mu \in \mathcal{A}$ et q est la densité de μ , nous pouvons définir une fonctionnelle df par $\forall \mathbf{w} \in \mathbb{R}^J$, $df(q, \mathbf{w}) = \partial f(\mu, \mathbf{w})$ où $df : \Omega \times \mathbb{R}^J \rightarrow \mathbb{R}$. Par ailleurs, comme nous considérons q séparable, nous pouvons obtenir $df(q, \mathbf{w}) = \sum_i df(q_i, \mathbf{w}_i)$, dans laquelle $\forall i = 1, \dots, P$, $df(q_i, \mathbf{w}_i)$ est exprimée par

$$df(q_i, \mathbf{w}_i) = \langle \log p(\mathbf{y}, \mathbf{w}) \rangle_{\prod_{j \neq i} q_j(\mathbf{w}_j)} - \log q_i(\mathbf{w}_i) - 1. \quad (2.49)$$

L'algorithme d'approximation BV proposé dans [FR14] est un algorithme itératif qui fournit une densité à chaque itération et approche la solution de (2.46) progressivement. Il s'appuie sur le théorème de Radon-Nikodym [Rud87].

Nous utilisons ici $k \in \mathbb{N}$, initialement fixé à zéro, comme l'indice des itérations et supposons que μ^k est une mesure de probabilité, avec une densité q^k , c'est à dire

$$d\mu^k(\mathbf{w}) = q^k(\mathbf{w})d\mathbf{w}, \quad (2.50)$$

Comme nous sommes dans l'espace des mesures de probabilité à densité, la prochaine itération devrait donner également une mesure de probabilité absolument continue par rapport à μ^k . Le théorème de Radon-Nikodym assure que cette mesure devrait être écrite comme

$$d\mu^{k+1}(\mathbf{w}) = h^k(\mathbf{w})d\mu^k(\mathbf{w}), \quad (2.51)$$

2.3.5 - Choix de séparation

où $h^k \in L^1(\mu^k)$ est une fonction positive¹. Puisque μ^k est une mesure de probabilité avec sa densité donnée par q^k , nous pouvons également écrire

$$q^{k+1}(\mathbf{w}) = h^k(\mathbf{w})q^k(\mathbf{w}), \quad (2.52)$$

comme l'équation de mise à jour pour la densité approximative. En ce qui concerne la fonction h^k , suivant le schéma itératif donné par la méthode du gradient, Fraysse *et al.* ont proposé une fonction donnée par :

$$h^k(\mathbf{w}) = K^k(\alpha^k) \exp [\alpha^k df(q^k, \mathbf{w})] \quad (2.53)$$

où $\alpha^k > 0$ est le pas de l'algorithme et $df(q^k, \mathbf{w})$ est définie par (2.49) qui est un terme provenant de la différentielle de F à μ^k . Par ailleurs, $K^k(\alpha^k)$ représente la constante de normalisation exprimée par

$$K^k(\alpha^k) = \left[\int_{\mathbb{R}^J} \exp [\alpha^k df(q^k, \mathbf{w})] q^k(\mathbf{w}) d\mathbf{w} \right]^{-1}. \quad (2.54)$$

Le choix pour la fonction h^k (2.53) est motivé par l'hypothèse de positivité et d'intégrabilité (propriétés d'une densité) ainsi que la cohérence avec la structure de la méthode du gradient. Par ailleurs, dans le cas d'un problème avec une contrainte de type entropie, une fonction exponentielle est un choix adéquat, voir [TB93].

L'algorithme d'approximation BV proposé par Fraysse *et al.* est donc basé sur l'équation itérative donnée par (2.52) et (2.53).

La performance de l'algorithme de type gradient dépend du pas de l'algorithme. Concernant le pas, on définit tout d'abord une fonction de α :

$$\begin{aligned} g^k : \mathbb{R} &\rightarrow \mathbb{R} \\ \alpha &\mapsto g^k(\alpha) = \mathcal{F} (K^k(\alpha)q^k(\mathbf{w}) \exp [\alpha df(q^k, \mathbf{w})]), \end{aligned} \quad (2.55)$$

ce qui nous permet de définir le pas optimal par :

$$\alpha^{opt} = \arg \max_{\alpha \in \mathbb{R}} g^k(\alpha). \quad (2.56)$$

Dans [FR14], pour un pas optimal de type (2.56), la convergence de cet algorithme de type gradient a été démontrée.

En pratique, la détermination du pas optimal est généralement coûteuse. Pour s'affranchir de cette difficulté, Fraysse *et al.* [FR14] ont adopté les pas sous-optimaux définis dans la suite.

Pas de l'algorithme

Pour s'affranchir de la difficulté à déterminer le pas optimal, Fraysse *et al.* ont adopté une stratégie de recherche linéaire de type backtracking comme décrit dans ce qui suit.

1. $h \in L^1(\mu) \Leftrightarrow \int_{\mathbb{R}^J} |h(\mathbf{w})| \mu(d\mathbf{w}) < \infty$

Tout d'abord, un pas initial est calculé à l'aide du développement de Taylor à l'ordre deux de $g^k(\alpha)$ en zéro. Pour α petit,

$$\tilde{g}^k(\alpha) = g^k(0) + \alpha \left(\frac{dg^k(\alpha)}{d\alpha} \Big|_{\alpha=0} \right) + \frac{1}{2} \alpha^2 \left(\frac{d^2g^k(\alpha)}{d\alpha^2} \Big|_{\alpha=0} \right) \quad (2.57)$$

est une bonne approximation de $g^k(\alpha)$. Le pas initial est déterminé comme le point critique de $\tilde{g}^k(\alpha)$ qui peut être obtenu par un calcul simple.

$$\hat{\alpha} = - \left(\frac{d^2g^k(\alpha)}{d\alpha^2} \Big|_{\alpha=0} \right)^{-1} \frac{dg^k(\alpha)}{d\alpha} \Big|_{\alpha=0} \quad (2.58)$$

Ensuite, nous déterminons notre pas sous-optimal α^{subopt} à partir de $\hat{\alpha}$. Si le pas $\hat{\alpha}$ obtenu par (2.58) fait augmenter le critère comme nous espérons, ce pas est utilisé. Sinon, on divise ce pas itérativement par deux jusqu'à obtenir un pas qui fait augmenter le critère.

$$\alpha^{subopt} = \begin{cases} \hat{\alpha} & \text{si } g^k(\hat{\alpha}) > g^k(0) \\ 2^{-t} \hat{\alpha} & \text{si } g^k(\hat{\alpha}) < g^k(0) \text{ et } g^k(2^{-t} \hat{\alpha}) > g^k(0) \end{cases} \quad (2.59)$$

où $t \geq 1$ est le nombre de division par deux.

On va appeler dans la suite cet algorithme « l'algorithme d'approximation bayésien variationnel gradient exponentialisé » et il peut être résumé comme suit :

Algorithm 2 Algorithme d'approximation Bayésienne Variationnelle Gradient exponentialisé (BV-Grad)

1. Initialiser ($q^0 \in \Omega$)
 2. répéter
 - a. déterminer $df(q^k, \mathbf{w})$ en utilisant (2.49)
 - b. déterminer le pas de l'algorithme α^{subopt} en utilisant (2.59)
 - c. calculer $q^{k+1}(\mathbf{w}) = K^k(\alpha^{subopt}) \exp[\alpha^{subopt} df(q^k, \mathbf{w})]$
- jusqu'à convergence
-

2.5 Conclusion

Nous avons introduit dans ce chapitre trois méthodes d'approximation analytique pour les densités de probabilité : la méthode d'approximation de Laplace, la méthode d'approximation bayésienne variationnelle classique et la méthode d'approximation bayésienne variationnelle de type gradient. Pour la méthode d'approximation de Laplace, son application est assez limitée en raison des inconvénients énumérés dans la partie 2.2. Comme illustré dans la partie 2.3, la méthode d'approximation BV classique souffre d'une vitesse de convergence faible. La méthode d'approximation BV de type gradient est une méthode récemment proposée et elle est plus rapide que la méthode d'approximation BV classique [FR14]. Cette méthode est basée sur une transposition de la méthode du gradient dans un

2.3.5 - Choix de séparation

espace de Hilbert vers l'espace des mesures. En fait, dans un espace de Hilbert, il existe d'autres méthodes itératives telles que la méthode du gradient conjugué, la méthode du sous-espace, qui sont plus rapides que la méthode du gradient. Inspiré par ce fait là, nous avons pu développer une méthode d'approximation BV qui est encore plus efficace que la méthode d'approximation BV de type gradient. Cette nouvelle méthode constitue l'objet du chapitre suivant et l'une des contributions majeurs de ma thèse.

3

Méthode d'approximation bayésienne variationnelle de sous-espace

3.1 Introduction

Nous avons introduit dans le chapitre précédent les algorithmes d'approximation BV de l'état de l'art : l'algorithme d'approximation BV classique (BVClass, voir Algorithme 1) et l'algorithme d'approximation BV gradient exponentialisé (BV-Grad, voir Algorithme 2). L'algorithme BV-Grad est fondé sur la transposition d'un algorithme d'optimisation classique – algorithme du gradient dans les espaces de Hilbert, vers l'espace fonctionnel impliqué dans le cadre BV. Fraysse et Rodet [FR14] ont montré qu'il est beaucoup plus efficace que les algorithmes BVClass. Un des objectifs principaux de cette thèse est de développer de nouveaux algorithmes encore plus efficaces que l'algorithme BV-Grad. L'idée est d'envisager une nouvelle direction de descente autre que celle du gradient. Un choix naturel serait de générer des directions conjuguées en utilisant la méthode du gradient conjugué, qui converge généralement plus vite que la méthode du gradient grâce à la mémoire des directions précédentes. Néanmoins, dans notre contexte BV, l'espace fonctionnel impliqué dans notre problème d'optimisation n'est plus un espace de Hilbert et à notre connaissance, il n'existe pas de notion de directions conjuguées. Par conséquent, la méthode du gradient conjugué ne s'adapte pas dans notre problème BV. Nous nous sommes orientés vers des méthodes de sous-espace où l'on peut s'affranchir de la notion de conjugaison. Le premier avantage de la méthode de sous-espace est sa direction de descente généralisée où la structure de Hilbert n'est plus nécessaire. Par ailleurs, la direction de descente peut être choisie librement dans un sous-espace de dimension supérieure à un. Cette flexibilité permet aux méthodes d'optimisation de sous-espace d'être généralement plus efficaces que les méthodes du gradient conjugué grâce au choix optimum de la direction. En transposant la méthode de sous-espace dans notre espace fonctionnel, nous proposons ici une méthode d'approximation bayésienne variationnelle accélérée. Dans la suite de ce chapitre, nous introduisons d'abord notre méthode d'approximation BV pro-

3.2.1 - Méthode itérative de sous-espace dans les espaces de Hilbert

posée. Ensuite, nous présentons une application de la méthode proposée dans un problème inverse linéaire avec une loi *a priori* de Student.

3.2 Développement de l'algorithme d'approximation BV de sous-espace

Dans la suite, nous donnons d'abord une présentation de la méthode de sous-espace dans les espaces de Hilbert. Ensuite, nous présentons notre méthode d'approximation BV proposée.

3.2.1 Méthode itérative de sous-espace dans les espaces de Hilbert

Les approches de type sous-espace généralisent les approches classiques de descente à pas optimaux. Au lieu de déterminer le pas optimal dans une direction donnée, ces approches déterminent l'optimum dans un sous-espace généralement de faible dimension (par exemple, 2, 3, 4, ...). En général, les approches itératives de sous-espace utilisent l'équation de mise à jour suivante :

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{d}^k = \mathbf{x}^k + \mathbf{D}^k \mathbf{s}^k, \quad (3.1)$$

où \mathbf{x}^k et \mathbf{x}^{k+1} représentent respectivement l'estimation à la k -ième et $(k+1)$ -ième itération, $\mathbf{d}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$ est la variation du vecteur \mathbf{x} à la $(k+1)$ -ième itération, $\mathbf{D}^k = [\boldsymbol{\delta}_1^k, \dots, \boldsymbol{\delta}_I^k]$ rassemble les I directions qui engendrent le sous-espace et le vecteur $\mathbf{s}^k = [s_1^k, \dots, s_I^k]^T$ inclut les pas le long de chaque direction $\{\boldsymbol{\delta}_i^k\}_{i=1, \dots, I}$. La méthode itérative de sous-espace offre une plus grande flexibilité pour le choix de la variation \mathbf{d}^k en prenant une combinaison linéaire des directions incluses dans \mathbf{D}^k .

A notre connaissance, le premier algorithme itératif de sous-espace a été proposé dans [MC69] avec un sous-espace engendré par l'opposé du gradient et la direction obtenue à l'itération précédente, sous le nom de *mémoire de gradient* (MG). Dans ce cas,

$$\mathbf{D}^k = [-\mathbf{g}^k, \mathbf{d}^{k-1}], \quad (3.2)$$

où $-\mathbf{g}^k$ est l'opposé du gradient à l'itération présente et $\mathbf{d}^{k-1} = \mathbf{x}^k - \mathbf{x}^{k-1}$. Le schéma itératif s'écrit dans ce cas comme

$$\mathbf{x}^{k+1} = \mathbf{x}^k - s_1^k \mathbf{g}^k + s_2^k \mathbf{d}^{k-1}, \quad (3.3)$$

où les pas s_1^k et s_2^k sont choisis afin d'avoir la plus grande diminution du critère. Cet algorithme peut être vu comme une généralisation de la méthode du gradient conjugué. En fait, lorsque le critère est quadratique, la méthode de sous-espace MG est équivalente à la méthode du gradient conjugué [Can69]. Plus récemment, un grand nombre d'autres méthodes itératives de sous-espace basées sur les sous-espaces différents, voir [NZ05] et [SS05] pour les exemples, ont été proposées. Un aperçu de l'ensemble des algorithmes itératifs de sous-espace existants, donné dans [CIM11], montre que \mathbf{D}^k est composé généralement d'une première composante qui correspond à la direction employée par

les approches classiques de descente, par exemple, l'opposé du gradient (l'algorithme du gradient), la direction de Newton (l'algorithme de Newton) ou la direction de Newton tronquée (l'algorithme de Newton tronqué), et d'une mémoire courte sur les directions aux itérations précédentes. Néanmoins, la direction de Newton s'écrit comme

$$\boldsymbol{\delta}^k = -(\mathbf{H}^k)^{-1}\mathbf{g}^k \quad (3.4)$$

où \mathbf{H}^k est la matrice hessienne du critère. Le calcul de la matrice hessienne n'est pas souvent facile et son inversion peut être très coûteuse voire impossible lorsque le problème est de grande taille. Dans ce cas, une direction approchée (la direction de Newton tronquée [DS83]) obtenue à l'aide de la méthode du gradient conjugué peut être utilisée. Néanmoins, dans notre cas, c'est difficile d'obtenir la matrice hessienne. En conséquence, nous considérons dans ce travail les sous-espaces qui ne comprennent pas la direction de Newton ou la direction de Newton tronquée.

En fait, comme montré dans le chapitre précédent, dans notre problème d'optimisation fonctionnelle du cadre BV, on sait calculer les gradients. Par conséquent, nous envisageons dans la suite deux types de sous-espace, un comprenant seulement les gradients, appelé *sous-espace de gradient* (SG) et l'autre composé de gradient ainsi que les directions aux itérations précédentes, connu sous le nom *sous-espace de super mémoire de gradient* (SMG). Le sous-espace SG est présenté dans les travaux de Shi & Shen [SS05, SS06, SS07] et décrit par :

$$\mathbf{D}^k = [-\mathbf{g}^k, \dots, -\mathbf{g}^{k-I+1}]. \quad (3.5)$$

Ici, $-\mathbf{g}^{k-l}$ est l'opposé du gradient à $(k-l)$ -ième itération avec l un entier entre 1 et $I-1$.

Le sous-espace SMG proposé dans [CL69] est une extension du sous-espace MG en incluant plusieurs directions précédentes. Ce sous-espace a été utilisé dans de nombreux travaux tels que [NY06, Yu08, CIM11]. Le sous-espace SMG possède la structure suivante :

$$\mathbf{D}^k = [-\mathbf{g}^k, \mathbf{d}^{k-1}, \dots, \mathbf{d}^{k-I+1}], \quad (3.6)$$

où $(\mathbf{d}^{k-l})_{l=1, \dots, I-1}$ sont des directions aux itérations précédentes définies par

$$\mathbf{d}^{k-l} = \mathbf{x}^{k-l+1} - \mathbf{x}^{k-l}. \quad (3.7)$$

Par rapport aux approches classiques de descente, les approches de sous-espace convergent vers la solution en moins d'itérations. Néanmoins, la complexité de chaque itération est plus importante. Plus le sous-espace est de grande dimension, plus chaque itération est efficace. Par contre, le coût de calcul de chaque itération est aussi plus élevé. Il faut donc faire un compromis. Dans le cas du sous-espace SMG, Chouzenoux *et al.* [CIM11] ont abordé une discussion à propos de la dimension du sous-espace basée sur les résultats de simulation sur les problèmes de restauration d'images. Ils ont démontré que, sous condition que le pas multi-dimensionnel \mathbf{s}^k soit défini en utilisant la stratégie de Majoration-Minimisation proposée dans [CIM11], un sous-espace SMG de dimension faible ($I = 2$), donne lieu au meilleur compromis. Dans ce cas là, le sous-espace SMG est un sous-espace MG.

3.2.2 Méthode d'approximation BV fondée sur la méthode d'optimisation de sous-espace

Nous allons définir dans cette partie notre méthode d'approximation BV fondée sur la transposition de la méthode itérative de sous-espace pour résoudre le problème d'optimisation fonctionnelle apparaissant dans le cadre BV, donné par (2.45). Notre méthode s'appuie sur un même schéma itératif que l'algorithme d'approximation BV-Grad et nous rappelons ici que l'équation de mise à jour est donnée par

$$q^{k+1}(\mathbf{w}) = h^k(\mathbf{w})q^k(\mathbf{w}), \quad (3.8)$$

Néanmoins, la fonction h^k prend maintenant en compte la méthode itérative de sous-espace. En gardant la forme exponentielle de h^k , nous proposons

$$h^k(\mathbf{w}) = K^k(\mathbf{s}^k) \exp [\mathbf{D}^k(\mathbf{w})\mathbf{s}^k], \quad (3.9)$$

où $K^k(\mathbf{s}^k)$ représente la constante de normalisation qui s'écrit :

$$K^k(\mathbf{s}^k) = \left[\int_{\mathbb{R}^J} \exp [\mathbf{D}^k(\mathbf{w})\mathbf{s}^k] q^k(\mathbf{w}) d\mathbf{w} \right]^{-1}, \quad (3.10)$$

et $\mathbf{D}^k(\mathbf{w}) = [\delta_1^k(\mathbf{w}), \dots, \delta_I^k(\mathbf{w})]$ est un ensemble de I (un entier supérieur à 1) directions qui engendrent le sous-espace. Nous devons signaler que les directions $(\delta_l^k(\mathbf{w}))_{l=1, \dots, I}$ ne sont plus données par des vecteurs mais par des fonctions. De même que dans les espaces de Hilbert, $\mathbf{s}^k = [s_1^k, \dots, s_I^k]^T$ est le pas multi-dimensionnel.

Dû à la forme exponentielle, (3.9) peut être aussi écrit comme :

$$h^k(\mathbf{w}) = K^k(\mathbf{s}^k) [\phi_1^k(\mathbf{w})]^{s_1^k} \dots [\phi_I^k(\mathbf{w})]^{s_I^k} \quad (3.11)$$

où $\phi_l^k(\mathbf{w}) = \exp[\delta_l^k(\mathbf{w})]$, pour $l = 1, \dots, I$.

On obtient finalement l'équation de mise à jour :

$$q^{k+1}(\mathbf{w}) = K^k(\mathbf{s}^k) q^k(\mathbf{w}) [\phi_1^k(\mathbf{w})]^{s_1^k} \dots [\phi_I^k(\mathbf{w})]^{s_I^k}. \quad (3.12)$$

Comme dans le cas des espaces de Hilbert, il faut déterminer le sous-espace utilisé.

Construction du sous-espace

Le sous-espace employé dans notre problème d'optimisation fonctionnelle peut être construit de la même façon que dans les espaces de Hilbert. Dans ce travail, nous envisageons seulement le sous-espace de gradient (3.5) et le sous-espace de super mémoire de gradient (3.6) car pour déterminer les directions de ces deux types de sous-espace, il suffit de savoir calculer le gradient du critère, qui est notre cas ici.

Nous présentons d'abord un sous-espace obtenu en transposant le sous-espace SG. Comme dans la partie 2.4, la transposition du gradient dans notre espace fonctionnel conduit à une fonction $df(q, \mathbf{w})$ provenant de la différentielle de Gâteaux de F qui est donnée explicitement par (2.49). Par conséquent, l'espace obtenu s'écrit comme :

$$\mathbf{D}_{SG}^k(\mathbf{w}) = [df(q^k, \mathbf{w}), \dots, df(q^{k-I+1}, \mathbf{w})], \quad (3.13)$$

Ce sous-espace est encore appelé le sous-espace de gradient (SG) et notre algorithme d'approximation BV basé sur ce sous-espace sera appelé l'algorithme d'approximation bayésienne variationnelle basé sur le sous-espace de gradient (BV-SG) dans la suite de cette thèse.

Le deuxième sous-espace envisagé ici est celui basé sur le sous-espace de super mémoire de gradient (SMG). Dans notre espace fonctionnel, il est constitué d'une composante donnée par $df(q, \mathbf{w})$ et des autres composantes correspondant aux directions précédentes. Sa structure est donnée par :

$$\mathbf{D}_{SMG}^k(\mathbf{w}) = [df(q^k, \mathbf{w}), d^{k-1}(\mathbf{w}), \dots, d^{k-I+1}(\mathbf{w})], \quad (3.14)$$

où $(d^{k-l}(\mathbf{w}))_{l=0, \dots, I-1}$ représentent les directions aux itérations précédentes. Comme expliqué auparavant, les directions $(d^{k-l})_{l=0, \dots, I-1}$ ne sont plus des vecteurs mais des fonctions dans notre problème d'optimisation. Elles sont données par :

$$d^{k-l}(\mathbf{w}) = \ln \left(\frac{q^{k-l+1}(\mathbf{w})}{K^{k-l}(\mathbf{s}^{k-l})q^{k-l}(\mathbf{w})} \right). \quad (3.15)$$

L'algorithme basé sur ce sous-espace sera nommé l'algorithme d'approximation bayésienne variationnelle basé sur le sous-espace de super mémoire de gradient (BV-SMG) dans la suite de cette thèse.

En ce qui concerne la dimension de sous-espace (I), de même que dans un espace de Hilbert, il faut faire un compromis entre l'efficacité de chaque itération et son coût de calcul. Dans ce travail, pour faire ce compromis, nous considérons $I = 2$.

Choix du pas de l'algorithme

La performance de nos algorithmes itératifs dépend du pas multi-dimensionnel \mathbf{s}^k . Concernant ce pas, nous définissons d'abord la fonction

$$\begin{aligned} g^k : \mathbb{R}^2 &\rightarrow \mathbb{R} \\ \mathbf{s} &\mapsto g^k(\mathbf{s}) = \mathcal{F} \left(K^k(\mathbf{s})q^k(\mathbf{w}) \exp [\mathbf{D}^k(\mathbf{w})\mathbf{s}] \right), \end{aligned} \quad (3.16)$$

puis le pas optimal est défini par

$$(\hat{\mathbf{s}}^{opt})^k = \arg \max_{\mathbf{s} \in \mathbb{R}^2} g^k(\mathbf{s}). \quad (3.17)$$

Un pas optimal donné par (3.17) est un choix idéal pour notre algorithme. Néanmoins, en général, le calcul du pas optimal est trop coûteux. Par conséquent, en pratique, beaucoup de stratégies ont été utilisées pour avoir les pas sous-optimaux en faisant un compromis entre le coût de calcul et la vitesse de convergence [NW00]. Une stratégie la plus simple est d'utiliser un pas fixe. L'inconvénient lié au pas fixe est qu'il faut choisir un pas suffisamment petit pour assurer la convergence de l'algorithme quelque soit l'initialisation. De plus, on peut voir dans [FR14] que notre fonctionnelle n'est pas *Lipschitz*. Dans ce cas, on ne sait pas quel pas fixe peut garantir la convergence de l'algorithme. Pour s'affranchir de ce problème, on favorise les stratégies qui pourraient choisir les pas adaptés aux itérations. Dans ce contexte, les approches de recherche linéaire ont été largement

3.2.2 - Méthode d'approximation BV fondée sur la méthode d'optimisation de sous-espace

utilisées dans le cas scalaire. Les approches de recherche linéaire typiques déterminent les pas sous-optimaux en essayant une séquence de valeurs jusqu'à trouver une valeur satisfaisant certaines conditions suffisantes, comme la condition de Wolfe [Wol69] ou la condition de Goldstein [GP67], voir aussi [NW00]. Ces conditions peuvent être facilement étendues aux cas multi-dimensionnels auxquels nous sommes confrontés dans nos algorithmes basés sur les sous-espaces. Néanmoins, la vitesse de convergence des approches de la recherche linéaire dépend fortement des paramètres qui contrôlent les limites de la condition choisie et du point de départ pour le pas. Un mauvais choix de ces paramètres entraîne une convergence lente, surtout dans le cas où le calcul du critère coûte cher, qui est notre cas ici. Dans ce travail, nous adoptons une stratégie de type backtracking pour déterminer les pas sous-optimaux, mais nous prenons un pas initial calculé en utilisant le développement de Taylor d'ordre deux pour avoir une convergence rapide, comme dans [FR14].

Nous calculons d'abord une approximation locale de $g^k(\mathbf{s})$ à l'origine en utilisant le développement de Taylor d'ordre deux.

$$\tilde{g}^k(\mathbf{s}) = g^k(\mathbf{0}) + \left(\frac{\partial g^k}{\partial \mathbf{s}} \Big|_{\mathbf{s}=\mathbf{0}} \right)^T \mathbf{s} + \frac{1}{2} \mathbf{s}^T \left(\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}} \right) \mathbf{s}, \quad (3.18)$$

où $\frac{\partial g^k}{\partial \mathbf{s}} \Big|_{\mathbf{s}=\mathbf{0}}$ est le vecteur du gradient ainsi que $\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}}$ est la matrice hessienne de la fonction g^k au point $\mathbf{s} = \mathbf{0}$. Lorsque s_1 et s_2 prennent les valeurs petites, $\tilde{g}^k(\mathbf{s})$ est une bonne approximation de $g^k(\mathbf{s})$. Nous calculons ensuite un pas en maximisant $\tilde{g}^k(\mathbf{s})$ qui est une fonction quadratique. Supposons que la matrice hessienne $\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}}$ soit inversible, nous pouvons obtenir que :

$$\hat{\mathbf{s}}^k = - \left(\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}} \right)^{-1} \frac{\partial g^k}{\partial \mathbf{s}} \Big|_{\mathbf{s}=\mathbf{0}}. \quad (3.19)$$

Ensuite, nous déterminons notre pas sous-optimal $(\hat{\mathbf{s}}^{subopt})^k$ de manière suivante : si $\hat{\mathbf{s}}^k$ obtenu par (3.19) fait augmenter notre critère, ce pas est utilisé. Sinon, on divise $\hat{\mathbf{s}}^k$ itérativement par deux jusqu'à obtenir un pas qui permet d'augmenter notre critère. En résumé,

$$(\hat{\mathbf{s}}^{subopt})^k = \begin{cases} \hat{\mathbf{s}}^k & \text{si } g^k(\hat{\mathbf{s}}^k) > g^k(\mathbf{0}) \\ 2^{-t} \hat{\mathbf{s}}^k & \text{si } g^k(\hat{\mathbf{s}}^k) < g^k(\mathbf{0}) \text{ et } g^k(2^{-t} \hat{\mathbf{s}}^k) > g^k(\mathbf{0}) \end{cases} \quad (3.20)$$

où $t \geq 1$ est le nombre de division par deux. Néanmoins, en pratique, le pas donné par (3.19) peut assurer l'augmentation de notre critère dans la plupart des itérations.

Notons que le calcul des pas sous-optimaux fait intervenir le calcul des coefficients d'une matrice hessienne. Le calcul de ces coefficients est relativement coûteux en temps de calculs, ce qui rend les sous-espaces de plus grande dimension (3 ou 4) peu efficace (car dans ce cas, la matrice hessienne à déterminer est de taille plus grande).

Finalement, notre algorithme d'approximation BV basé sur les sous-espaces peut être résumé par l'algorithme 3.

Jusqu'ici, nous avons défini nos algorithmes d'approximation BV basés sur le sous-espace SG et le sous-espace MG. L'objet ensuite est de tester les algorithmes proposés et de voir s'ils sont plus efficaces que les algorithmes d'approximation BV de l'état de l'art

Algorithm 3 Algorithme d'approximation bayésienne variationnelle basé sur le sous-espace

1. Initialiser ($q^0 \in \Omega$)
 2. répéter
 - a. déterminer le sous-espace $\mathbf{D}^k(\mathbf{w})$ en utilisant (3.13) ou (3.14)
 - b. déterminer le pas multi-dimensionnel de l'algorithme \mathbf{s}^k selon (3.20)
 - c. calculer $q^{k+1}(\mathbf{w}) = K^k(\mathbf{s}^k) q^k(\mathbf{w}) \exp[\mathbf{D}^k(\mathbf{w})\mathbf{s}^k]$
- jusqu'à convergence
-

ainsi que les autres méthodes d'approximation (MCMC par exemple). Pour ce faire, nous faisons une application des algorithmes proposés à un problème inverse linéaire en utilisant une loi *a priori* de Student et nous effectuons des comparaisons avec les algorithmes de l'état de l'art sur un problème de tomographie de petite taille afin de pouvoir comparer un grand nombre de méthodes.

3.3 Application à un problème inverse en utilisant une loi a priori de Student

L'objectif de cette partie est de donner un exemple de l'application de nos algorithmes d'approximation BV proposés sur un problème inverse linéaire où la matrice du système n'est pas facilement inversible. Le problème inverse considéré ici est associé au modèle direct (1.2) qui a été défini et utilisé dans les chapitres précédents. L'objectif est de développer des approches bayésiennes pour résoudre ce type de problème inverse, qui est généralement mal-posé, en utilisant une loi *a priori* de Student et les algorithmes d'approximation BV proposés.

3.3.1 Formulation bayésienne

Nous considérons encore un bruit blanc gaussien de moyenne nulle et de précision γ_n . En ce qui concerne les informations *a priori* sur les paramètres d'intérêt, nous nous intéressons ici aux informations de parcimonie en considérant les distributions à queue lourde. De plus, pour la facilité des implémentations numériques, comme dans l'approche BV classique (voir partie 2.3.3), nous devons considérer un *a priori* conjugué. Une famille des lois conjuguées avec la vraisemblance gaussienne est la famille de GSM [JM06, WS00] qui comprend des distributions à queue lourde telle que la loi de Student ou la loi gaussienne généralisée. Dans cette partie, nous choisissons la loi *a priori* de Student car elle peut être écrite sous une forme intégrale de GSM et la loi de sa variable cachée est aussi une loi conjuguée comme cela sera détaillé par la suite.

La loi de Student dépend d'un paramètre de forme ν pour lequel une petite valeur conduit à une distribution à queue lourde. Nous considérons ici une loi de Student

3.3.2 - Applications des méthodes d'approximation BV

séparable. Dans ce cas, pour $i = 1, \dots, N$, nous supposons que $X_i = U_i/\sqrt{Z_i}$ où

$$U_i \sim \mathcal{N}(u_i|0, \gamma_p^{-1}) \quad (3.21)$$

$$Z_i \sim \mathcal{G}(z_i|\nu/2, \nu/2), \quad (3.22)$$

avec γ_p la précision de la loi de U_i (gaussienne) et $\nu/2$ comme les paramètres de la loi de Z_i (Gamma).

Par conséquent, la densité de probabilité de X_i peut être écrite :

$$\begin{aligned} p(x_i) &= \int_{\mathbb{R}} p(x_i|z_i)p(z_i)dz_i \\ &= \int_{\mathbb{R}} \mathcal{N}(x_i|0, (\gamma_p z_i)^{-1})\mathcal{G}(z_i|\nu/2, \nu/2)dz_i \\ &\propto \int_{\mathbb{R}} \sqrt{\gamma_p z_i} \exp\left[-\frac{1}{2}\gamma_p z_i x_i^2\right] z_i^{\nu/2-1} \exp\left[-\frac{\nu}{2}z_i\right] dz_i, \end{aligned} \quad (3.23)$$

où z_i est une variable cachée. Dans la suite, nous utilisons un vecteur $\mathbf{z} = (z_1, \dots, z_N)$ pour regrouper toutes les variables cachées. Nous pouvons voir que $p(x_i|z_i)$ est une distribution gaussienne qui est conjuguée avec la vraisemblance gaussienne. La loi $p(z_i)$ est une loi Gamma qui est une loi *a priori* conjuguée pour z_i qui est un paramètre d'échelle dans $p(x_i|z_i)$. Ces conjugaisons jouent un rôle important dans le développement des approches bayésiennes variationnelles efficaces.

Dans ce cas, la loi *a posteriori* jointe de \mathbf{x} et de \mathbf{z} sachant \mathbf{y} est donnée par :

$$p(\mathbf{x}, \mathbf{z}|\mathbf{y}) \propto \gamma_n^{M/2} \exp\left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2}\right] \prod_{i=1}^N \sqrt{\gamma_p z_i} \exp\left[-\frac{1}{2}\gamma_p z_i x_i^2\right] z_i^{\nu/2-1} \exp\left[-\frac{\nu}{2}z_i\right]. \quad (3.24)$$

Au lieu d'estimer \mathbf{x} à partir de la loi *a posteriori* $p(\mathbf{x}|\mathbf{y})$, nous estimons conjointement \mathbf{x} et \mathbf{z} à partir de leur loi *a posteriori* conjointe $p(\mathbf{x}, \mathbf{z}|\mathbf{y})$ qui est de forme plus simple. Nous pouvons remarquer que

$$\begin{aligned} \hat{\mathbf{x}} &= \int \mathbf{x}p(\mathbf{x}, \mathbf{z}|\mathbf{y})d\mathbf{x}d\mathbf{z} \\ &= \int \mathbf{x}p(\mathbf{x}|\mathbf{y})d\mathbf{x}, \end{aligned} \quad (3.25)$$

qui nous permet de retrouver l'espérance *a posteriori* de la loi $p(\mathbf{x}|\mathbf{y})$.

3.3.2 Applications des méthodes d'approximation BV

Nous utilisons dans la suite une nouvelle variable Θ à regrouper tous les paramètres à estimer. Dans notre cas, $\Theta = \{\mathbf{x}, \mathbf{z}\}$. L'objectif de BV est de donner une loi approchante séparable q_{Θ} de la loi *a posteriori* $p(\Theta|\mathbf{y})$. Dans la mise en œuvre de nos algorithmes d'approximation BV, nous choisissons d'abord l'hypothèse de séparation. Comme nous avons évoqué dans la partie 2.3.5, on peut considérer soit la séparation totale soit la séparation partielle. Néanmoins, on a montré dans la partie 2.3.5 que l'hypothèse de

séparabilité partielle entraîne des calculs pour inverser des matrices qui sont coûteux. Pour éviter ce type de calcul, nous prenons ici l'hypothèse de séparabilité totale donnée par :

$$\begin{aligned} q_{\Theta}(\Theta) &= q_{\mathbf{x}}(\mathbf{x})q_{\mathbf{z}}(\mathbf{z}) \\ &= \prod_{i=1}^N q_i(x_i) \prod_{j=1}^N \tilde{q}_j(z_j). \end{aligned} \quad (3.26)$$

Sous cette hypothèse, nous menons une optimisation alternée par rapport à $q_{\mathbf{z}}$ et $q_{\mathbf{x}}$, qui nous conduit à résoudre les problèmes d'optimisation suivants :

$$q_{\mathbf{z}}^{k+1} = \arg \max_{q_{\mathbf{z}}} \mathcal{F}(q_{\mathbf{x}}^k q_{\mathbf{z}}), \quad (3.27)$$

$$q_{\mathbf{x}}^{k+1} = \arg \max_{q_{\mathbf{x}}} \mathcal{F}(q_{\mathbf{x}} q_{\mathbf{z}}^{k+1}), \quad (3.28)$$

Les optimisations fonctionnelles par rapport à $q_{\mathbf{z}}$ et $q_{\mathbf{x}}$ sont effectuées à l'aide des approches d'approximation BV. Comme la loi conditionnelle *a posteriori* $p(\mathbf{z}|\mathbf{x}, \mathbf{y})$ est elle-même séparable, elle peut être approchée efficacement en utilisant l'approximation BV classique, comme nous allons l'expliquer ci-dessous. Néanmoins, nos algorithmes BV accélérés sont utilisés à approcher la loi *a posteriori* de \mathbf{x} pour améliorer la vitesse de convergence.

En fait, nous avons employé des lois *a priori* conjuguées : $p(\mathbf{x}|\mathbf{z})$ est la loi *a priori* conjuguée avec la vraisemblance $p(\mathbf{y}|\mathbf{x}, \gamma_n)$ et $p(z_i)$ est la loi *a priori* conjuguée avec $p(x_i|z_i)$. Par conséquent, on sait que l'optimum pour $q_{\mathbf{x}}$ est une loi gaussienne et pour $q_{\mathbf{z}}$ une loi Gamma. Nous allons donc prendre des lois approchantes dans ces deux familles. En conséquence, on obtient

$$q_{\mathbf{x}}^k(\mathbf{x}) = \prod_i \mathcal{N}(x_i | (\mathbf{m}_k)_i, (\boldsymbol{\sigma}_k^2)_i), \quad (3.29)$$

$$q_{\mathbf{z}}^k(\mathbf{z}) = \prod_j \mathcal{G}(z_j | (\mathbf{a}_k)_j, (\mathbf{b}_k)_j), \quad (3.30)$$

où \mathbf{m}_k et $\boldsymbol{\sigma}_k^2$ sont des vecteurs regroupant les moyennes et les variances de tous les éléments de \mathbf{x} et \mathbf{a}_k et \mathbf{b}_k représentent des vecteurs incluant les paramètres de $q_{\mathbf{z}}^k$.

Ainsi, la mise à jour de ces lois approchantes revient à la réactualisation de leurs paramètres.

Optimisation de $q_{\mathbf{z}}$

L'algorithme d'approximation BV classique (BVClass) est résumé par l'algorithme 1 dans le chapitre 2. Celui-ci utilise la formule (2.12) qui nous permet d'obtenir que pour

3.3.2 - Applications des méthodes d'approximation BV

chaque $i = 1, \dots, N$,

$$\begin{aligned} \tilde{q}_i^{k+1}(z_i) &\propto \exp \left[\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{q_{\mathbf{x}}^k \prod_{j < i} \tilde{q}_j^{k+1} \prod_{j > i} \tilde{q}_j^k} \right] \\ &\propto \exp \left[\int \sum_{i=1}^N \left(\frac{\nu-1}{2} \ln(z_i) - \frac{\gamma_p}{2} z_i x_i^2 - \frac{\nu}{2} z_i \right) \right. \\ &\quad \left. \times \prod_{l=1}^N q_l^k(x_l) dx_l \prod_{j=1}^{i-1} \tilde{q}_j^{k+1}(z_j) \prod_{j=i+1}^N \tilde{q}_j^k(z_j) dz_j \right]. \end{aligned} \quad (3.31)$$

Puisque $p(z_i | \mathbf{x}, z_{j \neq i}) = p(z_i | \mathbf{x})$, nous pouvons obtenir que

$$\begin{aligned} \tilde{q}_i^{k+1}(z_i) &\propto \exp \left[\frac{\nu-1}{2} \ln(z_i) - \int \left(\frac{\gamma_p}{2} z_i x_i^2 + \frac{\nu}{2} z_i \right) q_i^k(x_i) dx_i \right] \\ &\propto \exp \left[\frac{\nu-1}{2} \ln(z_i) - \frac{\nu + \gamma_p ((\mathbf{m}_k)_i^2 + (\boldsymbol{\sigma}_k^2)_i)}{2} z_i \right]. \end{aligned} \quad (3.32)$$

La distribution $\tilde{q}_i^{k+1}(z_i)$ est identifiée comme une loi Gamma de paramètres

$$\forall i = 1, \dots, N \quad (\mathbf{a}_{k+1})_i = \frac{\nu + 1}{2} = (\mathbf{a})_i \quad (3.33)$$

$$(\mathbf{b}_{k+1})_i = \frac{\nu}{2} + \frac{\gamma_p}{2} [(\mathbf{m}_k)_i^2 + (\boldsymbol{\sigma}_k^2)_i] \quad (3.34)$$

où le paramètre \mathbf{a} est une constante.

Nous remarquons dans (3.33) et (3.34) que les paramètres de la loi \tilde{q}_i^{k+1} , i.e. $(\mathbf{a}_{k+1})_i$ et $(\mathbf{b}_{k+1})_i$ ne dépendent pas de paramètres des lois \tilde{q}_j^{k+1} avec $j \neq i$, i.e. $(\mathbf{a}_{k+1})_j$ et $(\mathbf{b}_{k+1})_j$. Les expressions des paramètres de $(\tilde{q}_i^{k+1})_{i=1, \dots, N}$ sont donc explicites. C'est parce que $p(\mathbf{z} | \mathbf{x}, \mathbf{y})$ est lui-même séparable.

Optimisation de $q_{\mathbf{x}}$

Pour l'optimisation de $q_{\mathbf{x}}$, nous allons appliquer nos deux algorithmes BV proposés : l'algorithme BV-SG et l'algorithme BV-MG.

Approximation BV basée sur le Sous-espace Gradient (BV-SG) Selon les équations (3.12), (3.13) et (2.49), nous obtenons une distribution de \mathbf{x} qui dépend du

pas \mathbf{s} :

$$\begin{aligned}
q_{\mathbf{x}}^{\mathbf{s}}(\mathbf{x}) &= K^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \exp \left(s_1 df(q_{\mathbf{x}}^k, \mathbf{x}) + s_2 df(q_{\mathbf{x}}^{k-1}, \mathbf{x}) \right) \\
&= K^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \exp \left(s_1 \sum_{i=1}^N d_i f(q_{\mathbf{x}}^k, x_i) + s_2 \sum_{i=1}^N d_i f(q_{\mathbf{x}}^{k-1}, x_i) \right) \\
&= K^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \left[\prod_{i=1}^N \exp(d_i f(q_{\mathbf{x}}^k, x_i)) \right]^{s_1} \left[\prod_{i=1}^N \exp(d_i f(q_{\mathbf{x}}^{k-1}, x_i)) \right]^{s_2} \\
&= \tilde{K}^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \prod_{i=1}^N \left[\frac{\exp \left(\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{(\prod_{j \neq i} q_j^k) q_{\mathbf{z}}^{k+1}} \right)}{q_i^k(x_i)} \right]^{s_1} \\
&\quad \times \left[\frac{\exp \left(\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{(\prod_{j \neq i} q_j^{k-1}) q_{\mathbf{z}}^k} \right)}{q_i^{k-1}(x_i)} \right]^{s_2} \\
&= \tilde{K}^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \prod_{i=1}^N \left(\frac{q_i^{r_1}(x_i)}{q_i^k(x_i)} \right)^{s_1} \left(\frac{q_i^{r_2}(x_i)}{q_i^{k-1}(x_i)} \right)^{s_2}, \tag{3.35}
\end{aligned}$$

où $(q_i^{r_1})_{i=1, \dots, N}$ et $(q_i^{r_2})_{i=1, \dots, N}$ sont des fonctions intermédiaires définies comme suit :

$$q_i^{r_1}(x_i) = \exp \left[\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{(\prod_{j \neq i} q_j^k) q_{\mathbf{z}}^{k+1}} \right] \tag{3.36}$$

$$q_i^{r_2}(x_i) = \exp \left[\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{(\prod_{j \neq i} q_j^{k-1}) q_{\mathbf{z}}^k} \right]. \tag{3.37}$$

Nous calculons ces distributions intermédiaires de la manière suivante :

$$\begin{aligned}
q_i^{r_1}(x_i) &= \exp \left[- \int \left(\frac{\gamma_n}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + \sum_{i=1}^N \frac{1}{2} \gamma_p z_i x_i^2 \right) \left(\prod_{j \neq i} q_j^k(x_j) dx_j \right) q_{\mathbf{z}}^{k+1}(\mathbf{z}) d\mathbf{z} \right] \\
&\propto \exp \left[- \frac{\gamma_n}{2} \left(x_i^2 \text{diag}(\mathbf{A}^T \mathbf{A})_i - 2x_i (\mathbf{A}^T \mathbf{y})_i + 2x_i (\mathbf{A}^T \mathbf{A} \mathbf{m}_k)_i \right. \right. \\
&\quad \left. \left. - 2x_i \text{diag}(\mathbf{A}^T \mathbf{A})_i (\mathbf{m}_k)_i \right) - \frac{\gamma_p}{2} \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} x_i^2 \right], \tag{3.38}
\end{aligned}$$

où $\text{diag}(\mathbf{M})$ représente un vecteur dont les éléments sont les éléments diagonaux de la matrice \mathbf{M} . Les détails de calcul peuvent être trouvés en annexe A.1.

A partir de (3.38), nous pouvons voir que $q_i^{r_1}$ correspond, à une constante de normalisation près, à la densité d'une loi gaussienne dont la moyenne $(\mathbf{m}_{r_1})_i$ et la variance $(\boldsymbol{\sigma}_{r_1}^2)_i$ sont exprimées explicitement par

$$(\boldsymbol{\sigma}_{r_1}^2)_i = \left[\gamma_n \text{diag}(\mathbf{A}^T \mathbf{A})_i + \gamma_p \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} \right]^{-1}, \tag{3.39}$$

$$(\mathbf{m}_{r_1})_i = (\boldsymbol{\sigma}_{r_1}^2)_i \left[\gamma_n (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_k + \text{diag}(\mathbf{A}^T \mathbf{A}) \circ \mathbf{m}_k) \right]_i. \tag{3.40}$$

Ici, \circ représente le produit d'Hadamard entre deux vecteurs.

En comparant les expressions (3.36) et (3.37), nous pouvons voir que $q_i^{r_2}$ à la k -ième itération est identique que $q_i^{r_1}$ à la $(k-1)$ -ième itération. En conséquence, $q_i^{r_2}$ correspond

3.3.2 - Applications des méthodes d'approximation BV

à une loi gaussienne avec la moyenne et la variance respectivement données par

$$(\boldsymbol{\sigma}_{r_2}^2)_i = \left[\gamma_n \text{diag}(\mathbf{A}^T \mathbf{A})_i + \gamma_p \frac{(\mathbf{a})_i}{(\mathbf{b}_k)_i} \right]^{-1}, \quad (3.41)$$

$$(\mathbf{m}_{r_2})_i = (\boldsymbol{\sigma}_{r_2}^2)_i \left[\gamma_n (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_{k-1} + \text{diag}(\mathbf{A}^T \mathbf{A}) \circ \mathbf{m}_{k-1}) \right]_i. \quad (3.42)$$

Après avoir obtenu les paramètres de $(q_i^{r_1})_{i=1,\dots,N}$ et $(q_i^{r_2})_{i=1,\dots,N}$, d'après (3.35), nous pouvons en déduire l'expression de $q_{\mathbf{x}}^{\mathbf{s}}(\mathbf{x}) = \prod_i q_i^{\mathbf{s}}(x_i)$ où chaque composante $q_i^{\mathbf{s}}(x_i)$ est calculée comme suit :

$$\begin{aligned} q_i^{\mathbf{s}}(x_i) &= \tilde{K}_i^k(\mathbf{s}) q_i^k(x_i) \left(\frac{q_i^{r_1}(x_i)}{q_i^k(x_i)} \right)^{s_1} \left(\frac{q_i^{r_2}(x_i)}{q_i^{k-1}(x_i)} \right)^{s_2} \\ &= \tilde{K}_i^k(\mathbf{s}) \frac{1}{\sqrt{2\pi(\boldsymbol{\sigma}_k^2)_i}} \exp\left(-\frac{(x_i - (\mathbf{m}_k)_i)^2}{2(\boldsymbol{\sigma}_k^2)_i}\right) \\ &\quad \times \left(\frac{\frac{1}{\sqrt{2\pi(\boldsymbol{\sigma}_{r_1}^2)_i}} \exp\left(-\frac{(x_i - (\mathbf{m}_{r_1})_i)^2}{2(\boldsymbol{\sigma}_{r_1}^2)_i}\right)}{\frac{1}{\sqrt{2\pi(\boldsymbol{\sigma}_k^2)_i}} \exp\left(-\frac{(x_i - (\mathbf{m}_k)_i)^2}{2(\boldsymbol{\sigma}_k^2)_i}\right)}} \right)^{s_1} \left(\frac{\frac{1}{\sqrt{2\pi(\boldsymbol{\sigma}_{r_2}^2)_i}} \exp\left(-\frac{(x_i - (\mathbf{m}_{r_2})_i)^2}{2(\boldsymbol{\sigma}_{r_2}^2)_i}\right)}{\frac{1}{\sqrt{2\pi(\boldsymbol{\sigma}_{k-1}^2)_i}} \exp\left(-\frac{(x_i - (\mathbf{m}_{k-1})_i)^2}{2(\boldsymbol{\sigma}_{k-1}^2)_i}\right)}} \right)^{s_2} \\ &= \tilde{K}_i^k(\mathbf{s}) \exp\left[-\frac{(x_i - (\mathbf{m}_k)_i)^2}{2(\boldsymbol{\sigma}_k^2)_i} - s_1 \frac{(x_i - (\mathbf{m}_{r_1})_i)^2}{2(\boldsymbol{\sigma}_{r_1}^2)_i} + s_1 \frac{(x_i - (\mathbf{m}_k)_i)^2}{2(\boldsymbol{\sigma}_k^2)_i} \right. \\ &\quad \left. - s_2 \frac{(x_i - (\mathbf{m}_{r_2})_i)^2}{2(\boldsymbol{\sigma}_{r_2}^2)_i} + s_2 \frac{(x_i - (\mathbf{m}_{k-1})_i)^2}{2(\boldsymbol{\sigma}_{k-1}^2)_i} \right]. \end{aligned} \quad (3.43)$$

Finalement, on peut voir que $q_i^{\mathbf{s}}(x_i)$ est encore une loi gaussienne avec sa moyenne et sa variance explicitement exprimées par

$$\boldsymbol{\sigma}_{\mathbf{s}}^2 = \left[\frac{1}{\boldsymbol{\sigma}_k^2} + s_1 \left(\frac{1}{\boldsymbol{\sigma}_{r_1}^2} - \frac{1}{\boldsymbol{\sigma}_k^2} \right) + s_2 \left(\frac{1}{\boldsymbol{\sigma}_{r_2}^2} - \frac{1}{\boldsymbol{\sigma}_{k-1}^2} \right) \right]^{-1}, \quad (3.44)$$

$$\mathbf{m}_{\mathbf{s}} = \boldsymbol{\sigma}_{\mathbf{s}}^2 \left[\frac{\mathbf{m}_k}{\boldsymbol{\sigma}_k^2} + s_1 \left(\frac{\mathbf{m}_{r_1}}{\boldsymbol{\sigma}_{r_1}^2} - \frac{\mathbf{m}_k}{\boldsymbol{\sigma}_k^2} \right) + s_2 \left(\frac{\mathbf{m}_{r_2}}{\boldsymbol{\sigma}_{r_2}^2} - \frac{\mathbf{m}_{k-1}}{\boldsymbol{\sigma}_{k-1}^2} \right) \right]. \quad (3.45)$$

Dans les équations ci-dessus, nous avons omis les indices de composante $(\cdot)_i$ pour des raisons de clarté.

Les distributions obtenues sont fonctions du pas \mathbf{s} . Nous prenons dans ce cas un pas sous-optimal défini par (3.20) (voir annexe B.1 pour les détails de calculs). Par conséquent, $\boldsymbol{\sigma}_{k+1}^2 = \boldsymbol{\sigma}_{\hat{\mathbf{s}}_{\text{subopt}}}^2$ et $\mathbf{m}_{k+1} = \mathbf{m}_{\hat{\mathbf{s}}_{\text{subopt}}}$.

Approximation BV basée sur le Sous-espace Mémoire de Gradient (BV-MG)

Les calculs dans cette partie sont faits de façon analogue que dans ceux de BV-SG. En utilisant les équations (3.12), (3.14), (2.49) et (3.15), nous obtenons une distribution de

\mathbf{x} qui dépend du pas \mathbf{s} :

$$\begin{aligned}
 q_{\mathbf{x}}^{\mathbf{s}}(\mathbf{x}) &= K^k(\mathbf{s})q_{\mathbf{x}}^k(\mathbf{x}) \exp(s_1 df(q_{\mathbf{x}}^k, \mathbf{x}) + s_2 d^{k-1}(\mathbf{x})) \\
 &= K^k(\mathbf{s})q_{\mathbf{x}}^k(\mathbf{x}) \exp\left(s_1 \sum_i df(q_{\mathbf{x}}^k, x_i) + s_2 d^{k-1}(\mathbf{x})\right) \\
 &= \tilde{K}^k(\mathbf{s})q_{\mathbf{x}}^k(\mathbf{x}) \prod_i \left[\frac{\exp\left(\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{(\prod_{j \neq i} q_j^k) q_{\mathbf{z}}^{k+1}}\right)}{q_i^k(x_i)} \right]^{s_1} \left[\frac{q_i^k(x_i)}{q_i^{k-1}(x_i)} \right]^{s_2} \\
 &= \tilde{K}^k(\mathbf{s})q_{\mathbf{x}}^k(\mathbf{x}) \prod_i \left(\frac{q_i^{r_1}(x_i)}{q_i^k(x_i)} \right)^{s_1} \left(\frac{q_i^k(x_i)}{q_i^{k-1}(x_i)} \right)^{s_2} \tag{3.46}
 \end{aligned}$$

où la fonction $q_i^{r_1}$ a été définie par (3.36) et correspond à une distribution gaussienne dont la variance et la moyenne sont données respectivement par (3.39) et (3.40). Par ailleurs, q_i^k et q_i^{k-1} sont aussi des distributions gaussiennes. En comparant (3.46) et (3.35), nous pouvons voir que (3.46) peut être obtenu à partir de (3.35) en remplaçant $q_i^{r_2}$ par q_i^k . En conséquence, dans ce cas, le calcul de $q_{\mathbf{x}}^{\mathbf{s}}(\mathbf{x}) = \prod_i q_i^{\mathbf{s}}(x_i)$ suit les mêmes étapes que (3.43) en remplaçant \mathbf{m}_{r_2} par \mathbf{m}_k et $\sigma_{r_2}^2$ par σ_k^2 .

Finalement, la fonction $q_i^{\mathbf{s}}(x_i)$ est aussi une distribution gaussienne dont la moyenne et la variance sont données par

$$\sigma_{\mathbf{s}}^2 = \left[\frac{1}{\sigma_k^2} + s_1 \left(\frac{1}{\sigma_{r_1}^2} - \frac{1}{\sigma_k^2} \right) + s_2 \left(\frac{1}{\sigma_k^2} - \frac{1}{\sigma_{k-1}^2} \right) \right]^{-1}, \tag{3.47}$$

$$\mathbf{m}_{\mathbf{s}} = \sigma_{\mathbf{s}}^2 \left[\frac{\mathbf{m}_k}{\sigma_k^2} + s_1 \left(\frac{\mathbf{m}_{r_1}}{\sigma_{r_1}^2} - \frac{\mathbf{m}_k}{\sigma_k^2} \right) + s_2 \left(\frac{\mathbf{m}_k}{\sigma_k^2} - \frac{\mathbf{m}_{k-1}}{\sigma_{k-1}^2} \right) \right]. \tag{3.48}$$

Encore une fois, nous avons omis les indices de composantes $(\cdot)_i$ dans (3.47) et (3.48). De plus, les équations de mise à jour sont de la même forme que celles obtenues par BV-SG. Observons (3.44) et (3.47). Nous pouvons voir que $\sigma_{\mathbf{s}}^2$ est égale à l'inverse d'une combinaison linéaire de trois termes : le premier terme est l'inverse de la variance à l'itération présente, le second terme est causé par le gradient et le troisième terme viens de la mémoire du gradient ou de la direction précédente. Les équations (3.45) et (3.48) montrent que $\mathbf{m}_{\mathbf{s}}$ a une structure similaire. À partir de ces expressions, on remarque que les équations de mise à jour des paramètres de distribution obtenues par BV-SG et BV-MG ont une structure similaire que celle des méthodes de sous-espace dans un espace Hilbert : $\mathbf{x}^{k+1} = \mathbf{x}^k + s_1 \boldsymbol{\delta}_1^k + s_2 \boldsymbol{\delta}_2^k$.

En effet, l'algorithme BV-Grad peut être identifié comme un cas particulière de l'algorithme BV-SG ou l'algorithme BV-MG en mettant s_2 à zéro qui fait disparaître le troisième terme dans (3.44) et (3.45) ou (3.47) et (3.48). Grâce au terme supplémentaire (le troisième terme), l'algorithme BV-SG et l'algorithme BV-MG sont capables de donner une approximation plus proche que l'algorithme BV-Grad en une seule itération.

Les distributions obtenues sont encore fonctions du pas \mathbf{s} . Nous prenons un pas sous-optimal défini par (3.20) (voir annexe B.2 pour plus de détails de calculs). Par conséquent, $\sigma_{k+1}^2 = \sigma_{\hat{\mathbf{s}}^{subopt}}^2$ and $\mathbf{m}_{k+1} = \mathbf{m}_{\hat{\mathbf{s}}^{subopt}}$.

En tout, nous obtenons deux algorithmes, l'un utilise BV-SG, l'autre utilise BV-MG, qui sont résumés par l'algorithme 4 et l'algorithme 5, respectivement.

3.3.3 - Les algorithmes de reconstruction non-supervisés

Algorithm 4 L'algorithme de reconstruction supervisé basé sur un *a priori* de Student en utilisant BV-SG

1. Initialiser $q_{\mathbf{x}}^0$ et $q_{\mathbf{z}}^0$: initialiser \mathbf{m}_0 , σ_0^2 et \mathbf{a}_0 , \mathbf{b}_0
 2. Calculer $q_{\mathbf{z}}^{k+1} = \prod_i \tilde{q}_i^{k+1}$: calculer ses paramètres en utilisant (3.33) et (3.34)
 3. Calculer $q_{\mathbf{x}}^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer q_i^{r1} et q_i^{r2} : calculer ses paramètres en utilisant (3.39)-(3.40) et (3.41)-(3.42), respectivement
 - (b) Calculer le pas sous-optimal selon l'annexe B.1
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (3.44) et (3.45) où le pas s est égal au pas sous-optimal obtenu à l'étape (b)
 4. Retourner à l'étape 2 jusqu'à convergence
-

Algorithm 5 L'algorithme de reconstruction supervisé basé sur un *a priori* de Student en utilisant BV-MG

1. Initialiser $q_{\mathbf{x}}^0$ et $q_{\mathbf{z}}^0$: initialiser \mathbf{m}_0 , σ_0^2 et \mathbf{a}_0 , \mathbf{b}_0
 2. Calculer $q_{\mathbf{z}}^{k+1} = \prod_i \tilde{q}_i^{k+1}$: calculer ses paramètres en utilisant (3.33) et (3.34)
 3. Calculer $q_{\mathbf{x}}^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer q_i^{r1} : calculer ses paramètres en utilisant (3.39)-(3.40)
 - (b) Calculer le pas sous-optimal selon l'annexe B.2
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (3.47) et (3.48) où le pas s est égal au pas sous-optimal obtenu à l'étape (b)
 4. Retourner à l'étape 2 jusqu'à convergence
-

3.3.3 Les algorithmes de reconstruction non-supervisés

Les deux algorithmes de reconstruction décrits dans la partie précédente sont dits « supervisés » car la performance de ces algorithmes dépend des valeurs des hyperparamètres choisies, particulièrement des valeurs de γ_n (le paramètre de précision du bruit) et γ_p (le paramètre de précision dans le modèle *a priori* (3.23)), qui contrôlent le compromis entre la fidélité aux données et les connaissances *a priori*. En pratique, les algorithmes supervisés déterminent ces valeurs de manière empirique : choisir les valeurs selon les expériences ou faire plusieurs essais pour trouver une valeur relativement meilleure. Cette manière de choisir les hyperparamètres s'avère peu rapide et peu pratique. C'est pour cette raison que nous allons étendre les algorithmes supervisés obtenus à des algorithmes non-supervisés en estimant conjointement les hyperparamètres. Au total, trois hyperparamètres sont introduits dans notre modélisation probabiliste : γ_n , γ_p et ν . Dans la suite, nous estimons les hyperparamètres γ_n et γ_p . Néanmoins, nous choisissons de ne pas estimer ν car il contrôle la forme de la loi et donc l'information *a priori*. On veut introduire l'information de parcimonie, donc on veut fixer la forme de la loi *a priori* en fixant la valeur de ν .

Désormais nous estimons donc aussi γ_n et γ_p . Les lois *a priori* conjuguées sont généralement introduites pour faciliter l'implémentation des algorithmes d'approximation BV. Dans ce cas, les lois Gamma sont des distributions conjuguées pour γ_n et γ_p .

Nous supposons donc que

$$p(\gamma_n) = \mathcal{G}(\gamma_n | \tilde{a}_n, \tilde{b}_n) \quad (3.49)$$

$$p(\gamma_p) = \mathcal{G}(\gamma_p | \tilde{a}_p, \tilde{b}_p). \quad (3.50)$$

En pratique, nous ne possédons pas d'informations *a priori* sur γ_p et γ_n . Ainsi, nous prenons $\tilde{a}_n = 0$, $\tilde{b}_n = 0$ et $\tilde{a}_p = 0$, $\tilde{b}_p = 0$ qui nous permettent d'obtenir des lois non-informatives (loi de Jeffreys).

Sous ces hypothèses, nous pouvons en déduire la loi *a posteriori* conjointe des paramètres d'intérêt ainsi que des hyperparamètres :

$$\begin{aligned} p(\mathbf{x}, \mathbf{z}, \gamma_n, \gamma_p | \mathbf{y}, \nu) &\propto \gamma_n^{M/2} \exp \left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2} \right] \\ &\times \prod_{i=1}^N \sqrt{\gamma_p z_i} \exp \left[-\frac{1}{2} \gamma_p z_i x_i^2 \right] z_i^{\nu/2-1} \exp \left[-\frac{\nu}{2} z_i \right] \\ &\times \gamma_n^{\tilde{a}_n-1} \exp \left[-\tilde{b}_n \gamma_n \right] \gamma_p^{\tilde{a}_p-1} \exp \left[-\tilde{b}_p \gamma_p \right]. \end{aligned} \quad (3.51)$$

Comme dans la partie précédente, cette loi *a posteriori* est approchée par des lois séparables en utilisant des algorithmes d'approximation BV. Ici, la loi conditionnelle *a posteriori* $p(\mathbf{z}, \gamma_n, \gamma_p | \mathbf{y}, \nu)$ est séparable, elle est donc approchée en utilisant l'approximation BV classique. Et la loi *a posteriori* de \mathbf{x} est approchée par nos algorithmes d'approximation BV accélérés, comme dans la partie précédente.

Ici, nous avons $\Theta = \{\mathbf{x}, \mathbf{z}, \gamma_n, \gamma_p\}$. En ce qui concerne la loi approchante, nous prenons encore l'hypothèse de séparation totale qui nous donne

$$\begin{aligned} q_\Theta(\Theta) &= q_{\mathbf{x}}(\mathbf{x}) q_{\mathbf{z}}(\mathbf{z}) q_{\gamma_n}(\gamma_n) q_{\gamma_p}(\gamma_p) \\ &= \prod_{i=1}^N q_i(x_i) \prod_{j=1}^N \tilde{q}_j(z_j) q_{\gamma_n}(\gamma_n) q_{\gamma_p}(\gamma_p). \end{aligned} \quad (3.52)$$

Sous cette hypothèse, nous menons une optimisation alternée par rapport à $q_{\mathbf{z}}$, $q_{\mathbf{x}}$, q_{γ_n} et q_{γ_p} , qui nous conduit à résoudre les problème d'optimisation suivants :

$$q_{\mathbf{z}}^{k+1} = \arg \max_{q_{\mathbf{z}}} \mathcal{F} \left(q_{\mathbf{x}}^k q_{\mathbf{z}}^k q_{\gamma_n}^k q_{\gamma_p}^k \right), \quad (3.53)$$

$$q_{\mathbf{x}}^{k+1} = \arg \max_{q_{\mathbf{x}}} \mathcal{F} \left(q_{\mathbf{x}}^{k+1} q_{\mathbf{z}}^k q_{\gamma_n}^k q_{\gamma_p}^k \right), \quad (3.54)$$

$$q_{\gamma_n}^{k+1} = \arg \max_{q_{\gamma_n}} \mathcal{F} \left(q_{\mathbf{x}}^{k+1} q_{\mathbf{z}}^{k+1} q_{\gamma_n}^k q_{\gamma_p}^k \right), \quad (3.55)$$

$$q_{\gamma_p}^{k+1} = \arg \max_{q_{\gamma_p}} \mathcal{F} \left(q_{\mathbf{x}}^{k+1} q_{\mathbf{z}}^{k+1} q_{\gamma_n}^{k+1} q_{\gamma_p}^k \right). \quad (3.56)$$

En ce qui concerne $q_{\mathbf{z}}$ et $q_{\mathbf{x}}$, les mises à jour sont faites de la même façon que dans le cas supervisé et nous ne rappelons pas les calculs. Néanmoins, l'introduction des lois q_{γ_n} et q_{γ_p} dans les calculs apporte un changement dans les équations de mise à jour obtenues

3.3.3 - Les algorithmes de reconstruction non-supervisés

(les équations (3.34) et (3.39)-(3.42)) : les paramètres γ_n et γ_p sont remplacés par leurs moyennes sous les lois $q_{\gamma_n}^k$ et $q_{\gamma_p}^k$.

Grâce à l'utilisation des lois *a priori* conjuguées, l'optimum pour q_{γ_n} et pour q_{γ_p} sont des lois Gamma. Nous prenons donc des lois approchantes dans la famille Gamma :

$$q_{\gamma_n}^k(\gamma_n) = \mathcal{G}(\gamma_n | a_{\gamma_n}^k, b_{\gamma_n}^k), \quad (3.57)$$

$$q_{\gamma_p}^k(\gamma_p) = \mathcal{G}(\gamma_p | a_{\gamma_p}^k, b_{\gamma_p}^k). \quad (3.58)$$

Par conséquent, l'optimisation des lois approchantes est équivalente à l'optimisation de leurs paramètres.

L'optimisation de q_{γ_n}

Encore une fois, en utilisant la formule (2.12), nous pouvons obtenir

$$\begin{aligned} q_{\gamma_n}^{k+1}(\gamma_n) &\propto \exp \left[\langle \ln p(\mathbf{x}, \mathbf{z}, \gamma_n, \gamma_p, \mathbf{y} | \nu) \rangle_{q_{\mathbf{x}}^{k+1} q_{\mathbf{z}}^{k+1} q_{\gamma_p}^k} \right] \\ &\propto \exp \left[\left(\frac{M}{2} + \tilde{a}_n - 1 \right) \ln(\gamma_n) - \gamma_n \left(\int \frac{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x} + \tilde{b}_n \right) \right] \\ &\propto \exp \left[\left(\frac{M}{2} + \tilde{a}_n - 1 \right) \ln(\gamma_n) \right. \\ &\quad \left. - \gamma_n \left(\frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{m}_{k+1}\|^2 + \frac{1}{2} \sum_{i=1}^N \text{diag}(\mathbf{A}^T \mathbf{A})_i (\sigma_{k+1}^2)_i + \tilde{b}_n \right) \right]. \end{aligned} \quad (3.59)$$

Donc $q_{\gamma_n}^{k+1}$ est une loi Gamma avec des paramètres :

$$a_{\gamma_n}^{k+1} = \frac{M}{2} + \tilde{a}_n = a_{\gamma_n} \quad (3.60)$$

$$b_{\gamma_n}^{k+1} = \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{m}_{k+1}\|^2 + \frac{1}{2} \sum_{i=1}^N \text{diag}(\mathbf{A}^T \mathbf{A})_i (\sigma_{k+1}^2)_i + \tilde{b}_n \quad (3.61)$$

En prenant la moyenne de la loi approchante comme l'estimation de l'hyperparamètre, nous avons

$$\langle \gamma_n \rangle^{k+1} = a_{\gamma_n} / b_{\gamma_n}^{k+1}. \quad (3.62)$$

L'optimisation de q_{γ_p}

La distribution q_{γ_p} est déterminée de façon analogue à q_{γ_n} . En utilisant (2.12), nous avons

$$\begin{aligned} q_{\gamma_p}^{k+1}(\gamma_p) &\propto \exp \left[\langle \ln p(\mathbf{x}, \mathbf{z}, \gamma_n, \gamma_p, \mathbf{y} | \nu) \rangle_{q_{\mathbf{x}}^{k+1} q_{\mathbf{z}}^{k+1} q_{\gamma_n}^{k+1}} \right] \\ &\propto \exp \left[\left(\frac{N}{2} + \tilde{a}_p - 1 \right) \ln(\gamma_p) - \gamma_p \left(\int \sum_{i=1}^N \frac{z_i x_i^2}{2} \prod_i q_i^{k+1}(x_i) \tilde{q}_i^{k+1}(z_i) d\mathbf{x} d\mathbf{z} + \tilde{b}_p \right) \right] \\ &\propto \exp \left[\left(\frac{N}{2} + \tilde{a}_p - 1 \right) \ln(\gamma_p) - \gamma_p \left(\frac{1}{2} \sum_{i=1}^N \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} ((\mathbf{m}_{k+1})_i^2 + (\sigma_{k+1}^2)_i) + \tilde{b}_p \right) \right] \end{aligned} \quad (3.63)$$

qui est identifiée comme une loi Gamma de paramètres :

$$a_{\gamma_p}^{k+1} = N/2 + \tilde{a}_p = a_{\gamma_p} \quad (3.64)$$

$$b_{\gamma_p}^{k+1} = \frac{1}{2} \sum_{i=1}^N \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} ((\mathbf{m}_{k+1})_i^2 + (\boldsymbol{\sigma}_{k+1}^2)_i) + \tilde{b}_p \quad (3.65)$$

De façon analogue, nous actualisons l'estimation de γ_p par

$$\langle \gamma_p \rangle^{k+1} = a_{\gamma_p} / b_{\gamma_p}^{k+1}. \quad (3.66)$$

Par conséquent, nous pouvons obtenir deux algorithmes de reconstruction non-supervisés décrits par l'algorithme 6 et l'algorithme 7.

Algorithm 6 L'algorithme de reconstruction Non-Supervisé basé sur un *a priori* de Student en utilisant BV-SG (NSBV-SG)

1. Initialiser $q_{\mathbf{x}}^0$ et $q_{\mathbf{z}}^0$: initialiser \mathbf{m}_0 , $\boldsymbol{\sigma}_0^2$ et \mathbf{a}_0 , \mathbf{b}_0
 2. Calculer $q_{\mathbf{z}}^{k+1} = \prod_i \tilde{q}_i^{k+1}$: calculer leurs paramètres en utilisant (3.33) et (3.34)
 3. Calculer $q_{\mathbf{x}}^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer $q_i^{r_1}$ et $q_i^{r_2}$: calculer leurs paramètres en utilisant (3.39)-(3.40) et (3.41)-(3.42), respectivement
 - (b) Calculer le pas sous-optimal selon l'annexe B.1
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (3.44) et (3.45) où le pas \mathbf{s} est égal au pas sous-optimal obtenu à l'étape (b)
 4. Calculer $q_{\gamma_n}^{k+1}$: calculer ses paramètres en utilisant (3.60) et (3.61)
 5. Calculer $q_{\gamma_p}^{k+1}$: calculer ses paramètres en utilisant (3.64) et (3.65)
 6. Retourner à l'étape 2 jusqu'à convergence
-

Algorithm 7 L'algorithme de reconstruction Non-Supervisé basé sur un *a priori* de Student en utilisant BV-MG (NSBV-MG)

1. Initialiser $q_{\mathbf{x}}^0$ et $q_{\mathbf{z}}^0$: initialiser \mathbf{m}_0 , $\boldsymbol{\sigma}_0^2$ et \mathbf{a}_0 , \mathbf{b}_0
 2. Calculer $q_{\mathbf{z}}^{k+1} = \prod_i \tilde{q}_i^{k+1}$: calculer leurs paramètres en utilisant (3.33) et (3.34)
 3. Calculer $q_{\mathbf{x}}^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer $q_i^{r_1}$: calculer ses paramètres en utilisant (3.39)-(3.40)
 - (b) Calculer le pas sous-optimal selon l'annexe B.2
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (3.47) et (3.48) où le pas \mathbf{s} est égal au pas sous-optimal obtenu à l'étape (b)
 4. Calculer $q_{\gamma_n}^{k+1}$: calculer ses paramètres en utilisant (3.60) et (3.61)
 5. Calculer $q_{\gamma_p}^{k+1}$: calculer ses paramètres en utilisant (3.64) et (3.65)
 6. Retourner à l'étape 2 jusqu'à convergence
-

3.4 Évaluations expérimentales

Nous avons développé dans les parties précédentes des algorithmes de reconstruction en utilisant les algorithmes d'approximation BV de sous-espace : BV-SG et BV-MG. Dans cette partie, nous allons évaluer l'efficacité de nos algorithmes en les comparant avec des méthodes d'approximation de la densité de probabilité existantes, notamment la méthode MCMC, des algorithmes d'approximation BV classique avec une hypothèse de séparabilité totale et avec une hypothèse de séparabilité partielle, et l'algorithme d'approximation BV gradient exponentialisé (BV-Grad).

Pour ce faire, nous appliquons ces méthodes d'approximation au même problème inverse traité dans les parties précédentes.

3.4.1 Descriptions des algorithmes à comparer

Le but est d'obtenir une approximation, soit numérique, soit analytique, de la loi *a posteriori* $p(\mathbf{x}, \mathbf{z}|\mathbf{y})$ (voir (3.24)) qui nous permet ensuite d'obtenir une estimation de paramètre d'intérêt \mathbf{x} .

- **Méthode de Monte Carlo par Chaîne de Markov – l'échantillonnage de Gibbs (MCMC-Gibbs)**

Nous avons présenté dans chapitre 1 deux algorithmes MCMC les plus utilisés : l'algorithme de Metropolis-Hastings et l'échantillonnage de Gibbs. Nous considérons ici l'échantillonnage de Gibbs car il s'appuie sur des lois conditionnelles simulables qui est notre cas ici : la loi conditionnelle $p(\mathbf{x}|\mathbf{z}, \mathbf{y})$ est une loi gaussienne multidimensionnelle et la loi conditionnelle $p(\mathbf{z}|\mathbf{x}, \mathbf{y})$ est une loi Gamma séparable. Même si la loi conditionnelle conjointe $p(\mathbf{x}, \mathbf{z}|\mathbf{y})$ est de forme compliquée, les lois conditionnelles associées appartient aux familles de distributions usuelles qui sont simulables. L'échantillonnage de Gibbs pour notre problème suit des étapes suivantes :

- (a) Choisir des valeurs initiales : \mathbf{z}^0 et \mathbf{x}^0
- (b) Étant donné \mathbf{x}^k , simuler $\mathbf{z}^{k+1} \sim p(\mathbf{z}|\mathbf{x}^k, \mathbf{y})$ où

$$\begin{aligned} p(\mathbf{z}|\mathbf{x}^k, \mathbf{y}) &\propto \prod_{i=1}^N \exp \left[-z_i \left(\frac{1}{2} \gamma_p (x_i^k)^2 + \frac{\nu}{2} \right) \right] z_i^{\nu/2-1/2} \\ &= \prod_{i=1}^N \mathcal{G} \left(z_i \middle| \frac{\nu}{2} + \frac{1}{2}, \frac{1}{2} \gamma_p (x_i^k)^2 + \frac{\nu}{2} \right) \end{aligned} \quad (3.67)$$

qui est une loi Gamma séparable facilement simulable en pratique.

- (c) Étant donné \mathbf{z}^{k+1} , simuler $\mathbf{x}^{k+1} \sim p(\mathbf{x}|\mathbf{z}^{k+1}, \mathbf{y})$ où

$$\begin{aligned} p(\mathbf{x}|\mathbf{z}^{k+1}, \mathbf{y}) &\propto \exp \left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2} \right] \prod_{i=1}^N \exp \left[-\frac{1}{2} \gamma_p z_i^{k+1} x_i^2 \right] \\ &= \mathcal{N}(\mathbf{x}|\mathbf{m}_{k+1}, \boldsymbol{\Sigma}_{k+1}) \end{aligned} \quad (3.68)$$

avec

$$\boldsymbol{\Sigma}_{k+1} = [\gamma_n \mathbf{A}^T \mathbf{A} + \gamma_p \text{Diag}(\mathbf{z}^{k+1})]^{-1} = \mathbf{R}_{k+1}^T \mathbf{R}_{k+1} \quad (3.69)$$

$$\mathbf{m}_{k+1} = \gamma_n \boldsymbol{\Sigma}_{k+1} \mathbf{A}^T \mathbf{y}. \quad (3.70)$$

Ici, $\text{Diag}(\mathbf{z})$ est une matrice diagonale où les composantes diagonales sont égaux aux éléments du vecteur \mathbf{z} et \mathbf{R}_{k+1} est une matrice obtenue par la décomposition de Cholesky de la matrice de covariance Σ_{k+1} . En pratique, les échantillons \mathbf{x}^{k+1} peuvent être générés en suivant deux étapes :

- i. générer un vecteur des échantillons \mathbf{t}^{k+1} suivant la loi normale centrée réduite,
- ii. générer le vecteur \mathbf{x}^{k+1} en utilisant

$$\mathbf{x}^{k+1} = \mathbf{R}_{k+1}\mathbf{t}^{k+1} + \mathbf{m}_{k+1}. \quad (3.71)$$

(d) Retourner à l'étape (b) jusqu'à convergence de la chaîne de Markov (un temps de chauffe est nécessaire).

• **Méthode d'approximation bayésienne variationnelle Classique basée sur une hypothèse de Séparabilité Partielle (BVClass-SP)**

La méthode d'approximation BV classique a été décrite par l'algorithme 1 dans chapitre 2. Cet algorithme dépend du degré de séparabilité choisi. Nous utilisons dans cette partie la séparabilité partielle qui a été largement utilisée :

$$q_{\Theta}(\Theta) = q_{\mathbf{x}}(\mathbf{x})q_{\mathbf{z}}(\mathbf{z}). \quad (3.72)$$

Sous cette hypothèse de séparabilité, nous avons $P = 2$. Nous imposons une séparabilité entre \mathbf{x} et \mathbf{z} , mais pas de séparabilité interne entre les éléments de \mathbf{x} et \mathbf{z} . Néanmoins, comme indiqué précédemment, la loi conditionnelle $p(\mathbf{z}|\mathbf{x}, \mathbf{y})$ est séparable, en conséquence, l'utilisation de la séparabilité interne ou non de \mathbf{z} ne fait aucune différence sur le résultat obtenu, comme nous allons le montrer dans la suite. Selon l'algorithme 1, la détermination de la loi approchante q_{Θ} est faite en alternant l'optimisation de $q_{\mathbf{x}}$ et de $q_{\mathbf{z}}$ avec (2.12). Encore, grâce à l'utilisation des lois conjuguées, la loi approchante optimale $q_{\mathbf{x}}$ est une distribution gaussienne multidimensionnelle et la loi $q_{\mathbf{z}}$ est une distribution Gamma :

$$q_{\mathbf{x}}^k(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mathbf{m}_k, \Sigma_k), \quad (3.73)$$

$$q_{\mathbf{z}}^k(\mathbf{z}) = \mathcal{G}(\mathbf{z}|\mathbf{a}_k, \mathbf{b}_k). \quad (3.74)$$

Par conséquent, l'optimisation des lois est réduite à optimiser leurs paramètres.

– L'optimisation de $q_{\mathbf{z}}$:

$$\begin{aligned} q_{\mathbf{z}}^{k+1}(\mathbf{z}) &\propto \exp(\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{q_{\mathbf{x}}^k}) \\ &\propto \exp \left[\sum_{i=1}^N \left(\frac{\nu}{2} - \frac{1}{2} \right) \ln z_i - \frac{\nu}{2} z_i - \int \frac{1}{2} \gamma_p z_i x_i^2 q_{\mathbf{x}}^k d\mathbf{x} \right] \\ &\propto \prod_{i=1}^N z_i^{\frac{\nu}{2} - \frac{1}{2}} \exp \left[-z_i \left(\frac{\nu}{2} + \frac{1}{2} \gamma_p ((\mathbf{m}_k)_i^2 + \text{diag}(\Sigma_k)_i) \right) \right]. \end{aligned} \quad (3.75)$$

Nous pouvons trouver que $q_{\mathbf{z}}^{k+1}$ est séparable. D'après (3.75), nous pouvons obtenir des équations de mise à jour des paramètres de la loi $q_{\mathbf{z}}^{k+1} = \prod_i \tilde{q}_i^{k+1} : \forall i = 1, \dots, N$

$$(\mathbf{a}_{k+1})_i = \frac{\nu}{2} + \frac{1}{2} = (\mathbf{a})_i, \quad (3.76)$$

$$(\mathbf{b}_{k+1})_i = \frac{\nu}{2} + \frac{1}{2} \gamma_p [(\mathbf{m}_k)_i^2 + \text{diag}(\Sigma_k)_i]. \quad (3.77)$$

3.4.1 - Descriptions des algorithmes à comparer

– L'optimisation de $q_{\mathbf{x}}$:

$$\begin{aligned} q_{\mathbf{x}}^{k+1}(\mathbf{x}) &\propto \exp\left(\langle \ln p(\mathbf{x}, \mathbf{z}, \mathbf{y}) \rangle_{q_{\mathbf{z}}^{k+1}}\right) \\ &\propto \exp\left(-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2} - \sum_{i=1}^N \frac{1}{2} \gamma_p \langle z_i \rangle^{k+1} x_i^2\right) \end{aligned} \quad (3.78)$$

où $\langle z_i \rangle^{k+1}$ est la moyenne de \tilde{q}_i^{k+1} . Nous pouvons donc obtenir les paramètres de $q_{\mathbf{x}}^{k+1}$:

$$\Sigma_{k+1} = [\gamma_n \mathbf{A}^T \mathbf{A} + \gamma_p \text{Diag}(\mathbf{a}./\mathbf{b}_{k+1})]^{-1} \quad (3.79)$$

$$\mathbf{m}_{k+1} = \gamma_n \Sigma_{k+1} \mathbf{A}^T \mathbf{y} \quad (3.80)$$

où $./$ représente une division composante par composante de deux vecteurs.

Remarque. Les équations ci-dessus montrent qu'à chaque itération, l'évaluation de la matrice de covariance de la loi approchante nécessite d'inverser une matrice de taille $N \times N$.

- **Méthode d'approximation bayésienne variationnelle basée sur une hypothèse de Séparabilité Totale (BVClass-ST)**

Nous considérons ici la séparabilité totale qui a été aussi employée dans nos algorithmes d'approximation BV accélérés (voir (3.26)). Sous cette hypothèse de séparabilité, nous avons le nombre de partitions des paramètres P égal à $2N$. D'après l'algorithme 1, l'optimisation de q_{Θ} est faite en alternant l'optimisation de chaque composante de la loi $q_{\Theta} : (q_i)_{i=1,\dots,N}$ et $(\tilde{q}_i)_{i=1,\dots,N}$.

Dans ce cas, les distributions $(q_i)_{i=1,\dots,N}$ sont gaussiennes et $(\tilde{q}_i)_{i=1,\dots,N}$ sont des lois Gamma, comme données par (3.29) et (3.30). En utilisant (2.12), nous pouvons obtenir des équations de mise à jour en suivant des étapes de calculs analogues au cas précédent : $\forall i = 1, \dots, N$,

$$(\mathbf{a}_{k+1})_i = \frac{\nu}{2} + \frac{1}{2} = (\mathbf{a})_i \quad (3.81)$$

$$(\mathbf{b}_{k+1})_i = \frac{\nu}{2} + \frac{1}{2} \gamma_p [(\mathbf{m}_k)_i^2 + (\sigma_k)_i^2] \quad (3.82)$$

$$(\sigma_{k+1}^2)_i = \left[\gamma_n \text{diag}(\mathbf{A}^T \mathbf{A})_i + \gamma_p \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} \right]^{-1}, \quad (3.83)$$

$$(\mathbf{m}_{k+1})_i = (\sigma_{k+1}^2)_i \left[\gamma_n (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \tilde{\mathbf{m}}(i) + \text{diag}(\mathbf{A}^T \mathbf{A}) \circ \tilde{\mathbf{m}}(i)) \right]_i. \quad (3.84)$$

où $\tilde{\mathbf{m}}(i) = ((\mathbf{m}_{k+1})_1, \dots, (\mathbf{m}_{k+1})_{i-1}, (\mathbf{m}_k)_i, \dots, (\mathbf{m}_k)_N)^T$ est un vecteur dépendant de i .

Nous pouvons voir que les expressions de \mathbf{a}_{k+1} et \mathbf{b}_{k+1} sont explicites. Cependant, ce n'est pas le cas pour \mathbf{m}_{k+1} . Par conséquent, l'algorithme alterné doit être utilisé pour faire les optimisations.

Remarque. Pour chaque pixel x_i , il faut déterminer $\mathbf{A}^T \mathbf{A} \tilde{\mathbf{m}}(i)$ et $\text{diag}(\mathbf{A}^T \mathbf{A}) \circ \tilde{\mathbf{m}}(i)$.

- **Méthode d'approximation bayésienne variationnelle gradient exponentialisé (BV-Grad)**

Comme discuté précédemment, la méthode d'approximation BV-Grad peut être considérée comme un cas particulier de BV-SG ou BV-MG en mettant le second

pas $s_2 = 0$. Par conséquent, il n'est pas nécessaire de donner des calculs ici. Les équations de mise à jour des paramètres de q_z sont données par (3.33) et (3.34). Les équations de mise à jour de q_x peuvent être facilement obtenues en mettant $s_2 = 0$ dans (3.47) et (3.48).

Nous appliquons ces algorithmes ainsi que nos algorithmes basés sur BV-SG et BV-MG à un même problème de tomographie, qui est un exemple de problème inverse avec la matrice \mathbf{A} identifiée comme la transformée de Radon. Dans la suite, nous donnons d'abord la configuration de simulation. Ensuite, les résultats de simulation sont montrés.

3.4.2 Configuration de simulation

Des algorithmes de reconstruction sont testés avec des données synthétiques générées à partir d'une image parcimonieuse de taille 64×64 composée de 7 pics dont les amplitudes varient entre 0.5 et 1. Une telle image de petite taille est utilisée en considérant des limitations des algorithmes à comparer. En fait, l'implémentation de la méthode MCMC-Gibbs et la méthode d'approximation BVClass-SP nécessite d'inverser des matrices (voir (3.69) et (3.79)) dont la taille augmente avec la taille d'image, l'opération d'inversion devient très coûteuse voire impossible lorsque la taille de l'image est trop grande¹.

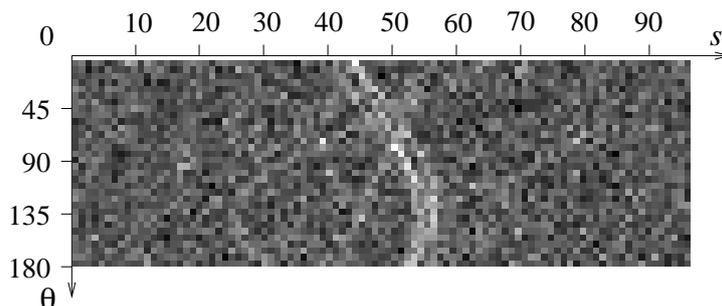


FIGURE 3.1 – Données : sinogramme obtenue avec 32 angles de projections et 95 détecteurs.

Les données simulées sont des projections en géométrie parallèle. Ces projections sont collectées à 32 angles différents θ , qui sont uniformément distribués sur $[0, \pi[$ avec 95 détecteurs. Nous avons ajoutés aussi un bruit blanc gaussien de moyenne nulle et d'écart-type 0.3. Par conséquent, les données (voir figure 3.1) présentent un faible rapport signal sur bruit. Dans ce problème, le nombre des paramètres inconnus est égal à $64 \times 64 = 4096$, qui est plus grand que la taille des données : $32 \times 95 = 3040$. En conséquence, nous abordons ici un problème inverse mal-posé.

3.4.3 Résultats de simulation dans le cas supervisé

Tous les algorithmes itératifs présentés précédemment ont été implémentés avec une même initialisation : zéro comme moyenne et un comme variance pour toutes les compo-

1. Comme expliqué dans la partie 2.3.5, les approches itératives ainsi que les approximations diagonales des matrices peuvent être utilisées pour contourner les inversions de matrices [BMK11]. Les comparaisons avec ce type d'approches sont données dans le chapitre 4.

3.4.3 - Résultats de simulation dans le cas supervisé

santes de \mathbf{x} . Comme nous utilisons ici des algorithmes supervisés, nous fixons les valeurs des hyperparamètres. Dans notre simulation, les hyperparamètres γ_n et γ_p sont mis à 1 et 20, respectivement. Le degré de liberté ν est mis à 0.1 pour avoir une loi *a priori* à queue lourde. Par ailleurs, la loi *a priori* qu'on a introduit est de moyenne nulle.

Afin d'évaluer la qualité de reconstruction, nous appliquons aussi une méthode analytique la plus couramment utilisée pour inverser la transformée de Radon : la méthode de rétroprojection filtrée (FBP). Elle consiste à filtrer toutes les projections et à les rétroprojeter sur toute l'image dans la même direction où ils avaient été projetés. L'avantage de cette méthode est qu'elle est rapide.

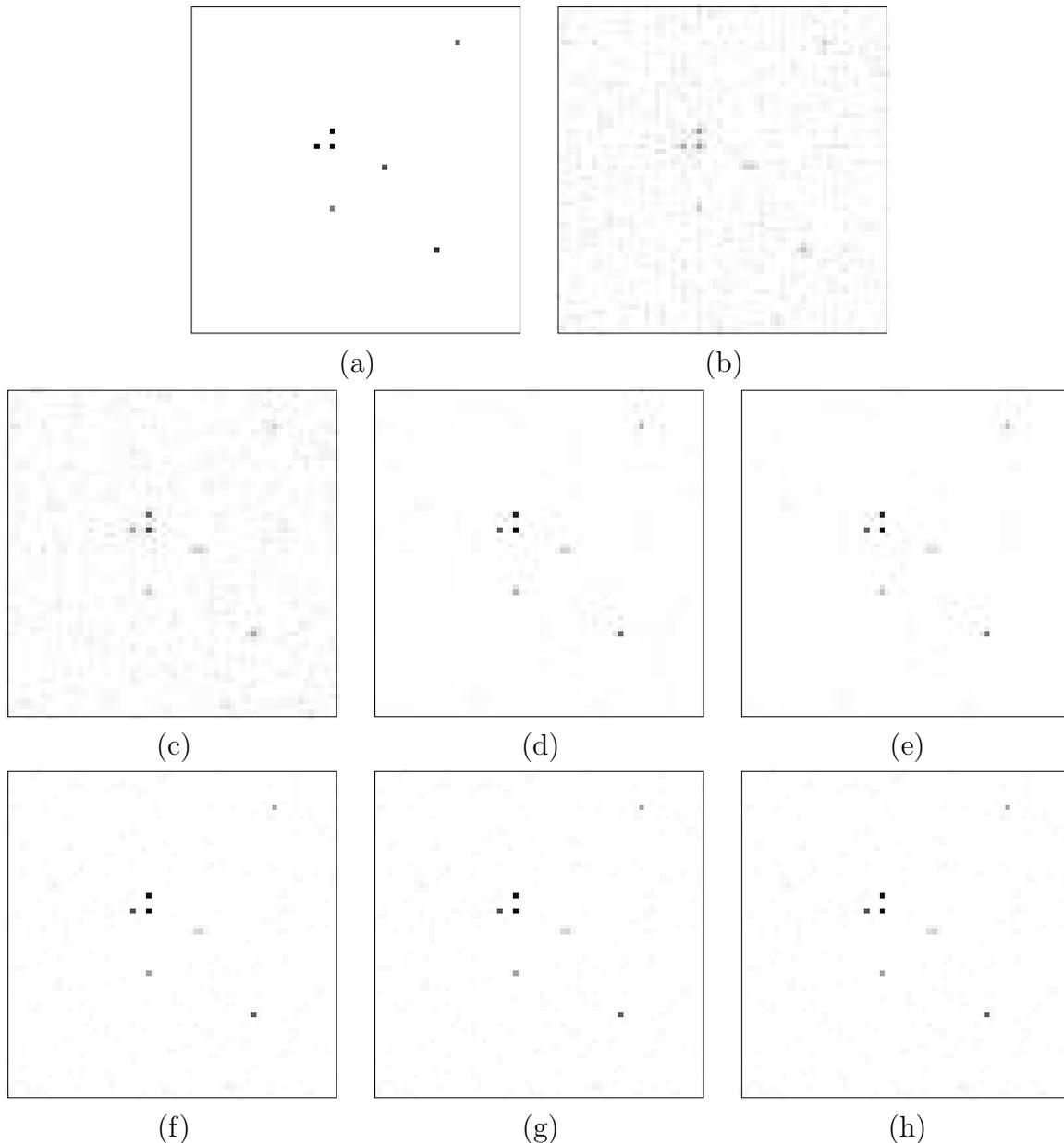


FIGURE 3.2 – Les images sont présentées avec le même niveau de gris inversé. (a) L'image Vrai, (b) FBP avec un filtre rampe, (c) l'échantillonnage de Gibbs, (d) BVClass-SP, (e) BVClass-ST, (f) BV-Grad, (g) BV-SG, (h) BV-MG.

Nous montrons dans la figure 3.2 l'image vraie (voir figure 3.2 (a)) ainsi que les reconstructions obtenues avec les différentes méthodes (voir figure 3.2 (b) - (h)). L'image donnée par la figure 3.2 (b) montre que la méthode de rétroprojection filtrée ne donne pas une reconstruction satisfaisante pour nos données bruitées. La comparaison de la figure 3.2 (b) avec les figures 3.2 (c)-(h) montre que les approches bayésiennes sont plus robustes au bruit que la méthode FBP.

Les reconstructions montrées par la figure 3.2 (g) et (h) sont obtenues avec les approches utilisant nos algorithmes d'approximation BV : BV-SG et BV-MG. Nous pouvons voir que ces deux reconstructions sont de qualité similaire que les images obtenues par les approches utilisant BVClass-SP, BVClass-ST et BV-Grad, qui sont montrées dans les figures 3.2 (d), (e) et (f). Théoriquement, la reconstruction de la méthode MCMC-Gibbs doit être la meilleure car elle converge asymptotiquement vers la vraie loi *a posteriori* plutôt que son approximation qui est l'objet des algorithmes d'approximation BV. Cependant, dans un temps limité, les échantillons obtenus par la méthode MCMC-Gibbs n'ont pas suffisamment convergés et donc ne donnent pas une bonne représentation de la loi *a posteriori*, ce qui explique la reconstruction de qualité relativement mauvaise montrée dans la figure 3.2 (c).

TABLE 3.1 – Comparaison de performance en termes de PSNR(dB)/Temps CPU (s).

Méthode	FBP	MCMC -Gibbs	BVClass -ST	BVClass -SP	BV-Grad	BV-SG	BV-MG
PSNR (dB)	27.32	28.83	35.12	35.24	35.90	35.84	35.92
Temps (s)	0.05	69313.8	723.5	327.6	23.4	6.8	3.2

En plus des images qui nous permettent d'avoir une comparaison de la qualité visuelle des reconstructions, nous montrons aussi dans la deuxième ligne du tableau 3.1 les valeurs de PSNR² des reconstructions obtenues, ce qui nous permet d'évaluer la qualité de reconstruction quantitativement. Nous pouvons voir que les cinq algorithmes d'approximation BV (BVClass-SP, BVClass-ST, BV-Grad, BV-SG, BV-MG) donnent des reconstructions de PSNR proche l'une de l'autre, qui sont plus grandes que les valeurs du PSNR données par FBP et la méthode MCMC-Gibbs, ce qui est cohérent avec la qualité visuelle des images montrées dans la figure 3.2.

La troisième ligne du tableau 3.1 montre les temps de calculs pris par toutes les méthodes implémentées. Nous pouvons voir que FBP est la plus rapide, comme nous avons déclaré précédemment. Cependant, FBP ne donne pas une reconstruction de qualité satisfaisante. Nous faisons donc ensuite une comparaison de la vitesse de convergence des six autres méthodes itératives. En bref, l'ordre de la vitesse de convergence de ces six algorithmes itératifs de la moins efficace à la plus efficace est :

$$\text{MCMC-Gibbs} < \text{BVClass-ST} < \text{BVClass-SP} < \text{BV-Grad} < \text{BV-SG} < \text{BV-MG}.$$

Nous pouvons voir dans le tableau 3.1 que nos deux algorithmes proposés BV-SG et BV-MG, notamment BV-MG, sont beaucoup plus efficaces que la méthode d'approximation BV-Grad, les méthodes d'approximation BV classiques BVClass-SP et BVClass-ST ainsi que la méthode MCMC-Gibbs. En fait, BV-MG a réussi à obtenir une reconstruction de la meilleure qualité (PSNR = 35.92dB) en prenant seulement 3.2 seconds, qui est

2. PSNR : Peak Signal to Noise Ratio.

3.4.4 - Résultats de simulation dans le cas non-supervisé

7 fois plus rapide que BV-Grad, plus de 100 fois plus rapide que les méthodes BVClass-SP et BVClass-ST, et 20000 fois plus rapide que la méthode MCMC-Gibbs. La méthode MCMC-Gibbs a pris 69313.8 seconds pour converger à une reconstruction de PSNR 28.83 dB pour ce problème de petite taille, ce qui montre que la méthode MCMC-Gibbs ne sera pas suffisamment efficace pour aborder des problèmes de grande taille.

L'autre algorithme proposé, BV-SG, est aussi beaucoup plus efficace que les algorithmes de l'état de l'art. Cependant, il est quand même moins efficace que BV-MG ce qui est explicable par le fait que le BV-MG utilise le sous-espace mémoire du gradient consistant en le gradient présent et la direction précédente, tandis que le BV-SG utilise le sous-espace du gradient consistant en le gradient présent et le gradient précédent qui est moins informatif que la direction précédente.

3.4.4 Résultats de simulation dans le cas non-supervisé

Comme montré dans la partie 3.3.3, les algorithmes de reconstruction supervisés où les hyperparamètres γ_n et γ_p sont choisis empiriquement : algorithme 4 et algorithme 5, ont été étendus dans le cadre entièrement bayésien, qui nous permet d'obtenir des algorithmes non-supervisés : algorithme 6 et algorithme 7. Les algorithmes non-supervisés estiment les valeurs des hyperparamètres γ_n et γ_p en même temps que \mathbf{x} et \mathbf{z} .

Nous avons appliqué ces algorithmes non-supervisés dans notre problème de tomographie. Comme dans les simulations précédentes, nous avons pris zéro comme la valeur initiale de la moyenne et un comme la valeur initiale de la variance pour toutes les composantes de \mathbf{x} . En ce qui concerne les initialisations des hyperparamètres γ_n et γ_p , nous avons pris 1 et 20 comme valeurs initiales.

Les images reconstruites sont résumées dans la figure 3.3. En comparant les images obtenues par les algorithmes supervisés (voir la figure 3.3 (a) et (b)) avec les images obtenues par les algorithmes non-supervisés (voir la figure 3.3 (c)-(d)), nous observons que les fonds des images dans les figures 3.3 (c) et (d) sont plus propres que ceux des images montrées dans les figures 3.3 (a) et (b) en reconstruisant les pics, ce qui montre que l'estimation des hyperparamètres permet de mettre plus de pixels à zéro, autrement dit, de renforcer la parcimonie de la reconstruction. Un défaut des reconstructions montrées par les figures 3.3 (c) et (d) est qu'elles ont deux pics faux. Malgré le défaut de pics faux, les PSNR des reconstructions obtenues par les algorithmes non-supervisés sont vers 3.6 dB plus grandes que les PSNR des reconstructions des algorithmes supervisés, comme montré dans le tableau 3.2. Dans ce cas, les algorithmes non-supervisés sont plus pertinents que les algorithmes supervisés.

TABLE 3.2 – Comparaison de performance en termes de PSNR(dB).

Méthode	BV-SG	BV-MG	NSBV-SG	NSBV-MG
PSNR(dB)	35.84	35.92	39.50	39.50

Nous montrons dans la figure 3.4 l'évolution des estimations des hyperparamètres en fonction du nombre d'itérations. Pour tracer ces courbes, nous avons utilisé une échelle logarithmique sur l'axe vertical pour avoir une représentation plus claire. En observant les courbes montrées dans les figures 3.4 (a) et (c), nous voyons que pour la variance du

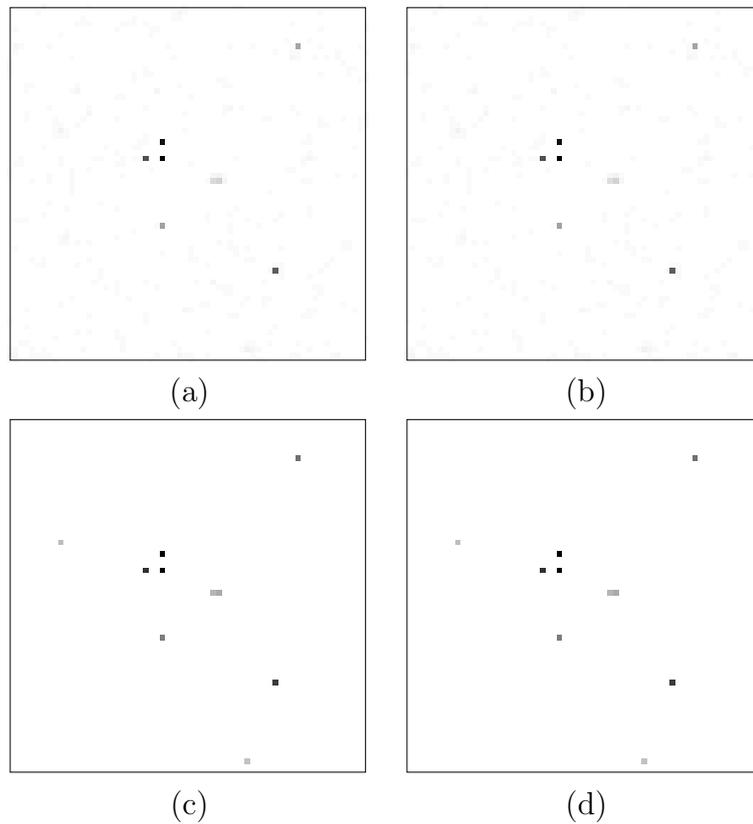


FIGURE 3.3 – Les images sont présentées avec le même niveau de gris inversé. Les algorithmes supervisés : (a) BV-SG, (b) BV-MG, les algorithmes non-supervisés : (c) NSBV-SG (algorithme 6), (d) NSBV-MG (algorithme 7).

bruit, les estimations convergent vers la valeur vraie (ligne en pointillé) pour soit NSBV-SG soit NSBV-MG. En ce qui concerne σ_p^2 , nous ne savons pas sa valeur vraie, mais les estimations convergent aussi.

Par ailleurs, si on compare les courbes obtenues par USBV-SG (figure 3.4 (a) et (b)) avec les courbes obtenues par USBV-MG (figure 3.4 (c) et (d)), nous pouvons voir que USBV-MG a besoin de moins d'itérations pour converger que USBV-SG, ce qui est cohérent avec les résultats obtenus dans le cas supervisé.

3.5 Conclusion

Nous avons proposé dans ce chapitre une nouvelle méthode d'approximation bayésienne variationnelle (BV) basée sur la méthode de sous-espace. En prenant deux différents sous-espaces : le sous-espace de gradient (SG) et le sous-espace de mémoire de gradient (MG), deux algorithmes d'approximation BV itératifs (BV-SG et BV-MG) ont été proposés. Nous avons montré expérimentalement sur un problème de tomographie de petite taille que les deux algorithmes proposés sont plus efficaces que les algorithmes de l'état de l'art. De plus, nous avons observé que l'algorithme BV-MG est plus efficace que celui basé sur le sous-espace SG (BV-SG). En conséquence, dans la suite de cette thèse, nous montrerons uniquement les résultats de l'algorithme BV-MG.

3.4.4 - Résultats de simulation dans le cas non-supervisé

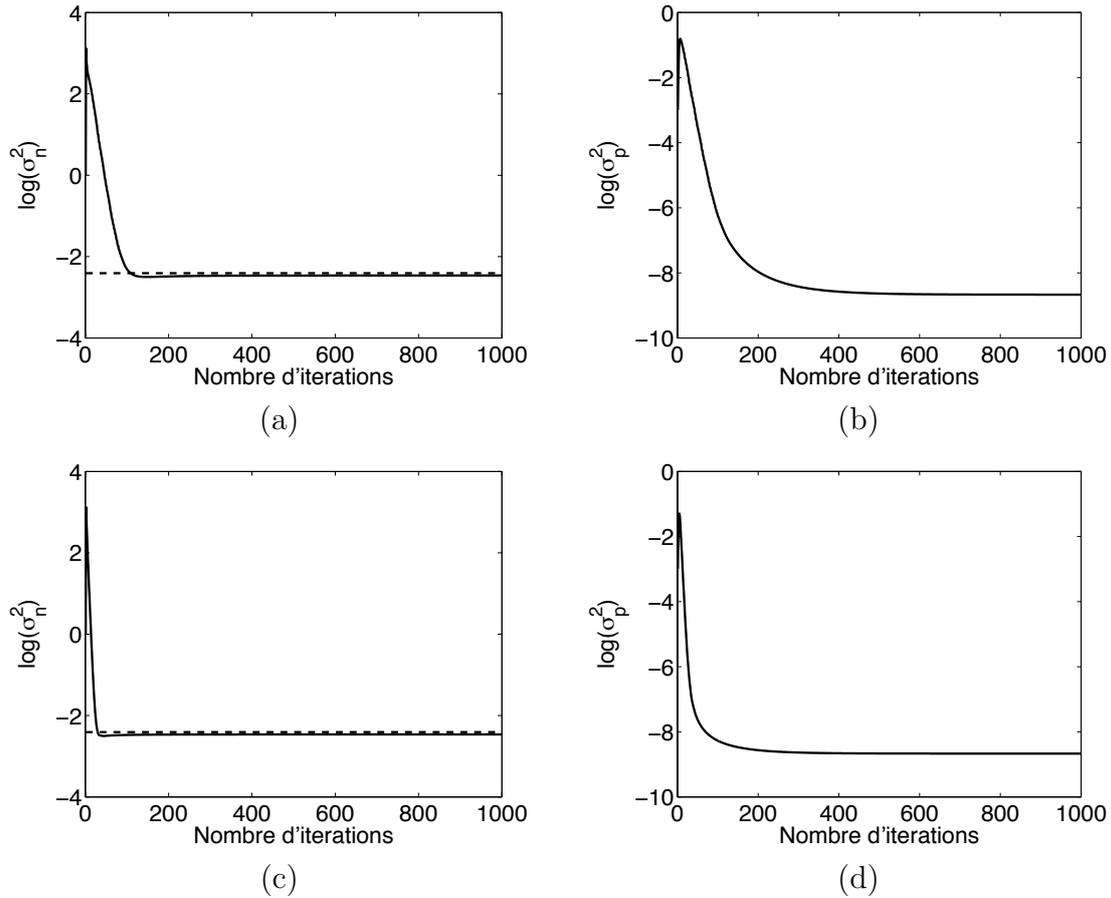


FIGURE 3.4 – Les évolutions des estimations des hyperparamètres. Les lignes en pointillé représentent les valeurs vraies des hyperparamètres. (a) Le logarithme de la variance du bruit ($\sigma_n^2 = 1/\gamma_n$) pour USBV-SG, (b) le logarithme de la variance de l'*a priori* ($\sigma_p^2 = 1/\gamma_p$) pour USBV-SG, (c) le logarithme de la variance du bruit ($\sigma_n^2 = 1/\gamma_n$) pour USBV-MG, (d) le logarithme de la variance de l'*a priori* ($\sigma_p^2 = 1/\gamma_p$) pour USBV-MG.

4

Applications aux problèmes inverses en traitement d'images

4.1 Introduction

L'objectif de ce chapitre est double : d'une part développer des approches bayésiennes non-supervisées en utilisant l'algorithme d'approximation BV-MG proposé dans le chapitre 3, d'autre part comparer les approches obtenues avec l'état de l'art afin d'évaluer l'efficacité de notre algorithme d'approximation BV-MG en termes de temps de calcul dans un problème de traitement d'images de grande taille.

Dans le contexte de problèmes inverses mal-posés en traitement d'images, un type d'informations *a priori* qu'on peut utiliser souvent est la régularité par morceau, qui est une caractéristique de nombreuses images. Pour introduire telle information, la variation totale a été beaucoup utilisée dans la littérature. La régularisation par variation totale (TV) a été premièrement introduite par Rudin *et al.* [ROF92] pour résoudre un problème de débruitage. Depuis, la variation totale a été largement utilisée dans diverses applications, telles que la déconvolution [VO98], la déconvolution aveugle [CW98, BMK09], inpainting [SC02], super-résolution [NSLZ07, BMK11], etc. L'avantage de la variation totale est son aptitude à préserver les contours tout en réduisant le bruit. Elle est donc bien adaptée aux images régulières par morceau.

Néanmoins, avec la loi *a priori* basée sur TV, l'estimation des hyperparamètres pose problème car la fonction de partition dépend de hyperparamètres mais cette fonction n'a pas une forme explicite. Ce problème peut être résolu grâce aux travaux menés par Bioucas-Dias *et al.* [BDF06, OBDF09] où une approximation analytique de la fonction de partition a été proposée. Néanmoins, lorsqu'on développe des approches non-supervisées à l'aide des algorithmes d'approximation BV, nous rencontrons une autre difficulté posée par le fait que la loi basée sur TV n'est pas une loi *a priori* conjuguée avec la vraisemblance gaussienne. Pour s'affranchir de cette difficulté, nous adoptons dans cette thèse la

technique MM (ici, Minoration-Maximisation) [HL04, BDF006], qui, à l'aide des variables auxiliaires, nous permet de construire une fonction minorante qui est conjuguée avec la vraisemblance [BMK08, BMK09, BMK11]. Grâce à l'approximation analytique de la fonction de partition et la technique MM, nous pouvons développer une approche bayésienne non-supervisée basée sur l'*a priori* TV en utilisant l'algorithme d'approximation BV-MG, ce qui est fait dans la partie 4.2.

Néanmoins, concernant l'*a priori* basé sur TV, le fait est que sa fonction de partition n'a pas d'expression explicite. Pour contourner ce problème, on cherche alors une autre loi permettant d'apporter des informations de régularité par morceau, et en même temps possédant une fonction de partition explicite. Dans ce contexte, nous avons considéré un modèle *a priori* à variables cachées qui est une extension du modèle proposé dans un travail récent de Giovannelli [Gio08]. Dans ce modèle, la loi du paramètre d'intérêt sachant les variables cachées est une loi gaussienne corrélée et les variables cachées suivent la loi de Laplace. Ici, les variables cachées correspondent à la moyenne de la différence entre deux pixels voisins. En supposant que les moyennes des différences sont parcimonieuses, cette loi à variables cachées permet de décrire les images régulières par morceau. Comme ce type de modèle a été inspiré par le modèle semi-quadratique de Geman *et al.* [GY95], nous allons l'appeler le modèle à variables cachées de type Geman dans la suite.

Dans la suite, nous allons construire deux approches non-supervisées basées sur ces deux modèles *a priori* en utilisant l'algorithme d'approximation BV-MG. Pour évaluer l'efficacité de notre algorithme BV-MG en termes de temps de calcul dans les problèmes en traitement d'images, nous appliquons les approches obtenues à un problème de super-résolution qui vise à reconstruire une image à haute résolution à partir de plusieurs images à basse résolution représentant la même scène. Néanmoins, dans ce travail, pour comparer les approches dans les problèmes de grande taille, nous considérons un problème de super-résolution linéaire où nous considérons que les mouvements entre les images à basse résolution et une image de référence pourrait soit être estimés à l'avance, soit être connus au travers d'autres sources d'information. Cette configuration apparaît par exemple dans les domaines de l'astronomie [ROGA08] et de l'imagerie médicale [GOKP02]. A travers un tel problème, nous effectuons des comparaisons de nos approches basées sur BV-MG avec les approches de l'état de l'art basées sur l'algorithme d'approximation BVClass et l'algorithme d'approximation BV-Grad. De plus, nous offrons des comparaisons des deux approches proposées qui ont utilisé deux modèles *a priori* différents.

4.2 Approche non-supervisée basée sur un *a priori* de TV

Nous nous intéressons toujours dans ce chapitre au problème inverse linéaire. L'objectif est de résoudre ce problème dans un cadre entièrement bayésien. Pour ce faire, nous introduisons des lois *a priori* pour l'objet d'intérêt ainsi que pour les hyperparamètres. Dans cette partie, nous introduisons la loi *a priori* basée sur la variation totale pour favoriser des images régulières par morceau. En ce qui concerne les lois *a priori* des hyperparamètres, nous prenons des lois de Jeffreys comme dans les chapitres précédents.

4.2.1 Formulation bayésienne

Loi *a priori* pour l'objet d'intérêt

La loi *a priori* basée sur la variation totale (TV) prend la forme suivante

$$p(\mathbf{x}|\gamma_p) = \frac{1}{Z_{TV}(\gamma_p)} \exp[-\gamma_p TV(\mathbf{x})], \quad (4.1)$$

où $Z_{TV}(\gamma_p)$ est la fonction de partition qui est une fonction de γ_p et

$$TV(\mathbf{x}) = \sum_{i=1}^N \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2}. \quad (4.2)$$

Ici \mathbf{D}_h et \mathbf{D}_v représentent les matrices des différences finies d'ordre un dans la direction horizontale et verticale, respectivement.

La fonction de partition est donnée par

$$Z_{TV}(\gamma_p) = \int_{\mathbb{R}^N} \exp \left[-\gamma_p \sum_{i=1}^N \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2} \right] d\mathbf{x}. \quad (4.3)$$

Néanmoins, cette intégrale est infinie. Pour résoudre ce problème, en pratique, on peut ajouter un terme tendant vers zéro pour avoir une intégrale finie :

$$Z_{TV}(\gamma_p) = \int_{\mathbb{R}^N} \exp \left[-\gamma_p \sum_{i=1}^N \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \epsilon x_i^2} \right] d\mathbf{x}. \quad (4.4)$$

où $\epsilon \rightarrow 0$. Le terme ϵx_i^2 peut être fusionné dans les deux termes $(\mathbf{D}_h \mathbf{x})_i^2$ et $(\mathbf{D}_v \mathbf{x})_i^2$ en ajoutant un coefficient tendant vers zéro aux opérateurs des différences finies : \mathbf{D}_h et \mathbf{D}_v . Néanmoins, la difficulté est que même dans ce cas, nous n'avons pas d'expression explicite pour $Z_{TV}(\gamma_p)$. La détermination de la fonction de partition d'une telle loi est un problème ouvert. Dans ce travail, pour s'affranchir de cette difficulté, nous avons utilisé la fonction de partition approchée proposée par Bioucas-Dias *et al.* [BDF06, ODBF09]. Cette fonction de partition approchée est basée sur l'hypothèse que chaque paire de différence finie $\{(\mathbf{D}_h \mathbf{x})_i, (\mathbf{D}_v \mathbf{x})_i\}$ est indépendante de toutes les autres paires. Cette hypothèse n'est jamais vérifiée en pratique. Notons que

$$\int_{\mathbb{R}^2} \exp \left(-\lambda \sqrt{u^2 + v^2} \right) dudv = \frac{2\pi}{\lambda^2}. \quad (4.5)$$

Basée sur l'hypothèse d'indépendance indiquée ci-dessus, on peut obtenir que

$$Z_{TV}(\gamma_p) = \int_{\mathbb{R}^N} \exp \left[-\gamma_p \sum_{i=1}^N \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2} \right] d\mathbf{x} \simeq C \gamma_p^{-\theta N}, \quad (4.6)$$

où C est une constante indépendante de γ_p . L'hypothèse d'indépendance conduit au résultat ci dessus avec $\theta = 2$. Néanmoins, en raison de la dépendance qui existe en effet entre les différences finies, θ est considéré comme un paramètre à régler. Il permet

4.2.2 - Application des méthodes d'approximation BV

en pratique d'obtenir de meilleurs résultats. En utilisant cette approximation analytique, nous pouvons obtenir

$$p(\mathbf{x}|\gamma_p) \simeq \tilde{p}(\mathbf{x}|\gamma_p) = \tilde{C}\gamma_p^{\theta N} \exp[-\gamma_p TV(\mathbf{x})] \quad (4.7)$$

avec $\tilde{C} = C^{-1}$.

Dans les travaux récents menés par Babacan *et al.* [BMK08, BMK09, BMK11], cette loi TV approchante a été employée avec le paramètre $\theta = 1/2$. Dans ce cas, la fonction de partition $Z_{TV}(\gamma_p)$ est approchée par $C\gamma_p^{-N/2}$ qui correspond à la fonction de partition d'une distribution gaussienne multivariée d'un vecteur de dimension N . Néanmoins, la loi *a priori* TV n'est pas similaire à une distribution gaussienne. Donc, cette approximation est, en général, pas suffisamment proche de la vraie fonction de partition. Par conséquent, dans notre travail, nous gardons ce paramètre θ et le réglons empiriquement afin d'améliorer les résultats.

Loi *a posteriori*

Dans notre cas, la loi *a posteriori* s'écrit comme suit :

$$\begin{aligned} p(\mathbf{x}, \gamma_n, \gamma_p | \mathbf{y}) &\propto p(\mathbf{y} | \mathbf{x}, \gamma_n) p(\mathbf{x} | \gamma_p) p(\gamma_n) p(\gamma_p) \\ &\tilde{\propto} \gamma_n^{M/2} \exp\left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2}\right] \gamma_p^{\theta N} \exp\left[-\gamma_p \sum_{i=1}^N \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2}\right] \\ &\times \gamma_n^{\tilde{a}_n - 1} \exp[-\tilde{b}_n \gamma_n] \gamma_p^{\tilde{a}_p - 1} \exp[-\tilde{b}_p \gamma_p]. \end{aligned} \quad (4.8)$$

où $\tilde{\propto}$ signifie "est approximativement proportionnel à". La loi *a posteriori* $p(\mathbf{x}, \gamma_n, \gamma_p | \mathbf{y})$ n'est pas connue explicitement car sa constante de normalisation n'est pas calculable. Afin de procéder à l'inférence statistique des variables inconnues à partir de la loi *a posteriori*, nous recourons aux méthodes d'approximation BV.

4.2.2 Application des méthodes d'approximation BV

Soit Θ l'ensemble des paramètres inconnus : $\Theta = \{\mathbf{x}, \gamma_n, \gamma_p\}$. L'objectif de l'approximation BV est de donner une approximation q_Θ de la loi *a posteriori* $p(\cdot | \mathbf{y})$ en maximisant l'énergie libre négative \mathcal{F} dont la définition est rappelée par la suite :

$$\mathcal{F}(q_\Theta) = \int q_\Theta(\Theta) \ln \frac{p(\mathbf{y}, \Theta)}{q_\Theta(\Theta)} d\Theta. \quad (4.9)$$

Néanmoins, comme la loi *a priori* TV n'est pas une loi *a priori* conjuguée avec la vraisemblance gaussienne, il est difficile d'effectuer la maximisation de l'énergie libre négative par rapport à q_Θ . Pour s'affranchir de cette difficulté, nous utilisons la technique Minoration-Maximisation (MM) [HL04, BDF06] qui nous permet de remplacer la maximisation de l'énergie libre négative par la maximisation d'une borne inférieure de cette énergie qui est plus facile à manipuler [BMK08, BMK09, BMK11]. Une telle borne

peut être obtenue de la façon suivante. L'inégalité arithmético-géométrique nous permet d'obtenir :

$$\forall a > 0, b > 0 \quad \sqrt{ab} \leq \frac{a+b}{2} \Rightarrow \sqrt{a} \leq \frac{a+b}{2\sqrt{b}}. \quad (4.10)$$

En utilisant cette inégalité, nous pouvons construire une inégalité comme suit en introduisant des variables auxiliaires positives $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_N]$,

$$\forall \lambda_i > 0, i = 1, \dots, N \quad \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2} \leq \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \lambda_i}{2\sqrt{\lambda_i}}, \quad (4.11)$$

qui nous permet d'obtenir une borne inférieure de la loi *a priori* TV, notée $M(\mathbf{x}, \gamma_p | \boldsymbol{\lambda})$ par :

$$\begin{aligned} \tilde{p}(\mathbf{x} | \gamma_p) &= \tilde{C} \gamma_p^{\theta N} \exp \left[-\gamma_p \sum_{i=1}^N \sqrt{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2} \right] \\ &\geq \tilde{C} \gamma_p^{\theta N} \exp \left[-\gamma_p \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \lambda_i}{2\sqrt{\lambda_i}} \right] \\ &= M(\mathbf{x}, \gamma_p | \boldsymbol{\lambda}) \end{aligned} \quad (4.12)$$

Dans (4.12), nous pouvons voir que la borne inférieure de la loi TV, $M(\mathbf{x}, \gamma_p | \boldsymbol{\lambda})$, est proportionnelle à une distribution gaussienne et est donc conjuguée avec la vraisemblance. Cette borne inférieure peut être ensuite utilisée pour construire une borne inférieure de la distribution jointe :

$$\begin{aligned} p(\mathbf{y}, \Theta) &\geq p(\mathbf{y} | \Theta) M(\mathbf{x}, \gamma_p | \boldsymbol{\lambda}) p(\gamma_n) p(\gamma_p) \\ &= L(\Theta, \mathbf{y} | \boldsymbol{\lambda}) \\ &\propto \gamma_n^{M/2} \exp \left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A} \mathbf{x}\|^2}{2} \right] \gamma_p^{\theta N} \exp \left[-\gamma_p \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \lambda_i}{2\sqrt{\lambda_i}} \right] \\ &\quad \times \gamma_n^{\tilde{a}_n - 1} \exp \left[-\tilde{b}_n \gamma_n \right] \gamma_p^{\tilde{a}_p - 1} \exp \left[-\tilde{b}_p \gamma_p \right] \end{aligned} \quad (4.13)$$

qui nous permet d'obtenir facilement une borne inférieure de l'énergie libre négative :

$$\begin{aligned} \mathcal{F}(q_\Theta) &\geq \mathcal{F}^L(q_\Theta, \boldsymbol{\lambda}) \\ &= \int q_\Theta(\Theta) \ln \left(\frac{L(\Theta, \mathbf{y} | \boldsymbol{\lambda})}{q_\Theta(\Theta)} \right) d\Theta. \end{aligned} \quad (4.14)$$

La maximisation de l'énergie libre négative \mathcal{F} peut être donc effectuée en alternant deux étapes : une étape de maximisation de la borne inférieure \mathcal{F}^L par rapport à la distribution q_Θ et une étape de mise à jour des variables auxiliaires $\boldsymbol{\lambda}$ afin de maximiser \mathcal{F}^L .

En ce qui concerne la distribution approchante q_Θ , elle est supposée séparable. Comme nous l'avons expliqué auparavant, nous prenons une hypothèse de séparabilité totale pour éviter les inversions matricielles qui sont trop coûteuses dans les problèmes de grande

4.2.2 - Application des méthodes d'approximation BV

taille. L'hypothèse de séparabilité totale entraîne que

$$\begin{aligned} q_{\Theta}(\Theta) &= q_{\mathbf{x}}(\mathbf{x})q_{\gamma_n}(\gamma_n)q_{\gamma_p}(\gamma_p) \\ &= \prod_i q_i(x_i)q_{\gamma_n}(\gamma_n)q_{\gamma_p}(\gamma_p). \end{aligned} \quad (4.15)$$

Sous cette hypothèse, nous menons une optimisation alternée de \mathcal{F}^L par rapport à $q_{\mathbf{x}}$, q_{γ_n} , q_{γ_p} et $\boldsymbol{\lambda}$. En bref, nous résolvons les problèmes suivants alternativement ¹ :

$$q_{\mathbf{x}}^{k+1} = \arg \max_{q_{\mathbf{x}}} \mathcal{F}^L \left(q_{\mathbf{x}} q_{\gamma_n}^k q_{\gamma_p}^k, \boldsymbol{\lambda}^k \right), \quad (4.16)$$

$$\boldsymbol{\lambda}^{k+1} = \arg \max_{\boldsymbol{\lambda} \in \mathbb{R}^N} \mathcal{F}^L \left(q_{\mathbf{x}}^{k+1} q_{\gamma_n}^k q_{\gamma_p}^k, \boldsymbol{\lambda} \right) \quad (4.17)$$

$$q_{\gamma_n}^{k+1} = \arg \max_{q_{\gamma_n}} \mathcal{F}^L \left(q_{\mathbf{x}}^{k+1} q_{\gamma_n} q_{\gamma_p}^k, \boldsymbol{\lambda}^{k+1} \right), \quad (4.18)$$

$$q_{\gamma_p}^{k+1} = \arg \max_{q_{\gamma_p}} \mathcal{F}^L \left(q_{\mathbf{x}}^{k+1} q_{\gamma_n}^{k+1} q_{\gamma_p}, \boldsymbol{\lambda}^{k+1} \right) \quad (4.19)$$

Les optimisations fonctionnelles par rapport à $q_{\mathbf{x}}$, q_{γ_n} et q_{γ_p} (données par (4.16), (4.18) et (4.19)) sont faites à l'aide des méthodes d'approximation BV. Comme la loi *a posteriori* conditionnelle $p(\gamma_n, \gamma_p | \mathbf{x}, \mathbf{y})$ est séparable elle même, elle peut être approchée de façon efficace grâce à la méthode d'approximation BV classique. Pour l'optimisation de $q_{\mathbf{x}}$, nous employons l'algorithme BV-MG proposé dans le chapitre 3. En ce qui concerne l'optimisation des variables auxiliaires $\boldsymbol{\lambda}$ (donnée par (4.17)), il s'agit d'un problème d'optimisation classique dans un espace de Hilbert.

Grâce à la technique de MM, nous pouvons obtenir une loi *a priori* de \mathbf{x} qui est conjuguée avec la vraisemblance. En outre, des lois Gamma sont utilisées comme les lois *a priori* conjuguées pour les hyperparamètres γ_n et γ_p . Par conséquent, l'approximation optimale $q_{\mathbf{x}}^{opt} = \prod_{i=1}^N (q_i^{opt})$ appartient à la famille gaussienne tandis que $q_{\gamma_n}^{opt}$ et $q_{\gamma_p}^{opt}$ appartiennent à la famille Gamma. Nous prenons donc

$$q_{\mathbf{x}}^k(\mathbf{x}) = \prod_i \mathcal{N}(x_i | (\mathbf{m}_k)_i, (\boldsymbol{\sigma}_k^2)_i), \quad (4.20)$$

$$q_{\gamma_n}^k(\gamma_n) = \mathcal{G}(\gamma_n | a_{\gamma_n}^k, b_{\gamma_n}^k), \quad (4.21)$$

$$q_{\gamma_p}^k(\gamma_p) = \mathcal{G}(\gamma_p | a_{\gamma_p}^k, b_{\gamma_p}^k), \quad (4.22)$$

L'optimisation de ces lois approchantes peut être donc réalisée uniquement par la mise à jour de leurs paramètres.

Optimisation de $q_{\mathbf{x}}$ avec l'algorithme d'approximation BV-MG

Comme donné par l'Eq. (4.16), au lieu de maximiser l'énergie libre \mathcal{F} , nous maximisons sa borne inférieure \mathcal{F}^L qui peut être écrite comme

$$\mathcal{F}^L(q_{\Theta}) = \langle \ln L(\cdot, \mathbf{y} | \boldsymbol{\lambda}) \rangle_{q_{\Theta}} + \mathcal{H}(q_{\Theta}). \quad (4.23)$$

1. Les variables auxiliaires $\boldsymbol{\lambda}$ sont mises à jour avant q_{γ_n} et q_{γ_p} pour obtenir une équation de mise à jour plus simple pour q_{γ_p} , voir [BMK11].

En comparant (4.23) avec (2.43), nous pouvons voir que \mathcal{F}^L a une structure comparable à \mathcal{F} . En fait, on peut obtenir \mathcal{F}^L à partir de \mathcal{F} en remplaçant la loi jointe par sa borne inférieure $L(\Theta, \mathbf{y}|\boldsymbol{\lambda})$. Par conséquent, nous pouvons définir une fonction $df^L(q_{\mathbf{x}}^k, \mathbf{x})$ pour la fonctionnelle \mathcal{F}^L . Nous obtenons la fonction $df^L(q_{\mathbf{x}}^k, \mathbf{x})$ de la même façon que $df(q_{\mathbf{x}}^k, \mathbf{x})$ (voir partie 2.4). Comme $q_{\mathbf{x}}$ est supposée séparable, on peut obtenir que $df^L(q_{\mathbf{x}}^k, \mathbf{x}) = \sum_i df^L(q_i^k, x_i)$ avec

$$\forall i = 1, \dots, N \quad d_i f^L(q_i, x_i) = \langle \ln L(\Theta, \mathbf{y}|\boldsymbol{\lambda}) \rangle_{\prod_{j \neq i} q_j q_{\gamma_n} q_{\gamma_p}} - \ln q_i(x_i) - 1. \quad (4.24)$$

Dans ce cas, le sous-espace de mémoire de gradient est donné par :

$$\mathbf{D}_{MG}^k(\mathbf{x}) = [df^L(q_{\mathbf{x}}^k, \mathbf{x}), d^{k-1}(\mathbf{x})]. \quad (4.25)$$

Nous déduisons de (3.12), (3.15), (4.24) et (4.25) une distribution de \mathbf{x} qui dépend du pas \mathbf{s} :

$$\begin{aligned} q_{\mathbf{x}}^{\mathbf{s}}(\mathbf{x}) &= K^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \exp(s_1 df^L(q_{\mathbf{x}}^k, \mathbf{x}) + s_2 d^{k-1}(\mathbf{x})) \\ &= K^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \exp\left(s_1 \sum_i d_i f^L(q_i^k, x_i) + s_2 d^{k-1}(\mathbf{x})\right) \\ &= \tilde{K}^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \prod_i \left[\frac{\exp\left(\langle \ln L(\Theta, \mathbf{y}|\boldsymbol{\lambda}^k) \rangle_{(\prod_{j \neq i} q_j^k) q_{\gamma_n}^k q_{\gamma_p}^k}\right)}{q_i^k(x_i)} \right]^{s_1} \left[\frac{q_i^k(x_i)}{q_i^{k-1}(x_i)} \right]^{s_2} \\ &= \tilde{K}^k(\mathbf{s}) q_{\mathbf{x}}^k(\mathbf{x}) \prod_i \left(\frac{q_i^{\gamma_1}(x_i)}{q_i^k(x_i)} \right)^{s_1} \left(\frac{q_i^k(x_i)}{q_i^{k-1}(x_i)} \right)^{s_2} \end{aligned} \quad (4.26)$$

où $q_i^{\gamma_1}(x_i) = \exp\left(\langle \ln L(\Theta, \mathbf{y}|\boldsymbol{\lambda}^k) \rangle_{(\prod_{j \neq i} q_j^k) q_{\gamma_n}^k q_{\gamma_p}^k}\right)$ est la fonction intermédiaire introduite pour clarifier les expressions.

Pour obtenir l'expression de $q_{\mathbf{x}}^{\mathbf{s}}$, nous calculons d'abord cette fonction intermédiaire :

$$\begin{aligned} q_i^{\gamma_1}(x_i) &= \exp \left[\langle \ln L(\mathbf{x}, \gamma_p, \gamma_n, \mathbf{y}|\boldsymbol{\lambda}^k) \rangle_{(\prod_{j \neq i} q_j^k) q_{\gamma_n}^k q_{\gamma_p}^k} \right] \\ &\propto \exp \left[- \int \left(\frac{\gamma_n}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + \gamma_p \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \lambda_i^k}{2\sqrt{\lambda_i^k}} \right) \right. \\ &\quad \times \left. \left(\prod_{j \neq i} q_j^k(x_j) dx_j \right) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p) d\gamma_n d\gamma_p \right] \\ &\propto \exp \left[- \frac{\langle \gamma_n \rangle^k}{2} \left(x_i^2 \text{diag}(\mathbf{A}^T \mathbf{A})_i - 2x_i (\mathbf{A}^T \mathbf{y})_i \right. \right. \\ &\quad \left. \left. + 2x_i (\mathbf{A}^T \mathbf{A} \mathbf{m}_k)_i - 2x_i \text{diag}(\mathbf{A}^T \mathbf{A})_i (\mathbf{m}_k)_i \right) \right. \\ &\quad \left. - \frac{\langle \gamma_p \rangle^k}{2} \left(x_i^2 \text{diag}(\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v)_i \right. \right. \\ &\quad \left. \left. + 2x_i (\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h \mathbf{m}_k + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v \mathbf{m}_k)_i \right. \right. \\ &\quad \left. \left. - 2x_i \text{diag}(\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v)_i (\mathbf{m}_k)_i \right) \right] \end{aligned} \quad (4.27)$$

4.2.2 - Application des méthodes d'approximation BV

où $\Lambda^k = \text{Diag}\left(\frac{1}{\sqrt{\lambda_i^k}}\right)$ est une matrice diagonale avec $\left(\frac{1}{\sqrt{\lambda_i^k}}\right)_{i=1,\dots,N}$ comme éléments diagonaux. Les détails de calculs pour obtenir (4.27) peuvent être trouvés en annexe A.2.

Les calculs ci-dessus montrent que chaque fonction intermédiaire $(q_i^{T1})_{i=1,\dots,N}$ correspond, à une constante de normalisation près, à la densité d'une distribution gaussienne dont la moyenne $(\mathbf{m}_{r1})_i$ et la variance $(\sigma_{r1}^2)_i$ sont données explicitement par les deux expressions suivantes :

$$(\sigma_{r1}^2)_i = [\langle \gamma_n \rangle^k \text{diag}(\mathbf{A}^T \mathbf{A})_i + \langle \gamma_p \rangle^k \text{diag}(\mathbf{D}_h^T \Lambda^k \mathbf{D}_h + \mathbf{D}_v^T \Lambda^k \mathbf{D}_v)_i]^{-1}, \quad (4.28)$$

$$\begin{aligned} (\mathbf{m}_{r1})_i = (\sigma_{r1}^2)_i \left[\langle \gamma_n \rangle^k (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_k + \text{diag}(\mathbf{A}^T \mathbf{A}) \circ \mathbf{m}_k)_i \right. \\ \left. - \langle \gamma_p \rangle^k (\mathbf{D}_h^T \Lambda^k \mathbf{D}_h \mathbf{m}_k + \mathbf{D}_v^T \Lambda^k \mathbf{D}_v \mathbf{m}_k)_i \right. \\ \left. + \langle \gamma_p \rangle^k \text{diag}(\mathbf{D}_h^T \Lambda^k \mathbf{D}_h + \mathbf{D}_v^T \Lambda^k \mathbf{D}_v)_i (\mathbf{m}_k)_i \right]. \end{aligned} \quad (4.29)$$

En ce qui concerne $q_{\mathbf{x}}^s$, les calculs sont les mêmes que les calculs montrés dans le chapitre 3. On a trouvé que $q_{\mathbf{x}}^s(\mathbf{x}) = \prod_i q_i^s(x_i)$ où chaque composante séparable $q_i^s(x_i)$ est une distribution gaussienne dont la moyenne et la variance ont les mêmes expressions que (3.47) et (3.48) et nous les rappelons par la suite :

$$\sigma_{\mathbf{s}}^2 = \left[\frac{1}{\sigma_k^2} + s_1 \left(\frac{1}{\sigma_{r1}^2} - \frac{1}{\sigma_k^2} \right) + s_2 \left(\frac{1}{\sigma_k^2} - \frac{1}{\sigma_{k-1}^2} \right) \right]^{-1}, \quad (4.30)$$

$$\mathbf{m}_{\mathbf{s}} = \sigma_{\mathbf{s}}^2 \left[\frac{\mathbf{m}_k}{\sigma_k^2} + s_1 \left(\frac{\mathbf{m}_{r1}}{\sigma_{r1}^2} - \frac{\mathbf{m}_k}{\sigma_k^2} \right) + s_2 \left(\frac{\mathbf{m}_k}{\sigma_k^2} - \frac{\mathbf{m}_{k-1}}{\sigma_{k-1}^2} \right) \right]. \quad (4.31)$$

Dans les équations ci-dessus, nous avons omis les indices de composante $(\cdot)_i$ pour avoir des expressions plus claires. En fait, les formes de $\sigma_{\mathbf{s}}^2$ et $\mathbf{m}_{\mathbf{s}}$ dépendent seulement du sous-espace utilisé. Les modifications des lois (vraisemblance ou les lois *a priori*) changent seulement l'expression de la fonction intermédiaire q_i^{T1} .

Les distributions obtenues dépendent du pas vectoriel \mathbf{s} . Nous prenons un pas sous-optimal défini par (3.20) (voir l'annexe B.2 pour plus de détails de calculs). Par conséquent, $\sigma_{k+1}^2 = \sigma_{\hat{\mathbf{s}}_{subopt}}^2$ et $\mathbf{m}_{k+1} = \mathbf{m}_{\hat{\mathbf{s}}_{subopt}}$.

Optimisation de λ dans les espaces de Hilbert

Les composantes du vecteur auxiliaire λ sont calculées en maximisant la borne inférieure \mathcal{F}^L par rapport à $(\lambda_i)_{i=1,\dots,N}$. La fonction \mathcal{F}^L est une fonction différentiable de $(\lambda_i)_{i=1,\dots,N}$ et son maximum est atteint au point critique qui est donné par

$$\begin{aligned} \lambda_i^{k+1} &= \mathbb{E}_{q_{\mathbf{x}}^{k+1}} [(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2] \\ &= \mathbb{E}_{q_{\mathbf{x}}^{k+1}} [\mathbf{x}^T (\mathbf{D}_h)_i^T (\mathbf{D}_h)_i \mathbf{x} + \mathbf{x}^T (\mathbf{D}_v)_i^T (\mathbf{D}_v)_i \mathbf{x}] \\ &= (\mathbf{D}_h \mathbf{m}_{k+1})_i^2 + (\mathbf{D}_v \mathbf{m}_{k+1})_i^2 + \text{trace} [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i \Sigma_{k+1}] \\ &\quad + \text{trace} [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i \Sigma_{k+1}], \end{aligned} \quad (4.32)$$

où $(\mathbf{D}_h)_i$ et $(\mathbf{D}_v)_i$ représentent la i ème ligne de \mathbf{D}_h et \mathbf{D}_v , respectivement et $\Sigma_{k+1} = \text{Diag}((\sigma^2)^{k+1})$ est la matrice de covariance de $q_{\mathbf{x}}^{k+1}$ qui est diagonale sous notre hypothèse

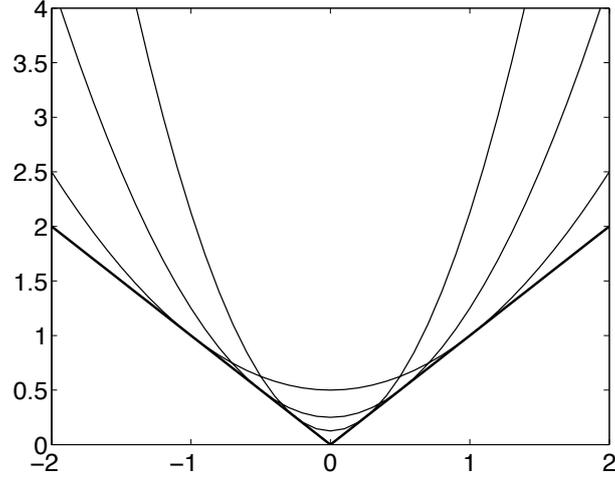


FIGURE 4.1 – Illustration de la procédure MM dans le cas TV.

La courbe en gras représente $|x|$ et les trois courbes au-dessus représentent les fonctions majorantes de $|x|$: $\frac{x^2+\lambda}{2\sqrt{\lambda}}$ où $\lambda > 0$.

de séparabilité totale. Les détails de calculs pour obtenir (4.32) peuvent être trouvés en annexe C.1.

Remarque. Cette étape de *Minoration-Maximisation* ressemble à l’approche *semi-quadratique* de Geman et Reynolds : la maximisation de notre critère est substituée par la maximisation d’une suite des fonctions majorantes dont la variance change avec la valeur de la variable auxiliaire λ_i et la moyenne reste inchangée, comme on peut voir par la figure 4.1.

Optimisation de q_{γ_n} et q_{γ_p}

L’optimisation de q_{γ_n} et q_{γ_p} se fait en utilisant l’algorithme d’approximation BV classique fondée sur l’expression (2.12).

Les calculs de q_{γ_n} sont identiques que ceux montrés par (3.59). En conséquence, les équations de mise à jour pour ses paramètres sont aussi identiques que (3.60) et (3.61).

Pour q_{γ_p} , en utilisant (2.12), nous pouvons obtenir

$$q_{\gamma_p}^{k+1}(\gamma_p) \propto \gamma_p^{\theta N + \tilde{a}_p - 1} \exp \left[-\gamma_p \left(\tilde{b}_p + \sum_{i=1}^N \frac{\mathbb{E}_{q_{\mathbf{x}}^{k+1}} [(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2] + \lambda_i^{k+1}}{2\sqrt{\lambda_i^{k+1}}} \right) \right]. \quad (4.33)$$

En combinant (4.33) et (4.32), l’expression ci-dessus peut être simplifiée et nous avons donc

$$q_{\gamma_p}^{k+1}(\gamma_p) \propto \gamma_p^{\theta N + \tilde{a}_p - 1} \exp \left[-\gamma_p \left(\tilde{b}_p + \sum_{i=1}^N \sqrt{\lambda_i^{k+1}} \right) \right], \quad (4.34)$$

4.2.3 - Des approches de l'état de l'art

qui est une distribution Gamma dont les paramètres sont identifiés comme

$$a_{\gamma_p}^{k+1} = \theta N + \tilde{a}_p = a_{\gamma_p} \quad (4.35)$$

$$b_{\gamma_p}^{k+1} = \tilde{b}_p + \sum_{i=1}^N \sqrt{\lambda_i^{k+1}}. \quad (4.36)$$

En bref, notre algorithme non-supervisé basé sur la loi *a priori* TV en utilisant l'algorithme d'approximation BV-MG est résumé dans l'algorithme 8.

Algorithm 8 L'algorithme de reconstruction non-supervisé basé sur un *a priori* TV en utilisant BV-MG

1. Initialiser les paramètres de $(q_i^0)_{i=1,\dots,N}$, $q_{\gamma_n}^0$, $q_{\gamma_p}^0$ et λ^0
 2. Calculer $q_{\mathbf{x}}^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (4.28)-(4.29)
 - (b) Calculer le pas sous-optimal selon annexe B.2
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (4.30) et (4.31) où le pas \mathbf{s} est égal au pas sous-optimal obtenu à l'étape (b)
 3. Calculer le vecteur auxiliaire λ^{k+1} en utilisant (4.32)
 4. Calculer $q_{\gamma_n}^{k+1}$: calculer ses paramètres en utilisant (3.60) et (3.61)
 5. Calculer $q_{\gamma_p}^{k+1}$: calculer ses paramètres en utilisant (4.35) et (4.36)
 6. Retourner à l'étape 2 jusqu'à convergence
-

4.2.3 Des approches de l'état de l'art

Pour évaluer notre approche, nous avons choisi de comparer avec deux approches non-supervisées utilisant deux algorithmes d'approximation BV de l'état de l'art : l'une approche de Babacan *et al.* [BMK11] utilisant l'algorithme d'approximation BV classique et l'autre approche utilisant l'algorithme d'approximation BV-Grad [FR14].

Approche basée sur l'algorithme d'approximation BV classique

La résolution d'un problème inverse linéaire avec un *a priori* TV a été récemment traitée par Babacan *et al.* [BMK11]. Leur approche résout le problème dans un cadre entièrement bayésien à l'aide de l'approximation BV. Bien que notre approche et l'approche de Babacan *et al.* soient très proches, il existe deux différences importantes. La première est que Babacan *et al.* utilise l'approximation BV classique pour l'optimisation de $q_{\mathbf{x}}$ tandis que nous avons appliqué l'algorithme d'approximation BV-MG proposé dans le chapitre 3. La deuxième est que nous prenons une hypothèse de séparabilité totale qui entraîne que $q_{\mathbf{x}}$ est séparable tandis que Babacan *et al.* prennent une hypothèse de séparabilité partielle où $q_{\mathbf{x}}$ est supposée non-séparable, i.e.

$$q_{\Theta}(\Theta) = q_{\mathbf{x}}(\mathbf{x})q_{\gamma_n}(\gamma_n)q_{\gamma_p}(\gamma_p). \quad (4.37)$$

Sous cette hypothèse, l'algorithme d'approximation BV classique génère une distribution gaussienne multivariée pour $q_{\mathbf{x}}^k$:

$$q_{\mathbf{x}}^k = \mathcal{N}(\mathbf{x} | \mathbf{m}_k, \Sigma_k), \quad (4.38)$$

où \mathbf{m}_k représente la moyenne et Σ_k est la matrice de covariance.

En utilisant la formule (2.12), nous pouvons obtenir

$$q_{\mathbf{x}}^{k+1} \propto \exp \left[-\frac{\langle \gamma_n \rangle^k}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 - \frac{\langle \gamma_p \rangle^k}{2} \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2}{\sqrt{\lambda_i}} \right], \quad (4.39)$$

à partir de laquelle nous identifions

$$\mathbf{m}_{k+1} = \Sigma_{k+1} [\langle \gamma_n \rangle^k \mathbf{A}^T \mathbf{y}], \quad (4.40)$$

$$(\Sigma_{k+1})^{-1} = \langle \gamma_n \rangle^k \mathbf{A}^T \mathbf{A} + \langle \gamma_p \rangle^k (\mathbf{D}_h^T \Lambda^k \mathbf{D}_h + \mathbf{D}_v^T \Lambda^k \mathbf{D}_v). \quad (4.41)$$

L'équation (4.40) montre que la moyenne \mathbf{m}_{k+1} dépend de la matrice de covariance Σ_{k+1} , cependant le calcul de Σ_{k+1} nécessite d'inverser la matrice $(\Sigma_{k+1})^{-1}$ donnée par (4.41). Néanmoins, l'inversion est une opération coûteuse et son coût de calcul croît rapidement avec l'augmentation de la taille de la matrice à inverser. Pour contourner les calculs d'inversion matricielle dans le calcul de \mathbf{m}_{k+1} , Babacan *et al.* ont employé la méthode du gradient conjugué pour approcher \mathbf{m}_{k+1} itérativement.

En ce qui concerne la variable auxiliaire λ ainsi que les lois approchantes de hyperparamètres, q_{γ_n} et q_{γ_p} , leur optimisation se fait de la même façon que notre approche. Par contre, comme on peut voir dans l'Eq. (4.32), le calcul des variables auxiliaires nécessite de connaître la matrice de covariance Σ_{k+1} . Ici, pour contourner les inversions matricielles, Babacan *et al.* ont approché Σ_{k+1} par une matrice diagonale $\tilde{\Sigma}_{k+1}$ dont les éléments diagonaux sont égaux à l'inverse des éléments diagonaux de $(\Sigma_{k+1})^{-1}$, c'est à dire

$$(\tilde{\Sigma}_{k+1})_{ii} = 1/(\Sigma_{k+1})_{ii}^{-1}. \quad (4.42)$$

On constate que cette approximation est grossière et qu'elle peut conduire à de fortes erreurs.

Approche basée sur l'algorithme d'approximation BV-Grad

La seule différence entre cette approche et notre approche proposée dans la partie 4.2.2 est que l'algorithme d'approximation BV-Grad est utilisé pour optimiser $q_{\mathbf{x}}$. En fait, l'algorithme d'approximation BV-Grad peut être considéré comme un cas particulier de BV-MG avec le second pas s_2 toujours égal à zéro. En conséquence, les équations de mise à jour pour les paramètres de $q_{\mathbf{x}}$ peuvent être facilement obtenues à partir des équations de mise à jour de notre approche en prenant $s_2 = 0$. Par ailleurs, l'optimisation de la variable auxiliaire λ et des lois de hyperparamètres est identique à celle de notre approche.

4.2.4 Simulations

La performance de l'algorithme proposé (voir l'algorithme 8) est évaluée sur un problème de Super-Résolution (SR). Ce problème de SR peut être vu comme un problème

4.2.4 - Simulations

inverse associé au modèle direct (1.2) en considérant que la matrice \mathbf{A} rassemble l'opérateur de déformation, l'opérateur de flou et l'opérateur de sous-échantillonnage. En effet, la préoccupation principale de cette partie est d'évaluer l'efficacité en termes de temps de calculs de notre algorithme d'approximation BV-MG dans un problème de reconstruction d'image. Pour ce faire nous comparons avec deux algorithmes d'approximation BV de l'état de l'art : BVClass [BMK11] et BV-Grad [FR14]. Pour des raisons de simplicité, nous traitons dans cette partie un problème de SR non classique où la matrice du système \mathbf{A} est supposée connue, c'est à dire que nous n'estimons pas conjointement les paramètres de mouvement. En même temps, cette hypothèse permettrait de réduire la limitation de l'approche de l'état de l'art basée sur BVClass [BMK11] dans les problèmes de dimension importante (nécessité de stocker une matrice de taille importante). Cette approche de l'état de l'art basée sur BVClass et une autre approche basée sur BVGrad ont été présentées brièvement dans la partie 4.2.3. Dans cette partie, nous montrons des résultats de reconstruction obtenus par ces trois approches.

Dans la suite, l'approche basée sur BVClass est appelée BVClassSR, l'approche basée sur BV-Grad est nommée BV-GradSR et notre approche proposée basée sur BV-MG est appelée BV-MGSR.

Configuration de simulation

L'objectif de la SR est de reconstruire une image à haute résolution (HR) à partir de plusieurs images à basse résolution (BR) représentant la même scène. Dans notre simulation, quatre groupes d'images à basse résolution sont générés à partir de quatre images standards, *Testpat* (figure 4.2 (a)), *Cameraman* (figure 4.2 (b)) de dimension 256×256 et *Lena* (figure 4.2 (c)), *Jetplane* (figure 4.2 (d)) de dimension 512×512 . Lors de la génération des images BR, un noyau de convolution uniforme de dimension 3×3 et un facteur de décimation 4 dans la direction horizontale et la direction verticale sont utilisés. De plus, nous ajoutons des bruits blancs gaussiens *i.i.d.* à cinq différents niveaux : le rapport signal sur bruit (SNR) égal à 5 dB, 15 dB, 25 dB, 35 dB et 45 dB. Par ailleurs, pour des raisons indiquées ci-dessus, nous traitons des problèmes simplifiés où les images BR sont uniquement translatées (pas de rotation ni d'homothétie). Pour la reconstruction des images HR, nous prenons douze images BR comme données et supposons que le noyau de convolution, le facteur de décimation ainsi que les déplacements des images BR sont tous connus. Comme le facteur de décimation est égale à 4 dans la direction horizontale et la direction verticale, la taille de chaque image BR est 16 fois plus petite que celle de l'image HR. Par conséquent, la taille des données (douze images BR) est plus petite que celle de l'image HR à estimer. Nous sommes donc bien en présence d'un problème mal posé.

Résultats de reconstruction et discussions

En fait, les trois approches considérées sont basées sur un même modèle bayésien et calculent l'estimateur de l'espérance *a posteriori* à l'aide des approximations BV (hormis le degré de séparabilité de l'approche BV). Par conséquent, ces trois approches conduisent généralement à des résultats similaires. La préoccupation principale de cette partie est de comparer leurs vitesses de convergence. Pour faire une comparaison équitable, nous pre-

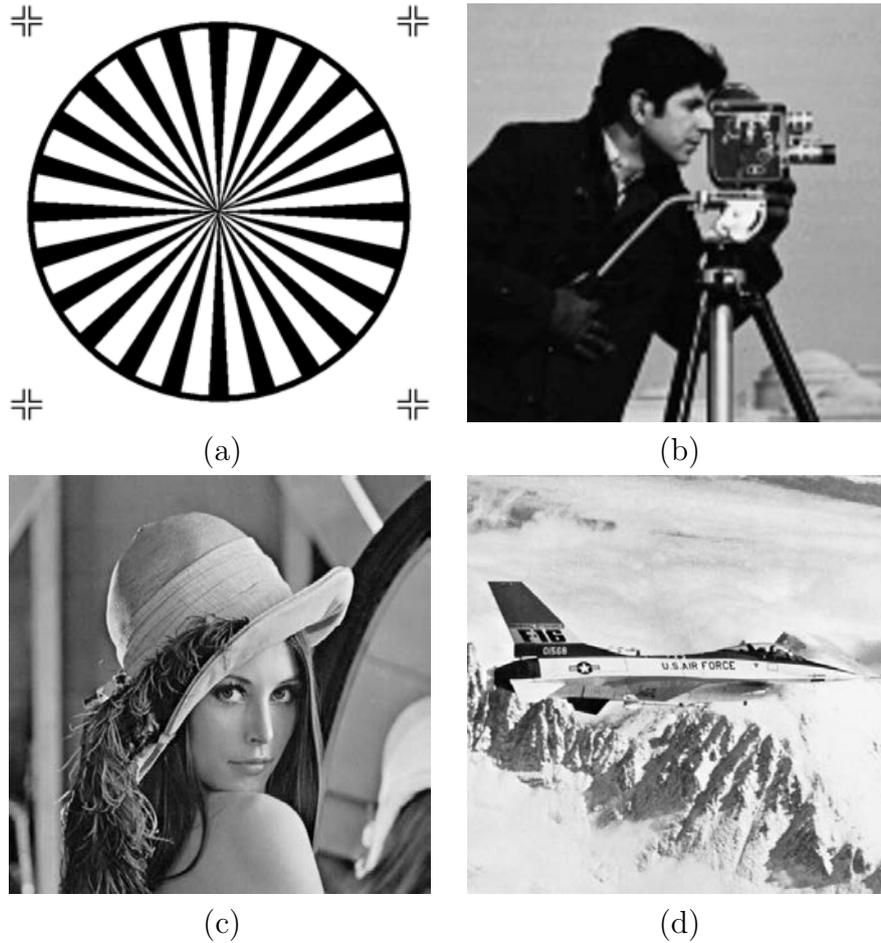


FIGURE 4.2 – Images à haute résolution : (a) *Testpat* de dimension 256×256 , (b) *Cameraman* de dimension 256×256 , (c) *Lena* de dimension 512×512 , (d) *Jetplane* de dimension 512×512 .

nous les mêmes initialisations pour toutes les approches : $\mathbf{m}_0 = \mathbf{A}^T \mathbf{y}$ comme la moyenne et 100 comme la variance de chaque pixel de l'image HR, les valeurs initiales des variables auxiliaires $\boldsymbol{\lambda}$ et des hyperparamètres sont calculées à partir de \mathbf{m}_0 en utilisant les équations de mise à jour. En outre, le paramètre θ impliqué dans la fonction de partition de la loi *a priori* TV a besoin d'être réglé. Une série d'expériences effectuées avec des images différentes montrent que les meilleurs résultats sont obtenus lorsque $\theta = 1.1$. En conséquence, nous prenons $\theta = 1.1$ pour toutes les expériences. Ce paramètre est lié au degré de parcimonie des images de différences finies.

En ce qui concerne le critère d'arrêt de BVClassSR, nous utilisons $\|\mathbf{m}_k - \mathbf{m}_{k-1}\| / \|\mathbf{m}_{k-1}\| < 10^{-5}$ où \mathbf{m}_k et \mathbf{m}_{k-1} représentent l'estimation de l'image HR à k -ième et à $(k - 1)$ -ième itération, respectivement. Pour BV-GradSR et BV-MGSR, elles s'arrêtent lorsqu'elles convergent à une image de valeur PSNR proche (différence inférieure à 1‰) de celle obtenue par BVClassSR.

Toutes les expériences sont effectuées sous Matlab R2013a sur Intel(R) Core(TM) i7-3770 CPU (3.40 GHz) avec 8.0 GB RAM. Nous montrons dans le tableau 4.1 les nombres d'itérations ainsi que les temps de calculs utilisés par BVClassSR [BMK11], BV-GradSR [FR14] et notre approche BV-MGSR pour obtenir des images HR de valeurs

4.2.4 - Simulations

TABLE 4.1 – Comparaisons des performances de BVClassSR [BMK11], BV-GradSR [FR14] et notre approche BV-MGSR en termes de nombre d’Itérations (Nb) et Temps de Calculs (en Secondes).

Data		PSNR	BVClassSR		BV-GradSR		BV-MGSR	
			Nb	Temps(s)	Nb	Temps(s)	Nb	Temps(s)
Testpat	5dB	16.84	83	12.8	100	1.5	99	2.0
	15dB	20.96	27	4.1	72	1.2	58	1.1
	25dB	25.55	24	5.3	165	2.5	55	1.2
	35dB	29.85	33	10.7	283	4.3	69	1.4
	45dB	32.60	49	19.3	412	6.3	78	1.6
Camera -man	5dB	23.24	181	51.9	327	5.0	254	4.9
	15dB	28.70	46	7.1	85	1.3	81	1.6
	25dB	33.36	23	4.6	137	2.0	69	1.5
	35dB	37.40	28	7.8	290	4.4	75	1.6
	45dB	40.50	34	13.5	425	6.4	110	2.3
Lena	5dB	27.08	96	87.2	300	16.1	232	17.7
	15dB	31.35	30	17.0	92	5.0	85	6.5
	25dB	34.59	22	16.1	147	8.0	71	5.4
	35dB	37.07	26	26.3	260	14.1	77	5.9
	45dB	38.44	32	40.0	405	21.8	87	6.7
Jetplane	5dB	32.91	123	72.7	197	10.6	180	13.7
	15dB	37.33	30	15.6	84	4.5	74	5.6
	25dB	41.21	21	15.9	150	8.0	71	5.4
	35dB	44.49	25	25.5	379	20.1	85	6.5
	45dB	46.60	31	38.8	362	19.3	99	7.5

PSNR similaires (variation $< 1\%$). Les valeurs PSNR des reconstructions sont données dans la troisième colonne du tableau 4.1. Et les résultats présentés dans ce tableau sont des valeurs moyennes de 20 simulations avec 20 réalisations du bruit différentes à chaque niveau de bruit pour toutes les quatre images.

Dans le tableau 4.1, nous utilisons les chiffres en gras pour indiquer les meilleurs résultats dans chaque cas, c’est à dire le temps de calculs le plus court. En comparant les temps de calcul, nous pouvons voir que l’approche BV-MGSR est beaucoup plus efficace que BVClassSR pour l’ensemble des données et plus efficace que BV-GradSR dans la plupart des cas, en particulier dans les cas les moins bruités, tels que SNR = 25, 35 et 45 dB. Par exemple, dans le cas où SNR = 5 dB de l’image *Cameraman*, l’approche BV-MGSR utilise 4.9 secondes qui est seulement 9% de 51.9 secondes utilisée par BVClassSR et est légèrement plus petite que 5.0 secondes de l’approche BV-GradSR, dans le cas où SNR = 25 dB de l’image *Lena*, l’approche BV-MGSR prend 5.4 secondes qui est à peu près 34% du temps pris par BVClassSR (16.1 secondes) et est 68% du temps pris par BV-GradSR (8.0 secondes), dans le cas où SNR = 45 dB de l’image *Jetplane*, l’approche BV-MGSR prend 7.5 secondes qui est seulement 19% du temps pris par BVClassSR (38.8 secondes) et 39% du temps pris par BV-GradSR (19.3 secondes). En moyenne, l’approche BV-MGSR est environs 5 fois plus rapide que BVClassSR et environs 1.9 fois plus vite que l’approche BV-GradSR. Le plus faible gain en termes de temps de calculs pour les images ayant un fort niveau du bruit peut s’expliquer par des critères à optimiser plus

réguliers où le choix de la direction du gradient permet d'avoir de bonne performance.

Nous comparons ensuite le nombre d'itérations. En comparant la sixième et la huitième colonne du tableau 4.1, nous pouvons voir que l'approche BV-MGSR prend toujours moins d'itérations que l'approche BV-GradSR. Ce résultat tend à montrer que l'introduction de sous-espace mémoire du gradient permet de générer une meilleure direction à chaque itération. L'approche est donc systématiquement plus efficace en termes d'itérations. Même si chaque itération de BV-MGSR est plus couteuse en temps de calculs que BV-GradSR en raison de sa complexité accrue, l'approche BV-MGSR reste plus efficace que l'approche BV-GradSR dans la plupart des cas. En ce qui concerne BVClassSR, elle prend moins d'itérations que les deux autres approches. Néanmoins, chaque itération de l'approche BVClassSR est beaucoup plus couteuse car elle contient une boucle interne qui sert à approcher itérativement \mathbf{m}_{k+1} en utilisant la méthode du gradient conjugué. En conséquence, BVClassSR est le moindre efficace en termes de temps de calcul.

Afin de comparer les qualités visuelles des images reconstruites, nous montrons les reconstructions sur-résoluées de quatre images différentes pour différents SNR, voir les figures (4.3)-(4.6). Les résultats dans le cas SNR = 45 dB ne sont pas montrés parce que dans ce cas, le niveau de bruit est très faible et les reconstructions sont très proches de l'image vraie. En comparant les images BR (montrées dans la première ligne) et les images HR obtenues (voir les trois lignes en bas des figures (4.3) - 4.6), nous pouvons voir que toutes les approches augmentent la résolution des images. Cette propriété est conservée même dans les cas où le bruit est élevé, SNR = 5 dB. On remarque aussi que les bruits présents dans les images BR sont réduits efficacement, et que les contours des images ne sont pas trop lissés. De plus, les trois approches donnent des images HR très similaires, ce qui est cohérent avec le fait que les valeurs de PSNR sont similaires. En outre, comme indiqué précédemment, toutes les images sont obtenues avec une même $\theta = 1.1$. Nous pouvons voir que cette valeur de θ marche bien pour toutes les données testées. Les images ici bien que très différentes ont un degré de parcimonie dans les images de différences finies proches, c'est pourquoi l'approche n'est pas sensible au paramètre θ .

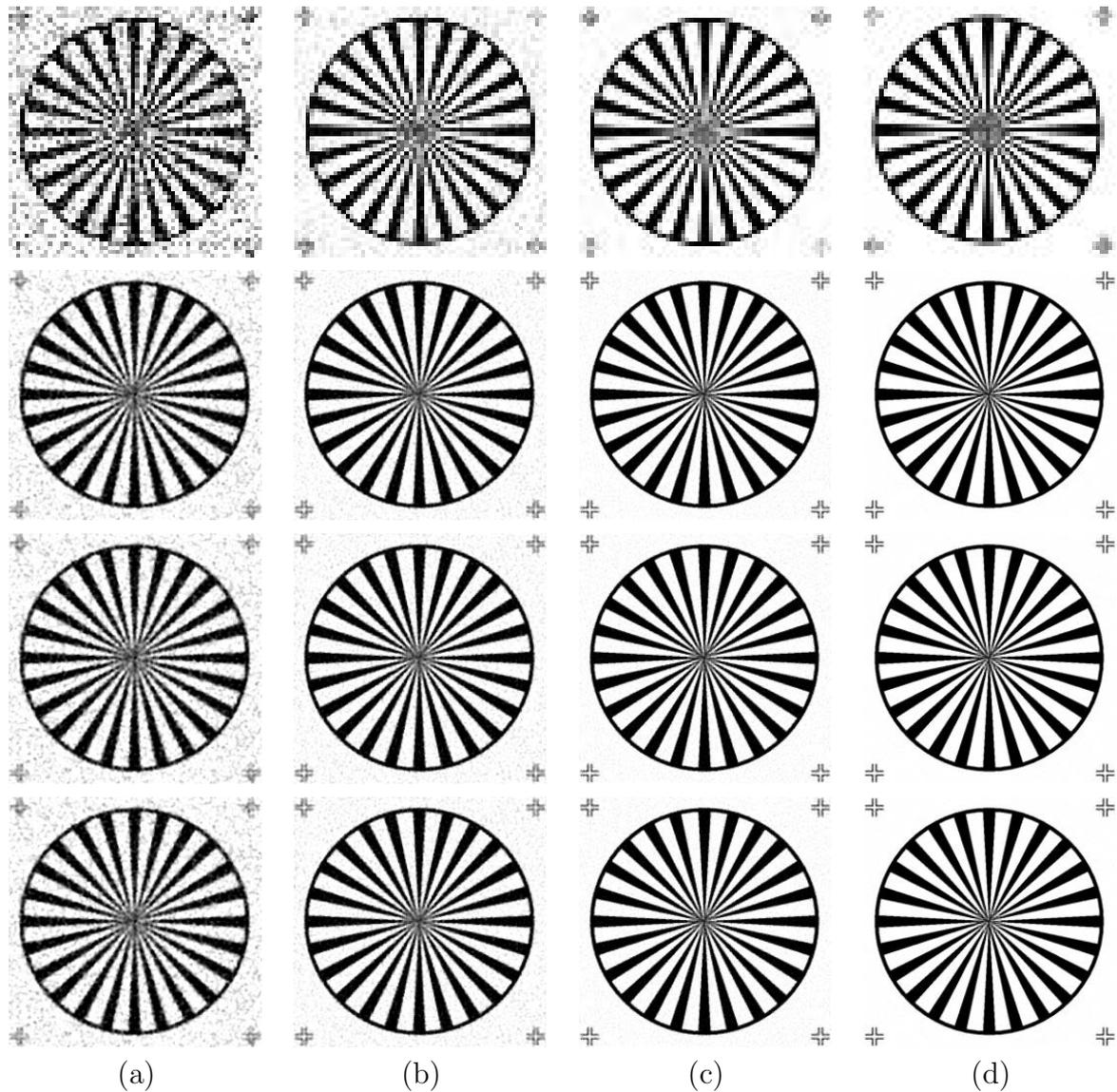


FIGURE 4.3 – Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour *Testpat* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.



FIGURE 4.4 – Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour *Cameraman* dans les cas où SNR = (a) 5 dB; (b) 15 dB; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.



FIGURE 4.5 – Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour *Lena* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.



FIGURE 4.6 – Exemples des images BR (ligne en haut), exemples des images HR obtenues par BVClassSR (la deuxième ligne), BV-GradSR (la troisième ligne), BV-MGSR (la quatrième ligne) pour *Jetplane* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.

4.3 Approche non-supervisée basée sur un *a priori* à variables cachées de type Geman

Dans cette partie, nous nous intéressons à une loi *a priori* favorisant les images régulières par morceau comme la loi TV et possédant une fonction de partition explicitement connue. Pour contourner ce problème nous avons étudié un travail présenté dans [Gio08]. Giovannelli a proposé un modèle *a priori* à variables cachées basé sur un modèle semi-quadratique proposé par Geman et Yang [GY95] et son interprétation bayésienne d'un modèle mélange de gaussiennes par changement de positions (GPM) [CI04] (voir aussi chapitre 1, partie 1.4.1 pour plus de détails). Notons le vecteur des variables cachées par \mathbf{b} . Pour le modèle proposé par Giovannelli, la loi conditionnelle de \mathbf{x} sachant \mathbf{b} prend la forme suivante :

$$p(\mathbf{x}|\mathbf{b}, \gamma_d) = Z_{\mathbf{x}|\mathbf{b}}^{-1} \exp \left[-\frac{\gamma_d}{2} \|\mathbf{D}\mathbf{x} - \mathbf{b}\|^2 \right] \quad (4.43)$$

où γ_d est un paramètre de précision et \mathbf{D} est une matrice à choisir, par exemple, elle peut être une matrice de différence finie. En outre, $Z_{\mathbf{x}|\mathbf{b}}$ est la fonction de partition de cette distribution conditionnelle qui est définie par

$$Z_{\mathbf{x}|\mathbf{b}} = \int \exp \left[-\frac{\gamma_d}{2} (\|\mathbf{D}\mathbf{x} - \mathbf{b}\|^2) \right] d\mathbf{x}. \quad (4.44)$$

Sous condition que la matrice \mathbf{D} est de taille $N \times N$ et qu'elle a des coefficients de la FFT non-nuls, $Z_{\mathbf{x}|\mathbf{b}}$ a une forme explicite qui ne dépend que de γ_d (voir [Gio08] pour les détails).

Dans le cas où la variable caché \mathbf{b} suit une loi de Laplace, la loi jointe pour (\mathbf{x}, \mathbf{b}) a une expression explicite : sa fonction de partition est explicitement connue. Ce modèle donne une distribution Laplace-convolved-Gauss pour la loi marginale $p(\mathbf{x})$, qui est une distribution ressemblant à l'exponentielle du potentiel de Huber [Hub81, CI04, Gio08].

En fait, si \mathbf{D} est une matrice de différence finie, la variable cachée \mathbf{b} correspond à la moyenne de la loi de la différence entre deux pixels voisins. Comme \mathbf{b} suit une loi de Laplace, les informations apportées par ce modèle *a priori* sont que la plupart des différences entre les pixels adjacents sont petites en autorisant certaines différences grandes. On favorise des gradients parcimonieux, donc des images régulières par morceau. Ce modèle permet de satisfaire nos contraintes : la densité de distribution est explicitement connue et il permet de décrire des images régulières par morceau.

Néanmoins, dans le travail de Giovannelli, la technique MCMC a été utilisée pour calculer une estimation de l'espérance *a posteriori*, ce qui est coûteux en calcul. Lorsque les problèmes sont de grande dimension, cette approche est limitée aux problèmes à des matrices facilement inversibles. Par ailleurs, il existe un problème pour déterminer les hyperparamètres car la forme de la loi *a priori* change avec les valeurs des hyperparamètres. Dans la suite de notre travail, nous abordons ces deux problèmes de façon différente par rapport à Giovannelli. Premièrement, pour calculer l'estimateur de l'espérance *a posteriori* plus rapidement, nous utilisons l'algorithme d'approximation BV-MG introduit dans le chapitre précédent et qui est beaucoup plus efficace que les approches MCMC. Deuxièmement, pour résoudre le problème concernant la détermination des hyperparamètres, nous effectuons une reparamétrisation des hyperparamètres qui nous permet d'obtenir un paramètre de forme et un paramètre d'échelle. Comme le paramètre de

forme détermine le type d'informations *a priori* introduites, nous le fixons. Le paramètre d'échelle permet de régler l'importance de l'information *a priori* dans la solution obtenue. En pratique, nous le laissons être déterminé automatiquement dans notre approche. Dans la suite, nous décrivons d'abord en détails la loi *a priori* utilisée, la reparamétrisation ainsi que la loi *a posteriori*. Ensuite, nous appliquons notre algorithme d'approximation BV-MG qui nous permet d'obtenir l'estimation de l'espérance *a posteriori*.

4.3.1 Formulation bayésienne

Loi *a priori* pour l'objet d'intérêt

Comme nous avons discuté ci-dessus, nous prenons le modèle à variables cachées proposé par Giovannelli [Gio08]. Néanmoins, contrairement à l'approche de Giovannelli [Gio08] où la matrice \mathbf{D} est supposée de taille $N \times N$, nous supposons que $\mathbf{D} = [\mathbf{D}_h^T, \mathbf{D}_v^T]^T$ où \mathbf{D}_h et \mathbf{D}_v sont toutes de taille $N \times N$ et définies comme dans le cas TV : \mathbf{D}_h et \mathbf{D}_v sont les matrices de différence finie d'ordre un dans la direction horizontale et dans la direction verticale, respectivement. Ici, une matrice \mathbf{D} de différence finie de taille $N \times N$ n'est pas prise car elle ne permet pas de bien décrire la régularité par morceau. Par ailleurs, le vecteur des variables cachées $\mathbf{b} = [\mathbf{b}_h^T, \mathbf{b}_v^T]^T$ avec $\mathbf{b}_h \in \mathbb{R}^N$ et $\mathbf{b}_v \in \mathbb{R}^N$. Dans ce cas, la loi conditionnelle de \mathbf{x} sachant \mathbf{b} s'écrit comme :

$$p(\mathbf{x}|\mathbf{b}, \gamma_d) = Z_{\mathbf{x}|\mathbf{b}}^{-1} \exp \left[-\frac{\gamma_d}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right], \quad (4.45)$$

qui est une distribution gaussienne. Ici, γ_d est le paramètre de précision et la fonction de partition $Z_{\mathbf{x}|\mathbf{b}}$ prend la forme suivante

$$Z_{\mathbf{x}|\mathbf{b}} = \int \exp \left[-\frac{\gamma_d}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right] d\mathbf{x}. \quad (4.46)$$

Dans le cas étudié dans [Gio08] où la matrice \mathbf{D} est de taille $N \times N$, la fonction de partition $Z_{\mathbf{x}|\mathbf{b}}$ est indépendante de la variable cachée \mathbf{b} et dépend de γ_d (voir [Gio08]). Néanmoins, dans notre cas, la dimension de \mathbf{D} est $2N \times N$ qui fait que la taille de $\mathbf{D}\mathbf{x}$ est deux fois celle de \mathbf{x} . Nous pouvons quand même calculer la fonction de partition en transposant les calculs de l'intégrale dans le domaine de la transformée de Fourier. En supposant que la matrice \mathbf{D} est circulante, de la même manière que dans [Gio08] (voir l'annexe C.2 pour les détails de calcul). Dans ce cas, la fonction de partition est un produit du terme $c\gamma_d^{-\frac{N}{2}}$ qui ne dépend que de γ_d et d'un autre terme qui dépend en même temps de l'hyperparamètre γ_d et de la variable cachée \mathbf{b} (voir (C.8)), plus précisément la transformée de Fourier de \mathbf{b} . Néanmoins, cette fonction de partition a une expression trop compliquée pour être calculée efficacement. Pour s'affranchir de cette difficulté, nous proposons de l'approcher par une fonction plus simple. On remarque que la fonction de partition est bornée supérieurement par un terme simple qui est donné par :

$$Z_{\mathbf{x}|\mathbf{b}} \leq c\gamma_d^{-N/2}. \quad (4.47)$$

Cette borne supérieure ne dépend que de γ_d . Dans la suite, nous utilisons cette borne supérieure comme une approximation de la fonction de partition afin de réduire la complexité du calcul.

4.3.1 - Formulation bayésienne

Pour construire une loi *a priori* permettant de décrire les connaissances de *régularité par morceau*, une loi de Laplace est introduite pour la variable auxiliaire \mathbf{b} . Nous pouvons alors écrire

$$p(\mathbf{b}|\gamma_b) \propto \gamma_b^\xi \exp \left[-\frac{\gamma_b}{2} \|\mathbf{b}\|_1 \right], \quad (4.48)$$

où $\|\mathbf{b}\|_1 = \sum_{i=1}^{2N} |b_i|$ est la norme L^1 , et γ_b^ξ avec $0 \leq \xi \leq 2N$ est la fonction de partition. On peut remarquer que nous avons introduit la constante ξ au lieu de prendre directement $2N$ (la taille de \mathbf{b}). Cette constante est prise considérant que, en pratique, nous pouvons avoir une certaine connaissance préalable sur le support de \mathbf{b} .

Après la définition de la loi conditionnelle de l'objet d'intérêt \mathbf{x} sachant la variable cachée \mathbf{b} ainsi que la loi de \mathbf{b} , la distribution *a priori* de \mathbf{x} peut être obtenue en intégrant par rapport à la variable auxiliaire

$$p(\mathbf{x}|\gamma_d, \gamma_b) \propto \int \gamma_d^{N/2} \exp \left[-\frac{\gamma_d}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right] \times \gamma_b^\xi \exp \left[-\frac{\gamma_b}{2} \|\mathbf{b}\|_1 \right] d\mathbf{b}. \quad (4.49)$$

Dans (4.49), nous pouvons voir que cette distribution dépend de deux hyperparamètres γ_d et γ_b . En pratique, il est difficile de déterminer automatiquement les deux paramètres car la liberté sur le choix de ces deux hyperparamètres fait que la forme de la loi *a priori* est aussi librement choisie. Dans ce cas, l'information *a priori* apportée pourrait ne pas correspondre à ce qu'on veut. Pour s'affranchir de cette difficulté, nous proposons ici de faire une reparamétrisation qui introduit un paramètre de forme et un paramètre d'échelle. Nous fixons le paramètre de forme selon notre connaissance *a priori* et estimons automatiquement le paramètre d'échelle.

À partir de (4.49), nous pouvons facilement trouver que

$$p(\mathbf{x}|\gamma'_d, \gamma'_b) = a^N p(\mathbf{x}|\gamma_d, \gamma_b) \quad \text{avec} \quad \gamma'_d = a^2 \gamma_d, \quad \gamma'_b = a \gamma_b, \quad (4.50)$$

où $a \in \mathbb{R}^+$. L'équation ci-dessus indique que, lorsque nous changeons les valeurs des paramètres γ_d et γ_b tout en gardant la valeur du rapport $\frac{\gamma_b}{\sqrt{\gamma_d}}$, la forme de la distribution ne change pas, c'est à dire que seulement l'échelle et l'amplitude de la distribution changent. Nous pouvons donc identifier le rapport de γ_b à $\sqrt{\gamma_d}$ comme le paramètre de forme de $p(\mathbf{x}|\gamma_d, \gamma_b)$ et prendre soit γ_d , soit γ_b , comme paramètre d'échelle. Dans la suite, nous utilisons ν pour représenter le paramètre de forme. Nous avons donc

$$\nu = \frac{\gamma_b}{\sqrt{\gamma_d}}. \quad (4.51)$$

Nous avons expérimentalement constaté que lorsque ν prend une valeur petite, la distribution *a priori* est à queue lourde, plus spécifiquement, qu'elle ressemble à une distribution de Cauchy et quand la valeur de ν augmente, la forme de la distribution devient à queue moins lourde qui ressemble plutôt à une distribution gaussienne. Par conséquent, la valeur du paramètre de forme ν est ajustée en fonction des caractéristiques de l'objet d'intérêt.

Prenons γ_d comme le paramètre d'échelle. En remplaçant γ_b par $\nu \sqrt{\gamma_d}$, nous pouvons

en déduire une distribution jointe de \mathbf{x} et \mathbf{b} comme ci-dessous :

$$p(\mathbf{x}, \mathbf{b} | \gamma_d, \nu) \propto \gamma_d^{N/2} \exp \left[-\frac{\gamma_d}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right] \\ \times (\nu \sqrt{\gamma_d})^\xi \exp \left[-\frac{\nu \sqrt{\gamma_d}}{2} \|\mathbf{b}\|_1 \right]. \quad (4.52)$$

4.3.2 Loi *a priori* pour les hyperparamètres

Comme nous avons fait auparavant, nous choisissons les lois *a priori* conjuguées pour faciliter l'application des approches d'approximation BV. En ce qui concerne γ_n , nous employons la loi Gamma,

$$p(\gamma_n) = \mathcal{G}(\gamma_n | \tilde{a}_n, \tilde{b}_n), \quad (4.53)$$

qui donne un *a priori* de Jeffreys si $\tilde{a}_n = 0$, $\tilde{b}_n = 0$.

Néanmoins, pour γ_d , une loi Gamma n'est plus une loi *a priori* conjuguée en raison du terme $\sqrt{\gamma_d}$ dans (4.52). Pour s'affranchir de cette difficulté, nous faisons d'abord une autre reparamétrisation où nous définissons un nouveau paramètre \mathcal{K}_d comme

$$\mathcal{K}_d = \sqrt{\gamma_d}. \quad (4.54)$$

Ensuite, nous définissons une densité de probabilité paramétrique inhabituelle pour les approches bayésiennes qui peut être utilisée comme la loi *a priori* conjuguée de \mathcal{K}_d . Dans la suite, cette densité est notée Ψ et paramétrée par α , β et μ . Cette densité s'écrit comme

$$\Psi(z | \alpha, \beta, \mu) = \frac{1}{Z_\Psi} z^\alpha \exp [-\beta(z + \mu)^2], \quad (4.55) \\ \text{pour } z > 0 \text{ et } \alpha, \beta, \mu > 0,$$

où la fonction de partition Z_Ψ est calculée comme suit :

$$Z_\Psi = \int_0^{+\infty} z^\alpha \exp [-\beta(z + \mu)^2] dz \\ = \sum_{i=0}^{\alpha} \binom{\alpha}{i} (-\mu)^i \int_{\mu}^{+\infty} z'^{\alpha-i} \exp(-\beta z'^2) dz' \\ = \sum_{i=0}^{\alpha} \binom{\alpha}{i} (-\mu)^i \frac{1}{2\beta^{\frac{\alpha-i+1}{2}}} \Gamma \left(\frac{\alpha-i+1}{2}, \beta\mu^2 \right), \quad (4.56)$$

ici $\Gamma(r, u) = \int_u^{\infty} t^{r-1} e^{-t} dt$ est la fonction gamma incomplète supérieure. Donc, Z_Ψ existe et $Z_\Psi \in (0, +\infty)$. Dans cette partie, nous prenons la distribution Ψ comme la loi *a priori* de \mathcal{K}_d ,

$$p(\mathcal{K}_d) = \Psi(\mathcal{K}_d | \tilde{\alpha}, \tilde{\beta}, \tilde{\mu}). \quad (4.57)$$

En pratique, pour obtenir la loi de Jeffreys, on peut prendre $\tilde{\alpha} = -1$, $\tilde{\beta} = 0$ et $\tilde{\mu} = 0$.

4.3.3 - Application des méthodes d'approximation BV

En remplaçant γ_d par \mathcal{K}_d^2 et en utilisant la loi *a priori* définie ci-dessus, nous pouvons déduire la loi *a posteriori* jointe :

$$\begin{aligned}
p(\mathbf{x}, \mathbf{b}, \gamma_n, \mathcal{K}_d | \mathbf{y}, \nu) &\propto p(\mathbf{y} | \mathbf{x}, \gamma_n) p(\mathbf{x} | \mathbf{b}, \mathcal{K}_d) p(\mathbf{b} | \mathcal{K}_d, \nu) p(\gamma_n) p(\mathcal{K}_d) \\
&\tilde{\propto} \gamma_n^{M/2} \exp \left[-\frac{\gamma_n}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 \right] \\
&\times \mathcal{K}_d^N \exp \left[-\frac{\mathcal{K}_d^2}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right] \\
&\times (\nu \mathcal{K}_d)^\xi \exp \left[-\frac{\nu \mathcal{K}_d}{2} \|\mathbf{b}\|_1 \right] \gamma_n^{\tilde{a}_n - 1} \exp \left[-\tilde{b}_n \gamma_n \right] \mathcal{K}_d^{\tilde{\alpha}} \exp \left[-\tilde{\beta} (\mathcal{K}_d + \tilde{\mu})^2 \right]
\end{aligned} \tag{4.58}$$

Encore une fois, cette loi *a posteriori* n'est pas connue explicitement dû à une constante de normalisation incalculable. Pour s'affranchir de cette difficulté, nous utilisons la méthode d'approximation BV.

4.3.3 Application des méthodes d'approximation BV

Dans la suite, $\Theta = \{\mathbf{x}, \mathbf{b}, \gamma_n, \mathcal{K}_d\}$ est le vecteur des variables à estimer. Pour ces paramètres, nous déterminons dans cette partie les estimateurs de l'espérance *a posteriori* à l'aide des méthodes d'approximation BV, qui génèrent une approximation séparable q_Θ de la loi *a posteriori* $p(\mathbf{x}, \mathbf{b}, \gamma_n, \mathcal{K}_d | \mathbf{y}, \nu)$.

Nous prenons ici la séparabilité totale pour \mathbf{x} pour des raisons expliquées précédemment. Nous écrivons donc

$$\begin{aligned}
q_\Theta(\Theta) &= q_{\mathbf{x}}(\mathbf{x}) q_{\mathbf{b}}(\mathbf{b}) q_{\gamma_n}(\gamma_n) q_{\mathcal{K}_d}(\mathcal{K}_d) \\
&= \prod_i q_i(x_i) q_{\mathbf{b}}(\mathbf{b}) q_{\gamma_n}(\gamma_n) q_{\mathcal{K}_d}(\mathcal{K}_d).
\end{aligned} \tag{4.59}$$

En fait, car $p(\mathbf{b} | \mathbf{y}, \mathbf{x}, \gamma_n, \mathcal{K}_d, \nu)$ est elle-même séparable, $q_{\mathbf{b}}$ obtenue par l'approximation BV est toujours séparable peu importe qu'on impose la séparabilité entre les éléments de \mathbf{b} ou non.

L'objectif est donc de trouver une distribution séparable qui maximise l'énergie libre négative \mathcal{F} . Pour ce faire, nous menons une optimisation alternée de \mathcal{F} par rapport à $q_{\mathbf{x}}$, $q_{\mathbf{b}}$, q_{γ_n} et $q_{\mathcal{K}_d}$:

$$q_{\mathbf{x}}^{k+1} = \arg \max_{q_{\mathbf{x}}} \mathcal{F} (q_{\mathbf{x}} q_{\mathbf{b}}^k q_{\gamma_n}^k q_{\mathcal{K}_d}^k), \tag{4.60}$$

$$q_{\mathbf{b}}^{k+1} = \arg \max_{q_{\mathbf{b}}} \mathcal{F} (q_{\mathbf{x}}^{k+1} q_{\mathbf{b}} q_{\gamma_n}^k q_{\mathcal{K}_d}^k), \tag{4.61}$$

$$q_{\gamma_n}^{k+1} = \arg \max_{q_{\gamma_n}} \mathcal{F} (q_{\mathbf{x}}^{k+1} q_{\mathbf{b}}^{k+1} q_{\gamma_n} q_{\mathcal{K}_d}^k), \tag{4.62}$$

$$q_{\mathcal{K}_d}^{k+1} = \arg \max_{q_{\mathcal{K}_d}} \mathcal{F} (q_{\mathbf{x}}^{k+1} q_{\mathbf{b}}^{k+1} q_{\gamma_n}^{k+1} q_{\mathcal{K}_d}). \tag{4.63}$$

Grâce à l'utilisation des lois *a priori* conjuguées pour \mathbf{x} et des hyperparamètres γ_n, \mathcal{K}_d , les méthodes d'approximation BV donnent les distributions gaussiennes pour $(q_i^{opt})_{i=1, \dots, N}$ avec les moyennes et les variances des $(x_i)_{i=1, \dots, N}$ incluses dans les vecteurs \mathbf{m}_k and $\boldsymbol{\sigma}_k^2$.

Par ailleurs, l'approximation optimale $q_{\gamma_n}^{opt}$ est une loi Gamma et l'approximation optimale $q_{\mathcal{K}_d}^{opt}$ est une distribution Ψ . Par conséquent, l'optimisation des distributions est ramenée à l'optimisation des paramètres. Nous prenons donc

$$q_{\mathbf{x}}^k(\mathbf{x}) = \prod_i \mathcal{N}(x_i | (\mathbf{m}_k)_i, (\sigma_k^2)_i), \quad (4.64)$$

$$q_{\gamma_n}^k(\gamma_n) = \mathcal{G}(\gamma_n | a_{\gamma_n}^k, b_{\gamma_n}^k), \quad (4.65)$$

$$q_{\mathcal{K}_d}^k(\mathcal{K}_d) = \Psi(\mathcal{K}_d | \alpha^k, \beta^k, \mu^k). \quad (4.66)$$

Néanmoins, dans notre problème, l'optimisation de $q_{\mathbf{b}}$ est plus compliquée car nous utilisons une loi de Laplace $p(\mathbf{b}|\gamma_b)$ (voir (4.48)) qui n'est pas conjuguée avec $p(\mathbf{x}|\mathbf{b}, \gamma_b)$ (voir (4.45)). En conséquence, l'approximation obtenue par les méthodes d'approximation BV n'appartient pas toujours à une même famille de distributions. L'optimisation de $q_{\mathbf{b}}$ ne peut donc pas être réalisée simplement en optimisant certains paramètres. Pour résoudre ce problème, nous proposons d'ici d'employer l'approche d'approximation bayésienne variationnelle restreinte (BVR) [ŠQ06] qui est résumée dans le corollaire 1 du chapitre 2. Cette approche d'approximation BVR s'appuie sur la fixation préalable de la forme de l'approximation distributionnelle de certains paramètres, ici, la forme de $q_{\mathbf{b}}$.

Pour contourner la difficulté posée par la distribution de Laplace, nous supposons que $q_{\mathbf{b}}^k$ est une distribution de Dirac,

$$q_{\mathbf{b}}^k(\mathbf{b}) = \delta(\mathbf{b} - \hat{\mathbf{b}}^k). \quad (4.67)$$

Remarque. *Cette approximation de la loi est grossière, mais elle s'applique sur des variables cachées. Donc, la distribution n'a pas d'intérêt.*

Par conséquent, au lieu de résoudre le problème (4.61), nous maximisons l'énergie libre négative sous la condition que $q_{\mathbf{b}}$ soit une distribution de Dirac. Ce problème est formulé comme

$$q_{\mathbf{b}}^{k+1} = \arg \max_{q_{\mathbf{b}} \text{ est Dirac}} \mathcal{F}(q_{\mathbf{x}}^{k+1} q_{\mathbf{b}} q_{\gamma_n}^k q_{\mathcal{K}_d}^k). \quad (4.68)$$

Grâce à cette contrainte, l'optimisation de $q_{\mathbf{b}}$ peut être aussi effectuée par l'optimisation du paramètre de la distribution de Dirac $\hat{\mathbf{b}}$. En fait, dans ce cas où on impose à l'approximation distributionnelle une distribution de Dirac², cette étape d'optimisation peut être identifiée comme l'étape M de l'algorithme EM [Att00, GB01, BG02].

En ce qui concerne les optimisations par rapport à $q_{\mathbf{x}}$, q_{γ_n} et $q_{\mathcal{K}_d}$, nous les effectuons de façon analogue que dans les parties précédentes : l'optimisation par rapport à q_{γ_n} et $q_{\mathcal{K}_d}$ se fait à l'aide de la méthode d'approximation BV classique car la loi conditionnelle $p(\gamma_n, \mathcal{K}_d | \mathbf{x}, \mathbf{b}, \mathbf{y}, \nu)$ est séparable, l'optimisation par rapport à $q_{\mathbf{x}}$ est faite en utilisant la méthode d'approximation BV-MG.

Optimisation de $q_{\mathbf{x}}$ avec l'algorithme d'approximation BV-MG

Comme donné par (4.64), $q_{\mathbf{x}}$ est une distribution gaussienne. Son optimisation est faite en mettant à jour ses paramètres : la moyenne \mathbf{m}_k et la variance σ_k^2 .

2. Dans ce cas, nous prenons directement une estimation ponctuelle.

4.3.3 - Application des méthodes d'approximation BV

Par rapport au problème traité dans le chapitre 3 et la partie précédente de ce chapitre, ce qui a changé est la loi *a priori* pour \mathbf{x} . Dans ce cas, les dérivations des expressions de \mathbf{m}_k et σ_k^2 sont les mêmes que les cas précédents et nous ne les répétons plus. Néanmoins, le changement de la loi *a priori* entraîne une expression différente pour la fonction intermédiaire $q_i^{r_1}$. Dans ce cas, $q_i^{r_1}$ est définie par

$$\begin{aligned} q_i^{r_1}(x_i) &= \exp \left[\langle \log p(\mathbf{y}, \Theta) \rangle_{\prod_{j \neq i} q_j^k q_{\mathbf{b}}^k q_{\gamma_n}^k q_{\mathcal{K}_d}^k} \right] \\ &\propto \exp \left[- \int \left(\frac{\gamma_n}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + \frac{\mathcal{K}_d^2}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right) \right. \\ &\quad \left. \prod_{j \neq i} q_j^k(x_j) dx_j q_{\mathbf{b}}^k(\mathbf{b}) q_{\gamma_n}^k(\gamma_n) q_{\mathcal{K}_d}^k(\mathcal{K}_d) d\mathbf{b} d\gamma_n d\mathcal{K}_d \right]. \end{aligned} \quad (4.69)$$

Même si l'expression de $q_i^{r_1}$ est différente, les détails de calculs ressemblent fortement à ceux montrés en annexe A.1 car les deux nouveaux termes $\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2$ et $\|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2$ ont une même structure que $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2$. En conséquence, nous ne montrons plus ici les calculs pour faciliter la lecture de cette partie. Encore une fois, $(q_i^{r_1})_{i=1, \dots, N}$ correspond à la densité d'une distribution gaussienne avec la moyenne et la variance explicitement exprimées par :

$$\begin{aligned} (\sigma_{r_1}^2)_i &= \left[\langle \gamma_n \rangle^k \text{diag}(\mathbf{A}^T \mathbf{A})_i + \langle \mathcal{K}_d^2 \rangle^k \left(\text{diag}(\mathbf{D}_h^T \mathbf{D}_h)_i + \text{diag}(\mathbf{D}_v^T \mathbf{D}_v)_i \right) \right]^{-1}, \quad (4.70) \\ (\mathbf{m}_{r_1})_i &= (\sigma_{r_1}^2)_i \left[\langle \gamma_n \rangle^k (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_k + \text{diag}(\mathbf{A}^T \mathbf{A}) \circ \mathbf{m}_k) \right. \\ &\quad + \langle \mathcal{K}_d^2 \rangle^k \left(\mathbf{D}_h^T \hat{\mathbf{b}}_h^k - \mathbf{D}_h^T \mathbf{D}_h \mathbf{m}_k + \text{diag}(\mathbf{D}_h^T \mathbf{D}_h) \circ \mathbf{m}_k \right) \\ &\quad \left. + \langle \mathcal{K}_d^2 \rangle^k \left(\mathbf{D}_v^T \hat{\mathbf{b}}_v^k - \mathbf{D}_v^T \mathbf{D}_v \mathbf{m}_k + \text{diag}(\mathbf{D}_v^T \mathbf{D}_v) \circ \mathbf{m}_k \right) \right]_i. \end{aligned} \quad (4.71)$$

Les mises à jour de \mathbf{m}_k et σ_k^2 sont donc effectuées en utilisant (4.30) et (4.31) où $\sigma_{r_1}^2$ et \mathbf{m}_{r_1} sont respectivement déterminées par (4.70) et (4.71).

Optimisation de $q_{\mathbf{b}}$ en utilisant l'algorithme d'approximation BVR

Comme discuté ci-dessus, nous cherchons une distribution de Dirac qui maximise l'énergie libre négative, c'est à dire résout le problème (4.68). Car la distribution de Dirac est paramétrée par un seul paramètre $\hat{\mathbf{b}}$, l'optimisation de $q_{\mathbf{b}}$ est effectuée en optimisant $\hat{\mathbf{b}}$. La résolution de (4.68) conduit à

$$\begin{aligned} \hat{\mathbf{b}}^{k+1} &= \arg \max_{\hat{\mathbf{b}}} \left[\langle \log p(\mathbf{y}, \Theta) \rangle_{q_{\mathbf{x}}^{k+1} \delta(\mathbf{b} - \hat{\mathbf{b}}) q_{\gamma_n}^k q_{\mathcal{K}_d}^k} \right] \\ &= \arg \max_{\hat{\mathbf{b}}} \left[- \int \left(\frac{\mathcal{K}_d^2}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) + \frac{\nu \mathcal{K}_d}{2} \|\mathbf{b}\|_1 \right) \right. \\ &\quad \left. \times q_{\mathbf{x}}^{k+1}(\mathbf{x}) \delta(\mathbf{b} - \hat{\mathbf{b}}) q_{\gamma_n}^k(\gamma_n) q_{\mathcal{K}_d}^k(\mathcal{K}_d) d\mathbf{x} d\gamma_n d\mathcal{K}_d \right] \\ &= \arg \min_{\hat{\mathbf{b}}} \left[\frac{\langle \mathcal{K}_d^2 \rangle^k}{2} (\|\mathbf{D}_h \mathbf{m}_{k+1} - \hat{\mathbf{b}}_h\|^2 + \|\mathbf{D}_v \mathbf{m}_{k+1} - \hat{\mathbf{b}}_v\|^2) + \frac{\nu \langle \mathcal{K}_d \rangle^k}{2} \|\hat{\mathbf{b}}\|_1 + \text{const} \right], \end{aligned} \quad (4.72)$$

qui nous permet de retrouver un problème de régression linéaire pénalisé par la norme L^1 [Tib96] [DDDM04] dont la solution est connue comme le seuillage doux qui est explicitement exprimé comme

$$\hat{\mathbf{b}}_h^{k+1} = \text{sgn}(\mathbf{D}_h \mathbf{m}_{k+1}) \max \left(|\mathbf{D}_h \mathbf{m}_{k+1}| - \frac{\nu \langle \mathcal{K}_d \rangle^k}{2 \langle \mathcal{K}_d^2 \rangle^k}, \mathbf{0} \right), \quad (4.73)$$

$$\hat{\mathbf{b}}_v^{k+1} = \text{sgn}(\mathbf{D}_v \mathbf{m}_{k+1}) \max \left(|\mathbf{D}_v \mathbf{m}_{k+1}| - \frac{\nu \langle \mathcal{K}_d \rangle^k}{2 \langle \mathcal{K}_d^2 \rangle^k}, \mathbf{0} \right). \quad (4.74)$$

Optimisation de q_{γ_n} et $q_{\mathcal{K}_d}$

Le calcul de q_{γ_n} est le même que précédent et nous ne le répétons pas ici.

En ce qui concerne $q_{\mathcal{K}_d}$, son expression est obtenue en utilisant la formule (2.12) de la méthode d'approximation BV classique :

$$\begin{aligned} q_{\mathcal{K}_d}^{k+1}(\mathcal{K}_d) &\propto \mathcal{K}_d^{N+\xi+\tilde{\alpha}-1} \exp \left[-\mathcal{K}_d^2 \left(\frac{1}{2} \mathbb{E}_{q_{\mathbf{x}}^{k+1} q_{\mathbf{b}}^{k+1}} [\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2] + \tilde{\beta} \right) \right. \\ &\quad \left. - \mathcal{K}_d \left(\frac{\nu}{2} \mathbb{E}_{q_{\mathbf{b}}^{k+1}} [\|\mathbf{b}\|_1] + 2\tilde{\beta}\tilde{\mu} \right) \right] \\ &= \Psi(\mathcal{K}_d | \alpha^{k+1}, \beta^{k+1}, \mu^{k+1}) \end{aligned} \quad (4.75)$$

avec les paramètres de distribution α^{k+1} , β^{k+1} et μ^{k+1} identifiés comme

$$\alpha^{k+1} = N + \xi + \tilde{\alpha} - 1 = \alpha, \quad (4.76)$$

$$\beta^{k+1} = \frac{1}{2} \mathbb{E}_{q_{\mathbf{x}}^{k+1} q_{\mathbf{b}}^{k+1}} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) + \tilde{\beta}, \quad (4.77)$$

$$\mu^{k+1} = \frac{\nu \mathbb{E}_{q_{\mathbf{b}}^{k+1}} (\|\mathbf{b}\|_1) + 4\tilde{\beta}\tilde{\mu}}{4\beta^{k+1}}. \quad (4.78)$$

Ici,

$$\begin{aligned} &\mathbb{E}_{q_{\mathbf{x}}^{k+1} q_{\mathbf{b}}^{k+1}} [\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2] \\ &= \left[\|\mathbf{D}_h \mathbf{m}_{k+1} - \hat{\mathbf{b}}_h^{k+1}\|^2 + \|\mathbf{D}_v \mathbf{m}_{k+1} - \hat{\mathbf{b}}_v^{k+1}\|^2 \right] \\ &\quad + \sum_i \text{diag}(\mathbf{D}_h^T \mathbf{D}_h)_i (\sigma_{k+1}^2)_i + \sum_i \text{diag}(\mathbf{D}_v^T \mathbf{D}_v)_i (\sigma_{k+1}^2)_i, \end{aligned} \quad (4.79)$$

$$\mathbb{E}_{q_{\mathbf{b}}^{k+1}} [\|\mathbf{b}\|_1] = \|\hat{\mathbf{b}}^{k+1}\|_1. \quad (4.80)$$

Après la mise à jour des paramètres de $q_{\mathcal{K}_d}^{k+1}$, il faut calculer son moment d'ordre un et moment d'ordre deux qui vont être utilisés dans les mises à jour des paramètres de $q_{\mathbf{x}}$ et $q_{\mathbf{b}}$ dans l'itération d'après, comme on peut voir dans les équations (4.70), (4.71), (4.73) et (4.74). Néanmoins, comme nous nous intéressons à des problèmes de grande taille, le paramètre α prend une valeur grande, qui rend le calcul direct de ces moments numériquement impossible. Pour aborder ce problème, nous proposons ici d'employer un algorithme de MCMC – l'algorithme de Metropolis-Hastings

4.3.3 - Application des méthodes d'approximation BV

à marche aléatoire [RC00] (voir partie 1.4.2 du chapitre 1 pour une brève introduction). L'algorithme MCMC nous permet d'obtenir un ensemble d'échantillons suivant la distribution $\Psi(\mathcal{K}_d|\alpha^{k+1}, \beta^{k+1}, \mu^{k+1})$. Les moments empiriques d'ordre un et d'ordre deux de ces échantillons sont ensuite pris comme les estimations des moments de $\Psi(\mathcal{K}_d|\alpha^{k+1}, \beta^{k+1}, \mu^{k+1})$.

Dans un algorithme de Metropolis-Hastings à marche aléatoire, la loi instrumentale $g(u|v)$ est choisie d'être de la forme $g(u-v)$, c'est à dire un échantillon $u^{(t)}$ peut s'écrire sous la forme $v^{(t)} + \epsilon^{(t)}$, $\epsilon^{(t)}$ étant une perturbation aléatoire de loi g qui est symétrique et indépendante de $v^{(t)}$. Dans notre algorithme, nous avons pris une distribution uniforme pour g . Nous générons des échantillons de la distribution $\Psi(\mathcal{K}_d|\alpha^{k+1}, \beta^{k+1}, \mu^{k+1})$ à partir de la loi uniforme sur $[-\tau/2, \tau/2]$, notée $\mathcal{U}_{[-\tau/2, \tau/2]}$ en suivant les étapes suivantes :

- (a) Choisir une valeur initiale $\mathcal{K}_d^{(0)}$
- (b) Générer $\epsilon^{(t)} \sim \mathcal{U}_{[-\tau/2, \tau/2]}$, ensuite calculer $\mathcal{K}_d^{(*)} = \mathcal{K}_d^{(t)} + \epsilon^{(t)}$
- (c) Prendre

$$\mathcal{K}_d^{(t+1)} = \begin{cases} \mathcal{K}_d^{(*)} & \text{avec probabilité } \rho(\mathcal{K}_d^{(t)}, \mathcal{K}_d^{(*)}) = \min \left\{ 1, \frac{\Psi(\mathcal{K}_d^{(*)}|\alpha^{k+1}, \beta^{k+1}, \mu^{k+1})}{\Psi(\mathcal{K}_d^{(t)}|\alpha^{k+1}, \beta^{k+1}, \mu^{k+1})} \right\} \\ \mathcal{K}_d^{(t)} & \text{avec probabilité } 1 - \rho(\mathcal{K}_d^{(t)}, \mathcal{K}_d^{(*)}) \end{cases}$$

- (d) Répéter jusqu'à la convergence des échantillons.

Pour l'algorithme de Metropolis-Hastings à marche aléatoire, la valeur initiale ($\mathcal{K}_d^{(0)}$) et le paramètre contrôlant la taille globale de déplacement proposé (τ) jouent un rôle crucial sur l'efficacité de l'algorithme. Ce fait peut être facilement expliqué. Si le point de départ $\mathcal{K}_d^{(0)}$ se trouve dans une région à faible densité, un temps de chauffe important est généralement nécessaire qui réduit l'efficacité de l'algorithme. Considérons le rôle du paramètre τ contrôlant la taille globale de déplacement proposé. Si la plupart des déplacements sont petits par rapport à une certaine mesure de la variabilité de la distribution ciblée, dans ce cas, même si le taux d'acceptation est élevé, la chaîne de Markov se déplace lentement et l'exploration de la distribution ciblée sera relativement inefficace. Dans le cas contraire où les déplacements sont relativement grands par rapport à la mesure de la variabilité de la distribution ciblée, beaucoup de propositions vont être rejetées, la chaîne de Markov ne bougera pas trop et l'exploration de la distribution ciblée sera une autre fois inefficace.

En conséquence, pour obtenir un algorithme relativement efficace et capable d'adapter aux différentes Ψ (les valeurs des paramètres différentes), nous avons pris la configuration suivante. Premièrement, une valeur aléatoire entre zéro et le maximiseur de la distribution Ψ est utilisée comme le point du départ de la marche aléatoire. Le maximiseur de Ψ a une formule analytique simple comme ci-dessous :

$$\mathcal{K}_d^{\max} = -\frac{\mu}{2} + \sqrt{\frac{\mu^2}{4} + \frac{\alpha}{2\beta}}, \quad (4.81)$$

qui peut être facilement calculé.

Deuxièmement, nous avons pris

$$\tau = \frac{1}{50} \mathcal{K}_d^{\max}. \quad (4.82)$$

Finale­ment, notre algorithme non-supervisé basé sur la loi *a priori* à variables cachés de type Geman est résumé dans l’algorithme 9.

Algorithm 9 L’algorithme de reconstruction non-supervisé basé sur un *a priori* de type Geman en utilisant BV-MG, BVR et MCMC

1. Initialiser les paramètres de $(q_i^0)_{i=1,\dots,N}$, $q_{\mathbf{b}}^0$, $q_{\gamma_n}^0$ et $q_{\mathcal{K}_d}^0$
 2. Calculer $q_{\mathbf{x}}^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (4.70)-(4.71)
 - (b) Calculer le pas sous-optimal selon annexe B.2
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (4.30) et (4.31) où le pas s est égal au pas sous-optimal obtenu à l’étape (b)
 3. Calculer le paramètre de $q_{\mathbf{b}}^{k+1}$ en utilisant (4.73) et (4.74)
 4. Calculer $q_{\gamma_n}^{k+1}$: calculer ses paramètres en utilisant (3.60) et (3.61) et calculer sa moyenne en utilisant (3.62)
 5. Calculer $q_{\mathcal{K}_d}^{k+1}$: calculer ses paramètres en utilisant (4.76)-(4.78), ensuite calculer ses moments d’ordre un et d’ordre deux en utilisant l’algorithme de Metropolis-Hastings à marche aléatoire
 6. Retourner à l’étape 2 jusqu’à convergence
-

4.3.4 Simulations

Nous évaluons l’approche proposée (voir l’algorithme 9), appelée BV&MCMC-SR, au travers d’une application au même problème de super-résolution traité dans la partie 4.2.4 qui nous permet de faire des comparaisons avec l’approche BVClassSR et l’approche BV-MGSR. En conséquence, pour la configuration de simulation, on se réfère à la partie 4.2.4. Nous montrons dans la suite des résultats de reconstruction.

Résultats de reconstruction et discussions

Pour avoir une comparaison équitable, nous avons pris la même initialisation pour l’image HR que celle prise par BVClassSR et BV-MGSR : $\mathbf{m}_0 = \mathbf{A}^T \mathbf{y}$ comme la moyenne et 100 comme la variance de chaque pixel de l’image HR. Par rapport aux approches basées sur l’*a priori* TV, cette approche estime en plus deux vecteurs des variables cachées parcimonieuses \mathbf{b}_h et \mathbf{b}_v . Dans notre simulation, le vecteur \mathbf{b}_h est initialisé par un vecteur parcimonieux \mathbf{b}_h^0 obtenu à partir de $\mathbf{D}_h \mathbf{m}_0$ de façon suivante : garder les cinq pour-cent plus grands éléments de $\mathbf{D}_h \mathbf{m}_0$ et mettre tous les autres éléments à zéro. L’initialisation de \mathbf{b}_v est obtenue à partir de $\mathbf{D}_v \mathbf{m}_0$ de la même façon. En ce qui concerne les hyper-paramètres, γ_n^0 est calculé à partir de l’initialisation pour l’image HR en utilisant son équation de mis à jour, \mathcal{K}_d est initialisé par $\sqrt{\gamma_n^0}$.

Dans l’approche proposée BV&MCMC-SR, nous avons deux paramètres à choisir selon nos connaissances *a priori* : ν et ξ . Pendant toutes les simulations, nous mettons le paramètre de forme ν à 0.8 pour avoir une distribution à queue lourde. Pour le paramètre ξ , il dépend du nombre d’entrées non nulles de \mathbf{b}_h et \mathbf{b}_v . En pratique, en raison des

4.3.4 - Simulations

étapes de seuillage doux (voir (4.73) et (4.74)), la plupart des entrées de \mathbf{b}_h et \mathbf{b}_v sont forcées à zéros. Par conséquent, ξ doit prendre les valeurs petites. Nous avons effectué une série d'expériences avec des images différentes et les meilleurs résultats sont obtenus avec $\xi = 0.1N$. En conséquence, nous avons mis $\xi = 0.1N$ pour toutes les expériences.

En ce qui concerne la condition d'arrêt de l'algorithme, nous utilisons la même condition que BVClassSR : $\|\mathbf{m}_k - \mathbf{m}_{k-1}\|/\|\mathbf{m}_{k-1}\| < 10^{-5}$ où \mathbf{m}_k et \mathbf{m}_{k-1} représentent l'estimation pour l'image HR à k -ième et à $(k-1)$ -ième itération, respectivement.

Par ailleurs, l'approche BV&MCMC-SR est appliquée dans les mêmes conditions matérielles : Matlab R2013a sur Intel(R) Core(TM) i7-3770 CPU (3.40 GHz) avec 8.0 GB RAM.

Dans la suite, nous présentons d'abord dans le tableau 4.2 des résultats numériques : PSNR des reconstructions obtenues et temps de calculs utilisé, pour les mêmes données que dans la partie 4.2.4.

TABLE 4.2 – Comparaisons des Performances de BVClassSR [BMK11], BV-MGSR et l'approche BV&MCMC-SR en termes de PSNR (dB) et Temps de Calculs (en Secondes).

Data		BVClassSR		BV-MGSR	BV&MCMC-SR	
		PSNR	Temps(s)	Temps(s)	PSNR	Temps(s)
Testpat	5dB	16.84	12.8	2.0	16.93	4.7
	15dB	20.96	4.1	1.1	20.83	4.1
	25dB	25.55	5.3	1.2	25.61	4.6
	35dB	29.85	10.7	1.4	30.60	4.7
	45dB	32.60	19.3	1.6	34.05	5.1
Camera-man	5dB	23.24	51.9	4.9	23.37	12.4
	15dB	28.70	7.1	1.6	28.65	5.2
	25dB	33.36	4.6	1.5	32.43	5.2
	35dB	37.40	7.8	1.6	36.34	6.2
	45dB	40.50	13.5	2.3	39.83	6.6
Lena	5dB	27.08	87.2	17.7	27.25	28.5
	15dB	31.35	17.0	6.5	31.09	14.6
	25dB	34.59	16.1	5.4	33.94	14.7
	35dB	37.07	26.3	5.9	36.68	15.8
	45dB	38.44	40.0	6.7	38.55	16.5
Jetplane	5dB	32.91	72.7	13.7	33.20	20.6
	15dB	37.33	15.6	5.6	36.97	10.0
	25dB	41.21	15.9	5.4	40.44	9.7
	35dB	44.49	25.5	6.5	43.49	9.6
	45dB	46.60	38.8	7.5	45.79	10.2

Les valeurs PSNR des reconstructions obtenues par BVClassSR et BV-MGSR sont données dans la troisième colonne du tableau 4.2 et les valeurs PSNR obtenues par BV&MCMC-SR sont données dans la sixième colonne. Dans le tableau 4.2, nous marquons la valeur du PSNR la plus haute en rouge pour chaque cas. En comparant les valeurs PSNR, nous pouvons voir que pour ces données testées, les approches basées sur l'*a priori* TV (BVClassSR et BV-MGSR) donnent des reconstructions de valeurs PSNR plus grande que BV&MCMC-SR basé sur l'*a priori* de type Geman dans la plupart des

cas, mais la différence n'est pas grande. En fait, la plus grande différence présente est dans le cas où $\text{SNR} = 35$ dB pour l'image *Cameraman*. Dans ce cas, la valeur PSNR obtenue par BVClassSR et BV-MGSR est 2.83% plus grande que celle obtenue par BV&MCMC-SR. En moyenne, la valeur PSNR obtenue par BVClassSR et BV-MGSR est 0.43% plus grande que celle de BV&MCMC-SR. En conséquence, nous pouvons conclure que, pour ces données testées, BV&MCMC-SR donne des reconstructions assez comparables avec celles obtenues par BVClassSR et BV-MGSR. En plus, nous remarquons que dans le cas où $\text{SNR} = 5$ dB, c'est toujours BV&MCMC-SR qui donne des reconstructions de PSNR un peu plus hautes que celles obtenues par BV-MGSR.

Par ailleurs, pour l'image *Testpat* qui est la plus structurée des images testées, l'approche BV&MCMC-SR est meilleure que les approches basées sur TV dans les cas où $\text{SNR} = 5, 25, 35$ et 45 dB. Même si la différence n'est pas grande, ce groupe de résultats suggère que l'approche BV&MCMC-SR pourrait être bien adaptée aux images structurées comme *Testpat*.

Nous comparons ensuite le temps de calcul utilisé pour ces trois approches (voir la 4ème, 5ème et 7ème colonnes). Nous pouvons voir que l'approche BV&MCMC-SR est plus rapide que BVClassSR pour presque tous les cas (une seule exception : l'image *Cameraman* avec $\text{SNR} = 25$ dB). En moyenne, l'approche BV&MCMC-SR est deux fois plus rapide que BVClassSR. En comparant BV-MGSR et BV&MCMC-SR, nous pouvons voir que l'approche BV-MGSR est toujours plus rapide que BV&MCMC-SR. Ceci est probablement dû aux faits suivants : premièrement, l'approche BV&MCMC-SR estime en plus les variables cachées \mathbf{b}_h et \mathbf{b}_v qui sont de la même dimension que l'image à estimer \mathbf{x} , deuxièmement, pour estimer \mathcal{K}_d , la méthode de Métropolis-Hastings à marche aléatoire est utilisée, ce qui augmente le temps de calcul de chaque itération.

Pour comparer les qualités visuelles des reconstructions, nous montrons dans la figure 4.7 – 4.10 les images HR obtenues par BV-MGSR et BV&MCMC-SR. Pour les cas où $\text{SNR} = 15, 25$ et 35 dB, nous ne voyons pas trop de différence entre les images reconstruites par les deux approches. Mais dans le cas très bruité où $\text{SNR} = 5$ dB, les images obtenues par BV&MCMC-SR sont un peu plus lisses que celles obtenues par BV-MGSR.

Observant la loi *a priori* à variables cachées de type Geman (4.49), nous pouvons voir que deux variables cachées différentes (\mathbf{b}_h et \mathbf{b}_v) ont été introduites pour les différences des images dans deux directions différentes : horizontale et verticale. Néanmoins, pour l'*a priori* TV, les mêmes paramètres ($\boldsymbol{\lambda}$) ont été utilisés pour les deux directions (voir (4.12)). En conséquence, un avantage potentiel de l'*a priori* à variables cachées de type Geman est de mieux décrire les informations de différences dans les directions horizontales et verticales. Cet avantage a été partiellement révélé par les simulations avec l'image *Testpat* montrées précédemment. Pour démontrer cet avantage, nous donnons dans la suite des résultats de simulation avec une image *Mire* (voir figure 4.11). Nous pouvons voir que la plupart des contours de l'image *Mire* sont horizontaux et verticaux.

Des données (images à basse résolution (BR)) sont générées de la même façon que précédemment (voir partie 4.2.4 pour la configuration des simulations). L'objectif ici est de comparer les effets des deux *a priori* sur les qualités des reconstructions. Pour ce faire, nous montrons dans le tableau 4.3 des valeurs de PSNR des images HR obtenues par l'approche BV-MGSR et l'approche BV&MCMC-SR. Dans le tableau 4.3, nous pouvons voir que pour tous les cas, l'approche BV&MCMC-SR donne des reconstructions de PSNR plus grande que l'approche BV-MGSR. En moyenne, la valeur PSNR de l'approche BV&MCMC-SR

4.3.4 - Simulations

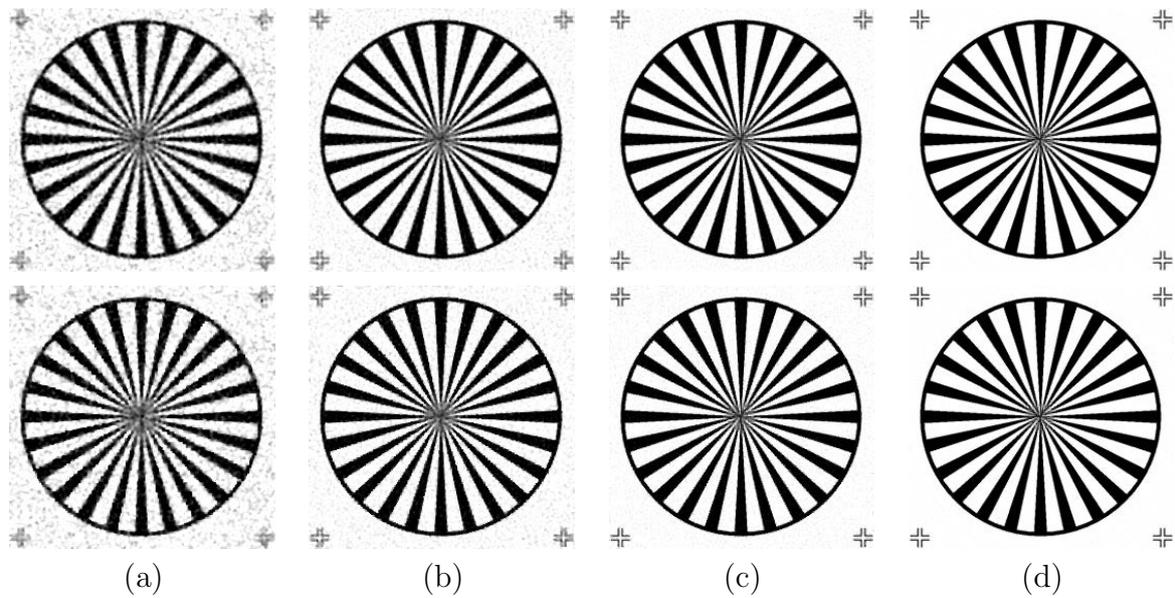


FIGURE 4.7 – Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour *Testpat* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.

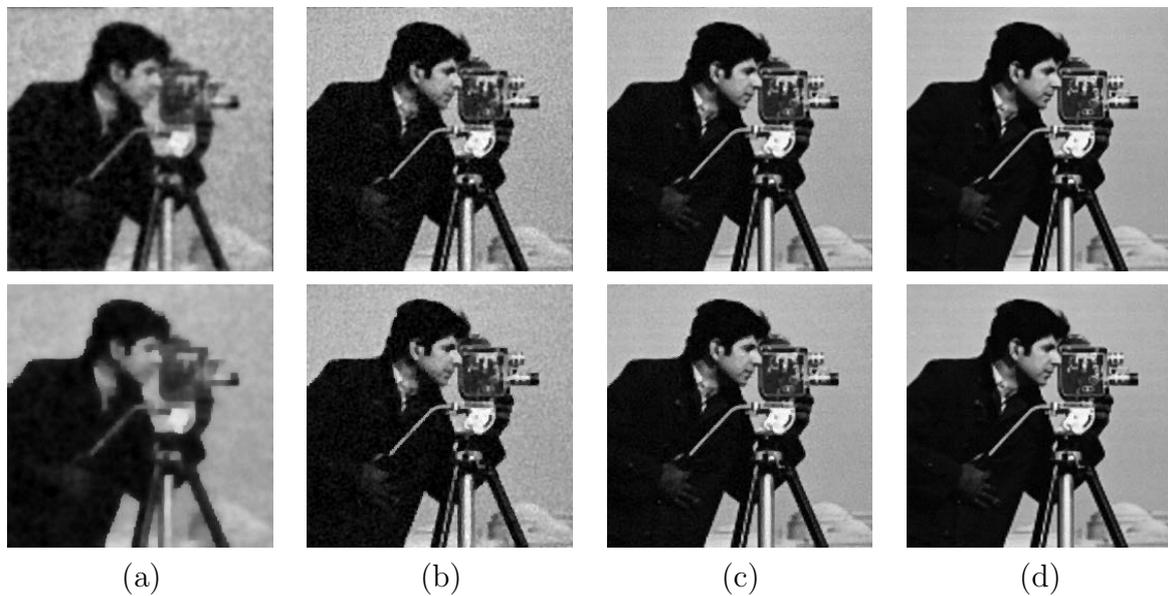


FIGURE 4.8 – Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour *Cameraman* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.

est 2.19 dB (7.75%) plus grande que la valeur PSNR obtenue par l'approche BV-MGSR.

Afin de comparer les qualités visuelles des reconstructions, nous montrons dans la figure 4.12 des images BR (la première ligne), des images HR obtenues par l'approche



FIGURE 4.9 – Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour *Lena* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.

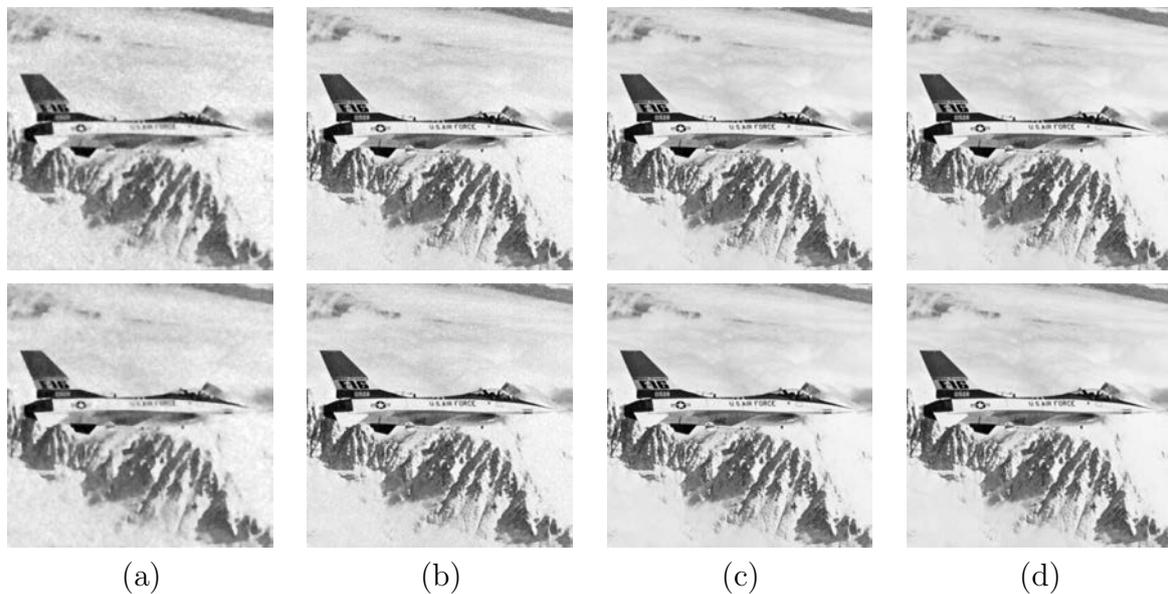


FIGURE 4.10 – Exemples des images HR obtenues par BV-MGSR (la première ligne) et BV&MCMC-SR (la deuxième ligne) pour *Lena* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.

BV-MGSR (la deuxième ligne) et des images HR obtenues par l'approche BV&MCMC-SR (la troisième ligne). En comparant des images BR et des images HR reconstruites, nous pouvons voir que soit l'approche BV-MGSR basée sur l'*a priori* TV, soit l'approche BV&MCMC-SR basée sur l'*a priori* à variables cachées de type Geman permet d'aug-

4.3.4 - Simulations

menter la résolution des images : les bruits présents dans les images BR sont réduits et les contours des images deviennent plus claires. Les images montrées dans la figure 4.12 (a) et (b) sont obtenues dans les cas où les bruits sont les plus forts : SNR = 5 et 15 dB. En comparant les images de la deuxième ligne et de la troisième ligne, nous pouvons voir qu'il y a moins de bruits dans les images de la troisième ligne que les images dans la deuxième ligne. Les figures 4.12 (c) et (d) montrent les résultats au SNR = 25 et 35 dB. Dans le cas, les qualités visuelles des reconstructions obtenues par les deux approches sont comparables.

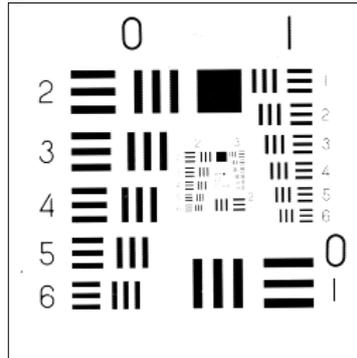


FIGURE 4.11 – Image *Mire* à haute résolution de dimension 256×256 .

TABLE 4.3 – Comparaisons des Performances de l'*a priori* TV (BV-MGSR) et l'*a priori* de type Geman (BV&MCMC-SR) en termes de PSNR (dB).

Data		PSNR (dB)	
		BV-MGSR	BV&MCMC-SR
Mire	5dB	18.83	20.13
	15dB	23.03	24.60
	25dB	27.81	29.94
	35dB	32.63	35.39
	45dB	35.99	39.20

4.4 Conclusion

Dans ce chapitre, nous avons proposé deux approches non-supervisées, toutes les deux utilisant l'approximation BV-MG mais basées sur deux lois *a priori* différentes introduisant une information de régularité par morceau de l'image d'intérêt : l'une sur la variation totale (TV) et l'autre sur la loi à variables cachées de type Geman. Nous avons montré sur un exemple de super-résolution que nos approches sont plus rapides que les approches de l'état de l'art. Nous avons aussi comparé nos deux approches basées sur deux *a priori* différents. Les résultats de simulations montrent que la loi *a priori* à variables cachées de type Geman peut être bien adaptée aux images possédant de contours horizontaux et verticaux très parcimonieux comme *Mire*, mais pour les images "naturelles" comme *Lena* ou *Jetplane*, c'est l'*a priori* TV qui est meilleur. L'*a priori* à variables cachées de type Geman peut être donc très intéressant pour les problèmes, par exemple, de contrôle

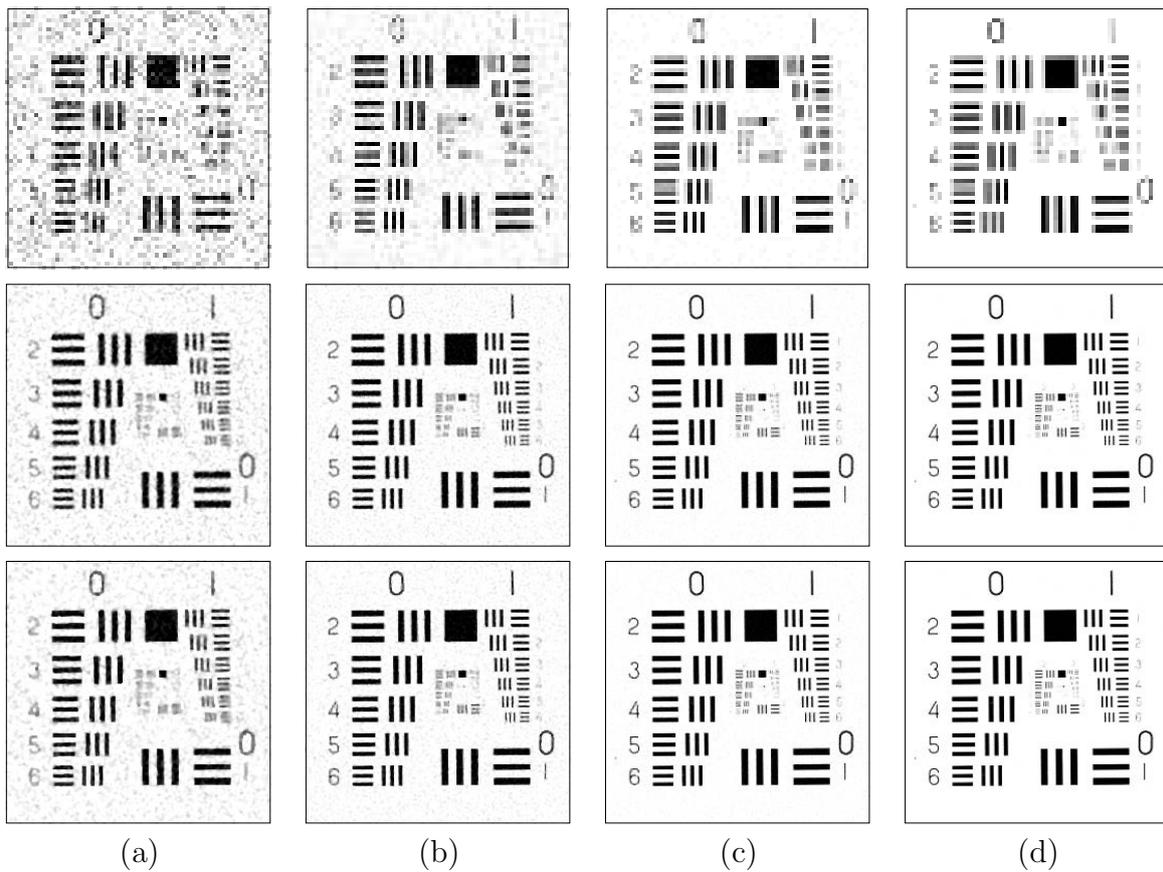


FIGURE 4.12 – Exemples des images BR (ligne en haut), exemples des images HR obtenues par l’approche BV-MGSR (la deuxième ligne), l’approche BV&MCMC-SR (la troisième ligne) pour *Mire* dans les cas où SNR = (a) 5 dB ; (b) 15 dB ; (c) 25 dB (d) 35 dB. Toutes les images sont présentées dans la même gamme de niveaux de gris.

non-destructif où la partie texture de l’objet d’intérêt est moins importante, donc très régulière.

5

Applications dans le domaine des ondelettes

5.1 Introduction

La résolution de problèmes inverses mal-posés s'appuie sur l'introduction d'informations supplémentaires sur les paramètres à estimer à l'aide d'un terme de régularisation ou d'une loi *a priori*, soit dans le domaine spatial comme nous l'avons fait dans les chapitres précédents, soit dans un domaine transformé (Fourier, ondelettes, par exemple) [SA96, ATB03, PSWS03, BD06]. Les ondelettes ont été beaucoup utilisées en traitement du signal et des images [Mal00]. La raison principale est que les caractéristiques de beaucoup de signaux/images naturelles peuvent être représentées plus simplement dans le domaine des ondelettes (Cette transformée permet de concentrer l'énergie du signal sur une faible proportion des coefficients). En traitement d'image, un type d'information essentiel est les contours des images. Dans le domaine spatial, pour bien reconstruire les contours, les lois *a priori* compliquées introduisant les informations de parcimonie sur les gradients des images, par exemple la loi basée sur la variation totale ou la densité à variables cachées de type Geman qui ont été utilisées dans le chapitre 4, sont généralement nécessaires. Néanmoins, dans le domaine des ondelettes, les contours sont simplement représentés par les grands coefficients et les coefficients d'ondelettes des images naturelles présentent un faible degré de corrélations, ce qui rend une hypothèse d'indépendance admissible. En conséquence, l'utilisation d'une loi séparable pour les coefficients permet d'apporter des informations désirées sur l'image originale.

Ces transformations sont particulièrement bien adaptées aux images qui sont régulières par morceau : elles sont constituées généralement en zones lisses séparées par des bords francs. Comme les ondelettes plébiscitées sont celles qui ont leurs premiers moments nuls, les zones possédant une régularité polynomiale donnent des coefficients proches de zéros [Mal00] et quelques coefficients peuvent prendre des valeurs significatives correspondant aux contours présents dans les images. Les coefficients sont donc parcimonieux. Dans ce cas, on peut introduire des informations de parcimonie sur les coefficients dans la

résolution de problèmes inverses. Pour ce faire, un terme de régularisation défini par la norme L^1 sur les coefficients d'ondelettes a été beaucoup utilisé, par exemple dans le travail de Donoho [Don95] où il introduit la méthode de seuillage doux. Pour avoir des reconstructions de meilleure qualité, d'autres termes de régularisation plus compliqués ont été aussi utilisés pour introduire les informations de parcimonie, par exemple, dans [DDDM04, EMSZ07], la norme L^p a été étudiée, ce qui a conduit à des méthodes de seuillage plus compliquées.

Néanmoins, il est difficile de faire un bon choix des hyperparamètres qui contrôlent le compromis entre la fidélité aux données et la confiance aux informations *a priori*, et dans un autre formalisme règlent le seuil utilisé. De plus, il est évident qu'un schéma optimal de seuillage est de déterminer un seuil par type de coefficients (échelle, direction). Dans ce cas, la détermination d'un nombre important de seuils augmente fortement la complexité. Grâce au travail de Stein [Ste81], dans un problème de débruitage où le bruit est *i.i.d.* gaussien, on peut utiliser l'estimateur de l'erreur quadratique moyenne non biaisé *Stein's unbiased risk estimate (SURE)* [DJ95, BL07, PL97, BBP05] afin de déterminer les seuils, ce qui donne de très bons résultats. Le critère SURE a été généralisé pour les problèmes inverses linéaires plus généraux (GSURE) [Eld09] : l'opérateur \mathbf{A} peut être quelconque, de plus, la loi du bruit \mathbf{n} n'est plus forcément *i.i.d.* et peut être une loi quelconque dans la famille exponentielle. Néanmoins, lorsque les problèmes sont mal-posés et de grande dimension, le calcul du GSURE est coûteux voire impossible. Il y a deux difficultés, il faut calculer des pseudo-inverses de matrice de grande taille et l'augmentation de nombre de seuils à déterminer va significativement augmenter le coût de calcul. Dans un cas particulier de problème linéaire – un problème de déconvolution, Pesquet *et al.* ont proposé une approche plus efficace pour calculer GSURE [PBBC09] à l'aide de la transformée de Fourier. Mais cette approche est limitée au problème de déconvolution.

Pour contourner ces problèmes, nous travaillons dans le cadre bayésien comme dans les chapitres précédents. En fait, récemment, de plus en plus de travaux proposent les approches bayésiennes dans le domaine des ondelettes [ATB03, BD06, SA96] car le cadre bayésien offre une grande richesse et flexibilité pour modéliser une grande variété d'informations incluant la parcimonie. Dans le cadre bayésien, l'utilisation de la loi de Laplace correspond à la régularisation par la norme L^1 . D'autres densités à queue lourde telles que la loi de Student [Vid98, SN11], la loi gaussienne généralisée [Mal89, ML99, SA96] et les distributions alpha-stables [ATB03] ont été utilisées avec succès. En effet, ces trois types de distributions appartiennent à une même famille de distribution : la famille de mélange de gaussiennes par changement d'échelle (GSM). Plusieurs travaux ont donc proposé des approches générales basées sur les GSM [BD06, PSWS03]. Néanmoins, dans ces approches, les hyperparamètres sont généralement déterminés de façons empiriques.

Dans ce travail, nous considérons aussi le modèle GSM pour l'*a priori* des paramètres d'intérêt. Néanmoins, nous profitons du cadre bayésien afin de développer des approches entièrement automatiques en estimant conjointement les hyperparamètres et les paramètres d'intérêt. Contrairement aux approches précédentes où la détermination de plusieurs seuils nécessite un coût de calcul important, dans le cadre bayésien, on peut déterminer beaucoup d'hyperparamètres sans trop augmenter le coût de calcul. Comme nous l'avons évoqué plusieurs fois dans cette thèse, la loi *a posteriori* dans ce cas non-supervisé est généralement complexe et ne peut pas être utilisée directement. Pour résoudre ce problème, nous recourons à l'approche d'approximation BV-MG qui a été introduit dans le chapitre 3.

Dans la suite de ce chapitre, nous présentons d'abord deux approches par seuillage de l'état de l'art dont le seuil est déterminé automatiquement en minimisant le critère SURE ou GSURE. Ensuite, nous développons un algorithme de reconstruction non-supervisé général pour des lois *a priori* de la famille des GSM à l'aide de l'approche d'approximation BV. Ensuite, nous étudions deux cas particulier de GSM : la loi de Student et la loi gaussienne généralisée. Finalement, nous montrons des résultats de l'application des approches non-supervisées basées sur la loi de Student et la loi gaussienne généralisée à deux problèmes inverses linéaires classiques : débruitage et déconvolution. Les approches sont évaluées en comparant avec des résultats obtenus par des méthodes de seuillage.

5.2 Énoncé du problème

Notons l'opérateur de la transformée directe par Φ^T et son opérateur adjoint par Φ . Nous allons présenter ce travail dans un cadre général où Φ^T est une transformation quelconque.

Les coefficients correspondant à la transformée directe sont notés par le vecteur $\mathbf{c} \in \mathbb{R}^J$. En conséquence, nous avons

$$\mathbf{c} = \Phi^T \mathbf{x}, \quad (5.1)$$

où $\mathbf{x} \in \mathbb{R}^N$, comme défini dans les chapitres précédents, représente le vecteur des paramètres d'intérêt et l'opérateur $\Phi^T \in \mathbb{R}^{J \times N}$.

Nous considérons dans ce travail un modèle linéaire comme ci-dessous :

$$\mathbf{y} = \mathbf{A}\Phi\mathbf{c} + \mathbf{n}. \quad (5.2)$$

Dans ce cas, l'estimation de l'objet d'intérêt se fait en deux étapes : une étape d'estimation des coefficients \mathbf{c} et une étape de reconstruction de \mathbf{x} communément appelée étape de synthèse : $\mathbf{x} = \Phi\mathbf{c}$.

5.3 Approches de l'état de l'art

Le problème d'estimation de \mathbf{c} est généralement mal-posé. Une classe des approches populaires cherche une estimation de \mathbf{c} en minimisant un critère pénalisé comme suit :

$$J(\mathbf{c}) = \frac{1}{2} \|\mathbf{y} - \mathbf{A}\Phi\mathbf{c}\|^2 + \lambda\rho(\mathbf{c}) \quad (5.3)$$

où le terme de régularisation $\rho(\mathbf{c})$ est choisi pour introduire les informations de parcimonie et le paramètre λ est automatiquement choisi en minimisant l'estimation SURE (problème de débruitage) ou GSURE (problème inverse linéaire général). Dans la suite, nous présentons deux types d'approches.

5.3.1 Approche *SUREShrink*

Pour résoudre un problème de débruitage (\mathbf{A} est alors la matrice identité), Donoho et Johnstone [DJ94, DJ95] ont proposé l'approche *SUREShrink* qui s'appuie sur un seuillage

5.3.2 - Approche par GSURE Shrinkage itérative

doux :

$$\hat{\mathbf{c}} = \eta_\lambda(\tilde{\mathbf{y}}) = \text{sgn}(\tilde{\mathbf{y}})\max(\tilde{\mathbf{y}} - \lambda, 0), \quad (5.4)$$

où η_λ représente l'opérateur de seuillage doux avec un seuil noté par λ , $\hat{\mathbf{c}}$ est l'estimation des coefficients et $\tilde{\mathbf{y}} = \Phi^T \mathbf{y}$ représente les coefficients de la transformé des données. En effet, cet estimateur donné par le seuillage doux est le minimiseur du critère pénalisé $J(\mathbf{c})$ lorsque $\rho(\mathbf{c}) = \|\mathbf{c}\|_1$ [DDDM04].

En ce qui concerne le seuil λ , la valeur optimale serait celle minimisant l'erreur quadratique moyenne (EQM) entre l'estimation proposée $\hat{\mathbf{c}}$ et les vrais coefficients \mathbf{c} . Néanmoins, en pratique, les vrais coefficients ne sont pas connus. Pour s'affranchir de cette difficulté, l'approche *SUREShrink* détermine le seuil en testant des valeurs dans un intervalle assez large et prenant la valeur minimisant le *Stein's unbiased risk estimate* (SURE) [Ste81], qui est une estimation non-biaisée de l'EQM dans un problème de débruitage avec un bruit gaussien.

Pour le seuillage doux $\eta_\lambda(\tilde{\mathbf{y}})$, nous montrons son SURE dans la suite (les détails de calcul peuvent être aussi trouvés dans [DJ95]). Notons $g(\tilde{\mathbf{y}}) = \eta_\lambda(\tilde{\mathbf{y}}) - \tilde{\mathbf{y}}$ et son i -ième composante $g_i(\tilde{\mathbf{y}})$. Le SURE pour le seuillage doux est calculé comme :

$$\begin{aligned} SURE(\lambda) &= J\sigma_n^2 + \|g(\tilde{\mathbf{y}})\|^2 + 2\sigma_n^2 \sum_{i=1}^J \frac{\partial g_i(\tilde{\mathbf{y}})}{\partial \tilde{y}_i} \\ &= J\sigma_n^2 + \|\hat{\mathbf{c}} - \tilde{\mathbf{y}}\|^2 + 2\sigma_n^2 \sum_{i=1}^J \frac{\partial(\eta_\lambda(\tilde{\mathbf{y}}) - \tilde{\mathbf{y}})_i}{\partial \tilde{y}_i} \\ &= J\sigma_n^2 + \|\hat{\mathbf{c}} - \tilde{\mathbf{y}}\|^2 + 2\sigma_n^2(\|\hat{\mathbf{c}}\|_0 - J) \\ &= \|\hat{\mathbf{c}} - \tilde{\mathbf{y}}\|^2 + 2\sigma_n^2\|\hat{\mathbf{c}}\|_0 - J\sigma_n^2 \end{aligned} \quad (5.5)$$

où $\|\hat{\mathbf{c}}\|_0$ est égal au nombre d'éléments non nuls du vecteur $\hat{\mathbf{c}}$.

Nous pouvons voir dans (5.5) que pour calculer le SURE du seuillage doux, il faut connaître la variance du bruit σ_n^2 . En pratique, cette variance est estimée à partir des données \mathbf{y} . Un estimateur classique pour σ_n est donnée par

$$\sigma_n = \frac{MAD}{0.6745} \quad (5.6)$$

avec MAD le *median absolute deviation* des coefficients d'ondelettes à l'échelle la plus fine [DJ94].

5.3.2 Approche par GSURE *Shrinkage* itérative

L'approche *SUREShrink* est valide seulement pour les problèmes de débruitage avec un bruit *i.i.d.* gaussien. Dans la littérature, des approches ont été proposées pour résoudre les problèmes inverses plus généraux où la matrice \mathbf{A} n'est plus une matrice identité.

Pour minimiser le critère (5.3), Daubechies *et al.* [DDDM04] ont proposé une approche itérative qui inclut une opération de seuillage élément par élément à chaque itération. Cette approche s'appuie sur la formule itérative suivante :

$$\mathbf{c}^{k+1} = S_{\rho, \alpha\lambda} \left(\alpha(\mathbf{A}\Phi)^T (\mathbf{y} - \mathbf{A}\Phi\mathbf{c}^k) + \mathbf{c}^k \right) \quad (5.7)$$

où \mathbf{c}^k et \mathbf{c}^{k+1} représente l'estimation à k -ième et à $(k+1)$ -ième itération, respectivement, $S_{\rho, \alpha\lambda}$ est un opérateur de seuillage dépendant de ρ avec le seuil $\alpha\lambda$. Le paramètre α dépend de l'opérateur $\mathbf{A}\Phi$.

Néanmoins, dans [DDDM04], le paramètre λ est choisi empiriquement. Ce type d'approches a ensuite été étudiée par Giryes *et al.* dans [GEE11] et une méthode de réglage automatique de paramètre λ a été proposée. Cette méthode de réglage s'appuie sur le *projected GSURE* qui a été développé dans [Eld09] pour les problèmes inverses où la matrice $\mathbf{A}^T\mathbf{A}$ n'est pas inversible.

Notons l'estimateur par $h(\mathbf{y})$. Dans le cas où la matrice \mathbf{A} est de plein rang colonne, le GSURE est donné par Eldar [Eld09] comme

$$GSURE(\lambda) = C_1 + \|h(\mathbf{u})\|_2^2 - 2\mathbf{x}_{ML}^T h(\mathbf{u}) + 2 \sum_{i=1}^N \frac{\partial h_i(\mathbf{u})}{\partial u_i} \quad (5.8)$$

où C_1 est une constante indépendante de $h(\mathbf{u})$, $\mathbf{u} = (1/\sigma_n^2)\mathbf{A}^T\mathbf{y}$ et $\mathbf{x}_{ML} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}$ est l'estimateur du maximum du vraisemblance de \mathbf{x} (la solution des moindres carrés).

Dans le cas où \mathbf{A} n'est pas de plein rang colonne, le GSURE ci-dessus n'est plus valable. Pour résoudre ce problème, le *projected GSURE* a été proposé. Le *projected GSURE* est calculé avec une même formule que (5.8) mais appliqué sur une projection de l'estimateur $\mathbf{P}h(\mathbf{u})$ où l'opérateur de projection \mathbf{P} donné par

$$\mathbf{P} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^\dagger\mathbf{A}. \quad (5.9)$$

Ici, \dagger représente la pseudo-inverse. Dans ce cas où $\mathbf{A}^T\mathbf{A}$ n'est pas inversible, l'estimateur de maximum de vraisemblance est choisi comme la solution des moindres carrés à norme euclidienne minimale :

$$\mathbf{x}_{ML} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^\dagger\mathbf{y}. \quad (5.10)$$

Grâce à GSURE et *projected GSURE*, le paramètre de la méthode pour les problèmes inverses plus généraux (\mathbf{A} n'est pas une matrice identité) peut être automatiquement réglé. Néanmoins, ses applications aux problèmes inverses mal-posés de grande dimension sont limitées. Premièrement, la matrice \mathbf{A} est généralement non-inversible ou mal-conditionnée. En conséquence, le *projected GSURE* doit être utilisé. Néanmoins, le calcul de ce critère consiste à calculer la pseudo inverse de la matrice $\mathbf{A}\mathbf{A}^T$ (l'opérateur de projection \mathbf{P} et \mathbf{x}_{ML}) et ce calcul de la pseudo inverse est très coûteux voire impossible lorsque le problème est de grande taille. Par ailleurs, la pseudo inverse dépend d'une valeur de tolérance qui doit être proprement choisi selon le problème traité. Enfin, comme dans le critère SURE, le calcul de *projected GSURE* nécessite de connaître la variance du bruit.

En fait, le réglage discuté ci-dessus est sur un seul paramètre λ , donc un même seuil pour tous les coefficients. Il est évident qu'un meilleur schéma de seuillage consiste à déterminer un seuil par type de coefficients. Dans ce cas, il est difficile d'adapter l'approche basée sur le GSURE aux problèmes avec plusieurs paramètres de réglage. Le coût de calcul d'une telle approche va augmenter de manière significativement si on se place sur un problème linéaire quelconque. L'application de telles approches va donc être très compliquée lorsqu'on va vouloir traiter des problèmes inverses de grandes dimensions. Par contre, dans le cadre bayésien, nous pouvons définir une approche non-supervisée

5.4.1 - Formulation bayésienne

où on peut déterminer automatiquement plusieurs paramètres différents pour les coefficients différents sans beaucoup augmenter le coût de calcul. En conséquence, nous allons développer notre approche dans le cadre bayésien.

5.4 Approche non-supervisée avec un a priori de la famille GSM

Dans cette partie, nous allons développer une approche non-supervisée générale pour les distributions de la famille GSM. Pour ce faire, l'approximation BV va être utilisée.

5.4.1 Formulation bayésienne

Loi *a priori* pour les coefficients

Nous considérons une loi *a priori* séparable :

$$p(\mathbf{c}) = \prod_{i=1}^J p(c_i). \quad (5.11)$$

Cette hypothèse est valable par exemple dans le domaine des ondelettes [CKM97, ML99, ATB03, BD06]. Néanmoins, les coefficients n'ont pas toujours les mêmes caractéristiques statistiques. Pour prendre en compte cela, nous divisons les coefficients \mathbf{c} en L sous-bandes où les coefficients dans la même sous-bande suivent une même loi et les coefficients dans les sous-bandes différentes ne sont pas identiquement distribués.

Dans cette partie, nous considérons une loi *a priori* de la famille GSM. Notons l'ensemble des indices des coefficients dans la l -ième sous-bande par $(\mathbf{I}_l)_{l=1,\dots,L}$. En conséquence, on écrit la loi *a priori* comme

$$\begin{aligned} p(\mathbf{c}) &= \prod_{l=1}^L \prod_{i \in \mathbf{I}_l} p(c_i) \\ &= \prod_{l=1}^L \prod_{i \in \mathbf{I}_l} \int_{\mathbb{R}} p(c_i | z_i, \gamma_p^l) p^l(z_i) dz_i \\ &= \prod_{l=1}^L \prod_{i \in \mathbf{I}_l} \int_{\mathbb{R}} (2\pi)^{-1/2} \sqrt{z_i \gamma_p^l} e^{-\frac{z_i \gamma_p^l c_i^2}{2}} p^l(z_i) dz_i, \end{aligned} \quad (5.12)$$

où $\mathbf{z} = (z_1, \dots, z_J)$ est le vecteur des variables cachées, $\boldsymbol{\gamma}_p = (\gamma_p^1, \dots, \gamma_p^L)$ est le vecteur des hyperparamètres et $p^l(z_i)$ représente la loi *a priori* de la variable cachée z_i . Ici, toutes les $p^l(z_i)$ appartiennent à une même famille de distribution, mais pour des coefficients dans des sous-bandes différentes, les paramètres de $p^l(z_i)$ prennent des valeurs différentes. En fait, les paramètres de $p^l(z_i)$ sont généralement les paramètres de forme de la loi $p(c_i)$. Par conséquent, en prenant les paramètres de forme différents pour les sous-bandes différentes, nous introduisons les informations de parcimonie différentes pour chaque sous-bande. Comme discuté précédemment, les paramètres de forme sont fixés selon notre

connaissance *a priori* et les paramètres d'échelles γ_p sont estimés conjointement avec les coefficients \mathbf{c} .

Comme nous l'avons indiqué dans le chapitre 3, lorsque $p(c_i)$ est la densité d'une loi de Student (un cas particulier de GSM), on sait que $p^l(z_i)$ est la densité d'une loi Gamma. Dans ce cas, on a développé une approche de reconstruction en estimant conjointement la variable cachée \mathbf{z} et le paramètre d'intérêt à l'aide de l'approximation BV. Néanmoins, pour les autres distributions de la famille de GSM, la densité de la variable cachée $p^l(z_i)$ est généralement inconnue. Dans ce cas, ce n'est pas évident que l'on puisse développer des approches de reconstruction identiques à celle du cas Student. Nous allons montrer que, avec un *a priori* dans la famille GSM, nous pouvons quand même développer des approches de reconstruction sous condition que l'espérance de z_i sous la loi conditionnelle $p(z_i|x_i)$ soit connue.

Loi *a priori* pour les hyperparamètres

Nous avons ici un plus grand nombre d'hyperparamètres : la précision du bruit γ_n ainsi que le vecteur γ_p de taille L ($L = 10$ dans nos simulations). Pour γ_p , on introduit un *a priori* Gamma séparable, donc

$$p(\gamma_n) = \mathcal{G}(\gamma_n | \tilde{a}_n, \tilde{b}_n), \quad (5.13)$$

$$p(\gamma_p) = \prod_{l=1}^L p(\gamma_p^l) = \prod_{l=1}^L \mathcal{G}(\gamma_p^l | \tilde{a}_p^l, \tilde{b}_p^l). \quad (5.14)$$

En mettant les paramètres $(\tilde{a}_n, \tilde{b}_n)$ et $(\tilde{a}_p^l, \tilde{b}_p^l)$ à zéro, nous pouvons obtenir des lois de Jeffreys.

En utilisant la règle de Bayes, nous pouvons en déduire une loi *a posteriori* jointe comme ci-dessous :

$$\begin{aligned} p(\mathbf{c}, \mathbf{z}, \gamma_n, \gamma_p | \mathbf{y}) &\propto \gamma_n^{M/2} \exp \left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\Phi\mathbf{c}\|^2}{2} \right] \\ &\times \prod_{l=1}^L \prod_{i \in \mathbf{I}_l} \sqrt{z_i \gamma_p^l} \exp \left[-\frac{\gamma_p^l}{2} z_i c_i^2 \right] p^l(z_i) \\ &\times \gamma_n^{\tilde{a}_n - 1} \exp \left[-\tilde{b}_n \gamma_n \right] \prod_{l=1}^L (\gamma_p^l)^{\tilde{a}_p^l - 1} \exp \left[-\tilde{b}_p^l \gamma_p^l \right]. \end{aligned} \quad (5.15)$$

Cette loi *a posteriori* est de forme compliquée. Pour s'affranchir de la difficulté de calcul, nous pouvons recourir aux approches d'approximation BV qui nous offre une approximation séparable de la loi *a posteriori*, comme nous avons fait dans les chapitres précédents. Néanmoins, une autre difficulté posée ici est qu'on ne connaît pas l'expression explicite de la densité $p^l(z_i)$ ce qui rend la détermination de la loi *a posteriori* approchante pour \mathbf{z} impossible. Cependant, notre principale préoccupation est d'obtenir une expression explicite de la loi approchante de \mathbf{c} qui nous permet de donner une estimation sur \mathbf{c} . Comme nous le verrons dans la suite, la mise à jour des distributions approchantes de \mathbf{c} nécessite de connaître seulement l'espérance *a posteriori* de \mathbf{z} . En conséquence, le problème posé par la densité inconnue $p^l(z_i)$ peut être résolu en effectuant seulement la mise à jour de l'espérance de \mathbf{z} mais pas sa distribution approchante.

5.4.2 Application des méthodes d'approximation BV

Dans la suite, $\Theta = \{\mathbf{c}, \mathbf{z}, \gamma_n, \gamma_p\}$ représente le vecteur incluant toutes les variables à estimer. On calcule alors la loi approchante séparable q_Θ de la loi *a posteriori* (5.15) en utilisant les méthodes d'approximation BV. Dans cette partie, comme nous avons fait dans les chapitres précédents, nous supposons que

$$\begin{aligned} q_\Theta(\Theta) &= q_{\mathbf{c}}(\mathbf{c})q_{\mathbf{z}}(\mathbf{z})q_{\gamma_n}(\gamma_n)q_{\gamma_p}(\gamma_p) \\ &= \left(\prod_{l=1}^L \prod_{i \in I_l} q_i(c_i) \tilde{q}_i(z_i) \right) q_{\gamma_n}(\gamma_n) \prod_{l=1}^L q_{\gamma_p^l}(\gamma_p^l). \end{aligned} \quad (5.16)$$

De plus, une optimisation alternée par rapport à $q_{\mathbf{z}}$, $q_{\mathbf{c}}$, q_{γ_n} et q_{γ_p} est effectuée. Ici, l'optimisation de $q_{\mathbf{c}}$ est faite avec l'algorithme d'approximation BV-MG et l'optimisation de $q_{\mathbf{z}}$, q_{γ_n} et q_{γ_p} est faite avec l'algorithme d'approximation BVClass.

Lorsqu'un *a priori* GSM est utilisé, la loi conditionnelle $p(\mathbf{c}|\mathbf{z})$ est une loi gaussienne qui est conjuguée avec la vraisemblance gaussienne $p(\mathbf{y}|\mathbf{c}, \gamma_n)$. Par conséquent, on sait que la loi approchante optimale $q_{\mathbf{c}}^{opt}$ est dans la famille gaussienne. De plus, on a choisi les lois *a priori* conjuguées pour les hyperparamètres γ_n et γ_p . Les lois approchantes optimales $q_{\gamma_n}^{opt}$ et $q_{\gamma_p^l}^{opt}$ sont donc dans la famille Gamma. Nous prenons alors

$$q_{\mathbf{c}}^k(\mathbf{c}) = \prod_i \mathcal{N}(c_i | (\mathbf{m}_k)_i, (\boldsymbol{\sigma}_k^2)_i), \quad (5.17)$$

$$q_{\gamma_n}^k(\gamma_n) = \mathcal{G}(\gamma_n | a_{\gamma_n}^k, b_{\gamma_n}^k), \quad (5.18)$$

$$q_{\gamma_p^l}^k(\gamma_p^l) = \mathcal{G}(\gamma_p^l | a_{\gamma_p^l}^k, b_{\gamma_p^l}^k). \quad (5.19)$$

Comme indiqué ci-dessus, pour la plupart des distributions dans la famille GSM, on ne connaît pas l'expression de $p^l(z_i)$. En conséquence, la loi approchante $q_{\mathbf{z}}$ n'est pas explicitement connue non plus car elle dépend de $p(z_i)$, comme on peut le voir dans l'équation suivante. Selon (2.12), on peut obtenir

$$\begin{aligned} \forall i \in I_l, \quad \tilde{q}_i^{k+1}(z_i) &\propto \exp \left[\langle \ln p(\Theta, \mathbf{y}) \rangle_{q_{\mathbf{c}}^k \prod_{j < i} \tilde{q}_j^{k+1} \prod_{j > i} \tilde{q}_j^k q_{\gamma_n}^k q_{\gamma_p^l}^k} \right] \\ &\propto \exp \left(\frac{1}{2} \ln(z_i) - \frac{\langle \gamma_p^l \rangle^k}{2} z_i \langle c_i^2 \rangle^k + \ln p^l(z_i) \right) \\ &\propto \sqrt{z_i} p^l(z_i) \exp \left(-\frac{\langle \gamma_p^l \rangle^k}{2} \langle c_i^2 \rangle^k z_i \right) \\ &= p(z_i | \sqrt{\langle c_i^2 \rangle^k}, \langle \gamma_p^l \rangle^k). \end{aligned} \quad (5.20)$$

Cette loi n'a pas d'expression explicite. Néanmoins, notre objectif n'est pas d'obtenir l'expression explicite de $q_{\mathbf{z}}$, mais d'obtenir $q_{\mathbf{c}}$ qui nous permet de donner une estimation de notre paramètre d'intérêt \mathbf{c} . Comme l'a fait remarquer Palmer dans son travail [PKDRW05], et comme on peut le voir dans l'équation (3.38), il suffit de connaître

l'espérance de $\tilde{q}_i^{k+1}(z_i)$ pour déterminer q_c :

$$\begin{aligned} \forall i \in I_l \quad \langle z_i \rangle^{k+1} &= \int z_i \tilde{q}_i^{k+1}(z_i) dz_i \\ &= \int z_i p \left(z_i | \sqrt{\langle c_i^2 \rangle^k}, \langle \gamma_p^l \rangle^k \right) dz_i \\ &= \mathbb{E}_{p(z_i | \sqrt{\langle c_i^2 \rangle^k}, \langle \gamma_p^l \rangle^k)} \{z_i\}. \end{aligned} \quad (5.21)$$

Dans ce cas, le défi principal est de déterminer cette espérance.

Lorsque $p(c_i)$ est une loi de la famille GSM (5.12) [PKDRW05], on peut avoir

$$\begin{aligned} \forall i \in I_l, \quad p'(c_i) &= \frac{\partial}{\partial c_i} \int_0^\infty p(c_i | z_i, \gamma_p^l) p^l(z_i) dz_i \\ &= - \int_0^\infty \gamma_p^l c_i z_i p(c_i | z_i, \gamma_p^l) p^l(z_i) dz_i \\ &= - \int_0^\infty \gamma_p^l c_i z_i p(z_i | c_i, \gamma_p^l) p(c_i) dz_i = -\gamma_p^l c_i p(c_i) \mathbb{E}_{p(z_i | c_i, \gamma_p^l)} \{z_i\}, \end{aligned} \quad (5.22)$$

qui nous permet d'obtenir

$$\mathbb{E}_{p(z_i | c_i, \gamma_p^l)} \{z_i\} = -\frac{p'(c_i)}{\gamma_p^l c_i p(c_i)}. \quad (5.23)$$

En combinant (5.21) et (5.23), nous pouvons avoir

$$\forall i \in I_l, \quad \langle z_i \rangle^{k+1} = -\frac{p'(c_i)}{\gamma_p^l c_i p(c_i)} \Big|_{c_i = \sqrt{\langle c_i^2 \rangle^k}, \gamma_p = \langle \gamma_p^l \rangle^k}. \quad (5.24)$$

Remarque. *Le calcul de l'espérance a posteriori de z_i ne nécessite pas l'expression explicite de $p^l(z_i)$, mais il nécessite de connaître l'expression explicite de $p(c_i)$.*

Optimisation de q_c avec l'algorithme d'approximation BV-MG

Comme la loi de Student fait partie de la famille GSM, les calculs pour le cas général de GSM devraient être similaires à ceux montrés dans le chapitre 3. En fait, les dérivations des équations de mise à jour des paramètres \mathbf{m}_k et σ_k^2 avec l'algorithme d'approximation BV-MG sont les mêmes que celles données par (3.46) – (3.48). De plus, le calcul de la fonction intermédiaire q_i^{T1} est le même que celui de l'Eq. (3.38) sauf que ici, l'espérance de z_i est donnée par (5.24).

$$\forall i \in I_l, \quad (\sigma_{r_1}^2)_i = [\langle \gamma_n \rangle^k \text{diag}(\Phi^T \mathbf{A}^T \mathbf{A} \Phi)_i + \langle \gamma_p^l \rangle^k \langle z_i \rangle^{k+1}]^{-1}, \quad (5.25)$$

$$(\mathbf{m}_{r_1})_i = (\sigma_{r_1}^2)_i [\langle \gamma_n \rangle^k (\Phi^T \mathbf{A}^T \mathbf{y} - \Phi^T \mathbf{A}^T \mathbf{A} \Phi \mathbf{m}_k + \text{diag}(\Phi^T \mathbf{A}^T \mathbf{A} \Phi) \circ \mathbf{m}_k)]_i. \quad (5.26)$$

Optimisation de q_{γ_n} et q_{γ_p}

Comme q_{γ_n} ne dépend pas de $p(\mathbf{c}, \mathbf{z})$, l'optimisation de q_{γ_n} est exactement la même que dans le cas de Student (voir (3.59)). Les équations de mise à jour pour les paramètres a_{γ_n} et b_{γ_n} sont données par (3.60) et (3.61).

5.4.3 - Cas d'un a priori gaussien généralisé

Pour l'optimisation de $q_{\gamma_p}^l$, le calcul est similaire à celui de l'Eq. (3.63). Les équations de mise à jour des paramètres $a_{\gamma_p}^l$ et $b_{\gamma_p}^l$ peuvent être obtenues à partir de (3.64) et (3.65) en remplaçant la moyenne de z_i par celle obtenue avec (5.24). En conséquence, on a

$$\forall l = 1, \dots, L \quad a_{\gamma_p}^{k+1} = \frac{\text{card}\{\mathbf{I}_l\}}{2} + \tilde{a}_p^l \quad (5.27)$$

$$b_{\gamma_p}^{k+1} = \frac{1}{2} \sum_{i \in \mathbf{I}_l} \langle z_i \rangle^{k+1} ((\mathbf{m}_{k+1})_i^2 + (\boldsymbol{\sigma}_{k+1})_i^2) + \tilde{b}_p^l. \quad (5.28)$$

où $\text{card}\{\mathbf{I}_l\}$ est le cardinal de l'ensemble \mathbf{I}_l , c'est à dire le nombre des éléments de l'ensemble \mathbf{I}_l .

En résumé, nous avons obtenu un algorithme de reconstruction non-supervisé basé sur un *a priori* de GSM en utilisant l'algorithme d'approximation BV-MG (voir l'algorithme 10).

Algorithm 10 L'algorithme de reconstruction non-supervisé basé sur un *a priori* de GSM en utilisant BV-MG (GSMBV-MG)

1. Initialiser q_c^0 , $q_{\gamma_n}^0$ et $q_{\gamma_p}^0$: initialiser \mathbf{m}_0 , $\boldsymbol{\sigma}_0^2$, $a_{\gamma_n}^0$, $b_{\gamma_n}^0$ et $a_{\gamma_p}^0$, $b_{\gamma_p}^0$
 2. Calculer $\langle z_i \rangle^{k+1}$ en utilisant (5.24)
 3. Calculer $q_c^{k+1} = \prod_i q_i^{k+1}$: pour $i = 1, \dots, N$
 - (a) Calculer $q_i^{r_1}$: calculer ses paramètres en utilisant (5.25) et (5.26)
 - (b) Calculer le pas sous-optimal selon annexe B.2
 - (c) Calculer q_i^{k+1} : calculer ses paramètres en utilisant (3.47) et (3.48) où le pas \mathbf{s} est égal au pas sous-optimal obtenu à l'étape (b)
 4. Calculer $q_{\gamma_n}^{k+1}$: calculer ses paramètres en utilisant (3.60) et (3.61)
 5. Calculer $q_{\gamma_p}^{k+1}$: calculer ses paramètres en utilisant (5.27) et (5.28)
 6. Retourner à l'étape 2 jusqu'à convergence
-

L'algorithme 10 est utilisable pour un problème inverse linéaire avec n'importe quel *a priori* de la famille GSM dont l'expression explicite de la densité est connue. Cette famille inclut des distributions qui nous permettent d'introduire une grande variété d'informations. Dans la suite, on s'intéresse à des informations de parcimonie. Pour ce faire, on considère deux lois à queue lourde de la famille GSM : la loi de Student et la loi gaussienne généralisée. Les calculs avec la loi de Student ont été faits dans le chapitre 3. Par conséquent, dans la suite, nous détaillerons uniquement les calculs pour la loi gaussienne généralisée.

5.4.3 Cas d'un a priori gaussien généralisé

En étudiant les histogrammes des coefficients de détails des ondelettes de plusieurs images différentes, Mallat [Mal89] a montré que les coefficients détails peuvent être bien modélisés par la loi gaussienne généralisée. Depuis, la loi gaussienne généralisée (GG) a été largement utilisée pour décrire les coefficients d'ondelettes et elle a conduit aux bons résultats [Mal89, ABMD92, SA96, ML99]. Dans cette partie, nous étudions cette loi, plus

spécifiquement, nous adaptions l'algorithme 10 pour la loi GG. Pour ce faire, il suffit de déterminer l'espérance donnée par (5.23).

La densité de probabilité d'une loi GG de moyenne nulle s'écrit comme :

$$GG(c|\gamma_p, \beta) = \frac{\sqrt{\gamma_p}\beta}{2\Gamma(1/\beta)} e^{-|\sqrt{\gamma_p}c|^\beta} \quad (5.29)$$

où $\gamma_p > 0$ est le paramètre d'échelle et $\beta \in (0, 2)$ est le paramètre de forme.

Pour apporter des informations différentes pour les coefficients de sous-bande différente, nous prenons

$$\forall i \in \mathbf{I}_l, \quad p(c_i) = GG(c_i|\gamma_p^l, \beta^l). \quad (5.30)$$

Pour la densité de la loi GG, nous pouvons calculer sa dérivée

$$\begin{aligned} \forall c_i \neq 0, \quad p'(c_i) &= -\frac{\sqrt{\gamma_p}\beta}{2\Gamma(1/\beta)} e^{-|\sqrt{\gamma_p}c_i|^\beta} \gamma_p^{\beta/2} \frac{\beta}{2} (c_i^2)^{\beta/2-1} 2c_i \\ &= -p(c_i) \beta c_i \gamma_p^{\beta/2} (c_i^2)^{\beta/2-1}. \end{aligned} \quad (5.31)$$

En utilisant (5.24) et (5.31), nous pouvons en déduire que

$$\begin{aligned} \langle z_i \rangle^{k+1} &= \beta \gamma_p^{\beta/2-1} (c_i^2)^{\beta/2-1} \Big|_{c_i = \sqrt{\langle c_i^2 \rangle^k}, \gamma_p = \langle \gamma_p \rangle^k} \\ &= \beta (\langle \gamma_p \rangle^k)^{\beta/2-1} [(\mathbf{m}_k)_i^2 + (\boldsymbol{\sigma}_k)_i^2]^{\frac{\beta}{2}-1} \\ &= \beta [\langle \gamma_p \rangle^k ((\mathbf{m}_k)_i^2 + (\boldsymbol{\sigma}_k)_i^2)]^{\frac{\beta}{2}-1} \end{aligned} \quad (5.32)$$

où $\beta = \beta^l, \gamma_p = \gamma_p^l$ pour $i \in \mathbf{I}_l$.

5.5 Résultats de simulation

Nous pouvons remarquer que l'approche développée dans la partie 5.4 est assez générale : premièrement, elle est utilisable pour divers problèmes inverses linéaires, deuxièmement, différentes paires d'opérateurs Φ^T/Φ peuvent être utilisées, troisièmement, nous pouvons exploiter toute la famille GSM pour la loi *a priori*.

Dans ce travail, nous exploitons la loi Student et la loi gaussienne généralisée de la famille GSM pour introduire les informations de parcimonie dans un domaine transformé. Dans la suite, l'approche basée sur la loi de Student est appelée BV-MGSt et l'autre approche utilisant la loi gaussienne généralisée est appelée BV-MGGG. Il est important de souligner que nos approches sont non-supervisées et tous les hyperparamètres sont automatiquement déterminés.

Pour évaluer notre approche, nous montrons dans la suite son application à deux problèmes inverses couramment rencontrés : un problème de débruitage et un problème de déconvolution. En plus, dans toutes les simulations, nous utilisons une frame construite par union de neuf bases d'ondelettes orthogonales déterminées par translation. Cette frame permet d'avoir la propriété d'invariance que la transformée en ondelettes (TO) orthogonale

5.5.1 - Problème de débruitage

ne possède pas. Par ailleurs, les ondelettes Daubechies 4 sont utilisées et nous faisons une décomposition sur 3 niveaux (voir la figure 5.1). Dans ce cas, les coefficients des échelles grossières sont généralement moins parcimonieux que les coefficients des échelles plus fines. En outre, les coefficients à différentes orientations (horizontale, verticale et diagonale) pourraient avoir des caractéristiques différentes. Pour prendre en compte ce fait, nous divisons les coefficients à différentes orientations et à différentes échelles en $L = 10$ sous-bandes différentes. Dans ce cas, on a \mathbf{I}_1 inclut les indices des coefficients dans la case LL3 de la figure 5.1, \mathbf{I}_2 inclut les indices des coefficients dans la case HL3, etc.

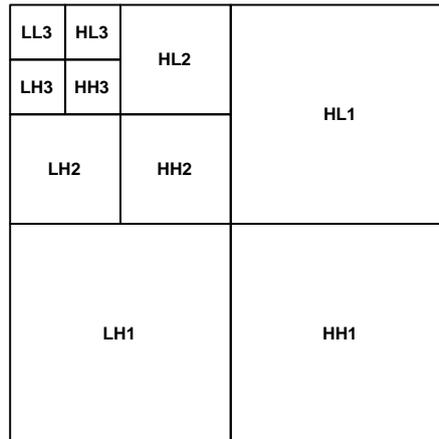


FIGURE 5.1 – Illustration de la décomposition en ondelettes sur 3 niveaux.

5.5.1 Problème de débruitage

En identifiant l'opérateur \mathbf{A} comme une matrice identité, nous pouvons utiliser les approches proposées pour résoudre un problème de débruitage. Pour évaluer nos approches, nous les comparons avec une approche classique : l'approche *SUREShrink* [DJ94, DJ95] qui a été brièvement introduite dans la partie 5.3.1.

Configurations de simulation

Dans les simulations de cette partie, nous testons avec l'image *Lena* de taille 256×256 (voir figure 5.2 (a)). À partir de cette image, on génère une image bruitée (voir la figure 5.2 (b)) en ajoutant un bruit *i.i.d.* gaussien pour obtenir un $\text{SNR} = 20$ dB.

Nous utilisons l'approche *SUREShrink* et les deux approches proposées BV-MGSt, BV-MGGG pour effectuer le débruitage. Pour l'approche *SUREShrink*, la variance du bruit (γ_n^{-1}) a besoin d'être estimée et fixée en avance. Pour comparer avec cette approche, nous avons aussi fixé la variance du bruit dans les deux autres approches. Dans toutes les simulations, la variance du bruit est fixée à sa vraie valeur.

Par ailleurs, nos approches proposées sont mises en œuvre avec les initialisations suivantes : la transformée en ondelettes des données (l'image bruitée) comme moyenne et un comme variance des coefficients à estimer et l'inverse de la vraie variance du bruit comme

l'initialisation de γ_p^l . Pour l'approche BV-MGSt basée sur la loi de Student, nous avons pris les paramètres de forme (ν^l) différents pour les coefficients à différentes échelles, mais pour les coefficients des différentes directions (horizontale, verticale, diagonale) à une même échelle, nous avons pris une même valeur car, pour l'image *Lena*, il n'y a pas de direction privilégiée. En fait, les coefficients d'approximations (correspondent à case LL3 de la figure 5.1) ne sont généralement pas parcimonieux. Par conséquent, nous avons pris 2000 comme paramètre de forme des coefficients d'approximation, qui donne une distribution ressemblant à la loi gaussienne. En plus, [5, 2.5, 1] sont utilisés comme les paramètres de forme des coefficients de détail de la plus grossière à la plus fine échelle afin de prendre en compte le fait que les coefficient à l'échelle plus fine sont généralement plus parcimonieux. Pour l'approche BV-MGGG basée sur la loi gaussienne généralisée, les paramètre de forme sont choisis de la même façon que pour la loi de Student : 2, qui conduit à une loi gaussienne, est utilisé comme paramètre de forme pour les coefficients d'approximation, et [1, 0.9, 0.8], qui conduit à distributions à queue lourde, sont utilisés pour les coefficients de détails de la plus grossière à la plus fine échelle.

Résultats

Les images reconstruites par ces approches sont montrées dans la figure 5.2 (c), (d) et (e). En comparant ces trois images avec la figure 5.2 (b), nous pouvons voir que les trois approches permettent de réduire le bruit présent dans les données. La comparaison de la figure 5.2 (d) et (e) et la figure 5.2 (c) montre que les approches BV-MGSt et BV-MGGG donnent des reconstructions moins lisses que celle obtenue par le *SUREShrink*. Nous donnons aussi des valeurs de PSNR des images reconstruites dans le tableau 5.1 pour avoir une comparaison plus directe. Nous pouvons voir que la valeur PSNR obtenue par *SUREShrink* est un peu plus haute (vers 0.2 dB) que les approches BV-MGSt et BV-MGGG. Nous pouvons voir que l'approche de seuillage doux marche bien pour un problème de débruitage. Mais les résultats de BV-MGSt et BV-MGGG sont assez comparables avec le résultat de *SUREShrink*.

TABLE 5.1 – Les valeurs PSNR des reconstructions obtenues par *SUREShrink*, BV-MGSt et BV-MGGG

	SUREShrink	BV-MGSt	BV-MGGG
PSNR (dB)	30.18	30.04	29.98

Contrairement à l'approche *SUREShrink* où un seul seuil est déterminé, dans nos approches, on détermine dix seuils différents (dix hyperparamètres γ_p^l différents), un pour chaque sous-bande. Nous montrons dans le tableau 5.2 les valeurs des $(\gamma_p^l)_{l=1,\dots,10}$ estimées par l'approche BV-MGSt et dans le tableau 5.3 les valeurs des $(\gamma_p^l)_{l=1,\dots,10}$ estimées par l'approche BV-MGGG. Comme nous avons fixé la valeur du γ_n , plus la valeur de γ_p^l est grande, plus le poids des informations *a priori* (parcimonie) est fort. Dans les deux tableaux 5.2 et 5.3, nous pouvons voir que plus l'échelle est fine, plus la valeur de γ_p^l est grande, c'est à dire qu'on introduit plus d'informations de parcimonie, ce qui est cohérent avec le fait que les coefficients à l'échelle plus fine sont plus parcimonieux que les coefficients à l'échelle plus grossière. Cette observation est cohérente avec les histogrammes des coefficients d'ondelettes de l'image *Lena* montrés dans la figure 5.3. De plus, les valeurs de γ_p^l pour les coefficients diagonaux sont aussi toujours plus grandes que les γ_p^l pour

5.5.1 - Problème de débruitage

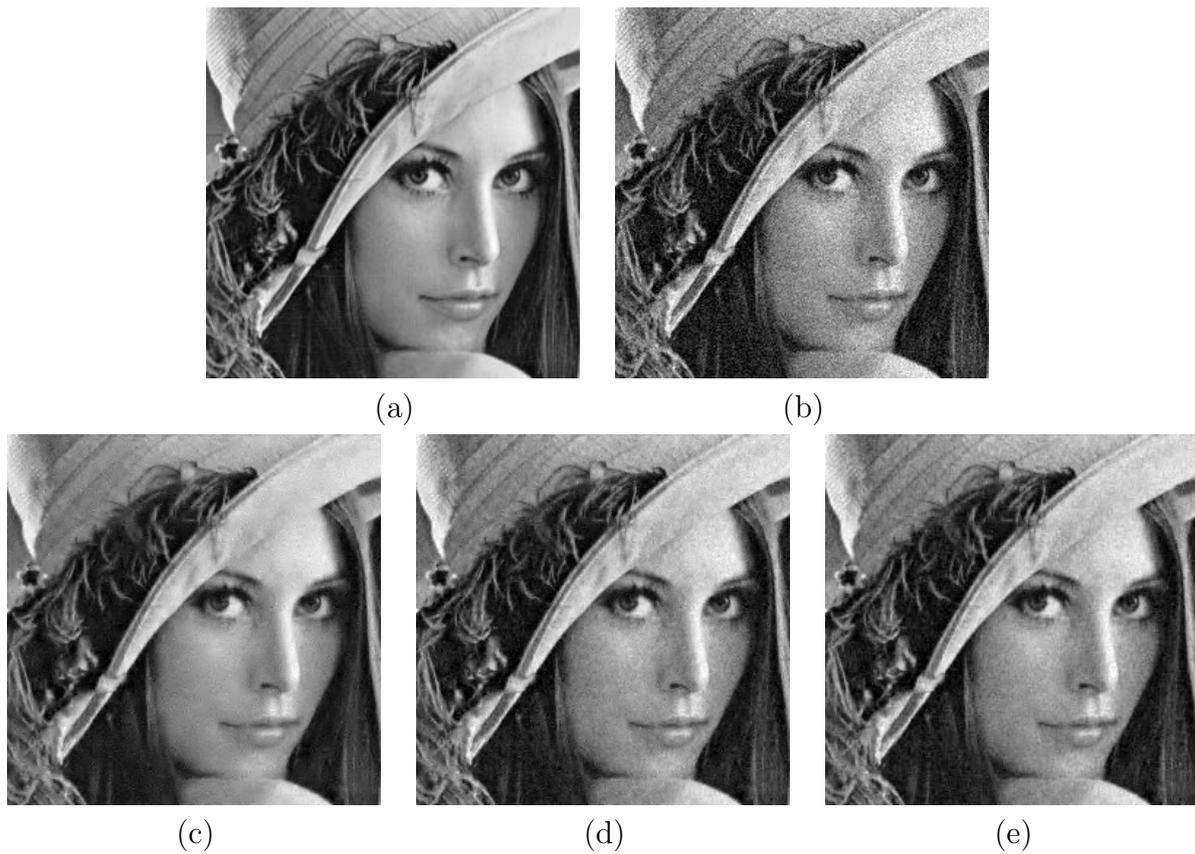


FIGURE 5.2 – (a) L'image vraie, (b) l'image bruitée, SNR = 20 dB, (c) l'approche *SUREShrink*, (d) l'approche BV-MGSt et (e) l'approche BV-MGGG

les deux autres directions. Ceci est aussi cohérent avec le fait que les coefficients de direction diagonale sont plus parcimonieux que les coefficients des deux autres directions pour l'image *Lena* (comme on peut voir dans la figure 5.3 que les histogrammes de la troisième colonne sont plus étroits que les histogrammes dans la première et la deuxième colonnes). En plus, pour les coefficients d'approximation, une petite valeur pour γ_p^l a été trouvée par l'approche BV-MGSt et BV-MGGG. Dans ce cas, le poids des informations *a priori* est faible. Les coefficients d'approximation sont donc principalement déterminés par les données. Tout cela montre les bonnes propriétés de notre approche pour régler le compromis entre l'information présente dans les données et les informations *a priori*.

TABLE 5.2 – Hyperparamètres γ_p^l estimés par l'approche BV-MGSt. Échelle de la plus grossière à la plus fine : $3 \rightarrow 1$.

	Approximation	Échelle	Horizontale	Verticale	Diagonale
γ_p^l	0.06	3	10.82	23.30	31.60
		2	131.91	231.11	369.21
		1	776.26	1.11e+03	1.59e+03

TABLE 5.3 – Hyperparamètres γ_p^l estimés par l’approche BV-MGGG. Échelle de la plus grossière à la plus fine : 3 \rightarrow 1.

	Approximation	Échelle	Horizontale	Verticale	Diagonale
γ_p^l	0.03	3	19.09	40.93	54.29
		2	130.33	233.47	392.09
		1	898.60	1.34e+03	2.07e+03

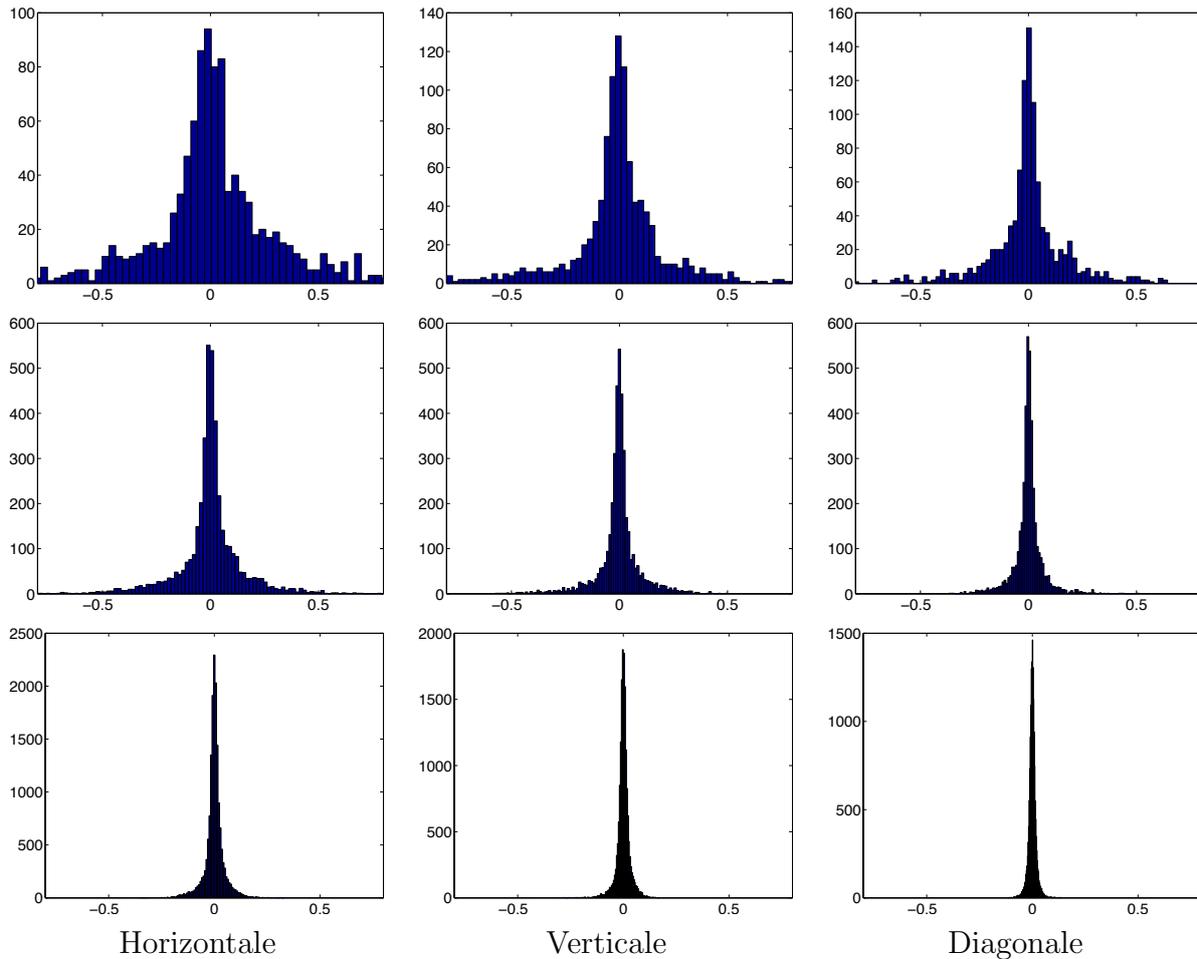


FIGURE 5.3 – Histogrammes des coefficients d’ondelettes de *Lena* à différente sous-bande. Les lignes de la plus haut à la plus bas correspondent aux échelles de la plus grossière à la plus fine.

5.5.2 Problème de déconvolution

Un problème de débruitage est un problème relativement simple car la matrice \mathbf{A} est une matrice identité. Par conséquent, c’est un problème bien posé. Dans cette partie, nous allons résoudre un autre problème inverse plus compliqué : un problème de déconvolution qui est généralement très mal-posé. Dans ce problème, l’opérateur \mathbf{A} correspond à un opérateur de convolution. L’approche BV-MGSt et l’approche BV-MGGG peuvent donc être aussi utilisées pour résoudre les problèmes de déconvolution.

Pour évaluer les deux approches proposées, nous faisons les comparaisons avec une

5.5.2 - Problème de déconvolution

approche de seuillage itératif présentée dans la partie 5.3.2. Dans notre simulation, la fonction de régularisation ρ est choisie comme la norme L^1 et le seuillage impliqué est donc un seuillage doux. Néanmoins, comme nous l'avons indiqué auparavant, la sélection de seuil avec le *Projected GSURE* ne peut pas être utilisée pour résoudre les problèmes mal-posés de grande taille. Pour contourner ce problème, comme nous connaissons les vraies images, au lieu de calculer le critère *Projected GSURE*, nous utilisons directement l'erreur quadratique vraie comme critère pour choisir le seuil optimal.

Nous testons les trois approches avec les données synthétiques générées à partir de l'image *Lena* de taille 256×256 (voir la figure 5.4 (a)) et de l'image *Cameraman* de taille 128×128 (voir la figure 5.5 (a)). Pour générer des données, un noyau de convolution uniforme de taille 9×9 a été utilisé, ce qui conduit à un problème mal posé. Ensuite, nous avons ajouté un bruit gaussien pour atteindre un $\text{SNR} = 40$ dB aux images convoluées qui nous permet d'obtenir des images floues montrées dans la figure 5.4 (b) et la figure 5.5 (b).

Nous avons utilisé ici les mêmes initialisations que celles utilisées dans le problème de débruitage, afin de tester la robustesse de l'approche lorsqu'on change de problème. Contrairement au cas du débruitage, on estime la précision du bruit γ_n automatiquement. Concernant l'approche de seuillage itératif, les mêmes initialisations ont été prises pour les coefficients à estimer.

En ce qui concerne les paramètres de forme de la loi de Student et de la loi gaussienne généralisée, nous avons pris les mêmes valeurs que dans le cas débruitage : $[2000, 5, 2.5, 1]$ comme paramètres de forme de la loi de Student et $[2, 1, 0.9, 0.8]$ comme paramètres de forme de la loi gaussienne généralisée.

TABLE 5.4 – Les valeurs PSNR (dB) des reconstructions obtenues par le seuillage itératif, BV-MGSt et BV-MGGG

	seuillage	BV-MGSt	BV-MGGG
Lena	29.22	28.00	29.31
Cameraman	28.42	25.60	28.20

Nous montrons dans la figure 5.4 (c)-(e) et la figure 5.5 (c)-(e) les reconstructions obtenues par les trois approches. En comparant les reconstructions avec les images floues (voir la figure 5.4 (b) et la figure 5.5 (b)), nous pouvons voir que les trois approches permettent d'améliorer la qualité de l'image, plus spécifiquement, les contours des images deviennent beaucoup plus nets. Nous pouvons aussi voir que l'approche BV-MGGG donne des reconstructions similaires à celles obtenues par l'approche de seuillage itératif avec un seuil optimal et meilleures que celles obtenues par BV-MGSt. Par ailleurs, nous remarquons un problème du bord pour l'image *Cameraman*. Ceci vient du fait que les coefficients d'ondelettes se calculent par la convolution circulaire.

Nous montrons dans le tableau 5.4 les valeurs PSNR des reconstructions montrées dans la figure 5.4 et 5.5. Nous observons que pour l'image *Lena*, notre approche BV-MGGG donne une reconstruction de PSNR (29.31 dB) plus grande que celle obtenue par le seuillage itératif (29.22 dB). Néanmoins, dans le cas *Cameraman*, le seuillage itératif donne la valeur PSNR (28.42 dB) un peu plus grande que celle obtenue par BV-MGGG (28.20 dB). On peut en conclure que dans ce problème de déconvolution, notre approche BV-MGGG donne des reconstructions de qualité très similaire à celles obtenues par le

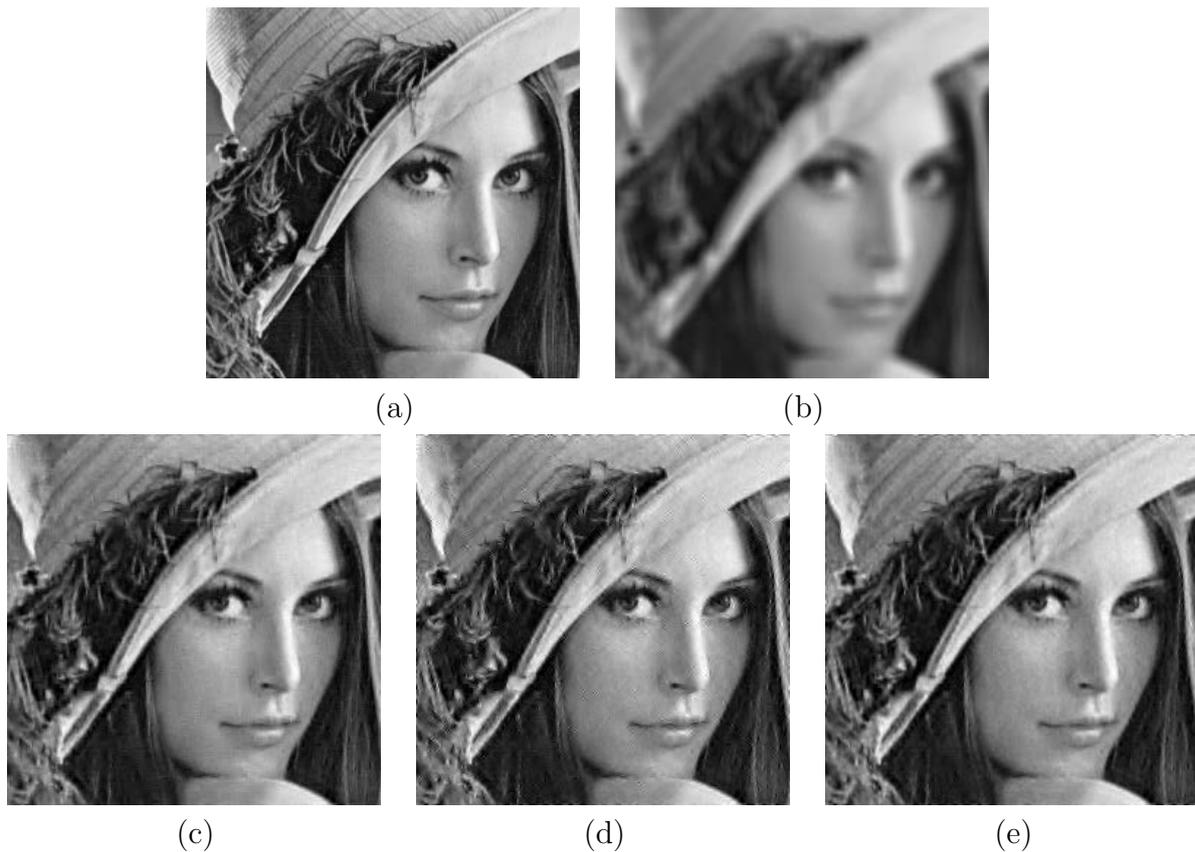


FIGURE 5.4 – (a) L'image vraie, (b) l'image floue, SNR = 40dB, (c) le seuillage itératif, (d) l'approche BV-MGSt et (e) l'approche BV-MGGG.

seuillage itératif. Il faut quand même mentionner que dans l'approche de seuillage utilisé nous considérons le meilleur paramètre (celui qui minimise EQM). C'est une information que bien évidemment nous n'utilisons pas nos approches non-supervisées. Les résultats obtenus sont donc très satisfaisants. En pratique, comme on ne connaît pas l'image vraie, l'EQM n'est pas calculable. Dans ce cas, on est obligé d'utiliser une approximation de l'EQM, par exemple, le critère GSURE introduit ci-dessus. Néanmoins, comme il existe des erreurs d'approximation, le seuil déterminé avec le critère GSURE est forcément moins bon que celui utilisé ici. Il est donc très probable que les résultats obtenus par les seuils sous-optimaux soient moins bons que les résultats obtenus par notre approche BV-MGGG. La bonne performance de notre approche BV-MGGG peut être expliquée par le fait qu'elle estime dix seuils différents pour les coefficients de dix sous-bandes différentes, comme nous l'avons expliqué dans le cas de débruitage.

Afin de montrer l'estimation pertinente des hyperparamètres γ_p^l , nous montrons leurs valeurs. Notons que ces hyperparamètres nous informent sur l'importance des informations de parcimonie introduites. Les valeurs de γ_p^l estimées par l'approche BV-MGGG pour l'image *Cameraman* sont montrées dans le tableau 5.5. Les valeurs de γ_p^l présentent le même comportement que dans le cas de débruitage : les valeurs de γ_p^l à l'échelle plus fine sont plus grandes que γ_p^l à l'échelle plus grossière ce qui est cohérent avec le fait que les coefficients d'ondelettes à l'échelle plus fine sont généralement plus parcimonieux.

En observant le tableau 5.4, on peut aussi remarquer que les résultats de BV-MGSt

5.5.2 - Problème de déconvolution



FIGURE 5.5 – (a) L'image vraie, (b) l'image floue, SNR = 40dB, (c) le seuillage itératif, (d) l'approche BV-MGSt et (e) l'approche BV-MGGG.

TABLE 5.5 – Hyperparamètres γ_p^l estimés par l'approche BV-MGGG pour *Cameraman*. Échelle de la plus grossière à la plus fine : 3 \rightarrow 1.

	Approximation	Échelle	Horizontale	Verticale	Diagonale
γ_p^l	0.04	3	12.32	10.07	41.37
		2	143.29	110.22	772.81
		1	1.16e+03	1.14e+03	2.41e+03

sont moins bons que les résultats de BV-MGGG. Nous montrons dans la figure 5.6 les histogrammes des coefficients reconstruits par l'approche BV-MGSt (voir la première ligne de la figure 5.6) et par l'approche BV-MGGG (voir la deuxième ligne de la figure 5.6). En comparant les histogrammes dans les deux lignes, nous pouvons voir que les histogrammes obtenus par BV-MGSt sont plus piqués autour de zéro que ceux obtenus par BV-MGGG. Il nous semble que l'utilisation de Student fait une opération ressemblant au seuillage dur qui donne des reconstructions de moins bonne qualité pour les images naturelles.

Au dessus, nous avons montré les valeurs de γ_p^l . Dans la suite, nous montrons aussi la valeur de la précision du bruit estimée. Comme nous connaissons la vraie valeur du bruit (γ_n^{-1}), nous comparons donc les valeurs estimées avec la vraie valeur de γ_n par le tableau 5.6. Nous pouvons voir que pour les deux images testées, nos deux approches donnent des estimations relativement précises, par exemple, dans le cas de l'image *Lena*, l'erreur relative de l'estimation de γ_n de l'approche BV-MGSt est 5.7% et l'erreur relative

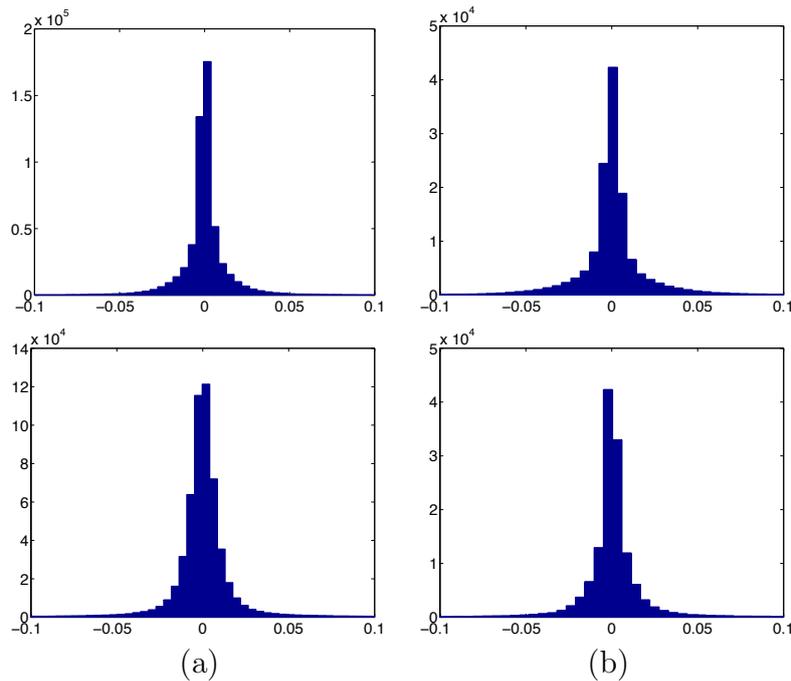


FIGURE 5.6 – Les histogrammes des coefficients des ondelettes dans l’intervalle $[-0.1, 0.1]$. Première ligne : BV-MGSt, deuxième ligne : BV-MGGG. (a) *Lena*, (b) *Cameraman*.

donnée par l’approche BV-MGGG est 9.7%, dans le cas de l’image *Cameraman*, l’erreur relative de l’approche BV-MGSt est 4.7% et l’erreur relative de l’approche BV-MGGG est 10.4%. Dans le tableau 5.6, nous pouvons remarquer que nos approches ont sous-estimé la précision du bruit. Ceci est probablement dû au fait qu’une partie du bruit est expliqué par l’*a priori* qui est blanc comme le bruit. Par conséquent, les approches ont mis moins de confiance aux données.

TABLE 5.6 – Les valeurs de γ_n estimées par nos approches : BV-MGSt et BV-MGGG

	valeur vraie	BV-MGSt	BV-MGGG
Lena	2.2619×10^5	2.1336×10^5	2.0404×10^5
Cameraman	1.0951×10^5	1.0432×10^5	9.8090×10^4

Comparaison avec la déconvolution basée sur TV (L’approche du chapitre 4)

Les approches basées sur l’*a priori* TV et l’*a priori* à variables cachées de type Geman développées au chapitre 4 permettent de résoudre des problèmes inverses linéaires quelconques (\mathbf{A} quelconque) de grande taille. Par conséquent, elles peuvent être aussi utilisées pour résoudre le problème de déconvolution. Comme pour l’image *Lena* et *Cameraman*, l’*a priori* TV marche mieux que l’*a priori* à variables cachées de type Geman. Nous montrons donc seulement les comparaisons avec l’approche basée sur l’*a priori* TV.

Nous avons appliqué l’approche basée sur TV sur les données de la figure 5.4 (b) et la figure 5.5 (b). Nous observons dans les figures 5.7 (a) et (b) les reconstructions

5.5.2 - Problème de déconvolution

obtenues utilisant l'approche basée sur TV. Les valeurs PSNR de chaque reconstruction sont montrées en bas de chaque image. Pour faciliter les comparaisons, nous remettons les reconstructions de l'approche BV-MGGG dans les figures 5.7 (c) et (d). En comparant les résultats de la ligne en haut et les résultats de la ligne en bas, nous pouvons voir que les qualités des images sont comparables. Si on compare les PSNR des reconstructions (montrées en bas de chaque image), nous pouvons voir que pour *Lena*, c'est l'approche BV-MGGG qui donne un résultat un peu meilleur, mais pour *Cameraman*, c'est l'approche basée sur TV qui est un peu meilleure. En fait, l'image *Cameraman* est plutôt régulière par morceau et l'*a priori* TV marche très bien pour ce type d'images. Par contre, dans l'image *Lena*, il existe plus de détails. Dans ce cas, c'est l'approche basée sur un *a priori* dans le domaine des ondelettes qui marche mieux.

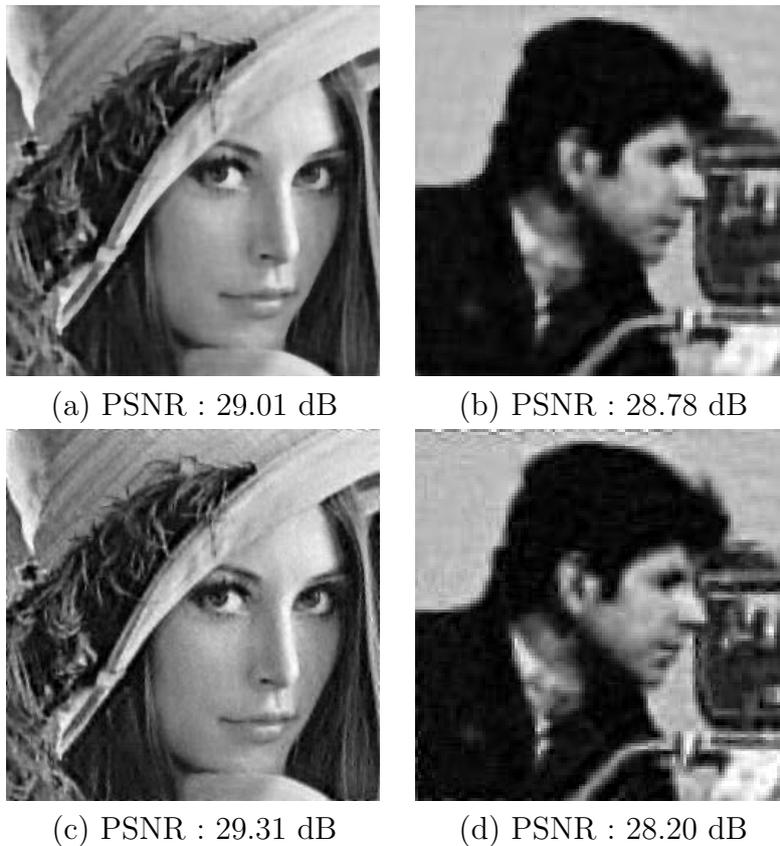


FIGURE 5.7 – Première ligne : les résultats obtenus avec l'*a priori* TV, deuxième ligne : les résultats obtenus par BV-MGGG.

5.6 Conclusion

Dans ce chapitre, nous avons développé une approche non-supervisée basée sur un *a priori* de la famille GSM dans un domaine transformé, à l'aide de l'approximation bayésienne variationnelle. Cette approche est assez générale : elle est valable pour divers problèmes inverses linéaires, différentes transformations peuvent être utilisées et la loi *a priori* peut être une loi quelconque dans la famille GSM pourvu que sa densité soit explicitement connue.

Pour introduire des informations de parcimonie, nous avons ensuite considéré deux lois de la famille GSM : la loi de Student et la loi gaussienne généralisée. Les approches basées sur les deux lois ont été testées dans deux problèmes inverses linéaires : un problème de débruitage bien-posé, un problème de déconvolution très mal-posé. Les résultats obtenus montrent que l'approche utilisant la loi gaussienne généralisée donne des résultats meilleurs que ceux obtenus par l'approche utilisant la loi de Student ce qui suggère que, la loi gaussienne généralisée permet de mieux décrire des coefficients d'ondelettes des images naturelles. Nous avons aussi montré les comparaisons avec des résultats d'un seuillage itératif avec un seuil optimal. Dans tous les problèmes, l'approche utilisant la loi gaussienne généralisée a donné des résultats de qualité équivalente au seuillage optimal. Ceci est dû au fait que notre approche choisit différents hyperparamètres (différents seuils) pour les coefficients de type différent. Nous avons montré à l'aide d'histogrammes sur les coefficients que les hyperparamètres étaient ajustés de manière cohérente. Enfin nos approches sont réellement non supervisées, nous avons montré qu'avec les mêmes initialisations et les mêmes paramètres de forme, nous obtenons de très bons résultats aussi bien sur un problème bien posé (débruitage) que sur un problème mal posé. Notons que nous avons fait d'autres types de simulation qui conforte ce résultat que nous n'avons pas montré ici (reconstruction de données tomographiques).

6

Conclusions et perspectives

6.1 Bilan des travaux

Dans le cadre de cette thèse, notre préoccupation principale est la résolution efficace de problèmes inverses de grande taille. De plus nous voulons développer des approches non supervisées. Pour ce faire, nous avons considéré des approches bayésiennes qui permettent d'estimer conjointement l'objet d'intérêt et les hyperparamètres. Dans ce cadre, la difficulté principale est que la loi *a posteriori* est en général complexe. La détermination d'un estimateur à partir de cette loi *a posteriori* dans ce contexte est très difficile. Pour résoudre ce problème, nous nous sommes intéressés à l'approximation bayésienne variationnelle (BV) qui offre une approximation séparable de la loi *a posteriori*. Cette approche est très intéressante car elle permet de lever les difficultés qui proviennent du lien complexe entre les paramètres d'intérêt et les hyperparamètres.

La première contribution de cette thèse consiste à transposer les méthodes d'optimisation de sous-espace dans l'espace fonctionnel impliqué dans le cadre BV. Notons que ce problème est loin d'être trivial car la structure des espaces fonctionnels est différente des espaces de Hilbert utilisés dans l'optimisation paramétrique. Nous avons donc proposé dans chapitre 3 une nouvelle méthode d'approximation bayésienne variationnelle. Cette méthode a été ensuite appliquée à un problème inverse linéaire de petite taille afin de mener une comparaison avec les approches classiques. Cette comparaison a montré que la méthode proposée est beaucoup plus efficace que celle de l'état de l'art : plus de 20000 fois plus rapide que MCMC, plus de 100 fois plus rapide que les approches BV classiques et vers 7 fois plus rapide que l'approche BV de type gradient. De plus, nous avons étudié deux sous-espaces d'ordre deux différents : un sous-espace de gradient (SG) et un sous-espace de mémoire de gradient (MG). Nous avons montré que la méthode d'approximation BV basé sur le sous-espace MG (BV-MG) est plus rapide que celle basée sur le sous-espace SG (BV-SG). Ceci peut être expliqué par le fait que la direction précédente incluse dans le sous-espace MG est plus informative que le gradient précédent inclus dans le sous-espace

SG.

Nous avons voulu ensuite confronter notre nouvelle méthodologie à des problèmes de traitement d'images de grande taille. De plus nous avons voulu favoriser les images régulières par morceau. Nous avons donc utilisé un *a priori* de Variation Total (TV). Les difficultés liées à l'utilisation de ce type d'*a priori* dans un contexte bayésien variationnel sont de deux ordres. Premièrement cet *a priori* n'est pas conjugué avec la vraisemblance ce qui est une des conditions de l'application des approches BV. Deuxièmement, l'estimation non supervisée nécessite de connaître de manière explicite la fonction de partition de l'*a priori*. Dans cette thèse, nous avons résolu le problème de conjugaison de la loi TV en utilisant la technique Minoration-Maximisation qui nous permet d'obtenir une variante conjuguée à chaque itération. Par conséquent, nous avons pu appliquer notre algorithme d'approximation BV-MG ce qui nous a permis d'obtenir une approche non-supervisée. Ensuite, l'approche proposée a été testée sur un problème de super-résolution. Nous avons montré que l'approche proposée est environs 5 fois plus rapide que l'approche BV classique et vers 1.9 fois plus rapide que l'approche BV de type gradient.

Dans le but d'introduire des informations de régularité par morceau, nous avons aussi considéré une loi *a priori* à variables cachées de type Geman : la loi du paramètre d'intérêt sachant les variables cachées est une loi gaussienne corrélée et les variables cachées suivent une loi de Laplace. Dans ce travail, deux difficultés ont été rencontrées. – Premièrement, la loi de Laplace n'est pas une loi conjuguée pour les variables cachées. Pour contourner ce problème, nous avons cherché une loi approchante sous-optimale dont la forme est fixée à une loi Dirac. – Deuxièmement, cette loi à variables cachées dépend de deux hyperparamètres. Cependant l'estimation de ces deux hyperparamètres est problématique. En effet, on estime conjointement deux paramètres qui modifient simultanément la forme et l'échelle de la loi. Dans ce modèle, nous ne pouvons pas introduire d'information de parcimonie (dans les gradients) car ceci nécessite de fixer la forme de la loi. Pour résoudre ce problème, nous avons proposé une reparamétrisation qui nous permet d'obtenir un paramètre de forme et un paramètre d'échelle. Nous fixons donc le paramètre de forme et nous estimons le paramètre d'échelle qui va nous permettre de régler le compromis entre l'information provenant des données et celle provenant de l'*a priori*. La détermination de la loi approchante de ce nouveau hyperparamètre donne une loi non conventionnelle donc les moments sont difficiles à déterminer lorsque la taille du problème est grande. Pour contourner ce problème, nous avons estimé ses moments en utilisant l'approche MCMC. Nous obtenons donc une approche hybride entre approximation analytique et approximation stochastique. Notons ici que cette approche stochastique ne réduit que faiblement la performance globale de l'approche car nous estimons seulement un paramètre scalaire. Pour finir nous avons comparé cette nouvelle approche (type Geman) à l'approche précédente (TV). Nous avons constaté que l'approche type Geman introduisait plus de parcimonie dans les gradients que l'approche TV. Contrairement à ce qui est couramment écrit dans l'état de l'art le maximum de parcimonie ne conduit pas forcément aux meilleurs résultats. Pour les images "naturelles" comme *Lena* ou le *Jetplane*, c'est l'approche TV qui donne les meilleurs résultats. Par contre, sur une image de *Mire* qui possède des différences finies très parcimonieuses c'est l'approche type Geman qui l'emporte.

La dernière contribution de cette thèse consiste à tirer partie de la décomposition sur un dictionnaire afin de simplifier notre problème bayésien variationnel. En effet, les lois *a priori* introduites précédemment sont corrélées ce qui rend l'estimation des paramètres d'échelle très compliquée : nous sommes souvent confronté à une fonction de partition non

explicite. Pour introduire les informations désirées en utilisant des lois de forme simple, nous avons considéré de travailler dans un domaine transformé, plus particulièrement, dans le domaine des ondelettes. En fait, les coefficients d'ondelettes des images naturelles sont généralement parcimonieux. De plus, nous pouvons introduire une information de corrélation sur l'objet recherché à travers un *a priori* séparable sur les coefficients d'ondelettes. Pour décrire la parcimonie, nous avons considéré des lois de la famille de mélange scalaire de gaussiennes par changement d'échelle (GSM) car cette famille comprend une variété des distributions permettant d'introduire une variété des informations. En plus, les distributions de la famille GSM peuvent être écrites sous une forme relativement plus simple pour l'application de l'approximation BV. Une contribution principale du chapitre 5 est de développer une approche non-supervisée pour les lois de la famille GSM dont la densité est explicitement connue, en utilisant l'approche d'approximation BV-MG. Nous avons testé notre nouvelle approche sur deux problèmes inverses de nature très différente : débruitage et déconvolution. Nous avons montré au travers de ces résultats que notre approche est réellement non supervisée car nous obtenons des très bons résultats pour les deux problèmes traités avec exactement les mêmes paramètres et les mêmes initialisations. Nous avons aussi fait la même constatation que pour l'approche TV, la plus forte parcimonie dans le domaine des ondelettes ne donne pas les meilleurs résultats, c'est pourquoi l'approche utilisant des gaussiennes généralisées donne les meilleurs résultats sur des images "naturelles" que les approches utilisant la loi de Student.

6.2 Perspectives

Perspectives à court terme :

- Une validation de l'approche sur des données réelles serait très utile. Nous pourrions voir si les comportements observés sur les simulations sont identiques à ceux sur données réelles. En particulier, la statistique du bruit sur les données réelles s'éloigne de la gaussianité et de l'indépendance.
- Dans le chapitre 4, nous avons vu que l'approche de type Geman avait des résultats limités car elle imposait une "trop" forte parcimonie dans les images de différences finies. Une idée serait d'introduire d'autres différences finies (pas uniquement horizontale et verticale), et d'appliquer ce type d'approche sur des problèmes de contrôle non destructif où les objets possèdent une grande régularité par morceau.
- Dans le chapitre 5, on estime des lois approchantes sur les coefficients d'ondelettes, puis on en déduit un estimateur de l'objet d'intérêt. Pour l'instant, cet estimateur est extrêmement simple. Nous pourrions utiliser en particulier la variance *a posteriori* des coefficients afin d'améliorer l'estimation de l'objet d'intérêt.

Stratégie de pas

Nous avons proposé une nouvelle méthode d'approximation bayésienne variationnelle basée sur la méthode d'optimisation de sous-espace. Cette méthode est une méthode itérative avec un pas sous-optimal déterminé par la stratégie de type backtracking : on calcule un pas initial en maximisant le développement de Taylor d'ordre deux du critère, et ensuite on teste si le pas obtenu fait augmenter le critère, sinon, on divise le pas

initial par deux itérativement jusqu'à obtenir un pas assurant l'augmentation du critère. En fait, la vitesse de convergence de la méthode dépend du pas choisi. En améliorant la stratégie de recherche de pas, on peut améliorer la vitesse de convergence de notre méthode. Chouzenoux *et al.* [CIM11] ont proposé une stratégie de pas par Majoration-Minoration efficace pour la méthode d'optimisation de sous-espace dans l'espace Hilbert. Une amélioration possible de notre méthode est donc d'adapter cette stratégie de pas par Majoration-Minoration dans notre espace fonctionnel.

Modèle du bruit

Durant ma thèse, nous sommes concentrés sur les problèmes inverses avec un bruit blanc additif gaussien. Néanmoins pour certaines applications, par exemple, en biologie ou en tomographie par émission de positon (TEP), on a besoin d'introduire des bruits de Poisson ou le bruit Poisson-gaussien. Dans ce cadre, le bruit n'est plus additif et dépend des données. Les lois *a priori* classiques telles que celles que nous avons utilisées dans ma thèse ne peuvent donc plus être conjuguées avec la vraisemblance déduite de la statistique du bruit. Dans un premier temps il faudra déterminer comment le bayésien variationnel peut s'adapter à ce type de bruit. Un second travail sera alors de déterminer quels *a priori* seraient envisageables et comment les adapter aux applications.

Estimation aveugle

Dans le travail de cette thèse, nous avons supposé que la matrice du système \mathbf{A} est connue. Néanmoins, en pratique, nous nous confrontons souvent des problèmes aveugles ou semi-aveugles où la matrice \mathbf{A} n'est pas connue ou est partiellement connue. Ce problème est envisageable dans le cadre bayésien en affectant des lois *a priori* sur les éléments de \mathbf{A} ou les paramètres lié à la matrice \mathbf{A} et estimant conjointement la matrice \mathbf{A} avec le paramètre d'intérêt \mathbf{x} . Dans ce cas, la loi *a posteriori* jointe de \mathbf{x} et \mathbf{A} est très probable de forme compliquée. Dans ce cas, ce qu'on peut faire est d'utiliser l'approche d'approximation BV pour générer des lois approchantes $q_{\mathbf{x}}$ et $q_{\mathbf{A}}$ qui nous permettent d'obtenir les estimations de \mathbf{x} et de \mathbf{A} . Une perspective d'ici est donc d'appliquer notre méthode d'approximation BV-MG au problème aveugle pour avoir des approches efficaces.

Choix de modèle

L'objectif du choix de modèle est de sélectionner un *meilleur* modèle parmi plusieurs différents candidats selon les observations \mathbf{y} . Le *meilleur* modèle est celui qui maximise l'*a posteriori* du modèle $p(\mathcal{M}_i|\mathbf{y})$ qui est proportionnel au produit de l'évidence de modèle $p(\mathbf{y}|\mathcal{M}_i)$ et de l'*a priori* de modèle $p(\mathcal{M}_i)$. Néanmoins, le calcul de l'évidence est généralement très difficile car on connaît de manière explicite uniquement la loi jointe $p(\mathbf{y}, \mathbf{x}|\mathcal{M}_i)$. Dans ce cas, le calcul de l'évidence consiste au calcul de l'intégrale $\int p(\mathbf{y}, \mathbf{x}|\mathcal{M}_i)d\mathbf{x}$ qui est généralement incalculable analytiquement. Néanmoins, l'approche d'approximation BV peut offrir une estimation de cette évidence. Comme nous avons vu dans (2.7) du chapitre 2, la log-évidence $\ln p(\mathbf{y}|\mathcal{M}_i)$ est égale à la somme de l'énergie libre négative et la divergence de Kullback-Leibler entre la loi approchante et la loi *a posteriori*. Lorsque l'algorithme BV converge, la divergence de Kullback-Leibler est quasiment nulle.

Par conséquent, l'énergie libre négative à convergence peut être donc une approximation de $\ln p(\mathbf{y}|\mathcal{M}_i)$.

Traitement récursif

Dans les applications telles que le guidage, la navigation et le contrôle des véhicules, on a besoin de faire les mesures en continu et prendre en compte les nouvelles mesures pour mettre à jour les estimations. Pour traiter ce type de problème, le filtre de Kalman est couramment utilisé afin de réduire le coût de calcul. Néanmoins, l'utilisation du filtre de Kalman nécessite le stockage et l'inversion de la matrice de covariance à chaque itération, ce qui est impossible lorsque le problème est de très grande taille. Pour résoudre ce problème, une possibilité est d'utiliser l'approximation bayésienne variationnelle, plus particulièrement notre approche BV-MG, qui offre une approximation séparable à chaque itération. Dans ce cas, la matrice de covariance est diagonale, on a besoin de stocker seulement ses éléments diagonaux qui sont de taille plus petite.

A

Détails des calculs de la fonction intermédiaire

A.1 La dérivation de $q_i^{r_1}$ pour le problème inverse en utilisant la loi a priori de Student

La fonction intermédiaire $q_i^{r_1}$ est une fonction exponentielle et son exposant est calculé de la manière suivante :

$$\begin{aligned}\ln q_i^{r_1}(x_i) &= - \int \left(\frac{\gamma_n}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + \sum_{i=1}^N \frac{1}{2} \gamma_p z_i x_i^2 \right) \left(\prod_{j \neq i} q_j^k(x_j) dx_j \right) q_{\mathbf{z}}^{k+1}(\mathbf{z}) d\mathbf{z} \\ &= C - \frac{\gamma_n}{2} \int (\mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} - 2\mathbf{x}^T \mathbf{A}^T \mathbf{y}) \prod_{j \neq i} q_j^k(x_j) dx_j \\ &\quad - \int \sum_{i=1}^N \frac{1}{2} \gamma_p z_i x_i^2 \left(\prod_{j \neq i} q_j^k(x_j) dx_j \right) q_{\mathbf{z}}^{k+1}(\mathbf{z}) d\mathbf{z},\end{aligned}\tag{A.1}$$

où C est une constante qui inclut tous les termes indépendants de x_i .

Pour obtenir $\ln q_i^{r_1}(x_i)$, il faut qu'on calcule les intégrales intervenues. Nous calculons d'abord l'intégrale $\int \mathbf{x}^T \mathbf{H} \mathbf{x} \prod_{j \neq i} q_j^k(x_j) dx_j$ avec la matrice \mathbf{H} identifiée comme $\mathbf{A}^T \mathbf{A}$. Dans ce cas, la matrice \mathbf{H} est une matrice symétrique. En gardant seulement les termes

ANNEXE A. DÉTAILS DES CALCULS DE LA FONCTION INTERMÉDIAIRE

dépendants de x_i , nous pouvons obtenir

$$\begin{aligned}
 \int \mathbf{x}^T \mathbf{H} \mathbf{x} \prod_{j \neq i} q_j^k(x_j) dx_j &= \int \sum_{p=1}^N \sum_{l=1}^N (\mathbf{H})_{pl} x_p x_l \prod_{j \neq i} q_j^k(x_j) dx_j \\
 &= \int \left[x_i^2 (\mathbf{H})_{ii} + 2x_i \sum_{p \neq i} (\mathbf{H})_{ip} x_p \right] \prod_{j \neq i} q_j^k(x_j) dx_j + C_1 \\
 &= x_i^2 (\mathbf{H})_{ii} + 2x_i \sum_{p \neq i} (\mathbf{H})_{ip} (\mathbf{m}_k)_p + C_1 \\
 &= x_i^2 (\mathbf{H})_{ii} + 2x_i \left[\sum_{p=1}^N (\mathbf{H})_{ip} (\mathbf{m}_k)_p - (\mathbf{H})_{ii} (\mathbf{m}_k)_i \right] + C_1 \\
 &= x_i^2 \text{diag}(\mathbf{H})_i + 2x_i (\mathbf{H} \mathbf{m}_k)_i - 2x_i \text{diag}(\mathbf{H})_i (\mathbf{m}_k)_i + C_1 \quad (\text{A.2})
 \end{aligned}$$

où $(\mathbf{H})_{ij}$ représente un élément de la matrice \mathbf{H} et C_1 est une constante égale à la somme de tous les termes indépendants de x_i .

Ensuite, nous pouvons calculer facilement l'intégrale suivante :

$$\int -2\mathbf{x}^T \mathbf{A}^T \mathbf{y} \prod_{j \neq i} q_j^k(x_j) dx_j = -2x_i (\mathbf{A}^T \mathbf{y})_i + C_2. \quad (\text{A.3})$$

De façon analogue, en gardant seulement les termes dépendants de x_i , la troisième intégrale est calculée comme

$$\begin{aligned}
 \int \sum_{i=1}^N \frac{1}{2} \gamma_p z_i x_i^2 \left(\prod_{j \neq i} q_j^k(x_j) dx_j \right) q_{\mathbf{z}}^{k+1}(\mathbf{z}) d\mathbf{z} &= \int \frac{1}{2} \gamma_p z_i x_i^2 \tilde{q}_i^{k+1}(z_i) dz_i \\
 &= \frac{\gamma_p}{2} \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} x_i^2 \quad (\text{A.4})
 \end{aligned}$$

Remplaçant (A.2), (A.3) et (A.4) dans (A.1), nous pouvons obtenir l'expression de $\log q_i^{r_1}(x_i)$ qui nous permet d'obtenir l'expression de $q_i^{r_1}(x_i)$ donnée par (3.38).

A.2 La dérivation de q_i^{r1} pour le problème inverse en utilisant la loi a priori de TV

Encore une fois, la fonction intermédiaire q_i^{r1} est une fonction exponentielle et son exposant est calculé de la manière analogue que le cas avec la loi de Student :

$$\begin{aligned}
 \log q_i^{r1}(x_i) &= \langle \log L(\Theta, \mathbf{y} | \boldsymbol{\lambda}^k) \rangle_{(\prod_{j \neq i} q_j^k) q_{\gamma_n}^k q_{\gamma_p}^k} \\
 &= \int \left[-\frac{\gamma_n \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2}{2} - \gamma_p \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2}{2\sqrt{\lambda_i^k}} \right] \\
 &\quad \times \left(\prod_{j \neq i} q_j^k(x_j) dx_j \right) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p) d\gamma_n d\gamma_p + C \\
 &= C - \frac{\langle \gamma_n \rangle^k}{2} \int (\mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} - 2\mathbf{x}^T \mathbf{A}^T \mathbf{y}) \prod_{j \neq i} q_j^k(x_j) dx_j \\
 &\quad - \frac{\langle \gamma_p \rangle^k}{2} \int \mathbf{x}^T [\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v] \mathbf{x} \prod_{j \neq i} q_j^k(x_j) dx_j, \tag{A.5}
 \end{aligned}$$

où C est une constante qui inclut tous les termes indépendants de x_i .

Dans l'Annexe A.1, j'ai montré les calculs pour l'intégrale $\int \mathbf{x}^T \mathbf{H} \mathbf{x} \prod_{j \neq i} q_j^k(x_j) dx_j$ avec \mathbf{H} comme une matrice symétrique. Dans (A.5), nous avons deux telles intégrales avec la matrice \mathbf{H} identifiée comme $\mathbf{A}^T \mathbf{A}$ et comme $\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v$. Remplaçant \mathbf{H} par ces deux matrices, nous obtenons les expressions des deux intégrales correspondantes :

$$\begin{aligned}
 \int \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} \prod_{j \neq i} q_j^k(x_j) dx_j &= x_i^2 \text{diag}(\mathbf{A}^T \mathbf{A})_i + 2x_i (\mathbf{A}^T \mathbf{A} \mathbf{m}_k)_i \\
 &\quad - 2x_i \text{diag}(\mathbf{A}^T \mathbf{A})_i (\mathbf{m}_k)_i + C_1 \tag{A.6}
 \end{aligned}$$

et

$$\begin{aligned}
 \int \mathbf{x}^T (\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v) \mathbf{x} \prod_{j \neq i} q_j^k(x_j) dx_j &= x_i^2 \text{diag}(\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v)_i \\
 &\quad + 2x_i (\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h \mathbf{m}_k + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v \mathbf{m}_k)_i - 2x_i \text{diag}(\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v)_i (\mathbf{m}_k)_i + C_2 \tag{A.7}
 \end{aligned}$$

Par ailleurs, l'intégrale $\int -2\mathbf{x}^T \mathbf{A}^T \mathbf{y} \prod_{j \neq i} q_j^k(x_j) dx_j$ est donnée par (A.3). En inserant (A.6), (A.7) et (A.3) dans (A.5), nous obtenons que

$$\begin{aligned}
 \log q_i^{r1}(x_i) &= \tilde{C} - \frac{\langle \gamma_n \rangle^k}{2} \left(x_i^2 \text{diag}(\mathbf{A}^T \mathbf{A})_i - 2x_i (\mathbf{A}^T \mathbf{y})_i \right. \\
 &\quad \left. + 2x_i (\mathbf{A}^T \mathbf{A} \mathbf{m}_k)_i - 2x_i \text{diag}(\mathbf{A}^T \mathbf{A})_i (\mathbf{m}_k)_i \right) \\
 &\quad - \frac{\langle \gamma_p \rangle^k}{2} \left(x_i^2 \text{diag}(\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v)_i \right. \\
 &\quad \left. + 2x_i (\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h \mathbf{m}_k + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v \mathbf{m}_k)_i \right. \\
 &\quad \left. - 2x_i \text{diag}(\mathbf{D}_h^T \boldsymbol{\Lambda}^k \mathbf{D}_h + \mathbf{D}_v^T \boldsymbol{\Lambda}^k \mathbf{D}_v)_i (\mathbf{m}_k)_i \right), \tag{A.8}
 \end{aligned}$$

qui nous permet d'obtenir facilement l'expression de q_i^{r1} donnée par (4.27).

B

Calculs du pas de l'algorithme

B.1 Calculs pour le pas sous-optimal de l'approche BV basé sur le Sous-espace Gradient (BV-SG)

Nous rappelons d'abord que la fonction g^k est définie comme

$$g^k : \mathbb{R}^2 \rightarrow \mathbb{R} \quad g^k(\mathbf{s}) = \mathcal{F} \left(K^k(\mathbf{s}) q^k(\mathbf{w}) \exp [\mathbf{D}^k(\mathbf{w})\mathbf{s}] \right). \quad (\text{B.1})$$

L'algorithme proposé adopte le pas sous-optimal défini par (3.20). Pour déterminer ce pas, il faut qu'on calcul le pas initial $\hat{\mathbf{s}}^k$ dont la formule est rappelée comme suit :

$$\hat{\mathbf{s}}^k = - \left(\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}} \right)^{-1} \frac{\partial g^k}{\partial \mathbf{s}} \Big|_{\mathbf{s}=\mathbf{0}}. \quad (\text{B.2})$$

Pour le calculer, il faut qu'on obtient tout d'abord le gradient (la dérivée partielle du premier ordre) et la matrice hessienne (la dérivée partielle du second ordre) au $\mathbf{s} = \mathbf{0}$. Les deux éléments du gradient de $g^k(\mathbf{s})$ au zéro sont trouvés comme pour $l = 1, 2$,

$$\begin{aligned} \frac{\partial g(\mathbf{s})}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} &= \langle \gamma_n \rangle^k \left(\frac{\partial \mathbf{m}_\mathbf{s}}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \right)^T (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_k) - \sum_{i=1}^N \left(\langle \gamma_p \rangle^k \frac{\partial (\mathbf{m}_\mathbf{s})_i}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} (\mathbf{m}_k)_i \right) \\ &+ \frac{1}{2} \sum_{i=1}^N \left(\frac{\partial (\boldsymbol{\sigma}_\mathbf{s}^2)_i}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \left(-\langle \gamma_n \rangle^k (\mathbf{A}^T \mathbf{A})_{ii} - \langle \gamma_p \rangle^k \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} + \frac{1}{(\boldsymbol{\sigma}_k^2)_i} \right) \right), \end{aligned} \quad (\text{B.3})$$

Car nous utilisons un sous-espace de dimension deux, la matrice hessienne est de

B.1.1 - Étude des dérivées partielles de la moyenne \mathbf{m}_s et de la variance σ_s^2

dimension 2×2 qui est donnée par

$$\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}} = \begin{bmatrix} \frac{\partial^2 g^k}{\partial s_1^2} \Big|_{\mathbf{s}=\mathbf{0}} & \frac{\partial^2 g^k}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \\ \frac{\partial^2 g^k}{\partial s_2 \partial s_1} \Big|_{\mathbf{s}=\mathbf{0}} & \frac{\partial^2 g^k}{\partial s_2^2} \Big|_{\mathbf{s}=\mathbf{0}} \end{bmatrix}. \quad (\text{B.4})$$

Nous donnons les quatre éléments de la matrice hessienne dans la suite. Ses deux éléments diagonaux sont trouvés comme, pour $l = 1, 2$,

$$\begin{aligned} \frac{\partial^2 g^k}{\partial s_l^2} \Big|_{\mathbf{s}=\mathbf{0}} &= \langle \gamma_n \rangle^k \left(\frac{\partial^2 \mathbf{m}_s}{\partial s_l^2} \Big|_{\mathbf{s}=\mathbf{0}} \right)^T (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_k) - \langle \gamma_n \rangle^k \left(\frac{\partial \mathbf{m}_s}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \right)^T \mathbf{A}^T \mathbf{A} \left(\frac{\partial \mathbf{m}_s}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \right) \\ &\quad - \langle \gamma_p \rangle^k \sum_{i=1}^N \left[\frac{\partial^2 (\mathbf{m}_s)_i}{\partial s_l^2} \Big|_{\mathbf{s}=\mathbf{0}} \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} (\mathbf{m}_k)_i + \left(\frac{\partial (\mathbf{m}_s)_i}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \right)^2 \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} \right] \\ &\quad + \frac{1}{2} \sum_{i=1}^N \left[\frac{\partial^2 (\sigma_s^2)_i}{\partial s_l^2} \Big|_{\mathbf{s}=\mathbf{0}} \left(-\langle \gamma_n \rangle^k (\mathbf{A}^T \mathbf{A})_{ii} - \langle \gamma_p \rangle^k \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} + \frac{1}{(\sigma_k^2)_i^2} \right) \right] \\ &\quad - \sum_{i=1}^N \left[\left(\frac{\partial (\sigma_s^2)_i}{\partial s_l} \Big|_{\mathbf{s}=\mathbf{0}} \right)^2 \frac{1}{2(\sigma_k^2)_i^2} \right]. \end{aligned} \quad (\text{B.5})$$

Ensuite, la matrice hessienne est symétrique. Donc les deux autres éléments sont égaux et ils sont trouvés d'être de forme suivante :

$$\begin{aligned} \frac{\partial^2 g^k}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} &= \langle \gamma_n \rangle^k \left(\frac{\partial^2 \mathbf{m}_s}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \right)^T (\mathbf{A}^T \mathbf{y} - \mathbf{A}^T \mathbf{A} \mathbf{m}_k) - \langle \gamma_n \rangle^k \left(\frac{\partial \mathbf{m}_s}{\partial s_1} \Big|_{\mathbf{s}=\mathbf{0}} \right)^T \mathbf{A}^T \mathbf{A} \left(\frac{\partial \mathbf{m}_s}{\partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \right) \\ &\quad - \langle \gamma_p \rangle^k \sum_{i=1}^N \left[\frac{\partial^2 (\mathbf{m}_s)_i}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} (\mathbf{m}_k)_i + \left(\frac{\partial (\mathbf{m}_s)_i}{\partial s_1} \Big|_{\mathbf{s}=\mathbf{0}} \frac{\partial (\mathbf{m}_s)_i}{\partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \right) \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} \right] \\ &\quad + \frac{1}{2} \sum_{i=1}^N \left[\frac{\partial^2 (\sigma_s^2)_i}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \left(-\langle \gamma_n \rangle^k (\mathbf{A}^T \mathbf{A})_{ii} - \langle \gamma_p \rangle^k \frac{(\mathbf{a})_i}{(\mathbf{b}_{k+1})_i} + \frac{1}{(\sigma_k^2)_i^2} \right) \right] \\ &\quad - \sum_{i=1}^N \left[\left(\frac{\partial (\sigma_s^2)_i}{\partial s_1} \Big|_{\mathbf{s}=\mathbf{0}} \frac{\partial (\sigma_s^2)_i}{\partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} \right) \frac{1}{2(\sigma_k^2)_i^2} \right]. \end{aligned} \quad (\text{B.6})$$

La formule du gradient (B.3) et les formules de la matrice hessienne (B.5) - (B.6) montrent que ses évaluations nécessitent des études de la dérivée partielle du premier ordre et de la dérivée partielle du second ordre de \mathbf{m}_s et de σ_s^2 au $\mathbf{s} = \mathbf{0}$.

B.1.1 Étude des dérivées partielles de la moyenne \mathbf{m}_s et de la variance σ_s^2

Nous donnons ici les expressions des dérivées de chaque élément de \mathbf{m}_s (voir (3.44) et σ_s^2 (3.45)). Pour la simplicité, nous omettons les indications de composantes $(\cdot)_i$ dans la suite, c'est à dire que \mathbf{u} représente $(\mathbf{u})_i$ dans la suite.

Les dérivées partielles du premier ordre de la moyenne et de la variance s'écrivent :

$$\frac{\partial \mathbf{m}_s}{\partial \mathbf{s}} \Big|_{\mathbf{s}=\mathbf{0}} = \left(\begin{array}{c} \frac{\sigma_k^2}{\sigma_{r_1}^2} (\mathbf{m}_{r_1} - \mathbf{m}_k) \\ \frac{\sigma_k^2}{\sigma_{r_2}^2} (\mathbf{m}_{r_2} - \mathbf{m}_k) + \frac{\sigma_k^2}{\sigma_{k-1}^2} (\mathbf{m}_k - \mathbf{m}_{k-1}) \end{array} \right), \quad (\text{B.7})$$

$$\frac{\partial \sigma_s^2}{\partial \mathbf{s}} \Big|_{\mathbf{s}=\mathbf{0}} = \left(\begin{array}{c} \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2)}{\sigma_{r_1}^2} \\ \frac{\sigma_k^4 (\sigma_{r_2}^2 - \sigma_{k-1}^2)}{\sigma_{r_2}^2 \sigma_{k-1}^2} \end{array} \right), \quad (\text{B.8})$$

$$(\text{B.9})$$

Ensuite, les dérivées partielles du second ordre de la moyenne sont trouvées de formes suivantes :

$$\frac{\partial^2 \mathbf{m}_s}{\partial s_1^2} \Big|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2) (\mathbf{m}_{r_1} - \mathbf{m}_k)}{\sigma_{r_1}^4}, \quad (\text{B.10})$$

$$\frac{\partial^2 \mathbf{m}_s}{\partial s_2^2} \Big|_{\mathbf{s}=\mathbf{0}} = 2 \sigma_k^2 \left(\frac{1}{\sigma_{k-1}^2} - \frac{1}{\sigma_{r_2}^2} \right) \left[\frac{\sigma_k^2}{\sigma_{r_2}^2} (\mathbf{m}_{r_2} - \mathbf{m}_k) + \frac{\sigma_k^2}{\sigma_{k-1}^2} (\mathbf{m}_k - \mathbf{m}_{k-1}) \right] \quad (\text{B.11})$$

$$\begin{aligned} \frac{\partial^2 \mathbf{m}_s}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} &= \sigma_k^4 \left(\frac{1}{\sigma_{k-1}^2} - \frac{1}{\sigma_{r_2}^2} \right) \left(\frac{\mathbf{m}_{r_1}}{\sigma_{r_1}^2} + \frac{\mathbf{m}_k}{\sigma_k^2} - 2 \frac{\mathbf{m}_k}{\sigma_{r_1}^2} \right) \\ &+ \sigma_k^4 \left(\frac{1}{\sigma_k^2} - \frac{1}{\sigma_{r_1}^2} \right) \left(\frac{\mathbf{m}_{r_2}}{\sigma_{r_2}^2} - \frac{\mathbf{m}_{k-1}}{\sigma_{k-1}^2} \right) \end{aligned} \quad (\text{B.12})$$

Et les dérivées partielles du second ordre sont données par

$$\frac{\partial^2 \sigma_s^2}{\partial s_1^2} \Big|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2)^2}{\sigma_{r_1}^4}, \quad (\text{B.13})$$

$$\frac{\partial^2 \sigma_s^2}{\partial s_2^2} \Big|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^6 (\sigma_{r_2}^2 - \sigma_{k-1}^2)^2}{\sigma_{r_2}^4 \sigma_{k-1}^4}, \quad (\text{B.14})$$

$$\frac{\partial^2 \sigma_s^2}{\partial s_1 \partial s_2} \Big|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^4 (\sigma_{r_1}^2 - \sigma_k^2) (\sigma_{r_2}^2 - \sigma_{k-1}^2)}{\sigma_{r_1}^2 \sigma_{r_2}^2 \sigma_{k-1}^2}. \quad (\text{B.15})$$

Ces formules nous permettent d'obtenir des dérivées partielles de la moyenne et de la variance qui sont nécessaires pour le calcul de $\frac{\partial g(\mathbf{s})}{\partial s_i} \Big|_{\mathbf{s}=\mathbf{0}}$ et $\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}}$. Ensuite, le pas $\hat{\mathbf{s}}^k$ peut être obtenu en utilisant (B.2).

B.2 Calculs pour le pas sous-optimal de l'approche BV basé sur le sous-espace Mémoire de Gradient (BV-MG)

Dans ce cas, un sous-espace différent est utilisé, ce qui entraîne des expressions différentes pour \mathbf{m}_s (voir (3.47)) et pour σ_s^2 (voir (3.48)). Néanmoins, le pas $\hat{\mathbf{s}}^k$ est encore déterminé par (B.2) et les formules de $\frac{\partial g(\mathbf{s})}{\partial s_i} \Big|_{\mathbf{s}=\mathbf{0}}$ et $\frac{\partial^2 g^k}{\partial \mathbf{s} \partial \mathbf{s}^T} \Big|_{\mathbf{s}=\mathbf{0}}$ sont aussi les mêmes que (B.3) et (B.5-B.6). En raison que \mathbf{m}_s et σ_s^2 sont calculés différemment, leurs dérivées partielles

B.1.1 - Étude des dérivées partielles de la moyenne \mathbf{m}_s et de la variance σ_s^2

sont alors de formes différentes. Premièrement, les dérivées partielles du premier ordre sont données par

$$\left. \frac{\partial \mathbf{m}_s}{\partial \mathbf{s}} \right|_{\mathbf{s}=\mathbf{0}} = \begin{pmatrix} \frac{\sigma_k^2}{\sigma_{r_1}^2} (\mathbf{m}_{r_1} - \mathbf{m}_k) \\ \frac{\sigma_k^2}{\sigma_{k-1}^2} (\mathbf{m}_k - \mathbf{m}_{k-1}) \end{pmatrix}, \quad (\text{B.16})$$

$$\left. \frac{\partial \sigma_s^2}{\partial \mathbf{s}} \right|_{\mathbf{s}=\mathbf{0}} = \begin{pmatrix} \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2)}{\sigma_{r_1}^2} \\ \frac{\sigma_k^2 (\sigma_k^2 - \sigma_{k-1}^2)}{\sigma_{k-1}^2} \end{pmatrix}. \quad (\text{B.17})$$

Ensuite, nous calculons les dérivées partielles du second ordre qui nous donne

$$\left. \frac{\partial^2 \mathbf{m}_s}{\partial s_1^2} \right|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2) (\mathbf{m}_{r_1} - \mathbf{m}_k)}{\sigma_{r_1}^4}, \quad (\text{B.18})$$

$$\left. \frac{\partial^2 \mathbf{m}_s}{\partial s_2^2} \right|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^4}{\sigma_{k-1}^2} \left(\frac{1}{\sigma_{k-1}^2} - \frac{1}{\sigma_k^2} \right) (\mathbf{m}_k - \mathbf{m}_{k-1}), \quad (\text{B.19})$$

$$\begin{aligned} \left. \frac{\partial^2 \mathbf{m}_s}{\partial s_1 \partial s_2} \right|_{\mathbf{s}=\mathbf{0}} &= \sigma_k^4 \left(\frac{1}{\sigma_{k-1}^2} - \frac{1}{\sigma_k^2} \right) \left(\frac{\mathbf{m}_{r_1}}{\sigma_{r_1}^2} + \frac{\mathbf{m}_k}{\sigma_k^2} - 2 \frac{\mathbf{m}_k}{\sigma_{r_1}^2} \right) \\ &\quad + \sigma_k^4 \left(\frac{1}{\sigma_k^2} - \frac{1}{\sigma_{r_1}^2} \right) \left(\frac{\mathbf{m}_k}{\sigma_k^2} - \frac{\mathbf{m}_{k-1}}{\sigma_{k-1}^2} \right). \end{aligned} \quad (\text{B.20})$$

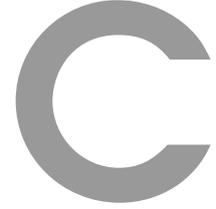
Et les dérivées partielles du second ordre sont données par

$$\left. \frac{\partial^2 \sigma_s^2}{\partial s_1^2} \right|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2)^2}{\sigma_{r_1}^4}, \quad (\text{B.21})$$

$$\left. \frac{\partial^2 \sigma_s^2}{\partial s_2^2} \right|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^2 (\sigma_k^2 - \sigma_{k-1}^2)^2}{\sigma_{k-1}^4}, \quad (\text{B.22})$$

$$\left. \frac{\partial^2 \sigma_s^2}{\partial s_1 \partial s_2} \right|_{\mathbf{s}=\mathbf{0}} = 2 \frac{\sigma_k^2 (\sigma_{r_1}^2 - \sigma_k^2) (\sigma_k^2 - \sigma_{k-1}^2)}{\sigma_{r_1}^2 \sigma_{k-1}^2}. \quad (\text{B.23})$$

Ces formules nous permettent d'obtenir des dérivées partielles de la moyenne et de la variance, ce qui nous permet de calculer la dérivée du premier ordre et la dérivée du second ordre de la fonction $g^k(\mathbf{s})$ au zéro. Finalement, le pas $\hat{\mathbf{s}}^k$ est obtenu en utilisant (B.2).



Détails de calculs du Chapitre 4

C.1 Détails de calculs de la variable auxiliaire λ_i

Le maximum de \mathcal{F}^L est atteint à son point critique qui est calculé comme suit :

$$\begin{aligned}\lambda_i^{k+1} &= \arg \max_{\lambda_i \in \mathbb{R}^+} \mathcal{F}^L(q_{\mathbf{x}}^{k+1} q_{\gamma_n}^k q_{\gamma_p}^k, \boldsymbol{\lambda}) \\ &= \arg \max_{\lambda_i \in \mathbb{R}^+} \int q_{\mathbf{x}}^{k+1}(\mathbf{x}) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p) \ln \left(\frac{L(\mathbf{x}, \gamma_n, \gamma_p, \mathbf{y} | \boldsymbol{\lambda})}{q_{\mathbf{x}}^{k+1}(\mathbf{x}) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p)} \right) d\mathbf{x} d\gamma_n d\gamma_p. \quad (\text{C.1})\end{aligned}$$

En fait, dans l'expression précédente, il n'y a que $L(\mathbf{x}, \gamma_n, \gamma_p, \mathbf{y} | \boldsymbol{\lambda})$ dépendant de $\boldsymbol{\lambda}$. En négligeant les termes indépendants de $\boldsymbol{\lambda}$, nous pouvons obtenir

$$\begin{aligned}\lambda_i^{k+1} &= \arg \max_{\lambda_i \in \mathbb{R}^+} \int q_{\mathbf{x}}^{k+1}(\mathbf{x}) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p) \ln L(\mathbf{x}, \gamma_n, \gamma_p, \mathbf{y} | \boldsymbol{\lambda}) d\mathbf{x} d\gamma_n d\gamma_p \\ &= \arg \max_{\lambda_i \in \mathbb{R}^+} - \int q_{\mathbf{x}}^{k+1}(\mathbf{x}) q_{\gamma_n}^k(\gamma_n) q_{\gamma_p}^k(\gamma_p) \gamma_p \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \lambda_i}{2\sqrt{\lambda_i}} d\mathbf{x} d\gamma_n d\gamma_p \\ &= \arg \min_{\lambda_i \in \mathbb{R}^+} \int q_{\mathbf{x}}^{k+1}(\mathbf{x}) \sum_{i=1}^N \frac{(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2 + \lambda_i}{2\sqrt{\lambda_i}} d\mathbf{x} \\ &= \arg \min_{\lambda_i \in \mathbb{R}^+} \sum_{i=1}^N \frac{\mathbb{E}_{q_{\mathbf{x}}^{k+1}} [(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2] + \lambda_i}{2\sqrt{\lambda_i}} \\ &= \mathbb{E}_{q_{\mathbf{x}}^{k+1}} [(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2]. \quad (\text{C.2})\end{aligned}$$

Lorsque $q_{\mathbf{x}}^{k+1}$ est la densité d'une loi gaussienne multi-dimensionnelle de moyenne \mathbf{m}_{k+1}

et de matrice de variance Σ_{k+1} , on peut calculer l'expectation (C.2) comme ci-dessous :

$$\begin{aligned}
 \lambda_i^{k+1} &= \mathbb{E}_{q_{\mathbf{x}}^{k+1}} [(\mathbf{D}_h \mathbf{x})_i^2 + (\mathbf{D}_v \mathbf{x})_i^2] \\
 &= \mathbb{E}_{q_{\mathbf{x}}^{k+1}} [\mathbf{x}^T (\mathbf{D}_h)_i^T (\mathbf{D}_h)_i \mathbf{x} + \mathbf{x}^T (\mathbf{D}_v)_i^T (\mathbf{D}_v)_i \mathbf{x}] \\
 &= \int \sum_{i=1}^N \sum_{j=1}^N x_i x_j [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i]_{ij} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x} + \int \sum_{i=1}^N \sum_{j=1}^N x_i x_j [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i]_{ij} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x} \\
 &= \sum_{i=1}^N \sum_{j=1}^N \int x_i x_j [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i]_{ij} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x} + \sum_{i=1}^N \sum_{j=1}^N \int x_i x_j [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i]_{ij} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x}
 \end{aligned} \tag{C.3}$$

Pour le calculer, nous calculons d'abord $\int x_i x_j (\mathbf{H})_{ij} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x}$. Car la matrice de covariance de $q_{\mathbf{x}}^{k+1}$ est Σ_{k+1} , nous pouvons écrire

$$\int x_i x_j q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x} = (\Sigma_{k+1})_{ij} + (\mathbf{m}_{k+1})_i (\mathbf{m}_{k+1})_j \tag{C.4}$$

Par conséquent,

$$\int x_i x_j (\mathbf{H})_{ij} q_{\mathbf{x}}^{k+1}(\mathbf{x}) d\mathbf{x} = (\Sigma_{k+1})_{ij} (\mathbf{H})_{ij} + (\mathbf{m}_{k+1})_i (\mathbf{m}_{k+1})_j (\mathbf{H})_{ij}. \tag{C.5}$$

En conséquence,

$$\begin{aligned}
 \lambda_i^{k+1} &= \sum_{i=1}^N \sum_{j=1}^N \left[(\mathbf{m}_{k+1})_i (\mathbf{m}_{k+1})_j [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i]_{ij} + (\Sigma_{k+1})_{ij} [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i]_{ij} \right] \\
 &\quad + \sum_{i=1}^N \sum_{j=1}^N \left[(\mathbf{m}_{k+1})_i (\mathbf{m}_{k+1})_j [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i]_{ij} + (\Sigma_{k+1})_{ij} [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i]_{ij} \right] \\
 &= \mathbf{m}_{k+1}^T (\mathbf{D}_h)_i^T (\mathbf{D}_h)_i \mathbf{m}_{k+1} + \mathbf{m}_{k+1}^T (\mathbf{D}_v)_i^T (\mathbf{D}_v)_i \mathbf{m}_{k+1} \\
 &\quad + \sum_{i=1}^N \sum_{j=1}^N (\Sigma_{k+1})_{ij} [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i]_{ij} + \sum_{i=1}^N \sum_{j=1}^N (\Sigma_{k+1})_{ij} [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i]_{ij} \\
 &= (\mathbf{D}_h \mathbf{m}_{k+1})_i^2 + (\mathbf{D}_v \mathbf{m}_{k+1})_i^2 + \text{trace} [(\mathbf{D}_h)_i^T (\mathbf{D}_h)_i \Sigma_{k+1}] \\
 &\quad + \text{trace} [(\mathbf{D}_v)_i^T (\mathbf{D}_v)_i \Sigma_{k+1}].
 \end{aligned} \tag{C.6}$$

C.2 Calculs de la fonction de partition de la loi a priori de type Geman $Z_{\mathbf{x}|\mathbf{b}}$

Nous calculons la fonction de partition $Z_{\mathbf{x}|\mathbf{b}}$ donnée par (4.46) en transposant les calculs de l'intégral dans le domaine de Fourier.

Notons d'abord les noyaux de convolution correspondant aux matrices \mathbf{D}_h et \mathbf{D}_v par \mathbf{d}_h et \mathbf{d}_v , respectivement. En outre, nous utilisons $*$ à dénoter l'opérateur de convolution. En conséquence, nous pouvons écrire $\mathbf{D}_h \mathbf{x} = \mathbf{d}_h * \mathbf{x}$ et $\mathbf{D}_v \mathbf{x} = \mathbf{d}_v * \mathbf{x}$. Par ailleurs, $\overset{\circ}{\mathbf{a}}$ représente la transformée de Fourier du vecteur \mathbf{a} .

En utilisant le théorème de Parseval, nous pouvons obtenir que

$$\begin{aligned}
 \|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2 &= \|\mathring{\mathbf{d}}_h \circ \mathring{\mathbf{x}} - \mathring{\mathbf{b}}_h\|^2 + \|\mathring{\mathbf{d}}_v \circ \mathring{\mathbf{x}} - \mathring{\mathbf{b}}_v\|^2 \\
 &= \sum_i \left\{ \left[(\mathring{\mathbf{d}}_h)_i (\mathring{\mathbf{x}})_i - (\mathring{\mathbf{b}}_h)_i \right]^2 + \left[(\mathring{\mathbf{d}}_v)_i (\mathring{\mathbf{x}})_i - (\mathring{\mathbf{b}}_v)_i \right]^2 \right\} \\
 &= \sum_i \left\{ \left[(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2 \right] (\mathring{\mathbf{x}})_i^2 + (\mathring{\mathbf{b}}_h)_i^2 + (\mathring{\mathbf{b}}_v)_i^2 \right. \\
 &\quad \left. - 2 \left[(\mathring{\mathbf{d}}_h)_i (\mathring{\mathbf{b}}_h)_i + (\mathring{\mathbf{d}}_v)_i (\mathring{\mathbf{b}}_v)_i \right] (\mathring{\mathbf{x}})_i \right\} \\
 &= \sum_i \left\{ \left[(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2 \right] \left((\mathring{\mathbf{x}})_i - \frac{(\mathring{\mathbf{d}}_h)_i (\mathring{\mathbf{b}}_h)_i + (\mathring{\mathbf{d}}_v)_i (\mathring{\mathbf{b}}_v)_i}{(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2} \right)^2 \right. \\
 &\quad \left. + \frac{\left((\mathring{\mathbf{d}}_h)_i (\mathring{\mathbf{b}}_v)_i - (\mathring{\mathbf{d}}_v)_i (\mathring{\mathbf{b}}_h)_i \right)^2}{(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2} \right\}. \tag{C.7}
 \end{aligned}$$

Donc, la fonction de partition $Z_{\mathbf{x}|\mathbf{b}}$ peut être calculée dans le domaine de la transformée de Fourier grâce à un changement de variable :

$$\begin{aligned}
 Z_{\mathbf{x}|\mathbf{b}} &= \int \exp \left[-\frac{\gamma_d}{2} (\|\mathbf{D}_h \mathbf{x} - \mathbf{b}_h\|^2 + \|\mathbf{D}_v \mathbf{x} - \mathbf{b}_v\|^2) \right] d\mathbf{x} \\
 &= (2\pi)^{N/2} \gamma_d^{-N/2} \prod_i \left[(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2 \right]^{-1/2} \\
 &\quad \times \exp \left[-\frac{\gamma_d}{2} \frac{\left((\mathring{\mathbf{d}}_h)_i (\mathring{\mathbf{b}}_v)_i - (\mathring{\mathbf{d}}_v)_i (\mathring{\mathbf{b}}_h)_i \right)^2}{(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2} \right] \\
 &= c \gamma_d^{-N/2} \exp \left[-\frac{\gamma_d}{2} \sum_i \frac{\left((\mathring{\mathbf{d}}_h)_i (\mathring{\mathbf{b}}_v)_i - (\mathring{\mathbf{d}}_v)_i (\mathring{\mathbf{b}}_h)_i \right)^2}{(\mathring{\mathbf{d}}_h)_i^2 + (\mathring{\mathbf{d}}_v)_i^2} \right] \tag{C.8}
 \end{aligned}$$

où c est un facteur qui est le produit de tous les termes indépendants de γ_d , $\mathring{\mathbf{b}}_h$ et $\mathring{\mathbf{b}}_v$. Nous pouvons donc facilement obtenir

$$Z_{\mathbf{x}|\mathbf{b}} \leq c \gamma_d^{-N/2} \tag{C.9}$$

car le troisième facteur exponentiel dans (C.8) est strictement inférieur à un.

Publications

Article de revue

1. Y. Zheng, A. Fraysse, T. Rodet. "Efficient variational Bayesian approximation method based on subspace optimization." *IEEE Trans. Image Process*, 24(2) :681–693, 2015.

Conférences Internationales

1. Y. Zheng, T. Rodet, A. Fraysse. "Fast variational bayesian approaches applied to large dimensional problems." *IEEE International Conference on Image Processing (ICIP)*, pages 478–483, 2013.
2. Y. Zheng, A. Fraysse, T. Rodet. "A subspace-based variational bayesian method." *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6620–6624, 2013.
3. Y. Zheng, C. Cai, T. Rodet. "Joint reduce of metal and beam hardening artifacts using multi-energy map approach in x-ray computed tomography." *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 737–740, 2011.

Conférences Nationales

1. Y. Zheng, T. Rodet, A. Fraysse, "Approches rapides du bayésien variationnel pour des problèmes de grande dimension", Septembre p. id 241, *Acte du Grets*, Brest, France, 2013.
2. T. Rodet, Y. Zheng, "Approche bayésienne variationnelle : application à la déconvolution conjointe d'une source ponctuelle dans une source étendue," Septembre p. id 432, *Acte du Grets*, Dijon, France, 2009.

Bibliographie

- [ABMD92] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. Image Process.*, 1(2) :205–220, 1992.
- [ATB03] A. Achim, P. Tsakalides, and A. Bezerianos. SAR image denoising via Bayesian wavelet shrinkage based on heavy-tailed modeling. *IEEE Trans. Geosci. Remote Sens.*, 41(8) :1773–1784, 2003.
- [Att00] H. Attias. A variational Bayesian framework for graphical models. *Advances in neural information processing systems*, 12(1-2) :209–215, 2000.
- [BAMCM97] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso, and E. Moulines. A blind source separation technique using second-order statistics. *IEEE Trans. Signal Process.*, 45(2) :434–444, 1997.
- [BBP05] A. Benazza-Benyahia and J. C. Pesquet. Building robust wavelet estimators for multicomponent images using Stein’s principle. *IEEE Trans. Image Process.*, 14(11) :1814–1830, 2005.
- [BD06] J. M. Bioucas-Dias. Bayesian wavelet-based image deconvolution : a GEM algorithm exploiting a class of heavy-tailed priors. *IEEE Trans. Image Process.*, 15(4) :937–951, 2006.
- [BDFO06] J. Bioucas-Dias, M. Figueiredo, and J. Oliveira. Adaptive total-variation image deconvolution : A majorization-minimization approach. In *Proc. EU-SIPCO*, pages 1–4, 2006.
- [Ber79] J.M. Bernardo. Expected information as expected utility. *The Annals of Statistics*, 7(3) :686–690, 1979.
- [BG02] M. J. Beal and Z. Ghahramani. The variational Bayesian EM algorithm for incomplete data : with application to scoring graphical model structures. In *Bayesian Statistics 7 : Proceedings of the 7th Valencia International Meeting*, page 453, 2002.
- [BI96] S. Brette and J. Idier. Optimized single site update algorithms for image deblurring. In *Image Processing, 1996. Proceedings., International Conference on*, volume 3, pages 65–68. IEEE, 1996.
- [BL07] T. Blu and F. Luisier. The SURE-LET approach to image denoising. *IEEE Trans. Image Process.*, 16(11) :2778–2786, 2007.

- [BMK08] S. D. Babacan, R. Molina, and A. K. Katsaggelos. Parameter estimation in TV image restoration using variational distribution approximation. *IEEE Trans. Image Process.*, 17(3) :326–339, 2008.
- [BMK09] S. D. Babacan, R. Molina, and A. K. Katsaggelos. Variational Bayesian blind deconvolution using a total variation prior. *IEEE Trans. Image Process.*, 18(1) :12–26, 2009.
- [BMK11] S.D. Babacan, R. Molina, and A.K. Katsaggelos. Variational Bayesian super resolution. *IEEE Trans. Image Process.*, 20(4) :984–999, 2011.
- [Can69] J. W. Cantrell. On the relation between the memory gradient and the Fletcher-Reeves method. *J. Optim. Theory Appl.*, 4(1) :67–71, 1969.
- [CBFAB94] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 168–172. IEEE, 1994.
- [CBFAB97] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Trans. Image Process.*, 6(2) :298–311, 1997.
- [CD85] G. Celeux and J. Diebolt. The SEM algorithm : a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational statistics quarterly*, 2(1) :73–82, 1985.
- [CDS01] S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1) :129–159, 2001.
- [CGLS08] G. Chantas, N. Galatsanos, A. Likas, and M. Saunders. Variational Bayesian image restoration based on a product of t -distributions image prior. *IEEE Trans. Image Process.*, 17(10) :1795–1805, October 2008.
- [Cho02] R. A. Choudrey. *Variational Methods for Bayesian Independent Component Analysis*. PhD thesis, University of Oxford, 2002.
- [CI04] F. Champagnat and J. Idier. A connection between half-quadratic criteria and EM algorithms. *IEEE Signal Process. Lett.*, 11(9) :709–712, September 2004.
- [CIM11] E. Chouzenoux, J. Idier, and S. Moussaoui. A Majorize-Minimize strategy for subspace optimization applied to image restoration. *IEEE Trans. Image Process.*, 20(18) :1517–1528, 2011.
- [CKM97] H. A. Chipman, E. D. Kolaczyk, and R. E. McCulloch. Adaptive Bayesian wavelet shrinkage. *Journal of the American Statistical Association*, 92(440) :1413–1421, 1997.
- [CL69] E. E. Cragg and A. V. Levy. Study on a supermemory gradient method for the minimization of functions. *Journal of Optimization Theory and Applications*, 4 :191–205, 1969. 10.1007/BF00930579.

- [CT12] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.
- [CW98] T. F. Chan and C. K. Wong. Total variation blind deconvolution. *IEEE Trans. Image Process.*, 7(3) :370–375, 1998.
- [DDDM04] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on pure and applied mathematics*, 57(11) :1413–1457, 2004.
- [Dem89] G. Demoment. Image reconstruction and restoration : Overview of common estimation structure and problems. *IEEE Trans. Acoust. Speech, Signal Processing*, 37(12) :2024–2036, December 1989.
- [DJ94] D. L. Donoho and J. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3) :425–455, 1994.
- [DJ95] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the american statistical association*, 90(432) :1200–1224, 1995.
- [DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal statistical Society*, 39(1) :1–38, 1977.
- [DMND⁺01] B. De Man, J. Nuyts, P. Dupont, G. Marchal, and P. Suetens. An iterative maximum-likelihood polychromatic algorithm for CT. *IEEE Trans. Med. Imag.*, 20(10) :999–1008, 2001.
- [Don95] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inf. Theory*, 41(3) :613–627, 1995.
- [DS83] R. S. Dembo and T. Steihaug. Truncated-Newton algorithms for large-scale unconstrained optimization. *Mathematical Programming*, 26(2) :190–212, 1983.
- [EF03] I. A. Elbakri and J. A. Fessler. Segmentation-free statistical image reconstruction for polyenergetic x-ray computed tomography with experimental validation. *Physics in Medicine and Biology*, 48(15) :2453, 2003.
- [Eld09] Y. C. Eldar. Generalized SURE for exponential families : Applications to regularization. *IEEE Trans. Signal Process.*, 57(2) :471–481, 2009.
- [EMSZ07] M. Elad, B. Matalon, J. Shtok, and M. Zibulevsky. A wide-angle view at iterated shrinkage algorithms. In *Optical Engineering+ Applications*, pages 670102–670102. International Society for Optics and Photonics, 2007.
- [FR14] A. Fraysse and T. Rodet. A measure-theoretic variational Bayesian algorithm for large dimensional problems. *SIAM Journal on Imaging Sciences.*, 7(4) :2591–2622, 2014.
- [GB01] Z. Ghahramani and M. J. Beal. Propagation algorithms for variational Bayesian learning. *Advances in neural information processing systems*, pages 507–513, 2001.

- [GEE11] R. Giryes, M. Elad, and Y. C. Eldar. The projected GSURE for automatic parameter tuning in iterative shrinkage methods. *Applied and Computational Harmonic Analysis*, 30(3) :407–422, 2011.
- [GG84] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, (6) :721–741, 1984.
- [Gio08] J-F Giovannelli. Unsupervised Bayesian convex deconvolution based on a field with an explicit partition function. *IEEE Trans. Image Process.*, 17(1) :16–26, 2008.
- [GM85] S. Geman and D. McClure. Bayesian image analysis : an application to single photon emission tomography. *Amer. Statist. Assoc.*, pages 12–18, 1985.
- [GOKP02] H. Greenspan, G Oz, N. Kiryati, and S. Peled. MRI inter-slice reconstruction using super-resolution. *Magnetic Resonance Imaging*, 20(5) :437–446, 2002.
- [GP67] A. Goldstein and J. Price. An effective algorithm for minimization. *Numerische Mathematik*, 10 :184–189, 1967.
- [GR92] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3) :367–383, March 1992.
- [GRG06] H. Greenspan, A. Ruf, and J. Goldberger. Constrained Gaussian mixture model framework for automatic segmentation of MR brain images. *IEEE Trans. Med. Imag.*, 25(9) :1233–1245, 2006.
- [GY95] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Trans. Image Process.*, 4(7) :932–946, July 1995.
- [HBJ⁺97] D. M. Higdon, J. E. Bowsher, V. E. Johnson, T. G. Turkington, D. R. Gilland, and R. J. Jaszczak. Fully Bayesian estimation of Gibbs hyperparameters for emission computed tomography data. *IEEE Trans. Med. Imag.*, 16(5) :516–526, oct 1997.
- [HL89] T. Hebert and R. Leahy. A generalized EM algorithm for 3-D Bayesian reconstruction from Poisson data using Gibbs priors. *IEEE Trans. Med. Imag.*, 8(2) :194–202, 1989.
- [HL04] D. R. Hunter and K. Lange. A tutorial on MM algorithms. *Am. Stat.*, 58(1) :30–37, 2004.
- [Hol92] J. H. Holland. *Adaptation in Natural and Artificial Systems : An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, 1992.
- [Hub81] P. J. Huber. *Robust statistics*. Wiley, New York, 1981.
- [Idi01a] J. Idier, editor. *Approche bayésienne pour les problèmes inverses*. Traité IC2, Série traitement du signal et de l’image, Hermès, Paris, nov. 2001.

- [Idi01b] J. Idier. Convex half-quadratic criteria and interacting auxiliary variables for image restoration. *IEEE Trans. Image Process.*, 10(7) :1001–1009, 2001.
- [JGJS99] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2) :183–233, 1999.
- [JJ93] M. Jamshidian and R. I. Jennrich. Conjugate gradient acceleration of the EM algorithm. *Journal of the American Statistical Association*, 88(421) :221–228, 1993.
- [JJ97] M. Jamshidian and R. I. Jennrich. Acceleration of the EM algorithm by using Quasi-Newton methods. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 59(3) :569–587, 1997.
- [JM06] S. Jana and P. Moulin. Optimality of KLT for high-rate transform coding of gaussian vector-scale mixtures : Application to reconstruction, estimation, and classification. *IEEE Trans. Inf. Theory*, 52(9) :4049–4067, 2006.
- [JW91] F. C. Jeng and J. W. Woods. Compound Gauss-Markov random fields for image estimation. *IEEE Trans. Signal Process.*, 39(3) :683–697, 1991.
- [KGV⁺83] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, et al. Optimization by simulated annealing. *science*, 220(4598) :671–680, 1983.
- [KR95] R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American statistical association*, 90(430) :773–795, 1995.
- [Lan95] K. Lange. A quasi-newton acceleration of the EM algorithm. *Statistica sinica*, 5(1) :1–18, 1995.
- [LCA07] T. Le, R. Chartrand, and T. J. Asaki. A variational approach to reconstructing images corrupted by poisson noise. *Journal of Mathematical Imaging and Vision*, 27(3) :257–263, 2007.
- [LG04] C. L. Likas and N. P. Galatsanos. A variational approach for Bayesian blind image deconvolution. *IEEE Trans. Signal Process.*, 52(8) :2222–2233, 2004.
- [LH87] E. Levitan and G. T. Herman. A maximum a posteriori probability expectation maximization algorithm for image reconstruction in emission tomography. *IEEE Trans. Med. Imag.*, 6(3) :185–192, 1987.
- [LT84] LR Lines and S Treitel. Tutorial : A review of least-squares inversion and its application to geophysical problems. *Geophysical prospecting*, 32(2) :159–186, 1984.
- [Mac95] D. J. C. MacKay. Ensemble learning and evidence maximization. <http://citeserx.ist.psu.edu/viewdoc/summary?doi=10.1.1.54.4083>, 1995.
- [Mal89] S. G. Mallat. A theory for multiresolution signal decomposition : the wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(7) :674–693, 1989.

- [Mal00] S. Mallat. *Une exploration des signaux en ondelettes*. Editions Ecole Polytechnique, 2000.
- [MC69] A. Miele and JW Cantrell. Study on a memory gradient method for the minimization of functions. *Journal of Optimization Theory and Applications*, 3(6) :459–470, 1969.
- [Mis00] J. W. Miskin. Ensemble learning for independent component analysis. Technical report, in *Advances in Independent Component Analysis*, 2000.
- [MK07] G. McLachlan and T. Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.
- [ML99] P. Moulin and J. Liu. Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors. *IEEE Trans. Inf. Theory*, 45(3) :909–919, 1999.
- [Mol94] R. Molina. On the hierarchical Bayesian approach to image restoration : Application to astronomical images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(11) :1122–1128, nov 1994.
- [NSLZ07] M. K. Ng, H. Shen, Edmund Y. Lam, and L. Zhang. A total variation regularization based super-resolution reconstruction algorithm for digital video. *EURASIP Journal on Advances in Signal Processing*, 2007, 2007.
- [NW00] J. Nocedal and S.J. Wright. *Numerical Optimization*. Series in Operations Research. Springer Verlag, New York, 2000.
- [NY06] Y. Narushima and H. Yabe. Global convergence of a memory gradient method for unconstrained optimization. *Computational Optimization and Applications*, 35(3) :325–346, 2006.
- [NZ05] G. Narkiss and M. Zibulevsky. *Sequential Subspace Optimization Method for Large-Scale Unconstrained Problems*. Technion-IIT, Department of Electrical Engineering, October 2005.
- [OBDF09] J. P. Oliveira, J. M. Bioucas-Dias, and M. Figueiredo. Adaptive total variation image deblurring : a majorization–minimization approach. *Signal Processing*, 89(9) :1683–1693, 2009.
- [Par88] G. Parisi. *Statistical Field Theory*. Addison Wesley, Redwood City, CA, 1988.
- [PBBC09] J. C. Pesquet, A. Benazza-Benyahia, and C. Chaux. A SURE approach for digital signal/image deconvolution problems. *IEEE Trans. Signal Process.*, 57(12) :4616–4632, 2009.
- [PKDRW05] J. Palmer, K. Kreutz-Delgado, B. D. Rao, and D. P. Wipf. Variational EM algorithms for non-Gaussian latent variable models. In *Advances in neural information processing systems*, pages 1059–1066, 2005.
- [PL97] J-C Pesquet and D. Leporini. A new wavelet estimator for image denoising. 1997.

- [PSWS03] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Process.*, 12(11) :1338–1351, 2003.
- [RC00] C. P. Robert and G. Casella. *Monte-Carlo Statistical Methods*. Springer Texts in Statistics. Springer, New York, 2000.
- [ROF92] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D : Nonlinear Phenomena*, 60(1) :259–268, 1992.
- [ROGA08] T. Rodet, F. Orieux, J.-F. Giovannelli, and A. Abergel. Data inversion for over-resolved spectral imaging in astronomy. *IEEE J. Sel. Top. Sign. Proces.*, 2(5) :802–811, October 2008.
- [RQD00] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1) :19–41, 2000.
- [Rud87] W. Rudin. *Real and Complex Analysis*. McGraw-Hill Book Co., New York, third edition, 1987.
- [SA96] E. P. Simoncelli and E. H. Adelson. Noise removal via Bayesian wavelet coring. In *IEEE International Conference on Image Processing (ICIP)*, volume 1, pages 379–382. IEEE, 1996.
- [SC02] J. h. Shen and T. F. Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3) :1019–1043, 2002.
- [SD00] H. Snoussi and A. M. Djafari. Bayesian source separation with mixture of Gaussians prior for sources and Gaussian prior for mixture coefficients. In *Bayesian Inference and Maximum Entropy Methods*, pages 388–406, Gif-sur-Yvette, July 2000. Proc. of MaxEnt.
- [SD04] H. Snoussi and A. M. Djafari. Bayesian unsupervised learning for source separation with mixture of gaussians prior. *Int. Journal of VLSI Signal Processing Systems*, 37(2-3) :263–279, 2004.
- [SN11] M. W. Seeger and H. Nickisch. Large scale Bayesian inference and experimental design for sparse linear models. *SIAM Journal on Imaging Sciences*, 4(1) :166–199, 2011.
- [ŠQ06] V. Šmídl and A. Quinn. *The Variational Bayes Method in Signal Processing*. Springer, 2006.
- [SS05] Z.J. Shi and J. Shen. A new super-memory gradient method with curve search rule. *Applied mathematics and computation*, 170(1) :1–16, 2005.
- [SS06] Z. J. Shi and J. Shen. Convergence of nonmonotone line search method. *Journal of Computational and Applied Mathematics*, 193(2) :397–412, 2006.
- [SS07] Z.J. Shi and J. Shen. Convergence of supermemory gradient method. *Journal of Applied Mathematics and Computing*, 24 :367–376, 2007. 10.1007/BF02832325.

- [Ste81] C. M. Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, pages 1135–1151, 1981.
- [SW10] M. Seeger and D. P. Wipf. Variational Bayesian inference techniques. *IEEE Signal Processing Magazine*, 27(6) :81–91, 2010.
- [TB93] P. Tseng and D. P. Bertsekas. On the convergence of the exponential multiplier method for convex programming. *Mathematical Programming*, 60(1-3) :1–19, 1993.
- [Tib96] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.
- [Tik63] A. Tikhonov. Regularization of incorrectly posed problems. *Soviet. Math. Dokl.*, 4 :1624–1627, 1963.
- [Vid98] B. Vidakovic. Nonlinear wavelet shrinkage with Bayes rules and Bayes factors. *Journal of the American Statistical Association*, 93(441) :173–179, 1998.
- [VO98] C. R. Vogel and M. E. Oman. Fast, robust total variation-based reconstruction of noisy, blurred images. *IEEE Trans. Image Process.*, 7(6) :813–824, 1998.
- [Wol69] P. Wolfe. Convergence conditions for ascent methods. *SIAM review*, 11(2) :226–235, 1969.
- [WS00] M. J. Wainwright and E. P. Simoncelli. Scale mixtures of gaussians and the statistics of natural images. *Advances in neural information processing systems*, 12(1) :855–861, 2000.
- [WT90] G. Wei and M. A. Tanner. A Monte Carlo implementation of the EM algorithm and the poor man’s data augmentation algorithms. *Journal of the American Statistical Association*, 85(411) :699–704, 1990.
- [Yu08] Z. Yu. Global convergence of a memory gradient method without line search. *Journal of Applied Mathematics and Computing*, 26(1-2) :545–553, 2008.

Titre : *Algorithmes bayésiens variationnels accélérés et applications aux problèmes inverses de grande taille.*

Résumé : Dans le cadre de cette thèse, notre préoccupation principale est de développer des approches non supervisées et efficaces pour les problèmes inverses de grande taille. Pour ce faire, nous avons considéré des approches entièrement bayésiennes. Dans ce cadre, la difficulté principale est que la loi a posteriori est en général complexe. Pour résoudre ce problème, nous nous intéressons à l'approximation bayésienne variationnelle (BV) qui détermine une approximation séparable de la loi *a posteriori*. Néanmoins, les approches BV classiques souffrent d'une vitesse de convergence faible.

La première contribution de cette thèse consiste à transposer les méthodes d'optimisation de sous-espace dans l'espace fonctionnel pertinent dans le cadre BV, ce qui nous permet de proposer une nouvelle méthode d'approximation BV plus efficace. Nous avons montré l'efficacité de notre nouvelle méthode en la comparant aux approches de l'état de l'art.

La deuxième contribution consiste à appliquer notre nouvelle méthodologie à des problèmes de traitement d'images où nous introduisons l'information de régularité par morceau. Pour cela, nous avons considéré trois types d'*a priori* différents : un *a priori* de la variation totale sur les valeurs des pixels, un *a priori* à variables cachées utilisant un mélange de gaussiennes par changement de positions sur les valeurs des pixels et les *a priori* de la famille de mélange de gaussiennes par changement d'échelle (GSM) sur les coefficients d'ondelettes d'images. Ces différentes approches permettent d'avoir de bons résultats sur les images régulières par morceau.

Title : *Fast variational Bayesian algorithms and application to large dimensional inverse problems.*

Abstract : In this thesis, our main objective is to develop efficient unsupervised approaches for large dimensional inverse problems. To do this, we consider fully Bayesian approaches. In this context, the major difficulty is that the posterior distribution is generally complex. To tackle this problem, we consider variational Bayesian (VB) approximation, which determines a separable approximation of the posterior distribution. Nevertheless, classical VB approaches suffer from slow rate of convergence.

The first contribution of this thesis is to transpose the subspace optimization methods to the functional space involved in the VB framework, which allows us to propose a more efficient VB approximation method. We have shown the efficiency of the proposed method by comparisons with state of the art approaches.

The second contribution is to apply our new methodology to image processing problems where numerous images are piecewise smooth. To introduce such prior information, we have considered three types of prior distributions : a total variation prior for the pixel values, a Gaussian location mixture-like prior for the pixel values and sparse priors of the Gaussian scale mixture family for the wavelet coefficients of images. These approaches allow us to obtain good results for piecewise smooth images.