



CPPM-T-2019-02

AIX-MARSEILLE UNIVERSITÉ  
Faculté des Sciences  
163, avenue de Luminy  
13288 Marseille cedex 09

## THÈSE DE DOCTORAT

*Spécialité : Optique, photonique et traitement d'image*

présentée par

# Souhil TAIRI

en vue d'obtenir le grade de docteur d'Aix-Marseille Université

## **Développement de méthodes itératives pour la reconstruction en tomographie spectrale**

Soutenue le 20 juin 2019 devant le jury composé de :

Françoise PEYRIN	Rapporteur
Émilie CHOUZENOUX	Rapporteur
Frederic NOO	Examineur
Francis VERDUN	Examineur
Cristinel DIACONU	Examineur
Sandrine ANTHOINE	Co-directrice de thèse
Yannick BOURSIER	Co-directeur de thèse
Christian MOREL	Directeur de thèse

---

Cette thèse a été co-financée par la région Sud Provence-Alpes-Côtes d'Azur





# Table des matières

<b>Table des matières</b>	<b>III</b>
<b>Liste des figures</b>	<b>VII</b>
<b>Liste des tableaux</b>	<b>XI</b>
<b>Remerciements</b>	<b>1</b>
<b>Résumé</b>	<b>5</b>
<b>Introduction générale</b>	<b>7</b>
<b>1 Introduction</b>	<b>11</b>
1.1 Problématique Générale	12
1.1.1 Contexte et enjeux	12
1.1.2 Problématique	14
1.1.2.1 Ce que l'on attend de la TDM spectrale	14
1.1.2.2 Modèle	16
1.1.2.3 Modèle de l'objet	17
1.1.2.4 Discrétisation	18
1.1.2.5 Problème mathématique	18
1.1.3 En bref	20
1.2 Physique d'acquisition	21
1.2.1 Physique de la TDM spectrale	21
1.2.1.1 Les rayons X et leur production	21
1.2.1.2 L'interaction photon-matière	22
1.2.1.3 Atténuation des photons	25
1.2.1.4 Détection	27
1.2.1.5 Conclusion	28
1.2.2 Le prototype PIXSCAN-FLI	28
1.2.2.1 Introduction	28
1.2.2.2 Description	28
1.2.2.3 Le bloc moteur	29
1.2.2.4 Le bloc source	30
1.2.2.5 Le bloc détecteur	32
1.2.2.6 Conclusion	35
1.2.3 Tomographie	35
1.2.3.1 Introduction	35
1.2.3.2 Projection	36
1.2.3.3 Transformée de Radon	37
1.2.3.4 Rétroprojection filtrée en faisceau parallèle	39
1.2.3.5 Rétroprojection filtrée en faisceaux divergents	41
1.2.3.6 Rétroprojection filtrée en faisceau conique	43
1.2.3.7 Conclusion	45

1.2.4	En bref	46
1.3	Méthodes et état de l'art	47
1.3.1	TDM d'absorption classique	47
1.3.1.1	Modélisation	47
1.3.1.2	Expression du problème	49
1.3.1.3	Recherche de solution	51
1.3.1.4	Méthodes algébriques	51
1.3.1.5	Méthodes par maximum de vraisemblance	52
1.3.1.6	Conclusion	52
1.3.2	TDM spectrale	54
1.3.2.1	Rappel du problème	54
1.3.2.2	Motivation	55
1.3.2.3	Acquisition de données en TDM spectrale	56
1.3.2.4	État de l'art reconstruction TDM spectrale	58
1.3.2.5	État de l'art algorithmique	60
1.4	Références	63
<b>2</b>	<b>Notations et généralités sur les problèmes inverses</b>	<b>69</b>
2.1	Problème inverse	70
2.1.1	Généralités	70
2.1.2	Optimisation	72
2.1.3	Notations et définitions	72
2.1.3.1	(Non) Convexité	73
2.1.3.2	Optimum	75
2.1.3.3	Identifier un optimum	76
2.2	Outils mathématiques	76
2.2.1	Notion de sous-différentiel	77
2.2.1.1	Cas convexe	78
2.2.1.2	Cas dimension 1	79
2.2.1.3	Conclusion	80
2.2.2	Opérateurs proximaux	80
2.2.2.1	Intérêt	80
2.2.2.2	Définitions	80
2.2.2.3	Propriétés	81
2.2.2.4	Cas particuliers	82
2.2.2.5	Conclusion	84
2.2.3	Inégalité de Kurdyka-Łojasiewicz	84
2.2.3.1	Idée de base	84
2.2.3.2	Inégalité de Łojasiewicz	85
2.2.3.3	Inégalité de Kurdyka-Łojasiewicz	85
2.3	Définitions des fonctions objectifs	86
2.3.1	Fonction d'attache aux données	87
2.3.1.1	Bruit gaussien corrélé	87
2.3.1.2	Bruit blanc gaussien	87
2.3.1.3	Bruit de Poisson	87
2.3.2	Fonction de régularisation	88
2.3.2.1	Contraintes	89
2.3.2.2	Parcimonie	89
2.3.2.3	Variation totale	91
2.4	Algorithmes d'optimisation	91
2.4.1	Algorithmes à direction de descente	92
2.4.1.1	Descente de gradient	92
2.4.1.2	Descente de Newton	93

2.4.1.3	Descente de gradient projeté . . . . .	94
2.4.1.4	Conclusion . . . . .	95
2.4.2	Algorithmes de région de confiance . . . . .	95
2.4.2.1	Solution « exacte » . . . . .	97
2.4.2.2	Méthode de la patte de chien ou <i>dog-leg</i> et de la minimisation en sous-espace à deux dimensions . . . . .	98
2.4.2.3	Point de Cauchy . . . . .	99
2.4.2.4	Remarque sur les contraintes . . . . .	99
2.4.2.5	Conclusion . . . . .	100
2.4.3	Algorithmes proximaux . . . . .	100
2.4.3.1	Introduction . . . . .	100
2.4.3.2	Algorithme ISTA . . . . .	100
2.4.3.3	Algorithme FISTA . . . . .	102
2.4.3.4	Algorithme proximal à métrique variable . . . . .	104
2.4.3.5	Algorithme VMILAn . . . . .	107
2.5	Références . . . . .	110
<b>3</b>	<b>Méthodes et Résultats</b> . . . . .	<b>115</b>
3.1	Les méthodes développées . . . . .	116
3.1.1	Rappel du problème de TDM spectrale . . . . .	118
3.1.2	Les fonctions objectifs . . . . .	119
3.1.2.1	Cas gaussien . . . . .	119
3.1.2.2	Cas Poisson . . . . .	120
3.1.3	Choix de la métrique . . . . .	121
3.1.3.1	Cas bruit gaussien . . . . .	121
3.1.3.2	Cas bruit de Poisson . . . . .	122
3.1.4	Filtre conique . . . . .	123
3.1.4.1	Appodisation du filtre . . . . .	124
3.1.5	Choix du pas de descente . . . . .	124
3.1.5.1	Algorithme FISTA . . . . .	124
3.1.5.2	Algorithmes VMFB et VMILAn . . . . .	125
3.2	Tests et résultats sur données simulées . . . . .	126
3.2.1	Tests sur données de TDM d'absorption classique . . . . .	126
3.2.1.1	Protocole de test . . . . .	126
3.2.1.2	Cas linéaire simple sans bruit . . . . .	128
3.2.1.3	Cas linéaire simple avec bruit . . . . .	129
3.2.1.4	Cas non-linéaire sans bruit . . . . .	130
3.2.1.5	Cas non-linéaire avec bruit . . . . .	133
3.2.1.6	Conclusion . . . . .	135
3.2.2	Tests sur données polychromatiques 2D . . . . .	135
3.2.2.1	Cadre de test . . . . .	135
3.2.2.2	Résultats obtenus avec l'algorithme à région de confiance . . . . .	136
3.2.2.3	Résultats obtenus avec FISTA et VMFB . . . . .	140
3.2.3	Tests sur donnée polychromatique 3D . . . . .	143
3.2.3.1	Résultats obtenus pour des données spectrales 3D non bruitées . . . . .	146
3.2.3.2	Résultats obtenus pour 1000 coups dans les pixels situés en dehors de l'objet projeté . . . . .	147
3.2.3.3	Remarque sur l'effet de l'ouverture du faisceau . . . . .	151
3.2.3.4	Cas pour bruit plus élevé et une ouverture de faisceau plus grande . . . . .	153
3.2.3.5	Étude de bruit . . . . .	157
3.2.3.6	Robustesse aux perturbations engendrées par les filtres . . . . .	158
3.2.4	Conclusion sur les données simulées . . . . .	165
3.3	Tests et résultats sur données réelles . . . . .	165

3.3.1	Acquisition de données le prototype PIXSCAN-FLI . . . . .	165
3.3.1.1	Interface d'acquisition . . . . .	165
3.3.1.2	Calibrations du prototype PIXSCAN-FLI . . . . .	166
3.3.2	Résultats sur données réelles . . . . .	172
3.3.2.1	Fantôme à micro-tubes Eppendorf . . . . .	172
3.3.2.2	Fantôme torsadé . . . . .	186
3.4	Références . . . . .	192
<b>Conclusion générale</b>		<b>195</b>
<b>A</b>	<b>Annexes</b>	<b>I</b>
A.1	Démonstrations . . . . .	II
A.1.1	Coupe centrale . . . . .	II
A.1.2	Rétroprojection filtrée . . . . .	II
A.1.3	Rétroprojection filtrée de faisceau en éventail . . . . .	IV
A.1.4	Démonstration sous problème de minimisation en sous-espace à deux dimensions . . . . .	VIII
A.1.5	Démonstration passage au proximal ISTA . . . . .	IX
A.1.6	Démonstration $G_1(A)$ analytique réelle pour le cas gaussien . . . . .	IX
A.1.7	Démonstration $G_2(A)$ analytique réelle pour le cas Poisson . . . . .	X
A.1.8	Calcul du gradient et de la hessienne de $G(A)$ pour le cas Gaussien . . . . .	XI
A.1.8.1	Prérequis . . . . .	XI
A.1.8.2	Jacobien . . . . .	XIII
A.1.8.3	Hessienne . . . . .	XIV
A.1.9	Calcul du gradient et de la hessienne de $G(A)$ pour le cas Poisson . . . . .	XV
A.1.9.1	Gradient . . . . .	XV
A.1.9.2	Hessienne . . . . .	XVI
A.1.10	Démonstration filtre conique . . . . .	XVIII
A.2	Références . . . . .	XXII

# Liste des figures

1.1	Schéma scanner et sinogramme	12
1.2	Schéma d'un cylindre d'eau contenant différents agents de contraste	15
1.3	Schéma reconstruction en <b>TDM</b> standard	15
1.4	Schéma reconstruction en <b>TDM</b> spectrale	15
1.5	Schéma d'acquisition en <b>TDM</b>	16
1.6	Schéma acquisition spectrale	17
1.7	Schéma problème spectrale	21
1.8	Représentation du Spectre	22
1.9	Processus du bremsstrahlung	22
1.10	Prédominance des effets	23
1.11	Photoélectrique	23
1.12	Section efficace	24
1.13	Effet Compton	25
1.14	Illustration de l'atténuation d'un faisceau	26
1.15	Illustration de la matrice A	27
1.16	La boîte du prototype PIXSCAN-FLI	29
1.17	Les blocs du prototype PIXSCAN-FLI	29
1.18	Le bloc du moteur	30
1.19	Le bloc source	30
1.20	Spectre à 90 kV de la source de tungstène	31
1.21	Spectre à 60 kV de la source de tungstène	31
1.22	Spectres en énergie modulés	32
1.23	Schéma détecteur à pixel hybrides	33
1.24	Le bloc détecteur : caméra XPAD3	34
1.25	Efficacité du détecteur	34
1.26	Efficacité du détecteur et influence spectre mesuré	35
1.27	Exemple de Projection	36
1.28	Géométrie des faisceaux	36
1.29	Exemple de modification de taille du à la géométrie	37
1.30	Exemple de transformée de Radon	38
1.31	Exemple de projection	39
1.32	Exemple de reconstrction	40
1.33	Reconstruction des billes	40
1.34	Exemple de combinaison de filtre rampe	41
1.35	Illustration des faisceaux divergents	42
1.36	Illustration de la géométrie fan-beam	42
1.37	Illustration de la condition de Tuy	43
1.38	Illustration du système de coordonnées de <b>FDK</b>	44
1.39	Illustration de la géométrie conique	45
1.40	Illustration du principe de discrétisation	48
1.41	Illustration des différentes modélisation de projections.	48
1.42	Angiographie de l'aorte à l'aide d'un agent de contraste iodé.	53

1.43 Séparation automatique des os . . . . .	55
1.44 Illustration de la méthode du kV switching . . . . .	56
1.45 Illustration du <b>dual-energy</b> . . . . .	57
1.46 Illustration du détecteur à double couches . . . . .	57
1.47 Illustration du détecteur à double couches . . . . .	60
2.1 Illustration de la convexité . . . . .	74
2.2 Illustration de la non convexité . . . . .	75
2.3 Différentiel de valeur absolue . . . . .	79
2.4 Opérateur de seuillage doux et dur . . . . .	84
2.5 Illustration de la parcimonie . . . . .	90
2.6 Illustration de la norme $l_\alpha$ . . . . .	90
2.7 Illustration de la méthode de région de confiance . . . . .	96
2.8 Illustration de la vitesse de convergence de l'algorithme FISTA . . . . .	104
3.1 Schéma d'un cylindre d'eau contenant différents agents de contrastes . . . . .	116
3.2 Schéma reconstruction en <b>TDM</b> standard . . . . .	117
3.3 Schéma reconstruction en <b>TDM</b> spectrale . . . . .	117
3.4 Filtre conique apodisé . . . . .	124
3.5 Fantôme de Shepp-Logan . . . . .	127
3.6 Sinogramme du protocole de test sur donnée monochromatique . . . . .	127
3.7 Décroissance des fonctions objectifs pour le test des données monochromatique sans bruit . . . . .	128
3.8 Images résiduelles de la reconstruction . . . . .	129
3.9 Décroissance des fonctions objectifs pour le test des données monochromatiques bruitées . . . . .	129
3.10 Images résiduelles de la reconstruction . . . . .	130
3.11 Résultat de décroissance des fonctionnelles pour $M_{nl}^1$ . . . . .	131
3.12 Résultat des images résiduelles pour $M_{nl}^1$ . . . . .	132
3.13 Résultat de décroissance des fonctionnelles pour $M_{nl}^2$ . . . . .	132
3.14 Résultat des images résiduelles pour $M_{nl}^2$ . . . . .	133
3.15 Résultat de décroissance des fonctionnelles pour $M_{nl}^1$ pour le cas avec bruit . . . . .	133
3.16 Résultat des images résiduelles pour $M_{nl}^1$ pour le cas avec bruit . . . . .	134
3.17 Résultat de décroissance des fonctionnelles pour $M_{nl}^2$ . . . . .	134
3.18 Résultat des images résiduelles pour $M_{nl}^2$ pour le cas avec bruit . . . . .	134
3.19 Fantôme de contraste pour l'expérience données 2D simulées . . . . .	135
3.20 Illustration des filtres passe-bande . . . . .	137
3.21 Sections efficaces pour les agents de contraste yttrium, argent et iode . . . . .	137
3.22 Sinogramme du protocole de test sur les données 2D spectrales . . . . .	138
3.23 Cartes reconstruites avec l'algorithme à région de confiance . . . . .	138
3.24 Évolution du <b>SNR</b> de la reconstruction en fonction du <b>SNR</b> des données . . . . .	139
3.25 Comparaison des décroissances de <b>FISTA</b> et <b>VMFB</b> sur les données 2D spectrales non bruitées . . . . .	141
3.26 Reconstructions générées par <b>FISTA</b> et <b>textbfVMFB</b> sur les données 2D spectrales non bruitées . . . . .	141
3.27 Comparaison des décroissances de <b>FISTA</b> et <b>VMFB</b> sur les données 2D spectrales bruitées . . . . .	142
3.28 Reconstructions générées par <b>FISTA</b> et <b>textbfVMFB</b> sur les données 2D spectrales bruitées . . . . .	142
3.29 Schéma d'un cylindre d'eau contenant différents agents de contraste pour le test des données <b>TDM</b> spectrale 3D . . . . .	143

3.30	Illustration des filtres passe-bande et des sections efficaces pour les agents de contraste baryum, argent et iode . . . . .	144
3.31	Projections issues de test sur les données 3D spectrales . . . . .	145
3.32	Évolution du <b>SNR</b> et de la fonction objectif $J(\mathbf{A})$ pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations . . . . .	146
3.33	Évolution du <b>SNR</b> des composants pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations	146
3.34	Profils de reconstruction de l'argent . . . . .	147
3.35	Reconstruction obtenue avec <b>VMILAn</b> et <b>FISTA</b> sur données 3D spectrale sans bruit	148
3.36	Évolution du <b>SNR</b> et de la fonction objectif $J(\mathbf{A})$ pour <b>FISTA</b> et <b>VMILAn</b> en fonction du nombre d'itérations sur donnée bruitées . . . . .	149
3.37	Évolution du <b>SNR</b> des composants pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations	149
3.38	Reconstruction obtenue avec <b>VMILAn</b> et <b>FISTA</b> avec des données 3D spectrale avec un taux de comptage à 1000 . . . . .	150
3.39	Reconstruction obtenue avec <b>VMILAn</b> et <b>FISTA</b> sur données 3D spectrale hors plan médian . . . . .	152
3.40	Projections obtenues avec le test sur les données 3D spectrales avec un taux de comptage de 300 dans les blancs . . . . .	153
3.41	Évolution du <b>SNR</b> et de la fonction objectif pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations . . . . .	153
3.42	Évolution du <b>SNR</b> des composants pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations	154
3.43	Reconstruction obtenue avec <b>VMILAn</b> et <b>FISTA</b> sur données 3D spectrale avec un taux de comptage à 300 . . . . .	155
3.44	Reconstruction obtenue avec <b>VMILAn</b> et <b>FISTA</b> sur données 3D spectrale hors plan médian . . . . .	156
3.45	Profils de reconstruction de l'eau à un taux de comptage de 300 . . . . .	157
3.46	Étude de bruit : <b>SNR</b> global et eau . . . . .	157
3.47	Étude de bruit : <b>SNR</b> iode, argent et baryum . . . . .	158
3.48	Filtre en énergie perturbé . . . . .	159
3.49	Étude de perturbation du filtre : <b>SNR</b> global et eau . . . . .	159
3.50	Étude de perturbation du filtre : <b>SNR</b> iode, argent et baryum . . . . .	160
3.51	Filtre en énergie sous échantillonné . . . . .	161
3.52	Comparaison de l'évolution du <b>SNR</b> et de la fonction objectif $J(\mathbf{A})$ pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations pour <b>F</b> sous-échantillonné . . . . .	162
3.53	Comparaison de l'évolution du <b>SNR</b> pour <b>FISTA</b> et <b>VMILAn</b> en fonction des itérations pour <b>F</b> sous-échantillonné . . . . .	162
3.54	Reconstruction obtenue avec <b>FISTA</b> sur données 3D spectrale avec <b>F</b> sous-échantillonnée	163
3.55	Reconstruction obtenue avec <b>FISTA</b> sur données 3D spectrale avec <b>F</b> normale . . . . .	163
3.56	Reconstruction obtenue avec <b>VMILAn</b> sur données 3D spectrale avec <b>F</b> sous-échantillonnée	164
3.57	Reconstruction obtenue avec <b>VMILAn</b> sur données 3D spectrale avec <b>F</b> normale . . . . .	164
3.58	Fantôme de la bille de calibration géométrique et trajectoire de projection . . . . .	167
3.59	Réponse d'un pixel pour la calibration . . . . .	169
3.60	Fonds blancs de la 4 ème barrette du XPAD3 . . . . .	170
3.61	Histogramme des fonds blanc du XPAD3 . . . . .	171
3.62	Sections efficaces et filtres pour l'eau, l'argent et l'iode . . . . .	173
3.63	Filtres idéal et filtres réelles . . . . .	173
3.64	Fantôme à microtubes Eppendorf . . . . .	174
3.65	Valeurs HU en fonction Concentration . . . . .	175
3.66	Valeurs HU simulées en fonction des HU mesurée . . . . .	175
3.67	Décroissance des résidus et de la fonctionnelle $J(\mathbf{A})$ pour <b>FISTA</b> et <b>VMILAn</b> . . . . .	177
3.68	Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 100 itérations sur le fantôme à microtubes Eppendorf . . . . .	178

3.69 Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 500 itérations sur le fantôme à microtubes Eppendorf . . . . .	179
3.70 Contraste pure des différentes méthodes . . . . .	181
3.71 Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 500 itérations de l'insert Ag 11 . . . . .	181
3.72 Concentration de l'eau . . . . .	182
3.73 Valeurs HU simulées en fonction des HU mesurée avec les concentrations estimées . . . . .	183
3.74 Filtres pour l'eau, l'argent et l'iode sans partage de charges . . . . .	183
3.75 Décroissance des résidus et de la fonctionnelle $J(A)$ pour <b>FISTA</b> et <b>VMILAn</b> sans partage de charges . . . . .	184
3.76 Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 500 itérations sur le fantôme à microtubes Eppendorf . . . . .	185
3.77 Fantôme en hélice . . . . .	186
3.78 Filtres pour le fantôme torsadé . . . . .	187
3.79 Reconstruction par l'algorithme <b>FDK</b> plan médian . . . . .	187
3.80 Décroissance des résidus et de la fonctionnelle $J(A)$ pour <b>FISTA</b> et <b>VMILAn</b> pour le fantôme torsadé . . . . .	188
3.81 Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 100 itérations sur le fantôme torsadé . . . . .	188
3.82 Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 500 itérations pour le fantôme torsadé . . . . .	189
3.83 Reconstruction et séparation par les algorithmes <b>VMILAn</b> et <b>FISTA</b> à 500 itérations du tube d'argent du fantôme torsadé . . . . .	190
3.84 Contraste pure des différentes méthodes pour le fantôme torsadé . . . . .	190
A.1 Illustration de la géométrie en éventail . . . . .	IV
A.2 Illustration de la géométrie en éventail . . . . .	VI



# Liste des tableaux

3.1	Table des valeurs pour l'argent . . . . .	180
3.2	Table des valeurs pour l'iode . . . . .	180
3.3	Table des valeurs pour l'argent (filtre sans partage de charges) . . . . .	184
3.4	Table des valeurs pour l'argent et l'iode du fantôme torsadé . . . . .	190



# Remerciements

Je voudrais dans un premier temps remercier le lecteur de ce manuscrit. Après tout, mes mots n'ont de vie que dans l'esprit de ceux qui les lisent.

Sache, cher lecteur, que ce manuscrit est le résultat d'une longue histoire, et que je ne suis pas le seul protagoniste de celle-ci. La graine qui a fait naître en moi un fort intérêt pour les mathématiques a été semée par **Pierre Stambul**, mon professeur de mathématiques en classe terminale. C'est sa forte détermination à transmettre le savoir qui m'a motivé à continuer dans cette voie malgré mon un niveau scolaire que je qualifierai de médiocre.

Durant mon parcours de licence et de master, j'ai rencontré beaucoup de personnes qui ont contribué à maintenir cette motivation. Parmi elles, **Jean-Yves Briend**, qui m'a transmis cette belle leçon de vie : les choix que l'on fait dans la vie ne doivent dépendre que de ce qui nous rend heureux. J'essaye jusqu'à ce jour d'appliquer cet enseignement.

Plus tard en master, j'ai découvert la branche qui me passionne le plus en mathématiques : le traitement du signal. Cette passion m'a été transmise par **Bruno Torrèsani**, qui m'a accompagné et encadré sur plusieurs projets tout au long de mon master. C'est à lui que je dois aussi mon stage au laboratoire de contrôle et traitement du son (SPCL) de l'université de McGill à Montréal.

Ce stage était l'élément clé qui a permis mon recrutement en alternance en deuxième année de master. Recrutement aussi rendu possible grâce à **Frédéric Richard**, qui était le responsable du master et mon professeur pendant plusieurs années. Il a été l'initiateur de mon parcours professionnel et c'est grâce à lui que l'offre de cette thèse m'est parvenue.

Si Frédéric Richard m'a autant aidé, c'est parce que j'avais obtenu de très bonnes notes en traitement du signal. Notes que je n'aurais jamais obtenu sans **Clothilde Mélot** qui a eu la gentillesse et la patience de m'expliquer les choses en détails et plusieurs fois, car hélas je n'ai pas l'esprit vif.

M'engager dans cette thèse a été une évidence pour moi au moment où j'ai découvert que **Sandrine Anthoine** encadrerait le projet. Avant d'être son thésard, j'étais son étudiant en problèmes inverses. Ce module allait être annulé car hormis moi, aucun autre étudiant de la promotion ne s'y était inscrit. Pourtant, Sandrine a bien voulu prendre en charge le cours malgré tout. Cet enseignement m'a été d'une extrême utilité durant ma thèse, car j'y ai appris les bases des problèmes inverses appliqués à la tomographie.

Dans le travail, Sandrine est minutieuse et acharnée : elle ne laisse rien au hasard. Elle a su me pousser dans mes retranchements et m'a accompagné dans ma réflexion, non pas en me donnant les réponses aux questions que je posais, mais en m'apprenant à me poser les bonnes questions. Cependant, son acharnement dans le travail ne rivalise pas avec la bienveillance et la patience qu'elle a eues à mon égard. C'est à elle que je dois une bonne partie de mon évolution en tant que thésard.

Cette thèse ne serait que la moitié de ce qu'elle est si **Yannick Boursier** n'avait pas encadré le projet. Pour être honnête, Yannick a tellement été présent pour m'aider durant ce travail que j'ai eu du mal à me l'approprier. Il a toujours eu les mots justes, ni abrupts ni âpres, bien au contraire. Ses phrases étaient toujours délicates, ses conseils bienveillants et bien intentionnés. Il a su chaapeuter mon projet avec toutes les compétences que requiert ce travail sans pour autant y mettre

le conformisme hiérarchique habituel entre l'encadrant et son thésard. Il a accompli cette mission comme un ami l'aurait fait, et grâce lui ces années de thèse ont été jusqu'ici les plus belles de ma vie.

Mon directeur de thèse **Christian Morel** a toujours été disponible aux moments clés. Il a particulièrement été présent durant la rédaction du manuscrit. Il a pris le temps de me corriger jusqu'à la moindre virgule et il a su pousser le travail jusqu'au bout afin d'obtenir les beaux résultats que le lecteur pourra lire en fin du chapitre 3. J'ai rarement croisé une personne autant dévouée au travail. Toujours parmi les premiers arrivés au laboratoire, et des derniers à en sortir. Chef de l'équipe imXgam, une petite équipe pluridisciplinaire pour laquelle il se démène afin de la faire vivre et prospérer. J'aurais aimé être un meilleur thésard, j'espère seulement laisser une bonne trace au sein de cette équipe à laquelle je suis fier d'avoir fait partie.

Si ce manuscrit a aujourd'hui une validité et une valeur, c'est parce que **Françoise Peyrin** et **Émilie Chouzenoux** ont accepté la charge d'en être les rapporteuses. De même, **Frédéric Noo**, **Francis Verdun** et **Cristinel Diaconu** ont été les examinateur de mon travail. Ils ont fourni un considérable effort de lecture et d'évaluation de mon manuscrit et c'est à eux que je dois aujourd'hui la validité de mon diplôme.

En règle générale, il y a trois élément clés au bon déroulement d'une thèse : un bon encadrement, un sujet motivant et un bon environnement de travail. J'ai eu l'encadrement dont rêvent tous les thésards, le sujet de la thèse me passionne, et je ne pouvais pas demander mieux que le Centre de Physique des Particules de Marseille (CPPM) comme laboratoire. J'ai une pensée pour toutes les personnes du CPPM qui ont contribué à rendre ce séjour merveilleux, ne serait-ce que par un sourire ou une poignet de main.

Comme tu le vois, cher lecteur, la brève description du parcours qui m'a conduit jusqu'au doctorat montre une imbrication d'une fluidité remarquable. Et pour être honnête, jamais je n'aurais pensé arriver là lorsque je regarde d'où je viens. Les personnes que j'ai précédemment citées ont été d'une importance capitale à cela, mais il y d'autres personnes tout autant importantes qui m'ont aidé à leurs manières.

En parallèle de mes études, je suis combattant professionnel de *Mixed Martial Arts* (MMA), un sport de combat libre qui se pratique dans une cage fermée, et où l'on peut mélanger toutes les techniques de combat qui existent. Le MMA est interdit en France, la première contrainte est donc de sortir des frontières pour pouvoir combattre. Une autre contrainte est l'énorme sacrifice physique et morale que requiert la préparation d'un combat, car combattre c'est avant tout un art de vivre. S'ajoutent à cela les incessantes remarques d'un grand nombre de personnes qui ne connaissent ce sport que de l'extérieur.

Si aujourd'hui je suis arrivé à ce stade c'est en premier lieu grâce à mon cher ami **Kheir-Eddine Bounadja**. Il m'a poussé dans cette salle Savinoise, un club de noble-art rustre respirant le camphre et la sueur et dirigé par **Mourad Haddu**. Ils sont les initiateurs de mon parcours de combattant et des modèles que j'ai longtemps suivis.

Après m'être essayé à d'autres disciplines, je suis arrivé à celle qui les mélange toutes : le MMA. Je suis passé professionnel au moment où j'ai commencé cette thèse, mais mon niveau est très modeste, je ne suis ni champion ni invaincu. Mes quelques victoires ne m'ont pas rendu spécialement plus fort, tout comme mes défaites ne m'ont pas affaibli. Cependant, la chose dont je suis fier, c'est d'avoir rallié deux domaines qui semblent à première vue antagonistes : beaucoup m'ont dit qu'il était impossible de mener de front une pratique sportive et des études de haut niveau. Mais bon, beaucoup ne savent pas que Pythagore était aussi pugiliste.

Tout cela a été rendu possible car il y a trois ans de cela **Jean-Michel Foissard**, mon coach actuel, a bien voulu croire en moi et me prendre sous son aile. Il a su faire sortir le meilleur de moi, m'a transmis la passion du sport et m'a formé afin de devenir son bras droit pour entraîner à mon tour des combattants. Je lui dois ma construction, et j'ai une pensée pour tous mes frères d'arme du club, de véritables guerriers avec lesquels j'ai l'honneur de m'entraîner.

Que serais-je sans mes proches ? Je ne peux répondre à cette question, car je n'existerais pas sans eux. Il y a mes amis et il y a ma famille : mon grand frère **Sami** qui est mon coach de vie depuis ma naissance, mes autres frères et mes sœurs qui me chérissent et me soutiennent de tout leur être depuis qu'ils m'ont vu naître. Mes chers parents, ces deux héros à la place extraordinaire dans mon cœur pour qui je dédie ma cause. Je m'excuse d'avoir été un courant d'air ces dernières années, je m'excuse de ne pas être à la hauteur de ce que vous méritez, je m'excuse de mes silences et de mes absences. Je vous dois la vie.

Pour achever cette lettre, comme tu le constates cher lecteur, il y a beaucoup de protagonistes dans l'histoire de cette thèse. De simples remerciements ne peuvent récompenser ce que ces personnes m'ont donné car la valeur en est inestimable à mes yeux. Alors à vous tous, vous avez ma profonde gratitude, vous avez donné un sens à ma vie.



# Résumé

Depuis quelques années les détecteurs à pixels hybrides ont ouvert la voie au développement de la tomographie à rayon X spectrale ou tomodensitométrie (TDM) spectrale.

La TDM spectrale permet d'extraire plus d'information concernant la structure interne de l'objet par rapport à la TDM d'absorption classique. Un de ses objectifs dans l'imagerie médicale est d'identifier et quantifier des composants d'intérêt dans un objet, tels que des marqueurs biologique appelés agents de contraste (iode, baryum, etc.).

La majeure partie de l'état de l'art procède en deux étapes : - la "pré-reconstruction" qui consiste à séparer les composants dans l'espace des projections puis reconstruire, - la "post-reconstruction", qui reconstruit l'objet puis sépare les composants.

On s'intéresse dans ce travail de thèse à une approche qui consiste à séparer et reconstruire simultanément les composants de l'objet. L'état de l'art des méthodes de reconstruction et séparation simultanées de données de TDM spectrale reste à ce jour peu fourni et les approches de reconstruction existantes sont limitées dans leurs performances et ne tiennent souvent pas compte de la complexité du modèle d'acquisition.

L'objectif principal de ce travail de thèse est donc de proposer des approches de reconstruction et séparation tenant compte de la complexité du modèle afin d'améliorer la qualité des images reconstruites. Notre contribution considère le modèle polychromatique non-linéaire du faisceau de rayons X et le combine avec un modèle antérieur sur les composants de l'objet à reconstruire. Le problème ainsi obtenu est un problème inverse mal-posé et non linéaire qui peut se traduire en un problème d'optimisation non-convexe et de très grande dimension.

Pour le résoudre, nous proposons un algorithme proximal à métrique variable. C'est un algorithme itératif qui permet de garantir une solution locale même lorsque le problème est non-convexe. La performance de l'approche proposée est analysée avec des données simulées et réelles acquises sur le prototype de micro-CT spectral PIXSCAN-FLI développé au Centre de Physique des Particules de Marseille (CPPM) pour France Life Imaging (FLI).

Des résultats prometteurs sont montrés sur les données 3D avec une simple régularisation qui englobe la positivité des quantités d'intérêt. Ils montrent que l'approche proposée permet une bonne séparation et reconstruction malgré la présence de bruit (gaussien ou Poisson). Par rapport aux approches existantes similaires telle que la descente proximale accélérée FISTA, l'approche proposée présente des avantages sur la vitesse de convergence.

## Abstract

In recent years, hybrid pixel detectors have paved the way for the development of spectral x-ray tomography or spectral tomography (CT).

Spectral CT provides more information about the internal structure of the object compared to conventional absorption CT. One of its objectives in medical imaging is to obtain images of com-

ponents of interest in an object, such as biological markers called contrast agents (iodine, barium, etc.).

The state of the art of simultaneous reconstruction and separation of spectral CT data methods remains to this day limited. Existing reconstruction approaches are limited in their performance and often do not take into account the complexity of the acquisition model.

The main objective of this thesis work is to propose better quality reconstruction approaches that take into account the complexity of the model in order to improve the quality of the reconstructed images. Our contribution considers the non-linear polychromatic model of the X-ray beam and combines it with an earlier model on the components of the object to be reconstructed. We thus obtain an ill-posed inverse problem that may be solved by a non-convex optimization problem of very large dimensions.

To solve it, we propose a proximal algorithm with variable metrics. It is an iterative algorithm that guarantees a local solution even when the problem is non-convex. The performance of the proposed approach is analyzed with simulated and real data acquired on the PIXSCAN-FLI micro-CT spectral prototype developed at the Centre de Physique des Particules de Marseille (CPPM) for France Life Imaging (FLI).

Promising results are shown on the 3D data with a simple regularization that includes the positivity of the quantities of interest. They show that the proposed approach allows good separation and reconstruction despite the presence of noise (Gaussian or Poisson). Compared to existing approaches, the proposed approach has advantages over the speed of convergence.



# Introduction générale

Le **tomodensitomètre** ou **scanner à rayons X** représente l'une des modalités d'imagerie clinique les plus courantes. On doit son invention au physicien Alan Cormack et à l'ingénieur Godfrey Hounsfield tous deux récompensés par le prix Nobel de physiologie en 1979. Cet outil permet aujourd'hui de contrôler et diagnostiquer les patients et sa prescription est en constante augmentation malgré son caractère ionisant. A l'instar de la radiographie, le principe du scanner est d'obtenir une image d'un objet, mais reconstruit en **trois dimensions** à partir de ses projections. En effet, l'interaction entre les rayons X et la matière se fait selon des lois bien connues de la physique et permet donc une description du volume interne d'un objet.

Pour pouvoir reconstruire un tel volume, on emploie une méthode mathématique appelée **tomographie**. On appelle ainsi **tomodensitométrie (TDM)**, la tomographie par transmission de rayon X pour laquelle l'objet est exposé à des rayons X émis par une source externe. Les photons transmis sont détectés de l'autre côté de l'objet à l'aide d'une caméra, ce qui permet de rendre compte de leur **atténuation** dans l'objet, dont l'amplitude dépend des **constituants** de l'objet et de **l'énergie des photons X**.

Le principe de reconstruction habituellement utilisé avec les scanners à rayons X repose uniquement sur la mesure de **l'atténuation** des rayons X par l'objet que nous appellerons **TDM d'absorption classique**. Cependant, dès 1976, Alvarez et Macovski avancent la possibilité d'utiliser aussi l'information portée par **l'énergie** des rayons X détectés pour en déduire des informations sur la composition de l'objet. C'est cette exploitation de **l'énergie des photons** qui va donner naissance à la **TDM spectrale**.

En **TDM** d'absorption classique, on ne peut obtenir qu'une image d'atténuation de l'objet. De plus, si deux constituants présentent une atténuation similaire du faisceau de rayons X, ils ne pourront pas être différenciés dans l'image reconstruite. La **TDM** spectrale basée sur l'analyse d'images d'atténuation avec **différents spectres** d'énergie entraînant des contrastes différents entre différents constituants va ainsi permettre de les discriminer.

Dans les années 90, le développement de **détecteurs à pixels hybrides** pour la physique des particules va être exploité pour obtenir des détecteurs pouvant fonctionner en mode **comptage de photons seuillés** en énergie. Ces derniers vont ouvrir la porte au développement de la tomographie spectrale. Plus particulièrement, le Centre de Physique des Particules de Marseille (CPPM) qui avait participé au développement du détecteur de vertex à pixels hybrides de l'expérience ATLAS au CERN, a lancé sous l'impulsion de Pierre Delpierre le développement d'une lignée de détecteurs nommé **XPAD** (*X-ray Pixel with Adaptive Dynamic*) pour la cristallographie et le développement de la **TDM** spectrale. Ce projet a abouti en 2015 à la construction du prototype de micro-TDM spectrale PIXSCAN-FLI pour le Centre Européen de Recherche en Imagerie Médicale (CERIMED) financé par le Programme d'Investissements d'Avenir (PIA) France Life Imaging (FLI).

L'objectif de cette thèse est de développer des outils mathématiques pour le traitement des données issues de la **TDM** spectrale en exploitant les possibilités offertes par l'instrumentation développée au CPPM. Une première preuve de concept de la **TDM** spectrale a été établie en utilisant

des caméras XPAD avec une méthode appelée imagerie au **K-edge**. Néanmoins, de nouvelles approches plus robustes au bruit et plus spécifiques sont nécessaires pour rendre cette modalité émergente utilisable dans les contextes préclinique et clinique.

Le but est donc de développer des méthodes de reconstruction qui vont nous permettre de :

- distinguer les constituants spécifiques d'un objet imagé, comme les **marqueurs biologiques** (aussi appelés **agents de contraste**), en fonction de leurs signatures d'absorption d'énergie,
- garantir des reconstructions plus robustes au bruit que l'imagerie au K-edge, en particulier pour des caméras ne disposant que d'un seul seuil d'énergie réglable par pixel,
- réaliser des reconstructions plus rapidement.

Pour cela, les méthodes développées doivent s'appuyer sur des stratégies de détection optimales et doivent permettre de **quantifier, séparer et reconstruire** simultanément les constituants d'intérêt de l'objet.

Pour pouvoir répondre à cet objectif, il est nécessaire de modéliser toute la physique d'acquisition de la **TDM** spectrale dans un opérateur de transition de l'objet vers les mesures qui va permettre de traiter le problème de reconstruction comme un **problème inverse**. Les problèmes inverses mal posés, mais « simples » tels que ceux liés aux problèmes d'optimisation convexes, sont abondamment décrits dans la littérature scientifique et ont des fondements mathématiques et algorithmiques bien compris. Mais la reconstruction en **TDM** spectrale relève des problèmes inverses non linéaires se traduisant par des problèmes d'optimisation non convexes, qui sont nettement plus difficiles à traiter et pour lesquels les méthodes proposées dans la littérature donnent des garanties moins fortes. La question de l'application des résultats théoriques et pratiques à la **TDM** spectrale constitue l'enjeu méthodologique central de cette thèse.

Ainsi ce manuscrit s'articule de la manière suivante pour pouvoir répondre à la problématique annoncée :

- le premier chapitre aura pour but dans la première section de poser les bases du problème et d'expliquer les enjeux et la problématique de la **TDM** spectrale. La seconde section détaillera les principes physiques qui gravitent autour de la **TDM** spectrale et qui nous permettent de mieux aborder le problème. Un rappel des principes de base de la tomographie sera fait pour poser le contexte mathématique. Enfin, la troisième section parlera des différentes méthodes existantes en **TDM** d'absorption classique et la **TDM** spectrale. Ceci permettra de mieux situer notre problème dans un contexte global.
- Le deuxième chapitre est une introduction à l'optimisation. En effet, un problème inverse est posé pour modéliser la reconstruction d'images en **TDM** spectrale. La complexité de ce problème est double : d'une part c'est un problème de reconstruction en **TDM** d'absorption classique et d'une autre part c'est un problème de séparation de sources, qui consiste à reconstruire plusieurs signaux (les constituants) à partir de plusieurs observations. On introduira dans ce chapitre les différents outils mathématiques qui permettent d'aborder un tel problème. On présentera par la même occasion les algorithmes utilisés durant ce travail de thèse.
- Le troisième chapitre présentera dans une première partie les outils développés spécifiquement pour nos méthodes et adaptés au problème de la **TDM** spectrale. Nous présenterons ensuite les tests et les résultats obtenus en deux phases : d'abord l'étude menée sur des données simulées, puis l'étude menée sur des données réelles. Les tests sur données simulées sont divisés en trois parties : une première partie sur des données de **TDM** d'absorption classique, qui va permettre de valider nos méthodes sur un problème tomographique simple, une deuxième partie sur des données de **TDM** spectrale en deux dimensions, qui va permettre de se faire une idée sur le comportement de nos méthodes face à un problème de

**TDM** spectrale à dimension réduite et une troisième partie sur des données de **TDM** spectrale en trois dimensions.

Pour finir, on présentera les résultats obtenus sur deux jeux de données réelles acquis au CPPM : le premier sur un fantôme à microtubes Eppendorf conçu durant ce travail de thèse et contenant deux agents de contraste et le second sur un fantôme à tubes torsadés conçu pour une étude antérieure à ce travail de thèse et contenant également deux agents de contraste.

- Enfin, une conclusion des différents travaux menés au cours de ce travail de thèse viendra clore le manuscrit.



# Chapitre 1

## Introduction

*« Soit dans ce monde comme un étranger ou un voyageur ... »*

---

Al moustapha

### Sommaire

---

<b>1.1 Problématique Générale</b> . . . . .	<b>12</b>
1.1.1 Contexte et enjeux . . . . .	12
1.1.2 Problématique . . . . .	14
1.1.3 En bref . . . . .	20
<b>1.2 Physique d'acquisition</b> . . . . .	<b>21</b>
1.2.1 Physique de la TDM spectrale . . . . .	21
1.2.2 Le prototype PIXSCAN-FLI . . . . .	28
1.2.3 Tomographie . . . . .	35
1.2.4 En bref . . . . .	46
<b>1.3 Méthodes et état de l'art</b> . . . . .	<b>47</b>
1.3.1 TDM d'absorption classique . . . . .	47
1.3.2 TDM spectrale . . . . .	54
<b>1.4 Références</b> . . . . .	<b>63</b>

---

## Introduction

Ce chapitre pose les fondements de cette thèse. Il explique le contexte dans lequel le sujet se situe et introduit tous les outils nécessaires à la compréhension du problème de traitement de données en **tomodensitométrie spectrale**. Ce chapitre est divisé en trois sections.

La première section parle du contexte général et des notions qui seront développées tout au long de ce manuscrit. On introduira la problématique de la tomodensitométrie spectrale.

La deuxième section présente les principes physiques mis en jeu lors d'une acquisition tomographique. On parlera aussi du prototype PIXSCAN-FLI, le micro-tomodensitomètre spectral développé au CPPM grâce auquel des données réelles ont pu être acquises pour nos expériences.

Par ailleurs, elle introduit les notions mathématiques relatives à la tomographie (ou tomodensitométrie) classique. Ces notions permettent de situer dans un contexte global le problème de traitement de donnée en tomodensitométrie spectrale. Des définitions mathématiques plus détaillées seront données au chapitre 2.

La dernière section traitera l'état de l'art relatif à la tomodensitométrie spectrale ainsi qu'aux techniques algorithmiques existantes qui peuvent résoudre le problème mathématique sous-jacent à celle-ci, ceci dans le but de mettre en perspective le travail effectué pour cette thèse de doctorat.

## 1.1 Problématique Générale

### 1.1.1 Contexte et enjeux

La tomographie par rayons X, aussi appelée **tomodensitométrie (TDM)** ou *Computed Tomography (CT)*, est une technique d'imagerie non invasive largement utilisée dans des applications médicales pour contrôler et diagnostiquer des patients.

Le principe de base de la **TDM** est de reconstruire le volume interne d'un objet imagé à partir d'une série de mesures effectuées sous différentes incidences angulaires autour de l'objet. En **TDM**, l'objet comme illustré en figure 1.1, est illuminé par des rayons X, et un détecteur se situant à l'opposé de la source mesure la semi-transparence de l'objet à ces rayons X. Au cours des rotations, les **pixels** du détecteur enregistrent pour chaque angle d'incidence une projection des rayons X traversant l'objet. En fin d'acquisition chaque ligne du détecteur aura enregistré un ensemble de projections pour former ce que l'on appelle un **sinogramme**, dont un exemple est illustré en figure 1.1.

La semi-transparence des objets aux rayons X découle d'une propriété physique fondamentale :

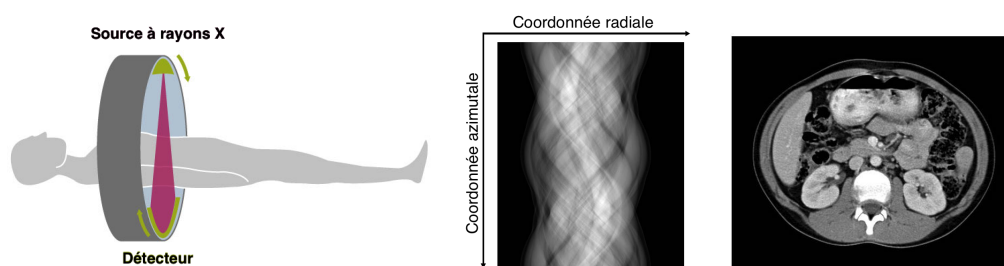


FIGURE 1.1 – Schéma du principe de la **TDM** : la figure de gauche illustre une acquisition lors d'un scanner à rayons X : on illumine à l'aide de rayons X la partie du corps cible du patient puis on mesure la semi-transparence de la cible à l'aide d'un détecteur pixelisé situé à l'opposé de la source. Chaque ligne de pixel du détecteur enregistre alors les rayons projetés. Au cours des rotations un sinogramme (figure du milieu) est obtenu. La figure de droite montre une coupe transversale du corps humain reconstruite à partir de ce sinogramme.

le principe d'**atténuation** (ou d'**absorption**). En effet, l'absorption des rayons X par la matière varie en fonction de la nature de la matière, en particulier en fonction du numéro atomique et de la

densité de la matière. Par exemple un os absorbera beaucoup plus de rayons X que des graisses. Des algorithmes de reconstruction sont ensuite utilisés pour estimer la distribution de l'atténuation des rayons X dans des coupes d'un objet (**tomogramme**) à partir de la mesure de leurs projections (**sinogramme**). On trouve dans la littérature de nombreux algorithmes, qui peuvent être **analytiques** ou **itératifs** et tenir compte plus ou moins fidèlement de la chaîne d'acquisition des mesures. Ils seront discutés plus en détail en sous-section 1.2.3. Un exemple d'une image reconstruite que l'on peut obtenir avec ces algorithmes est donnée en figure 1.1.

Cependant, les technologies actuelles en **TDM** comportent plusieurs limites :

1. du fait de la similarité de la composition, et donc des propriétés d'absorption, des différents tissus mous (muscles, graisses, vaisseaux sanguins, etc.), le contraste entre ces tissus est souvent insuffisant. En effet, en **TDM** standard, les images reconstruites ne sont pas spécifiques au type de tissu et donc différents types de tissus peuvent apparaître avec des valeurs de pixels similaires,
2. les doses de rayonnement ionisants peuvent être relativement élevées et leur réduction est un enjeu majeur de santé publique,
3. les valeurs des pixels des images reconstruites en **TDM** standard ne représentent qu'une approximation des **coefficients d'atténuation linéique**. En conséquence, elles ne sont pas quantitatives mais qualitatives ce qui ne donne pas d'information sur la concentration d'un composant sur une image.

La nature des détecteurs de rayons X utilisés dans le tomodensitomètre a également une grande influence sur les performances du **TDM**. En effet, les limites précédentes peuvent de plus être détériorées par le bruit présent dans les détecteurs à intégration de charges (**DIC**), comme les détecteurs **CCD** (*charge-coupled devices*) couplés à un matériau scintillant [AMELIO, 1975] utilisés dans les tomodensitomètres classiques. Ces **DIC** mesurent les signaux en intégrant toute l'énergie des photons provenant du faisceau de rayons X, perdant ainsi toute l'information dépendante de l'énergie des rayons X détectés. Or, comme nous le verrons en section 1.2.1, les coefficients d'atténuation linéique dépendent de l'énergie des photons incidents, de la composition chimique ainsi que de la masse volumique de l'objet. Par conséquent, le spectre de rayons X transmis contient des informations énergétiques sur différents types de tissus, ce qui permettrait de les distinguer entre eux. C'est pour cela qu'il est important de tenir compte de l'énergie.

L'avènement des détecteurs à comptage de photons (**DCP**) avec des capacités de discrimination énergétique a ouvert la voie à la **TDM** spectrale pour l'imagerie médicale [SHIKHALIEV, 2005]. Ces détecteurs comptent non seulement le nombre de photons du spectre à rayons X transmis (et non plus l'intégrale de leur énergie), mais ils ont en plus la spécificité de pouvoir compter ces photons qu'à partir d'un certain **seuil minimal** ou bien dans une fenêtre en énergie. Avec ceci, il devient possible de discriminer des composants d'intérêt comme des marqueurs biologiques disposant d'une signature en énergie spécifique très utilisés en imagerie médicale que l'on appelle **agents de contraste**. Ces agents de contraste peuvent être utilisés à différentes fins, parmi lesquelles marquer certains tissus mous ou organes spécifiques (foie, système urinaire, etc.). Ces systèmes ont le potentiel d'améliorer les limitations discutées plus haut et peuvent donc mener à de nouvelles applications cliniques [SHIKHALIEV et collab., 2005]. En effet, il y a plusieurs avantages à la **TDM spectrale** :

1. la possibilité d'imager et de localiser plusieurs composants d'intérêt simultanément (os, agents de contraste, etc.) ce qui ouvre la voie à l'imagerie en « couleur »,
2. l'obtention d'une imagerie quantitative donnant par exemple accès aux concentrations en agents de contraste dans le corps,
3. l'amélioration du rapport contraste/bruit, ce qui garantit des images d'une qualité supérieure aux images standards [OUAMARA, 2013].

On trouvera plus de détails sur les motivations de la **TDM** spectrale en sous section 1.3.2.

A l'heure actuelle le potentiel des systèmes **DCP** n'a pas encore été complètement exploité car les **DCP** sont arrivés que tardivement parmi les systèmes d'imagerie contrairement aux scanners conventionnels [FLOHR et collab., 2006; VAN ABBEMA et collab., 2012]. De ce fait, il n'y a pas autant d'étude sur ces systèmes que sur les approches de reconstruction tomographique en **TDM** standard. Les aspects algorithmiques de la **TDM** spectrale, tenant compte de l'énergie du faisceau de rayons polychromatique, n'ont été donc pris en compte que relativement tard et c'est sur ce point que porte le sujet de cette thèse.

### 1.1.2 Problématique

Dans cette sous-section nous allons poser la problématique de la thèse de manière formelle. Des définitions et informations complémentaires sur la physique de la **TDM** spectrale seront données en section 1.2.

Le principal objectif de ce travail est de développer des outils mathématiques permettant d'utiliser tout le potentiel des **DCP** et pour cela nous proposons ici des **algorithmes de reconstruction en 4D** (reconstruction d'objet 3D + séparation des composants d'intérêt) efficaces permettant de traiter les données issues d'une **TDM** spectrale. Ceci permet d'ouvrir de nouvelles perspectives pour distinguer les composants d'un objet en fonction de la signature d'absorption énergétique qui est spécifique à chaque composant (agents de contraste, os, etc.).

Afin de répondre aux objectifs fixés, des méthodes théoriques et algorithmiques doivent être développées. Ces méthodes répondront à plusieurs critères :

1. la **précision** : la modélisation du problème doit tenir compte de tous les paramètres qui peuvent intervenir lors d'une acquisition (profil du spectre de la source de rayons X, efficacité en énergie du détecteur de rayons X, etc.). Une modélisation précise du problème permettra de garantir une qualité élevée des résultats obtenus.
2. La **robustesse** : les méthodes doivent être robustes face aux différentes sources de bruit présentes lors de l'acquisition comme le bruit de mesure, assimilable à un bruit de Poisson, ou les imprécisions dues à la calibration approximative du détecteur.
3. La **rapidité et l'efficacité** : dans l'optique d'appliquer nos méthodes sur des données réelles, celles-ci doivent pouvoir fournir un résultat en un temps raisonnable, compatible avec le contexte pré-clinique ou clinique, en utilisant les ressources informatiques d'un serveur de calcul standard afin d'éviter des coûts d'infrastructure trop élevés.

#### 1.1.2.1 Ce que l'on attend de la **TDM** spectrale

Nous allons illustrer ici les avantages attendus de la **TDM** spectrale. On prend pour exemple le cylindre d'eau en figure 1.2 contenant des agents de contraste :

- trois inserts d'yttrium en solution à différentes concentrations,
- deux inserts d'argent en solution à différentes concentrations,
- un insert d'iode en solution.

Si nous reconstruisons l'objet en utilisant des méthodes en **TDM** standard, comme la **rétroprojection filtrée** qui sera expliquée en section 1.2.3.5, nous obtenons une image où le contraste entre les composants peut être très faible. En effet, cette méthode suppose la monochromaticité du rayonnement, ce qui est rarement le cas en pratique. Ceci résulte donc en une reconstruction équivalente à celle que l'on aurait obtenue si le rayonnement était monochromatique à une énergie de l'ordre de l'énergie moyenne du vrai spectre d'énergie délivré par un tube à rayons X. Il reste cependant difficile de définir mathématiquement et précisément :



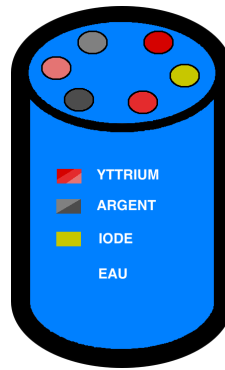


FIGURE 1.2 – Schéma d'un cylindre d'eau contenant différents agents de contraste à différentes concentrations.

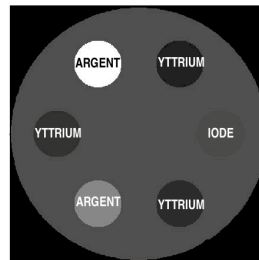


FIGURE 1.3 – Exemple type de reconstruction possible en TDM standard.

- l'énergie à laquelle la carte d'atténuation est reconstruite,
- la nature de la quantité reconstruite, même si celle-ci est homogène à une atténuation.

Par exemple, en figure 1.3, l'insert contenant l'iode peut être confondu avec l'eau car il n'y a pas assez de contraste pour le confirmer, donc nous ne pouvons pas identifier clairement ce composant sur l'image. De plus, même si le contraste des solutions d'yttrium et d'argent sont importants, on

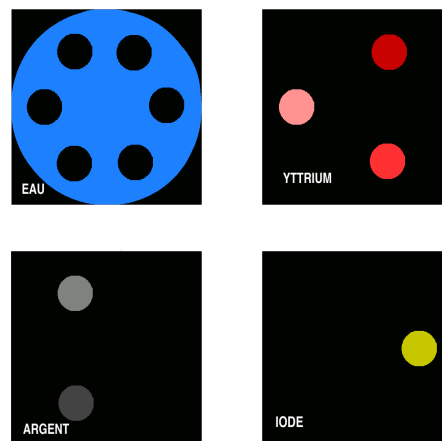


FIGURE 1.4 – Exemple type de reconstruction en TDM spectrale.

ne peut pas discriminer ces deux agents de contraste.

Nonobstant, en TDM spectrale, il est possible de **localiser** et d'**identifier** chaque composant de l'objet et d'estimer aussi la concentration réelle de chaque composant directement sur l'image. Comme on peut le voir sur l'exemple en figure 1.4, cette fois on distingue l'iode de l'eau et aussi l'argent de l'yttrium. De plus, on est capable de quantifier chaque concentration car les cartes de

l'argent et de l'yttrium ont des intensités de couleurs différentes.

Ce cas idéal permet de décrire les attentes et bénéfices de la **TDM** spectrale et ainsi de fixer les objectifs des méthodes que nous allons développer au cours de cette thèse. Nous allons voir dans les paragraphes suivants la modélisation théorique et la formulation du problème mathématique du traitement de données en **TDM** spectrale.

### 1.1.2.2 Modèle

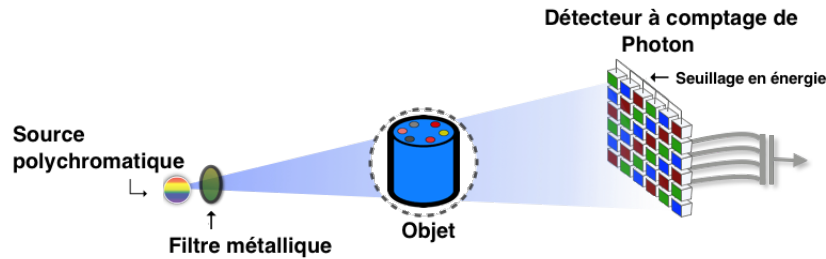


FIGURE 1.5 – Schéma d'acquisition en **TDM** : au milieu, l'objet imagé en rotation ; à droite, le détecteur, dans notre exemple il s'agit d'un détecteur à comptage de photon ; à gauche, la source de rayons X modulée par un filtre métallique.

Une acquisition tomodensitométrique est obtenue lorsqu'on illumine un objet avec une source de rayons X (voir figure 1.5). Un détecteur, qui se situe à l'opposé de la source par rapport à l'objet, mesure alors la semi-transparence de l'objet aux rayons X. Notons que l'objet peut être soit en rotation sur lui-même, soit fixe. Dans ce cas, c'est le couple source-détecteur qui est alors en rotation autour de l'objet. A chaque angle de vue, on obtient une **radiographie**, qui permet de déterminer une image projetée de la distribution des coefficients d'atténuation linéique en soustrayant son logarithme à celui de la radiographie obtenue sans l'objet. L'ensemble de ces projections forme ce que nous avons précédemment appelé le **sinogramme**.

La caractérisation de l'objet va dépendre de son **coefficient d'absorption** (ou **d'atténuation**), noté communément  $\mu$ , qui varie en fonction de l'énergie  $E$  de manière unique selon chaque composant de l'objet. Lorsque la source est **polychromatique** et l'objet **hétérogène** (*i.e.* constitué de plusieurs composants), on quantifie l'intensité du faisceau mesuré  $y_p$  dans chaque pixel  $p$  du détecteur avec la loi de Beer-Lambert

$$y_p = \int_{\mathbb{R}^+} I_0(E) e^{-\int_{\mathcal{C}_p} \mu(l,E) dl} dE \quad (1.1)$$

où  $I_0(E)$  est le spectre en énergie de la source et  $\mathcal{C}_p$  la ligne de visée (ou rayon) selon laquelle l'intégrale de  $\mu$  est calculée.

Comme on peut le voir en figure 1.5, le spectre d'énergie de la source peut être modifié de plusieurs manières. On peut changer la tension d'alimentation, ce qui change la forme du spectre et on peut aussi moduler ce dernier par un filtre métallique choisi par l'utilisateur placé entre la source et l'objet. Afin de tenir compte de ce paramètre, on notera  $f(E)$  la **réponse du filtre métallique**.

Par ailleurs, il y a aussi l'**efficacité du détecteur**, notée  $D(E)$ , qui dépend du détecteur utilisé et de la calibration effectuée. Cette efficacité se traduit par la probabilité qu'un photon d'une certaine énergie qui tape le détecteur soit détecté. Idéalement, on aimerait que  $D(E)$  prenne la valeur 1 tout le temps, mais ce n'est pas le cas en pratique.

Enfin, pour avoir l'acquisition **spectrale** que l'on souhaite (voir figure 1.6), il faut modéliser un paramètre important du détecteur : la configuration des **seuils** en énergie. Dans notre cas, chaque pixel de détection est muni d'un unique seuil, noté  $s$ , qui permet de choisir une énergie minimale

au-dessus de laquelle les photons seront comptés. C'est en modifiant ces différents paramètres d'une acquisition à une autre que l'on discrimine l'information pour mieux séparer les composants de l'objet. On indexe alors par  $m$  les différents paramètres et on note alors

$$F_m(E) = I_0(E) f_m(E) D_m(E) \mathbb{1}_{\{E > s_m\}} \quad (1.2)$$

où  $\mathbb{1}_{\{E > s_m\}}$  est la fonction indicatrice telle que

$$\mathbb{1}_{\{E > s_m\}} = \begin{cases} 1 & \text{si } E > s_m \\ 0 & \text{sinon.} \end{cases}$$

Dans le cas où chaque pixel dispose de plusieurs seuils ( $s_m^1$  et  $s_m^2$ ), l'indicatrice s'écrit

$$\mathbb{1}_{\{s_m^1 < E < s_m^2\}} = \begin{cases} 1 & \text{si } s_m^1 < E < s_m^2 \\ 0 & \text{sinon.} \end{cases}$$

$F_m(E)$  est la **fonction de pondération** rassemblant les variations en énergie induites par les différentes configurations, on peut alors modéliser une mesure tomographique spectrale  $y_{p,m}$  du  $p$ -ème pixel avec la  $m$ -ème configuration énergétique à partir de (1.1)

$$y_{p,m} = \int_{\mathbb{R}^+} F_m(E) e^{-\int_{\mathcal{L}_p} \mu(l,E) dl} dE. \quad (1.3)$$

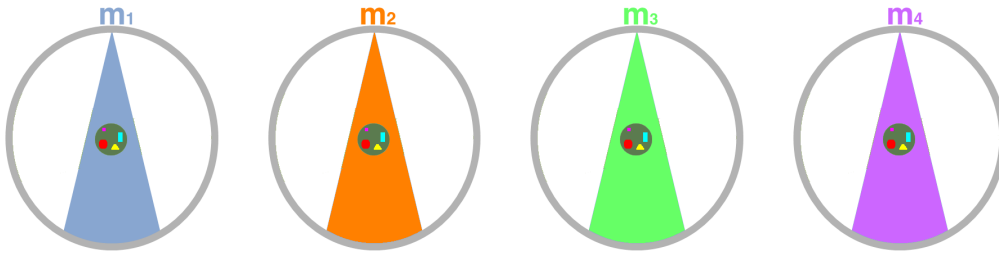


FIGURE 1.6 – Schéma d'une TDM spectrale : c'est un ensemble de plusieurs acquisitions d'un même objet. Les acquisitions se différencient entre elles par des paramétrisations différentes, notées  $\{m_1, m_2, m_3, m_4\}$ .

### 1.1.2.3 Modèle de l'objet

Afin de mieux séparer les composants, il est nécessaire de concevoir un modèle d'objet approprié. Les deux modèles de représentation d'objets les plus utilisés sont la décomposition par **interaction** et la décomposition en **éléments de base (DEB)**.

Le modèle **DEB** offre l'approximation la plus précise particulièrement pour l'imagerie biomédicale [CAI, 2013] où le nombre de composants d'un objet est en général limité. Le modèle que l'on propose ici est plus souple et peut être appliqué à plusieurs types de décomposition. Il est basé sur l'hypothèse que la dépendance en énergie des coefficients d'atténuation pour un objet donné est séparable et peut être représentée par plusieurs composants. En séparant alors la contribution des  $K$  composants dans l'objet, on peut réécrire le coefficient d'absorption

$$\mu(l, E) = \sum_{k=1}^K \mu^k(l, E) = \sum_{k=1}^K a^k(l) \sigma^k(E), \quad (1.4)$$

où  $a^k(l)$  est la **concentration du composant  $k$**  au point  $l$ , et  $\sigma^k(E)$  sa **section efficace d'interaction**. L'équation (1.3) devient alors :

$$y_{p,m} = \int_{\mathbb{R}^+} F_m(E) e^{-\sum_{k=1}^K \sigma^k(E) \int_{\mathcal{L}_p} a^k(l) dl} dE. \quad (1.5)$$

### 1.1.2.4 Discrétisation

Afin d'adapter notre problème à des algorithmes efficaces, il est nécessaire de discrétiser l'équation (1.5) pour avoir un modèle formulé matriciellement dans le but de faire apparaître des opérateurs linéaires.

On échantillonne l'énergie  $E$  en  $N$  valeurs et on discrétise l'intégrale en énergie de (1.5). De même, on échantillonne le volume 3D contenant l'objet en  $D$  voxels<sup>1</sup>, et on discrétise l'intégrale selon la ligne de visée  $\mathcal{C}_p$ . On notera alors  $S[p, d]$  la contribution du voxel  $d$  dans l'intégrale selon la ligne de visée  $\mathcal{C}_p$ .

Par ailleurs, remarquons que discrétisé ainsi, le modèle de l'objet (1.4) peut s'écrire sous forme matricielle

$$\mu = \mathbf{A}\Sigma \quad (1.6)$$

où  $\mathbf{A} \in \mathbb{R}^{D \times K}$  est une matrice représentant les **cartes spatiales des concentrations** inconnues des  $D$  voxels pour chacun des  $K$  matériaux et  $\Sigma \in \mathbb{R}^{K \times N}$  est le dictionnaire des **sections efficaces énergétiques** discrètes des  $K$  matériaux. Des exemples seront donnés en section 1.2.

L'équation (1.5) devient alors :

$$Y[p, m] = \sum_{n=1}^N F[n, m] e^{-\sum_{k=1}^K \sum_{d=1}^D S[p, d] \mathbf{A}[d, k] \Sigma[k, n]} \quad (1.7)$$

avec  $F[n, m]$  la forme discrétisée de la fonction de pondération  $F_m(E)$ . Ce qui donne sous forme matricielle :

$$\mathbf{Y} = e^{-\mathbf{S}\mathbf{A}\Sigma} \mathbf{F} \quad (1.8)$$

où

- $\mathbf{Y} \in \mathbb{R}^{P \times M}$  est la **matrice de données** qui contient dans chaque colonne la  $m$ -ième acquisition,
- $\mathbf{F} \in \mathbb{R}^{N \times M}$  est le **dictionnaire de filtrage en énergie** qui contient dans chaque colonne les facteurs des variations énergétiques induites par les configurations de la  $m$ -ième acquisition,
- $\mathbf{S} \in \mathbb{R}^{P \times D}$  est l'**opérateur de la transformée** à rayons X qui modélise la projection de l'objet sur le détecteur,
- $\mathbf{A} \in \mathbb{R}^{D \times K}$  est la **matrice des cartes des concentrations** que l'on souhaite reconstruire, elle contient dans chaque colonne la répartition du  $k$ -ième composant dans le volume,
- $\Sigma \in \mathbb{R}^{K \times N}$  est le **dictionnaire des sections efficaces** d'interaction contenant dans chaque ligne l'évolution de la probabilité d'interaction du  $k$ -ième composant.

On note que  $P$  est le nombre de mesures, qui est égal au nombre de pixels multiplié par le nombre d'angles de vue,  $M$  est le nombre d'acquisitions,  $N$  le nombre d'échantillons de l'énergie,  $D$  le nombre de voxels de l'objet et  $K$  le nombre de composants. Notons aussi que pour une matrice  $X$ , la fonction  $X \mapsto e^{-X}$  est appliquée à chaque élément de la matrice  $X$  telle que

$$e^{-X}[i, j] = e^{-X[i, j]}.$$

### 1.1.2.5 Problème mathématique

#### Formulation

On cherche à estimer la matrice  $\mathbf{A}$  - dont la  $k$ -ième colonne représente la carte de concentration du composant  $k$  - connaissant les paramètres du système  $\mathbf{S}$ ,  $\mathbf{F}$ , et les signatures spectrales  $\Sigma$ , ainsi

1. Le voxel est l'unité de base permettant de définir un objet 3D numérique. Le voxel est au volume ce que le pixel est à l'image.

que des versions bruitées des mesures parfaites établies à l'équation (1.8) que l'on notera encore  $Y$ .

Pour cela on cherchera à trouver  $\hat{\mathbf{A}}$  de sorte que l'écart entre les données et le modèle, évalué en  $\hat{\mathbf{A}}$ , soit le plus petit possible. Mathématiquement cela revient à minimiser une fonction que l'on nommera  $G(\mathbf{A})$

$$\hat{\mathbf{A}} = \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\operatorname{argmin}} G(\mathbf{A}). \quad (1.9)$$

La fonction  $G(\mathbf{A})$  est plus communément appelée la fonction **d'attache aux données**, qui tient compte du type de bruit de mesure. Lorsque le bruit est gaussien ce terme s'écrit

$$G(\mathbf{A}) = \|Y - e^{-\mathbf{SA}\Sigma} \mathbf{F}\|_{\text{fro}}^2 = \sum_{p,m} (\mathbf{Y}_{p,m} - \{e^{-\mathbf{SA}\Sigma} \mathbf{F}\}_{p,m})^2. \quad (1.10)$$

Dans le cas où le bruit supposé est un bruit de Poisson, le terme d'attaches aux données s'écrit

$$G(\mathbf{A}) = \sum_{p,m} \{e^{-\mathbf{SA}\Sigma} \mathbf{F}\}_{p,m} - \mathbf{Y}_{p,m} \log(\{e^{-\mathbf{SA}\Sigma} \mathbf{F}\}_{p,m}). \quad (1.11)$$

Nous verrons plus tard que c'est le cas Poisson qui nous intéresse. Notre modèle (1.8) est non linéaire et la fonction d'attache aux données  $G(\mathbf{A})$  est une fonction **non convexe**, mais elle demeure néanmoins **différentiable**<sup>2</sup>. La conséquence de ceci est qu'on ne peut affirmer avec certitude qu'un minimiseur de (1.9) est nécessairement global.

Par ailleurs, pour augmenter la qualité de reconstruction, il est d'usage de tenir compte d'information *a priori* dont on dispose sur l'objet. On cherche à reconstruire des cartes de concentration, c'est une quantité physique qui est positive par définition donc une première contrainte de notre modèle est la **positivité**. Notons aussi qu'on peut utiliser la **parcimonie** spatiale comme deuxième contrainte à notre modèle. En effet, les colonnes de la matrice des cartes de concentrations  $\mathbf{A}$  sont parcimonieuses lorsque le composant à reconstruire est peu présent dans l'objet comme un agent de contraste par exemple.

On modélise ces contraintes par une fonction  $R(\mathbf{A})$ , appelée fonction de **régularisation**, convexe mais non différentiable faite d'un terme contraignant la positivité sur tous les éléments ( $\mathbf{A}[d, k] \geq 0, \forall (d, k)$ ) et d'un terme de régularisation pour chaque carte  $\mathbf{A}[:, k] \stackrel{\text{def}}{=} \mathbf{A}_k$  :

$$R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A}) + \sum_{k=1}^K \lambda_k R_k(\mathbf{A}_k), \quad (1.12)$$

où

$$\chi_{\{\geq 0\}}(\mathbf{A}) = \begin{cases} 0 & \text{si } \mathbf{A}[d, k] \geq 0, \forall (d, k) \\ +\infty & \text{sinon.} \end{cases}$$

Les régularisations spatiales  $R_k(\mathbf{A}_k)$  sont choisies en fonction des composants de l'objet, par exemple un terme de parcimonie. On exprime alors le problème comme suit

$$\hat{\mathbf{A}} = \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\operatorname{argmin}} \{J(\mathbf{A}) = G(\mathbf{A}) + R(\mathbf{A})\} \quad (1.13)$$

### Difficultés

La reconstruction est donc obtenue en résolvant un **problème inverse** de la forme :

$$\underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\operatorname{argmin}} \{G(\mathbf{A}) + R(\mathbf{A})\}, \quad (1.14)$$

où  $G(\mathbf{A})$  est une fonction différentiable, mais non convexe, qui représente la mesure de l'écart entre les données et le modèle (terme d'attache aux données) et  $R(\mathbf{A})$  est une fonction convexe, mais

2. Les notions de convexité et de différentiabilité sont définies au chapitre 2.

non différentiable, qui modélise les contraintes imposées à l'objet que l'on souhaite reconstruire. Comme nous l'avons dit précédemment, la première difficulté est la non convexité du problème qui rend difficile la recherche d'un point optimal.

Ensuite, on peut voir la reconstruction en **TDM** spectrale comme la superposition de deux sous-problèmes :

- (i) un problème de reconstruction tomographique pour lequel l'état de l'art est très fourni pour le cas standard (par opposition au cas spectral),
- (ii) un problème de séparation de sources, car il s'agit ici de distinguer différents composants d'un seul objet.

Le tout forme un problème inverse, mal posé et non convexe pour lequel, à ce jour, l'état de l'art reste peu fourni (voir section 1.3).

Enfin, résoudre (1.13) revient à manipuler des matrices de très grandes tailles notamment l'opérateur de projection  $S \in \mathbb{R}^{P \times D}$ , où en pratique le produit  $P \times D$  est de l'ordre de  $10^{16}$ . Cette contrainte nous pousse donc à chercher des méthodes itératives efficaces qui peuvent résoudre le problème (1.13) en un temps raisonnable et en appliquant le moins possible l'opérateur de projection  $S$  et son adjoint, l'opérateur de rétroprojection  $S^t$ .

### 1.1.3 En bref

Notre approche consiste à exploiter la richesse de l'information fournit par les **DCP** afin de prendre en compte la dépendance d'énergie des composants d'un objet et le premier problème qui se pose est celui de la **modélisation**. Notre modèle est obtenu à partir de la loi de Beer-Lambert discrétisée qui prend en compte l'énergie ainsi que des configurations énergétiques propres aux **DCP**. On se base ensuite sur une décomposition de l'objet en fonction des composants qui le constituent et on utilise la connaissance physique des signatures énergétiques de l'absorption de chacun de ces composants.

Notre modèle tient donc compte du caractère **polychromatique** de la source car c'est le principe inhérent à la **TDM** spectrale rendue possible grâce à plusieurs mesures discriminantes en énergie. Cela a pour but de **localiser**, **d'isoler** et **quantifier** les composants au sein de l'objet imagé.

Ce qui constitue la difficulté de notre travail est la **non-linéarité** du modèle de la **TDM** spectrale et la **non convexité** du problème de traitement de données qui en découle. Viennent s'ajouter à cela la séparation des composants et la grande quantité de données à traiter qui font du problème de reconstruction en **TDM** spectrale un **problème inverse, mal posé** et de **grande dimension**. La figure 1.7 résume notre problématique.

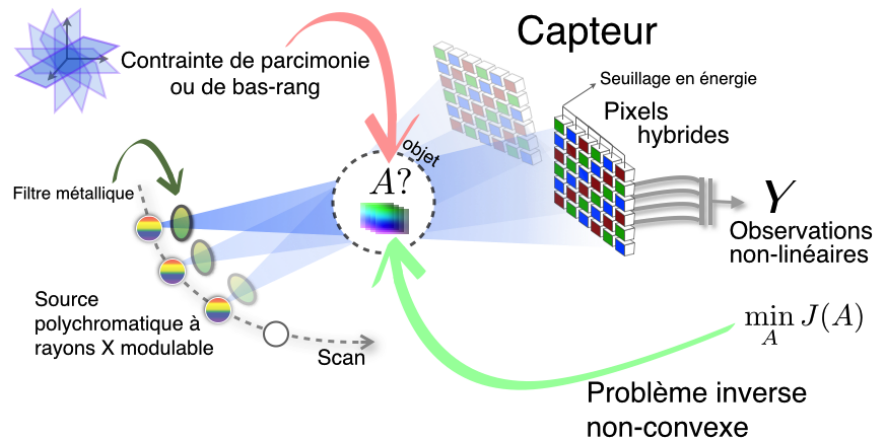


FIGURE 1.7 – Schéma général du problème en TDM spectrale.

Un objet composé de plusieurs composants est imagé à l’aide d’une source dont le spectre d’énergie est modulé par un filtre métallique placé entre la source et l’objet. Un DCP disposant de seuils en énergie compte les photons X qui ont traversé l’objet, permettant ainsi d’obtenir des données spectrales. Ces observations sont non-linéaires en fonction de la matrice inconnue  $\mathbf{A}$ . À partir d’un modèle incorporant des contraintes spécifiques à l’objet, nous résolvons un problème **non-convexe** qui consiste à simultanément **reconstruire** en 3D et **séparer** les composants de l’objet.

## 1.2 Physique d’acquisition

Afin de mieux modéliser les données issues d’une TDM spectrale, il faut comprendre les phénomènes physiques à l’origine de l’interaction des rayons X avec la matière. Le but de cette section est de définir les composants qui nous mènent au modèle (1.8) suivant

$$\mathbf{Y} = e^{-\mathbf{S}\mathbf{A}\Sigma}\mathbf{F}.$$

Nous allons dans un premier temps introduire les notions physiques qui permettent de comprendre le principe d’interaction entre les rayons X et la matière. Ainsi nous pourrions expliquer d’où vient notre modèle et ce que contiennent les matrices  $\mathbf{A}$  et  $\Sigma$ .

Nous donnerons ensuite une description du tomodensitomètre PIXSCAN-FLI avec lequel les données expérimentales de cette thèse ont été acquises. On expliquera par la même occasion ce que contient la matrice  $\mathbf{F}$ .

Nous parlerons enfin du concept même de tomographie et des outils mathématiques qui interviennent en TDM. Nous pourrions ainsi poser les bases permettant de mieux comprendre ce que contient la matrice  $\mathbf{S}$  du modèle (1.8).

### 1.2.1 Physique de la TDM spectrale

Dans cette sous-section, nous allons rappeler les propriétés physiques qui interviennent dans le processus d’imagerie par rayon X. Nous verrons donc dans un premier temps les propriétés essentielles des rayons X et comment ceux-ci sont générés. Puis nous verrons les différents types d’interaction entre les rayons X et la matière. Enfin nous parlerons de la détection de ces rayons X.

#### 1.2.1.1 Les rayons X et leur production

L’imagerie médicale utilise majoritairement l’interaction de **rayons X** avec la matière afin de produire des images. Les rayons X sont des ondes électromagnétiques (EM) **ionisantes** capables de par leur forte énergie (voir figure 1.8) de traverser la matière. Ces rayonnements électromagnétiques peuvent être exprimés en termes d’énergie et de longueur d’onde. On s’intéressera en général à l’énergie des photons exprimée en électron-volts (eV). Les rayons X peuvent être pro-

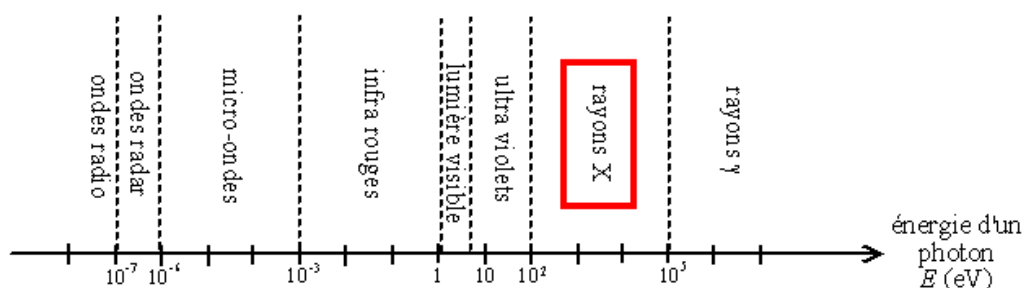


FIGURE 1.8 – Classement des ondes électromagnétiques par rapport à l'énergie des photons. Les rayons X utilisés en imagerie médicale comptent parmi les rayonnements les plus énergétiques ( $\geq 100$  eV).

duits selon deux processus distincts :

- par **fluorescence**, ou émission secondaire de rayon X. Elle peut être provoquée par des rayons X ou par un bombardement d'électrons qui causent l'ionisation d'un électron d'une couche interne de l'atome cible, ce qui entraîne le réarrangement des couches électroniques générant l'émission de rayons X d'énergies définies.
- par **rayonnement de freinage**, aussi appelé **bremstrahlung**. Ces deux phénomènes ont lieu dans l'**anode** (qui peut être en tungstène ou en molybdène par exemple) d'un tube sous vide (voir schéma en figure 1.9) dans lequel se trouve également un filament métallique, la **cathode**. Lorsque la cathode est chauffée par un courant, celle-ci émet des électrons. En appliquant une haute tension ( $\sim 10^2$  kV) entre l'anode et la cathode, les électrons sont accélérés vers l'anode. La majeure partie de l'énergie cinétique des électrons arrivant dans l'anode est alors convertie en chaleur, le reste générant un spectre de rayons X par rayonnement de freinage dont l'énergie maximale est égale à la tension d'accélération des électrons (*i.e.* une tension de par exemple 50 kV correspondant à une énergie maximale de 50 keV).

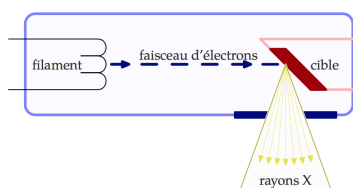


FIGURE 1.9 – Schéma d'un tube à rayons X.

### 1.2.1.2 L'interaction photon-matière

Les rayons X interagissent avec la matière suivant 3 processus distincts :

1. soit le photon est absorbé par la matière en transférant toute son énergie au milieu par effet **photoélectrique**,
2. soit le photon est diffusé de manière cohérente ou incohérente par effet **Rayleigh** ou **Compton**, respectivement,
3. soit le photon est converti en une paire électron-positon par effet **création de paires**.

Ces effets sont basés sur divers processus physiques dont la probabilité de se produire dépend de l'énergie du photon incident et de la densité ou alors du numéro atomique du milieu absorbant ou diffusant.



La prédominance des différents effets en fonction du milieu et de l'énergie du photon incident est illustrée sur la figure 1.10. Il faut préciser qu'aux énergies mises en jeu en **TDM**, le processus de production de paires, qui nécessite un seuil d'énergie de 1022 keV, peut être ignoré. Les énergies employées en TDM clinique et pré-clinique dépendent de l'examen et varient entre 10 et 100 keV.

### Effet photoélectrique

L'effet **photoélectrique** est le phénomène physique dominant pour atténuer les rayons X de basse ou moyenne énergie. Il correspond à l'absorption d'un photon incident par l'atome cible et à l'éjection d'un électron d'une couche atomique interne que l'on appelle **photoélectron**. Lorsque le photoélectron est éjecté, un électron d'une couche atomique supérieure va immédiatement combler le vide laissé et l'énergie de cette transition électromagnétique va être émise soit par un photon de fluorescence caractéristique, soit par un électron Auger. Un schéma illustre ce phénomène en figure 1.11.

Le photon n'arrache un électron d'une couche que si son énergie est supérieure à l'énergie de liai-

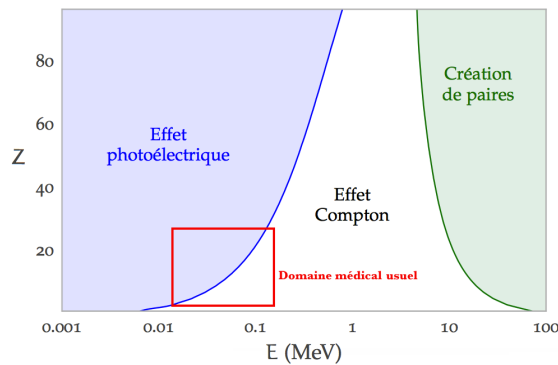


FIGURE 1.10 – Répartition de la prédominance de l'effet photoélectrique, de la diffusion Compton et de la création de paire en fonction de l'énergie exprimée en MeV et du numéro atomique du milieu d'interaction.

son de l'électron sur sa couche. En dessous de cette valeur, la probabilité, appelée **section efficace**, d'arracher un électron de cette couche est nulle. La figure 1.12 illustre l'évolution de la section efficace en fonction de l'énergie. On y voit des cassures, appelées **edges**, aux énergies correspondant aux énergies de liaison des couches atomiques de l'atome considéré (dans l'ordre d'éloignement : K, L, puis M).

Ce qui justifie l'apparition de ces **cassures** est la différence de densité électronique des diffé-

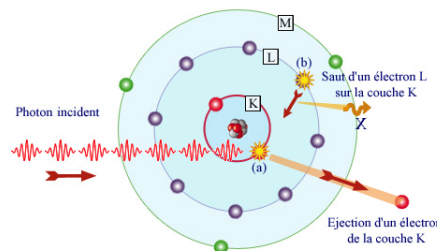


FIGURE 1.11 – Schéma illustrant l'effet photoélectrique et le réarrangement subséquent des couches atomiques : (a) le photon arrache un électron lié d'un atome (dans cet exemple il s'agit d'un électron de la couche K), puis (b) un électron d'une couche atomique plus externe vient occuper la lacune laissée par l'électron ionisé, ce faisant générant un rayon X de fluorescence caractéristique.

rentes couches atomiques. En effet, lorsque l'énergie du photon est supérieure à l'énergie de liaison d'une couche atomique donnée, une nouvelle voie d'ionisation est accessible avec une densité électronique plus élevée, on observe alors une augmentation brutale du coefficient d'atténuation photoélectrique à l'origine d'une **cassure**. Ces **cassures** ont toutes des énergies différentes qui sont

caractéristiques de l'atome cible considéré et c'est pour cette raison qu'elles serviront de **signature spectrale** du composant cible.

### Lien avec notre problème

Le dictionnaire  $\Sigma$  du modèle (1.8) est constitué de ces signatures, elles sont déterminées en fonction des composants de l'objet. Si on prend par exemple un objet constitué d'eau, de calcium et d'un agent de contraste tel que l'iode, alors on mettra dans chaque ligne de  $\Sigma$  la section efficace respective de ces composants sur la gamme d'énergie  $\{e_1, e_2, \dots, e_N\}$

$$\Sigma = \begin{pmatrix} \sigma_{eau}(e_1) & \cdots & \sigma_{eau}(e_N) \\ \sigma_{Ca}(e_1) & \cdots & \sigma_{Ca}(e_N) \\ \sigma_I(e_1) & \cdots & \sigma_I(e_N) \end{pmatrix}. \quad (1.15)$$

Le dictionnaire  $\Sigma$  contiendra autant de signatures qu'il y a de composants dans l'objet. Par ailleurs, il s'avère que les signatures d'éléments comme l'eau, le plastique ou les tissus mous sont très similaires. On pourra alors les regrouper sous une même signature par approximation.

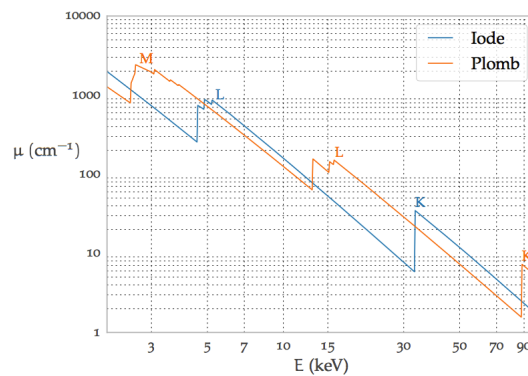


FIGURE 1.12 – Illustration des **cassures** caractéristiques (ou **edges**) du coefficient d'atténuation linéique en fonction de l'énergie de photon incident. Lorsque l'énergie d'un photon augmente, il devient capable d'ioniser un électron d'une couche plus profonde, la section efficace augmente brusquement. Sur la figure, on peut observer les **cassures** caractéristiques des couches K, L et M pour le plomb et l'iode. On note que la **cassure** caractéristique de la couche K est nette contrairement à celles des couches L et M moins prononcées.

### Effet Compton

Lors d'une interaction **Compton**, le photon incident transmet son impulsion à un électron et un photon diffusé (voir figure 1.13). C'est un phénomène de diffusion au cours duquel la trajectoire du photon diffusé fait un angle  $\theta \in [0, 180^\circ]$  avec la trajectoire du photon incident, qui transmet une partie de son impulsion à l'électron avec lequel il a interagi. Cet électron est diffusé avec un angle  $\Phi \in [0, 90^\circ]$  par rapport à la trajectoire du photon incident.

Dans le contexte de la **TDM**, la détection de photons diffusés dégrade l'image car la loi de Beer-Lambert sur laquelle est basée la modélisation de la formation des projections utilisées pour reconstruire l'image ne tient compte que des photons dont les trajectoires entre la source et le détecteur sont rectilignes. Notons qu'en clinique, une solution consiste à placer des grilles de collimation devant les détecteurs pour éliminer les photons ayant subi une trop grande déviation.

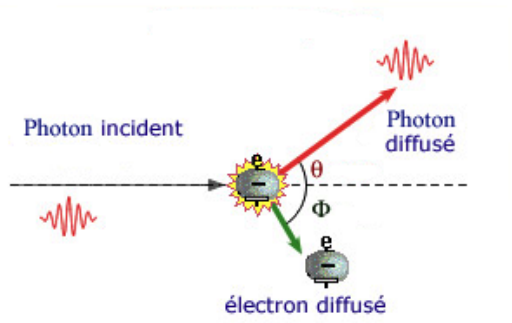


FIGURE 1.13 – Illustration de l'effet Compton.

### Diffusion Rayleigh

Au cours d'une diffusion cohérente ou diffusion de **Rayleigh**, le photon incident est diffusé par le nuage électronique de l'atome, et celui-ci n'est ni excité ni ionisé et l'angle de diffusion est relativement faible. L'énergie du photon diffusé est la même que celle du photon incident. Cet effet survient plus fréquemment pour des photons incidents de basses énergies ( $\leq 40$  keV) et dans des matériaux de  $Z$  élevé.

#### 1.2.1.3 Atténuation des photons

Les différents mécanismes décrits précédemment sont responsables de la perte d'énergie des photons dans la matière au niveau atomique. En imagerie médicale, on considère des faisceaux de photons et donc l'interaction d'un grand nombre de photons avec une épaisseur de matériau.

Pour décrire quantitativement l'atténuation d'un faisceau, on peut imaginer une intensité  $I_0$  traversant un matériau homogène d'une épaisseur  $\Delta x$ . La diminution d'intensité obtenue en sortie du matériau dépendra évidemment de  $I_0$  et de l'épaisseur  $\Delta x$  mais aussi de l'efficacité avec laquelle le matériau traversé agit sur les photons, ce que l'on fait apparaître au travers d'un paramètre noté  $\mu$  et appelé **coefficient d'atténuation** ou **d'absorption** du matériau concerné. La figure 1.14 illustre cette atténuation.

$\mu$  peut être exprimé en  $cm^{-1}$ , on l'appelle alors le **coefficient d'atténuation linéique (CAL)**. On utilise aussi le **coefficient d'atténuation massique (CAM)** en divisant  $\mu$  par la masse volumique, ce coefficient s'exprime alors en  $cm^2/g$ . L'avantage d'une telle expression est qu'elle ne dépend que de la quantité de matière qui se présente sur le chemin du faisceau.

Dans le cas de l'eau par exemple, les états "glace", "eau liquide" et "vapeur" auront des **CAL** différents mais des **CAM** identiques. C'est pour cela que la plupart des tables numériques sont présentées en fonction des **CAM**, mais en définitive ce sont les **CAL** qui importent en radiologie car ils décrivent mieux l'interaction des photons avec une épaisseur de tissus.

La relation qui permet de mesurer l'intensité du faisceau en sortie de l'objet imagé est décrite par la loi de **Beer-Lambert**, elle s'écrit comme suit :

$$I = I_0 e^{-\mu \Delta x}, \quad (1.16)$$

avec  $I_0$  l'intensité initiale du faisceau,  $\mu$  le **CAL** et  $\Delta x$  l'épaisseur du matériau. Lorsque l'objet imagé n'est pas homogène (c'est-à-dire lorsqu'il est composé de plusieurs matériaux), le **CAL** devient dépendant de l'endroit de l'objet où les photons interagissent. On considère un chemin  $\mathcal{C}$  parcouru par le faisceau de photons dans l'objet :

$$I = I_0 e^{-\int_{\mathcal{C}} \mu(x) dx}. \quad (1.17)$$

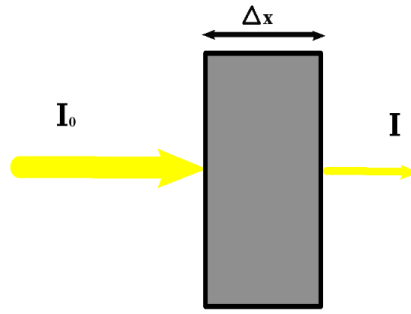


FIGURE 1.14 – Illustration de l’atténuation d’un faisceau de rayons X. L’intensité du faisceau en sortie  $I$  dépendra de l’intensité initiale  $I_0$ , de l’épaisseur  $\Delta x$  ainsi que du type de matériau caractérisé par son **coefficient d’atténuation**  $\mu$ .

En considérant le caractère polychromatique du faisceau de photons incident et de la dépendance du **CAL** à l’énergie du faisceau on obtient le nombre total de photons sortant de l’objet :

$$I = \int_0^{+\infty} I_0(E) e^{-\int_{\ell} \mu(x,E) dx} dE. \quad (1.18)$$

### Lien avec notre problème

Afin de retomber sur l’équation (1.1) vue précédemment, il suffit de tenir compte de l’intensité mesurée pour un pixel  $p$  :

$$y_p = \int_{\mathbb{R}^+} I_0(E) e^{-\int_{\ell_p} \mu(l,E) dl} dE.$$

On note par ailleurs que, comme cité en 1.1.2.3, le coefficient d’atténuation  $\mu$  peut être décomposé soit par somme de coefficients d’atténuation propre à chaque interaction citée en haut, soit par éléments de bases.

On rappelle que ce modèle s’écrit  $\mu(l,E) = \sum_{k=1}^K a_k(l) \sigma_k(E)$ , ce qui équivaut sous forme matricielle à  $\mu = \mathbf{A}\Sigma$ . Un exemple de matrice  $\Sigma$  est donné en 1.15 ci dessus.

La matrice  $\mathbf{A}$  est la matrice des concentrations que l’on souhaite reconstruire, elle modélise dans chacune de ses colonnes la répartition et la concentration d’un composant dans l’espace de l’objet. Si l’on reprend l’exemple 1.15, on peut illustrer cela comme suit :

$$\mathbf{A} = \begin{pmatrix} a_{eau}(x_1) & a_{Ca}(x_1) & a_1(x_1) \\ \vdots & \vdots & \vdots \\ a_{eau}(x_D) & a_{Ca}(x_D) & a_1(x_D) \end{pmatrix} \quad (1.19)$$

où  $\{x_1, x_2, \dots, x_D\}$  est l’ensemble de voxels sur lequel l’objet est échantillonné.  $a_k(x_d)$  est la concentration du composant  $k$  au voxel  $d$ , qui est nulle s’il n’y a pas de présence du composant  $k$  dans ce voxel.

On remarque alors que les colonnes de la matrice  $\mathbf{A}$  sont **parcimonieuses** pour des composants rares (tels que les agents de contraste), c’est-à-dire qu’il n’y a que peu de voxels  $k$  contenant une concentration  $a_k$  non nulle. Ceci est principalement dû au fait qu’un composant  $k$  ne peut pas être présent partout dans l’objet. Cette information *a priori* nous servira plus tard comme contrainte sur l’objet afin d’augmenter la qualité de la reconstruction.

La reconstruction de  $\mathbf{A}$  permettra alors d’identifier et localiser les composants et aussi d’estimer la concentration de ceux-ci au sein de chaque voxel. Ceci est schématisé en figure 1.15.

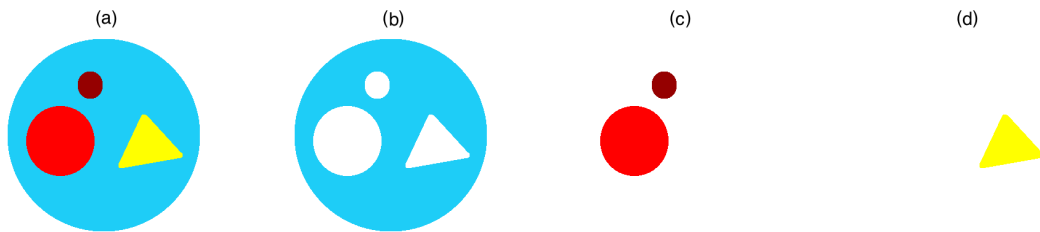


FIGURE 1.15 – Schéma d'une reconstruction de  $\mathbf{A}$  : (a) l'objet dans sa globalité, il est composé de 3 composants (eau, calcium et iode), si l'on vectorise les éléments de cette image, on obtient les colonnes de la matrice  $\mathbf{A}$  - (b) est la composante eau, (c) est la composante calcium et (d) la composante iode. Les couleurs ne sont données qu'à titre indicatif sur la concentration et sur la nature du composant. Par exemple, le calcium est présent à deux endroits avec des concentrations différentes indiquées par une intensité de couleur différente.

#### 1.2.1.4 Détection

Les interactions des rayonnements ionisants avec la matière conditionnent le choix de la technologie de détections de ces radiations. En calorimétrie, la détection consiste en la mesure de la quantité (totale dans l'idéal) d'énergie déposée ou perdue par les rayonnements lors de leur passage dans la zone « pixélisée » sensible du détecteur.

Généralement, les détecteurs de rayonnements ionisants convertissent l'énergie déposée en un signal numérique. Selon la technologie de détection, ils peuvent être classés en deux catégories : les détecteurs à conversion **indirecte** et les détecteurs à conversion **directe**.

- Un détecteur à conversion indirecte convertit d'abord les photons X en photons de longueurs d'onde du domaine visible dans un scintillateur. En imagerie médicale, cette conversion se fait en général à l'aide de **cristaux inorganiques**. On utilise ensuite un photo-détecteur ou un écran à couplage de charge pour détecter et mesurer le nombre de photons visibles et ainsi remonter à l'énergie de photon X détecté pour les **DCP** ou à l'intensité du faisceau de rayons X pour les **DIC**.
- Un détecteur à conversion directe convertit les photons directement en charges électriques en utilisant des matériaux de détection **semi-conducteurs**. Les matériaux de détection directe couramment utilisés sont le silicium (Si) et le tellure de cadmium (CdTe) [JOHNSON et collab., 2007]. En effet, dans ces détecteurs, le passage des rayons provoque l'ionisation de la matière et donc la création d'un grand nombre de paires électron-trou. Sous l'effet d'un champ électrique les charges vont se déplacer dans le détecteur. Le nombre de charges détectées est alors proportionnel à la quantité d'énergie déposée par le rayonnement.

En général, les détecteurs à conversion indirecte ont une large gamme d'énergie de détection et une bonne efficacité de détection. En revanche, les détecteurs à conversion directe présentent des avantages pour la sélection de l'énergie photonique, la résolution spatiale et parfois le rapport signal sur bruit (SNR). On trouvera dans [IWANCZYK et collab., 2009] une brève description des performances des différents types de détecteurs à rayons X.

Comme nous l'avons déjà mentionné, les détecteurs de rayons X peuvent également être classés en détecteurs à intégration de charges (**DIC**) et en détecteurs à comptage de photons (**DCP**) selon le mode de fonctionnement de l'électronique de détection utilisée.

Un **DIC** accumule l'énergie totale déposée à l'intérieur du détecteur pendant un temps de pose (ou temps d'exposition) et transmet l'information à la fin. Il mesure donc l'énergie totale des rayons X transmis ou en d'autres termes l'intensité du faisceau de rayons X.

En revanche, un **DCP** recueille les charges générées à l'arrivée de chaque photon et émet le nombre total d'impulsions à la fin. Le signal de sortie correspond au nombre de photons arrivés pendant le temps d'exposition.

Les détecteurs à intégration de charge sont souvent des détecteurs à conversion indirecte, tandis que les détecteurs à comptage de photons sont souvent des détecteurs à conversion directe.

### Lien avec notre problème

Le Centre de Physique des Particules de Marseille a conçu et développé plusieurs générations de détecteurs à comptage de photons avec une capacité d'acquisition dynamique importante (plus de 300 images/s) et la possibilité de définir un seuil d'énergie pour chaque pixel. Ceux-ci ont été intégrés dans 3 prototypes de micro-tomodensitomètres avec lesquels la preuve de concept de la **TDM** spectrale *in vivo* a été réalisée avec succès [HAMONET, 2016; KRONLAND-MARTINET, 2015], parmi lesquels le dernier-né, le prototype PIXSCAN-FLI (voir 1.2.2), est destiné à la recherche pré-clinique. Ce dernier nous permettra d'acquérir des données réelles avec lesquelles nous testerons les algorithmes développés dans cette thèse.

#### 1.2.1.5 Conclusion

Cette sous-section traite et définit les éléments physiques qui interviennent dans le processus d'acquisition d'une **TDM** spectrale. Nous avons expliqué les différentes interactions possibles entre les rayons X et la matière et comment cela va nous aider à mieux modéliser le problème de traitement de données. En effet, nous avons illustré ce que les matrices **A** et  **$\Sigma$**  (**cartes des concentrations** et **dictionnaire des sections efficaces**) de notre modèle (1.8) peuvent contenir.

Dans la suite nous allons présenter le tomodensitomètre PIXSCAN-FLI sur lequel les expériences de cette thèse ont été menées. Nous expliquerons par la même occasion comment le **dictionnaire de filtrage en énergie F** est construit.

## 1.2.2 Le prototype PIXSCAN-FLI

### 1.2.2.1 Introduction

Dans cette sous-section, nous allons décrire le prototype de micro-CT appelé **PIXSCAN-FLI**. C'est un tomodensitomètre pour le petit animal développé au CPPM par l'équipe imXgam dans le cadre d'un financement du Programme d'Investissement d'Avenir (PIA) *France Life Imaging* (FLI). Cette sous-section est utile à la compréhension des données réelles que l'on étudiera dans le chapitre 3. On décrira les différents constituants et les spécificités du prototype PIXSCAN-FLI, parmi lesquels les détecteurs à **pixels hybrides** et particulièrement la caméra XPAD3 installée dans le prototype PIXSCAN-FLI. Nous préciserons aussi pourquoi le prototype PIXSCAN-FLI est un micro-tomodensitomètre adapté à la TDM spectrale et ce que contient la matrice **F** de notre modèle.

### 1.2.2.2 Description

Le prototype PIXSCAN-FLI est un micro-tomodensitomètre complet installé dans une boîte de 1.01 m de hauteur, 1.39 m de longueur et une profondeur de 0.99 m. Cette boîte est constituée d'acier inoxydable et de plomb d'une épaisseur variant entre 3 et 7 mm pour un poids total inférieur à 1200 kg. Elle est ainsi conçue pour contenir les rayons X émis par la source. Une photographie de la boîte se trouve en figure 1.16.

La boîte comporte deux portes permettant à l'utilisateur d'accéder au prototype PIXSCAN-FLI ainsi qu'une chicane permettant le passage de câbles et la ventilation. Le contrôle des portes et les acquisitions tomodensitométriques sont assurés par un logiciel asservi à une boucle de sécurité





FIGURE 1.16 – Photographie de la boîte où est installé le prototype PIXSCAN-FLI.

interlock informatique développé au CPPM garantissant des conditions de sécurité maximales satisfaisant la réglementation en vigueur [DUPONT, 2014].

Le prototype est composé de 3 blocs : le **bloc moteur**, le **bloc source** et le **bloc détecteur**. Comme on peut le voir sur la photographie en figure 1.17, le bloc moteur est situé entre la source et le détecteur. Nous allons dans la suite décrire chacun de ces blocs.

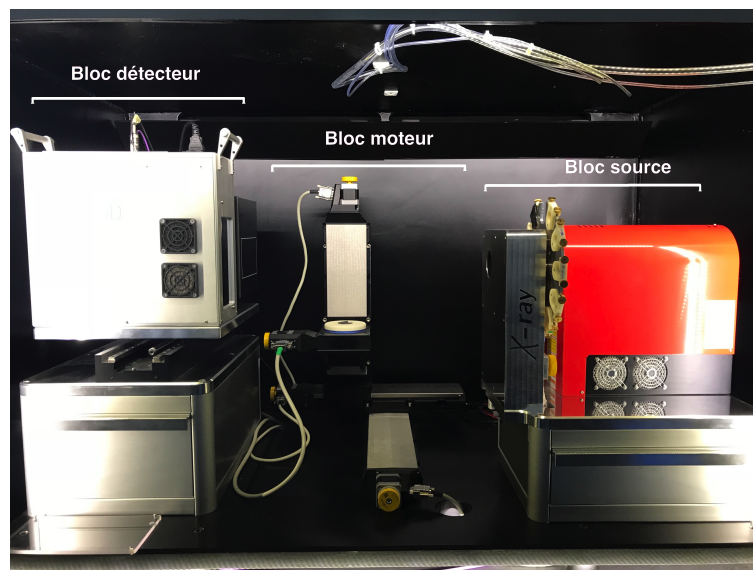


FIGURE 1.17 – Photographie du prototype PIXSCAN-FLI : à gauche, le bloc détecteur où est installée la caméra XPAD3, au milieu, le bloc moteur sur lequel on fixe l'objet ou l'animal à étudier et à droite, le bloc source.

### 1.2.2.3 Le bloc moteur

Dans le prototype PIXSCAN-FLI, le bloc moteur sur lequel se place l'objet ou l'animal se situe entre la source et le détecteur. Afin de modifier l'angle des projections, on place l'objet sur une platine de rotation motorisée permettant de contrôler la position angulaire avec une précision de 0.03 degré (voir figure 1.18). Celle-ci est elle-même positionnée sur un support de translation selon trois directions de l'espace afin de pouvoir ajuster correctement la position de l'objet. Les moteurs assurant la translation permettent de contrôler la position dans les trois directions avec une précision variant entre 5 et 8  $\mu m$ .

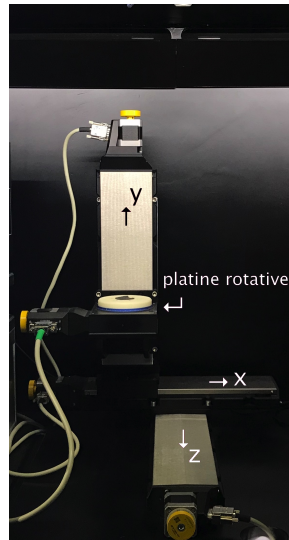


FIGURE 1.18 – Photographie du bloc moteur. La platine de rotation peut effectuer une course de 360 degrés avec une précision de 0.03 degré.

#### 1.2.2.4 Le bloc source

Le bloc source se situe à droite dans le prototype PIXSCAN-FLI. Il dispose d'une roue à filtres qui permet de moduler le spectre de la source de rayons X. Une photographie du bloc se trouve en figure 1.19. La source est un tube à anode Hamamatsu Microfocus<sup>3</sup> L12161 – 07 (Hamamatsu



FIGURE 1.19 – Photographie du bloc source.

City, Japon). La tube à rayons X comprend une anode en tungstène et permet de travailler avec une tension allant jusqu'à 150 kVp, ce qui permet de faire des acquisitions à haute énergie. La gamme d'énergie opérationnelle s'étale entre 40 et 150 keV. Le spectre de la source comporte deux groupe de raies :

- un groupe de raies à basses énergies : transitions de la couches M vers la couche L à 8.4, 9.7, 9.9 et 11.3 keV,
- un groupe de raies à hautes énergies : transitions de la couche L vers la couche K à 59.32 et 57.98 keV et de la couche M vers la couche K à 66,95 et 67.24 keV.

3. Site officiel : <https://www.hamamatsu.com/>



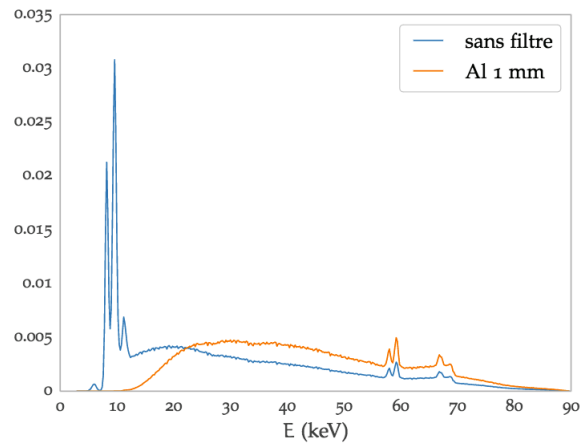


FIGURE 1.20 – Spectres d’énergie normalisés d’un tube à anode de tungstène pour une tension de 90 kV avec et sans filtre de 1 mm d’aluminium. En ordonnée nous avons l’intensité du faisceau (normalisé à l’intégrale) et en abscisse l’énergie en keV.

La figure 1.20 présente un spectre normalisé<sup>4</sup> de la source W à 90 kV sans filtre et avec un filtre d’aluminium d’un millimètre d’épaisseur.

Il est important de connaître la réponse du tube que l’on utilise car cette information sera directe-

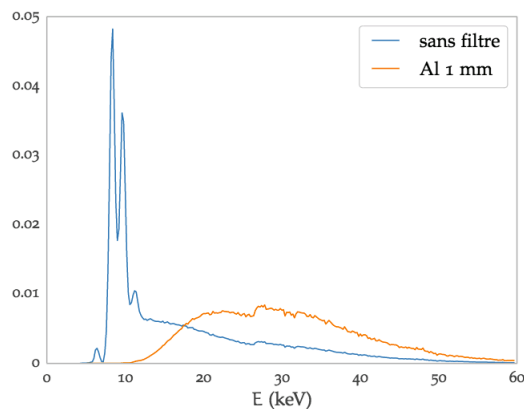


FIGURE 1.21 – Spectres d’énergie normalisés d’un tube à anode de tungstène pour une tension de 60 kV avec et sans filtre de 1 mm d’aluminium. En ordonnée nous avons l’intensité du faisceau (normalisé à l’intégrale) et en abscisse l’énergie en keV.

ment injectée dans le modèle (1.8), plus particulièrement dans la matrice **F**. Les différents facteurs qui influencent le spectre d’énergie délivré par le tube à rayons X sont :

- la tension appliquée entre l’anode et la cathode du tube<sup>5</sup>,
- le courant circulant dans la cathode (filament),
- le filtre placé devant la source.

L’illustration en figure 1.21 montre comment le spectre dépend de la tension et du filtre. En effet, on remarque que la diminution de la tension fait disparaître le deuxième groupe de raies tandis que le filtre d’aluminium atténue le premier groupe de raies.

Comme on peut le voir en figure 1.19, devant la source se trouve une roue à filtres motorisée. Dans chaque emplacement de la roue se trouve un filtre et le changement de filtre est piloté par l’interface de contrôle-commande du prototype PIXSCAN-FLI. Les filtres se présentent sous la forme de disques métalliques à insérer dans les emplacements de la roue. Nous disposons de plusieurs

4. Le spectre est normalisé dans le sens où chaque bin en énergie est divisé par l’intégrale totale du spectre.

5. Voir le paragraphe 1.2.1.

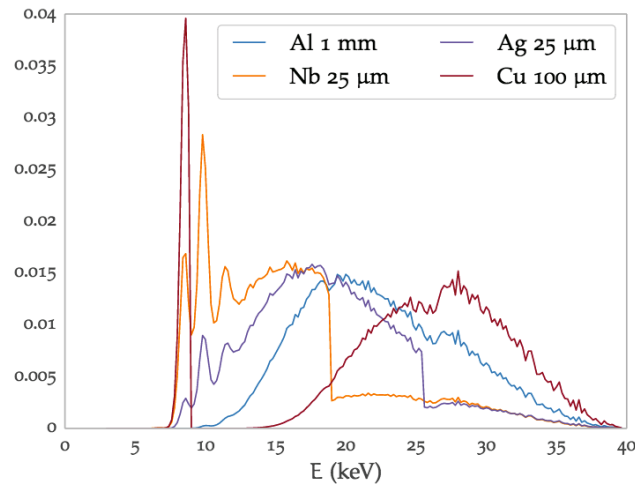


FIGURE 1.22 – Spectres d'énergie normalisés d'un tube à anode de tungstène pour une tension de 40 kV avec différents filtres d'aluminium, d'argent, de niobium et de cuivre. Le choix de filtres nous permet de créer des spectres d'énergie très différents, ce qui sera nécessaire pour identifier et séparer des composants différents.

disques de filtre d'aluminium, de niobium, d'argent, de cuivre et de molybdène de différentes épaisseurs que nous pouvons combiner entre eux pour obtenir des filtres combinés. Nous avons également la possibilité de créer des filtres liquides, comme par exemple une solution d'iode ou de nitrate d'argent. Les filtres modulent le spectre d'énergie de manière caractéristique et le choix des filtres nous permet de créer des spectres d'énergie très différents (illustration en figure 1.22), nous pouvons ainsi mieux discriminer l'information d'une acquisition à l'autre, comme nous l'avons illustré en figure 1.6.

Le dernier paramètre que l'on peut faire varier avec le prototype PIXSCAN-FLI est au niveau du détecteur, nous allons exposer ce point dans les paragraphes suivants.

### 1.2.2.5 Le bloc détecteur

Les détecteurs à **pixels hybrides** ont vu le jour pour la première fois en 1994 dans le domaine de la physique des hautes énergies, qui observe des collisions entre particules à hautes énergies, et plus particulièrement les trajectoires des particules engendrées par ces collisions. Les premiers détecteurs électroniques utilisés en trajectographie étaient basés sur des chambres proportionnelles multifilaires, puis plus tard des détecteurs à bandes de semiconducteur, qui permettant de détecter le passage d'une particule croisant un fil ou une bande de semiconducteur. L'association de plusieurs plans de détection parallèles avec des fils ou des bandes de semiconducteurs orientés dans des directions différentes permettaient de déterminer les points de passage des particules, mais posaient cependant des difficultés dues à l'ambiguïté de détection en présence de plusieurs particules détectées en même temps.

Pour pallier ce problème, les détecteurs à pixels où chaque pixel peut être associé sans ambiguïté au passage d'une particule ont été développés. Le moyen d'effectuer cette manœuvre est d'associer une électronique de lecture indépendante à chaque pixel du détecteur [DELPIERRE, 2014]. Deux schémas illustrent le principe des détecteurs à pixels en figure 1.23. Le principal défi de ce genre de détecteur est de miniaturiser l'électronique de lecture associée au pixel.

C'est ce qui est réalisé avec les pixels CMOS ou les caméras CCD, qui équipent nos appareils numériques, pour lesquels cette électronique de lecture est intégrée aux capteurs de lumière, mais dont la sensibilité n'est pas suffisante pour détecter des rayons X. Une autre alternative est **d'hybrider** l'électronique de lecture au capteur avec une micro-bille de soudure assurant à la fois le

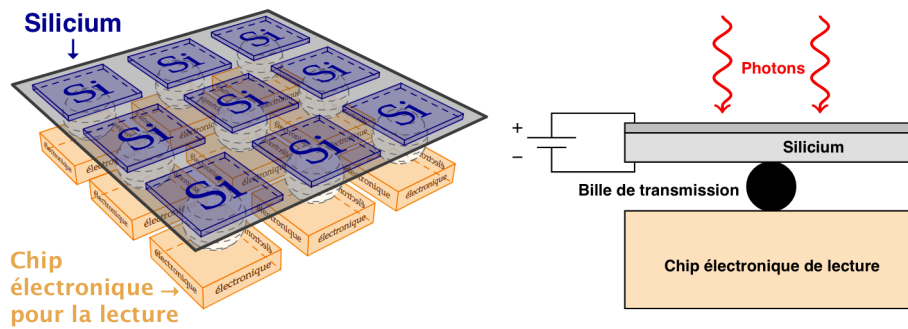


FIGURE 1.23 – Schémas de détecteur à pixels hybrides dans lesquels une électronique de lecture est associée à un pixel à l'aide d'une micro bille de soudure : à gauche, une vue en perspective, à droite, le schéma en coupe d'un pixel.

contact électrique et la rigidité mécanique de l'ensemble, ce qui permet de choisir le type de capteur et son épaisseur afin d'optimiser l'efficacité de détection de rayons X.

En mode « comptage de photons », le détecteur traite séparément chaque interaction. Chaque photon dépose une charge **proportionnelle à son énergie**. La charge est alors collectée et comparée à **un seuil**<sup>6</sup> préalablement fixé par l'utilisateur pour chaque pixel. On ne prend alors en compte que les interactions déposant une charge supérieure au seuil fixé. Ainsi, chaque pixel compte le nombre de photons détectés et non plus le nombre total de charges déposées par le faisceau de rayons X.

De plus, l'utilisation d'un seuil permet de s'affranchir totalement du bruit électronique aussi appelé bruit de **courant noir** du détecteur. La translation des pixels hybrides de la physique des particules vers les applications transverses s'est rapidement effectuée. En effet, les avantages des détecteurs à pixels hybrides ont suscité l'intérêt, particulièrement dans le domaine médical et pour la cristallographie principalement pour les raisons suivantes :

1. l'absence de bruit de courant noir, ce qui permet de travailler avec peu de signal et donc à bas flux, c'est-à-dire à faible dose,
2. chaque photon compté a le même poids dans la mesure, quelle que soit son énergie, ce qui permet un meilleur contraste par rapport aux **DIC** pour lesquels les photons de haute énergie ont un poids plus élevé que les photons de basse énergie dans la mesure,
3. l'existence d'un seuil d'énergie réglable et le comptage individuel de chaque photon permet d'étudier le spectre d'énergie des photons détectés dans des fenêtres d'énergie, ce qui ouvre la voie à la tomographie spectrale.

Le détecteur du prototype PIXSCAN-FLI est une caméra à pixels hybrides XPAD3, un détecteur à comptage de photons commercialisé par la société imXPAD S.A.S, qui est une startup du CPPM<sup>7</sup>. Cette caméra est issue des travaux de l'expérience du LEP DELPHI, puis de l'expérience du LHC ATLAS au CERN. La caméra installée sur le prototype PIXSCAN-FLI (voir photographie figure 1.24) est basée sur la troisième génération du circuit intégré de lecture ou **chip** XPAD3.2.

La caméra XPAD3 est constituée de 8 modules de 7 chips assemblés en tuiles inclinées avec un recouvrement pour ne pas avoir d'espaces morts entre les barrettes. Chacun de ces chips est composé 120 pixels en hauteur par 80 pixels en largeur. Chaque pixel mesure  $130 \mu\text{m} \times 130 \mu\text{m}$  et dispose d'un seuil d'énergie réglable indépendamment des autres pixels. Ceci donne une surface totale de  $120 \text{ cm} \times 75 \text{ cm}$ , ce qui permet de faire des images de souris corps entier (1.24) [PORTAL, 2018].

6. Remarquons aussi qu'il existe des détecteurs sur lesquels plusieurs seuils peuvent être fixés pour chaque pixel.

7. imXPAD S.A.S. a été active de mai 2010 à décembre 2016, ses activités ont été reprises par la société Cegitek Innovation S.A.S. (<https://www.cegitek.com>) sous la marque RebirX.

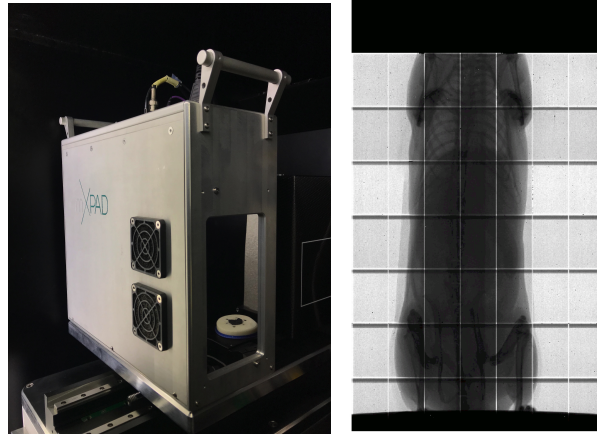


FIGURE 1.24 – A gauche, une photographie de la caméra à comptage de photons XPAD3. A droite, radiographie d’une souris vivante (la caméra XPAD3 comprend ici seulement 7 barrettes assemblées en tuiles au lieu de 8).

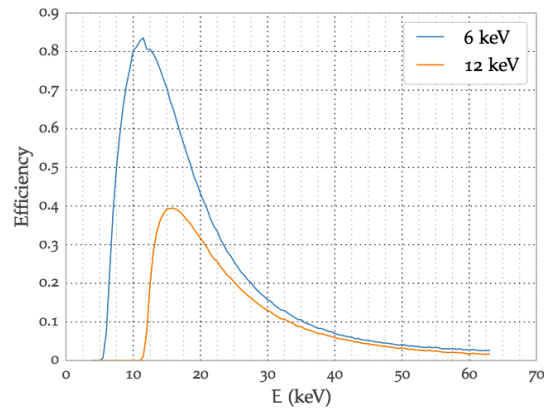


FIGURE 1.25 – Efficacité d’un pixel pour deux seuils d’énergie différents à 6 et 12 keV. On remarque que le silicium est très rapidement peu efficace lorsque l’énergie augmente, notamment au dessus de 25 keV où l’efficacité est inférieure à 20 %.

### Efficacité du détecteur

Dans le modèle (1.8), un facteur indispensable à connaître est **l’efficacité des pixels du détecteur**  $D(E)$ . Cette efficacité est calculée à partir des coefficients d’absorption théorique du silicium pour une épaisseur de  $500 \mu m$ . Cependant l’efficacité de comptage est un phénomène appelé « partage de charges » qui est dû à la réduction apparente de la surface sensible des pixels [MATHIESON et collab., 2002]. Une méthode [TAKOUKAM TALLA, 2011] a été développée pour simuler ce phénomène et calculer une estimation de l’efficacité. Le détail de cette méthode est décrit au chapitre 4 de [DUPONT, 2014].

La valeur de l’efficacité est le rapport entre le nombre de photons comptés par le détecteur et le nombre de photons incidents sur la surface du pixel. L’efficacité est relative à chaque pixel et dépend donc du seuil attribué à celui-ci et de l’énergie du photons. La figure 1.25 illustre l’efficacité d’un pixel pour deux seuils d’énergie différents à 6 et 12 keV. L’efficacité du détecteur peut influencer la forme du spectre mesuré comme on le voit en figure 1.26, c’est pour cette raison qu’il est important d’en tenir compte dans le modèle.

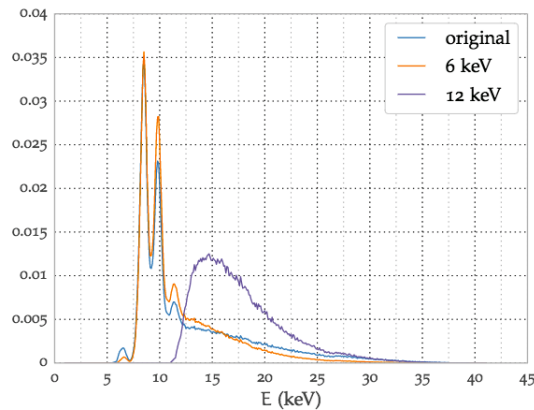


FIGURE 1.26 – Spectres d'énergie d'un tube à anode de tungstène pour une tension de 45 kV sans filtre et sa pondération par l'efficacité de détection d'un pixel pour deux seuils d'énergie différents à 6 et 12 keV. On remarque que les raies de basses énergies sont très atténuées pour une courbe d'efficacité correspondant à un seuil d'énergie à 12 keV.

### 1.2.2.6 Conclusion

#### Lien avec notre problème

C'est à partir de toutes ces informations que nous construisons **le dictionnaire de filtrage en énergie**  $F$  qui contient l'intensité initiale de la source  $I_0(E)$ , la réponse du filtre métallique  $f(E)$  et l'efficacité des pixels du détecteur  $D(E)$ . Ces paramètres changent d'une acquisition à l'autre, ce qui se traduit par un changement des valeurs d'une colonne à l'autre de la matrice  $F$ . Nous avons déjà expliqué au paragraphe 1.1.2.2 que nous disposions de deux leviers pour pouvoir faire de la **tomographie spectrale** :

- le premier levier est l'utilisation des seuils d'énergie que permet la caméra XPAD3,
- le deuxième levier se trouve du côté de la source de rayons X en amont de l'objet : c'est en multipliant les mesures d'un objet éclairé sous différents spectres que nous pourrions obtenir l'information spectrale (illustration en figure 1.22).

Nous avons désormais introduit toutes les notions de physiques permettant la compréhension du modèle d'acquisition (1.8). Nous allons voir maintenant le dernier élément qui nous permet d'expliquer le modèle complet : **la tomographie**.

## 1.2.3 Tomographie

### 1.2.3.1 Introduction

On peut décomposer la reconstruction **tomographique** d'un objet en deux étapes :

- élaborer un modèle direct décrivant suffisamment fidèlement les phénomènes physiques tels qu'ils sont mesurés. Ces phénomènes ont été en partie décrits précédemment dans la sous-section 1.2,
- déterminer un modèle de reconstruction servant à retrouver l'objet initial étudié, et c'est le but de cette sous-section.

Pour cela, on fait appel à des outils mathématiques qui décrivent la **projection** de l'objet à reconstruire déduite du passage du faisceau de rayons X à travers l'objet. C'est à partir de ces projections que l'on peut reconstruire l'objet en effectuant l'opération duale de la projection : la **rétroprojection**. Nous allons définir dans les paragraphes qui suivent ces deux outils fondamentaux ainsi que les algorithmes de reconstruction qui en découlent. Nous pourrions ainsi poser les bases pour mieux comprendre le contenu de la matrice  $S$  du modèle (1.8).

### 1.2.3.2 Projection

Lorsqu'un faisceau de rayons X passe à travers un objet, on peut en déduire des **projections** d'une propriété de l'objet, en l'occurrence du **coefficient d'atténuation** des rayons X dans le cas de la tomodensitométrie. Un exemple générique de cela est illustré en figure 1.27.

La géométrie de ces faisceaux incidents définit alors le type de projections :

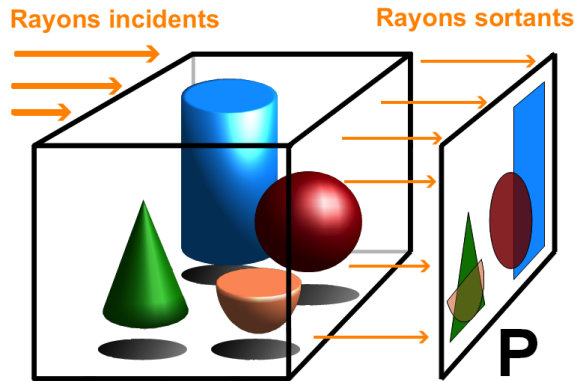


FIGURE 1.27 – Exemple de projection d'un objet. L'objet imagé ici est un ensemble de figures géométriques en trois dimensions (3D). On projette l'objet le long de lignes de projection ou rayons qui sont parallèles dans cet exemple. Lorsqu'on projette l'objet on obtient la figure en deux dimensions (2D) P à droite sur laquelle se superposent des figures délimitées par un rectangle pour la projection du cylindre, un triangle pour la projection du cône de révolution et un (demi-)cercle pour la projection de la (demi-)boule. Cette figure est donc la projection parallèle en 2D de l'objet en 3D sous une direction donnée.

- **parallèle** d'une coupe 2D ou d'un volume 3D,
- **divergente** ou en **éventail** (*fan beam* en anglais) d'une coupe 2D,
- **conique** d'un volume 3D.

Différentes géométries de faisceau de rayons X sont illustrées en figure 1.28. Les faisceaux paral-

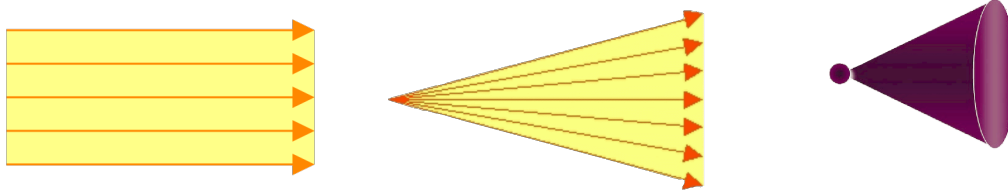


FIGURE 1.28 – Illustration de différentes géométries de faisceau, à gauche, parallèle, au centre, en éventail et, à droite, conique.

lèles correspondent aux faisceaux générés par une source théoriquement infiniment éloignée de l'objet tandis que les faisceaux coniques correspondent aux faisceaux que peut générer un tube à rayons X ordinaire. Les géométries divergentes en 2D ou 3D ont tendance à « magnifier » les proportions de l'objet projeté sur le détecteur, comme illustré en figure 1.29. Nous verrons au paragraphe 1.2.3.5 que ceci peut avoir un impact sur la reconstruction des images et doit être pris en compte dans les modèles de reconstruction.

Comme nous l'avons précédemment cité, en **TDM** les images de transmission de l'objet acquises sous différentes incidences permettent d'obtenir un **sinogramme**, dont un exemple est donné pour rappel en figures 1.1 et 1.31.

Le problème de reconstruction en **TDM** consiste à estimer la distribution du **CAL** de l'objet à

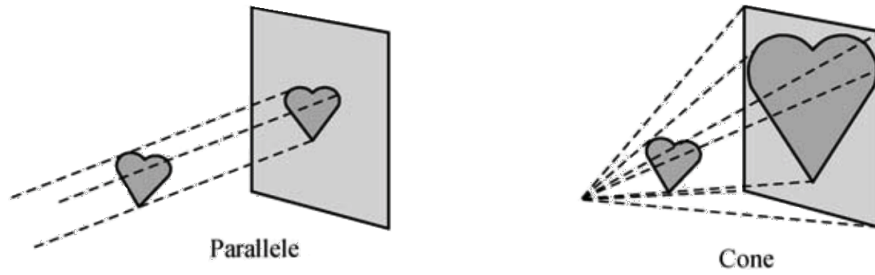


FIGURE 1.29 – Illustration de l'influence de la géométrie du faisceau sur la magnification des objets projetés sur le détecteur.

partir du sinogramme. Dans le cas de la **TDM** d'absorption classique, la source de rayons X est supposée monochromatique, ainsi tous les rayons X émis par la source ont la même énergie. On peut alors représenter le CAL d'une coupe 2D d'un objet par une fonction  $f(x, y)$  qu'il s'agit de reconstruire. L'intensité  $I_0$  du tube à rayons X atténué par l'objet le long d'un rayon L peut être représentée selon la formule (1.17) que l'on a vue précédemment :

$$I = I_0 e^{-\int_L f(\mathbf{r}) d\mathbf{r}}, \quad (1.20)$$

avec  $\mathbf{r} = (x, y)$ . En prenant alors le logarithme de l'atténuation relative, on obtient une valeur de l'intégrale de ligne de la fonction

$$-\log\left(\frac{I}{I_0}\right) = \int_L f(\mathbf{r}) d\mathbf{r}. \quad (1.21)$$

Mathématiquement, la relation entre les projections et le **CAL** de l'objet peut être exprimée en 2D par la **transformée de Radon** que nous allons définir dans les paragraphes suivants.

Le problème mathématique en **tomographie d'absorption classique** consiste à **reconstruire** une fonction dont on ne mesure que des intégrales selon des lignes. Il convient toutefois de noter que dans le cas de la TDM, l'équation (1.21) représente le cas monochromatique idéal et non réaliste, bien que cette approximation soit tout de même utilisée. D'une certaine manière, la **TDM spectrale**, qui exploite la polychromaticité du faisceau de rayons X, va permettre de prendre compte des phénomènes tels que le durcissement du faisceau et l'efficacité du détecteur.

### 1.2.3.3 Transformée de Radon

La **transformée de Radon**, qui doit son nom au mathématicien Johann Radon [RADON, 1917], est la formulation mathématique d'une projection tomographique. Il établit la possibilité de reconstituer une fonction réelle à deux variables assimilable à une image à l'aide de la totalité de ses projections selon des droites concourantes.

Soit une fonction  $f \in L^2(\mathbb{R}^2)$ <sup>8</sup>, on définit de manière générale l'opérateur de projection de la fonction  $f$  par

$$\begin{aligned} P & : L^2(\mathbb{R}^2) & \rightarrow & L^2([0, \pi] \times \mathbb{R}) \\ f & & \mapsto & P_f. \end{aligned}$$

On définit alors  $P_f \forall \theta \in [0, \pi[$  et  $\forall t \in \mathbb{R}$ , la **projection** de  $f$  le long d'un rayon noté  $\Delta_{t,\theta}$  par :

$$\begin{aligned} P_f & : [0, \pi] \times \mathbb{R} & \rightarrow & \mathbb{R} \\ (\theta, t) & & \mapsto & P_f(\theta, t) = p_\theta(t) \end{aligned}$$

8.  $L^2(\mathbb{R}^2)$  est l'espace des fonctions de carré intégrable à valeurs réelles, plus de détails sont donnés en chapitre 2.



où

$$p_{\theta}(t) = \int_{\Delta_{t,\theta}} f(\mathbf{r}) \, d\mathbf{r} = \int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) \delta(x \cos \theta + y \sin \theta - t) \, dx dy \quad (1.22)$$

avec  $\delta$  est la fonction de **Dirac** dont l'intégrale est normée à 1 défini par

$$\delta(x) = \begin{cases} +\infty & \text{si } x = 0, \\ 0 & \text{sinon.} \end{cases}$$

C'est la fonction de **distribution ponctuelle**<sup>9</sup> d'un événement en l'abscisse 0. La transformée de Radon « cartographie »  $f$  le long des directions définies par  $\theta$  perpendiculaires aux directions de projection. Un exemple est en figure 1.30, on peut voir une projection comme une représentation à une dimension (1D) de l'objet en fonction d'une coordonnée radiale qui court le long d'une direction définie par  $\theta$ .

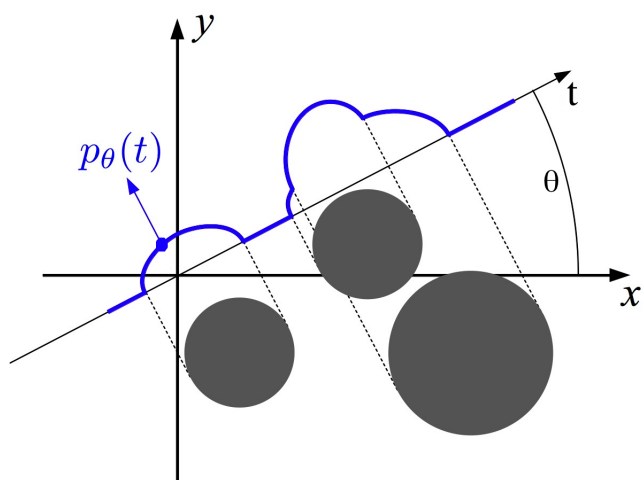


FIGURE 1.30 – Exemple : transformée de Radon selon l'axe paramétré par  $t$  et l'angle  $\theta$ . La courbe bleue obtenue représente l'ensemble des projections  $p_{\theta}(t)$  en fonction de la coordonnée radiale  $t$  le long de cet axe.

**Lemme 1.2.3.1: projection**

On peut réécrire la projection (1.22) de la manière suivante :

$$p_{\theta}(t) = \int_{\mathbb{R}} f(t \cos \theta - r \sin \theta, t \sin \theta + r \cos \theta) \, dr. \quad (1.23)$$

Le théorème de la **coupe centrale** décrit ci-dessous est l'un des plus importants en tomographie :

**Théorème 1.2.3.1: coupe centrale**

La transformée de Fourier de la projection  $p_{\theta}$  vérifie

$$\forall \theta \in [0, \pi[ \text{ et } \forall \xi \in \mathbb{R} \quad \mathcal{F} [p_{\theta}] (\xi) = \mathcal{F} [f] (\xi \cos \theta, \xi \sin \theta), \quad (1.24)$$

où  $\mathcal{F}$  est l'opérateur de transformée de Fourier.

La démonstration de ce théorème est en annexe A.1.1. Un exemple de projections est donné en figure 1.31.

La transformée de Fourier donne une représentation fréquentielle d'un signal et ceci peut simplifier les opérations que l'on peut effectuer sur le signal. Dans notre cas, cela facilite la reconstruction de l'image à partir de ses projections. L'idée est de se servir de l'ensemble des projections

9. Weisstein, Eric W. *Delta Function*. MathWorld <http://mathworld.wolfram.com/DeltaFunction.html>.



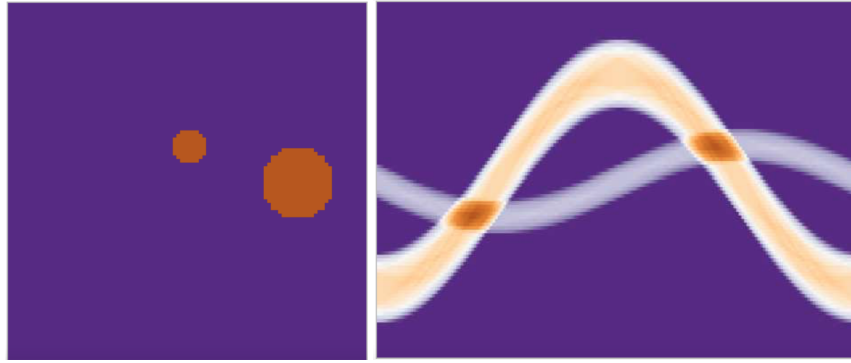


FIGURE 1.31 – A gauche, exemple de fantôme simulé avec deux billes - A droite, sinogramme du fantôme, en abscisse l'angle  $\theta$  (*i.e.* l'angle de rotation du bloc moteur) et en ordonnée la coordonnée radiale (*i.e.* les pixels du détecteur). Chaque colonne de cette image représente une projection du fantôme selon un angle donné.

et d'effectuer l'opération duale de celle-ci : **la rétroprojection.**

La rétroprojection dépend de la géométrie des faisceaux, mais le principe fondamental est le même, il sera expliqué dans les prochains paragraphes.

### 1.2.3.4 Rétroprojection filtrée en faisceau parallèle

Il est possible de reconstruire l'image  $f(x, y)$  en définissant l'opérateur adjoint de  $P$  de la manière suivante :

$$\begin{aligned} P^* &: L^2([0, \pi] \times \mathbb{R}) \rightarrow L^2(\mathbb{R}^2) \\ u &\mapsto P^* u. \end{aligned}$$

Son adjoint  $P^*$  est tel que

$$\forall f \in L^2(\mathbb{R}^2), \forall u \in L^2([0, \pi] \times \mathbb{R}) \quad \langle P f, u \rangle_{L^2([0, \pi] \times \mathbb{R})} = \langle f, P^* u \rangle_{L^2(\mathbb{R}^2)}$$

où  $\langle \cdot, \cdot \rangle_{L^2}$  dénote le produit scalaire usuel sur  $L^2$ . On peut alors réécrire l'égalité

$$\int_0^\pi \int_{\mathbb{R}} \overline{P_f(\theta, t)} u(\theta, t) dt d\theta = \int_{\mathbb{R}} \int_{\mathbb{R}} \overline{f(x, y)} P^* u(x, y) dx dy.$$

Remarquons que les valeurs de  $\overline{P_f(\theta, t)} \in L^2([0, \pi] \times \mathbb{R})$  sont réelles et ainsi on a  $\overline{P_f(\theta, t)} = P_f(\theta, t)$ . On utilise la définition (1.22) à gauche de l'égalité

$$\int_0^\pi \int_{\mathbb{R}} \left[ \int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy \right] u(\theta, t) dt d\theta = \int_{\mathbb{R}} \int_{\mathbb{R}} \overline{f(x, y)} P^* u(x, y) dx dy.$$

On change alors l'ordre d'intégration pour obtenir

$$\int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) \left[ \int_0^\pi \int_{\mathbb{R}} \delta(x \cos \theta + y \sin \theta - t) u(\theta, t) dt d\theta \right] dx dy = \int_{\mathbb{R}} \int_{\mathbb{R}} \overline{f(x, y)} P^* u(x, y) dx dy.$$

Comme  $\overline{f(x, y)} \in L^2(\mathbb{R}^2)$ , s'ensuit que  $\overline{f(x, y)} = f(x, y)$ . On en déduit par identification que

$$P^* u(x, y) = \int_0^\pi \int_{\mathbb{R}} \delta(x \cos \theta + y \sin \theta - t) u(\theta, t) dt d\theta$$

ce qui donne au final

$$P^* u(x, y) = \int_0^\pi u(\theta, x \cos \theta + y \sin \theta) d\theta. \quad (1.25)$$

Cependant, utiliser directement la rétroprojection (1.25) donne une version « floutée » de l'image originale car celle-ci n'est pas l'inverse de la transformée de Radon. On peut le voir sur un exemple en figure 1.32.

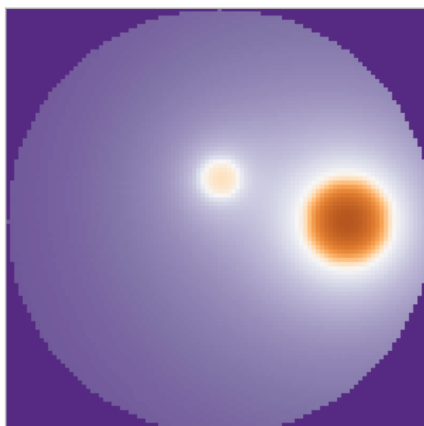


FIGURE 1.32 – Rétroprojection directe de la transformée de Radon. On peut voir des bords floutés (disques blancs) apparaître sur l'image.

Une méthode de reconstruction analytique utilisée depuis longtemps est la **rétroprojection filtrée** ou « *filtered back projection* » (**FBP**). Pour ce faire on utilise alors le théorème suivant :

#### Théorème 1.2.3.2: Rétroprojection filtrée

Soit  $p_\theta(t)$  la projection de  $f$  au point  $t$  et à l'angle  $\theta$ . Alors  $\forall x, y$

$$f(x, y) = \int_0^\pi (p_\theta \star h)(x \cos \theta + y \sin \theta) d\theta, \quad (1.26)$$

où  $\star$  dénote le produit de convolution et  $h$  est défini par  $\hat{h}(\xi) = |\xi|$ ,  $\forall \xi \in \mathbb{R}$ .  $h$  est appelé **filtre rampe**.

La démonstration de ce théorème est en annexe [A.1.2](#).

La rétroprojection filtrée est largement utilisée en pratique car elle permet d'inverser exactement la transformée de Radon et par conséquent permet d'obtenir des bords nets comme on peut le voir sur l'exemple en figure [1.33](#). Toutefois des artefacts peuvent survenir si l'échantillonnage angulaire est insuffisant.

L'algorithme classique de reconstruction **FBP** en faisceaux parallèles est alors basé sur le théo-

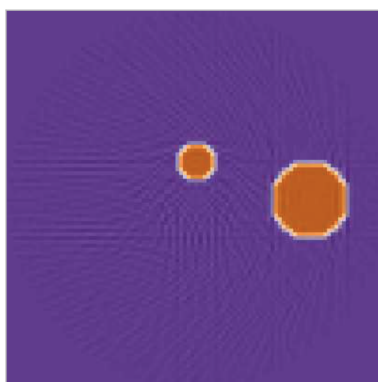


FIGURE 1.33 – Reconstruction par **FBP** du fantôme à deux billes. On remarque que les « disques blancs » ont disparu, mais l'image est moins nette et présente des effets de moiré, car les hautes fréquences ont été coupées en raison d'un sous-échantillonnage angulaire.

rème de rétroprojection filtrée précédent. Il consiste à estimer l'objet en appliquant successivement deux étapes :

1. filtrage des projections par filtre rampe dans le domaine fréquentiel à l'aide de la transformée de Fourier rapide (FFT) et de la transformée de Fourier inverse rapide (IFFT),
2. rétroprojection des projections filtrées dans l'espace de l'objet.

**Remarque**

Le filtre rampe est un filtre passe-haut qui conserve donc le signal en haute fréquence et filtre le signal à basse fréquence. Comme les erreurs d'interpolation et les bruits sont principalement des signaux de hautes fréquences, le filtre rampe a tendance à amplifier ces composantes hautes fréquences indésirables et l'approche de reconstruction **FBP** avec le filtre rampe devient très sensible au bruit lorsqu'il s'agit de données expérimentales.

Afin d'améliorer la robustesse de l'algorithme, le filtre utilisé dans **FBP** combine souvent le filtre rampe avec une **fenêtre d'apodisation** qui atténue le filtrage à haute fréquence. Un exemple de filtre est donné en figure 1.34. Le problème est que l'on perd aussi en détail dans l'image et on peut donc se retrouver avec une image reconstruite très lisse.

Notons qu'un même filtre n'est pas adapté à toutes les situations et devrait donc être choisi en

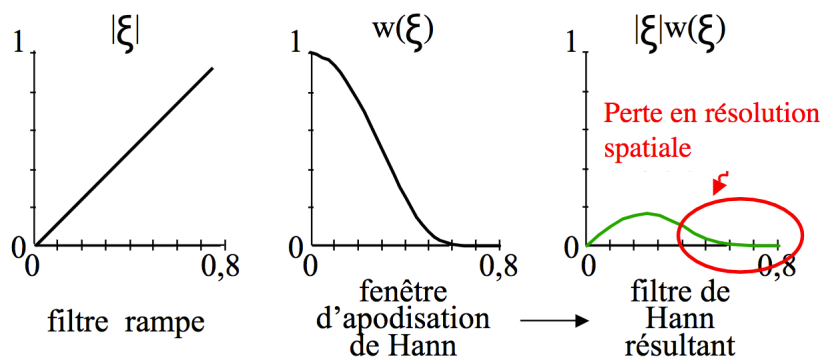


FIGURE 1.34 – Exemple de filtre rampe avec une fenêtre d'apodisation de Hann. La combinaison des deux donne un filtre qui redistribue la moyenne des fréquences, ce qui atténue le bruit, mais lisse potentiellement l'image.

fonction de celles-ci comme par exemple la finalité des images, la statistique de comptage, la géométrie des faisceaux, etc.

Le faisceau parallèle de rayons X est le cas de figure le plus simple à traiter du point de vue de la reconstruction. A contrario de modalités d'imagerie en médecine nucléaire, on ne trouve ce type de faisceau que très rarement dans les appareils modernes (hormis quasi exclusivement avec les sources de lumière synchrotron ou les accélérateurs linéaires), car les tubes à rayons X produisent un faisceau divergent. Dans les paragraphes qui suivent, nous allons voir comment traiter ce cas de figure.

**1.2.3.5 Rétroprojection filtrée en faisceaux divergents**

Les systèmes d'acquisition en tomodensitométrie sont caractérisés par un faisceau de rayons divergents. Ces rayons X sont en faisceau conique pour le cas 3D et en faisceau en **éventail** (ou « *fan-beam* ») si l'on ne considère qu'un seul plan du faisceau ou si l'on considère une seule ligne de pixels du détecteur (illustration en figure 1.35). Dans cette situation, la reconstruction **FBP** doit tenir compte de la transformation de la géométrie du faisceau. La géométrie de projection est alors différente et peut être ramenée au cas parallèle par des considérations géométriques.

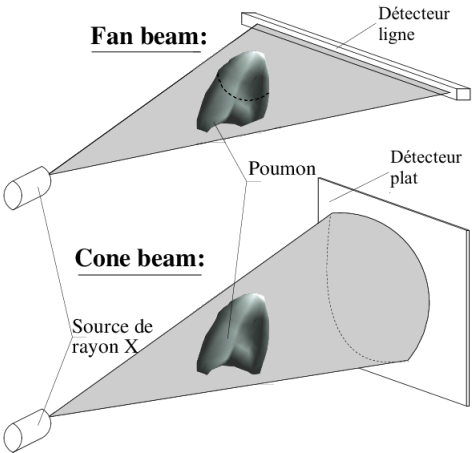


FIGURE 1.35 – Illustration des faisceaux divergents. En haut, une TDM avec un faisceau en éventail et un détecteur ligne. En bas, une TDM avec un faisceau conique et un détecteur plan.

On représente sur la figure 1.36 la géométrie d’un scanner dans le plan  $(x, y)$  où le point S représente la source.

Une projection est définie par l’angle  $\beta$  entre l’axe  $y$  et le rayon passant par la source et l’origine du repère. Chaque rayon est caractérisé par l’angle  $\alpha$  entre le rayon central et lui-même.

Il existe une relation de passage entre le rayon central qui passe par l’origine et les rayons divergents. En utilisant les changements de variables adéquats et l’équation (1.26) on peut montrer que

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \frac{1}{R^2} \int_{-\infty}^{\infty} Q(\alpha, \beta) H(\alpha' - \alpha) d\alpha d\beta, \tag{1.27}$$

avec

- $H(\gamma) = \left(\frac{\gamma}{\sin \gamma}\right)^2 h(\gamma), \forall \gamma \in [0, 2\pi], h$  étant le filtre rampe,
- $Q(\alpha, \beta) = D \cos(\alpha) p_{(\alpha+\beta)} (D \sin(\alpha)).$

On trouvera en annexe A.1.3 une démonstration détaillée qui mène à l’équation (1.27). C’est cette équation qui caractérise l’approche FBP pour le cas où le faisceau est en géométrie divergente.

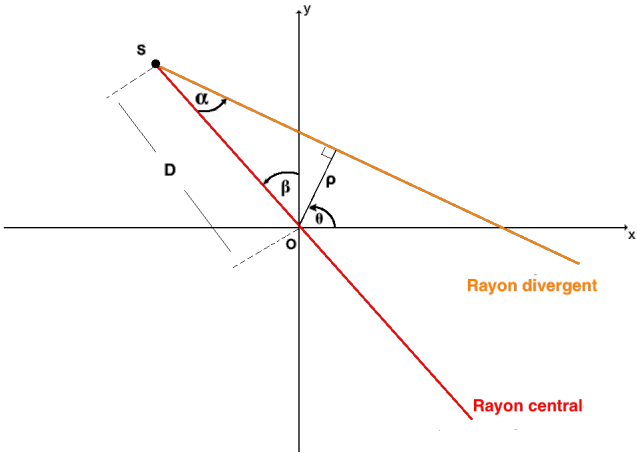


FIGURE 1.36 – Illustration de la géométrie fan-beam. En rouge le rayon central qui passe par le centre du détecteur. Le rayon orange est le rayon divergent qui forme un angle  $\alpha$  avec le rayon central. C’est ce rayon divergent que l’on souhaite caractériser (voir A.1.3 pour plus de détail).

### 1.2.3.6 Rétroprojection filtrée en faisceau conique

Contrairement à la géométrie en faisceau en éventail, la géométrie en faisceau conique capture immédiatement une image bidimensionnelle par le biais d'un détecteur plan (2D) (voir figure 1.35). Ce faisceau 3D s'avère efficace puisqu'une rotation autour de l'objet fournit suffisamment d'informations pour reconstruire une **image tridimensionnelle**. Bien que l'on reconstruise des images 3D, une grande partie de la logique sous-jacente est similaire au cas bidimensionnel et peut donc être considérée comme une extension du processus de reconstruction 2D en géométrie en éventail.

Néanmoins, le problème de la géométrie en faisceau conique est qu'aucun algorithme ne peut fournir une reconstruction exacte et stable de l'objet, car cette géométrie ne satisfait pas la **condition de Tuy** [ZENG, 2010], ce qui entraîne des problèmes de complétude des données [HERMAN, 1981]. Cette condition est respectée lorsque chaque plan qui intersecte l'objet imagé contient le point source du faisceau.

On illustre cela en figure 1.37. Si l'on dessine un plan traversant l'objet au-dessus (ou au-dessous) du plan **orbital**<sup>10</sup> de la figure de gauche et parallèle à ce plan, on voit que ce plan ne coupera jamais l'orbite circulaire et par conséquent il ne contiendra jamais le point source. Les orbites hélicoïdales et circulaires dans le cas de la coupe 2D correspondant au plan orbital telles qu'illustrées sur cette même figure 1.37 satisfont toutes deux la condition de Tuy et peuvent être utilisées pour acquérir des projections coniques, respectivement en éventail pour le cas 2D, pour une reconstruction exacte de l'image.

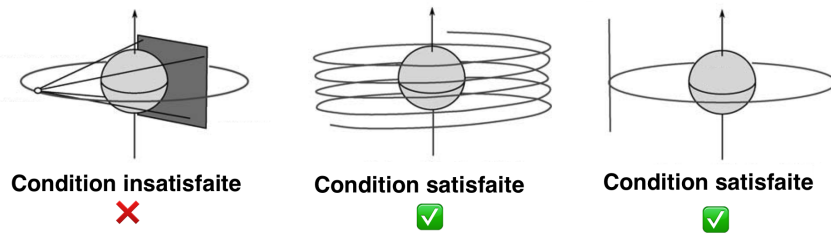


FIGURE 1.37 – Illustration de la **condition de Tuy**. A gauche, un système d'acquisition en orbite circulaire comparable à celui que nous retrouvons avec les prototype PIXSCAN-FLI, qui ne satisfait pas la condition de Tuy. Au milieu et à droite, des systèmes d'acquisition en orbites hélicoïdales et circulaire-ligne qui satisfont la condition de Tuy.

### Reconstruction

L'algorithme le plus populaire utilisé pour la reconstruction d'images en géométrie conique pour notre système d'acquisition à point focal circulaire est celui de Feldkamp, Davis et Kress (**FDK**) [FELDKAMP et collab., 1984]. Il s'agit d'un algorithme basé sur la rétroprojection filtrée, qui est facile à utiliser.

Comme nous l'avons cité précédemment, notre système d'acquisition ne satisfait pas la condition de Tuy et par conséquent l'algorithme **FDK** ne peut fournir que des reconstructions approximatives. Ainsi, des artefacts peuvent apparaître surtout à des endroits éloignés du plan orbital.

L'angle conique, tel que défini en figure 1.38, est un paramètre déterminant pour la qualité de la reconstruction. En pratique, on veille à ce que cet angle ne dépasse pas 10 degrés [ZENG, 2010] pour que cet algorithme donne des images assez bonnes. Notons que dans le plan orbital, les images reconstruites par l'algorithme **FDK** sont exactes.

10. Le plan qui contient l'axe optique, la projection orthogonale du point source sur le détecteur.

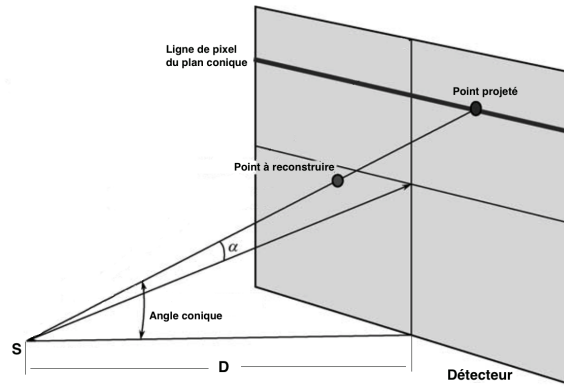


FIGURE 1.38 – Illustration du système de coordonnées de l’algorithme **FDK**.

### Algorithme FDK

Comme nous l’avons mentionné précédemment, l’algorithme FDK implémente une rétroprojection filtrée. Il s’appuie principalement sur la conversion du problème de reconstruction en faisceau conique à un problème de reconstruction en faisceau en éventail à chaque hauteur (ou valeur de  $z$ ). Les étapes décrites dans l’algorithme **FDK** sont donc assez similaires à celles en faisceaux en éventail. L’algorithme général est décrit comme suit :

1. pondérer les projections selon le cosinus de l’angle conique,
2. appliquer le filtre rampe aux projections ainsi échelonnées,
3. appliquer la rétroprojection aux données ainsi filtrées et échelonnées en veillant à les pondérer par un coefficient qui dépend de la distance  $D$  (distance source-origine).

Nous illustrons dans la figure 1.39 l’orbite circulaire de la source ainsi que le système de coordonnées de projection pour la géométrie du faisceau conique. Ce système est similaire au système de coordonnées du faisceau en éventail. Cependant, une dimension supplémentaire  $z$  est ajoutée pour représenter la troisième dimension de la projection du faisceau conique.

Le point  $(x, y, z)$  représente un point quelconque de l’objet imagé, où  $Oxyz$  représente un système de coordonnées cartésiennes.

Afin de simplifier les notations, on place le détecteur sur l’axe de rotation  $z$  de sorte que l’axe  $b$  de ce détecteur coïncide avec l’axe  $z$ . Les pixels du détecteur sont donc situés avec des coordonnées cartésiennes  $a$  et  $b$ . Un rayon  $X$  et sa projection peuvent être déterminés de façon unique par  $p(\beta, a, b)$ , où  $\beta$  est l’angle de rotation.

Basé sur la géométrie de la figure 1.39 on peut écrire [Li et collab., 2011] la formule de reconstruction utilisée dans l’algorithme **FDK** de la manière suivante

$$f(x, y, z) = \frac{1}{2} \int_0^{2\pi} \frac{D^2}{(D + x \sin(\beta) - y \cos(\beta))^2} \left( \left( \frac{D}{\sqrt{D^2 + A^2 + B^2}} p(\beta, A, B) \right) \star h(A) \right) d\beta \quad (1.28)$$

où

- $h$  est le filtre rampe,
- $p(\beta, A, B)$  est la projection conique,
- $A = \frac{D(x \cos(\beta) + y \sin(\beta))}{D + x \sin(\beta) - y \cos(\beta)}$ ,
- $B = \frac{zD}{D + x \sin(\beta) - y \cos(\beta)}$ .

On y reconnaît une convolution des projections par le filtre  $h$  suivie d'une rétroprojection pondérée. Ces opérations sont simples et parallélisables, ce qui en fait l'algorithme de prédilection en tomodensitométrie. On se bornera en pratique à ne considérer que le volume reconstruit proche du plan de rotation  $z = 0$  car, comme nous l'avons précédemment cité, en dehors de ce plan il n'y a pas de garantie quant à la qualité de la reconstruction. Notons toutefois qu'il existe diverses méthodes dérivant de l'algorithme **FDK** et qui permettent d'améliorer la qualité de la reconstruction [Li et collab., 2011].

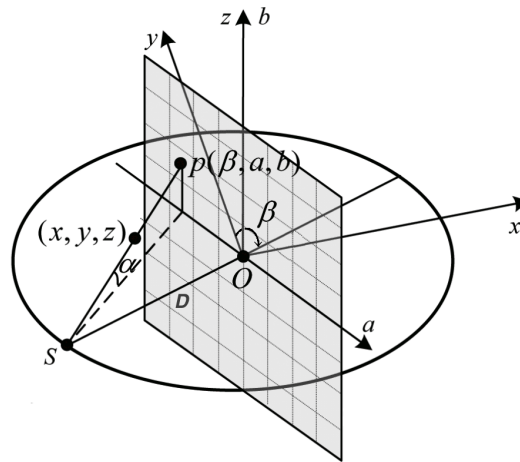


FIGURE 1.39 – Illustration de la géométrie conique pour l'algorithme **FDK**.

### 1.2.3.7 Conclusion

Nous parlons dans cette sous section de l'outil mathématique au cœur même du sujet de thèse : **la projection tomographie**. Nous avons dans un premier temps expliqué ce que sont les projections et les rétroprojections et quelles sont les opérations mathématiques qui les définissent. Nous avons par la suite introduit les méthodes fondamentales de reconstruction d'un objet à partir des projections obtenues selon une géométrie de faisceaux donnée. De ces méthodes découlent deux algorithmes très populaires dans le monde de la tomographie : **FBP** et **FDK**.

Ces algorithmes ont été largement utilisés dans les applications de tomodensitométrie pour la reconstruction d'objets. Leur principal avantage est le faible coût de calcul car ces méthodes utilisent des opérations qui peuvent être parallélisables. Ils comportent principalement trois étapes :

- la pondération par un coefficient qui dépend de la géométrie,
- le filtrage par un filtre rampe ou un substitut de ce filtre,
- la rétroprojection des données.

Un autre mérite de ces algorithmes est qu'ils permettent une reconstruction de l'objet presque aussi instantanément que l'acquisition des données, cela permet un gain de temps total considérable.

Cependant, ces méthodes comportent certaines limites critiques lorsqu'il s'agit de données expérimentales. Tout d'abord, elles sont toujours sensibles au bruit, même avec des filtres améliorés. En effet, comme mentionné en paragraphe 1.2.3.4, les filtres améliorés fonctionnent en bloquant les signaux en hautes fréquences. Or, les signaux hautes fréquences comprennent non seulement le bruit, mais aussi des structures fines et détaillées de l'objet. Effectuer donc un tel filtrage supprimera également les détails, ce qui limite la résolution des images reconstruites. Il faut souvent faire un compromis entre la résolution de reconstruction et la robustesse face au bruit dans le choix d'un filtre approprié et sa fréquence de coupure [CAI, 2013].

D'autre part, ces algorithmes donnent les mêmes poids pour toutes les observations sur tous les

pixels de l'objet sans considérer les propriétés physiques de l'acquisition. Par exemple, un détecteur peut avoir plusieurs pixels morts qui n'enregistrent pas l'information ou bien l'objet peut contenir des composants à forte atténuation qui absorbent tous les rayons X qui passent (comme un métal par exemple). Le premier cas donne des artefacts en anneau tandis que le second donne des artefacts métalliques sur la reconstruction.

De plus, ces méthodes sont basées sur le modèle dans lequel les projections sont continues. En pratique, elles sont représentées dans un format discret. On trouvera néanmoins dans la littérature [DESBAT et collab., 2004; NATTERER, 1993] des méthodes d'échantillonnage pour la reconstruction dans ce type de cas.

L'alternative est d'utiliser des méthodes **itératives**. Dans ces approches, on décrit la relation entre l'objet et les mesures à l'aide d'une équation discrète en formulant le problème de reconstruction tomographique comme un **problème inverse**<sup>11</sup>. L'objectif de la reconstruction est alors de résoudre ce problème et d'obtenir la description de l'objet. Ces approches, qui peuvent considérablement augmenter la qualité de la reconstruction, seront discutées dans la section 1.3.

#### 1.2.4 En bref

Cette section avait pour but d'expliquer les différentes briques physiques de notre problème qui nous permettent de comprendre le modèle (1.8)

$$\mathbf{Y} = e^{-\mathbf{SA}\Sigma\mathbf{F}}.$$

Dans un premier temps nous avons introduit dans les paragraphes 1.2.1.2 et 1.2.1.3 le principe d'interaction et le principe d'atténuation entre les rayons X et la matière. Ceci nous a permis de décrire ce que contiennent les matrices  $\mathbf{A}$  et  $\Sigma$  du modèle précédent.

Nous avons par la suite décrit en sous section 1.2.2 le tomodensitomètre PIXSCAN-FLI sur lequel les expériences de cette thèse ont été menées ainsi que les caractéristiques qui nous permettent de faire de la tomographie spectrale. Ceci nous a aidé à mieux comprendre ce que contient la matrice  $\mathbf{F}$ .

Enfin nous avons introduit en sous section 1.2.3 la tomographie standard et les outils mathématiques qui interviennent en **TDM**. Nous avons par la même occasion présenté les algorithmes fondamentaux de la famille des **méthodes analytiques** utilisés pour la reconstruction en **TDM** d'absorption classique. Le problème général de ces méthodes est qu'elles ne tiennent compte ni des fluctuations statistiques des données (taux de comptage), ni des perturbations physiques liées à l'acquisition (comme par exemple le fait que le source de rayon X soit **polychromatique**).

Ceci nous a permis de poser les bases afin d'introduire les **méthodes itératives**. C'est dans cette famille de méthodes que s'inscrit le travail effectué durant cette thèse, elles seront développées dans la section 1.3. Nous expliquerons par la même occasion ce que contient la matrice  $\mathbf{S}$  de notre modèle.

11. Voir chapitre 2 pour plus de détails sur les problèmes inverses.



## 1.3 Méthodes et état de l'art

L'objectif de cette section est de mettre la lumière sur les principales méthodes de reconstructions en **TDM**. Ces approches de reconstruction peuvent être classées en deux familles : **analytique** et **itérative**.

Les approches analytiques permettent d'estimer le **CAL** de l'objet rapidement et avec un faible coût de calcul comme nous l'avons vu en sous-section 1.2.3.5. Mais ces méthodes sont limitées en terme de performance car elles sont sensibles au bruit.

On peut retrouver deux types d'approche parmi les méthodes itératives : les approches **déterministes** (ou **algébriques**) et les approches **statistiques**.

Les méthodes itératives, bien que plus coûteuses en terme de temps de calcul, ont des avantages en terme de qualité de reconstruction intéressants.

Dans un premier temps nous allons survoler quelques-unes de ces méthodes dans le cas de la **TDM** d'absorption classique. L'objectif est de montrer le problème de reconstruction dans le cas **discret**. Ceci va nous permettre trois choses :

- expliquer le contenu de la matrice **S** de notre modèle (1.8),
- introduire la notion de problème inverse, qui sera par la suite développée plus avant au chapitre 2,
- mettre en perspective le problème de **TDM spectrale** par rapport à la **TDM** d'absorption classique.

Nous rappellerons par la suite la problématique de **TDM spectrale**, puis nous verrons l'état de l'art des méthodes de reconstructions dans le cadre de la **TDM spectrale**.

### 1.3.1 TDM d'absorption classique

#### 1.3.1.1 Modélisation

Les méthodes itératives de reconstruction en **TDM** d'absorption classique reposent sur la modélisation numérique du phénomène physique qui met en relation les données et le **CAL** de l'objet imagé, que l'on représentera ici par la fonction  $f$ . Comme nous l'avons vu en sous section 1.2.3.3 pour le cas continu, la relation entre les projections et le **CAL** peut s'écrire sous la forme suivante

$$p_{\theta}(t) = \int_{L_{\theta}} f(v) dv = \int_{\mathbb{R}} f(t \cos \theta - r \sin \theta, t \sin \theta + r \cos \theta) dr. \quad (1.29)$$

Or dans le cas pratique, les données à reconstruire ainsi que les projections mesurées forment un ensemble discret d'éléments. Afin de s'adapter à cela, le modèle (1.29) doit lui aussi être discrétisé. Cette **discrétisation** consiste en un premier temps à échantillonner l'objet en le considérant comme un signal.

On se place par commodité dans le cas 2D, le problème étant le même en 3D. L'idée est de superposer l'image sur une grille et de considérer que l'image est constante par morceau dans chaque case de la grille, donc plus une grille a de cases plus l'image discrétisée est nette. On peut formaliser cela en considérant l'image  $f(x, y)$  comme une somme de coefficients pondérés par une fonction d'échantillonnage

$$f(x, y) = \sum_{i=1}^D \mu_i \delta_i(x, y) \quad (1.30)$$

avec  $D$  le nombre total de pixels de l'objet,  $\mu_i$  la valeur moyenne de  $f$  au pixel  $i$  et  $\delta$  une fonction **porte** qui définit spatialement les pixels. Un exemple de fonction porte est le symbole de **Kronecker** défini par

$$\delta_i(x, y) = \begin{cases} 1 & \text{si } (x, y) \in \text{pixel } i, \\ 0 & \text{sinon.} \end{cases}$$

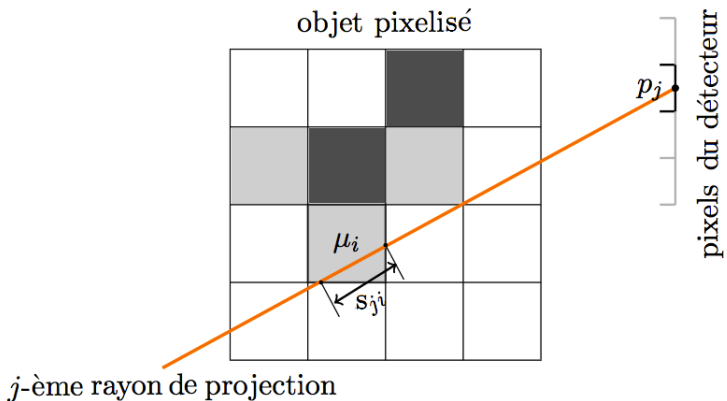


FIGURE 1.40 – Illustration du principe de discrétisation de l’objet.

Notons par ailleurs que dans le cas 3D, on parlera de **voxels** pour l’objet.

On considère alors un rayon traversant l’objet, tel qu’illustré en figure 1.40.  $p_j$  correspond à la projection du  $j$  ème rayon sur un pixel du détecteur. On note alors  $s_{ji}$  l’intersection du  $j$  ème rayon avec le  $i$  ème voxel de l’objet, cette quantité détermine la **contribution** de chaque voxel de l’image dans les pixels du détecteur et on la mesure à l’aide de la **longueur d’intersection** du rayon avec le voxel. Nous pouvons alors formuler la projection  $p_j$  par

$$p_j = \sum_{i=1}^D s_{ji} \mu_i. \tag{1.31}$$

Cette projection peut être vue comme la version discrétisée de l’intégrale le long de ligne L dans l’équation (1.29). Notons aussi qu’il existe plusieurs modèles de contribution de l’intensité des pixels comme cela est illustré en figure 1.41.

En effet la figure de gauche illustre le modèle que nous utilisons, il consiste à prendre la longueur d’intersections du rayon avec le voxel.

La figure du milieu illustre le modèle basique dit modèle de Dirac. Ce modèle est assez simple, toute intersection avec un voxel compte de la même manière, la contribution vaut donc 1 si le rayon passe par le voxel, 0 sinon. La figure de droite illustre le modèle exact qui consiste à prendre la surface intersectée. Chacun de ces modèles présente des avantages et des inconvénients qui sont discutés dans [DU et collab., 2017; PRATX et XING, 2011; ZHAO et collab., 2009].

Si on note  $N_p$  le nombre de pixels du détecteur et  $N_\theta$  le nombre d’angles alors le nombre de

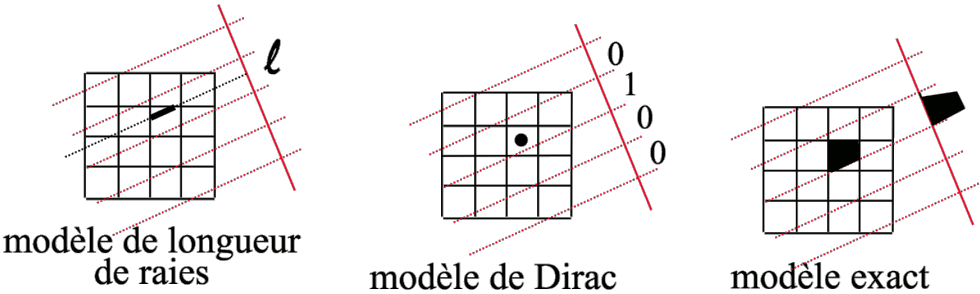


FIGURE 1.41 – A gauche, le modèle de longueur d’intersection, au milieu, le modèle de Dirac, à droite, le modèle exact.

mesures totales est  $N_t = N_p N_\theta$ . On peut alors exprimer matriciellement les projections

$$P = S f \tag{1.32}$$

avec

$$S = \begin{pmatrix} s_{11} & \cdots & s_{1D} \\ \vdots & \ddots & \vdots \\ s_{N_t,1} & \cdots & s_{N_t,D} \end{pmatrix} \quad f = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_D \end{pmatrix} \quad P = \begin{pmatrix} p_1 \\ \vdots \\ p_{N_t} \end{pmatrix}$$

L'équation (1.32) peut alors être vue comme la version discrétisée de l'équation (1.29). On appelle **S** la **matrice système** (ou **opérateur de projection**), elle dépend de la physique d'acquisition considérée ainsi que de sa géométrie. **P** constitue le sinogramme et **f** le vecteur des valeurs de l'objet que l'on souhaite reconstruire.

### Lien avec notre problème

Notre matrice **S** du modèle (1.8) est ainsi définie que ses coefficients décrivent le « taux d'intersection » d'un faisceau avec les voxels de l'objet. La modélisation numérique illustrée par l'équation (1.32) du système prend toute son importance dans le calcul des **projections**, modélisées par la matrice **S**, et des **rétroprojections**, modélisées par **S<sup>t</sup>** [PRATX et XING, 2011]. De nombreuses possibilités existent pour la modélisation des projections par la matrice **S** dont dépendent la qualité, mais aussi la stabilité numérique et la vitesse de convergence de la reconstruction.

On sépare souvent en deux catégories les différentes implémentations : **orientées pixels** (*ray-driven*) ou **orientées voxels** (*voxel-driven*). Dans notre cas, nous utilisons la reconstruction **OpenRTK Toolkit**, un software développé pour la reconstruction rapide en **TDM** à faisceau conique [RIT et collab., 2014].

Il existe plusieurs études comparatives entre ces implémentations de projections dans [XU et MUELLER, 2006]. Notons toutefois qu'un critère de choix important reste la rapidité du calcul car, dans notre cas, la taille de la matrice **S** est très grande (de l'ordre de  $10^{14}$  éléments de matrice) et donc impossible à stocker en mémoire durant la reconstruction. Pour cette raison nous utilisons les routines de calcul sur GPU de OpenRTK permettant de calculer les projections et rétroprojections à la « volée ».

### 1.3.1.2 Expression du problème

Nous allons montrer ici comment poser le problème de reconstruction en **TDM** d'absorption classique. Certains termes mathématiques seront employés ici sans être développés. On trouvera au chapitre 2 toutes les informations nécessaires à la compréhension de ces termes.

On peut tenter de résoudre le **problème inverse** lié à l'équation (1.32) de manière algébrique. Pour cela il suffirait de calculer la pseudo-inverse de la matrice **S** puis de multiplier les projections **P** par cette inverse pour obtenir la solution

$$\hat{f} = S^\dagger P. \tag{1.33}$$

Mais cette méthode **d'inversion directe** présente deux difficultés majeures :

- la matrice de projection **S** est mal **conditionnée** ce qui rend la solution (1.33) trop sensible au bruit présent dans les projections **P**,
- l'inversion directe est numériquement impossible car la taille de la matrice **S** dans le cas pratique est beaucoup trop grande. De plus, elle n'est pas carrée.

Une autre manière est de considérer le **problème d'optimisation** suivant

$$\hat{f} = \underset{f}{\operatorname{argmin}} h(f), \quad (1.34)$$

où  $h$  est le **terme d'attache aux données**, qui peut avoir différentes expressions basées sur le modèle d'observation assigné ainsi que le modèle de mesure. C'est ce problème d'optimisation qui est généralement résolu par des algorithmes itératifs.

Afin d'améliorer la robustesse au bruit de l'approche de reconstruction basée sur le problème (1.34), des **régularisations** sont introduites sur l'objet  $f$ . Ces régularisations ont pour but de modéliser des informations *a priori* dont on dispose sur l'objet à reconstruire. On peut alors réécrire le problème comme suit

$$\hat{f} = \underset{f}{\operatorname{argmin}} h(f) + \lambda g(f), \quad (1.35)$$

où  $\lambda$  est un **hyperparamètre** qui permet de faire un compromis entre  $h$  et  $g$ . La fonction de régularisation  $g$  est introduite pour réduire le bruit présent sur les résultats de reconstruction causés par le mauvais conditionnement que l'on a en **TDM** d'absorption. La qualité de la reconstruction finale dépend de la fonction de régularisation choisie et de l'hyperparamètre  $\lambda$ .

### Exemple de résolution

On peut réécrire le problème (1.34)

$$\hat{f} = \underset{f}{\operatorname{argmin}} \|P - \mathbf{S}f\|^2, \quad (1.36)$$

où  $\|\cdot\|^2$  est la norme euclidienne. Ce problème a une solution analytique que l'on peut directement exprimer à l'aide de **l'estimateur des moindres carrés**

$$\hat{f} = (\mathbf{S}^t \mathbf{S})^{-1} \mathbf{S}^t P, \quad (1.37)$$

où  $\mathbf{S}^t$  est la matrice transposée de  $\mathbf{S}$ , c'est aussi la matrice de **rétroprojection** dans la reconstruction en **TDM** d'absorption. L'invertibilité de l'opérateur de **projection/rétroprojection** ( $\mathbf{S}^t \mathbf{S}$ ) dépend du rang de la matrice  $\mathbf{S}$  et peut être lui aussi coûteux. Par ailleurs, il peut lui aussi être mal conditionné et l'estimateur (1.37) souffre également du bruit de mesure.

Une régularisation couramment utilisée en **TDM** d'absorption classique est la régularisation de **Tikhonov** [TIKHONOV, 1963]. Elle a été introduite pour diminuer le bruit et lisser l'objet, on peut l'exprimer dans (1.36)

$$\hat{f} = \underset{f}{\operatorname{argmin}} \|P - \mathbf{S}f\|^2 + \lambda \|\Gamma f\|^2, \quad (1.38)$$

où  $\Gamma$  est la matrice de Tikhonov qui est choisie selon les données à traiter. Celle-ci peut être simple (égale à la matrice identité) ou plus élaborée (opérateur passe-haut, opérateur de Fourier). L'objectif de cette régularisation est d'améliorer le conditionnement du problème.

Il existe bien entendu d'autres estimateurs pour résoudre le problème (1.35). Ils dépendent du bruit de mesure supposé et du modèle d'observation, mais le problème inverse reste en général le même.

Il faut noter aussi qu'il existe d'autres méthodes pour exprimer le problème en **TDM** telles que les méthodes statistiques et les méthodes bayésiennes. On trouvera dans [BRUYANT, 2002] plus de détails sur ces méthodes, qui toutes donnent en général le même terme d'attache aux données.

### 1.3.1.3 Recherche de solution

Le problème (1.35) précédemment cité se résout en général à l'aide d'algorithmes itératifs. Un algorithme d'optimisation itératif commence par un point d'initialisation et se met à jour de manière itérative pour approcher la solution. Selon les caractéristiques de la fonction objectif, différents algorithmes d'optimisation peuvent être utilisés. La performance de l'algorithme dépend de la fonction objectif et de la stratégie de mise à jour.

En ce qui concerne la fonction objectif, la **convexité** est une propriété importante dont il faut tenir compte dans le choix d'un algorithme d'optimisation approprié. En effet, lorsqu'une fonction est convexe, on peut montrer que le minimum de cette fonction est unique lorsque celui-ci existe, on appelle cela une solution **globale**. Pour cette raison, une fonction convexe est plus simple à minimiser qu'une fonction non convexe<sup>12</sup>.

Une autre propriété importante est la complexité de la fonction objectif. En général, on cherche à ce que celle-ci soit quadratique et donc issue d'un modèle linéaire. C'est pour cela qu'en **TDM** d'absorption classique, le choix de la linéarité est souvent fait. En **TDM** spectrale, la fonction (1.8) que l'on étudie est non linéaire, ce qui rend difficile la recherche de minimum. Dans l'état de l'art, les algorithmes qui résolvent un tel problème sont généralement appelés **algorithmes d'optimisation non linéaire**. On trouvera dans [BOYD et VANDENBERGHE, 2004] une description détaillée de ces algorithmes.

Un autre problème est celui de la **différentiabilité** du terme d'attache aux données **et** de la fonction de régularisation. En effet, les algorithmes classiques de **descente de gradient** et de **gradient conjugué** nécessitent de calculer la dérivée de ces fonctions à chaque itération. Dans le cas non différentiable, on devra recourir à d'autres outils.

Dans la suite nous allons présenter quelques méthodes de reconstructions qui sont classées en deux catégories dans la littérature relative à la **TDM** : les méthodes **déterministes** et les méthodes **statistiques**.

### 1.3.1.4 Méthodes algébriques

Une des premières méthodes de résolution du système numérique proposée est la méthode **Algebraic Reconstruction Technique (ART)** [GORDON, 1974]. On considère ici chaque équation, c'est-à-dire chaque rayon, de manière indépendante. L'objet est reconstruit itérativement à partir d'un volume initial  $f^0$  que l'on projette sur un pixel en utilisant la première équation du système et que l'on met par la suite à jour à partir de l'équation

$$f_i^{n+1} = f_i^n + \frac{p_j - (\mathbf{S}f_i^n)_j}{\sum_i s_{ji}^2} s_{ji}. \quad (1.39)$$

On répète l'opération pour  $i \in \{1, \dots, D\}$ . Cet ensemble d'opérations constitue un cycle du processus de reconstruction, répété jusqu'à convergence. Les reconstructions obtenues avec **ART** sont généralement assez bruitées, avec notamment un fort bruit poivre et sel. Ces artefacts proviennent de l'écart entre le modèle de projection décrit par **S** et la projection réelle.

La méthode **SART (Simultaneous Algebraic Reconstruction Technique)** [ANDERSEN et KAK, 1984] est basée sur la méthode précédente. Ici l'équation de mise à jour est

$$f_i^{n+1} = f_i^n + \lambda \sum_j \left( \frac{p_j - (\mathbf{S}f_i^n)_j}{\sum_i s_{ji}^2} s_{ji} \right) \quad (1.40)$$

où  $\lambda$  est une constante de relaxation comprise entre 0 et 1. La différence entre (1.40) et (1.39) est la prise en compte de tous les rayons simultanément à chaque itération. **SART** permet obtenir des reconstructions plus lisses dès les premières itérations [KAK et SLANEY, 2001].

12. Ceci sera mieux détaillé au chapitre 2.

**ART** et **SART** sont des algorithmes itératifs utilisés dans les systèmes pratiques de **TDM** pour reconstruire l'objet. Ils ont la garantie de converger vers la solution (1.37) [JIANG et WANG, 2003]. Cependant, même si **SART** s'avère plus robuste face au bruit de mesure, ces méthodes souffrent de l'amplification du bruit causée par le mauvais conditionnement relatif au problème de reconstruction en **TDM** d'absorption classique.

Notons par ailleurs que d'autres méthodes déterministes pour la **TDM** d'absorption classique ont été développées ces dernières années. On trouvera dans [BRUYANT, 2002] une classification et une comparaison des performances de celles-ci.

### 1.3.1.5 Méthodes par maximum de vraisemblance

Les méthodes du maximum de vraisemblance [LANGE et FESSLER, 1995] utilisent en général une formulation probabiliste du problème de reconstruction afin de modéliser explicitement la nature du bruit des données. Elles permettent de mieux traiter le problème de données incomplètes.

On considère ici un vecteur aléatoire  $Y = [Y_1, \dots, Y_{N_t}]$ . Le vecteur des mesures  $P = [p_1, \dots, p_{N_t}]$  est une réalisation du vecteur aléatoire  $Y$ . L'estimation du maximum de vraisemblance est obtenu en maximisant la probabilité conditionnelle  $\mathbb{P}(Y | f)$  dite aussi probabilité *a priori* :

$$\hat{f} = \operatorname{argmax} \mathbb{P}(Y | f). \quad (1.41)$$

Lorsque le bruit des données est poissonnien, par exemple, calculer la vraisemblance (1.41) n'est pas triviale et on utilise en règle générale des approches de types *expectation-maximization* (**EM**). Pour chaque itération, cette méthode procède en deux étapes :

- une étape d'évaluation de l'espérance, où l'on calcule l'espérance de la vraisemblance en tenant compte des dernières variables observées,
- une étape de maximisation, où l'on estime le maximum de vraisemblance des paramètres en maximisant la vraisemblance trouvée à la première étape.

On trouvera en annexe de [DUPONT, 2014] un descriptif assez détaillé de l'algorithme pour le cas de la **TDM** d'absorption classique.

Cette méthode peut être utilisé pour résoudre des problèmes d'optimisation complexes dans la reconstruction en **TDM** d'absorption. Elle peut garantir automatiquement une contrainte naturelle sur l'objet imagé : **la positivité**. Cependant on ne peut utiliser cette méthode sur d'autres types de contraintes.

On notera par ailleurs que d'autres méthodes existent telles que **OSEM**, une version accélérée de l'algorithme présenté qui considère une vraisemblance poissonnienne [HUDSON et LARKIN, 1994], ou encore **RAMLA** [XING et collab., 2013].

### 1.3.1.6 Conclusion

Les récents développements pour la reconstruction en **TDM** d'absorption classique se concentrent principalement sur les algorithmes **itératifs** (approches déterministes ou statistiques). Ceux-ci sont capables de donner une qualité de reconstruction plus élevée que celle obtenue avec des méthodes analytiques comme **FBP**.

Ces approches transforment habituellement le problème de reconstruction en un **problème d'optimisation** et sont ensuite résolues en utilisant des algorithmes d'optimisation. La qualité de la reconstruction va dépendre alors

- du modèle utilisé ainsi que du bruit d'observation assigné,
- de la méthode d'estimation choisie (déterministe ou statistique).

En pratique, la **complexité** de l'optimisation qui en résulte doit également être prise en compte. Cette complexité dépend de la taille des données à traiter, de la méthode de calcul du projecteur **S** vu en 1.3.1.1 et de l'algorithme choisi.

Cependant, les acquisitions en **TDM** d'absorption classique ne révèlent que la morphologie du patient, elles ne fournissent aucune information sur la composition chimique des structures examinées, car on ne tient pas compte de la dépendance en énergie des composants de l'objet imagé. En effet, certains composants peuvent avoir une composition chimique différente, mais le même coefficient d'atténuation apparaîtra avec la même valeur d'échelle de gris sur l'image reconstruite en **TDM** d'absorption. Comme cela est illustré en figure 1.42 [HEISMANN et collab., 2012], les plaques calcifiées et l'agent de contraste iodé sont chimiquement très différents, mais ils peuvent présenter une atténuation similaire et par conséquent être difficile à distinguer sur une image **TDM** d'absorption.

L'information en énergie est la caractéristique principale prise en compte en **TDM** spectrale.



FIGURE 1.42 – Angiographie de l'aorte à l'aide d'un agent de contraste iodé. Les vaisseaux remplis de l'agent contraste et les plaques calcifiées présentent une atténuation similaire des rayons X et sont difficiles à distinguer.

Dans les prochains paragraphes nous allons faire un bref rappel du problème en **TDM** spectrale ainsi que des motivations cliniques de celle-ci. Nous verrons ensuite l'état de l'art des méthodes qui existent pour la reconstruction en **TDM** spectrale.



## 1.3.2 TDM spectrale

### 1.3.2.1 Rappel du problème

La majeure différence entre la **TDM** d'absorption classique et la **TDM** spectrale vient de la prise en compte de la dépendance énergétique du **CAL**  $\mu$  d'un objet imagé. Cette prise en compte complexifie le modèle d'observation, mais peut fournir plus d'informations sur l'objet, notamment sur sa composition chimique.

On rappelle que, dans notre cas, lorsque la source est **polychromatique** et l'objet est **hétérogène**, on quantifie l'observation mesurée, noté ici  $y_{p,m}$ , avec la loi de Beer-Lambert

$$y_{p,m} = \int_{\mathbb{R}^+} F_m(E) e^{-\int_{\mathcal{C}_p} \mu(l,E) dl} dE, \quad (1.42)$$

où  $\mathcal{C}_p$  la ligne de visée et  $F_m(E)$  est la fonction de pondération rassemblant les variations énergétiques induites par les différentes configurations du tomodensitomètre ainsi que de la physique d'acquisition. Ce terme  $F_m(E)$  peut s'écrire pour des pixels de détection muni d'un unique seuil d'énergie :

$$F_m(E) = I_0(E) f_m(E) D_m(E) \mathbb{1}_{\{E > s_m\}}, \quad (1.43)$$

avec

- $I_0(E)$  est le spectre d'énergie de la source,
- $f(E)$  la réponse de filtres métalliques,
- $D(E)$  l'efficacité du détecteur,
- $\mathbb{1}_{\{E > s\}}$  est la fonction indicatrice du seuil en énergie telle que définie en (1.2).

Lorsque l'on discrétise le modèle (1.42), on retrouve sur notre modèle fondamentale

$$Y = e^{-SA\Sigma F} \quad (1.44)$$

où

- $Y \in \mathbb{R}^{P \times M}$  est la **matrice de données** qui contient dans chaque colonne la  $m$ -ième acquisition,
- $F \in \mathbb{R}^{N \times M}$  est le **dictionnaire de filtrage en énergie** qui contient dans chaque colonne les facteurs des variations d'énergie induites par les configurations de la  $m$ -ième acquisition,
- $S \in \mathbb{R}^{P \times D}$  est l'**opérateur de transformée** à rayons X qui modélise la projection de l'objet sur le détecteur,
- $A \in \mathbb{R}^{D \times K}$  est la **matrice des cartes des concentrations** que l'on souhaite reconstruire qui contient dans chaque colonne la répartition du  $k$ -ième composant dans le volume,
- $\Sigma \in \mathbb{R}^{K \times N}$  est le **dictionnaire des sections efficaces** d'interaction contenant dans chaque ligne l'évolution de la probabilité d'interaction en fonction de l'énergie du  $k$ -ième composant.

On rappelle que  $P$  est le nombre de mesures,  $M$  le nombre d'acquisitions,  $N$  le nombre d'échantillons en énergie,  $D$  le nombre de voxels de l'objet et  $K$  le nombre de composants de l'objet.

Il y a plusieurs différences notables entre le modèle (1.44) et le modèle (1.32) de la **TDM** d'absorption classique :

- La première est, comme nous l'avons cité, la prise en compte de l'énergie due à une source de rayons X polychromatique. Cette prise en compte se manifeste dans le modèle (1.44) par le biais de la matrice  $F$  et c'est ce qui empêche d'avoir une expression analytique de la solution.



- La seconde différence réside dans la non linéarité due à l'exponentielle. Ceci rend le problème de reconstruction plus difficile à résoudre.
- Enfin, on modélise l'objet comme une somme de composantes d'intérêt, chacune ayant une signature en énergie dont on tient compte dans le dictionnaire  $\Sigma$ . Ceci nous permet d'avoir la matrice des cartes des concentrations  $\mathbf{A}$ .

Cette modélisation est beaucoup plus proche de la réalité que celle de (1.32), on pourrait donc espérer avoir des reconstructions enrichies, car il y a plus d'informations disponibles dans les reconstructions idéales en TDM spectrale qu'en TDM d'absorption classique, mais le problème à résoudre devient plus complexe.

Ce problème de reconstruction peut être par la suite exprimé comme un problème inverse tel que nous l'avons fait en 1.1.2.5 :

$$\hat{\mathbf{A}} = \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\operatorname{arg\,min}} \{J(\mathbf{A}) = G(\mathbf{A}) + R(\mathbf{A})\}, \quad (1.45)$$

avec  $J(\mathbf{A})$  la fonction objectif,  $G(\mathbf{A})$  le terme d'attache aux données et  $R(\mathbf{A})$  le terme de régularisation. Nous verrons dans le chapitre 2 plus de détails sur les problèmes inverses.

Ce problème de reconstruction regroupe deux sous problèmes : celui d'une reconstruction tomographique et celui d'une séparation de source, car il s'agit ici de distinguer différents composants d'un seul objet. Ceci complexifie donc la résolution du problème mais nous allons dans les prochains paragraphes exposer ce que cela peut apporter en terme d'application.

### 1.3.2.2 Motivation

L'idée de la TDM spectrale est avancée dès 1973 par G. Hounsfield qui donne l'une de ses premières descriptions. Il suggérait d'utiliser des spectres de sources à des énergies différentes pour différencier les matériaux d'un objet [HOUNSFIELD, 1973]. Cependant, cette technique n'a jamais vraiment été mise en œuvre en raison de limitations technologiques qui impactaient la résolution spatiale, le bruit sur les reconstructions et la dose élevée de radiation requise à l'époque.

L'objectif de la TDM spectrale est d'ajouter de l'information supplémentaire sur les composants



FIGURE 1.43 – Séparation automatique des os : à gauche, image avec squelette, à droite, image de l'arbre vasculaire uniquement [HEISMANN et collab., 2012].

de l'objet imagé afin de surmonter la limitation de la TDM d'absorption classique. Ceci peut aider à améliorer le diagnostic médical de manière significative dans plusieurs applications cliniques. En effet, l'un des principaux avantages de la TDM spectrale pourrait être l'identification et la quantification de l'iode. Cette capacité répondrait à des besoins cliniques, comme par exemple caractériser une lésion bénigne ou maligne en quantifiant son absorption d'iode, ou déterminer l'impact

d'un caillot dans les artères pulmonaires sur la perfusion pulmonaire.

Par ailleurs, on pourrait aussi simplifier le diagnostic tomодensitométrique en séparant par exemple précisément et automatiquement les os et les vaisseaux pour obtenir une visualisation sans les os de l'arbre vasculaire lors des examens d'angiographie comme illustré en figure 1.43. On peut étendre cela à la séparation de plus de 2 composants. On trouvera dans le chapitre 5 de [HEISMANN et collab., 2012] plus d'informations sur les motivations cliniques de la TDM spectrale.

### 1.3.2.3 Acquisition de données en TDM spectrale

#### Côté source

Il existe plusieurs façons d'acquérir des données avec des informations spectrales. L'idée de base est d'utiliser différents spectres de rayons X avec différents réglages de haute tension. En TDM d'absorption classique, la haute tension appliquée au tube à rayons X peut être modifiée entre différents balayages en la commutant plus ou moins rapidement entre les différentes projections. C'est ce qu'on appelle communément le *kV switching* [KALENDER et collab., 1986] illustré en figure 1.44.

Une autre technique similaire et assez répandue depuis les récents développements technolo-

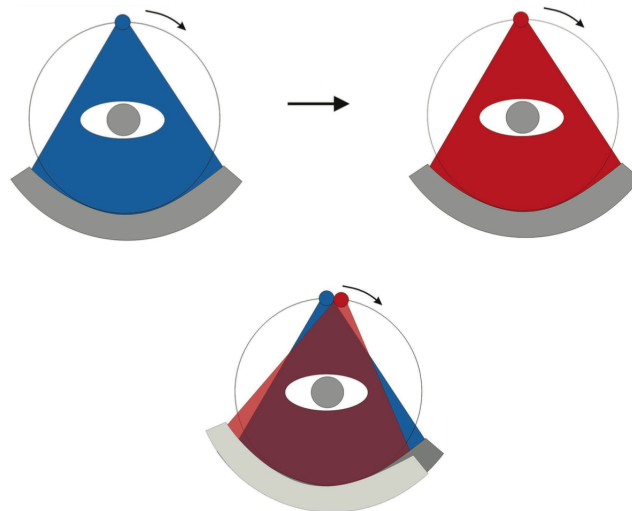


FIGURE 1.44 – En haut, exemple de commutation lente : on effectue une rotation complète en réglant la source à une haute tension donnée, puis on recommence après l'avoir modifiée. En bas, exemple de commutation rapide, la haute tension de la source est modifiée entre chaque rotation.

giques est l'approche **bi-énergie** ou *dual-energy*. Cette technique consiste à utiliser deux tubes à rayons X et deux détecteurs correspondants décalés d'environ 90 degrés comme cela est illustré en figure 1.45. Lorsque les deux tubes à rayons X fonctionnent à des tensions différentes, deux spectres de rayons X différents sont obtenus simultanément, ce qui donne lieu à une caractérisation de l'objet à double énergie, mais cela peut également augmenter significativement la dose de rayons X prise par le patient. En 2006, les premiers systèmes de scanner **bi-énergie** ont été commercialisés.

On trouvera dans [OMOUMI et collab., 2015] une comparaison entre les différentes méthodes précédemment citées. On trouvera aussi dans le chapitre 4 de [HEISMANN et collab., 2012] des informations plus détaillées sur ces méthodes.

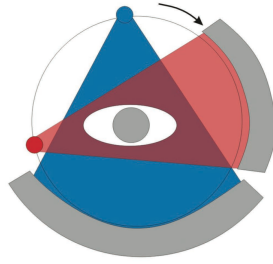


FIGURE 1.45 – Illustration de la méthode **bi-énergie** : l'approche de la **TDM** à double source fait appel à deux tubes à rayons X et deux détecteurs correspondants décalés angulairement. Les deux tubes à rayons X sont soumis à des hautes tensions différentes, ce qui permet une acquisition simultanée de données bi-énergétiques.

### Côté détecteur

Une autre méthode d'acquisition consiste à utiliser des détecteurs à scintillation conventionnels avec une structure en **double couche** (*sandwich*) (voir figure 1.46). Ces détecteurs permettent l'acquisition de données spectrales avec un seul spectre polychromatique de rayons X et évitent donc une surexposition du patient.

Dans ces détecteurs, la couche supérieure absorbe principalement les photons X d'énergie peu élevée, tandis que la couche inférieure détecte les photons X d'énergie plus élevée ayant traversé la première couche. Par conséquent, les signaux des deux couches de détecteurs correspondent à des spectres de rayons X différents avec des énergies moyennes différentes, ce qui permet de discriminer l'information.

Enfin il y a les détecteurs à comptage photons, tels que celui évoqué en 1.2.2.5, qui permettent

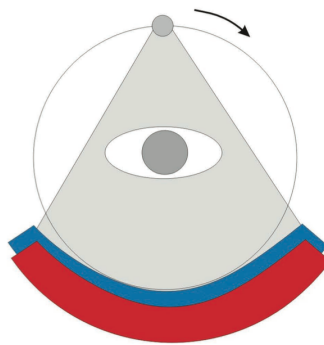


FIGURE 1.46 – Schéma de principe d'un détecteur à deux couches, qui permet l'acquisition de données spectrales à partir d'un seul spectre polychromatique.

d'acquérir des images à différentes énergies. C'est avec ces détecteurs que les données spectrales étudiées au cours de cette thèse ont été acquises.

L'électronique de ces détecteurs permet de discriminer la quantité d'énergie déposée par chaque photon incident. Un compteur est alors incrémenté lorsque l'énergie déposée est supérieure au seuil d'énergie de manière individuelle pour chaque pixel. Il est alors possible de régler un ensemble de pixels sur des énergies différentes de manière à estimer la part du spectre détectée dans des fenêtres d'énergie par cet ensemble de pixel formant un pixel composite de plus grande taille [DEBARBIEUX et collab., 2008].

Pour le moment ces détecteurs ne sont utilisés que dans un cadre pré-clinique sur le petit animal [ANDERSON et collab., 2010; PORTAL, 2018; ROESSL et collab., 2011], mais les avancées de cette technologie rythment le développement de la tomographie spectrale. Les avantages de tels détecteurs sont détaillés dans [DUPONT, 2014; HEISMANN et collab., 2012; MELZER et collab., 2008].

### 1.3.2.4 État de l'art reconstruction TDM spectrale

Les méthodes de traitement des données en TDM spectrale ont été classées en deux grandes catégories : **pré-reconstruction** et **post-reconstruction**.

#### Pré-reconstruction

Dans le traitement **pré-reconstruction** des données spectrales, on sépare d'abord les sinogrammes en sinogramme de concentration des composants de l'objet [ALVAREZ et MACOVSKI, 1976; ROESSL et PROKSA, 2007], puis n'importe quelle technique de reconstruction d'image peut être utilisée pour inverser ces sinogrammes et ainsi donner accès aux cartes spatiales de concentration des composants [DUPONT, 2014; SAWATZKY et collab., 2014; SCHIRRA et collab., 2013; SCHLOMKA et collab., 2008]. Cette méthode en deux étapes est sensible au bruit, en raison de l'instabilité de l'étape de séparation.

#### Post-reconstruction

Pour le traitement **post-reconstruction**, les données spectrales sont passées au logarithme afin d'obtenir des sinogrammes moyen d'une carte d'atténuation pour chaque fenêtre d'énergie suivie d'une reconstruction d'image utilisant un algorithme dédié à la TDM d'absorption classique [BRUNNER et collab., 2013; MAASS et collab., 2011]. L'avantage du traitement post-reconstruction est qu'il est relativement simple, parce qu'il ne s'agit que d'une petite modification de la façon dont les données en TDM d'absorption classique sont traitées. Une technique de post-reconstruction a été implémentée au CPPM : l'imagerie au **K-edge** par soustraction. Sa faisabilité a été démontrée dans [BRUNNER et collab., 2013], une explication détaillée se trouve ci-dessous.

#### Séparation et reconstruction simultanées

Il existe toutefois une troisième option pour le traitement des données spectrales : l'estimation **directe** des cartes de concentration. Il s'agit ici de séparer les cartes de concentration et de reconstruire les images de manière **simultanée**. Cette méthode présente l'avantage que le modèle de transmission spectrale est précis et que des contraintes sur les cartes de concentration de l'objet peuvent être incorporées directement dans le modèle. Les principales difficultés de l'approche sont dues à la présence de l'opérateur de projection  $\mathbf{S}$  à inverser, la grande dimension des matrices du modèle et le mauvais conditionnement du problème.

Actuellement, il existe quelques méthodes pour la reconstruction itérative en TDM spectrale qui tiennent compte de la complexité accrue du modèle de données. Dans [ELBAKRI et FESSLER, 2002], les auteurs minimisent la somme de deux fonctions : un terme d'attache aux données qui tient compte du bruit de Poisson et un terme de régularisation employant la fonction de Huber. Ils utilisent ensuite une approximation quadratique de la fonction coût puis un algorithme de type *expectation-maximization* (EM) appelé **OSEM** (*ordered subset expectation maximization*) calcule itérativement une solution. Des résultats de séparations sur des données simulées avec un objet à deux composants sont obtenues.

D'autres encore comme [CAIFANG et collab., 2013], utilisent une approche bayésienne. Pour cela, ils estiment la carte des coefficients de concentration de l'objet à l'aide de la méthode d'estimation conjointe du maximum *a posteriori* (MAP) en supposant que le bruit des données est gaussien et en imposant la régularisation donnée par la fonction de Huber. La reconstruction est obtenue en minimisant une approximation quadratique de la fonction coût à l'aide d'un algorithme de gradient conjugué monotone incorporant des pas de descente sous-optimaux.

Dans [LONG et FESSLER, 2014], les auteurs utilisent un terme d'attache aux données en supposant un bruit de Poisson et une régularisation employant une fonction hyperbolique qui conserve

les bords, ils obtiennent des résultats sur données simulées en minimisant des substituts quadratiques séparables de la fonction coût à l'aide d'un algorithme de minimisation séquentielle généralisée (**GSMO**).

Enfin plus récemment, une série de publications a montré des résultats intéressants. La première [**BARBER et collab., 2016**] présente une méthode basée sur la résolution du problème de **TDM** spectrale sans approximation. Pour cela les auteurs utilisent un algorithme **primal-dual** appelé **MOCCA** (*mirrored convex-concave algorithm*) en tenant compte d'une régularisation de type variation totale ou **TV** (*total variation*) et d'un bruit de type gaussien ou de Poisson. Des résultats sur données simulées sont obtenus. Dans la seconde publication, [**SCHMIDT et collab., 2017**] proposent d'ajouter un paramètre supplémentaire dans le modèle afin d'estimer un terme correctif sur l'intensité du spectre en énergie de la source polychromatique. Cette méthode, appelée **OC-CIRs**, a montré des améliorations en terme de qualité, car elle réduit les artefacts circulaires et les erreurs de séparation de composants. Enfin, [**SIDKY et collab., 2018**] proposent une méthode d'accélération de l'algorithme **MOCCA** basée sur un préconditionnement de la descente de gradient à l'aide de matrices diagonales par bloc. Cette méthode, initialement proposée dans [**CHAMBOLLE et POCK, 2011**] pour un contexte général, montre une nette accélération de la convergence de l'algorithme. Des résultats sur données simulées sont présentés.

### Imagerie au K-edge par soustraction

L'**imagerie au K-edge** utilise le fait que l'énergie de liaison de la couche K se présente pour certains agents de contraste comme l'iode, le baryum ou le gadolinium dans la gamme d'énergie discriminée par les détecteurs à comptage de photons, ceci a été illustré précédemment en figure 1.12. En utilisant alors la particularité de **seuil** d'énergie des détecteurs à comptage de photons XPAD3 (voir sous section 1.2.2), c'est-à-dire de ne compter que les photons qui ont déposé une énergie supérieure à un certain seuil, on peut estimer l'information spectrale nécessaire.

En toute généralité, on note  $E_K$  l'énergie de liaison de la couche K. On effectue alors trois acquisitions à trois seuils différents :  $E_K^+ = E_K + \Delta E$ ,  $E_K^- = E_K - \Delta E$  et à l'énergie de liaison  $E_K$  (illustration en figure 1.47).  $\Delta E$  est un paramètre réglable dont le choix est déterminant :

- s'il est trop petit, il diminuera la statistique de comptage, dégradera le rapport signal à bruit de l'estimation de la part du spectre détectée dans la fenêtre d'énergie obtenue par soustraction de deux mesures indépendantes et donc la qualité de la reconstruction,
- s'il est trop grand, l'effet du K-edge ne sera plus visible et l'agent de contraste ne pourra pas être identifié.

On trouvera dans [**HE et collab., 2012**] une méthode d'optimisation de ce paramètre. En appliquant le modèle de mesure (1.42) on obtient pour chaque seuil

$$I(E_K) = \int_{E_K}^{+\infty} F(E) e^{-\int_{E_p} \mu(l,E) dl} dE,$$

$$I(E_K^+) = \int_{E_K + \Delta E}^{+\infty} F(E) e^{-\int_{E_p} \mu(l,E) dl} dE,$$

$$I(E_K^-) = \int_{E_K - \Delta E}^{+\infty} F(E) e^{-\int_{E_p} \mu(l,E) dl} dE.$$

On soustrait alors ces quantités pour obtenir :

$$I^+ = I(E_K) - I(E_K^+) = \int_{E_K}^{E_K + \Delta E} F(E) e^{-\int_{E_p} \mu(l,E) dl} dE,$$

$$I^- = I(E_K^-) - I(E_K) = \int_{E_K - \Delta E}^{E_K} F(E) e^{-\int_{E_p} \mu(l,E) dl} dE,$$

On reconstruit ensuite les sinogrammes issus de  $I^-$  et  $I^+$  afin d'avoir les volumes  $\mu^-$  et  $\mu^+$ . Si  $\Delta E$  est correctement choisi, on aura alors  $\mu^-(l) < \mu^+(l)$  dans les voxels contenant de l'agent de contraste et  $\mu^-(l) > \mu^+(l)$  sinon. Ceci est basé sur l'hypothèse que l'algorithme de reconstruction utilisé, qui est un algorithme de **TDM** d'absorption classique, permet de reconstruire un **CAL** approximant la valeur moyenne du **CAL** sur la fenêtre d'énergie considérée.

La faisabilité de cette méthode a été démontrée dans [BRUNNER et collab., 2013] sur des données acquises à l'aide de la caméra XPAD3 précédemment cité en sous-section 1.2.2. Dans cette étude, des solutions d'iode (qui présentent toutes un K-edge à 33,2 keV) et d'argent (qui présentent toutes un K-edge à 25,5 keV) de différentes concentrations ont été identifiées et quantifiées. Le problème de cette méthode est l'amplification du bruit. En effet, à l'étape de soustraction pour obtenir les quantités  $I^+$  et  $I^-$  les variances du bruit de mesure s'additionnent du fait que le détecteur ne dispose qu'un seul seuil en énergie et que chaque mesure est statistiquement indépendante des autres. De plus, elle se limite au cas où l'on veut mettre en évidence un seul agent de contraste. Enfin, comme on s'appuie d'une part sur des valeurs moyennes du **CAL** et d'autre part, on sépare dans l'espace des sinogrammes, cette méthode peut souffrir d'une perte de sensibilité importante selon les cas de figure.

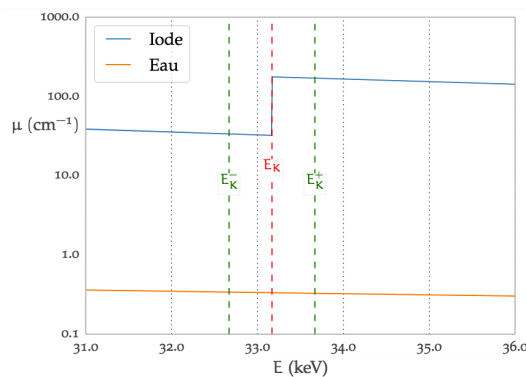


FIGURE 1.47 – Illustration de l'imagerie au K-edge par soustraction.

### 1.3.2.5 État de l'art algorithmique

Il est aussi intéressant de donner un bref aperçu des méthodes algorithmiques existantes pour la résolution de problèmes inverses. Certaines notions mathématiques vont être mentionnées ici sans détail. Elles seront expliquées au chapitre 2.

Il faut savoir que les techniques de base de minimisation lorsqu'on est confronté à une fonction différentiable sont basées sur des schémas itératifs visant à estimer progressivement un point qui annule le gradient de la fonction objectif. Le plus populaire est la **descente de gradient** : à partir d'un point initial, on construit une suite de points, chacun étant construit à partir du précédent en ajoutant un terme correctif proportionnel à l'opposé du gradient de la fonction objectif. Ce terme garantit que la valeur de la fonction objectif décroît.

Dans le cas d'une fonction convexe, il y a la garantie que lorsqu'on converge vers un point critique, celui-ci est un minimiseur global, donc une descente de gradient classique pourrait suffire. Dans le cas non-convexe, il n'y a aucune garantie de solution globale et les algorithmes tombent au mieux dans un minimum local.

Dans le cas non-convexe, les points critiques peuvent être des points selles ou maxima, il faudra donc en plus d'annuler le gradient, vérifier si le point trouvé est bien un minimum et s'il est global.

Malgré ces difficultés, on peut appliquer les méthodes simples de minimisation citées plus haut



telles que la **descente de gradient**, la **descente de gradient projeté**, la méthode du **gradient conjugué** [NOCEDAL et WRIGHT, 2006]. Les garanties de convergence ne sont alors que locales et sous condition d’initialiser l’algorithme au bon endroit (dans un bassin d’attraction du minimum qui doit être atteint) tandis que dans le cas convexe elles sont globales.

Une autre méthodologie consiste à utiliser à chaque itération une approximation de la fonctionnelle à minimiser qui aura les qualités suivantes :

1. elle est proche de la fonctionnelle autour du point courant,
2. elle est facile à minimiser.

Un exemple de ce type de méthode sont les algorithmes **majoration-minimisation (MM)** [CHOUZENOUX et collab., 2012; ERDOGAN et FESSLER, 1999] qui approchent la fonction coût par une fonction souvent quadratique, convexe, majorante et tangente au point courant. Le fait que l’approximation soit majorante garantit le bon comportement de l’algorithme. Un autre exemple est celui des méthodes de **région de confiance** [NOCEDAL et WRIGHT, 2006] qui approchent également la fonction par une fonction quadratique et tangente au point courant. Cette approximation n’est cette fois-ci pas obligatoirement majorante, mais on restreint la recherche d’un de ces minimiseurs à une région, appelée **région de confiance**, proche du point courant. C’est la présence de cette région qui garantit le bon comportement de l’algorithme. On trouvera plus de détails sur ces méthodes au chapitre 2.

Une autre méthodologie est d’utiliser le principe de **minimisation alternée** en subdivisant le problème en plusieurs sous-problèmes convexes et en résolvant ainsi ces problèmes de manière alternée, de façon à converger vers la solution du problème global. Un exemple type de ce genre de méthode est celui de la séparation **aveugle** de source où l’on cherche à trouver la matrice de mélange  $A$  et la matrice des sources  $S$  ne connaissant que leur mélange (ou produit)

$$X = AS.$$

On minimise par exemple  $\|X - AS\|_2^2$  en fixant  $A$  et en mettant à jour  $S$  et vis-versa.

De manière analogue, on peut parfois écrire la fonctionnelle à minimiser comme une différence de deux fonctions convexes  $f = f_1 - f_2$ , et alterner la minimisation sur  $f_1$  et celle sur  $f_2$  [HIRIART-URRUTY, 1985].

Parfois le problème ne peut être réécrit à l’identique comme un problème convexe. On peut choisir alors de le « rendre convexe », c’est-à-dire de le remplacer par un problème proche mais convexe. On appelle cette approche une **relaxation convexe**. L’exemple typique est celui de la relaxation  $l_1$  pour le problème de parcimonie  $l_0$ . Dans ce cas précis on peut montrer que sous certaines conditions, le problème relaxé a les mêmes minimiseurs que l’initial [CANDES et WAKIN, 2008]. En général la relaxation n’est pas nécessairement convexe, mais le problème relaxé est plus simple à résoudre.

Enfin on peut chercher à utiliser des techniques d’optimisation qui sont garanties dans le cas convexe. Récemment, dans la littérature beaucoup de travaux se sont intéressés à l’utilisation d’une classe de techniques pour l’optimisation convexe non-différentiable dans le cadre non-convexe. Un certain nombre **d’algorithmes proximaux** [BONETTINI et collab., 2017; CHOUZENOUX et collab., 2014] sont garantis théoriquement de converger vers un point critique de la fonction coût si celle-ci a une propriété supplémentaires appelée **propriété de Kurdyka-Łojasiewicz** [KURDYKA, 1998]. Ceci sera aussi discuté au chapitre 2.

## Conclusion du chapitre

Ce chapitre avait pour objectif d'introduire les principales notions permettant de comprendre et de situer le problème de **TDM** spectrale.

La section 1.1 a permis d'introduire la problématique générale et détaillé de la **TDM** spectrale et ainsi de comprendre ce que l'on souhaite faire dans ce travail de thèse. On y a expliqué le modèle que l'on étudie et le problème inverse que l'on souhaite résoudre.

On trouve en section 1.2 les éléments physiques qui permettent de comprendre le modèle d'acquisition. Ainsi on peut avoir une meilleure compréhension des phénomènes physiques qui interviennent dans la **TDM** spectrale. On y a expliqué aussi la tomographie et les principes de projection et rétroprojection qui sont à la base de l'acquisition tomodensitométrique.

Enfin, la section 1.3 a mis en lumière les enjeux de la **TDM** spectrale et donné aussi une perspective par rapport à l'état de l'art en **TDM**.

Ce chapitre a donc donné les outils nécessaires pour comprendre les enjeux et les difficultés liés au problème de **TDM** spectrale. A l'instar de ce chapitre, le chapitre 2 traitera des outils mathématiques qui nous ont permis de modéliser et traiter notre problème. On y introduira des définitions fondamentales d'analyse variationnelle et on donnera les propriétés essentielles qui nous permettront d'étudier et de résoudre un problème aussi complexe que celui de la **TDM** spectrale. On présentera aussi quelques algorithmes de résolution génériques.



## 1.4 Références

- R. E. ALVAREZ and A. L. MACOVSKI. 1976, «Energy-selective reconstructions in X-ray computerized tomography», *Phys. Med. Biol.*, vol. 21 5, p. 733–44. [58](#)
- G. F. AMELIO. 1975, «Charge-coupled devices for memory applications», dans *Proceedings of the May 19-22, 1975, National Computer Conference and Exposition, AFIPS '75*, ACM, New York, NY, USA, p. 515–522, doi :10.1145/1499949.1500053. URL <http://doi.acm.org/10.1145/1499949.1500053>. [13](#)
- A. ANDERSEN and A. KAK. 1984, «Simultaneous algebraic reconstruction technique (SART) : A superior implementation of the ART algorithm», *Ultrasonic Imaging*, vol. 6, n° 1, doi : 10.1016/0161-7346(84)90008-7, p. 81 – 94, ISSN 0161-7346. URL <http://www.sciencedirect.com/science/article/pii/0161734684900087>. [51](#)
- N. G. ANDERSON *et al.* 2010, «Spectroscopic (multi-energy) CT distinguishes iodine and barium contrast material in MICE», *Eur. Radiol.*, vol. 20, p. 2126–2134. [57](#)
- R. F. BARBER *et al.* 2016, «An algorithm for constrained one-step inversion of spectral CT data», *Phys. Med. Biol.*, vol. 61, n° 10, p. 3784. URL <http://stacks.iop.org/0031-9155/61/i=10/a=3784>. [59](#)
- S. BONETTINI *et al.* 2017, «On the convergence of a linesearch based proximal-gradient method for nonconvex optimization», *Inverse Probl.*, vol. 33, n° 5, p. 055 005. [61](#)
- S. BOYD and L. VANDENBERGHE. 2004, «Convex Optimization», Cambridge University Press, New York, NY, USA, ISBN 0521833787. [51](#)
- F. C. BRUNNER *et al.* 2013, «First K-edge imaging with a micro-CT based on the XPAD3 hybrid pixel detector», *IEEE T. Nucl. Sci.*, vol. 60, n° 1, p. 103–108, doi :10.1109/TNS.2012.2217753. [58](#), [60](#)
- P. P. BRUYANT . 2002, «Analytic and iterative reconstruction algorithms in SPECT», *J. Nucl. Med.*, vol. 43 10, p. 1343–58. [50](#), [52](#)
- C. CAI . 2013, «Tomographie par rayons X multi-énergétiques pour l’analyse de la structure interne de l’objet appliquée dans l’imagerie médicale», thèse de doctorat, Univ Paris 11. URL <http://www.theses.fr/2013PA112244>. [17](#), [45](#)
- C. CAIFANG *et al.* 2013, «A full-spectral Bayesian reconstruction approach based on the material decomposition model applied in dual-energy computed tomography», *Med. Phys.*, vol. 40, p. 111 916–111 931. URL <https://hal.archives-ouvertes.fr/hal-00911736>, doi:10.1118/1.4820478. [58](#)
- E. J. CANDÉS and M. B. WAKIN. 2008, «An introduction to compressive sampling», *IEEE Signal Proc. Mag.*, vol. 25, n° 2, p. 21–30, doi :10.1109/MSP.2007.914731. [61](#)
- A. CHAMBOLLE and T. POCK. 2011, «A first-order primal-dual algorithm for convex problems with applications to imaging», *J. Math. Imaging Vis.*, vol. 40, n° 1, p. 120–145, ISSN 1573-7683, doi : 10.1007/s10851-010-0251-1. [59](#)
- E. CHOUZENOUX, S. MOUSSAOUI and J. IDIER. 2012, «Majorize-minimize linesearch for inversion methods involving barrier function optimization», *Inverse Probl.*, vol. 28, n° 6, p. 065 011. URL <http://stacks.iop.org/0266-5611/28/i=6/a=065011>. [61](#)
- E. CHOUZENOUX, J.-C. PESQUET and A. REPETTI. 2014, «Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function», *J. Optimiz. Theory App.*, vol. 162, n° 1, p. 107–132, doi :10.1007/s10957-013-0465-7. [61](#)

- J. CHUNG, J. NAGY and I. SECHOPOULOS. 2010, «Numerical algorithms for polyenergetic digital breast tomosynthesis reconstruction», *SIAM J. Imaging Sci.*, vol. 3, n° 1, doi :10.1137/090749633, p. 133–152. URL <https://doi.org/10.1137/090749633>.
- F. DEBARBIEUX, P. DELPIERRE, C. MEESEN and C. MOREL. 2008, «Dispositif d'imagerie par rayons X a source poly-chromatique», FR2007/054974, PCT/FR2008/050807, WO/2008/149003. 57
- P. DELPIERRE. 2014, «A history of hybrid pixel detectors, from high energy physics to medical imaging», *J. Instrum.*, vol. 9, n° 05, p. C05059. URL <http://stacks.iop.org/1748-0221/9/i=05/a=C05059>. 32
- L. DESBAT, S. ROUX, P. GRANGEAT and A. KOENIG. 2004, «Sampling conditions of 3D parallel and fan-beam X-ray CT with application to helical tomography», *Phys. Med. Biol.*, vol. 49 11. 46
- Y. DU, G. YU, X. XIANG and X. WANG. 2017, «GPU accelerated voxel-driven forward projection for iterative reconstruction of cone-beam CT», *BioMed. Eng. OnLine*, vol. 16, n° 1, doi :10.1186/s12938-016-0293-8, p. 2. URL <https://doi.org/10.1186/s12938-016-0293-8>. 48
- M. DUPONT. 2014, «Tomographie spectrale à comptage de photons : développement du prototype PIXSCAN et preuve de concept», thèse de doctorat, Aix-Marseille Univ. URL <http://www.theses.fr/2014AIXM4011/document>. 29, 34, 52, 57, 58
- M. DUPONT *et al.* 2013, «Component separation for spectral X-ray imaging using the XPAD3 hybrid pixel camera», *Conf. Rec. NSS/MIC'2013*, Oct 27-Nov 2 2013, Seoul, Korea, IEEE Press.
- I. A. ELBAKRI and J. A. FESSLER. 2002, «Statistical image reconstruction for polyenergetic X-ray computed tomography», *IEEE T. Med. Imaging*, vol. 21, n° 2, p. 89–99, doi :10.1109/42.993128. 58
- H. ERDOGAN and J. A. FESSLER. 1999, «Monotonic algorithms for transmission tomography», *IEEE T. Med. Imaging*, vol. 18, p. 801–814. 61
- L. A. FELDKAMP, L. C. DAVIS and J. W. KRESS. 1984, «Practical cone-beam algorithm», *J. Opt. Soc. Am. A*, vol. 1, n° 6, p. 612–619. URL <http://josaa.osa.org/abstract.cfm?URI=josaa-1-6-612>, doi :10.1364/JOSAA.1.000612. 43
- T. G. FLOHR *et al.* 2006, «First performance evaluation of a dual-source CT (DSCT) system», *Eur. Radiol.*, vol. 16, n° 2, p. 256–268. 14
- R. GORDON. 1974, «A tutorial on ART (algebraic reconstruction techniques)», *IEEE Tr. Nucl. Sci.*, vol. 21, n° 3, p. 78–93, ISSN 0018-9499, doi :10.1109/TNS.1974.6499238. 51
- M. HAMONET. 2016, «Tomographie hybride simultanée TEP/TDM combinant détecteurs à pixels hybrides et modules phoswich à scintillateurs», thèse de doctorat, Aix-Marseille Univ. URL <http://www.theses.fr/2016AIXM4012/document>. 28
- M. HAMONET *et al.* 2015, «The ClearPET/XPAD prototype : development of a simultaneous PET/CT scanner for mice», *Conf. Rec. NSS/MIC'2015*, Oct 31-Nov 7 2015, San Diego, California, USA, IEEE Press.
- P. HE, B. WEI, W. CONG and G. WANG. 2012, «Optimization of K-edge imaging with spectral CT», *Med. I Phys.*, vol. 39, n° 11, doi :10.1118/1.4754587, p. 6572–6579. URL <https://aapm.onlinelibrary.wiley.com/doi/abs/10.1118/1.4754587>. 59
- B. HEISMANN, B. SCHMIDT and T. FLOHR. 2012, «Spectral Computed Tomography», SPIE press monograph, ISBN 9780819492579. 53, 55, 56, 57

- G. T. HERMAN. 1981, «Three-dimensional imaging from tomograms», dans «Digital Image Processing in Medicine», édité par K. H. Höhne, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 93–118. [43](#)
- J.-B. HIRIART-URRUTY. 1985, «Generalized differentiability / duality and optimization for problems dealing with differences of convex functions», dans *Convexity and Duality in Optimization*, édité par J. Ponstein, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-642-45610-7, p. 37–70. [61](#)
- G. N. HOUNSFIELD. 1973, «Computerized transverse axial scanning (tomography) : Part 1. description of system», *Brit. J. Radiol.*, vol. 46, n° 552, doi :10.1259/0007-1285-46-552-1016, p. 1016–1022. URL <https://doi.org/10.1259/0007-1285-46-552-1016>, pMID : 4757352. [55](#)
- H. M. HUDSON and R. S. LARKIN. 1994, «Accelerated image reconstruction using ordered subsets of projection data», *IEEE T. Med. Imaging*, vol. 13, n° 4, p. 601–609, ISSN 0278-0062, doi :10.1109/42.363108. [52](#)
- J. S. IWANCZYK *et al.* 2009, «Photon counting energy dispersive detector arrays for X-ray imaging», *IEEE T. Nucl. Sci.*, vol. 56, n° 3, p. 535–542, ISSN 0018-9499, doi :10.1109/TNS.2009.2013709. [27](#)
- M. JIANG and G. WANG. 2003, «Convergence of the simultaneous algebraic reconstruction technique (SART)», *IEEE T. Imag. Process.*, vol. 12, n° 8, p. 957–961, ISSN 1057-7149, doi : 10.1109/TIP.2003.815295. [52](#)
- T. R. C. JOHNSON *et al.* 2007, «Material differentiation by dual energy CT : initial experience», *Eur. Radiol.*, vol. 17, n° 6, p. 1510–1517, doi :10.1007/s00330-006-0517-6. [27](#)
- A. C. KAK and M. SLANEY. 2001, «Principles of computerized tomographic imaging», Society for Industrial and Applied Mathematics, Philadelphia, ISBN 089871494X 9780898714944. [51](#)
- W. A. KALENDER, W. H. PERMAN, J. R. VETTER and E. KLOTZ. 1986, «Evaluation of a prototype dual-energy computed tomographic apparatus. i. phantom studies», *Med. Phys.*, vol. 13, n° 3, doi :10.1118/1.595958, p. 334–339. URL <https://aapm.onlinelibrary.wiley.com/doi/abs/10.1118/1.595958>. [56](#)
- KRONLAND-MARTINET, C. 2015, «Développement de la tomographie intra-vitale au K-edge avec la camera à pixels hybrides XPAD3», thèse de doctorat, Aix-Marseille Univ. URL <http://www.theses.fr/2015AIXM4045>. [28](#)
- C. KRONLAND-MARTINET *et al.* 2014, «Development of K-edge spectral tomography using XPAD3 composite pixels», *Conf. Rec. NSS/MIC'2014*, Nov 8-15 2014, Seattle, Washington State, USA, IEEE Press.
- KURDYKA, K. 1998, «On gradients of functions definable in o-minimal structures», *Annales de l'institut Fourier*, vol. 48, n° 3, p. 769–783, doi :10.5802/aif.1638. [61](#)
- K. LANGE and J. A. FESSLER. 1995, «Globally convergent algorithms for maximum a posteriori transmission tomography», *IEEE T. Imag. Process.*, vol. 4, n° 10, p. 1430–1438, ISSN 1057-7149. [52](#)
- L. LI, *et al.* 2011, «A curve-filtered FDK (C-FDK) reconstruction algorithm for circular cone-beam CT», *J. X-ray Sci. Technol.*, vol. 19 3, p. 355–71. [44](#), [45](#)
- Y. LONG and J. A. FESSLER. 2014, «Multi-material decomposition using statistical image reconstruction for spectral CT», *IEEE T. Med. Imaging*, vol. 33, n° 8, p. 1614–1626, doi :10.1109/TMI.2014.2320284. [58](#)

- C. MAASS, E. MEYER and M. KACHELRIESS. 2011, «Exact dual energy material decomposition from inconsistent rays (MDIR)», *Med. Phys.*, vol. 38, n° 2, p. 691–700, doi :10.1118/1.3533686. 58
- K. MATHIESON *et al.* 2002, «Charge sharing in silicon pixel detectors», *Nucl. Instrum. Meth. A*, vol. 487, n° 1, p. 113 – 122, ISSN 0168-9002. 3rd International Workshop on Radiation Imaging Detectors, doi :10.1016/S0168-9002(02)00954-3. 34
- T. R. MELZER *et al.* 2008, «Spectroscopic biomedical imaging with the Medipix2 detector», *Australas. Phys. Eng. S.*, vol. 31, n° 4, p. 300, ISSN 1879-5447, doi :10.1007/BF03178599. 57
- F. NATTERER 1993, «SIAM J. Appl. Math.», *SIAM Journal on Applied Mathematics*, vol. 53, n° 2, p. 358–380, ISSN 00361399. URL <http://www.jstor.org/stable/2102341>. 46
- J. NOCEDAL and S. J. WRIGHT. 2006, «Numerical Optimization», 2<sup>e</sup> éd., Springer, New York, NY, USA. 61
- P. OMOUMI *et al.* 2015, «Dual-energy CT : Basic principles, technical approaches, and applications in musculoskeletal imaging (part 1)», *Semin Musculoskelet Radiol*, vol. 19, n° 05, p. 431–437, doi :10.1055/s-0035-1569253. 56
- H. OUAMARA. 2013, «Comparaison de la micro-tomodensitométrie par comptage de photons et par intégration de charges avec le dispositif d'irradiation PIXSCAN», thèse de doctorat, Aix-Marseille Univ. URL <http://www.theses.fr/2013AIXM4004>. 13
- H. OUAMARA *et al.* 2012, «Comparison of the performance of the photon counting hybrid pixel camera XPAD3 versus the CCD camera DALSA XR-4 for cone-beam micro-CT», *Conf. Rec. NSS/MIC'2012*, Oct 29-Nov 3 2012, Anaheim, California, USA, IEEE Press, pp. 3756-3759.
- L. PORTAL. 2018, «Etude de la tomographie à comptage de rayons X avec des pixels hybrides en Si et en CdTe et application au suivi longitudinal du carcinome hépatocellulaire chez la souris», thèse de doctorat, Aix-Marseille Univ. 33, 57
- G. PRATX and L. XING. 2011, «GPU computing in medical physics : a review.», *Med. Phys.*, vol. 38 5, p. 2685–97. 48, 49
- J. RADON. 1917, «Über die bestimmung von funktionen durch ihre integralwerte längs gewisser», *Mannigfaltigkeiten. Ber. Verh. Sächs. Akad. Wiss.*, n° 69, p. 262–277. 37
- S. RIT *et al.* 2014, «The reconstruction toolkit (RTK), an open-source cone-beam CT reconstruction toolkit based on the insight toolkit (ITK)», *International Conference on the Use of Computers in Radiation Therapy (ICCR) 2013, May 2013, Melbourne, Australia*, vol. 489, n° 1, p. 012 079. URL <http://stacks.iop.org/1742-6596/489/i=1/a=012079>. 49
- E. ROESSL, *et al.* 2011, «Preclinical spectral computed tomography of gold nano-particles», *Nucl. Instrum. Meth. Phys. A.*, vol. 648, p. 259–264, doi :10.1016/j.nima.2010.11.072. 57
- E. ROESSL and R. PROKSA. 2007, «K-edge imaging in X-ray computed tomography using multi-bin photon counting detectors.», *Phys. Med. Biol.*, vol. 52 15, p. 4679–96. 58
- A. SAWATZKY *et al.* 2014, «Proximal ADMM for multi-channel image reconstruction in spectral X-ray CT», *IEEE T. Med. Imaging*, vol. 33, n° 8, p. 1657–1668, doi :10.1109/TMI.2014.2321098. 58
- C. O. SCHIRRA *et al.* 2013, «Statistical reconstruction of material decomposed data in spectral CT», *IEEE T. Med. Imaging*, vol. 32, n° 7, p. 1249–1257, doi :10.1109/TMI.2013.2250991. 58
- J. P. SCHLOMKA *et al.* 2008, «Experimental feasibility of multi-energy photon-counting K-edge imaging in pre-clinical computed tomography», *Phys. Med. Biol.*, vol. 53, n° 15, p. 4031. URL <http://stacks.iop.org/0031-9155/53/i=15/a=002>. 58

- T. G. SCHMIDT, R. F. BARBER and E. Y. SIDKY. 2017, «A spectral CT method to directly estimate basis material maps from experimental photon-counting data», *IEEE T. Med. Imaging*, vol. 36, n° 9, p. 1808–1819, doi :10.1109/TMI.2017.2696338. 59
- P. M. SHIKHALIEV. 2005, «Beam hardening artefacts in computed tomography with photon counting, charge integrating and energy weighting detectors : a simulation study», *Phys. Med. Biol.*, vol. 50, n° 24, p. 5813. URL <http://stacks.iop.org/0031-9155/50/i=24/a=004>. 13
- P. M SHIKHALIEV, T. XU and S. MOLLOI. 2005, «Photon counting computed tomography : Concept and initial results», *Med. Phys.*, vol. 32, n° 2, p. 427–436, doi :10.1118/1.1854779. 13
- E. SIDKY *et al.* 2018, «Three material decomposition for spectral computed tomography enabled by block-diagonal step-preconditioning», The Fifth International Conference on Image Formation in X-Ray Computed Tomography, May 20-23, 2018, Salt Lake City. 59
- P. TAKOUKAM TALLA. 2011, «Investigation of photon counting pixel detectors for X-ray spectroscopy and imaging», thèse de doctorat, Friedrich Alexander Univ. Erlangen-Nurnberg. 34
- A. N. TIKHONOV. 1963, «Solution of incorrectly formulated problems and the regularization method», *Soviet Math. Dokl.*, vol. 4, p. 1035–1038. URL <https://www.bibsonomy.org/bibtex/260c5508ba257aedce141c6ca10b38ec0/brefeld>. 50
- J. K VAN ABBEMA *et al.* 2012, «Feasibility and accuracy of tissue characterization with dual source computed tomography», *Phys. Medica.*, vol. 28, n° 1, p. 25–32. 14
- Z. XING *et al.* 2013, «An accelerated ramla reconstruction algorithm for X-ray cone-beam CT», *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 55, n° 5, p. 237–242, doi :10.1784/insi.2012.55.5.237. 52
- F. XU and K. MUELLER. 2006, «A comparative study of popular interpolation and integration methods for use in computed tomography», dans *3rd IEEE International Symposium on Biomedical Imaging : Nano to Macro, 2006.*, ISSN 1945-7928, p. 1252–1255, doi :10.1109/ISBI.2006.1625152. 49
- G. ZENG. 2010, *Medical image reconstruction: A conceptual tutorial*, Springer Berlin Heidelberg, ISBN 9783642053672, doi :10.1007/978-3-642-05368-9. 43
- X. ZHAO, J. JING HU and P. ZHANG. 2009, «GPU-based 3D cone-beam ct image reconstruction for large data volume», *Int. J. Biomed. Imaging*, doi :10.1155/2009/149079. 48



# Chapitre 2

## Notations et généralités sur les problèmes inverses

*« ...If you can meet with Triumph  
and Disaster  
And treat those two impostors just  
the same...  
... you'll be a man ... »*

Rudyard Kipling

### Sommaire

---

<b>2.1 Problème inverse</b> . . . . .	<b>70</b>
2.1.1 Généralités . . . . .	70
2.1.2 Optimisation . . . . .	72
2.1.3 Notations et définitions . . . . .	72
<b>2.2 Outils mathématiques</b> . . . . .	<b>76</b>
2.2.1 Notion de sous-différentiel . . . . .	77
2.2.2 Opérateurs proximaux . . . . .	80
2.2.3 Inégalité de Kurdyka-Łojasiewicz . . . . .	84
<b>2.3 Définitions des fonctions objectifs</b> . . . . .	<b>86</b>
2.3.1 Fonction d'attache aux données . . . . .	87
2.3.2 Fonction de régularisation . . . . .	88
<b>2.4 Algorithmes d'optimisation</b> . . . . .	<b>91</b>
2.4.1 Algorithmes à direction de descente . . . . .	92
2.4.2 Algorithmes de région de confiance . . . . .	95
2.4.3 Algorithmes proximaux . . . . .	100
<b>2.5 Références</b> . . . . .	<b>110</b>

---



## Introduction

Dans ce chapitre nous introduisons brièvement les problèmes inverses et les outils fondamentaux d'analyse et d'optimisation qui ont été utilisés durant cette thèse pour résoudre notre problème de **TDM** spectrale. Dans le chapitre 1 nous avons formulé ce problème de la manière suivante :

$$\hat{\mathbf{A}} = \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\operatorname{argmin}} \{J(\mathbf{A}) = G(\mathbf{A}) + R(\mathbf{A})\} \quad (2.1)$$

avec  $G(\mathbf{A})$  le terme **d'attache aux données** et  $R(\mathbf{A})$  le terme de **régularisation**. L'objectif de ce chapitre est de présenter les différents outils de résolution que nous avons utilisés pour résoudre ce problème.

Ainsi ce chapitre sera organisé de la manière suivante :

- la première section introduit de manière générale les problèmes inverses ainsi que les notations et définitions mathématiques relatives à cette branche, ceci permettra de situer le problème de **TDM** spectrale dans le contexte mathématique ,
- la seconde section traite des outils mathématiques d'analyse variationnelle qui nous ont permis de traiter le problème de **TDM** spectrale,
- on définira dans la troisième section la formulation de ce problème selon différentes hypothèses,
- nous exposerons en dernière sections quelques algorithmes de résolution de problèmes inverses ainsi que ceux utilisés durant cette thèse.

## 2.1 Problème inverse

### 2.1.1 Généralités

Un problème inverse peut être défini d'un point de vue « expérimental » (ou physique) comme une situation où l'on souhaite évaluer une certaine grandeur  $\mathbf{X}$  (**inconnue**) inaccessible à partir de la mesure d'une autre grandeur  $\mathbf{b}$  (**observation**) directement accessible connaissant un modèle mathématique du problème qui donne, de manière implicite,  $\mathbf{X}$  à partir de l'observation  $\mathbf{b}$  [BONNET, 2008; COHEN, 2000]. On note généralement ce problème

$$\mathbf{b} = \varphi(\mathbf{X}) \quad (2.2)$$

Un problème inverse peut être « bien posé » lorsqu'il admet une solution unique  $\mathbf{X}$  continue par rapport à  $\mathbf{b}$  [HADAMARD, 1902]. Il est en revanche « mal posé » lorsque l'existence, l'unicité et/ou la continuité de la solution par rapport à l'observation  $\mathbf{b}$  ne sont pas toutes vérifiées, un exemple étant notre problème (1.13).

La non-unicité est un problème sérieux car si un problème admet plusieurs solutions, il faut un moyen de choisir l'une d'entre elles. Du point de vue physique, cela signifie qu'une observation  $\mathbf{b}$ , compte tenu de la plage d'incertitude qui l'accompagne, peut correspondre à un grand nombre de valeurs de  $\mathbf{X}$ , ces dernières pouvant être fort éloignées les unes des autres. Pour faire un choix correct, il faut disposer d'informations supplémentaires sur  $\mathbf{X}$ , informations dites *a priori*.

Par ailleurs, le manque de continuité est sans doute le plus problématique, en particulier en vue d'une résolution approchée ou numérique. Cela veut dire qu'il ne sera pas possible, **indépendamment** de la méthode numérique, d'approcher de façon satisfaisante la solution du problème inverse, puisque les données disponibles seront bruitées donc différentes des véritables données. Un des exemples les plus courants est le problème de **déconvolution** [BIRAUD, 1976].

Le caractère mal posé des problèmes a longtemps été ignoré car il était considéré comme dénué



de sens physique ou reflétait une modélisation inadéquate. Aujourd’hui ce caractère fondamental est reconnu et motive de nombreuses recherches en mathématique. On peut lister les causes d’incertitude d’un problème comme suit :

- les données ont une origine expérimentale, ce qui implique l’existence d’erreurs de mesure (bruit),
- elles sont collectées en nombre fini (discret), même si, dans le modèle mathématique, elles sont décrites par des fonctions continues,
- le modèle lui-même provient d’une « idéalisation » de la réalité physique et repose sur des hypothèses simplificatrices, il est donc également une source d’incertitudes. De plus, certains paramètres du modèle ne sont connus que de manière expérimentale, donc approximative,
- même les méthodes d’inversion du problème peuvent parfois créer une altération des données (erreur dues à l’interpolation par exemple).

Les problèmes inverses sont sensibles à ces incertitudes et cela impacte la recherche de solutions. En effet, tout  $\mathbf{X}$  qui reproduit, aux incertitudes près, l’observation  $\mathbf{b}$  via le modèle est une solution possible au problème. Par conséquent le caractère incomplet, imprécis et/ou redondant des données doit être pris en compte dans le modèle.

L’approche idéale consisterait à inventorier l’ensemble complet des solutions (2.2), parmi lesquelles on opérerait ensuite un choix suivant des critères additionnels (vraisemblance physique, informations *a priori* supplémentaires) afin de retenir la ou les solutions jugées justes. Malheureusement, une telle approche pose des problèmes pratiques insurmontables si le nombre de variables à identifier est grand, ce qui est le cas dans la majorité des problèmes actuels.

En pratique, les méthodes de résolution des problèmes inverses se répartissent selon Nashed [NASHED, 1974] en trois catégories principales :

1. Les méthodes relevant de **l’analyse mathématique et de la théorie des fonctions**. Elles proposent de transformer un problème mal posé en un problème bien posé en jouant sur le choix des espaces qui servent à décrire les variables.
2. Les **méthodes d’inversion stochastique** [TARANTOLA, 2005] dont le principe est de considérer toutes les variables comme aléatoires afin de représenter toutes les incertitudes. La solution du problème inverse serait alors la fonction densité de probabilité associée à l’inconnue  $\mathbf{X}$  et l’observation  $\mathbf{b}$ .
3. Les **méthodes de régularisation** des problèmes mal posé [TIKHONOV et ARSENIN, 1977]. Il s’agit ici de redéfinir les notions d’inversion et de solution de sorte que la solution régularisée obtenue par ces méthodes dépende continûment des données et soit proche de la solution exacte (si celle-ci existe).

Les méthodes 1 et 2 ne seront pas abordées dans ce manuscrit, mais on trouvera dans [MENKE, 2012; TARANTOLA, 2005] une documentation détaillée sur ce type de méthodes. C’est dans la troisième catégorie que s’inscrivent les méthodes utilisées durant ce travail de thèse.

La démarche de ces méthodes va consister à trouver (ou approximer) au mieux les paramètres qui permettent d’avoir les mesures observées. Cette résolution peut se faire de différentes manières :

- par **simulation numérique** en utilisant par exemple des programmes informatiques afin de simuler un phénomène physique [LE MARÉCHAL et COQUIDÉ, 2006],
- ou de manière **analytique**, en utilisant des outils mathématiques afin d’étudier les fonctions qui modélisent le phénomène étudié, c’est le cas de **l’optimisation**.

Dans cette thèse nous avons développé des méthodes d’optimisation dont nous allons en donner une brève introduction générale dans le paragraphe qui suit.

### 2.1.2 Optimisation

D'un point de vue mathématique, résoudre un problème inverse revient en général à optimiser une fonction. L'**optimisation** est la **minimisation** ou **maximisation**, sous contraintes ou non, de cette fonction [BONNANS et collab., 2003]. On note le problème d'optimisation

$$\begin{aligned} & \underset{\mathbf{X} \in \mathbb{R}^n}{\operatorname{argmin}} f(\mathbf{X}) \\ & \text{sous } c(\mathbf{X}) \leq 0, \end{aligned} \tag{2.3}$$

avec  $f$  la **fonction objectif** qui prend en compte le modèle du problème,  $\mathbf{X}$  le vecteur des variables inconnues que l'on souhaite trouver et  $c$  la fonction représentant des contraintes que  $\mathbf{X}$  doit satisfaire.

L'optimisation est découpée en plusieurs catégories qui se chevauchent (optimisation linéaire ou non linéaire, convexe ou non convexe, stochastique, etc.). Ces différents types d'optimisation dépendent de la forme de la fonction objectif  $f$  et celle des contraintes  $c$ . Il est important de connaître la catégorie associée au problème afin d'élaborer une méthode de résolution qui soit applicable au sens numérique.

Ainsi, une des méthodes pour résoudre le problème (2.3) consiste à utiliser des **algorithmes d'optimisation itératifs** choisis selon le type de problème étudié. Le schéma de ces algorithmes commence par une estimation initiale de valeur optimale de l'inconnue  $\mathbf{X}$ , que l'on note  $\mathbf{X}_0$ , et génère une séquence d'estimations améliorées jusqu'à ce qu'il atteigne une solution satisfaisante selon un ou plusieurs critères.

La stratégie utilisée pour passer d'une itération à l'autre distingue un algorithme d'un autre. La plupart des stratégies utilisent les valeurs de la fonction objectif  $f$ , la contrainte  $c$ , et éventuellement les dérivées première et seconde de ces fonctions. Indépendamment de ces spécificités, les propriétés suivantes sont désirables :

- **robustesse** : ils doivent être performants sur toute une variété de problèmes dans leur catégorie, pour tous les choix raisonnables des variables initiales  $\mathbf{X}_0$ ,
- **efficacité** : ils ne devraient pas nécessiter trop de temps de calcul ou solliciter beaucoup de mémoire,
- **précision** : ils devraient être capables d'identifier une solution avec précision, sans être trop sensibles aux erreurs des données ou aux erreurs d'approximation.

Il est très difficile de trouver un algorithme satisfaisant ces trois propriétés simultanément. Souvent un compromis doit être fait entre vitesse de convergence et capacité de mémoire, entre robustesse et rapidité, etc. Ceci a fait l'objet de recherche centrale durant cette thèse, on trouvera plus de détails au chapitre 3.

Les algorithmes d'optimisation les plus rapides ne recherchent qu'une solution **locale**, un point où la fonction objectif est plus petite (*resp.* plus grande) que tous les autres points possibles à proximité. Ils ne trouvent pas toujours le meilleur de tous ces minima (*resp.* maxima), c'est-à-dire la solution **globale** même lorsque celle-ci existe (voir définition en 2.1.3.2). Ces solutions globales sont généralement difficiles à identifier et encore plus difficiles à localiser.

Il existe des cas où les solutions trouvées sont des solutions globales, par exemple lorsque la fonction objectif est **convexe** (voir 2.1.3.1 ci-dessous). Toutefois, les problèmes dans les cas pratiques, qu'ils soient contraints ou non, n'ont pas toujours cette spécificité comme c'est le cas pour notre problème (2.1).

### 2.1.3 Notations et définitions

Avant d'entrer dans les détails, il est important de poser le contexte mathématique de notre problème et donner quelques définitions et notions fondamentales qui interviennent en optimisation.

Soit  $\mathcal{H}$  un espace de Hilbert réel muni d'un produit scalaire noté  $\langle \cdot, \cdot \rangle$  et d'une norme notée  $\| \cdot \| = \sqrt{\langle \cdot, \cdot \rangle}$ . Dans le cas où  $\mathcal{H} = \mathbb{R}^n$  on note  $\mathbb{S}_n := \mathbb{S}(\mathbb{R}^n)$  l'ensemble des matrices **symétriques** de  $\mathbb{R}^{n \times n}$ , et  $\mathbb{S}_n^+ := \mathbb{S}^+(\mathbb{R}^n)$  l'ensemble des matrices **symétriques définies positives** de  $\mathbb{R}^{n \times n}$ , i.e. les matrices  $M$  vérifiant

$$\forall x \in \mathcal{H} \ x \neq 0, \ x^t M x > 0 \quad \text{et} \quad M = M^t. \quad (2.4)$$

De plus, notons  $\mathbf{Id}$  l'**opérateur identité** de  $\mathcal{H}$  et  $\mathbf{I}_n$  la **matrice identité** de  $\mathbb{R}^{n \times n}$ . Dans la suite nous poserons  $\mathcal{H} = \mathbb{R}^n$ .

### Définition 2.1.3.1: définitions de base

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ .

► Le **domaine** de  $f$  est défini par

$$D_f := \{x \in \mathbb{R}^n \mid f(x) < +\infty\}. \quad (2.5)$$

On dit que  $f$  est **propre** si  $D_f$  est non vide.

► La fonction  $f$  est dite **coercive** si

$$\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty. \quad (2.6)$$

► La fonction  $f$  est **semi-continue inférieurement** (s.c.i.) en  $x_0 \in \mathbb{R}^n$  si

$$\liminf_{x \rightarrow x_0} f(x) \geq f(x_0). \quad (2.7)$$

► On note  $\leq$  l'ordre partiel de **Loewner** défini comme suit :

$$\forall U_1, U_2 \in \mathbb{S}_N \quad U_1 \leq U_2 \Leftrightarrow \langle x, U_1 x \rangle \leq \langle x, U_2 x \rangle \quad \forall x \in \mathbb{R}^n. \quad (2.8)$$

► Nous noterons également le produit scalaire et la norme pondérés, relativement à  $U \in \mathbb{S}_N^+$ , de la façon suivante :

$$\forall x, y \in \mathbb{R}^n \quad \langle x, y \rangle_U := \langle x, U y \rangle = x^t U y \quad \text{et} \quad \|x\|_U := \sqrt{\langle x, U x \rangle}. \quad (2.9)$$

► Le **conditionnement** d'une matrice inversible  $M$  relativement à la norme spectrale (correspondant à sa plus grande valeur singulière) notée  $\| \cdot \|$  est défini par la formule :

$$\kappa_M = \|M^{-1}\| \|M\|. \quad (2.10)$$

Le conditionnement donne une borne de l'erreur relative commise. C'est la mesure de la dépendance de la solution d'un problème numérique par rapport aux données du problème. Lorsque  $\kappa_M$  est bas, on dit que  $M$  est **bien conditionnée**. Si au contraire il est élevé on dit que  $M$  est **mal conditionnée**.

► Un ensemble  $D \subset \mathbb{R}^n$  est **fermé** si et seulement si toute limite (dans  $\mathbb{R}^n$ ) d'une suite à valeurs dans  $D$  appartient à  $D$ . Par ailleurs, tout **point d'accumulation** de  $D$  est un élément de  $D$ .

#### 2.1.3.1 (Non) Convexité

Afin de comprendre la notion de non convexité il faut d'abord aborder le concept de convexité car celui-ci est fondamental dans l'optimisation et donne plusieurs informations sur le problème

que l'on souhaite résoudre. Le terme convexe peut s'appliquer aussi bien aux ensembles qu'aux fonctions.

### Définition 2.1.3.2: convexité d'un ensemble et d'une fonction

Soit  $C$  un sous-ensemble de  $\mathbb{R}^n$ . L'ensemble  $C$  est **convexe** si le segment de ligne droite reliant deux points en  $C$  se trouve entièrement à l'intérieur de  $C$ . Formellement, pour deux points  $x \in C$  et  $y \in C$ , nous avons

$$\lambda x + (1 - \lambda)y \in C, \forall \lambda \in [0, 1]. \quad (2.11)$$

De même qu'une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  est convexe si son **domaine de définition**  $D_f$  est un ensemble convexe et si son graphe, noté

$$\mathbf{Graph}_f := \{(x, y) \in \mathbb{R}^n \mid y = f(x)\} \quad (2.12)$$

est en dessous de toute droite reliant deux points quelconques de ce graphe. Formellement, si on prend  $(x, f(x)) \in \mathbf{Graph}_f$  et  $(y, f(y)) \in \mathbf{Graph}_f$  alors

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y), \forall \lambda \in [0, 1]. \quad (2.13)$$

Par ailleurs, on dit que  $f$  est une fonction **concave** si  $-f$  est convexe. Une illustration d'une fonction convexe est donné en figure 2.1 pour le cas unidimensionnel.

### Illustration de la convexité

La convexité nous permet de faire des affirmations plus fortes sur la convergence des algorithmes. Le principal avantage des fonctions convexes est l'assurance que si l'on converge vers un **point critique**, *i.e.* un point d'annulation du gradient de la fonction, ce point est un minimum (ou maximum) **global**, il n'y pas de minima (maxima) local ou de **point selle**.

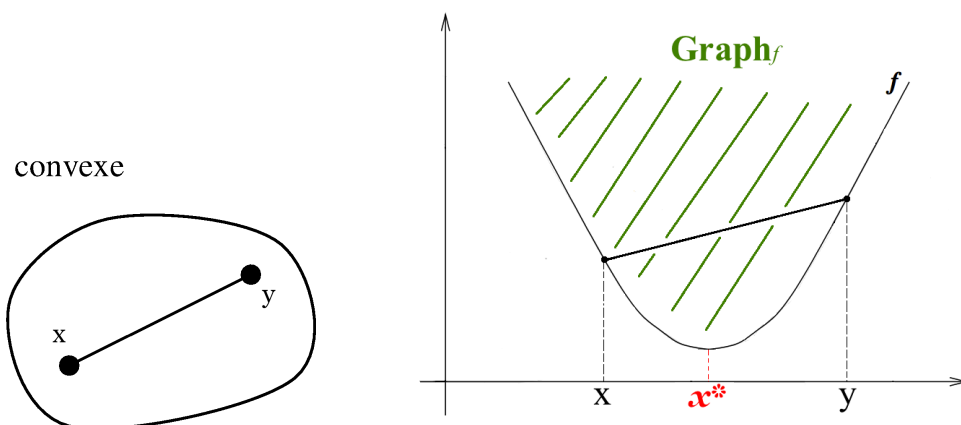


FIGURE 2.1 – La figure de gauche donne un exemple d'un ensemble convexe tandis que la figure de droite montre une fonction convexe. On note que le minimum de cette fonction est unique et il est atteint au point  $x^*$ .

### Illustration de la non convexité

Ainsi, c'est à partir de la convexité que nous pouvons définir la non convexité de la manière suivante :

**Définition 2.1.3.3: non Convexité**

Un ensemble  $C$  (respec. une fonction  $f$ ) est **non convexe** si il ne respecte pas la définition (2.11) (respec. (2.13)). Une illustration de la non convexité est donnée en figure 2.2.

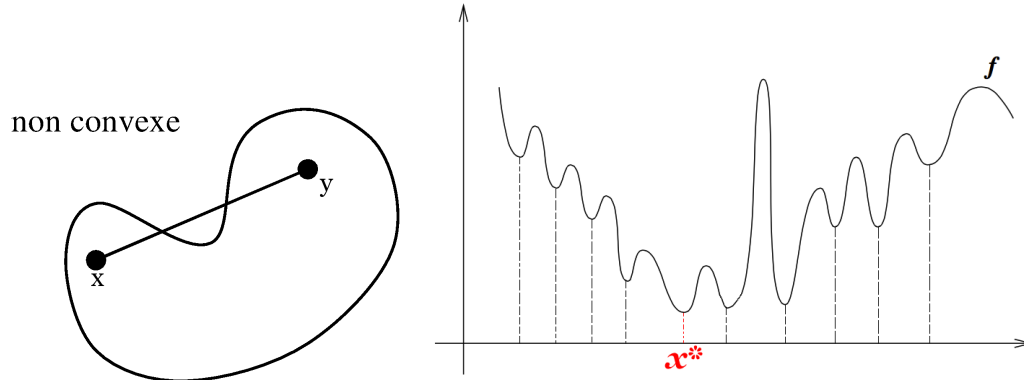


FIGURE 2.2 – La figure de gauche donne un exemple d'un ensemble non convexe tandis que la figure de droite montre une fonction non convexe. On note que le minimum global est atteint au point  $x^*$ , mais qu'il existe plusieurs minima et maxima locaux.

**2.1.3.2 Optimum**

La solution « parfaite » est une solution pour laquelle le minimum (respec. le maximum) de  $f$  est atteint pour l'ensemble du domaine  $D_f$ . Cette solution est appelée minimum (respec. maximum) **global**.

**Définition 2.1.3.4: optimum global et local**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ .

► Un point  $x^*$  est un **minimum global** (respec. **maximum global**) de  $f$  si

$$f(x^*) \leq f(x), \forall x \in D_f \quad (2.14)$$

(respec.  $f(x^*) \geq f(x)$ ).

► Un point  $x^*$  est un **minimum local** (respec. **maximum local**) de  $f$  si il existe un voisinage de  $x^*$ ,  $V \subset D_f$  tel que

$$f(x^*) \leq f(x), \forall x \in V \quad (2.15)$$

(respec.  $f(x^*) \geq f(x)$ ).

Lorsque  $f$  est une fonction convexe, comme sur la figure 2.1, on peut garantir la globalité de la solution lorsque celle-ci existe. Mais l'optimum global peut être difficile à trouver lorsque  $f$  est non convexe (voir figure 2.2), car la connaissance de  $f$  n'est généralement que locale et on ne peut parcourir la fonction sur tout son domaine de définition.

D'ordinaire, l'un des objectifs d'un algorithme est de visiter le moins de points possible afin de limiter le nombre d'itérations et le temps de calcul. Or généralement, l'ensemble des valeurs que prend la fonction  $f$  peut être inconnu. De plus, on ne peut avoir accès à l'expression de ses minima, et il est donc difficile d'assurer qu'un optimum trouvé est global. Un exemple est illustré en figure 2.2, on voit que la fonction admet plusieurs minimums locaux sur des petits intervalles. Il est généralement difficile de trouver l'optimum global pour de telles fonctions, parce que les algorithmes ont tendance à être « piégés » dans ces minima locaux.

### 2.1.3.3 Identifier un optimum

D'après ce qui a été dit précédemment, afin de déterminer si un point  $x^*$  est un minimum local on peut examiner tous les points dans son voisinage immédiat pour s'assurer qu'aucun d'entre eux n'a une valeur de fonction plus petite.

Cependant, il existe des moyens beaucoup plus efficaces et pratiques d'identifier les optima locaux. Pour cela on utilise les propriétés de **différentiabilité** de la fonction  $f$ . En particulier, si  $f$  est deux fois continûment différentiable, on peut affirmer qu'un point  $x^*$  est un minimum ou maximum local en examinant seulement le **gradient**  $\nabla f(x^*)$  et la **hessienne**  $\nabla^2 f(x^*)$ . Les théorèmes suivants donnent les fondements de ces opérations.

#### Théorème 2.1.3.1: caractérisation des minima locaux et globaux

- ▶ **Condition nécessaire de premier ordre** : Si  $x^*$  est un minimum local et que  $f$  est continûment différentiable dans un voisinage de  $x^*$ , alors  $\nabla f(x^*) = 0$ .
- ▶ **Condition nécessaire de second ordre** : Si  $x^*$  est un minimum local et que  $\nabla^2 f$  est continue dans un voisinage de  $x^*$ , alors  $\nabla f(x^*) = 0$  et  $\nabla^2 f(x^*)$  est semi-définie positive.
- ▶ **Cas particulier** : lorsque  $f$  est convexe, tout minimum local  $x^*$  est un minimum global de  $f$ .

La preuve de ces théorèmes peut être trouvée dans [NOCEDAL et WRIGHT, 2006]. Ces résultats, qui sont basés sur des calculs élémentaires, fournissent les bases de la majorité des algorithmes d'optimisation sans contrainte. Souvent, les algorithmes d'optimisation cherchent le point où le gradient  $\nabla f$  s'annule. Ces propriétés sont à l'origine des méthodes qui seront exposées en section 2.4.

Enfin, le théorème suivant nous permettra d'expliquer le point de départ des méthodes algorithmiques que l'on présentera en section 2.4.

#### Théorème 2.1.3.2: théorème de Taylor

Si  $f$  est deux fois continûment différentiable sur son domaine, alors

$$f(x+p) = f(x) + \nabla f(x)^t p + \frac{1}{2} p^t \nabla^2 f(x+tp) p, \quad (2.16)$$

où  $p \in \mathbb{R}^n$  et  $t \in [0, 1]$ .

En pratique, les fonctions étudiées ne sont pas nécessairement différentiables et nous ne pouvons donc pas directement appliquer les précédentes assertions. L'idée est alors de généraliser la notion de différentiabilité et définir des propriétés pour le cas des fonctions non différentiables. Ceci sera discuté en sous section 2.2.1. Dans la section suivante nous allons introduire les outils mathématiques essentiels en analyse variationnelle pour l'étude de notre cas.

## 2.2 Outils mathématiques

Cette section présente trois outils d'analyse variationnelle que l'on rencontre en optimisation non convexe. En effet, on rappelle que le problème d'optimisation en **TDM** spectrale est un problème de minimisation de fonctions **non convexes** incorporant des régularisations **non différentiables**.

Dans la première sous-section nous allons introduire le **sous-différentiel** qui est un outil essentiel pour substituer la dérivée. Nous définirons dans la seconde sous-section l'**opérateur proximal** qui permet de simplifier le problème de minimisation dans le cas non différentiable. Enfin, nous parlerons en troisième sous-section de l'inégalité de **Kurdyka-Łojasiewicz** qui nous donne une garantie de convergence des algorithmes que nous allons utiliser.

## 2.2.1 Notion de sous-différentiel

le **sous-différentiel** est un concept permettant de décrire la **variation locale** d'une fonction non nécessairement différentiable dans un sens classique, celui auquel on attache aujourd'hui le nom de **Fréchet** (définition (2.17)).

Lorsqu'une fonction est convexe, la **dérivée** classique de celle-ci en un point est la **pen**te de l'application linéaire tangente au point considéré. Mais cette dérivée n'existe pas nécessairement. Par ailleurs le sous-différentiel est l'**ensemble des pentes** de toutes les minorantes affines de la fonction, ceci sera illustré en 2.2.1.2.

Les fonctions apparaissant dans les problèmes inverses peuvent être non différentiables et/ou non convexes, ce qui peut donc compliquer la recherche d'optimum. C'est pour cette raison que le concept de différentiation joue un rôle important en analyse. Il a été étendu dans de nombreux domaines suivant l'usage qu'on veut lui réserver [BORWEIN et LEWIS, 2000].

Nous allons donner les notions de base de **sous-différentiabilité** pour les fonctions non nécessairement convexes. Ainsi nous pourrons faire le lien entre les optimums d'une fonction et son sous-différentiel.

### Définition 2.2.1.1: différentiabilité de Fréchet

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre.

Soit  $a \in D_f$ , on dit que  $f$  est **différentiable** en  $a$  au sens de **Fréchet** si il existe une application linéaire  $L : \mathbb{R}^n \rightarrow \mathbb{R}$  et une fonction  $\epsilon : \mathbb{R} \rightarrow \mathbb{R}$  vérifiant  $\epsilon(t) \xrightarrow[t \rightarrow 0]{} 0$  telles que

$$\forall v \in \mathbb{R}^n, \left| \frac{f(a + tv) - f(a)}{t} - L(v) \right| \leq \epsilon(|t|) \quad (2.17)$$

on appelle alors l'application  $L$  la **différentielle** de  $f$  en  $a$ .

Un lien peut être établi entre le sous-différentiel d'une fonction et ses points critiques par le théorème suivant

### Théorème 2.2.1.1: caractérisation des points critiques

Si  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  est une fonction propre et différentiable sur son domaine alors on

$$x \in \operatorname{argmin} f \quad \Rightarrow \quad \mathbf{0} \in \partial f(x), \quad (2.18)$$

où  $\partial f(x)$  est le **sous-différentiel** de  $f$  en  $x$  (voir définition 2.2.1.3).

Lorsqu'une fonction  $f : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$  est la somme d'une fonction différentiable et d'une fonction non différentiable, la règle de différentiation est alors donnée par la proposition suivante :

**Proposition 2.2.1.1: [ROCKAFELLAR et WETS, 1998]**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre telle que

$$\forall x \in \mathbb{R}^n, \quad f(x) = f_1(x) + f_2(x)$$

où  $f_1$  est une fonction propre différentiable sur  $D_f$  et  $f_2$  est une fonction propre (non nécessairement différentiable). On a alors

$$\forall x \in D_f, \quad \partial f(x) = \nabla f_1(x) + \partial f_2(x) \quad (2.19)$$

Cette proposition est particulièrement utile dans notre cas, car nous utilisons un terme de régularisation qui n'est pas différentiable. Elle stipule que l'on peut calculer la forme de la sous-différentielle pour le cas d'une somme de fonctions, ce qui nous permettra de trouver des points critiques à étudier.

Enfin, la définition suivante est utile pour l'analyse la convergence des algorithmes d'optimisation étudiés durant cette thèse.

**Définition 2.2.1.2: continuité du gradient**

Une fonction différentiable  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  est dite de **gradient  $\beta$ -Lipschitz** sur  $C \subset \mathbb{R}^n$ , si  $\beta > 0$  et

$$\forall (x, y) \in C^2 \quad \|\nabla f(x) - \nabla f(y)\| \leq \beta \|x - y\| \quad (2.20)$$

**2.2.1.1 Cas convexe**

Comme nous l'avons précédemment dit, le sous-différentiel permet de décrire la variation locale d'une fonction convexe (à valeurs réelles) non nécessairement différentiable dans un sens classique. On peut le définir de manière générale comme l'ensemble des pentes de toutes les minorants de la fonction [BERTSEKAS, 1999; ROCKAFELLAR, 1970].

**Définition 2.2.1.3: sous-différentiabilité cas convexe**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction convexe.

Un vecteur  $\eta \in \mathbb{R}^n$  est appelé **sous-gradient** de  $f$  au point  $a \in D_f$  si

$$\forall x \in D_f, f(x) \geq f(a) + \langle \eta, x - a \rangle. \quad (2.21)$$

Le **sous-différentiel** de  $f$  en  $a$ , noté  $\partial f(a)$ , correspond alors à l'ensemble des sous-gradients vérifiant (2.21).

Les minimiseurs globaux d'une fonction convexe peuvent être caractérisés par la règle de Fermat suivante :



**Théorème 2.2.1.2:**

► **Règle de Fermat** : Si  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  est une fonction convexe, alors l'ensemble des minimiseurs globaux de  $f$  vérifie

$$\operatorname{argmin} f = \{x \in \mathbb{R}^n \mid \mathbf{0} \in \partial f(x)\} \quad (2.22)$$

► Soit  $x \in D_f$ . Alors  $f$  est différentiable en  $x$  au sens de Fréchet si et seulement si  $\partial f(x)$  est un singleton, et dans ce cas

$$\partial f(x) = \{\nabla f(x)\}$$

où  $\nabla f(x)$  est le **gradient** de  $f$  en  $x$ .

Le théorème ci-dessus nous permet donc d'établir un lien entre la solution du problème de minimisation et le sous-différentiel de  $f$ .

**2.2.1.2 Cas dimension 1**

Pour mieux comprendre le sous-différentielle, examinons le cas unidimensionnel. Lorsque  $\mathcal{H} = \mathbb{R}$ ,  $\partial f(x)$  est un sous ensemble de  $\mathbb{R}$ .

**Théorème 2.2.1.3:**

Soit  $f : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction convexe, propre et semi-continue inférieurement. Soit  $x \in D_f$  tel que les dérivées à gauche et à droite de  $x$ , notées  $f'_g(x)$  et  $f'_d(x)$ , existent, Alors

$$\partial f(x) = [f'_g(x), f'_d(x)].$$

**Exemple :**

Soit la fonction  $f(x) = |x|$  définie sur  $\mathbb{R}$ , on a d'après le théorème précédent

$$\partial f(0) = [-1, 1].$$

L'interprétation géométrique est la suivante : le sous-différentiel est formé par toutes les directions des hyperplans (ici des droites) qui passent par le point  $(0, f(0))$  et restent "sous" le graphe de la fonction  $f$  (voir figure 2.3). Ces hyperplans sont appelés **hyperplans support** ou **hyperplans d'appui** au graphe de  $f$  en  $x$ .

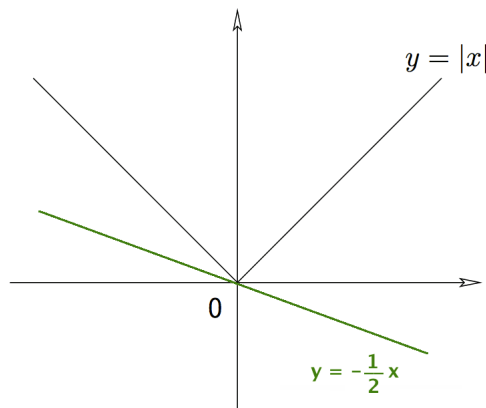


FIGURE 2.3 – Illustration de la fonction  $f(x) = |x|$  avec l'hyperplan d'équation  $y = -\frac{1}{2}x$ .

### 2.2.1.3 Conclusion

Les notions définies précédemment sont essentielles, car elles permettent de garantir l'existence de points critiques à étudier. En effet, on rappelle que notre problème d'optimisation (2.1) utilise des régularisations non-différentiables (voir 2.3), or pour trouver une solution on doit pouvoir calculer le gradient mais celui-ci n'existe pas toujours. La notion de sous-différentiabilité nous assure de pouvoir substituer cela et ainsi de trouver une solution. Un autre outil qui intervient également en optimisation non-différentiable et que l'on va utiliser dans les algorithmes présentés en section 2.4 est l'**opérateur proximal**, nous le présentons dans la prochaine sous-section.

## 2.2.2 Opérateurs proximaux

### 2.2.2.1 Intérêt

Cette sous section traite d'une méthode qui donne lieu a une classe d'algorithmes, dite des **algorithmes proximaux** pour résoudre des problèmes d'optimisation.

Tout comme la méthode de Newton [NOCEDAL et WRIGHT, 2006] est un outil standard pour résoudre les problèmes de minimisation de fonctions deux fois différentiables et de taille modeste, les algorithmes proximaux peuvent être considérés comme un outil analogue pour des problèmes d'optimisation de fonctions non différentiables, contraints et à grande dimension tels que le nôtre (2.1) [CHAUX et collab., 2007; COMBETTES et PESQUET, 2008; COMBETTES et WAJS, 2005].

Les méthodes proximales sont des méthodes d'optimisation plus élaborées que les méthodes d'optimisation classiques comme la méthode de Newton, qui consistent à faire des opérations basiques d'algèbre linéaire et de calcul de gradients et de hessiennes [PARIKH et BOYD, 2014].

Dans les algorithmes proximaux, l'opération de base évalue l'**opérateur proximal** (définition en (2.23)) d'une fonction, qui est lui même un problème d'optimisation convexe. Ces sous-problèmes admettent souvent des solutions analytiques ou peuvent être résolus très rapidement avec des méthodes simples et spécialisées (voir section 2.4).

### 2.2.2.2 Définitions

L'opérateur proximal est central pour la résolution à l'aide de méthodes de premier ordre des problèmes non différentiables tels que (2.1).

#### Définition 2.2.2.1: opérateur proximal

L'**opérateur proximal** en un point  $x \in \mathbb{R}^n$  a été défini dans [MOREAU, 1965] pour une fonction  $f$  convexe, propre et semi-continue inférieurement, comme étant l'unique minimiseur de la fonction

$$f(\cdot) + \frac{1}{2} \|\cdot - x\|^2.$$

On a ainsi  $\forall \gamma > 0$

$$\begin{aligned} \text{prox}_{\gamma, f} &: \mathbb{R}^n \rightarrow \mathbb{R}^n \\ x &\mapsto \text{prox}_{\gamma, f}(x) \end{aligned}$$

où

$$\text{prox}_{\gamma, f}(x) = \arg \min_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2\gamma} \|y - x\|^2 \right\} \quad (2.23)$$

La définition précédente montre que  $\text{prox}_{\gamma, f}(x)$  est un point qui permet de faire le compromis entre la **minimisation** de la fonction  $f$  et la **proximité** du point  $x$ . Pour cette raison,  $\text{prox}_{\gamma, f}(x)$  est parfois appelé **point proximal de  $x$**  par rapport à  $f$ . Par ailleurs, le paramètre  $\gamma$  peut être interprété comme un poids relatif qui permet d'équilibrer les deux termes de l'opérateur proximal.

Cet opérateur a été largement étudié pour des fonctions convexes usuelles du traitement du signal [CHAUX et collab., 2007; COMBETTES et PESQUET, 2008; COMBETTES et WAJS, 2005; PARIKH et BOYD, 2014]. On peut généraliser la définition (2.23) pour des fonctions  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  non nécessairement convexes dans le cas de normes pondérées [ATTOUCH et collab., 2011; CHOUZENOUX et collab., 2014; COMBETTES et VÛ, 2014] :

#### Définition 2.2.2.2: opérateur proximal généralisé

Soient  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre et semi-continue inférieurement,  $\mathbf{U} \in \mathcal{S}^+(\mathbb{R}^n)$  et  $x \in \mathbb{R}^n$ . L'opérateur proximal de  $f$  en  $x$  relatif à la métrique induite par  $\mathbf{U}$  est l'ensemble défini par

$$\text{prox}_{\mathbf{U}, f}(x) = \arg \min_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2} \|y - x\|_{\mathbf{U}}^2 \right\} \quad (2.24)$$

On notera également  $\text{prox}_{\mathbf{U}, f}(x)$  lorsque cet ensemble est égal à un singleton.

En dimension finie, lorsque  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  est une fonction non nécessairement convexe, la question d'existence de l'opérateur proximal de  $f$  se pose. Nous allons donc énoncer un ensemble de conditions permettant d'assurer que l'ensemble  $\text{prox}_{\gamma, f}(x)$  soit non vide pour tout  $x \in \mathbb{R}^n$ .

#### Définition 2.2.2.3:

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre et semi-continue inférieurement. On dit que  $f$  est **prox-bornée** s'il existe  $\gamma > 0$  tel que

$$\exists x \in \mathbb{R}^n, \inf_{y \in \mathbb{R}^n} \left\{ f(y) + \frac{1}{2\gamma} \|y - x\|^2 \right\} > -\infty \quad (2.25)$$

On notera alors  $\gamma_f = \sup\{\gamma \mid (2.25) \text{ satisfaite}\}$  et on dit que la fonction  $f$  est  $\gamma_f$ -**prox-bornée**.

#### Théorème 2.2.2.1: [ROCKAFELLAR et WETS, 1998]

Si  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre, semi-continue inférieurement et  $\gamma_f$ -prox-bornée, de seuil  $\gamma_f > 0$ , alors pour tout  $\gamma \in ]0, \gamma_f[$  et  $x \in \mathbb{R}^n$ , l'ensemble  $\text{prox}_{\gamma, f}(x)$  est non vide et compact.

On trouvera la démonstration dans [ROCKAFELLAR et WETS, 1998]. Ce théorème est important car il généralise l'utilité de l'opérateur proximal aux fonctions non nécessairement convexes, ce qui est particulièrement utile dans notre cas.

### 2.2.2.3 Propriétés

L'opérateur proximal est très utilisé dans la résolution de problèmes variationnels par le biais d'**algorithmes proximaux**. Il a largement été étudié ces dernières années, particulièrement pour une fonction  $f$  propre, convexe et semi-continue inférieurement dans le cadre de la métrique usuelle euclidienne [CHAUX et collab., 2007; COMBETTES et PESQUET, 2008; COMBETTES et WAJS, 2005; PARIKH et BOYD, 2014] et de métriques pondérées [COMBETTES et VÛ, 2014]. Nous listons ici quelques propriétés utiles des opérateurs proximaux.

**Proposition 2.2.2.1: caractérisation**

Soient  $\mathbf{U} \in \mathbb{S}^+(\mathbb{R}^n)$ ,  $x \in \mathbb{R}^n$  et  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre, semi-continue inférieurement, telle que  $\text{prox}_{\mathbf{U}, f}(x) \neq \emptyset$ . On a alors

- Si  $p \in \text{prox}_{\mathbf{U}, f}(x)$  alors  $x - p \in \mathbf{U}^{-1}\partial f(p)$
- Si  $f$  est convexe alors on a équivalence

$$p = \text{prox}_{\mathbf{U}, f}(x) \Leftrightarrow x - p \in \mathbf{U}^{-1}\partial f(p) \quad (2.26)$$

**Proposition 2.2.2.2: point fixe**

Soient  $\mathbf{U} \in \mathbb{S}^+(\mathbb{R}^n)$  et  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre, semi-continue inférieurement, telle que  $\text{prox}_{\mathbf{U}, f}(x) \neq \emptyset$  et cela  $\forall x \in \mathbb{R}^n$ . On a alors

- Si  $x \in \text{prox}_{\mathbf{U}, f}(x)$  alors  $x$  est un point critique de  $f$ .
- Si  $f$  est convexe alors on a équivalence

$$x = \text{prox}_{\mathbf{U}, f}(x) \Leftrightarrow x \in \text{argmin } f \quad (2.27)$$

Cette propriété est importante car elle stipule que les points fixes de l'opérateur proximal de  $f$  sont précisément les minimiseurs de  $f$ . Cela implique une connexion étroite entre les opérateurs proximaux et la théorie des points fixes, et suggère que les algorithmes proximaux (voir section 2.4 pour plus de détails) peuvent être interprétés comme une solution aux problèmes d'optimisation en trouvant des points fixes d'opérateurs appropriés.

**Proposition 2.2.2.3: post-combinaison**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre, semi-continue inférieurement, telle que  $\text{prox}_{\gamma, f}(x) \neq \emptyset$  et cela  $\forall x \in \mathbb{R}^n$ . Si  $f(x) = \alpha g(x) + b$ , avec  $\alpha > 0$  alors

$$\text{prox}_{\gamma, f}(x) = \text{prox}_{\alpha\gamma, g}(x). \quad (2.28)$$

**Proposition 2.2.2.4: pre-combinaison**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction propre, semi-continue inférieurement, telle que  $\text{prox}_{\gamma, f}(x) \neq \emptyset$  et cela  $\forall x \in \mathbb{R}^n$ . Si  $f(x) = g(\alpha x + b)$ , avec  $\alpha \neq 0$  alors

$$\text{prox}_{\gamma, f}(x) = \frac{1}{\alpha} \left( \text{prox}_{\alpha^2\gamma, g}(\alpha x + b) - b \right). \quad (2.29)$$

Par ailleurs, si  $f(x) = g(\mathbf{U}x)$  avec  $\mathbf{U}$  une matrice orthogonale (i.e.  $\mathbf{U}^t \mathbf{U} = \mathbf{U} \mathbf{U}^t = \mathbf{I}$ ), alors

$$\text{prox}_{\gamma, f}(x) = \mathbf{U}^t \text{prox}_{\gamma, g}(\mathbf{U}x). \quad (2.30)$$

### 2.2.2.4 Cas particuliers

Dans ce paragraphe nous donnerons quelques exemples utiles d'opérateurs proximaux de fonctions usuelles pour la résolution de problèmes inverses.

#### Opérateur de projection

**Définition 2.2.2.4: projection**

Soient  $E$  un ensemble fermé non vide de  $\mathbb{R}^n$  et  $x \in \mathbb{R}^n$ . La **projection** du point  $x$  sur le sous-ensemble  $E$  est définie par

$$\Pi_E(x) \in \operatorname{argmin}_{y \in E} \|y - x\| \quad (2.31)$$

En d'autres termes, un élément de  $\Pi_E(x)$  est un point de  $E$  tel que sa distance à  $x$  soit minimale.

**Remarque :**

L'opérateur proximal peut être vu comme une généralisation de l'opérateur de projection au sens où, pour tout sous-ensemble  $E$  de  $\mathbb{R}^n$  on a

$$\Pi_E = \operatorname{prox}_{\chi_E} \quad (2.32)$$

où  $\chi_E$  désigne la fonction indicatrice de  $E$  définie comme suit :

$$\chi_E(x) = \begin{cases} 0 & \text{si } x \in E \\ +\infty & \text{sinon.} \end{cases} \quad (2.33)$$

**Proposition 2.2.2.5: [BAUSCHKE et COMBETTES, 2011]**

Si  $C$  est un sous-ensemble convexe fermé non vide de  $\mathbb{R}^n$ , alors la projection  $\Pi_C(x)$  existe et est unique  $\forall x \in \mathbb{R}^n$ .

**Opérateurs de fonctions puissances**

Nous allons donner ici les opérateurs proximaux des fonctions  $f_\alpha : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  de la forme

$$f_\alpha(x) = \sum_{n=1}^N |x_n|^\alpha \quad (2.34)$$

pour différents  $\alpha \geq 0$ .

Lorsque  $\alpha$  prend la valeur 0 ou 1 on appelle l'opérateur proximal associé **opérateur de seuillage**. En particulier, lorsque  $\alpha = 1$  nous parlerons de **seuillage doux** [COMBETTES et PESQUET, 2010], et lorsque  $\alpha = 0$  nous parlerons de **seuillage dur** [BLUMENSATH et DAVIES, 2008] (voir figure 2.4).

**Définition 2.2.2.5: opérateur de seuillage**

► Soit  $\lambda > 0$ . On définit  $\forall x \in \mathbb{R}$  l'opérateur de **seuillage doux** de seuil  $\lambda$  comme suit :

$$\operatorname{soft}_\lambda(x) = \operatorname{sign}(x) \max\{|x| - \lambda, 0\} = \begin{cases} x - \lambda & \text{si } x > \lambda \\ 0 & \text{si } x \in [-\lambda, \lambda] \\ x + \lambda & \text{si } x < -\lambda \end{cases} \quad (2.35)$$

où  $\operatorname{sign}(x) = 1, 0$  ou  $-1$  selon si  $x$  est positif, nul ou négatif.

► Soit  $\lambda > 0$ . On définit  $\forall x \in \mathbb{R}$  l'opérateur de **seuillage dur** de seuil  $\lambda$  comme suit :

$$\operatorname{hard}_\lambda(x) = \operatorname{sign}(x) \max\{|x| - \lambda, 0\} = \begin{cases} 0 & \text{si } |x| \leq \lambda \\ x & \text{sinon.} \end{cases} \quad (2.36)$$

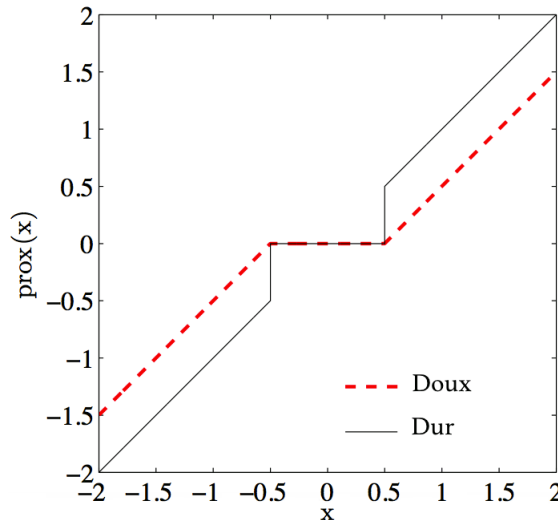


FIGURE 2.4 – Opérateurs de seuillage doux (courbe rouge) et de seuillage dur (courbe noire) pour la valeur de seuil  $\lambda = 0.5$ .

L'opérateur de seuillage doux intervient lorsque nous utilisons la parcimonie comme régularisation avec la norme  $l_1$ .

### 2.2.2.5 Conclusion

Les deux sous-sections précédentes présentent les outils nécessaires pour l'analyse variationnelle d'une fonction non nécessairement différentiable et permettent une caractérisation mathématique des points critiques et ainsi aider à la résolution de problèmes inverses non triviaux. Dans le cas d'optimisation convexe, les outils précédent suffisent pour garantir la convergence vers un optimum. Or dans notre cas les fonctions que l'on étudie ne sont pas convexes. Nous ne pouvons pas non plus être sûr de la nature d'un point critique d'une fonction non convexe lorsqu'un algorithme converge vers l'un de ces points, et c'est particulièrement le cas pour notre fonction d'attache aux données (voir (2.46)).

Il faut donc des outils pour garantir la convergence d'algorithmes dans notre cas. Dans la sous-section suivante nous allons introduire l'**inégalité Kurdyka-Łojasiewicz**, un outil fondamental en optimisation non convexe qui permet de démontrer la convergence d'algorithmes pour des fonctions non nécessairement convexes.

## 2.2.3 Inégalité de Kurdyka-Łojasiewicz

### 2.2.3.1 Idée de base

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction différentiable. On considère l'équation différentielle suivante :

$$\begin{cases} \frac{dx(t)}{dt} = -\nabla f(x(t)) & \text{si } \forall t \in ]0, +\infty[ \\ x(0) = x_0. & \end{cases}$$

Cette équation peut être vue comme une version continue de l'algorithme de descente de gradient (voir section 2.4 pour plus de détail) donné par

$$x_{k+1} = x_k - \mu_k \nabla f(x_k), \quad \text{avec } \mu_k > 0 \quad \forall k \in \mathbb{N}.$$

La résolution de ces équations génère une trajectoire de gradient (*resp.* une suite d'itérés). La solution  $x(t)$  générée assure que  $f(x(t))$  soit décroissante (*resp.* que la suite  $f(x_k)$  est décroissante).

On considère alors une trajectoire de  $x(t)$  (resp. une suite  $\{x_k\}_{k \in \mathbb{N}}$ ) bornée.

On voudrait pouvoir assurer la convergence de  $x(t)$  (resp. de la suite  $\{x_k\}_{k \in \mathbb{N}}$ ) lorsque  $t \rightarrow +\infty$  (resp.  $k \rightarrow +\infty$ ), vers un point critique de  $f$ , en d'autres termes assurer que la trajectoire  $x(t)$  soit de longueur finie.

C'est dans ce genre de situation qu'intervient l'inégalité de **Kurdyka-Łojasiewicz**, qui est un outil puissant pour analyser la convergence des méthodes de type descente de gradient ainsi que les problèmes apparentés.

### 2.2.3.2 Inégalité de Łojasiewicz

Cette inégalité a d'abord été proposée en premier dans [ŁOJASIEWICZ, 1963] où l'on démontre qu'elle est satisfaite par toute fonction analytique réelle :

#### Proposition 2.2.3.1: inégalité de Łojasiewicz

Soient  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  une fonction analytique réelle, et  $x^*$  un point critique de cette fonction. Alors  $\exists c > 0$  et  $\theta \in [\frac{1}{2}, 1[$  tels que pour tout  $x$  dans un voisinage  $V(x^*)$  de  $x^*$ ,  $f$  satisfait l'**inégalité de Łojasiewicz** suivante :

$$|f(x) - f(x^*)|^\theta \leq c \|\nabla f(x)\|. \quad (2.37)$$

Cette inégalité a été utilisée pour démontrer la convergence d'algorithmes de descente de gradient (voir section 2.4) pour la minimisation de fonctions différentiables non nécessairement convexes, elle permet d'assurer que la courbe de gradient générée par ce type d'algorithme est de longueur finie [ABSIL et collab., 2005].

### 2.2.3.3 Inégalité de Kurdyka-Łojasiewicz

L'inégalité de Łojasiewicz a été généralisée dans [KURDYKA, 1998] pour les fonctions différentiables, on parlera alors d'**inégalité de KL**. Plus récemment, dans [BOLTE et collab., 2006] il a été démontré que l'inégalité de KL peut être généralisée pour des fonctions non nécessairement différentiables en utilisant les sous-différentiel. Cette inégalité, plus générale que (2.37), s'écrit comme suit :

#### Proposition 2.2.3.2: inégalité de Kurdyka-Łojasiewicz

La fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  satisfait l'inégalité de Kurdyka-Łojasiewicz si  $\forall \hat{x} \in \text{dom} \partial f$  il existe  $\xi \in ]0, +\infty]$ , un voisinage  $V(\hat{x})$  de  $\hat{x}$  et une fonction concave  $\varphi : ]0, \xi[ \rightarrow \mathbb{R}^+$  tels que :

- $\varphi(0) = 0$ ,
- $\varphi$  est continue en 0 et continument différentiable sur  $]0, \xi[$ ,
- $\forall u \in ]0, \xi[, \varphi'(u) > 0$ ,
- $\forall x$  tel que  $f(\hat{x}) \leq f(x) \leq f(\hat{x}) + \xi$ , on a :

$$\varphi'(f(\hat{x}) - f(x)) \mathbf{dist}(0, \partial f(x)) \geq 1, \quad (2.38)$$

$$\text{avec } \mathbf{dist}(0, \partial f(x)) = \inf_{u \in \partial f(x)} \|u\|$$

On remarque que pour une fonction analytique réelle, si on pose  $\varphi(u) = u^{1+\theta}$  avec  $\theta \in [\frac{1}{2}, 1[$ , on retombe sur l'inégalité (2.37). Une démonstration de ce résultat se trouve dans [DENKOWSKA et collab., 2008].

**Lien avec notre problème**

Cette propriété a permis de démontrer la convergence d’algorithmes proximaux [ATTOUCH et collab., 2011], et pour des fonctions **non nécessairement convexes**. De plus, Il a été démontré que l’inégalité de Kurdyka-Łojasiewicz est satisfaite par les fonctions **semi-algébriques** et **analytiques réelles** [ATTOUCH et collab., 2010; BOLTE et collab., 2010]. Nous allons donc nous servir de cela pour montrer que les fonctions que nous étudions dans le cas de la **TDM** spectrale sont analytiques réelles. Elles satisferont ainsi l’inégalité **KL** et nous aurons donc une garantie de convergence pour les méthodes employées pour résoudre le problème.

**Conclusions**

Cette section nous a permis d’introduire trois outils mathématiques importants pour aborder le problème d’optimisation en **TDM** spectrale (2.1).

Dans la première sous-section 2.2.1 nous avons introduit le **sous-différentiel** qui est un outil essentiel pour substituer la dérivée au sens classique, ceci permet de trouver des points critiques à étudier même lorsque les fonctions sont non différentiables.

Dans la seconde sous-section 2.2.2 nous avons parlé de l’**opérateur proximal**, un outil très étudié dans la résolution de problèmes variationnels par le biais d’algorithmes proximaux. Il permet de simplifier le problème de minimisation et ainsi faciliter la recherche de points critiques. Enfin, nous avons brièvement défini en troisième sous-section 2.2.3 l’inégalité de **Kurdyka-Łojasiewicz**, une propriété qui donne une garantie de convergence des algorithmes que nous allons utiliser.

Ces outils constituent le support de base pour aborder notre problème de **TDM** spectrale. Dans la prochaine section nous allons voir comment construire le problème inverse lié à la **TDM** spectrale et expliciter les différents choix possibles des fonctions à minimiser. Nous pourrons ainsi par la suite voir des méthodes pour la résolution de ce problème.

**2.3 Définitions des fonctions objectifs**

Comme nous l’avons précédemment dit en 2.1.2, de nombreuses applications en ingénierie se ramènent à la résolution de problèmes inverses. Cette résolution consiste en l’estimation du signal original inconnu à partir d’une version dégradée ou incomplète de celui-ci.

Une méthode usuelle pour trouver le signal original consiste à le définir comme étant un minimiseur d’une somme de deux fonctions [DEMOMENT, 1989]

$$\hat{x} \in \operatorname{argmin}_{x \in \mathbb{R}^n} \{f(x) = h(x) + g(x)\} \tag{2.39}$$

où  $h : \mathbb{R}^n \rightarrow ]-\infty, +\infty]$  est le terme **d’attache aux données** qui contient les informations directement liées au problème traité et  $g : \mathbb{R}^n \rightarrow ]-\infty, +\infty]$  est le terme de **régularisation** (ou de **pénalisation**) permettant d’introduire des informations que l’on connaît *a priori* sur le signal recherché.

On rappelle que dans notre cas le problème inverse que l’on souhaite résoudre est similaire et s’écrit comme suit :

$$\hat{A} \in \operatorname{argmin}_{A \in \mathbb{R}^{D \times K}} \{J(A) = G(A) + R(A)\} \tag{2.40}$$

avec pour modèle direct sous-jacent

$$Y = \mathcal{B}(e^{-SA\Sigma}F) \tag{2.41}$$



où  $\mathcal{B}(\cdot)$  est le bruit de mesure. On s'intéressera dans la suite au choix de la fonction d'attache aux données  $G(\mathbf{A})$ . On peut montrer [REPETTI, 2015] que ce choix dépend du type de bruit  $\mathcal{B}(\cdot)$  considéré. On donnera dans la suite quelques exemples de fonctions d'attache aux données ainsi que des fonctions de régularisation pour notre cas.

## 2.3.1 Fonction d'attache aux données

### 2.3.1.1 Bruit gaussien corrélé

Si on suppose  $\mathcal{B}(\cdot)$  est un bruit gaussien centré et corrélé alors  $Y$  est une réalisation d'une variable aléatoire  $\mathbf{w}$  suivant une **loi normale** d'espérance nulle et de matrice de variance-covariance définie positive  $\Omega \in \mathbb{R}^{\text{PM} \times \text{PM}}$ , avec  $P$  le nombre de mesures et  $M$  le nombre d'acquisitions :

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}_{\text{PM}}, \Omega)$$

dont la densité de probabilité est donnée par la fonction

$$\phi(\mathbf{w}) = \frac{1}{\sqrt{(2\pi)^{\text{PM}} \det(\Omega)}} \exp\left(-\frac{1}{2} \mathbf{w}^t \Omega^{-1} \mathbf{w}\right) \quad (2.42)$$

où  $\det(\Omega)$  désigne le déterminant de la matrice  $\Omega$ . On peut en déduire alors la fonction d'attache aux données  $G$  en passant par le logarithme de vraisemblance négatif

$$\begin{aligned} G(\mathbf{A}) &= -\log(\mathbb{P}(Y | \mathbf{A})) \\ G(\mathbf{A}) &= -\log(\phi(\mathbf{vec}(Y - e^{-\text{SA}\Sigma} \mathbf{F}))) \\ G(\mathbf{A}) &= \frac{1}{2} \mathbf{vec}(Y - e^{-\text{SA}\Sigma} \mathbf{F})^t \Omega^{-1} \mathbf{vec}(Y - e^{-\text{SA}\Sigma} \mathbf{F}) + \kappa \end{aligned}$$

où  $\mathbf{vec}(U)$  la forme vectorisée de  $U$  et  $\kappa = \frac{1}{2} \log((2\pi)^{\text{PM}} \det(\Omega)) \in \mathbb{R}$  une constante qui ne dépend pas de  $\mathbf{A}$ . Par conséquent on peut choisir comme terme d'attache aux données dans (1.45)

$$G(\mathbf{A}) = \frac{1}{2} \mathbf{vec}(Y - e^{-\text{SA}\Sigma} \mathbf{F})^t \Omega^{-1} \mathbf{vec}(Y - e^{-\text{SA}\Sigma} \mathbf{F}). \quad (2.43)$$

### 2.3.1.2 Bruit blanc gaussien

On dit que le vecteur aléatoire  $\mathbf{w}$  modélise un **bruit blanc gaussien** s'il suit une loi normale d'espérance nulle, et ses composantes sont indépendantes, identiquement distribuées, de variance égale à  $\sigma^2$ . On se place donc dans un cas particulier du bruit corrélé où  $\Omega = \sigma^2 \mathbf{I}_{\text{PM}}$ , où  $\mathbf{I}_{\text{PM}}$  étant la matrice identité de  $\mathbb{R}^{\text{PM} \times \text{PM}}$ . Ainsi, la fonction d'attache aux données associée se réduit à

$$G(\mathbf{A}) = \frac{1}{2\sigma^2} \|Y - e^{-\text{SA}\Sigma} \mathbf{F}\|_{\text{fro}}^2 \quad (2.44)$$

### 2.3.1.3 Bruit de Poisson

Le modèle gaussien précédent est un modèle de bruit simple, mais pas réaliste car il n'utilise pas *a priori* sur la relation qui existe entre le modèle  $e^{-\text{SA}\Sigma} \mathbf{F}$  et les mesures  $Y$ . Or, il existe en pratique une relation particulière qui est la **loi de Poisson** qui permet de mieux décrire la physique d'acquisition pour la détection de rayons X, qui interagissent de manière stochastique dans la matière. En effet, le nombre de photons détectés par le détecteur suit une loi de Poisson qui dévie les mesures autour de la valeur théorique du modèle. Pour une mesure  $\{p, m\}$  on peut exprimer cette relation comme suit :

$$\mathbb{P}\{Y_{p,m} | \{e^{-\text{SA}\Sigma} \mathbf{F}\}_{p,m}\} = \frac{\{e^{-\text{SA}\Sigma} \mathbf{F}\}_{p,m}^{Y_{p,m}}}{Y_{p,m}!} \exp(-\{e^{-\text{SA}\Sigma} \mathbf{F}\}_{p,m}). \quad (2.45)$$

En utilisant l'indépendance des mesures, on peut augmenter la vraisemblance

$$\begin{aligned}\mathbb{P}(Y_{p,m} | e^{-\text{SA}\Sigma\text{F}}) &= \mathbb{P}(Y_{1,1} | \{e^{-\text{SA}\Sigma\text{F}}\}_{1,1} \cap \dots \cap Y_{p,m} | \{e^{-\text{SA}\Sigma\text{F}}\}_{p,m}) \\ &= \prod_{p,m} \frac{\{e^{-\text{SA}\Sigma\text{F}}\}_{p,m}^{Y_{p,m}}}{Y_{p,m}!} \exp(-\{e^{-\text{SA}\Sigma\text{F}}\}_{p,m})\end{aligned}$$

On peut ainsi en déduire alors la fonction d'attache aux données  $G$  en passant par le logarithme de vraisemblance négatif

$$\begin{aligned}G(\mathbf{A}) &= -\log\left(\prod_{p,m} \frac{\{e^{-\text{SA}\Sigma\text{F}}\}_{p,m}^{Y_{p,m}}}{Y_{p,m}!} \exp(-\{e^{-\text{SA}\Sigma\text{F}}\}_{p,m})\right) \\ G(\mathbf{A}) &= \sum_{p,m} \{e^{-\text{SA}\Sigma\text{F}}\}_{p,m} - Y_{p,m} \log(\{e^{-\text{SA}\Sigma\text{F}}\}_{p,m}) + \log(Y_{p,m}!)\end{aligned}$$

Le terme  $\log(Y_{p,m}!)$  ne dépend pas de  $\mathbf{A}$  on peut donc le supprimer afin d'obtenir la fonction d'attache aux données vue en (1.11)

$$G(\mathbf{A}) = \sum_{p,m} \{e^{-\text{SA}\Sigma\text{F}}\}_{p,m} - Y_{p,m} \log(\{e^{-\text{SA}\Sigma\text{F}}\}_{p,m}). \quad (2.46)$$

### 2.3.2 Fonction de régularisation

Comme nous l'avons mentionné, le terme de régularisation permet de prendre en compte des caractéristiques connues des données à estimer. De manière plus générale, la fonction  $R$  peut s'écrire comme une somme de plusieurs fonctions de régularisation :

$$R(\mathbf{A}) = \sum_{j=1}^J \lambda_j R_j(\mathbf{A}) \quad (2.47)$$

où pour tout  $j \in \{1, \dots, J\}$  les  $\lambda_j \in [0, +\infty)$  sont appelés **hyperparamètres de régularisation**. L'utilisation de plusieurs fonctions de régularisation permet d'espérer de meilleurs résultats de reconstruction, puisque chacune d'elles permet de contraindre l'estimé à vérifier une caractéristique particulière.

Les hyperparamètres de régularisation  $\{\lambda_j\}_{j=1\dots J}$  sont à fixer de façon à obtenir la meilleure qualité de reconstruction possible. Notons que fixer  $\lambda_j = 0, \forall j \in \{1, \dots, J\}$  revient à trouver une solution  $\hat{\mathbf{A}}$  non régularisée :

$$\hat{\mathbf{A}} \in \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\text{argmin}} G(\mathbf{A}).$$

A l'inverse, faire tendre  $\lambda_j$  vers  $+\infty$  pour tout  $j \in \{1, \dots, J\}$  revient généralement à trouver une solution  $\hat{\mathbf{A}}$  ne prenant en compte que le terme de régularisation :

$$\hat{\mathbf{A}} \in \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\text{argmin}} R(\mathbf{A})$$

Il faut donc régler les hyperparamètres de régularisation de manière à obtenir un compromis entre le terme d'attache aux données et le terme de régularisation. En pratique, ces paramètres peuvent être réglés de manière à minimiser l'erreur entre le signal original  $\mathbf{A}^*$  (lorsque celui ci est disponible) et son estimé  $\hat{\mathbf{A}}$ . Par exemple, on peut utiliser le **rapport signal sur bruit** ou **SNR** (*signal to noise ratio*)

$$\text{SNR}(\mathbf{A}^*, \hat{\mathbf{A}}) = 20 \log_{10} \left( \frac{\|\mathbf{A}^*\|}{\|\hat{\mathbf{A}} - \mathbf{A}^*\|} \right), \quad (2.48)$$

ou encore l'**erreur quadratique moyenne** ou **MSE** (*mean squared error*)

$$\text{MSE}(\mathbf{A}^*, \hat{\mathbf{A}}) = \frac{1}{DK} \|\hat{\mathbf{A}} - \mathbf{A}^*\|^2 \quad (2.49)$$

avec  $D$  le nombre de voxels et  $K$  le nombre d'éléments de l'objet. Dans le cas où les données originales ne sont pas disponibles, on peut avoir recours à des estimateurs d'erreur [CHAUX et collab., 2008; CHOUZENOUX et collab., 2011; DELEDALLE et collab., 2013]. Par ailleurs, pour le cas des images, on peut calculer le **rapport contraste sur bruit** ou **CNR** (*contrast to noise ratio*), qui quantifie le bruit de manière plus spécifique au sein d'une image, comme suit :

$$\text{CNR}(X, Y) = \frac{\mu_X - \mu_Y}{\sqrt{\sigma_X^2 + \sigma_Y^2}} \quad (2.50)$$

où  $X$  et  $Y$  sont des régions de l'image et  $\mu$  et  $\sigma^2$  sont respectivement une moyenne et une variance. On donnera dans la suite quelques exemples de fonctions de régularisation souvent rencontrées en problèmes inverses.

### 2.3.2.1 Contraintes

Le problème de minimisation (1.45) permet de prendre en compte des contraintes lorsque la fonction de régularisation  $R$  correspond à la fonction indicatrice d'un sous-ensemble  $C \subset \mathbb{R}^n$  définie comme suit :

$$\chi_C(x) = \begin{cases} 0 & \text{si } x \in C, \\ +\infty & \text{sinon.} \end{cases} \quad (2.51)$$

Ici,  $C$  représente un ensemble de contraintes, qui peut s'écrire comme une intersection de contraintes distinctes. Par exemple, on peut considérer des **contraintes de boîte** de la forme

$$C = \{x \in \mathbb{R}^n \mid a_i \leq x_i \leq b_i, \forall i = 1 \dots N\}, \quad (2.52)$$

où les vecteurs  $a$  et  $b$  appartiennent à  $\mathbb{R}^n$ . Remarquons qu'en choisissant  $a = b$ , on peut considérer des contraintes d'égalité.

Dans notre cas nous cherchons à reconstruire une carte de concentrations d'éléments, par conséquent nous cherchons une quantité physique positive. On utilise donc une contrainte de la forme

$$C = \{x \in \mathbb{R}^n \mid 0 \leq x\}. \quad (2.53)$$

En utilisant ce type de régularisation nous n'avons plus besoin de l'hyper paramètre  $\lambda$ , ce qui peut être utile dans les cas où nous n'avons pas accès aux estimateurs de type (2.49) et (2.48).

### 2.3.2.2 Parcimonie

Un signal est dit **parcimonieux** si la plupart de ses composantes sont (approximativement) nulles. Certains signaux sont parcimonieux dans leur domaine d'acquisition comme les signaux sismiques par exemple (figure 2.5), d'autres sont parcimonieux après application d'un opérateur comme la transformée de Fourier par exemple.

Dans notre cas les éléments que l'on cherche à estimer peuvent être parcimonieux dans le domaine de l'objet comme les agents de contraste car ceux-ci ne sont présent que dans une petite partie de l'objet comme on peut le voir en figure 2.5. Dans le cas où le signal subit une transformation par le biais d'un opérateur on considèrera une fonction de régularisation de la forme

$$R(x) = \sum_{j=1}^J R_j(F_j x) \quad (2.54)$$

où, pour tout  $j \in \{1, \dots, J\}$   $F_j \in \mathbb{R}^{M \times N}$ . Ici, le signal sera supposé parcimonieux après application des matrices  $\{F_j\}_{j=1 \dots J}$ , c'est-à-dire que ce signal aura un nombre important de coefficients  $(F_j x)_i$  nuls.

Il existe plusieurs choix de fonctions  $R_j$  permettant de promouvoir la parcimonie d'un signal. Nous en donnerons ici quelques exemples. Intuitivement, la fonction permettant de favoriser au

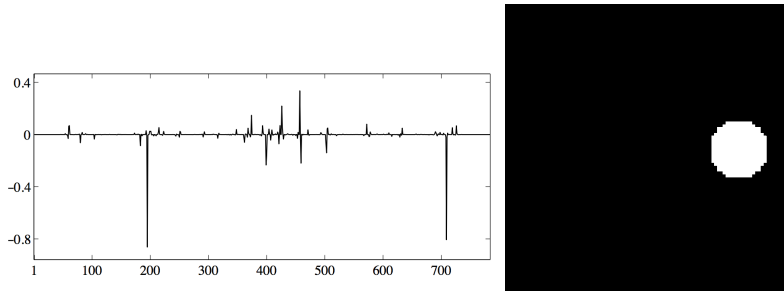


FIGURE 2.5 – La figure de gauche donne un exemple d'un signal sismique tandis que la figure de droite représente une coupe d'un objet contenant un agent de contraste. On remarque qu'il y a très peu d'information dans ces signaux : sur la figure de gauche les points temporel, à droite les pixels.

mieux la parcimonie d'un vecteur  $x \in \mathbb{R}^n$  est la **pseudo-norme**  $l_0$  [DONOHO et collab., 1995] qui compte le nombre de coefficients non nuls de  $x$  :

$$l_0 = \|x\|_0 = \sum_{n=1}^N \delta(x_n) \quad (2.55)$$

où la fonction  $\delta : \mathbb{R} \rightarrow \{0, 1\}$  est définie  $\forall a \in \mathbb{R}$  par

$$\delta(a) = \begin{cases} 0 & \text{si } a = 0 \\ 1 & \text{sinon.} \end{cases} \quad (2.56)$$

De façon plus générale, les fonctions  $l_\alpha$  [BOUMAN et SAUER, 1996], pour  $0 \leq \alpha < 1$ , sont des fonctions de régularisation permettant de promouvoir la parcimonie d'un objet, elles sont définies comme suit :

$$l_\alpha = \|x\|_\alpha = \left( \sum_{n=1}^N |x_n|^\alpha \right)^{\frac{1}{\alpha}}, \quad (2.57)$$

quand  $\alpha \neq 0$ . Cependant, ces fonctions ne sont ni différentiables ni convexes et peuvent donc se révéler difficiles à optimiser en pratique. Une façon usuelle de réduire la difficulté du problème est d'utiliser la norme  $l_1$  [BECT et collab., 2004; DONOHO, 2006; FIGUEIREDO et collab., 2007] comme approximation convexe des pénalisations ci-dessus :

$$l_1 = \|x\|_1 = \sum_{n=1}^N |x_n|. \quad (2.58)$$

La figure 2.6 représente différentes fonctions  $l_\alpha$ , pour  $0 \leq \alpha < 1$ . Nous pouvons voir que plus  $\alpha$  tend vers 0, moins les coefficients proches de 0 seront pénalisés. La norme  $l_1$  est la norme que

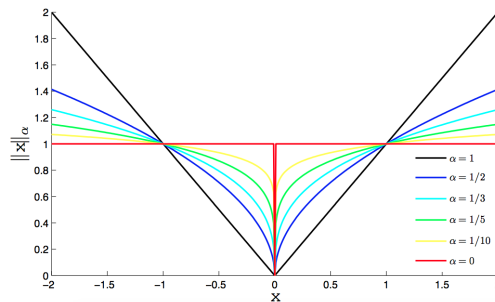


FIGURE 2.6 – Fonctions  $l_\alpha$  pour  $\alpha \in \{0, \frac{1}{10}, \frac{1}{5}, \frac{1}{3}, \frac{1}{2}, 1\}$ .

nous avons utilisé durant nos expériences car elle permet de s'affranchir de la non-convexité de la régularisation. Mais cette fonction demeure néanmoins non différentiable. Nous avons vu en sous-sections 2.2.1 et 2.2.2 comment pallier ce problème.

### 2.3.2.3 Variation totale

Une des fonctions de régularisation les plus utilisées en traitement des images est la **variation totale** (*total variation TV*) introduite dans [RUDIN et collab., 1992] pour des données continues. Elle mesure la parcimonie des gradients de l'image à l'aide de la norme  $l_1$ . Utilisée comme régularisation elle permet d'obtenir des images lisses par morceau séparés par des bords nets. Elle donne donc une idée de la quantité de contours dans une image. La version discrète proposée notamment dans [CHAMBOLLE, 2004] est définie par :

#### Définition 2.3.2.1: norme TV

Soit  $x \in \mathbb{R}^{N_1 \times N_2}$  une matrice modélisant une image de dimension  $N = N_1 \times N_2$ . La **variation totale** de  $x$  est donnée par

$$R(x) = tv(x) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \sqrt{([\nabla_v x]_{(i,j)})^2 + ([\nabla_h x]_{(i,j)})^2} \quad (2.59)$$

où  $[\nabla_v x]_{(i,j)}$  et  $[\nabla_h x]_{(i,j)}$  appartiennent à  $\mathbb{R}^{N_1 \times N_2}$ . Ce sont les gradients verticaux et horizontaux de  $x = \{x_{i,j}\}_{\substack{i=1,\dots,N_1 \\ j=1,\dots,N_2}}$  définis par

$$[\nabla_v x]_{(i,j)} = \begin{cases} x_{i+1,j} - x_{i,j} & \text{si } i < N_1 \\ 0 & \text{sinon,} \end{cases}$$

et

$$[\nabla_h x]_{(i,j)} = \begin{cases} x_{i,j+1} - x_{i,j} & \text{si } j < N_2 \\ 0 & \text{sinon.} \end{cases}$$

Remarquons que dans la définition précédente nous avons fait l'hypothèse que les coefficients se trouvant sur les bords droit et inférieur de l'image sont égaux à 0. Cependant, d'autres choix sont possibles pour la valeur de ces coefficients (par exemple, l'hypothèse de périodicité ou de symétrie miroir).

## Conclusion

Nous avons introduit dans cette sous-section les différentes fonctions objectifs que l'on peut être amené à étudier en **TDM** spectrale. Ces fonctions dépendent du bruit de mesure considéré et des contraintes supposées qui donnent des régularisations différentes.

Nous avons présenté jusqu'ici différents outils d'analyse qui vont nous permettre de résoudre le problème de **TDM** spectrale. Nous allons maintenant nous concentrer sur les méthodes pour résoudre ce type de problème. L'objet de la prochaine section est donc de présenter quelques algorithmes de résolution en problème inverse.

## 2.4 Algorithmes d'optimisation

Cette section a pour but de présenter des méthodes d'optimisation de base permettant de résoudre des problèmes variationnels usuels.

Les algorithmes itératifs cherchent un **minimum** d'une fonction  $f$  en générant une suite de points  $\{x_k\}_{k \geq 1}$ , appelés **itérés**, qui approchent de mieux en mieux un minimiseur  $x^*$  de  $f$ . Nous nous intéresserons dans un premier temps à la famille des **algorithmes à direction de descente** « classiques ». En effet, on peut utiliser des méthodes du **premier ordre** telle que la **descente de gradient** ou du **second ordre** telle la **descente de Newton** ou de **Gauss-Newton** [NOCEDAL et WRIGHT, 2006] afin de trouver une solution.

Nous parlerons ensuite des **algorithmes à régions de confiance** [MORÉ, 1983]. Ces méthodes s'apparentent aux algorithmes à direction de descente mais elles améliorent légèrement leurs vitesses

de convergence en essayant d'adapter la taille des régions dans laquelle l'optimisation est effectuée.

Nous verrons enfin les **algorithmes proximaux** [PARIKH et BOYD, 2014], une famille de méthodes qui peuvent s'apparenter à une descente de sous-gradient implicite et s'avérer efficaces lorsqu'on minimise une fonction non différentiable.

### 2.4.1 Algorithmes à direction de descente

Les algorithmes à direction de descente, qui s'appliquent à des fonctions différentiables, construisent une suite d'itérés en procédant en deux étapes :

- le calcul d'une **direction de descente**  $d_k$ ,
- la détermination d'un pas  $\gamma_k > 0$  le long de la direction de descente de telle sorte à avoir

$$f(x_{k+1}) \leq f(x_k)$$

La mise à jour de l'itéré se fait par le calcul suivant

$$x_{k+1} = x_k + d_k, \tag{2.60}$$

où  $d_k$  a pour forme

$$d_k = -\gamma_k \nabla f(x_k). \tag{2.61}$$

Nous verrons aussi qu'il existe des formes plus générales s'écrivant

$$d_k = -B_k^{-1} \nabla f(x_k), \tag{2.62}$$

où  $B_k$  est une matrice symétrique inversible. La direction de descente  $d_k$  caractérise le nom de l'algorithme utilisé. Ces méthodes diffèrent les unes des autres par rapport au choix de la matrice  $B_k$ . Dans la suite, nous allons voir quelques une de ces méthodes.

#### 2.4.1.1 Descente de gradient

L'algorithme de **descente de gradient**<sup>1</sup> est l'un des algorithmes les plus populaires de par sa simplicité d'implémentation et ses résultats de convergence théoriques que l'on peut retrouver dans [NOCEDAL et WRIGHT, 2006].

C'est un algorithme qui s'applique lorsque la fonction à minimiser est différentiable. Il est par conséquent destiné à résoudre un problème d'optimisation de la forme

$$\hat{x} \in \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} h(x), \tag{2.63}$$

où  $h : \mathbb{R}^n \rightarrow ]-\infty, +\infty]$  est une fonction réelle différentiable définie sur un espace hilbertien. On prendra ici l'espace  $\mathbb{R}^n$ .

Les techniques de base de minimisation dans un tel cas consistent à utiliser la caractérisation des points critiques vue en sous-section 2.1.3.3. En effet, annuler le gradient est une méthode simple pour trouver un minimiseur en résolvant, lorsque cela est possible, l'équation

$$\mathbf{0} = \nabla h(x). \tag{2.64}$$

Cependant, ceci peut être difficile à résoudre dès lors que  $h$  a une forme compliquée que l'on ne sait pas résoudre, comme par exemple les polynômes de degré supérieur ou égal à 5 [WANTZEL, 1845]. L'idée de la descente du gradient est alors de partir d'un point et d'aller dans la **direction opposée** au gradient en procédant de manière itérative et par améliorations successives et ainsi faire décroître la fonction. La direction de descente utilisée ici est de la forme  $d_k = -\gamma_k \nabla f(x_k)$ . Le schéma principal de l'algorithme est en Algorithme 1.

Le déplacement le long de la direction de descente est déterminé par un scalaire appelé le **pas de descente**  $\gamma_k$ , il peut être :

1. ou algorithme de la plus **forte pente** ou de la plus **profonde descente**.

---

**Algorithm 1** Algorithme du gradient

---

**Require:** Soient  $x_0 \in \mathbb{R}^n$ ,  $\gamma_0 > 0$

**for**  $k = 0, 1, 2, 3..$  **do**

$$x_{k+1} = x_k - \gamma_k \nabla h(x_k) \quad \triangleright \text{(e1)}$$

Calcul du pas  $\gamma_{k+1} > 0$   $\triangleright \text{(e2)}$

**end for**

---

- **fixe** dans un intervalle défini,
- **variable** dans un intervalle défini,
- **optimal**, *i.e.* construit de manière à accélérer la convergence de l'algorithme.

Chacun des choix du type de pas donne lieu à un algorithme particulier (gradient à pas fixe, gradient à pas optimal, etc.). On trouvera davantage d'informations dans [NOCEDAL et WRIGHT, 2006].

Notez que le cas optimal consiste à calculer le pas de descente à l'aide d'une règle de **recherche linéaire** sur  $h$  en  $x_k$  le long de la direction  $-\nabla h(x_k)$ . Pour cela, il existe plusieurs techniques pour optimiser la descente de gradient (critère d'Armijo-Goldstein).

La descente de gradient a de remarquables points forts. En effet, on peut appliquer cette méthode à une large famille de problèmes où la convergence vers un point critique (ou un minimiseur) est garantie [AVRIEL, 2003]. De plus, un algorithme à pas constant ne nécessite qu'une évaluation de  $\nabla h(x)$  à chaque itération. Lorsque le pas est variable ou optimal, cela nécessite un calcul supplémentaire mais qui est en général peu coûteux.

Néanmoins elle présente des points faibles, son principal étant la **faible vitesse de convergence** même dans le cas d'une fonction convexe. De plus, cette méthode est très sensible au **conditionnement** du problème (voir définition 2.1.3.1 page 73). Ainsi elle est peu efficace pour résoudre des problèmes mal-posés où une solution de grande précision est requise.

Un autre problème se pose aussi lorsque les fonctions étudiées sont non différentiables. Notons toutefois qu'il existe des méthodes de **descente de sous-gradient** pour contourner ce problème, mais elles sont en pratique plus lentes qu'une descente de gradient classique [KIWIEL, 2001].

Comme nous l'avons mentionné en introduction, les méthodes précédentes s'inscrivent dans le cadre de méthodes de **premier ordre**, c'est-à-dire qu'on n'exploite que la première dérivée de la fonction objectif afin d'en étudier les variations. On peut être plus précis et accélérer la convergence si on exploite l'information de **second ordre**, c'est l'idée de base des méthodes du paragraphe suivant.

### 2.4.1.2 Descente de Newton

La **descente de Newton** peut être vue comme une amélioration de la descente de gradient classique. Pour comprendre comment cela fonctionne nous nous concentrons sur le problème d'optimisation suivant

$$\hat{x} \in \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} h(x), \quad (2.65)$$

où  $h : \mathbb{R}^n \rightarrow ]-\infty, +\infty]$  est une fonction réelle deux fois différentiable. On se sert alors du développement de Taylor à l'ordre deux de  $h$  (voir (2.16)) au point  $x_k$

$$h(x) \approx h(x_k) + \nabla h(x_k)^t (x - x_k) + \frac{1}{2} (x - x_k)^t \nabla^2 h(x_k) (x - x_k). \quad (2.66)$$

C'est une **approximation quadratique** de la fonction  $h$  au point  $x_k$ . Elle est facile à minimiser et son minimum a une expression analytique qui donne la **direction de descente** de la méthode de



Newton

$$d_k = -\nabla^2 h(x_k)^{-1} \nabla h(x_k). \quad (2.67)$$

Le schéma de l'algorithme est en Algorithme 2. Remarquons qu'un pas de descente peut être calculé ici aussi, celui-ci devrait être dans l'intervalle  $(0, 1)$  afin de respecter le critère de Wolf. On appelle alors cette méthode la descente de Newton **relaxée** [WU et collab., 2008].

La méthode de Newton peut s'interpréter géométriquement. En effet, à chaque point courant

---

**Algorithm 2** Algorithme de Newton

---

**Require:** Soit  $x_0 \in \mathbb{R}^n$ ,

**for**  $k = 0, 1, 2, 3..$  **do**

$$x_{k+1} = x_k - \nabla^2 h(x_k)^{-1} \nabla h(x_k) \quad \triangleright \text{(e1)}$$

**end for**

---

$x_k$  on calcule une **approximation quadratique** de la fonction  $h$  dans un voisinage  $V(x_k)$ , puis on prend la direction donnée par le minimum de cette approximation quadratique. On peut montrer qu'une telle méthode converge plus rapidement qu'une descente de gradient classique lorsque l'algorithme est initialisé dans le bassin d'attraction du minimum à atteindre [NOCEDAL et WRIGHT, 2006].

En effet, les méthodes de descente de Newton comptent parmi les outils les plus efficaces pour résoudre un problème d'optimisation différentiable et sans contrainte. Elles s'adaptent à la « géométrie locale » (**courbure**) de la fonction objectif et permettent ainsi une convergence plus rapide. Mais elles ne sont pas parfaites pour plusieurs raisons :

1. il n'y a aucune garantie que la matrice hessienne  $\nabla^2 h$  soit inversible,
2. même si celle-ci est inversible, le coût de calcul de son inverse peut être élevé, car cela reviendrait à résoudre un système linéaire qui peut être de grande dimension comme c'est le cas dans le problème de **TDM** spectrale,
3. la matrice hessienne peut présenter un conditionnement élevé, ainsi son utilisation peut sérieusement amplifier le bruit et dégrader le résultat.

Pour rendre cette méthode efficace, on peut effectuer des modifications sur la direction de descente et notamment sur le pas de descente. En effet, lorsque la hessienne n'est pas accessible ou très coûteuse en calcul on peut lui substituer des **opérateurs** plus simples à calculer et qui ont des propriétés similaires à la hessienne et permettent ainsi de préserver une vitesse de convergence quadratique. On peut citer la descente de **quasi Newton**, de **Gauss-Newton** [NOCEDAL et WRIGHT, 2006] ou encore celle de Levenberg-Marquardt [MARQUARDT, 1963].

### 2.4.1.3 Descente de gradient projeté

Jusqu'ici nous avons étudié des problèmes de minimisation différentiables sans contraintes. Nous allons voir dans ce paragraphe comment peut on résoudre un problème inverse de la forme

$$\hat{\mathbf{x}} \in \underset{x \in \Omega}{\operatorname{argmin}} h(x), \quad (2.68)$$

où  $h$  est une fonction différentiable et  $\Omega$  est un sous-ensemble de  $\mathbb{R}^n$  qui représente l'ensemble des contraintes. Il y a d'abord une question de **solvabilité** et de **faisabilité** qui se pose. On dit qu'une solution est **faisable** pour le problème (2.68) si celle-ci se trouve être dans  $\Omega$ . L'ensemble composé de ces solutions est appelé **ensemble faisable** du problème<sup>2</sup> (2.68).

On appelle un point  $x^*$  une **solution optimale** de (2.68) si c'est une solution faisable et si elle appartient à l'ensemble des minimiseurs de ce problème. Si un tel point existe, on dira alors que (2.68) est un **problème solvable**.

---

2. Notons aussi qu'un problème est **faisable** si son ensemble faisable est non vide.



Lorsque  $\Omega$  est un ensemble non vide, convexe et fermé, on peut appliquer ce qu'on appelle **une descente de gradient projetée**. Le principe est de calculer à chaque itération la descente de gradient classique puis son **projeté orthogonal** sur l'ensemble des contraintes. La description de l'algorithme est en Algorithme 3.

La fonction  $\mathcal{P}_\Omega(\cdot)$  à l'étape (e2) est la fonction de projection orthogonale sur  $\Omega$ , sa forme dépend des contraintes imposées. Les propriétés de la descente de gradient projetée ainsi que des résultats de convergence peuvent être trouvés dans [CALAMAI et MORÉ, 1987].

---

**Algorithm 3** Algorithme de gradient projeté

---

**Require:** Soit  $x_0 \in \mathbb{R}^n$ ,  $\gamma_0 > 0$

**for**  $k = 0, 1, 2, 3..$  **do**

$z = x_k - \gamma_k \nabla h(x_k)$  ▷ (e1)

$x_{k+1} = \mathcal{P}_\Omega(z)$  ▷ (e2)

Calcul du pas  $\gamma_{k+1} > 0$  ▷ (e3)

**end for**

---

#### 2.4.1.4 Conclusion

Nous avons vu au cours de cette sous-section les algorithmes à **direction de descente** qui consistent à utiliser une direction de descente de la forme

$$d_k = -B_k^{-1} \nabla h(x_k), \tag{2.69}$$

où la matrice  $B_k$  est choisie selon la méthode utilisée. Par exemple, c'est une descente de gradient classique si celle-ci vaut l'identité ou bien une descente de Newton si celle-ci est égale à la Hessienne de  $h$ . Ces méthodes peuvent être optimisées si on choisit un pas de descente adéquat à chaque itération.

On peut subdiviser ces méthodes en deux catégories : les descentes de premier ordre et de second ordre. L'avantage des descentes de premier ordre est la simplicité d'implémentation et le faible coût de calcul, mais la convergence vers une solution peut être lente. Les descentes de second ordre ont certes une convergence plus rapide, mais elles requièrent un coût de calcul élevé. Néanmoins, elles sont très utilisées en pratique et applicables à un vaste champ de problèmes inverses. Un autre problème de ces méthodes est que les garanties de convergence pour les fonctions non convexes sont moins fortes contrairement au cas convexe et différentiable. Or, on rappelle que le problème de TDM spectrale n'est pas convexe et les régularisations imposées sont non différentiables.

Nous allons voir dans la suite une méthode qui s'apparente à la descente de second ordre, mais qui peut offrir une amélioration de la vitesse de convergence.

#### 2.4.2 Algorithmes de région de confiance

Les **algorithmes de région de confiance** sont une classe d'algorithmes procédant de manière itérative et utilisant à chaque étape une **approximation quadratique** de la fonction objectif, un peu comme la descente de Newton vue en 2.4.1.2. La différence réside dans l'utilisation de cette approximation quadratique. En effet, dans la méthode de Newton, l'approximation quadratique est utilisée pour trouver une direction de descente. Tandis que dans les méthodes de région de confiance, on choisit de minimiser l'approximation quadratique sur un ensemble de points autour du point courant. Cet ensemble de points est appelé **la région de confiance**, car on suppose que dans cette région l'approximation quadratique est très proche de la fonction objectif. On modifie la taille de la région de confiance selon un critère de qualité qui mesure l'adéquation de l'approximation quadratique à la fonction objectif. La figure 2.7 illustre cette méthode.

De manière plus formelle, les méthodes de région de confiance s'attaquent à des problèmes de la

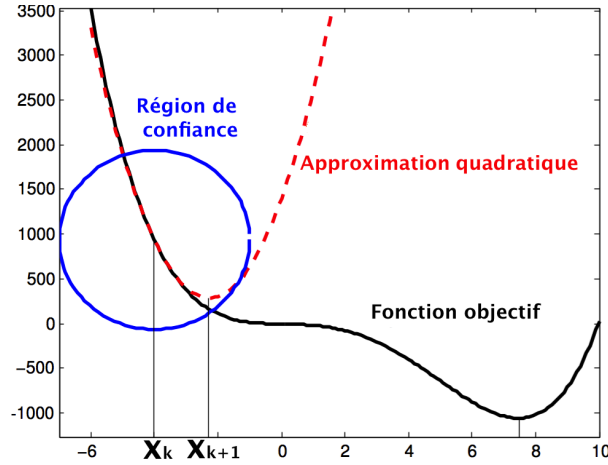


FIGURE 2.7 – Illustration de la méthode de région de confiance. La courbe noire représente fonction objectif à minimiser, la courbe rouge représente l'approximation quadratique à l'itéré  $x_k$  et le cercle bleu représente la région de confiance dans laquelle on minimise l'approximation quadratique. Cette minimisation nous conduit à l'itéré suivant  $x_{k+1}$ .

forme

$$\operatorname{argmin}_{x \in \mathbb{R}^n} f(x), \quad (2.70)$$

où  $f$  est une fonction différentiable. Le principe est de simplifier ce problème en résolvant un sous problème quadratique plus simple. Si on utilise le théorème vu en (2.16), on obtient un développement à l'ordre 2 de  $f$  au point  $x_k$  de  $f$  de la forme

$$f(x) \approx f(x_k) + \nabla f(x_k)^t (x - x_k) + \frac{1}{2} (x - x_k)^t \nabla^2 f(x_k) (x - x_k). \quad (2.71)$$

On peut généraliser cette approximation en remplaçant la hessienne  $\nabla^2 f(x_k)$  par une matrice  $B_k$  symétrique. On pose alors  $p = x - x_k$  et on appelle  $m_k(p)$  l'approximation quadratique ainsi obtenue s'écrivant

$$m_k(p) = f(x_k) + \nabla f(x_k)^t p + \frac{1}{2} p^t B_k p. \quad (2.72)$$

C'est la fonction  $m_k(p)$  qu'on vient donc minimiser à chaque itération au sein d'une région de taille  $\Delta_k$ . Ainsi le sous problème s'exprime comme suit :

$$\operatorname{argmin}_{p \in \mathbb{R}^n, \|p\| \leq \Delta_k} \{m_k(p)\}. \quad (2.73)$$

Le problème (2.73) est un problème contraint qui n'a pas de solution analytique. On va donc le résoudre de manière algorithmique. L'idée de l'algorithme est alors de calculer un point critique  $\hat{p}_k$  de (2.73) à chaque itération et de mettre à jour les itérés de la manière suivante :

$$x_{k+1} = x_k + \hat{p}_k. \quad (2.74)$$

Le choix de la taille de région  $\Delta_k$  est crucial pour la qualité du résultat, car il va déterminer si le point  $\hat{p}_k$  est acceptable pour une mise à jour ou non. Ce seuil d'acceptabilité est déterminé selon **un critère de qualité**, que l'on nomme  $\rho_k$ , calculé à chaque itération de la manière suivante :

$$\rho_k = \frac{f(x_k) - f(x_k + \hat{p}_k)}{m_k(0) - m_k(\hat{p}_k)}. \quad (2.75)$$

Le numérateur représente la **réduction réelle** de la fonction objectif et le dénominateur la **réduction approchée** obtenue avec l'approximation quadratique. Ce rapport mesure donc l'accord

entre la fonction objectif et l'approximation quadratique à l'itéré  $x_k$ . Ainsi lorsque celui-ci est proche de 1 on peut considérer que l'approximation quadratique (2.72) est très bonne et on peut donc augmenter la taille de la région de confiance et mettre à jour l'itéré  $x_{k+1}$ . Inversement, si celui-ci est très petit, on diminue la taille de la région de confiance sans mettre à jour l'itéré  $x_{k+1}$ , celui-ci ne sera mis à jour que si la qualité de l'approximation est bonne. Le schéma général de la méthode est donné en Algorithme 4.

Les étapes (e2), (e3), (e4) et (e5) de l'algorithme déterminent la taille de la région pour l'itération suivante selon que le critère de qualité  $\rho_k$  est plus ou moins proche de 1. L'étape (e6) est l'étape de mise à jour de l'itéré  $x_{k+1}$ . L'étape (e1) est l'étape de résolution du problème (2.73). Pour résoudre ce problème, il existe plusieurs méthodes. Une description détaillée de celles-ci se trouve dans [NOCEDAL et WRIGHT, 2006]. Nous allons brièvement mentionner dans la suite quelques une d'entre elles.

---

**Algorithm 4** Algorithme de région de confiance
 

---

**Require:** Soit  $\bar{\Delta} > 0$ ,  $\Delta_0 \in [0, \bar{\Delta}]$ , et  $\eta \in [0, \frac{1}{4}[$

```

for  $k = 0, 1, 2, 3..$  do
    Calculer  $\hat{p}_k$  en minimisant le sous problème (2.73)                                ▷ (e1)
    Calculer  $\rho_k$  à l'aide de (2.75)                                                ▷ (e2)
    if  $\rho_k < \frac{1}{4}$  then
         $\Delta_{k+1} = \frac{1}{4} \|\hat{p}_k\|$                                                 ▷ (e3)
    else
        if  $\rho_k > \frac{3}{4}$  and  $\Delta_k = \|\hat{p}_k\|$  then
             $\Delta_{k+1} = \min(2\Delta_k, \bar{\Delta})$                                         ▷ (e4)
        else
             $\Delta_{k+1} = \Delta_k$                                                     ▷ (e5)
        end if
    end if
    if  $\rho_k > \eta$  then
         $x_{k+1} = x_k + \hat{p}_k$                                                         ▷ (e6)
    else
         $x_{k+1} = x_k$ 
    end if
end for
    
```

---

### 2.4.2.1 Solution « exacte »

Une solution « exacte » à (2.73) peut être trouvée en utilisant un algorithme décrit dans [MORÉ et SORENSEN, 1983]. Cet algorithme vise à trouver précisément le minimum du modèle quadratique  $m_k$  en résolvant le problème inverse en grande dimension. On peut utiliser n'importe quelle matrice  $B_k$  symétrique pour construire le modèle quadratique. Notons que si on prend la hessienne, alors la direction de descente est celle de Newton. Cette technique est très efficace car elle donne des résultats d'une grande précision et converge en peu d'itérations. Cependant, elle est très coûteuse, car elle fait appel à la **factorisation de Cholesky** à chaque itération [NOCEDAL et WRIGHT, 2006].

Résoudre le problème de minimisation de l'approximation quadratique sur l'espace entier des solutions, qui peut être de grande dimension, peut se révéler difficile dans le cadre de la **TDM** spectrale comme nous le verrons au chapitre 3. On peut alors simplifier le problème en réduisant la dimension de l'espace des solutions. C'est l'idée de base des deux méthodes que l'on va présenter au paragraphe suivant.

### 2.4.2.2 Méthode de la patte de chien ou *dog-leg* et de la minimisation en sous-espace à deux dimensions

Comme nous l'avons dit, afin de simplifier la résolution du problème (2.73), on réduit la dimension de l'espace des solutions tout en veillant à avoir une solution acceptable. On peut réécrire le problème (2.73) en

$$\operatorname{argmin}_{p \in \Omega, \|p\| \leq \Delta_k} \{m_k(p)\}, \quad (2.76)$$

où  $\Omega$  est un sous-espace de  $\mathbb{R}^n$  de petite dimension.

#### *Dog-leg*

La méthode de la **patte de chien** ou *dog-leg* consiste à utiliser deux directions de descentes successives. On pose

$$d_1 = -B_k^{-1} \nabla f(x_k), \quad d_2 = -\frac{\|\nabla f(x_k)\|^2}{\nabla f(x_k)^t B_k \nabla f(x_k)} \nabla f(x_k). \quad (2.77)$$

On remarque immédiatement que  $d_1$  est similaire à une direction de descente de Newton et  $d_2$  une direction de descente de gradient. L'idée est de combiner les deux directions en choisissant un pas de descente adéquat. La descente du *dog-leg*, noté  $\tilde{p}$ , consiste alors à descendre dans un premier temps dans la direction  $d_2$  puis dans la direction  $d_1$

$$\tilde{p}(\tau) = \begin{cases} \tau d_2 & \text{pour } \tau \in [0, 1] \\ d_2 + (\tau - 1)(d_1 - d_2) & \text{pour } \tau \in [1, 2]. \end{cases} \quad (2.78)$$

La descente  $\tilde{p}$  est simple à calculer, mais les résultats obtenus peuvent être améliorés en calculant, par exemple, une solution exacte du problème (2.76), ce que nous allons voir dans le paragraphe suivant.

#### Minimisation en sous-espace 2D

On peut améliorer la méthode précédente en élargissant la recherche de  $\hat{p}_k$  à l'ensemble du **sous-espace bidimensionnel** généré par les directions  $d_1$  et  $d_2$ . Formellement, il s'agit de trouver un vecteur qui s'écrive comme combinaison linéaire des deux directions, *i.e.* de la forme

$$p = \alpha d_1 + \beta d_2, \quad (2.79)$$

avec  $(\alpha, \beta) \in \mathbb{R}^2$ . On montre alors (voir annexe A.1.4) que résoudre (2.76) sous la contrainte (2.79) revient à résoudre

$$\operatorname{argmin}_{\theta \in \mathbb{R}^2, \|\theta\| \leq \Delta_k} \{\tilde{m}_k(\theta) = f(x_k) + \tilde{G}^t \theta + \frac{1}{2} \theta^t \tilde{A} \theta\}, \quad (2.80)$$

avec

- $\theta = \sqrt{D} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ ,
- $\tilde{G}^t = G^t (\sqrt{D})^{-1}$ ,
- $\tilde{A} = (\sqrt{D})^{-1} A (\sqrt{D})^{-1}$ ,

où  $A$ ,  $D$  et  $G$  sont définis comme suit :

$$G = \begin{pmatrix} \nabla f(x_k)^t d_1 \\ \nabla f(x_k)^t d_2 \end{pmatrix} \quad A = \begin{pmatrix} d_1^t B_k d_1 & d_1^t B_k d_2 \\ d_1^t B_k d_2 & d_2^t B_k d_2 \end{pmatrix} \quad D = \begin{pmatrix} d_1^t d_1 & d_1^t d_2 \\ d_1^t d_2 & d_2^t d_2 \end{pmatrix}$$

De plus, on peut traiter le cas lorsque la matrice  $B_k$  a des valeurs propres négatives en lui ajoutant un paramètre de relaxation  $\lambda$  choisi de sorte que la matrice

$$B_k + \lambda \mathbf{I}$$

soit définie positive [NOCEDAL et WRIGHT, 2006]. Cette méthode est très efficace, mais peut être coûteuse car trouver le bon paramètre  $\lambda$  peut nécessiter le calcul des valeurs propres de  $B_k$ . Dans la suite, nous allons voir une dernière méthode simple et non coûteuse pour résoudre le problème (2.73).

### 2.4.2.3 Point de Cauchy

Une des manière les plus classique est de calculer une descente selon la direction du gradient avec un pas dépendant de la région de confiance, c'est ce qu'on appelle le **point de Cauchy**. Pour cela il suffit de ne pas tenir compte de l'information de second ordre et donc de minimiser une approximation du premier ordre de la forme

$$\operatorname{argmin}_{p \in \mathbb{R}^n, \|p\| \leq \Delta_k} \{f(x_k) + \nabla f(x_k)^t p\}. \quad (2.81)$$

Ceci donne la direction de descente suivante :

$$p_{\text{Cauchy}} = -\tau_k \frac{\Delta_k}{\|\nabla f(x_k)\|} \nabla f(x_k), \quad (2.82)$$

où le pas de descente  $\tau_k$ , qui respecte la contrainte de la taille de région de confiance, est calculé comme suit :

$$\tau_k = \begin{cases} 1 & \text{si } \nabla f(x_k)^t B_k \nabla f(x_k) \leq 0, \\ \min\left(1, \frac{\|\nabla f(x_k)\|^3}{\Delta_k \nabla f(x_k)^t B_k \nabla f(x_k)}\right) & \text{sinon.} \end{cases} \quad (2.83)$$

Ainsi, la mise à jour de l'itéré se fait par le calcul suivant

$$x_{k+1} = x_k + p_{\text{Cauchy}}. \quad (2.84)$$

Le point de Cauchy est simple et peu coûteux en calcul mais il est aussi efficace qu'une descente de gradient et on a vu dans 2.4.1.1 que ce type de descente ne donne pas des résultats très satisfaisants. D'autre part, le point de Cauchy ne tient pas compte de l'information de second ordre présente dans le modèle quadratique et particulièrement dans la matrice  $B_k$ , alors que celle-ci joue un rôle important dans la vitesse de convergence.

Nous avons décrit trois méthodes de résolution du problème (2.73). Ces méthodes peuvent être classées selon la complexité du calcul qu'elles requièrent. Ainsi, la méthode de Cauchy vue en 2.4.2.3 est peu coûteuse, mais donne des résultats insatisfaisants tandis que la méthode de solution exacte vue en 2.4.2.1 donne des résultats très satisfaisants, mais s'avère être coûteuse en calcul. Les méthodes en 2.4.2.2 sont des méthodes intermédiaires de minimisation en sous-espace à dimensions réduites et elles donnent un bon compromis entre la qualité du résultat et le coût de calcul.

### 2.4.2.4 Remarque sur les contraintes

Il est possible d'intégrer au problème (2.73) des contraintes de régularisation particulières. Nous verrons au chapitre 3 que les contraintes que nous avons appliquées sont des **contraintes de boîte** ( voir paragraphe 2.3.2.1) de la forme

$$C_P = \{x \in \mathbb{R}^n \mid 0 \leq x_i, \forall i = 1 \dots N\}. \quad (2.85)$$

La contrainte  $C_P$  est une contrainte de **positivité** qui force les valeurs d'un vecteur à être positives ou nulles. Ainsi, le problème (2.73) peut se réécrire comme suit :

$$\operatorname{argmin}_{p \in C_P, \|p\| \leq \Delta_k} \{m_k(p)\}. \quad (2.86)$$

Notons qu'il existe d'autres méthodes plus élaborées pour intégrer des contraintes de régularisation, mais celles-ci doivent être différentiables [CARTIS et collab., 2012; ZHANG, 2004].

### 2.4.2.5 Conclusion

Nous avons vu dans la précédente sous-section les méthodes de région de confiance. Elles peuvent être vues comme une amélioration des méthodes de descente de gradient et peuvent contribuer à l'accélération de la convergence. L'inconvénient de ces méthodes est qu'elles ne permettent d'intégrer que des contraintes de boîte, comme vu au paragraphe 2.4.2.4. Il est très difficile d'intégrer des régularisations un peu plus sophistiquées correspondant à des contraintes plus réalistes pour notre problème de **TDM** spectrale, comme des contraintes de parcimonie par exemple.

Dans la suite, nous allons voir des méthodes qui combinent les deux familles de méthodes vues précédemment. L'idée est d'utiliser une descente de gradient facile à implémenter, mais qui peut être accélérée et, surtout, appliquée à des contraintes de régularisation non différentiables.

## 2.4.3 Algorithmes proximaux

### 2.4.3.1 Introduction

Il existe de nombreuses raisons d'étudier les algorithmes proximaux. D'abord, ils sont utiles dans des conditions extrêmement générales, y compris les cas où les fonctions sont non différentiables à valeur réelle. Ils peuvent être rapides, puisqu'il peut y avoir de simples opérateurs proximaux pour certaines fonctions. Ils sont mathématiquement simples, faciles à mettre en œuvre pour un problème particulier comme celui impliquant la régularisation  $l_1$ .

En effet, dans certains cas particuliers, les algorithmes proximaux peuvent être assimilés à des algorithmes bien connus et largement utilisés, comme la méthode du gradient projeté. Ils sont donc un outil assez commun à l'optimisation générale. Un autre avantage de ces méthodes est qu'elles permettent d'incorporer des contraintes de régularisation non différentiables. Ces contraintes de régularisation sont plus adaptées aux problèmes de la **TDM** spectrale, contrairement aux méthodes de région de confiance vues en 2.4.2, car il est possible de minimiser ce type de fonction à l'aide de l'**opérateur proximal** vu en 2.2.2 [COMBETTES et PESQUET, 2010].

Dans cette sous-section nous allons parler de quatre algorithmes proximaux utilisés durant ce travail de thèse. Nous parlerons dans un premier temps des algorithmes **ISTA** et **FISTA**, qui sont incontournables de par leur simplicité. Ensuite nous introduirons un algorithme proximal à métrique variable, qui permet d'accélérer la convergence. Enfin nous parlerons de l'algorithme **VMI-LAn**, qui a permis d'avoir des résultats encourageants sur données réelles (voir chapitre 3).

### 2.4.3.2 Algorithme ISTA

On s'intéresse au problème de la forme

$$\operatorname{arg\,min}_{x \in \mathbb{R}^n} \{f(x) = h(x) + g(x)\}, \quad (2.87)$$

où  $h$  est le terme d'attache aux données supposé ici convexe et  $g$  le terme de régularisation lui aussi convexe, mais non nécessairement différentiable. On suppose ici que  $h$  est continue et différentiable de gradient  $L_h$ -Lipschitz (voir définition (2.20)).

Il existe une classe d'algorithmes capables de résoudre ce genre de problème. Ce sont les algorithmes **explicites-implicites** (en anglais *forward-backward*). Ils sont composés principalement de deux étapes : une étape **explicite** de calcul de gradient sur la partie différentiable de la fonction  $f$  et une étape **implicite** de calcul du point proximal sur la partie non différentiable de  $f$ .

L'algorithme **ISTA** (*Iterative Shrinkage-Thresholding Algorithm*) est un cas particulier d'algorithme explicite-implicite connu pour sa simplicité et ses résultats théoriques de convergence. Il a été utilisé dans de nombreux cas pratiques et sa convergence est démontrée pour divers problèmes inverses [COMBETTES et WAJS, 2005; DAUBECHIES et collab., 2004; FIGUEIREDO et NOWAK,

2003].

En effet, l'algorithme **ISTA** est utilisé lorsque le terme de régularisation  $g$  est une pénalisation  $l_1$ . Pour cette raison, l'algorithme **ISTA** peut être assimilé à une descente de gradient projeté. Nous allons le montrer dans la suite.

On rappelle que la mise à jour d'un itéré vue en 2.4.1.1 pour la descente de gradient est

$$x_{k+1} = x_k - \gamma_k \nabla h(x_k). \quad (2.88)$$

Il est démontré dans [BERTSEKAS, 1999] que l'équation (2.88) peut être vue comme un point proximal calculé à partir d'une **approximation à l'ordre 1** de la fonction  $h$  au point  $x_k$ , que l'on peut réécrire

$$x_{k+1} = \arg \min_{x \in \mathbb{R}^n} \left\{ h(x_k) + \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} \|x - x_k\|^2 \right\}. \quad (2.89)$$

Ceci est la mise à jour pour le cas où on ne minimise que la fonction  $h$ . Dans le cas de l'algorithme **ISTA**, on utilise la régularisation  $l_1$ . On introduit alors ce terme dans l'expression (2.89) pour obtenir la mise à jour suivante :

$$x_{k+1} = \arg \min_{x \in \mathbb{R}^n} \left\{ h(x_k) + \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} \|x - x_k\|^2 + \lambda \|x\|_1 \right\}, \quad (2.90)$$

où  $\lambda$  est l'hyperparamètre de régularisation. On peut alors montrer par un calcul simple (voir annexe A.1.5) que (2.90) peut se réécrire comme suit :

$$x_{k+1} = \arg \min_{x \in \mathbb{R}^n} \left\{ \lambda \|x\|_1 + \frac{1}{2\gamma_k} \|x - (x_k - \gamma_k \nabla h(x_k))\|^2 \right\}, \quad (2.91)$$

ce qui, d'après la définition (2.23), peut s'écrire

$$x_{k+1} = \text{prox}_{\gamma_k, \lambda \|\cdot\|_1} (x_k - \gamma_k \nabla h(x_k)). \quad (2.92)$$

Or nous avons vu avec la définition (2.35) que ceci revient à appliquer l'opérateur de **seuillage doux** à la descente de gradient  $(x_k - \gamma_k \nabla h(x_k))$ . On peut donc ainsi réécrire la mise à jour comme

$$x_{k+1} = \text{soft}_\lambda (x_k - \gamma_k \nabla h(x_k)), \quad (2.93)$$

qui a la même forme que la mise à jour dans le cas de l'algorithme à descente de gradient projeté vu en 3.

### Généralisation : algorithme explicite-implicite

On peut généraliser le résultat (2.92) pour n'importe quelle fonction de régularisation  $g$  convexe. Pour cela, on considère l'approximation quadratique de la fonction  $f(x) = h(x) + g(x)$  suivante :

$$Q_L(x, y) := h(y) + \nabla h(y)^t (x - y) + \frac{L}{2} \|x - y\|^2 + g(x), \quad (2.94)$$

où  $L \geq L_h > 0$ . L'approximation (2.94) admet un minimiseur, noté  $p_L(y)$ , que l'on définit comme suit :

$$p_L(y) = \arg \min_{x \in \mathbb{R}^n} \{Q_L(x, y)\}. \quad (2.95)$$

On montre alors par les mêmes calculs vus en annexe A.1.5 l'égalité

$$p_L(y) = \text{prox}_{\frac{1}{L}, g} \left( y - \frac{1}{L} \nabla h(y) \right) = \arg \min_{x \in \mathbb{R}^n} \left\{ g(x) + \frac{L}{2} \|x - \left( y - \frac{1}{L} \nabla h(y) \right)\|^2 \right\}. \quad (2.96)$$



Nous obtenons de la sorte la mise à jour de l'algorithme explicite-implicite en Algorithme 5.

Dans le cas de l'algorithme **ISTA**, il est prouvé dans [BECK et TEBoulLE, 2009; COMBETTES et WAJS, 2005] que lorsque l'approximation quadratique  $Q_L(x, y)$  **major**e la fonction  $f$  au point proximal  $p_L(y)$ , *i.e.*

$$f(p_L(y)) \leq Q(p_L(y), y), \quad (2.97)$$

on peut garantir la convergence de **ISTA**. Le théorème (2.98) donne alors un ordre de vitesse de convergence de la suite d'itérés générée par **ISTA**.

**Théorème 2.4.3.1: vitesse de convergence de l'algorithme ISTA**

Soit  $\{x_k\}_{k \geq 1}$  la suite générée par **ISTA**. Alors  $\forall k \geq 1$  :

$$f(x_k) - f(x^*) \leq \frac{L_h \|x_0 - x^*\|^2}{2k}, \quad (2.98)$$

où  $x^*$  appartient à l'ensemble des minimiseurs du problème (2.87) et  $x_0$  est le point d'initialisation de Algorithme 5.

On notera par ailleurs que le paramètre  $L_h$  n'est pas toujours accessible. On peut alors utiliser une règle de calcul supplémentaire, appelée **backtracking**, dans Algorithme 5 qui permet de trouver une estimation de ce paramètre qui respecte la condition (2.97).

Comme nous l'avons précédemment dit, l'algorithme **ISTA** a l'avantage d'être simple et permet de traiter des problèmes de minimisation avec une régularisation non nécessairement différentiable. Mais il est montré dans [BREDIES et LORENZ, 2008] que l'algorithme **ISTA** peut présenter des vitesses de convergence lentes. Il existe néanmoins une technique permettant d'accélérer la convergence de l'algorithme **ISTA** de manière significative dont nous allons parler dans le prochain paragraphe.

---

**Algorithm 5** Explicite-implicite

---

**Require:** Soit  $x_0 \in \mathbb{R}^n$ ,  $L \geq L_h$

**for**  $k = 0, 1, 2, 3..$  **do**

$$z = x_k - \frac{1}{L} \nabla h(x_k)$$

$$x_{k+1} = p_L(z)$$

▷ (e1)

**end for**

---

**2.4.3.3 Algorithme FISTA**

Dans le contexte de problèmes de grande taille, tel que celui rencontré en **TDM** spectrale, un des enjeux principaux est de pouvoir produire de bons résultats numériques en un temps de calcul raisonnable. L'algorithme explicite-implicite précédemment présenté est caractérisé par un faible coût de calcul à chaque itération. Cependant, comme la plupart des méthodes de minimisation de premier ordre, sa vitesse de convergence peut être lente. Il existe alors deux grandes familles de stratégies d'accélération de cet algorithme dans la littérature :

- la première repose sur une accélération utilisant les informations obtenues lors des itérations précédentes pour construire le nouvel itéré comme l'algorithme que nous allons présenter ici,
- la seconde stratégie d'accélération est basée sur l'introduction d'une **métrique** qui peut varier au cours des itérations et que l'on présentera dans la sous-section 2.4.3.4.

L'un des algorithmes proximaux les plus populaires de l'état de l'art est celui présenté dans [BECK et TEBoulLE, 2009] : **FISTA** (*Iterative Shrinkage-Thresholding Algorithm*). La simplicité computationnelle de l'algorithme ainsi que sa vitesse de convergence ont contribué à le populariser.



L'idée fondamentale de **FISTA** est d'appliquer la descente de gradient proximal à un point, noté  $y_k$ , construit à partir des itérations précédentes et facile à calculer permettant d'accélérer la vitesse de convergence [BIOUCAS-DIAS et FIGUEIREDO, 2007; NESTEROV et collab., 2013; OCHS et collab., 2014].

Ce choix, inspiré de la méthode de [NESTEROV, 1983], consiste à utiliser un « effet mémoire » en calculant à chaque itération un point défini à partir de la combinaison linéaire de l'itéré **actuel** ainsi que de l'itéré **précédent**. Comme on le voit à l'étape (e2) de l'Algorithme 6, ce calcul est très peu coûteux. Il est démontré dans [BECK et TBOULLE, 2009] que sous les mêmes hypothèses vé-

---

**Algorithm 6** FISTA
 

---

**Require:**  $y_0 = x_0 \in \mathbb{R}^n$ ,  $L \geq L_h$ ,  $t_0 = 1$

**for**  $k = 0, 1, 2, 3..$  **do**

$$z = y_k - \frac{1}{L} \nabla h(y_k)$$

$$x_{k+1} = p_L(z) \quad \triangleright \text{(e1)}$$

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$$

$$y_{k+1} = x_{k+1} + \left( \frac{t_k - 1}{t_{k+1}} \right) (x_{k+1} - x_k) \quad \triangleright \text{(e2)}$$

**end for**

---

rifiées par (2.97), l'algorithme **FISTA** converge théoriquement deux fois plus vite que l'algorithme **ISTA** comme le stipule le théorème 2.4.3.2.

**Théorème 2.4.3.2: vitesse de convergence de l'algorithme FISTA**

Soit  $\{x_k\}_{k \geq 1}$  la suite générée par l'algorithme **FISTA**. Alors  $\forall k \geq 1$  :

$$f(x_k) - f(x^*) \leq \frac{2L_h \|x_0 - x^*\|^2}{(k+1)^2}, \quad (2.99)$$

où  $x^*$  appartient à l'ensemble des minimiseurs du problème (2.87) et  $x_0$  est le point d'initialisation de l'Algorithme 6.

On peut à nouveau utiliser ici une règle de calcul supplémentaire permettant de trouver une estimation du paramètre  $L_h$  qui respecte la condition (2.97) lorsque celui-ci n'est pas accessible directement. Notons aussi qu'il existe dans la littérature des méthodes basées sur la recherche linéaire qui améliorent légèrement la vitesse de convergence de l'algorithme **FISTA** [CALATRONI et CHAMBOLLE, 2017; CHAMBOLLE et DOSSAL, 2015; FLOREA et VOROBYOV, 2017].

Dans [BECK et TBOULLE, 2009], les algorithmes **ISTA** et **FISTA** sont testés sur un problème de déconvolution. Les résultats de vitesse de décroissance de la fonction résiduelle  $f(x_k) - f(x^*)$  obtenus avec l'algorithme **FISTA** montrent une nette amélioration par rapport à l'algorithme **ISTA**. Nous avons obtenu des résultats similaires sur un problème de **TDM** d'absorption classique comme l'illustre la figure 2.8. D'autres résultats seront présentés au chapitre 3.

**Remarque sur le pas  $\frac{1}{L}$  :**

Le pas de descente pour les algorithmes 5 et 6 peut être calculé de plusieurs manières, mais son calcul dépend de la forme de la fonction étudiée car il dépend de la constante de Lipschitz de la fonction  $h$ . Par exemple, pour un problème linéaire de la forme

$$\operatorname{argmin}_{x \in \mathbb{R}^n} \{\|Ax - b\|^2 + \|x\|_1\}, \quad (2.100)$$

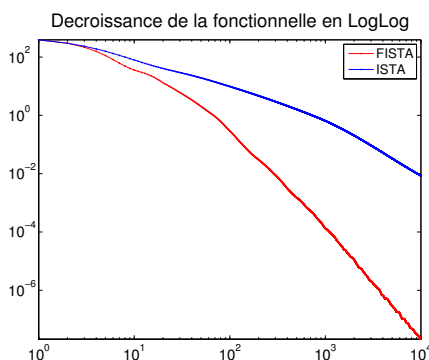


FIGURE 2.8 – Comparaison de la décroissance résiduelle  $f(x_k) - f(x^*)$  en échelle logarithmique en fonction des itérations pour les algorithmes **FISTA** (en rouge) et **ISTA** (en bleu) pour un problème de reconstruction en **TDM** d'absorption classique.

la constante de Lipschitz de  $\|Ax - b\|^2$  peut être obtenue en calculant

$$L = 2\lambda_{\max}(A^t A), \quad (2.101)$$

où  $\lambda_{\max}(A^t A)$  est la plus grande valeur propre de la matrice symétrique  $A^t A$ . Notons aussi que des méthodes génériques, comme celle proposée dans [WOOD et ZHANG, 1996], peuvent être utilisées pour estimer la constante de Lipschitz. Nous verrons comment calculer une bonne estimation de cette constante pour le cas de la **TDM** spectrale au chapitre 3.

## Conclusion

Nous avons vu que contrairement à la descente de gradient projeté classique, l'algorithme **ISTA** permet de traiter le cas d'une fonction non différentiable tout en gardant le caractère computationnel simple de la descente de gradient. Néanmoins, la vitesse de convergence de celle-ci, qui est de l'ordre de  $O(\frac{1}{k})$ , demeure trop lente pour pouvoir traiter le problème de **TDM** spectrale.

Une manière d'accélérer l'algorithme **ISTA** est d'avoir recours à un effet mémoire en appliquant une légère modification sur la direction de descente. Cette modification donne lieu à l'algorithme **FISTA**, qui garantit une vitesse de convergence de l'ordre de  $O(\frac{1}{k^2})$ . De plus, cette méthode conserve le caractère computationnel simple de l'algorithme **ISTA**. Mais malgré une vitesse de convergence rapide, il n'existe à ce jour que très peu d'études sur la convergence de l'algorithme **FISTA** pour des fonctions non nécessairement convexes, comme c'est le cas en **TDM** spectrale. On trouvera dans [PARIKH et BOYD, 2014] des informations très détaillées sur les algorithmes proximaux classiques ainsi que des exemples d'applications.

Par ailleurs, il existe d'autres méthodes pour accélérer la convergence de l'algorithme **ISTA**. Ces méthodes sont similaires aux descentes de second ordre que l'on a vu précédemment en 2.4.1.2 et 2.4.2. Nous allons dans la suite présenter ces méthodes.

### 2.4.3.4 Algorithme proximal à métrique variable

Comme nous l'avons précédemment dit, il existe deux grandes familles de stratégies d'accélération de l'Algorithme 5 dans la littérature :

- la première repose sur une accélération utilisant les informations obtenues lors des itérations précédentes pour construire le nouvel itéré comme nous l'avons vu en 2.4.3.3,
- la seconde stratégie d'accélération est basée sur l'introduction d'une **métrique** qui peut varier au cours des itérations [BECKER et FADILI, 2012; BONNANS et collab., 1995; CHEN

et ROCKAFELLAR, 1997; COMBETTES et VÜ, 2014; PARENTE et collab., 2008] et que l'on obtient à partir d'une matrice de préconditionnement. C'est ce qu'on appelle un **algorithme explicite-implicite à métrique variable** ou VMFB (*Variable Metric Forward-Backward*).

La convergence de l'algorithme VMFB dans le cas où les fonctions à minimiser sont convexes a été démontrée dans [COMBETTES et VÜ, 2014; TRAN-DINH et collab., 2015]. Une extension au cas de fonctions non convexes, mais qui satisfont l'inégalité de Kurdyka-Lojasiewicz (voir 2.2.3) a été démontrée dans [CHOUZENOUX et collab., 2014; REPETTI, 2015].

On s'intéresse ici au problème de minimisation suivant :

$$\operatorname{argmin}_{x \in \mathbb{R}^n} \{f(x) = h(x) + g(x)\}. \quad (2.102)$$

Cette fois ci on émettra des hypothèses un peu plus générales que celles vues précédemment. On supposera ici que

- h.1**  $f$  est une fonction coercive satisfaisant l'inégalité **K-L**,
- h.2**  $h$  est une fonction différentiable sur  $\mathbb{R}^n$  et que son gradient est  $L_h$ -Lipschitz, avec  $L_h > 0$ ,
- h.3**  $g$  est une fonction convexe, propre et semi-continue inférieurement (voir définition en 2.1.3).

Ces hypothèses assurent que l'ensemble des minimiseurs de  $f$  est non-vide. L'algorithme VMFB a été introduit dans [BECKER et FADILI, 2012; BONNANS et collab., 1995; CHEN et ROCKAFELLAR, 1997; COMBETTES et VÜ, 2014; PARENTE et collab., 2008] dans le but d'accélérer la convergence du schéma explicite-implicite vers un point critique. Le schéma de cette méthode est en Algorithme 7. Ce qui le différencie des méthodes classiques est qu'à l'étape **(e1)**, on préconditionne la descente à l'aide d'une métrique notée  $\mathbf{M}_k$ . Cette métrique joue un rôle essentiel dans **l'accélération de la convergence**. On peut faire le lien avec la descente de second ordre vue en 2.4.1.2.

Notons aussi que l'opérateur proximal relatif à une métrique quelconque n'a pas d'expression

---

#### Algorithm 7 VMFB

---

**Require:**  $x_0 \in \mathbb{R}^n$ ,  $\gamma_k > 0$ ,  $\lambda_k \in [0, 1]$ , et  $\mathbf{M}_k \in \mathbb{S}_N^+$

**for**  $k = 0, 1, 2, 3..$  **do**

$z = x_k - \gamma_k \mathbf{M}_k^{-1} \nabla h(x_k)$  ▷ (e1)

$y_k = \operatorname{prox}_{\gamma_k^{-1} \mathbf{M}_k, g}(z)$  ▷ (e2)

$x_{k+1} = \lambda_k y_k + (1 - \lambda_k) x_k$  ▷ (e3)

**end for**

---

simple explicite. Afin de pallier cette difficulté, il existe une version **inexacte** de Algorithme 7 s'inspirant de l'algorithme proposé dans [ARTOUCH et collab., 2011]. L'idée de cet algorithme est de calculer le point proximal à l'étape **(e2)** de manière itérative en utilisant par exemple un algorithme de descente de gradient projeté.

Enfin, l'étape **(e3)** est une étape de relaxation qui consiste à choisir un meilleur point entre  $y_k$  et l'itéré précédent  $x_k$ . En pratique, nous fixons souvent  $\lambda_k = 1$  mais nous verrons au paragraphe 2.4.3.5 une manière d'obtenir un meilleur pas. On supposera néanmoins dans la suite que les paramètres  $\gamma_k$  et  $\lambda_k$  satisfont les hypothèses suivantes :

- h.4**  $\exists \lambda_{\min} \in \mathbb{R}_*^+$  tel que  $\forall k \in \mathbb{N}$ ,  $\lambda_{\min} \leq \lambda_k \leq 1$ ,
- h.5**  $\exists \eta_1, \eta_2 \in \mathbb{R}_*^+$ , tels que  $\forall k \in \mathbb{N}$ ,  $\eta_1 \leq \lambda_k \gamma_k \leq 2 - \eta_2$ .

#### Choix d'une métrique

Dans l'article [CHOUZENOUX et collab., 2014], l'étape **(e1)** de Algorithme 7 est obtenue par un calcul similaire de celui effectué précédemment en 2.4.3.2. L'idée est de considérer cette fois ci une **approximation à l'ordre 2** de la fonction  $h$  pour la mise à jour de l'itéré

$$y_k = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ h(x_k) + \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} (x - x_k)^t \mathbf{M}_k (x - x_k) + g(x) \right\}. \quad (2.103)$$

On obtient alors par des calculs assez similaires de ceux en annexe A.1.5 que

$$y_k = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ g(x) + \frac{1}{2\gamma_k} \|x - (x_k - \gamma_k \mathbf{M}_k^{-1} \nabla h(x_k))\|_{\mathbf{M}_k}^2 \right\}, \quad (2.104)$$

d'où

$$y_k = \operatorname{prox}_{\gamma_k^{-1} \mathbf{M}_k, g} (x_k - \gamma_k \mathbf{M}_k^{-1} \nabla h(x_k)). \quad (2.105)$$

Ainsi, les hypothèses suivantes doivent être satisfaites pour pouvoir garantir une convergence de l'Algorithme 7 :

**h.6**  $\forall k \in \mathbb{N}$  la fonction quadratique définie  $\forall x \in \mathbb{R}^n$  par

$$Q_{\mathbf{M}_k}(x, x_k) = h(x_k) + (x - x_k)^t \nabla h(x_k) + \frac{1}{2} (x - x_k)^t \mathbf{M}_k (x - x_k), \quad (2.106)$$

est une majorante de  $h$  au point  $x_k$  (i.e.  $Q_{\mathbf{M}_k}(x, x_k) \geq h(x) \forall x$ ) et  $\mathbf{M}_k \in \mathbb{S}_N^+$ ,

**h.7** il existe  $\alpha, \beta \in \mathbb{R}_*^+$  tels que  $\forall k \in \mathbb{N}$

$$\alpha \mathbf{I}_n \leq \mathbf{M}_k \leq \beta \mathbf{I}_n \quad (2.107)$$

où  $\leq$  est l'ordre partiel de Loewner défini en 2.1.3 et  $\mathbf{I}_n$  la matrice identité.

Remarquons que de telles matrices existent car la fonction  $g$  est convexe et la fonction  $h$  a un gradient  $L_h$ -Lipschitz. Par exemple, on peut considérer le cas simple et utiliser la métrique constante qu'on définit  $\forall k \in \mathbb{N}$

$$\mathbf{M}_k = L_h \mathbf{I}_n,$$

on retomberait ainsi sur une descente explicite-implicite vue en l'Algorithme 5.

Il faut noter que l'obtention de bonnes performances de l'Algorithme 7 dépend fortement de l'exactitude de l'approximation de second ordre définie en (2.106). Cette approximation est caractérisée par la métrique  $\mathbf{M}_k$ . Plus précisément, il faut construire  $\mathbf{M}_k$  de telle façon que, pour tout  $k \in \mathbb{N}$ , la fonction quadratique  $Q_{\mathbf{M}_k}(\cdot, x_k)$  soit aussi proche que possible de la fonction  $h$  au voisinage de  $x_k$ , tout en satisfaisant les hypothèses **h.6** et **h.7**. Il existe toutefois dans la littérature des techniques de construction de telles métriques pour certaines classes particulières de fonctions  $h$  [ERDOGAN et FESSLER, 2002; GEMAN et YANG, 1995; HUNTER et collab., 2004].

### Convergence

Comme nous l'avons dit précédemment, la convergence de l'Algorithme 7 a déjà été démontrée pour le cas où les fonctions  $h$  et  $g$  sont convexes. Mais le résultat qui nous intéresse est celui présenté dans le chapitre 3 de [REPETTI, 2015], qui traite le cas plus général et qu'on présente ci-dessous.

#### Théorème 2.4.3.3: convergence de l'algorithme VMFB

Supposons que les hypothèses **h.1**, **h.2**, **h.3**, **h.4**, **h.5**, **h.6** et **h.7** soient satisfaites, alors les assertions suivantes sont vraies :

- (1) les suites  $\{x_k\}_{k \in \mathbb{N}}$  et  $\{y_k\}_{k \in \mathbb{N}}$  générées par l'Algorithme 7 convergent toutes les deux vers un point critique  $\hat{x}$  de  $f$ .
- (2) Ces deux suites satisfont :

$$\sum_{k=0}^{+\infty} \|x_{k+1} - x_k\| < +\infty \quad \text{et} \quad \sum_{k=0}^{+\infty} \|y_{k+1} - y_k\| < +\infty \quad (2.108)$$

- (3) les suites  $\{f(x_k)\}_{k \in \mathbb{N}}$  et  $\{f(y_k)\}_{k \in \mathbb{N}}$  convergent vers  $f(\hat{x})$ . De plus  $\{f(x_k)\}_{k \in \mathbb{N}}$  est décroissante.

La démonstration de ce théorème se trouve dans [CHOUZENOUX et collab., 2014; REPETTI, 2015]. Des résultats complémentaires dans [REPETTI, 2015] permettent d'affirmer que, si l'algorithme 7 est initialisé avec un point  $x_0$  suffisamment proche d'un minimum global de  $f$ , alors les suites  $\{f(x_k)\}_{k \in \mathbb{N}}$  et  $\{f(y_k)\}_{k \in \mathbb{N}}$  convergeront vers cette solution.

### Conclusion

Nous proposons dans ce paragraphe de minimiser le cas général d'une fonction  $f = h + g$  où  $h$  est une fonction différentiable non nécessairement convexe,  $g$  est une fonction convexe non nécessairement différentiable, et  $f$  est une fonction coercive satisfaisant l'inégalité de **K-L**. Pour cela, nous avons introduit une **version accélérée** de l'algorithme proximal classique, nommée **VMFB**. Cette accélération est basée sur l'introduction d'une métrique qui préconditionne la descente de gradient.

Un résultat de convergence théorique existe pour cette méthode et affirme que la suite générée par l'algorithme **VMFB** converge vers un point critique de  $f$ . Ce résultat est intéressant car il correspond au cas de la **TDM** spectrale. Nous présenterons au chapitre 3 les résultats obtenus avec cette méthode. Nous allons présenter dans la suite une méthode un peu plus élaborée que la précédente permettant de traiter le cas où le calcul du point proximal est inexact.

#### 2.4.3.5 Algorithme VMILAn

Comme nous l'avons remarqué, l'opérateur proximal relatif à une métrique quelconque n'a pas forcément une expression simple explicite et peut s'avérer difficile à calculer. L'idée serait alors d'estimer le point proximal de manière itérative. Nous considérons toujours ici le cas de minimisation suivant

$$\operatorname{argmin}_{x \in \mathbb{R}^n} \{f(x) = h(x) + g(x)\}, \quad (2.109)$$

avec les hypothèses que

- h.1**  $f$  est une fonction coercive satisfaisant l'inégalité **K-L**,
- h.2**  $h$  est une fonction différentiable sur  $\mathbb{R}^n$  et son gradient est  $L_h$ -Lipschitz, avec  $L_h > 0$ ,
- h.3**  $g$  est une fonction convexe, propre et semi-continue inférieurement (voir définition en 2.1.3).

L'algorithme que l'on va présenter est une version « améliorée » de l'algorithme **VMFB** 7. Il a été introduit pour la première fois dans [BONETTINI et collab., 2016] sous le nom **VMILA** (*Variable Metric Inexact Line-search Algorithm*). L'idée fondamentale de cet algorithme reprend et généralise des méthodes établies en [BIRGIN et collab., 2003; VILLA et collab., 2013]. Nous présenterons ici la nouvelle version de l'algorithme **VMILA**, appelée **VMILAn** (voir Algorithme 8) et qui a été introduite dans [BONETTINI et collab., 2017]. L'idée de cette méthode réside en deux points :

- (p1) le calcul **approché** (ou **inexact**) du point proximal,
- (p2) le calcul d'un paramètre de relaxation par une méthode de recherche linéaire (ou **line-search**) permettant d'optimiser la convergence.

#### Le point proximal inexact

Le point (p1) consiste à calculer une **approximation** assez bonne du point proximal pour garantir la convergence de l'algorithme. On rappelle que l'étape proximale est la suivante :

$$y_k = \operatorname{prox}_{\gamma_k^{-1} \mathbf{M}_k, g} (x_k - \gamma_k \mathbf{M}_k^{-1} \nabla h(x_k)), \quad (2.110)$$

avec  $\mathbf{M}_k$  une matrice symétrique définie positive et  $\gamma_k > 0$ . On a vu précédemment qu'on pouvait réécrire l'équation (2.110) de la manière suivante :

$$y_k = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ h(x_k) + \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} (x - x_k)^t \mathbf{M}_k (x - x_k) + g(x) \right\}. \quad (2.111)$$

Or on sait que les termes constants n'interviennent pas lors de la minimisation, car ceux-ci sont mis à zéro lors de l'étape de dérivation. Ainsi, on injecte le terme  $g(x_k)$  et on enlève le terme  $h(x_k)$  dans (2.111). On obtient alors

$$y_k = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} (x - x_k)^t \mathbf{M}_k (x - x_k) + g(x) - g(x_k) \right\}. \quad (2.112)$$

On pose alors

$$P_k(x) = \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} (x - x_k)^t \mathbf{M}_k (x - x_k) + g(x) - g(x_k). \quad (2.113)$$

On définit ensuite deux conditions sous lesquelles on aurait une bonne approximation  $\tilde{y}_k$  du point proximal  $y_k$  (proposition 3.3 de [BONETTINI et collab., 2016]). Ces conditions sont

$$P_k(\tilde{y}_k) < 0 \quad (2.114)$$

$$P_k(\tilde{y}_k) - P_k(y_k) \leq \epsilon_k, \quad (2.115)$$

où  $\epsilon_k \geq 0$  pour tout  $k$ , représente le critère de tolérance fixé pour la **qualité** de l'approximation. Il faut savoir qu'en pratique le point  $P_k(y_k)$  n'est pas calculé directement. On propose alors dans [BONETTINI et collab., 2017; RUDIN et collab., 1992; VILLA et collab., 2013] de résoudre un sous problème en maximisant une certaine fonction  $\Psi(v)$  qui substitue le calcul de  $P_k(y_k)$  de sorte à respecter la condition (2.115). Ceci est fait en général en appliquant une méthode itérative [BONETTINI et collab., 2017].

### Line-search

L'une des principales caractéristiques de l'algorithme **VMILAn** porte sur le second point (**p2**). Cette étape consiste à trouver le paramètre  $\lambda_k$  optimal avec

$$x_{k+1} = \lambda_k y_k + (1 - \lambda_k) x_k. \quad (2.116)$$

Le paramètre  $\lambda_k$  doit être choisi de sorte à garantir une valeur  $f(x_{k+1})$  la plus petite possible, ce qui accélère la convergence de l'algorithme vers un point critique. Pour cela, on considère la fonction suivante :

$$L(\lambda) = f(\lambda y_k + (1 - \lambda) x_k). \quad (2.117)$$

L'idée serait de trouver le paramètre  $\hat{\lambda}$  tel que

$$\hat{\lambda} = \operatorname{argmin}_{\lambda \in \mathbb{R}^+} L(\lambda). \quad (2.118)$$

Il existe plusieurs règles de calcul d'un tel paramètre [BERTSEKAS, 1999; NOCEDAL et WRIGHT, 2006]. On présente dans [BONETTINI et collab., 2017] une méthode suivant la règle **d'Armijo**, qui donne un bon compromis entre facilité d'implémentation et décroissance de la fonctionnelle. Il s'agit de trouver le paramètre  $\lambda_k$  satisfaisant l'hypothèse suivante :

$$f(\lambda_k y_k + (1 - \lambda_k) x_k) \leq f(x_k) + \lambda_k P_k(\tilde{y}_k). \quad (2.119)$$

En pratique on ne peut tester toutes les valeurs  $\lambda_k$  possibles qui vérifient cette inégalité. On effectue donc une recherche selon une grille de raison géométrique sur quelques valeurs  $\lambda_k$ .

Nous avons implémenté deux versions de l'algorithme 8. Une version avec la règle de calcul de **line-search** présentée ci-dessus et une avec une version plus simple où le paramètre  $\lambda_k$  est fixé à 1. Nous verrons au chapitre 3 pourquoi il n'y a pas de grande différence entre les résultats générés par ces deux versions de l'algorithme.

---

**Algorithm 8** VMILAn

---

**Require:**  $x_0 \in \mathbb{R}^n$ ,  $\gamma_k > 0$  et  $\mathbf{M}_k \in \mathbb{S}_N^+$

**for**  $k = 0, 1, 2, 3..$  **do**

$$z = x_k - \gamma_k \mathbf{M}_k^{-1} \nabla h(x_k) \quad \triangleright \text{(e1)}$$

Calculer  $\tilde{y}_k$  vérifiant (2.114) et (2.115) ▷ (e2)

Calculer  $\lambda_k$  vérifiant l'hypothèse (2.119) ▷ (e3)

$$x_{k+1} = \lambda_k \tilde{y}_k + (1 - \lambda_k) x_k \quad \triangleright \text{(e4)}$$

**end for**

---

**Convergence**

La particularité de l'algorithme **VMILAn** réside dans les conditions de convergence de l'algorithme. Celles-ci sont plus générales que les conditions présentées précédemment pour l'algorithme **VMFB** en paragraphe 2.4.3.4. Ici aucune condition particulière n'est requise pour le pas de descente  $\gamma_k$ .

On trouvera par ailleurs dans [BONETTINI et collab., 2017] les principaux résultats de convergence de l'algorithme, ainsi que des résultats sur la vitesse de convergence pour une certaine classe de fonctions satisfaisant l'inégalité **K-L**.

**Conclusion**

Nous présentons ici un algorithme proximal plus élaboré que les précédentes méthodes décrites. La particularité de cet algorithme réside en deux points :

1. le **calcul inexact** du point proximal, permettant ainsi de pallier le cas où le point proximal ne peut être obtenu de manière analytique,
2. le calcul du **paramètre de line-search**, permettant d'accélérer la convergence de l'algorithme.

L'avantage de cette méthode est qu'elle peut résoudre des problèmes de minimisation sous des hypothèses générales de non différentiabilité et non convexité. Nous présenterons au chapitre 3 les résultats obtenus avec cette méthode.

**Conclusion**

Ce chapitre avait pour objectif d'introduire les notions mathématiques fondamentales permettant de comprendre notre approche pour résoudre le problème de **TDM** spectrale.

La section 2.1 introduit généralement les problèmes inverses. Elle donne une définition permettant de comprendre pourquoi les outils de cette discipline sont essentiels à notre sujet. On y introduit par la même occasion les définitions et notations mathématiques utilisées tout au long du chapitre.

On trouve en section 2.2 les trois outils importants de ce travail de thèse. En effet, ces outils permettent de traiter le cas d'optimisation des fonctions non différentiables et/ou non convexes. Ils sont importants et fondamentaux car ils serviront de base aux algorithmes que nous avons utilisés pour résoudre le problème de **TDM** spectrale.

Enfin, la section 2.4 présente les algorithmes d'optimisation que l'on peut être amener à utiliser pour résoudre un problème inverse. Cette section se divise en trois parties. La première partie parle des algorithmes fondamentaux à descente de gradient sur lesquels on se base pour construire des algorithmes plus élaborés.

La seconde partie parle des algorithmes à région de confiance, qui sont une généralisation des algorithmes à descente de gradient et qui permettent une convergence plus rapide. Enfin, la troisième partie parle des algorithmes proximaux qui permettent de traiter des cas de fonctions plus générales et qui présentent des vitesses de convergence rapides.

Ce chapitre donne donc les outils nécessaire pour palier les difficultés liés au problème de la **TDM** spectrale. Le chapitre 3 qui suit présente les tests menés et les résultats obtenus durant ce travail de thèse.



## 2.5 Références

- P.-A. ABSIL, R. MAHONY and B. ANDREWS. 2005, «Convergence of the iterates of descent methods for analytic cost functions», *SIAM J. Optimiz.*, vol. 6, n° 2, p. 531–547. [85](#)
- H. ATTOUCH, J. BOLTE, P. REDONT and A. SOUBEYRAN. 2010, «Proximal alternating minimization and projection methods for nonconvex problems : An approach based on the Kurdyka-Łojasiewicz inequality», *Math. Oper. Res.*, vol. 35, n° 2, p. 438–457. [86](#)
- H. ATTOUCH, J. BOLTE and B. F. SVAITER. 2011, «Convergence of descent methods for semi-algebraic and tame problems : proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods», *Math. Program., Ser. A*, vol. 137, n° 1, doi :10.1007/s10107-011-0484-9, p. 91–124. URL <https://hal.archives-ouvertes.fr/hal-00790042>. [81](#), [86](#), [105](#)
- AVRIEL, M. 2003, «Nonlinear Programming : Analysis and Methods», Dover Books on Computer Science Series, Dover Publications, ISBN 9780486432274. URL <https://books.google.fr/books?id=byF4Xb1QbvMC>. [93](#)
- H. H BAUSCHKE and P. L. COMBETTES. 2011, «Convex Analysis and Monotone Operator Theory in Hilbert Spaces», 1<sup>re</sup> éd., Springer Publishing Company, Incorporated. [83](#)
- A. BECK and M. TEOULLE. 2009, «A fast iterative shrinkage-thresholding algorithm for linear inverse problems», *SIAM J. Imaging Sci.* 2(1) : 183–202. [102](#), [103](#)
- S. BECKER and J. FADILI. 2012, «A quasi-newton proximal splitting method», Proceeding NIPS'12 Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2 p. 2618–2626. [104](#), [105](#)
- J. BECT, L. BLANC-FÉRAUD, G. AUBERT and A. CHAMBOLLE. 2004, «A l1-unified variational framework for image restoration», in Pajdla T., Matas J. (eds) Computer Vision - ECCV 2004. ECCV 2004. Lecture Notes in Computer Science, vol 3024. Springer, Berlin, Heidelberg, p. 1–13. [90](#)
- D. BERTSEKAS. 1999, «Nonlinear Programming», Athena Scientific. [78](#), [101](#), [108](#)
- J. BIOUSCAS-DIAS and M. A. T. FIGUEIREDO. 2007, «A new twist : Two-step iterative shrinkage/thresholding algorithms for image restoration», *IEEE T. Imag. Process.* 16(12), p. –, ISSN 1057-7149, doi :10.1109/TIP.2007.909319. [103](#)
- Y. BIRAUD. 1976, «Les méthodes de déconvolution et leurs limitations fondamentales», *Revue de Physique Appliquée*, vol. 11, n° 2, doi :10.1051/rphysap:01976001102020300, p. 203–214. URL <https://hal.archives-ouvertes.fr/jpa-00244050>. [70](#)
- E. G. BIRGIN, J. M. MARTÍNEZ and M. RAYDAN. 2003, «Inexact spectral projected gradient methods on convex sets», *IMA J. Numer. Anal.*, vol. 23, n° 4, p. 539–559. [107](#)
- T. BLUMENSATH and M. E. DAVIES. 2008, «Iterative thresholding for sparse approximations», *J. Fourier Anal. Appl.*, vol. 14, n° 5, doi :10.1007/s00041-008-9035-z, p. 629–654, ISSN 1531-5851. URL <https://doi.org/10.1007/s00041-008-9035-z>. [83](#)
- J. BOLTE, A. DANIILIDIS and A. LEWIS. 2006, «The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems», *SIAM J. on Optimiz.*, vol. 17, n° 4, doi :10.1137/050644641, p. 1205–1223, ISSN 1052-6234. URL <http://dx.doi.org/10.1137/050644641>. [85](#)
- J. BOLTE, A. DANIILIDIS, O. LEY and L. MAZET. 2010, «Characterizations of Łojasiewicz inequalities : subgradient flows, talweg, convexity», *SIAM J. Optimiz.*, vol. 362, n° 6, p. 3319–3363. [86](#)



- S. BONETTINI, I. LORIS, F. PORTA and M. PRATO. 2016, «Variable metric inexact line-search-based methods for nonsmooth optimization», *SIAM J. Optimiz.*, vol. 26, n° 2, p. 891–921. [107](#), [108](#)
- S. BONETTINI, I. LORIS, F. PORTA, M. PRATO and S. REBEGOLDI. 2017, «On the convergence of a line-search based proximal-gradient method for nonconvex optimization», *Inverse Probl.*, vol. 33, n° 5, p. 055 005. [107](#), [108](#), [109](#)
- J. BONNANS, J. GILBERT, C. LEMARÉCHAL and C. SAGASTIZÁBAL. 2003, «Numerical Optimization : theoretical and practical aspects», Universitext (Berlin. Print), Springer, ISBN 9783540001911. URL <https://books.google.fr/books?id=cmM01T3aP1UC>. [72](#)
- J. F. BONNANS, J. C. GILBERT, C. LEMARÉCHAL and C. A. SAGASTIZÁBAL. 1995, «A family of variable metric proximal methods», *Math. Program.*, vol. 68, n° 1, doi :10.1007/BF01585756, p. 15–47, ISSN 1436-4646. URL <https://doi.org/10.1007/BF01585756>. [104](#), [105](#)
- M. BONNET 2008, «Cours de problèmes inverses pour le master recherche», Lecture. [70](#)
- J. M. BORWEIN and A. S. LEWIS. 2000, «Convex analysis and nonlinear optimization, theory and examples», Springer, New York. [77](#)
- C. A. BOUMAN and K. D. SAUER. 1996, «A unified approach to statistical tomography using coordinate descent optimization», *IEEE T. Imag. Process.*, vol. 5 3, p. 480–92. [90](#)
- K. BREDIES and D. A. LORENZ. 2008, «Linear convergence of iterative soft-thresholding», *J Fourier Anal Appl*, doi :10.1007/s00041-008-9041-1. [102](#)
- P. H. CALAMAI and J. J. MORÉ. 1987, «Projected gradient methods for linearly constrained problems», *Math. Program.*, vol. 39, n° 1, doi :10.1007/BF02592073, p. 93–116, ISSN 1436-4646. URL <https://doi.org/10.1007/BF02592073>. [95](#)
- L. CALATRONI and A. CHAMBOLLE. 2017, «Backtracking strategies for accelerated descent methods with smooth composite objectives», *arXiv preprint arXiv :1709.09004*. [103](#)
- C. CARTIS, N. I. M. GOULD and P. L. TOINT. 2012, «An adaptive cubic regularization algorithm for nonconvex optimization with convex constraints and its function-evaluation complexity», *IMA J. Numer. Anal.*, vol. 32, n° 4, doi :10.1093/imanum/drr035, p. 1662–1695, ISSN 0272-4979. [99](#)
- A. CHAMBOLLE. 2004, «An algorithm for total variation minimization and applications», *J. Math. Imaging Vis.*, vol. 20, n° 1-2, doi :10.1023/B:JMIV.0000011325.36760.1e, p. 89–97, ISSN 0924-9907. URL <http://dx.doi.org/10.1023/B:JMIV.0000011325.36760.1e>. [91](#)
- A. CHAMBOLLE and C. DOSSAL. 2015, «On the convergence of the iterates of "FISTA"», *J. Optimiz. Theory App.*, vol. Volume 166, n° Issue 3, p. 25. URL <https://hal.inria.fr/hal-01060130>. [103](#)
- C. CHAUX, P. L. COMBETTES, J.-C. PESQUET and V. R. WAJS. 2007, «A variational formulation for frame-based inverse problems», *Inverse Probl.*, vol. 23, n° 4, p. 1495. URL <http://stacks.iop.org/0266-5611/23/i=4/a=008>. [80](#), [81](#)
- C. CHAUX, L. DUVAL, A. BENAZZA-BENYAHIA and J. PESQUET. 2008, «A nonlinear Stein-based estimator for multichannel image denoising», *IEEE T. Signal Proces.*, vol. 56, n° 8-2, p. 3855–3870. [89](#)
- G. H. -G. CHEN and R. T. ROCKAFELLAR. 1997, «Convergence rates in forward–backward splitting», *SIAM J. Optimiz.*, vol. 7, n° 2, doi :10.1137/S1052623495290179, p. 421–444, ISSN 1052-6234. URL <http://dx.doi.org/10.1137/S1052623495290179>. [104](#), [105](#)
- E. CHOZENOUX, A. JEZIERSKA, J.-C. PESQUET and H. TALBOT. 2011, «A Majorize-Minimize subspace approach for l2-l0 image regularization», *SIAM J. Imaging Sci.*, vol. 6,n° 1, p. 563-591. [89](#)

- E. CHOUZENOUX, J.-C. PESQUET and A. REPETTI. 2014, «Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function», *J. Optimiz. Theory App.*, vol. 162, n° 1, doi :10.1007/s10957-013-0465-7, p. 107–132. [81](#), [105](#), [107](#)
- G. COHEN. 2000, «Convexité et Optimisation», URL <https://cel.archives-ouvertes.fr/cel-00356686>, lecture. [70](#)
- P. COMBETTES and J. PESQUET. 2008, «Proximal thresholding algorithm for minimization over orthonormal bases», *SIAM J. Optimiz.*, vol. 18, n° 4, doi :10.1137/060669498, p. 1351–1376. URL <https://doi.org/10.1137/060669498>. [80](#), [81](#)
- P. L. COMBETTES and J.-C. PESQUET. 2010, «Proximal Splitting Methods in Signal Processing», *Springer Optimization and Its Applications*, vol. 49. [83](#), [100](#)
- P. L. COMBETTES and B. C. VŪ. 2014, «Variable metric forward–backward splitting with applications to monotone inclusions in duality», *Optimization*, vol. 63, n° 9, p. 1289–1318. [81](#), [105](#)
- P. L. COMBETTES and V. WAJS. 2005, «Signal recovery by proximal forward-backward splitting», *SIAM J. Multiscale Model. Sim.*, vol. 4 n°4, p. 1164–1200. URL <https://hal.archives-ouvertes.fr/hal-00017649>. [80](#), [81](#), [100](#), [102](#)
- I. DAUBECHIES, M. DEFRISE and C. DE MOL. 2004, «An iterative thresholding algorithm for linear inverse problems with a sparsity constraint», *Commun. Pur. App. Math.*, vol. 57, n° 11, p. 1413–1457. [100](#)
- C.-A. DELEDALLE, G. PEYRÉ and J. FADILI. 2013, «Stein consistent risk estimator (score) for hard thresholding», SPARS’13, Jul 2013, Lausanne, Switzerland. [89](#)
- G. DEMOMENT 1989, «Image reconstruction and restoration : overview of common estimation structures and problems», *IEEE T. Acoust. Speech Signal Process.*, vol. 37, p. 2024–2036. [86](#)
- S. DENKOWSKA, J. STASICA and M. P. DENKOWSKI. 2008, «Ensembles sous-analytiques à la polonaise : avec une introduction aux fonctions et ensembles analytiques», Editions Hermann, Paris, ISBN 9782705666880, 144 p.. URL <http://okina.univ-angers.fr/publications/ua199>. [85](#)
- D. L. DONOHO 2006, «Compressed sensing», *IEEE T. Inform. Theory*, vol. 52, p. 1289–1306. [90](#)
- D. L. DONOHO, I. M. JOHNSTONE, G. KERKYACHARIAN and D. PICARD. 1995, «Wavelet shrinkage : asymptopia», *J. R. Stat. Soc. B*, p. 371–394. [90](#)
- H. ERDOGAN and J. A. FESSLER. 2002, «Monotonic algorithms for transmission tomography», *IEEE Transactions on Medical Imaging*, vol. 18, p. 801–814. [106](#)
- M. A. T. FIGUEIREDO and R. NOWAK. 2003, «An em algorithm for wavelet-based image restoration», *IEEE T. Med. Imaging*, vol. 12, n° 8, p. 906–916,doi :10.1109/TIP.2003.814255. [100](#)
- M. A. T. FIGUEIREDO, R. NOWAK and S. WRIGHT. 2007, «Gradient projection for sparse reconstruction : application to compressed sensing and other inverse problems», *IEEE JSTSP*, vol. 1, n° 4, doi :10.1109/JSTSP.2007.910281, p. 586–597. [90](#)
- M. I. FLOREA and S. A. VOROBYOV. 2017, «A robust fista-like algorithm», *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, doi :10.1109/ICASSP.2017.7953012, p. 4521–4525, ISSN 2379-190X. [103](#)
- D ; GEMAN and C. YANG. 1995, «Nonlinear image recovery with half-quadratic regularization», *IEEE T. Image Process.*, vol. 4, n° 7, p. 932–946. [106](#)

- J. HADAMARD. 1902, «Sur les problèmes aux dérivés partielles et leur signification physique», *Princeton University Bulletin*, vol. 13, p. 49–52. [70](#)
- D. R. HUNTER, K. LANGE, D. O. BIOMATHEMATICS and H. GENETICS. 2004, «A tutorial on MM algorithms», *Am. Stat.*, p. 30–37. [106](#)
- K. C. KIWIEL. 2001, «Convergence and efficiency of subgradient methods for quasiconvex minimization», *Math. Program.*, vol. 90, n° 1, doi :10.1007/PL00011414, p. 1–25, ISSN 1436-4646. URL <https://doi.org/10.1007/PL00011414>. [93](#)
- K. KURDYKA. 1998, «On gradients of functions definable in o-minimal structures», *Annales de l'institut Fourier*, vol. 48, n° 3, doi :10.5802/aif.1638, p. 769–783. [85](#)
- J.-F. LE MARÉCHAL and M. COQUIDÉ. 2006, «Modélisation et simulation dans l'enseignement scientifique : usages et impacts», *Aster (Paris. En Ligne)*, vol. 43, p. 47–56. URL <https://hal.archives-ouvertes.fr/hal-00372868>. [71](#)
- D. MARQUARDT. 1963, «An algorithm for least-squares estimation of nonlinear parameters», *J. Soc. Ind. Appl. Math.*, vol. 11, n° 2, doi :10.1137/0111030, p. 431–441. URL <https://doi.org/10.1137/0111030>. [94](#)
- W. MENKE. 2012, «Inverse problems», dans *Geophysical Data Analysis : Discrete Inverse Theory (Third Edition)*, édité par W. Menke, third edition éd., Academic Press, Boston, ISBN 978-0-12-397160-9, p. 231 – 259, doi :https://doi.org/10.1016/B978-0-12-397160-9.00012-6. URL <http://www.sciencedirect.com/science/article/pii/B9780123971609000126>. [71](#)
- J. J. MORÉ. 1983, «Recent Developments in Algorithms and Software for Trust Region Methods», Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-642-68874-4, p. 258–287, doi :10.1007/978-3-642-68874-4\_11. URL [https://doi.org/10.1007/978-3-642-68874-4\\_11](https://doi.org/10.1007/978-3-642-68874-4_11). [91](#)
- J. J. MOREAU. 1965, «Proximité et dualité dans un espace hilbertien», *Bulletin de la S. M. F.*, vol. 93, p. 273–299. [80](#)
- J. MORÉ and D. SORENSEN. 1983, «Computing a trust region step», *SIAM J. Sci. Comput.*, vol. 4, n° 3, doi :10.1137/0904038, p. 553–572. URL <https://doi.org/10.1137/0904038>. [97](#)
- M. Z. NASHED. 1974, «Approximate regularized solutions to improperly posed linear integral and operator equations», dans *Constructive and Computational Methods for Differential and Integral Equations*, édité par D. L. Colton et R. P. Gilbert, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-540-37302-5, p. 289–332. [71](#)
- Y. E. NESTEROV. 2013, «Gradient methods for minimizing composite objective function», *Math. Program.*, doi :10.1007/s10107-012-0629-5. [103](#)
- Y. E. NESTEROV. 1983, «A method for solving the convex programming problem with convergence rate  $o(1/k^2)$ », *Dokl. Akad. Nauk SSSR*, vol. 269, p. 543–547. URL <https://ci.nii.ac.jp/naid/10029946121/en/>. [103](#)
- J. NOCEDAL and S. J. WRIGHT. 2006, «Numerical Optimization», 2<sup>e</sup> éd., Springer, New York, NY, USA. [76](#), [80](#), [91](#), [92](#), [93](#), [94](#), [97](#), [99](#), [108](#)
- P. OCHS, Y. CHEN, T. BROX and T. POCK. 2014, «iPiano : inertial proximal algorithm for nonconvex optimization», *SIAM J. Imaging Sci.*, vol. 7, n° 2, p. 1388–1419. [103](#)
- L. A. PARENTE, P. A. LOTITO and M. V. SOLODOV. 2008, «A class of inexact variable metric proximal point algorithms», *SIAM J. Optimiz.*, vol. 19, n° 1, p. 240–260. [105](#)

- N. PARIKH and S. BOYD. 2014, «Proximal algorithms», *Found. Trends Optim.*, vol. 1, n° 3, doi :10.1561/2400000003, p. 127–239, ISSN 2167-3888. URL <http://dx.doi.org/10.1561/2400000003>. 80, 81, 92, 104
- A. REPETTI. 2015, «Algorithmes d'optimisation en grande dimension : applications à la résolution de problèmes inverses», thèse de doctorat Univ. Paris-Est. URL <http://www.theses.fr/2015PESC1032/document>. 87, 105, 106, 107
- R. T. ROCKAFELLAR and R. J.-B. WETS. 1998, *Variational Analysis*, Springer Verlag, Heidelberg, Berlin, New York. 78, 81
- R. T. ROCKAFELLAR. 1970, *Convex analysis*, Princeton Mathematical Series, Princeton University Press, Princeton, N. J. 78
- L. I. RUDIN, S. OSHER and E. FATEMI. 1992, «Nonlinear total variation based noise removal algorithms», *J. Phys. D*, vol. 60, n° 1-4, p. 259–268, ISSN 0167-2789. URL [https://doi.org/10.1016/0167-2789\(92\)90242-F](https://doi.org/10.1016/0167-2789(92)90242-F). 91, 108
- A. TARANTOLA. 2005, «Inverse Problem Theory and Methods for Model Parameter Estimation», Other titles in applied mathematics, Society for Industrial and Applied Mathematics, ISBN 9780898715729. URL <https://books.google.fr/books?id=jKeHEWtzdKgC>. 71
- A. N. TIKHONOV and V. Y. ARSEININ. 1977, «Solutions of ill-posed problems», W.H. Winston. 71
- Q. TRAN-DINH, A. KYRILLIDIS and V. CEVHER. 2015, «Composite self-concordant minimization», *Mach. Learn.*, vol. 16, p. 371–416. 105
- S. VILLA, S. SALZO, L. BALDASSARRE and A. VERRI. 2013, «Accelerated and inexact forward-backward algorithms», *SIAM J. Optimiz.*, vol. 23, n° 3, p. 1607–1633. URL <https://doi.org/10.1137/110844805>. 107, 108
- P.-L. WANTZEL. 1845, «Démonstration de l'impossibilité de résoudre toutes les équations algébriques avec des radicaux», *Nouvelles annales de mathématiques : journal des candidats aux écoles polytechnique et normale*, vol. 1, n° 4, p. 57–65. 92
- G. R. WOOD and B. P. ZHANG. 1996, «Estimation of the lipschitz constant of a function», *J. Global Optim.*, vol. 8, n° 1, doi :10.1007/BF00229304, p. 91–103, ISSN 1573-2916. URL <https://doi.org/10.1007/BF00229304>. 104
- S. WU, C. HUANG and Y. LIU. 2008, «Newton waveform relaxation method for solving algebraic nonlinear equations», *Appl. Math. Comput.*, vol. 201, n° 1-2, p. 553–560. 94
- J.-L. ZHANG. 2004, «A robust trust region method for general constrained optimization», *Math. Method. Oper. Res.*, vol. 60, n° 1, doi :10.1007/s001860300333, p. 73–85, ISSN 1432-5217. URL <https://doi.org/10.1007/s001860300333>. 99
- S. ŁOJASIEWICZ. 1963, «Une propriété topologique des sous-ensembles analytiques réels», Institut des Hautes Études Scientifiques Bures-sur-Yvette (Seine-et-Oise) France. 85

# Chapitre 3

## Méthodes et Résultats

*« aussi laisserons-nous le lecteur répondre à cette question : Qui est le plus heureux ? L'homme qui aura bravé la tempête de la vie et vécu ou celui qui sera resté en sécurité sur la berge et se sera contenté d'exister ? »*

---

Hunter S. Thompson

### Sommaire

---

<b>3.1 Les méthodes développées</b> . . . . .	<b>116</b>
3.1.1 Rappel du problème de TDM spectrale . . . . .	118
3.1.2 Les fonctions objectifs . . . . .	119
3.1.3 Choix de la métrique . . . . .	121
3.1.4 Filtre conique . . . . .	123
3.1.5 Choix du pas de descente . . . . .	124
<b>3.2 Tests et résultats sur données simulées</b> . . . . .	<b>126</b>
3.2.1 Tests sur données de TDM d'absorption classique . . . . .	126
3.2.2 Tests sur données polychromatiques 2D . . . . .	135
3.2.3 Tests sur donnée polychromatique 3D . . . . .	143
3.2.4 Conclusion sur les données simulées . . . . .	165
<b>3.3 Tests et résultats sur données réelles</b> . . . . .	<b>165</b>
3.3.1 Acquisition de données le prototype PIXSCAN-FLI . . . . .	165
3.3.2 Résultats sur données réelles . . . . .	172
<b>3.4 Références</b> . . . . .	<b>192</b>

---

## Introduction

Dans ce dernier chapitre nous allons présenter les méthodes développées ainsi que les tests et les résultats obtenus durant ce travail de thèse. Dans la première section nous allons rappeler le problème de **TDM** spectrale que l'on a souhaité résoudre. On présentera ainsi les fonctions objectifs que nous avons choisies de minimiser et les contraintes de régularisation que nous avons utilisées. Nous expliquerons aussi les choix des différents paramètres relatifs aux algorithmes présentés au chapitre 2. Nous détaillerons particulièrement le choix et le calcul de la métrique qui intervient dans les algorithmes **VMFB** et **VMILAn**.

La seconde section traitera essentiellement des tests menés sur données simulées que l'on regroupe en deux groupes : les tests en **TDM** d'absorption classique (avec l'hypothèse de spectre **monochromatique**) et les tests en **TDM** spectrale (avec l'hypothèse de spectre **polychromatique**). Pour les tests en **TDM** spectrale, nous avons obtenu des résultats sur des données en 2D et en 3D.

Enfin, la dernière section traitera des tests menés sur données réelles acquises sur le tomodesitomètre PIXSCAN-FLI présenté au chapitre 1.

## 3.1 Les méthodes développées

Nous allons dans cette section présenter les outils développés durant ce travail de thèse. Dans un premier temps nous ferons un bref rappel sur le problème de **TDM** spectrale auquel nous sommes confrontés ainsi que les fonctions objectifs que l'on souhaite minimiser.

A partir de l'expression du gradient et de la hessienne de ces fonctions, nous allons voir comment construire une métrique adéquate permettant le bon fonctionnement des algorithmes proximaux présentés en 2.4.3. Nous discuterons par la même occasion du pas de descente que l'on choisit pour les algorithmes.

### Ce que l'on attend de la TDM spectrale

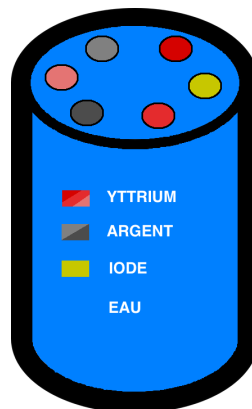


FIGURE 3.1 – Schéma d'un cylindre d'eau contenant différents agents de contrastes à différentes concentrations.

On rappelle ici les avantages attendus de la **TDM** spectrale. On prend pour exemple le cylindre d'eau en figure 3.1 contenant des agents de contrastes :

- trois inserts d'yttrium en solution à différentes concentrations,
- deux inserts d'argent en solution à différentes concentrations,
- un insert d'iode en solution.

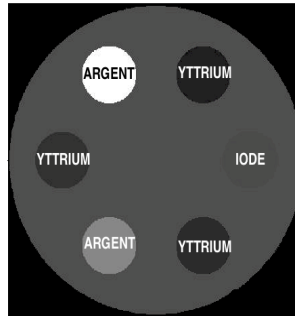


FIGURE 3.2 – Exemple de reconstruction possible en **TDM** d'absorption classique.

Si nous reconstruisons l'objet en utilisant des méthodes en **TDM** d'absorption classique, nous obtenons une image où le contraste entre les composants peut être très faible. En effet, cette méthode suppose que le spectre est **monochromatique**, ce qui est rarement le cas en pratique. Ceci résulte donc en une reconstruction équivalente à celle que l'on aurait obtenue si le spectre était monochromatique à une énergie de l'ordre de l'énergie moyenne du vrai spectre. Par exemple, en figure 3.2, l'insert contenant l'iode peut être confondu avec l'eau car il n'y a pas assez de contraste pour le distinguer, donc nous ne pouvons pas identifier clairement ce composant sur l'image. De plus, même si le contraste des solutions d'yttrium et d'argent est important, on ne peut pas discriminer ces deux agents de contraste.

En **TDM** spectrale cependant, il est possible de **localiser** et d'**identifier** chaque composant de

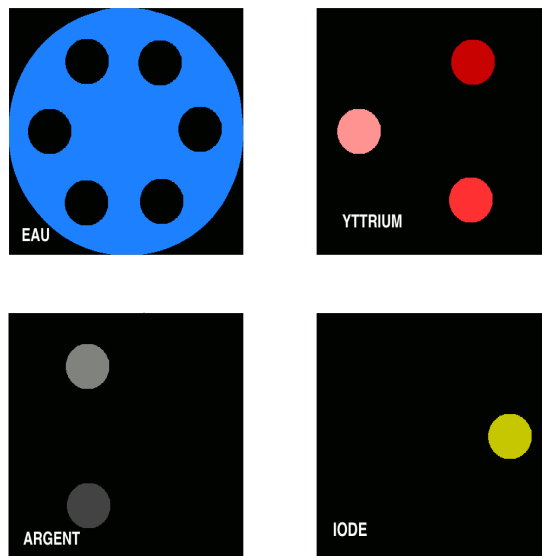


FIGURE 3.3 – Illustration de ce que l'on attend d'une reconstruction en **TDM** spectrale (échelles de couleur arbitraires).

l'objet et d'estimer aussi la concentration réelle de chaque composant directement sur l'image. Comme on peut le voir sur l'exemple en figure 3.3, cette fois on distingue l'iode de l'eau et on distingue aussi l'argent de l'yttrium. De plus on est capable de quantifier chaque concentration car les cartes de l'argent et de l'yttrium ont des intensités de couleurs différentes.

Ce cas idéal permet de décrire les attentes et bénéfices de la **TDM** spectrale. Nous allons présenter dans les sections 3.2 et 3.3 des résultats assez proches de ces attentes.



### 3.1.1 Rappel du problème de TDM spectrale

Nous avons vu au chapitre 1 que lorsque la source est **polychromatique**, on quantifie l'observation mesurée avec la loi de Beer-Lambert. Le modèle reliant les observations à l'objet imagé sous forme discrétisée s'écrit alors

$$Y = e^{-SA\Sigma F} \quad (3.1)$$

où

- $Y \in \mathbb{R}^{P \times M}$  est la **matrice de données** qui contient dans chaque colonne la  $m$ -ième acquisition,
- $F \in \mathbb{R}^{N \times M}$  est le **dictionnaire de filtrage en énergie** qui contient dans chaque colonne les facteurs des variations énergétiques induites par les configurations de la  $m$ -ième acquisition,
- $S \in \mathbb{R}^{P \times D}$  est l'**opérateur de transformée** à rayons X qui modélise la projection de l'objet sur le détecteur,
- $A \in \mathbb{R}^{D \times K}$  est la **matrice des cartes des concentrations** que l'on souhaite reconstruire, sa colonne  $k$  contient la répartition du  $k$ -ième composant dans le volume,
- $\Sigma \in \mathbb{R}^{K \times N}$  est le **dictionnaire des sections efficaces** d'interaction contenant dans la ligne  $k$  l'évolution de la probabilité d'interaction du  $k$ -ième composant.

Le problème de reconstruction peut être par la suite exprimé comme un problème inverse :

$$\hat{\mathbf{A}} = \underset{\mathbf{A} \in \mathbb{R}^{D \times K}}{\operatorname{argmin}} \{J(\mathbf{A}) = G(\mathbf{A}) + R(\mathbf{A})\}, \quad (3.2)$$

avec  $J(\mathbf{A})$  la fonction objectif,  $G(\mathbf{A})$  le terme d'attache aux données et  $R(\mathbf{A})$  le terme de régularisation. Nous avons vu au chapitre 2 que ces fonctions peuvent avoir plusieurs formes. Pour le terme d'attache aux données, nous retiendrons les deux formes suivantes :

- lorsque le bruit appliqué aux données observées est un bruit blanc gaussien, alors le terme d'attache aux données est de la forme :

$$G(\mathbf{A}) = \frac{1}{2\sigma^2} \|\operatorname{vec}(Y - e^{-SA\Sigma F})\|^2, \quad (3.3)$$

- lorsque le bruit appliqué aux données observées est un bruit de Poisson, alors le terme d'attache aux données est de la forme :

$$G(\mathbf{A}) = \sum_{p,m} \{e^{-SA\Sigma F}\}_{p,m} - Y_{p,m} \log(\{e^{-SA\Sigma F}\}_{p,m}). \quad (3.4)$$

Nous avons utilisé une contrainte de boîtes (voir 2.3.2.1) contraignant l'objet  $\mathbf{A}$  à appartenir à l'ensemble convexe  $C$  suivant :

$$C = \{\mathbf{A} \in \mathbb{R}^{D \times K} \mid 0 \leq \mathbf{A}_{d,k}, \forall (d, k)\}, \quad (3.5)$$

et une contrainte de parcimonie spatiale avec une régularisation  $l_1$  de la forme :

$$R(\mathbf{A}) = \lambda \|\mathbf{A}\|_1 = \lambda \sum_{k=1}^K \sum_{d=1}^D |\mathbf{A}_{d,k}|. \quad (3.6)$$

Nous allons dans la suite donner plus de détails sur ces fonctions.



### 3.1.2 Les fonctions objectifs

#### 3.1.2.1 Cas gaussien

Lorsque le bruit appliqué aux données est supposé blanc gaussien, le terme d'attache aux données a pour forme :

$$G_1(\mathbf{A}) = \frac{1}{2\sigma^2} \|\text{vec}(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\|^2. \quad (3.7)$$

Nous avons vu en section 2.4 que pour avoir des garanties de convergence des algorithmes que l'on utilise, il faut respecter certaines hypothèses. En effet, la fonction d'attache aux données  $G_1(\mathbf{A})$  doit satisfaire l'inégalité de **Kurdyka-Łojasiewicz** (voir 2.2.3). La proposition suivante, dont la démonstration est en annexe A.1.6, assure que  $G_1(\mathbf{A})$  respecte cette hypothèse.

#### Théorème 3.1.2.1: propriété de $G_1(\mathbf{A})$ dans le cas du bruit gaussien

*Le terme d'attache aux données  $G_1(\mathbf{A})$  pour le cas du bruit gaussien*

$$G_1(\mathbf{A}) = \frac{1}{2\sigma^2} \|\text{vec}(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\|^2,$$

*est une fonction analytique réelle, par conséquent d'après la proposition donnée en (2.38),  $G_1(\mathbf{A})$  satisfait l'inégalité de **Kurdyka-Łojasiewicz**.*

D'autre part, les méthodes algorithmiques citées en 2.4 requièrent le calcul du gradient du terme d'attache aux données  $G_1(\mathbf{A})$  à chaque itération. A l'aide des propriétés du calcul matriciel (voir annexe A.1.8), on peut montrer que ce gradient s'écrit :

$$\nabla G_1(\mathbf{A}) = 2\mathbf{S}^t \left( (\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma} \right) \Sigma^t, \quad (3.8)$$

où  $\odot$  désigne le produit de **Hadamard** (ou produit point par point), pour deux matrices de même taille  $\mathbf{X}$  et  $\mathbf{Y} \in \mathbb{R}^{n \times m}$  :

$$(\mathbf{X} \odot \mathbf{Y})_{ij} = X_{ij}Y_{ij}.$$

Par ailleurs, on montre aussi dans A.1.8 que la hessienne de  $G_1(\mathbf{A})$  s'exprime :

$$\mathbf{HG}_1(\mathbf{A}) = 2(\Sigma \otimes \mathbf{S}^t) \left( \mathbf{Q}_1(\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_K)\mathbf{Q}_1 + \mathbf{Q}_2 \right) (\Sigma^t \otimes \mathbf{S}), \quad (3.9)$$

avec

$$\mathbf{Q}_1 = \text{diag}\left(\text{vec}\left(e^{-\mathbf{SA}\Sigma}\right)\right)$$

$$\mathbf{Q}_2 = \text{diag}\left(\text{vec}\left((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot -e^{-\mathbf{SA}\Sigma}\right)\right).$$

Ici  $\text{diag}(V)$ , pour  $V \in \mathbb{R}^n$ , est la matrice de  $\mathbb{R}^{n \times n}$  dont la diagonale est composée des éléments de  $V$ . Par ailleurs,  $\otimes$  dénote le produit de **Kronecker**, un cas particulier du **produit tensoriel** [SCHAFER, 1966]. Pour deux matrices  $\mathbf{X} \in \mathbb{R}^{n \times m}$  et  $\mathbf{Y} \in \mathbb{R}^{p \times q}$ , le produit de Kronecker est la matrice  $(\mathbf{X} \otimes \mathbf{Y}) \in \mathbb{R}^{np \times mq}$  définie par blocs successifs de taille  $p \times q$  de la manière suivante :

$$(\mathbf{X} \otimes \mathbf{Y}) = \begin{pmatrix} x_{11}\mathbf{Y} & \cdots & x_{1m}\mathbf{Y} \\ \vdots & \ddots & \vdots \\ x_{n1}\mathbf{Y} & \cdots & x_{nm}\mathbf{Y} \end{pmatrix}.$$

La matrice hessienne  $\mathbf{HG}_1(\mathbf{A})$  est de taille  $\text{DK} \times \text{DK}$  où  $D$  est le nombre de voxels caractérisant la taille de l'objet et  $K$  le nombre de composants de l'objet. Donc plus l'objet est de grande taille plus la dimension de la hessienne augmente.

C'est à partir de l'expression de  $\mathbf{HG}_1(\mathbf{A})$  qu'on construit une métrique adaptée aux algorithmes proximaux à métriques variables présentés en section 2.4 du chapitre 2.

### 3.1.2.2 Cas Poisson

Lorsque le bruit appliqué aux données est supposé de Poisson, le terme d'attache aux données a pour forme :

$$G_2(\mathbf{A}) = \sum_{p,m} \{e^{-\text{SA}\Sigma \mathbf{F}}\}_{p,m} - Y_{p,m} \log(\{e^{-\text{SA}\Sigma \mathbf{F}}\}_{p,m}). \quad (3.10)$$

Dans ce cas de figure également, la fonction  $G_2(\mathbf{A})$  satisfait l'inégalité de **Kurdyka-Łojasiewicz**.

#### Théorème 3.1.2.2: propriété de $G_2(\mathbf{A})$ dans le cas du bruit de Poisson

*Le terme d'attache aux données  $G_2(\mathbf{A})$  pour le cas du bruit de Poisson*

$$G_2(\mathbf{A}) = \sum_{p,m} \{e^{-\text{SA}\Sigma \mathbf{F}}\}_{p,m} - Y_{p,m} \log(\{e^{-\text{SA}\Sigma \mathbf{F}}\}_{p,m}),$$

*est une fonction analytique réelle, par conséquent d'après la proposition donnée en (2.38),  $G_2(\mathbf{A})$  satisfait l'inégalité de **Kurdyka-Łojasiewicz**.*

La démonstration de cette proposition est en annexe A.1.7. On trouvera également en annexe A.1.9 le détail du calcul du gradient et de la hessienne du terme d'attache aux données  $G_2(\mathbf{A})$  pour le cas Poisson. On peut réécrire la fonction d'attache aux données (3.10) de manière plus générale de la façon suivante :

$$G_2(\mathbf{A}) = (e^{-\text{SA}\Sigma \mathbf{F}} - \mathbf{Y} \odot \log(e^{-\text{SA}\Sigma \mathbf{F}})) \cdot \mathbf{W}, \quad (3.11)$$

où  $\mathbf{W} \in \mathbb{R}^{P \times M}$  est une matrice de poids qui pondère le terme général et le produit de **Frobenius** (noté ici «  $\cdot$  ») de deux matrices de même taille est défini comme le produit **scalaire** suivant :

$$\mathbf{X} \cdot \mathbf{Y} = \sum_{i,j} \mathbf{X}_{ij} \mathbf{Y}_{ij}.$$

En pratique nous considérons des poids de valeur 1, mais les calculs du gradient et de la hessienne se font dans le cas général. En effet, le gradient de  $G_2(\mathbf{A})$  pour le cas du bruit de Poisson a pour expression :

$$\nabla G_2(\mathbf{A}) = \mathbf{S}^t \left( \left( \mathbf{W} \odot \left( \mathbf{Y} \odot e^{-\text{SA}\Sigma \mathbf{F}} - \mathbb{1}_{P \times M} \right) \right) \mathbf{F}^t \odot e^{-\text{SA}\Sigma} \right) \Sigma^t, \quad (3.12)$$

où  $\mathbb{1}_{P \times M}$  est une matrice de taille  $P \times M$  contenant 1 et  $\odot$  est la division de **Hadamard** appliquée point par point (voir définition en A.1.8.1). On montre également en A.1.9 que l'expression de la hessienne est la suivante :

$$\mathbf{H}G_2(\mathbf{A}) = (\Sigma \otimes \mathbf{S}^t) \left( \mathbf{D}_1 \left( \mathbf{F} \otimes \mathbf{I}_P \right) \mathbf{D}_2 \left( \mathbf{F}^t \otimes \mathbf{I}_P \right) \mathbf{D}_1 + \mathbf{D}_3 \right) (\Sigma^t \otimes \mathbf{S}), \quad (3.13)$$

avec

$$\mathbf{D}_1 = \text{diag} \left( \text{vec} \left( e^{-\text{SA}\Sigma} \right) \right) \quad (3.14)$$

$$\mathbf{D}_2 = \text{diag} \left( \text{vec} \left( (\mathbf{W} \odot \mathbf{Y}) \odot (e^{-\text{SA}\Sigma \mathbf{F}})^2 \right) \right) \quad (3.15)$$

$$\mathbf{D}_3 = \text{diag} \left( \text{vec} \left( (\mathbf{W} \odot (\mathbb{1}_{P \times M} - \mathbf{Y} \odot e^{-\text{SA}\Sigma \mathbf{F}})) \mathbf{F}^t \odot e^{-\text{SA}\Sigma} \right) \right). \quad (3.16)$$

Notons que dans la formule (3.15), la fonction  $X \mapsto X^2$  est appliquée à chaque élément de la matrice  $X$  telle que

$$X_{i,j}^2 = (X_{i,j})^2.$$

On remarque que la hessienne en équation (3.13) pour le cas du bruit de Poisson est assez similaire à la hessienne pour le cas du bruit gaussien en équation (3.9). Ceci nous aide à construire une métrique qui fonctionne assez bien dans les deux cas. Dans le prochain paragraphe nous allons voir comment construire une telle métrique.

### 3.1.3 Choix de la métrique

#### 3.1.3.1 Cas bruit gaussien

Nous avons vu dans la section 2.4 du chapitre 2 que pour garantir la convergence des algorithmes proximaux à métriques, certaines conditions doivent être respectées. On rappelle que l'incorporation d'une métrique est une stratégie consistant à accélérer la convergence des algorithmes proximaux [BECKER et FADILI, 2012; BONNANS et collab., 1995; CHEN et ROCKAFELLAR, 1997; COMBETTES et VŮ, 2014; PARENTE et collab., 2008]. On trouvera en sous-section 2.4.3.4 du chapitre 2 les détails mathématiques sur les hypothèses que doit satisfaire la métrique. Les deux principales caractéristiques que l'on souhaite avoir pour une métrique sont les suivantes :

- on souhaite une métrique facile à inverser et dont le calcul de l'inverse est peu coûteux, car on peut être amené à effectuer ce calcul à chaque itération de l'algorithme,
- on souhaite une métrique telle que l'approximation quadratique, à une itération  $k$  donnée, de la fonction  $G(\mathbf{A})$  construite avec cette métrique soit la plus proche possible de  $G(\mathbf{A})$ . Autrement dit, on souhaite que la métrique contiennent de l'information de second ordre sur la fonction  $G(\mathbf{A})$ .

On sait que la matrice hessienne contient l'information de second ordre nécessaire pour cela, c'est la raison pour laquelle elle est utilisée dans la descente de Newton (voir sous-section 2.4.1.2). Pour le cas du bruit gaussien, on rappelle que la hessienne s'exprime comme suit :

$$\mathbf{HG}_1(\mathbf{A}) = 2(\Sigma \otimes \mathbf{S}^t) \left( \mathbf{Q}_1 (\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_K) \mathbf{Q}_1 + \mathbf{Q}_2 \right) (\Sigma^t \otimes \mathbf{S}),$$

avec

$$\mathbf{Q}_1 = \mathbf{diag}(\mathbf{vec}(e^{-\mathbf{S}\mathbf{A}\Sigma}))$$

$$\mathbf{Q}_2 = \mathbf{diag}(\mathbf{vec}((\mathbf{Y} - e^{-\mathbf{S}\mathbf{A}\Sigma} \mathbf{F}) \mathbf{F}^t \odot -e^{-\mathbf{S}\mathbf{A}\Sigma})).$$

On ne peut prendre la hessienne  $\mathbf{HG}_1(\mathbf{A})$  comme métrique, car celle-ci n'est pas définie positive à cause du terme résiduel  $\mathbf{Q}_2$ . Une première métrique qu'on peut utiliser est l'approximation de **Gauss-Newton** de la hessienne, négligeant le terme  $\mathbf{Q}_2$ , on obtient ainsi :

$$\mathbf{M}_{\text{GN}} = 2(\Sigma \otimes \mathbf{S}^t) (\mathbf{Q}_1 (\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_K) \mathbf{Q}_1) (\Sigma^t \otimes \mathbf{S}). \quad (3.17)$$

On remarque que négliger  $\mathbf{Q}_2$  a du sens, car on peut supposer qu'à chaque itération le modèle se rapproche plus des données, ainsi le terme  $(\mathbf{Y} - e^{-\mathbf{S}\mathbf{A}\Sigma} \mathbf{F})$  se rapproche de plus en plus de zéro conduisant par conséquent le terme  $\mathbf{Q}_2$  à se rapprocher de zéro.

La métrique  $\mathbf{M}_{\text{GN}}$  est un premier bon candidat pour les algorithmes proximaux, car elle est proche de la hessienne et contient donc de l'information de second ordre. Cependant, elle dépend de l'inconnue  $\mathbf{A}$  et nécessite donc de calculer son inverse à chaque itération, ce qui peut se révéler coûteux. On remarque que le terme  $\mathbf{Q}_1$  dans  $\mathbf{M}_{\text{GN}}$  est le seul qui dépend de  $\mathbf{A}$ , or ce terme peut être majoré par l'identité. En effet, on rappelle que la matrice des cartes de concentrations  $\mathbf{A}$  ne contient que des éléments positifs et il en va de même pour la matrice de projection  $\mathbf{S}$  et la matrice des sections efficaces  $\Sigma$ , on a donc bien  $\mathbf{S}\mathbf{A}\Sigma \geq 0$  et, par définition de l'exponentielle, on a :

$$\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{S}\mathbf{A}\Sigma})) \leq \mathbf{I}_{\mathbf{P} \times \mathbf{N}}.$$

En tenant compte de cela dans la métrique  $\mathbf{M}_{\text{GN}}$ , on peut construire une nouvelle métrique :

$$\mathbf{M}_1 = 2(\Sigma \otimes \mathbf{S}^t) (\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_K) (\Sigma^t \otimes \mathbf{S}), \quad (3.18)$$

que les propriétés du produit de Kronecker nous permettent de simplifier en :

$$\mathbf{M}_1 = 2(\Sigma \mathbf{F}\mathbf{F}^t \Sigma^t \otimes \mathbf{S}^t \mathbf{S}). \quad (3.19)$$

Cette métrique est idéale pour nos algorithmes proximaux, car elle respecte les deux principales conditions citées précédemment. C'est la métrique  $\mathbf{M}_1$  que nous allons utiliser lors de nos calculs.

### 3.1.3.2 Cas bruit de Poisson

On rappelle que la hessienne de  $G_2(\mathbf{A})$  pour le cas du bruit de Poisson s'exprime :

$$\mathbf{H}G_2(\mathbf{A}) = (\Sigma \otimes \mathbf{S}^t) \left( \mathbf{D}_1 (\mathbf{F} \otimes \mathbf{I}) \mathbf{D}_2 (\mathbf{F}^t \otimes \mathbf{I}) \mathbf{D}_1 + \mathbf{D}_3 \right) (\Sigma^t \otimes \mathbf{S}),$$

avec

$$\begin{aligned} \mathbf{D}_1 &= \text{diag}(\text{vec}(e^{-\mathbf{SA}\Sigma})) \\ \mathbf{D}_2 &= \text{diag}(\text{vec}((\mathbf{W} \odot \mathbf{Y}) \odot (e^{-\mathbf{SA}\Sigma} \mathbf{F})^2)) \\ \mathbf{D}_3 &= \text{diag}(\text{vec}((\mathbf{W} \odot (\mathbb{1}_{P \times M} - \mathbf{Y} \odot e^{-\mathbf{SA}\Sigma} \mathbf{F})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma})). \end{aligned}$$

Tout comme pour le cas précédent, on peut construire différentes métriques en considérant différentes approximations. En effet, l'hypothèse que les données collent au modèle induit que le terme  $\mathbf{Y} \odot e^{-\mathbf{SA}\Sigma} \mathbf{F}$  s'approche de la matrice  $\mathbb{1}_{P \times M}$ , ainsi on peut construire la métrique de **Gauss-Newton** en négligeant  $\mathbf{D}_3$ . On obtient ainsi :

$$\mathbf{M}_{\text{GNP}} = (\Sigma \otimes \mathbf{S}^t) \left( \mathbf{D}_1 (\mathbf{F} \otimes \mathbf{I}) \mathbf{D}_2 (\mathbf{F}^t \otimes \mathbf{I}) \mathbf{D}_1 \right) (\Sigma^t \otimes \mathbf{S}). \quad (3.20)$$

On peut par ailleurs majorer la matrice  $\mathbf{D}_1$  par  $\mathbf{I}_{\text{PN}}$  pour construire la métrique :

$$\mathbf{M}_{\text{P1}} = \max(\mathbf{W})^2 (\Sigma \otimes \mathbf{S}^t) \left( (\mathbf{F} \otimes \mathbf{I}) \mathbf{D}_2 (\mathbf{F}^t \otimes \mathbf{I}) \right) (\Sigma^t \otimes \mathbf{S}). \quad (3.21)$$

Par ailleurs, l'hypothèse que les données suivent bien le modèle induit le fait que l'on puisse considérer l'équivalence  $\mathbf{Y} \simeq e^{-\mathbf{SA}\Sigma} \mathbf{F}$  et par continuité de la fonction  $X \mapsto X^2$ ,  $\mathbf{Y}^2 \simeq (e^{-\mathbf{SA}\Sigma} \mathbf{F})^2$ . Ainsi, la matrice  $\mathbf{D}_2$  peut être majorée par  $\frac{\max(\mathbf{W})}{|\min(\mathbf{Y})|} \mathbf{I}_{\text{PN}}$ , donnant ainsi la métrique :

$$\mathbf{M}_{\text{P2}} = C (\Sigma \otimes \mathbf{S}^t) \left( (\mathbf{F} \otimes \mathbf{I}) (\mathbf{F}^t \otimes \mathbf{I}) \right) (\Sigma^t \otimes \mathbf{S}), \quad (3.22)$$

où  $C = \frac{\max(\mathbf{W})}{|\min(\mathbf{Y})|}$ . On remarque alors que, grâce aux propriétés de calcul du produit de Kronecker, on obtient :

$$\mathbf{M}_{\text{P2}} = C (\Sigma \mathbf{F} \mathbf{F}^t \Sigma^t \otimes \mathbf{S}^t \mathbf{S}). \quad (3.23)$$

On en conclut que les métriques pour le cas du bruit de Poisson  $\mathbf{M}_{\text{P2}}$  et du bruit gaussien  $\mathbf{M}_1$  sont égales à un facteur multiplicatif près :

$$\mathbf{M}_1 = C' \mathbf{M}_{\text{P2}}. \quad (3.24)$$

**Remarque** : On ne stocke pas en mémoire la métrique  $\mathbf{M}_1$  qui est de taille  $DK \times DK$ , mais on calcule son action sur  $\mathbf{A}$  en tirant parti du produit de Kronecker. En effet, l'action de  $\mathbf{S}^t \mathbf{S}$  agit indépendamment sur les colonnes de  $\mathbf{A}$  et l'action de  $\Sigma \mathbf{F} \mathbf{F}^t \Sigma^t$  agit indépendamment sur les lignes de  $\mathbf{A}$ . Ainsi on calcule dans un premier temps l'action du filtre conique sur les colonnes de  $\mathbf{A}$ , puis on multiplie le résultat par l'opérateur  $\Sigma \mathbf{F} \mathbf{F}^t \Sigma^t$ .

L'autre avantage de la métrique  $\mathbf{M}_1$  est que son expression ne dépend pas de  $\mathbf{A}$ , c'est une métrique qui **ne varie pas** et qui ne nécessite donc pas le calcul de son inverse à chaque itération. Il suffit donc de calculer son inverse une seule fois et de l'utiliser à chaque itération, ce qui représente un gain de temps considérable. Nous allons voir dans la sous-section suivante comment calculer l'inverse de la métrique  $\mathbf{M}_1$ .

### 3.1.4 Filtre conique

La métrique  $\mathbf{M}_1$  est composée de deux termes : le terme  $\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t$  et l'opérateur de **projection-rétroprojection**  $\mathbf{S}^t\mathbf{S}$ . Remarquons tout d'abord que l'opérateur  $\mathbf{S}^t\mathbf{S}$  est un candidat naturel en tant que métrique pour le cas de la **TDM** d'absorption classique [PERELLI et collab., 2016; RAMANI et FESSLER, 2012]. Si on étudie l'action de l'opérateur  $\mathbf{S}^t\mathbf{S}$  dans le cas continu sur une fonction à deux dimensions  $f$ , on trouve alors le résultat suivant [FESSLER, 2017] :

$$\mathbf{S}^t\mathbf{S}f = f \star h, \quad (3.25)$$

où  $\star$  dénote le produit de convolution défini au chapitre 1 à l'équation (1.26) et  $h$  est le filtre **rampe**. Nous démontrons en annexe A.1.10 ce résultat pour le cas général où les projections sont pondérées par une matrice  $\mathbf{W}$ .

En effet, dans le cas discret, on obtient :

$$\mathbf{S}^t\mathbf{S}f = \mathcal{F}^{-1} \left[ \mathcal{F}[f] \circ \mathcal{F}[h] \right], \quad (3.26)$$

où  $\mathcal{F}$  est l'opérateur de transformée de Fourier et  $\mathcal{F}^{-1}$  est son opérateur inverse. Il devient alors simple d'inverser  $\mathbf{S}^t\mathbf{S}$  en inversant  $\mathcal{F}[h]$  :

$$(\mathbf{S}^t\mathbf{S})^{-1}f = \mathcal{F}^{-1} \left[ \mathcal{F}[f] \oslash \mathcal{F}[h] \right],$$

où  $\oslash$  est la division de Hadamard. Ainsi, appliquer l'inverse de l'opérateur  $\mathbf{S}^t\mathbf{S}$  revient juste à effectuer un produit de convolution dans le domaine réel et donc un produit de Hadamard dans le domaine de Fourier. C'est une opération de filtrage et on appelle  $\mathcal{F}[h]$  **filtre conique** (en anglais *cone filter*). Ainsi, lorsqu'on applique la métrique  $\mathbf{M}_1$  à l'étape de descente dans le schéma des algorithmes proximaux, on utilise l'inverse du filtre conique  $\mathcal{F}[h]$  à la place du terme  $(\mathbf{S}^t\mathbf{S})^{-1}$ .

L'égalité (3.25) est vraie lorsque la géométrie du faisceau est en 2D (parallèle ou en éventail), mais ce n'est pas le cas lorsque le faisceau est conique pour des raisons de complétude de données expliquées en sous-section 1.2.3 du chapitre 1. On peut pallier cette difficulté en utilisant une méthode analogue à celle utilisée dans l'algorithme **FDK** que l'on a présenté en sous-section 1.2.3.6 du chapitre 1.

Dans cette méthode, on considère que le calcul effectué à chaque plan de l'objet est similaire à celui effectué au plan médian, pondéré par des facteurs qui dépendent de la géométrie (voir équation (1.28)).

L'idée ici serait donc d'insérer une matrice diagonale qui pondère les coefficients de la métrique  $\mathbf{M}_1$ . Or effectuer ce calcul ferait perdre la structure de la métrique et rendrait donc plus difficile le calcul de l'inverse.

Nous négligeons donc ces facteurs de pondération et nous appliquons dans nos expériences le **même** filtre pour chaque coupe de projection. De cette manière, et pour tous les cas géométriques traités, on calcule l'inverse de la métrique  $\mathbf{M}_1$  comme suit :

$$\begin{aligned} \mathbf{M}_1^{-1} &= \left( 2 \left( \Sigma\mathbf{F}\mathbf{F}^t\Sigma^t \otimes \mathbf{S}^t\mathbf{S} \right) \right)^{-1} \\ &= \frac{1}{2} \left( \left( \Sigma\mathbf{F}\mathbf{F}^t\Sigma^t \right)^{-1} \otimes \left( \mathbf{S}^t\mathbf{S} \right)^{-1} \right). \end{aligned}$$

Or le terme  $(\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t)^{-1}$  est simple à inverser<sup>1</sup> car c'est une matrice carrée de taille  $K \times K$ , avec  $K$  le nombre de composants de l'objet qui est toujours assez petit ( $K \leq 5$  dans notre cas). Même si

1. Il faut tout de même veiller au bon conditionnement des matrices  $\Sigma$  et  $\mathbf{F}$  en normalisant les colonnes de celles-ci par exemple.

aucun résultat théorique ne confirme que ce produit est inversible, il s'est toujours avéré inversible numériquement. Quant au second terme il suffit d'utiliser le filtre conique  $\mathcal{F}[h]$  que l'on a vu précédemment.

Un autre paramètre important des algorithmes proximaux dont il faut tenir compte est le pas de descente, car ce dernier est crucial pour la garantie de convergence des algorithmes et il doit respecter certaines conditions. Ce choix de pas est discuté dans la prochaine sous-section.

### 3.1.4.1 Appodisation du filtre

Comme nous l'avons mentionné en sous-section 1.2.3.4 du chapitre 1, le filtre rampe est un filtre **passé-haut** qui conserve le signal en haute fréquence et filtre le signal à basse fréquence. Les erreurs d'interpolation et le bruit sont principalement des signaux hautes fréquences, le filtre rampe a tendance à les amplifier et devient très sensible au bruit lorsqu'il s'agit de données expérimentales.

Afin d'améliorer la robustesse de nos reconstructions et éviter d'amplifier le bruit, on utilise une **fenêtre d'apodisation** qui atténue le filtrage à haute fréquence. Il existe plusieurs types de fenêtres d'apodisation [ACTON et OF AMERICA, 1990]. Celle que nous avons employée lors de nos expériences est la fenêtre de **Hamming**. La figure 3.4 illustre le filtre que l'on utilise et ses versions apodisées.

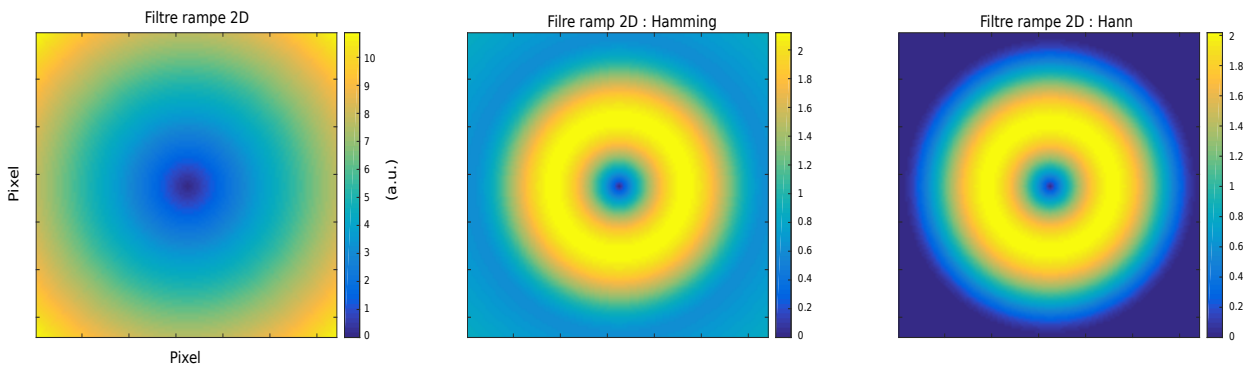


FIGURE 3.4 – À Gauche, le filtre conique 2D de taille  $128 \times 128$ . Au milieu, sa version apodisée par une fenêtre de Hamming. À droite, sa version apodisée par une fenêtre de Hann.

## 3.1.5 Choix du pas de descente

### 3.1.5.1 Algorithme FISTA

Nous savons que la constante de majoration  $L$  pour l'algorithme **FISTA** doit respecter l'inégalité suivante (voir sous-section 2.4.3 du chapitre 2) :

$$0 < L_G \leq L,$$

où  $L_G$  est la constante de Lipschitz du gradient de la fonction d'attache aux données du modèle. On rappelle que la constante  $L_G$  est définie comme suit :

$$L_G = \sup \{ L \in \mathbb{R}^+ \mid \forall \mathbf{A}, \mathbf{A}' \|\nabla G(\mathbf{A}) - \nabla G(\mathbf{A}')\|_{\text{fro}} \leq L \|\mathbf{A} - \mathbf{A}'\|_{\text{fro}} \} \quad (3.27)$$

$$L_G = \sup_{\mathbf{A}} \|\text{HG}(\mathbf{A})\|_2, \quad (3.28)$$

où  $\|\cdot\|_2$  est la norme opérateur et  $\|\cdot\|_{\text{fro}}$  est la norme de Frobenius. Ces calculs peuvent être coûteux ou impossibles à effectuer. L'idée est alors de trouver un majorant de  $L_G$  en utilisant la norme opérateur de la métrique  $\mathbf{M}_1$ , car on a montré que celle-ci majore la hessienne  $\text{HG}(\mathbf{A})$ . Remarquons

que la norme opérateur de cette métrique s'écrit :

$$L_1 = \|\mathbf{M}_1\|_2 = 2\|(\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t \otimes \mathbf{S}^t\mathbf{S})\|_2,$$

que l'on peut simplifier grâce aux propriétés du produit de Kronecker :

$$L_1 = 2\|(\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t \otimes \mathbf{S}^t\mathbf{S})\|_2 = 2\|\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t\|_2\|\mathbf{S}^t\mathbf{S}\|_2.$$

On rappelle que pour une matrice  $X$ , la norme opérateur est définie par :

$$\|X\|_2 = \max_{\lambda} \{|\lambda| \mid \lambda \in \mathbb{R} \text{ est valeur propre de } X\}.$$

Ainsi on peut aisément calculer numériquement la norme  $\|\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t\|$ , car  $\Sigma\mathbf{F}\mathbf{F}^t\Sigma^t$  est une matrice de taille  $K \times K$ .

La norme de  $\|\mathbf{S}^t\mathbf{S}\|_2$  dépend des paramètres de discrétisation et donc directement de la routine de calcul utilisée pour calculer les projections et les rétroprojections. On rappelle que dans notre cas, nous utilisons *OpenRTK Toolkit*, un software développé pour la reconstruction rapide pour la TDM à faisceau conique [RIT et collab., 2014]. Le choix des paramètres de discrétisation n'est pas clairement explicité dans les programmes de *OpenRTK Toolkit*. Nous avons majoré la norme de l'opérateur  $\|\mathbf{S}^t\mathbf{S}\|_2$  de manière empirique et trouvé que

$$\|\mathbf{S}^t\mathbf{S}\|_2 \leq \text{Lop} := 2PN_v^4, \quad (3.29)$$

avec  $P$  est le nombre de mesures et  $N_v$  la taille<sup>2</sup> d'un voxel de l'objet.

### 3.1.5.2 Algorithmes VMFB et VMILAn

Le pas de descente pour les algorithmes **VMFB** et **VMILAn** est soumis à un large choix. En effet, les auteurs de [BONETTINI et collab., 2017] stipulent que le pas doit juste appartenir à l'ensemble des réels strictement positifs ( $\mathbb{R}_*^+$ ).

On a vu précédemment que le schéma de descente de gradient pour **FISTA** était :

$$\mathbf{A}_{k+1} = \mathbf{A}_k - \gamma_k \nabla G(\mathbf{A}_k),$$

avec  $\gamma_k \sim \frac{1}{L}$ , où  $L$  majore la constante de Lipschitz du gradient de  $G(\mathbf{A})$ . Ici le schéma de descente de gradient est le suivant :

$$\mathbf{A}_{k+1} = \mathbf{A}_k - \gamma_k \mathbf{M}^{-1} \nabla G(\mathbf{A}_k),$$

avec  $\mathbf{M}$  la métrique utilisée. Par « mimétisme » avec cas précédent, on choisit un pas  $\gamma_k \sim \frac{1}{L\|\mathbf{M}^{-1}\|_2}$ , avec  $L$  une majorante de la constante de Lipschitz de  $\nabla G$ . Le dénominateur se simplifie pour obtenir :

$$\frac{\|\mathbf{S}^t\mathbf{S}\|_2}{\|\mathcal{F}[h]\|_2},$$

avec  $\mathcal{F}[h]$  le filtre conique. Ce rapport de norme doit en théorie être égal à 1, mais en pratique des erreurs d'approximations peuvent apparaître à cause de l'échantillonnage des matrices de projection et rétroprojection.

En effet, ces erreurs sont dues au passage du cas continu au cas discret, mais peuvent être approchées de manière empirique.

Afin de quantifier cette erreur numérique entre  $\mathbf{S}^t\mathbf{S}$  et le filtre conique, nous sélectionnons un ensemble de vecteurs aléatoires  $\mathbf{V}^1 = \{v_n\}_{n \in \mathbb{I}}$ , tous de norme unitaire, avec  $\mathbb{I}$  un sous-ensemble de  $\mathbb{N}$ , et nous calculons le rapport suivant :

$$\hat{\alpha} = \max \left\{ \frac{\|\mathbf{S}^t\mathbf{S}v\|}{\|\mathcal{F}[h]v\|} \mid v \in \mathbf{V}^1 \right\}. \quad (3.30)$$

Ce rapport permet de donner une estimation de l'erreur entre la norme  $\|\mathbf{S}^t\mathbf{S}\|_2$  et la norme du filtre conique. On trouve dans nos expériences que  $\hat{\alpha} \approx 1.2$ . Ce paramètre permet d'ajuster la descente de nos algorithmes à métrique.

2. Cette taille représente un rapport de distance.

## Conclusion

Cette première section avait pour but de déterminer les paramètres des algorithmes utilisés. Le premier calcul important à faire est le gradient des fonctions d'attaches aux données, car l'expression de celui-ci est utilisée dans tous les algorithmes que l'on a présentés en section 2.4. Par la même occasion, nous avons montré que les fonctions d'attaches aux données satisfaisaient l'inégalité de **Kurdyka-Łojasiewicz**. Ainsi, on peut garantir la convergence des algorithmes dans le cas de la **TDM** spectrale.

Nous avons par ailleurs montré l'expression de la hessienne des termes d'attaches aux données. C'est à partir de l'expression de la hessienne qu'une métrique et un pas de descente ont été déduits pour les algorithmes proximaux à métriques variables. Nous avons vu aussi que l'expression de cette métrique est simple et permet de calculer son inverse rapidement.

Ces outils sont les derniers éléments théoriques qui permettent d'élaborer des protocoles de test pour les algorithmes. Dans les sections suivantes, nous allons présenter ces différents protocoles et commenter les différents résultats obtenus.

## 3.2 Tests et résultats sur données simulées

Nous allons dans cette section présenter les résultats obtenus sur des données simulées qui nous permettent de tester nos algorithmes dans un cas **maîtrisé**. À l'aide de l'étude de ces données, nous avons pu améliorer la robustesse des algorithmes et affiner les valeurs des différents paramètres des algorithmes.

Nous présenterons dans un premier temps les résultats obtenus sur des données issues de la **TDM** d'absorption classique, notre propos étant de mesurer l'efficacité de nos algorithmes par rapport à l'état de l'art.

Nous présenterons ensuite les résultats obtenus sur des données de **TDM** spectrale en deux dimensions. Enfin nous présenterons les résultats obtenus pour les données de **TDM** spectrale en trois dimensions.

### 3.2.1 Tests sur données de TDM d'absorption classique

Une première phase de test consiste à étudier les résultats que l'on peut obtenir sur des données simples issues de la **TDM** d'absorption classique. Ceci permet de voir si d'une part nos algorithmes sont capables de résoudre un tel problème et comment ceux-ci peuvent être comparés à des méthodes standards, comme la rétro-projection filtrée vue en sous-section 1.2.3.5 du chapitre 1. Si l'on veut se ramener à un cas simple impliquant l'opérateur **S**, on peut considérer le cas d'une projection sur un objet à une seule composante et ceci pour une seule acquisition. Ceci revient à considérer que dans notre modèle (3.46), le nombre de composants de l'objet est  $K = 1$  et le nombre d'acquisitions est  $M = 1$ . Ainsi les matrices **F** et  $\Sigma$  deviennent des vecteurs.

L'objet que l'on va étudier est le **fantôme de Shepp-Logan** [SHEPP et LOGAN, 1974] dont une illustration est montrée en figure 3.5.

#### 3.2.1.1 Protocole de test

Les protocoles de test que nous allons présenter seront sous forme de fiches. On y trouvera l'essentiel de l'information, les résultats seront présentés par la suite.





FIGURE 3.5 – Fantôme de Shepp-Logan de taille 64×64 pixels.

**Protocole 3.2.1.1: test sur le fantôme de Shepp-Logan**

**Objet :** fantôme de Shepp-Logan.

**Taille :**  $D = 64 \times 64$  pixels.

**Nombre de projections (angles) :** 180.

**Nombre de composants :**  $K = 1$ .

**Nombre d'acquisitions :**  $M = 1$ .

**Nombre total de mesures :**  $P = 64 \times 180 = 1.1 \times 10^4$ .

**Source  $I_0(E)$  :** la géométrie est parallèle et l'énergie de la source est une constante.

**Efficacité  $D(E)$  :** elle est constante pour tous les pixels du détecteur.

**Réponse du filtre  $F(E)$  :** pas de filtre dans ce test.

**Dictionnaire des sections efficaces  $\Sigma$  :** pas de composants à identifier.

Le sinogramme en figure 3.6, éventuellement bruité, constitue les données de départ, c'est l'en-

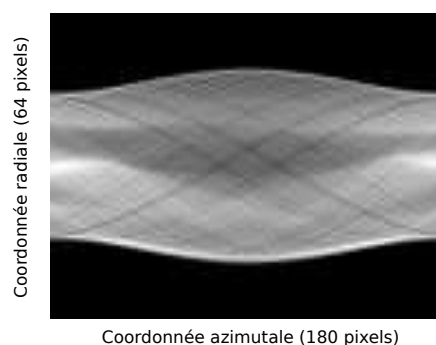


FIGURE 3.6 – Sinogramme obtenu à partir de la projection du fantôme de Shepp-Logan.

trée principale des algorithmes. Les performances seront mesurées à l'aide du rapport signal sur bruit (SNR) que l'on a défini en sous-section 2.3.2 du chapitre 2.

Nous allons dans la suite présenter les résultats pour deux cas :

- un cas où l'on considère que le modèle avec les paramètres cités dans le protocole 3.2.1.1 est linéaire, *i.e.* on n'applique pas l'exponentielle à nos mesures après la projection :

$$Y_1 = SA + Br, \quad (3.31)$$

où  $Br$  est un bruit blanc gaussien. La fonction objectif est la suivante :

$$J_l(\mathbf{A}) = \|\mathbf{Y}_l - \mathbf{S}\mathbf{A}\|_{\text{fro}}^2 + R(\mathbf{A}),$$

avec  $R(\mathbf{A})$  la fonction de régularisation contraignant la positivité,

- un cas où l'on considère le modèle non-linéaire en appliquant l'exponentielle :

$$\mathbf{Y}_{\text{nl}} = e^{-\mathbf{S}\mathbf{A}} + Br, \quad (3.32)$$

où  $Br$  est un bruit blanc gaussien. La fonction objectif est la suivante :

$$J_{\text{nl}}(\mathbf{A}) = \|\mathbf{Y}_{\text{nl}} - e^{-\mathbf{S}\mathbf{A}}\|_{\text{fro}}^2 + R(\mathbf{A}),$$

avec  $R(\mathbf{A})$  la fonction de régularisation contraignant la positivité.

Le but d'étudier de tels cas est d'une part de vérifier que les algorithmes que l'on utilise sont capables de résoudre un cas simple et d'autre part de voir quelle serait la différence entre les résultats que l'on peut obtenir entre ces deux cas. Ceci permettra de vérifier la robustesse de nos méthodes de manière graduelle.

### 3.2.1.2 Cas linéaire simple sans bruit

Pour les premiers tests, nous avons voulu vérifier si les performances théoriques de l'algorithme **FISTA** se vérifient aussi en pratique et les comparer à la méthode **FBP**.

Remarquons que les versions des algorithmes **FISTA** et **ISTA** utilisées dans nos tests sont celles sans *backtracking*, *i.e.* on n'estime pas le pas de descente. En effet, nos expériences sont menées sur un objet de petite taille, nous pouvons donc calculer directement la constante de Lipschitz du gradient. Dans le cas de nos tests, la constante est la suivante :

$$L_f = \|2\mathbf{S}^t\mathbf{S}\| = 2 \max_{\lambda} \{|\lambda| \mid \lambda \in \mathbb{R} \text{ est valeur propre de } \mathbf{S}^t\mathbf{S}\}. \quad (3.33)$$

Les résultats présentés ici correspondent au cas où le sinogramme en figure 3.6 n'est pas bruité.

On peut déjà noter que la vitesse de convergence illustrée par la décroissance de la fonctionnelle

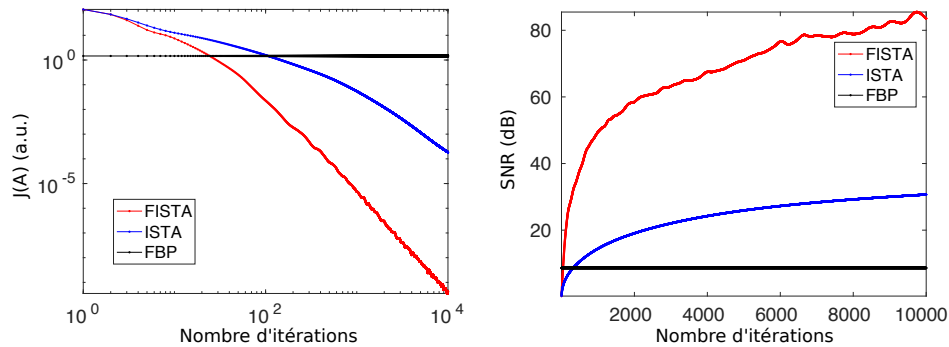


FIGURE 3.7 – A gauche, la décroissance en échelle logarithmique des fonctionnelles  $J(\mathbf{A})$  générées par les algorithmes **FISTA**, **ISTA** et la méthode **FBP** en fonction des itérations. A droite, l'évolution du **SNR** en fonction des itérations pour les algorithmes **FISTA**, **ISTA** et la méthode **FBP**.

en figure 3.7 correspond bien aux résultats théoriques présentés en sous-section 2.4.3 du chapitre 2. En effet, on constate que la décroissance du terme résiduel généré par l'algorithme **FISTA** est en  $\frac{1}{k^2}$ , avec  $k$  le nombre d'itérations, bien plus rapide que pour l'algorithme **ISTA**.

D'autre part, les performances en termes de **SNR** de l'approche classique **FBP** sont dépassées au bout de quelques dizaines d'itérations et l'image reconstruite par l'algorithme **FISTA** est plus proche de l'image originale comme on peut le voir en figure 3.8.

**Remarque :** Un artefact apparaît au centre de l'image résiduelle reconstruite par l'algorithme **FISTA** en figure 3.8. Il est causé par une erreur d'approximation numérique d'échantillonnage angulaire.

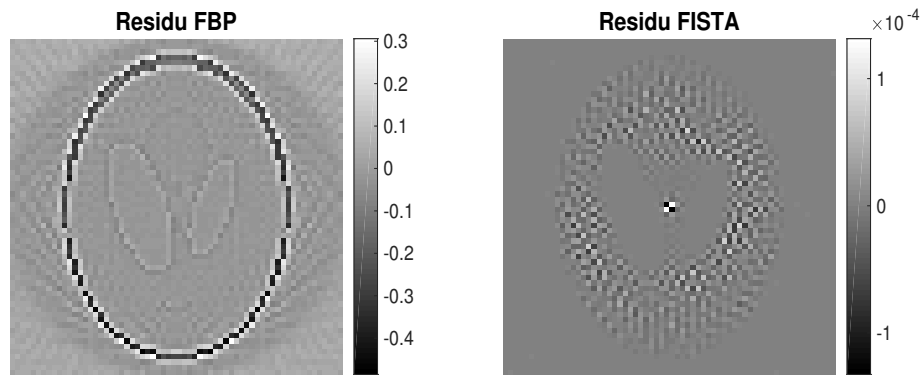


FIGURE 3.8 – Images résiduelles ( $64 \times 64$  pixels) pour la méthode **FBP** et l'algorithme **FISTA** : soustraction de la vérité de terrain et de l'image reconstruite par les différents algorithmes.

### 3.2.1.3 Cas linéaire simple avec bruit

Les résultats présentés ici correspondent au cas où le bruit appliqué au sinogramme en figure 3.6 est un bruit blanc gaussien. Le niveau de bruit est choisi de manière à avoir le **SNR** des données en entrée compris entre 10 et 40 dB, ce qui correspond en moyenne à ce que l'on peut rencontrer en application réelle.

On constate en figure 3.9 que dans le cas bruité aussi, la fonctionnelle que génère l'algorithme

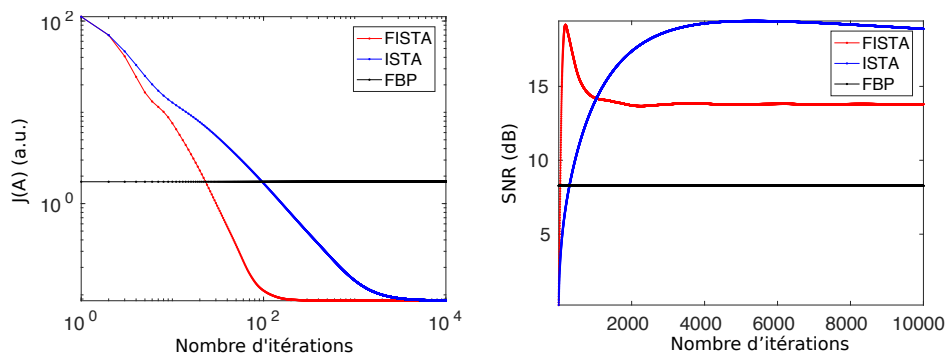


FIGURE 3.9 – **SNR** entrée : 20 dB - A gauche, décroissance en échelle logarithmique des fonctionnelles  $J(A)$  générées par les algorithmes **FISTA**, **ISTA** et la méthode **FBP** - A droite, **SNR** au cours des itérations pour les algorithmes **FISTA**, **ISTA** et la méthode **FBP**.

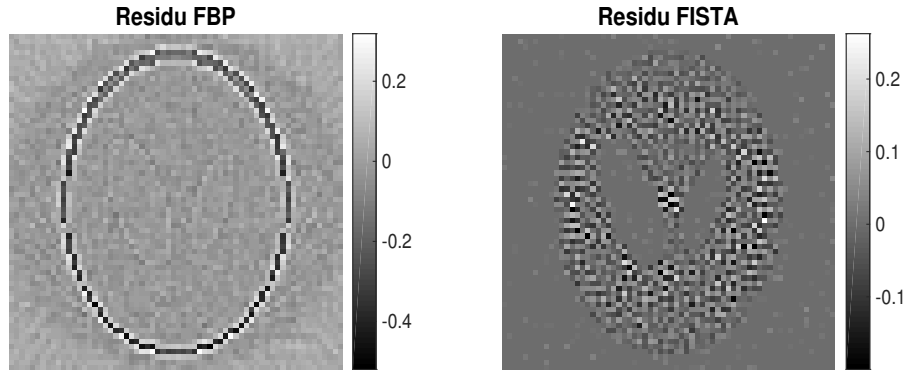
**FISTA** décroît plus vite, mais le **SNR** stagne au bout de 2000 itérations pour atteindre un niveau de 14 dB.

Là aussi, **FISTA** dépasse les performances de la méthode **FBP** au bout d'une dizaine d'itérations. Par ailleurs, le **SNR** de l'algorithme **ISTA** est meilleur que l'algorithme **FISTA**, mais ceci est seulement dû à la vitesse lente de convergence de l'algorithme **ISTA**. En effet, on remarque qu'en fin d'itération, le **SNR** décroît, il finira par stagner à 14 dB, mais pour un nombre d'itérations plus grand.

On peut expliquer cette faible robustesse au bruit des algorithmes à cause de la régularisation choisie. En effet, le terme de régularisation  $R(A)$  utilisé pour ces tests est la projection sur les positifs. Ceci peut être amélioré si on utilise la norme  $l_1$  [ZHANG et collab., 2016].

Par ailleurs, la dynamique des images reconstruites par la méthode **FBP** et l'algorithme **FISTA** est sensiblement la même, toutefois les contours de l'image sont mieux reconstruits avec l'algorithme **FISTA** comme on le constate sur la figure 3.10.

**Conclusion :**


 FIGURE 3.10 – Images résiduelles (64 × 64 pixels) pour la méthode **FBP** et l'algorithme **FISTA**

Les premiers tests sont encourageants. Nous avons vérifié que nos algorithmes atteignent bien les performances prédites par la théorie sur la vitesse de convergence. D'autre part, les performances des reconstructions standards telles que la méthode **FBP** sont rapidement dépassées. Dans la suite, nous allons présenter les résultats que l'on obtient sur un jeu de données non linéaires.

### 3.2.1.4 Cas non-linéaire sans bruit

Nous allons pour ces tests considérer le modèle non-linéaire en appliquant l'exponentielle au sinogramme en figure 3.6. Cela ne change pas la forme du sinogramme, mais seulement ses valeurs. L'objectif ici est de voir si nos méthodes peuvent résoudre efficacement ce type de problème.

Nous allons utiliser les algorithmes **FISTA** et **VMFB** pour résoudre cela et comparer les performances que l'on peut obtenir lorsqu'on utilise une métrique d'accélération de la convergence.

#### Calcul du point proximal

Nous rappelons que pour l'algorithme **VMFB**, il est nécessaire de calculer le point proximal de manière itérative. On résout à chaque itération  $k$  le sous problème suivant :

$$\mathbf{prox}_{L_{\text{nl}}, \mathbf{M}, \mathbf{R}}(\mathbf{A}_k) := \underset{z}{\operatorname{argmin}} \left\{ \frac{L_{\text{nl}}}{2} \|z - \mathbf{A}_k\|_{\mathbf{M}}^2 + \mathbf{R}(z) \right\}, \quad (3.34)$$

où  $\mathbf{M}$  est la métrique préalablement choisie et  $\mathbf{R}(z) = \chi_{\{z \geq 0\}}(z)$ , l'indicatrice sur l'ensemble des réels positifs définie au chapitre 2. Le problème (3.34) est un problème linéaire dont le gradient associé est :

$$\mathbf{G}_{\text{nl}}(\mathbf{A}) = L_{\text{nl}} \mathbf{M}(z - \mathbf{A}_k),$$

que l'on résout à l'aide de l'algorithme itératif **FISTA** en prenant pour paramètre :

- l'inverse de la norme de la métrique  $\mathbf{M}$  comme pas de descente du gradient,
- la projection<sup>3</sup> de  $\mathbf{A}_k$  sur les positifs comme initialisation,
- un critère d'arrêt : un nombre d'itérations déterminé (en général 200).

#### Comparaison des résultats :

Dans la suite, nous comparerons les résultats de l'algorithme **VMFB** à ceux de l'algorithme **FISTA** pour le cas où les données ne sont pas bruitées. Les paramètres pour l'algorithme **VMFB** sont les suivants :

3. Il s'agit de la projection directe qui consiste à mettre à zero tous les éléments négatifs de  $\mathbf{A}_k$ .

- **initialisation** : la même que celle utilisée pour l'algorithme **FISTA**, proche de la solution, *i.e.* la vérité de terrain bruitée,
- **métrique** : la hessienne, noté  $\mathbf{HG}_{\text{nl}}(\mathbf{A})$ , pour le cas non-linéaire est la suivante :

$$\mathbf{H}_{\text{nl}} := \mathbf{HG}_{\text{nl}}(\mathbf{A}) = 2\mathbf{S}^t(\mathbf{T}_1 + \mathbf{T}_2)\mathbf{S}, \quad (3.35)$$

avec

$$\mathbf{T}_1 = \text{diag}\left(\text{vec}\left(e^{-2\mathbf{S}\mathbf{A}}\right)\right),$$

$$\mathbf{T}_2 = \text{diag}\left(\text{vec}\left((e^{-\mathbf{S}\mathbf{A}} - \mathbf{Y}_{\text{nl}}) \odot e^{-\mathbf{S}\mathbf{A}}\right)\right).$$

D'ordinaire on ne peut pas utiliser la hessienne comme métrique pour les algorithmes proximaux. Nous avons donc utilisé une approximation de  $\mathbf{HG}_{\text{nl}}(\mathbf{A})$  qui consiste à mettre à zéro les éléments non-positifs du terme  $\mathbf{T}_2$ . On la définit comme suit :

$$\mathbf{M}_{\text{nl}}^1 := 2\mathbf{S}^t(\mathbf{T}_1 + \mathbf{T}_2|_{\geq 0})\mathbf{S}, \quad (3.36)$$

où  $\mathbf{T}_2|_{\geq 0}$  est le terme  $\mathbf{T}_2$  dont les éléments non-positifs sont mis à zéro. D'autre part, nous avons aussi utilisé l'opérateur  $\mathbf{S}^t\mathbf{S}$ , un candidat naturel en tant que métrique dans le cas de la **TDM** d'absorption classique [PERELLI et collab., 2016]. On définit alors cette métrique comme suit :

$$\mathbf{M}_{\text{nl}}^2 := \delta\mathbf{S}^t\mathbf{S}, \quad (3.37)$$

où  $\delta = 2 + \max |\mathbf{Y}_{\text{nl}}|$  est un facteur correctif permettant de mieux approximer la véritable hessienne  $\mathbf{HG}_{\text{nl}}(\mathbf{A})$ .

- **pas de descente** : l'inverse de la norme de la métrique, donc le même que pour l'algorithme **FISTA**,
- **paramètre de relaxation** : on fixe  $\lambda = 1$ , donc pas de relaxation,
- **critère d'arrêt** : un nombre d'itérations (ici 2000).

Les paramètres pour **FISTA** sont les suivants :

- **initialisation** : proche de la solution, *i.e.* la vérité de terrain bruitée,
- **pas de descente** : on utilise l'inverse de la norme de la métrique  $\mathbf{M}_{\text{nl}}^2$  suivante :

$$L_{\text{nl}} = (2 + \max |\mathbf{Y}_{\text{nl}}|)L_f, \quad (3.38)$$

où  $L_f$  est la constante de Lipschitz calculée pour le cas linéaire dans l'équation (3.33),

- **critère d'arrêt** : nombre d'itérations (ici 2000).

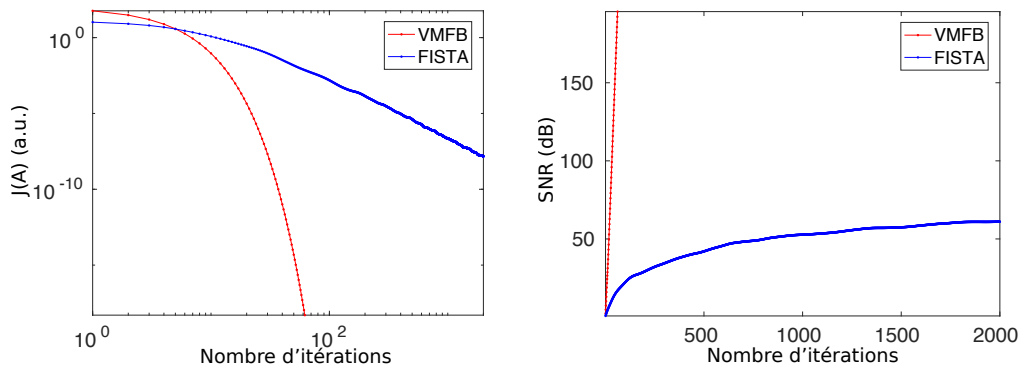


FIGURE 3.11 – Résultats pour  $\mathbf{M}_{\text{nl}}^1$  : à gauche, décroissance en échelle logarithmique des fonctionnelles  $J(\mathbf{A})$  générées par les algorithmes **VMFB** et **FISTA** - A droite, **SNR** au cours des itérations pour les algorithmes **VMFB** et **FISTA**.

On se permet d'utiliser une métrique variant au cours des itérations, car les dimensions sont assez petites pour pouvoir stocker, appliquer et inverser la métrique. Elle nous permet par ailleurs d'évaluer les performances que l'on peut obtenir pour l'algorithme **VMFB** en termes de vitesse de convergence, car la métrique  $\mathbf{M}_{nl}^1$  est très proche de la véritable hessienne.

Les résultats en figure 3.11 montrent bien que la décroissance de la fonctionnelle est rapide et que l'algorithme **VMFB** converge au bout de quelques dizaines d'itérations. Le résultat en sortie donne un **SNR** de 195 dB, une reconstruction quasi-parfaite comme on peut le constater en figure 3.12. Ceci indique que la métrique  $\mathbf{M}_{nl}^1$  apporte une information pertinente pour la convergence. D'autre part, l'algorithme **FISTA** converge lui aussi, mais plus lentement en comparaison à l'algorithme **VMFB**, ce qui confirme bien l'utilité d'une métrique.

L'inconvénient d'utiliser la métrique  $\mathbf{M}_{nl}^1$  est qu'elle change à chaque itération, il faut donc calcu-

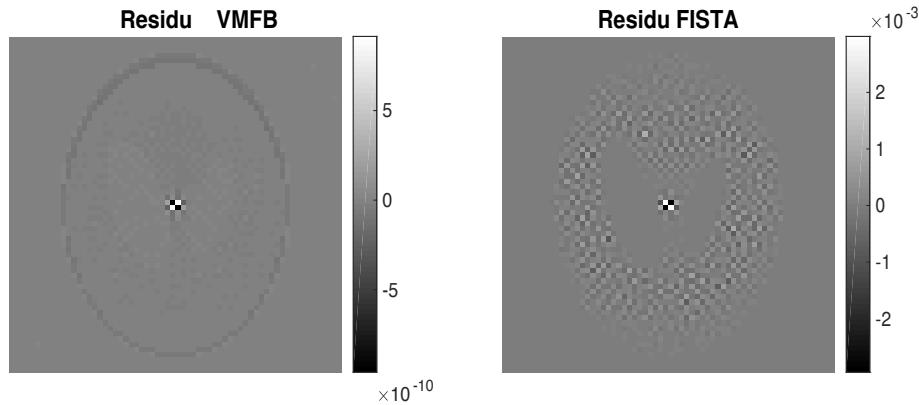


FIGURE 3.12 – Résultats pour  $\mathbf{M}_{nl}^1$  : images résiduelles ( $64 \times 64$  pixels) pour les algorithmes **VMFB** et **FISTA**.

ler son inverse à chaque étape. Nous avons donc testé la métrique  $\mathbf{M}_{nl}^2$ , qui approxime la hessienne  $\mathbf{HG}_{nl}(\mathbf{A})$  et qui ne dépend pas des itérés  $\mathbf{A}_k$ .

La figure 3.13 montre que la décroissance de la fonctionnelle générée par l'algorithme **VMFB** est plus rapide que celle de l'algorithme **FISTA** dans ce cas aussi, mais stagne au bout de quelques dizaines d'itérations pour un résultat **SNR** en sortie de 59 dB. Dans ce cas, cette stagnation est

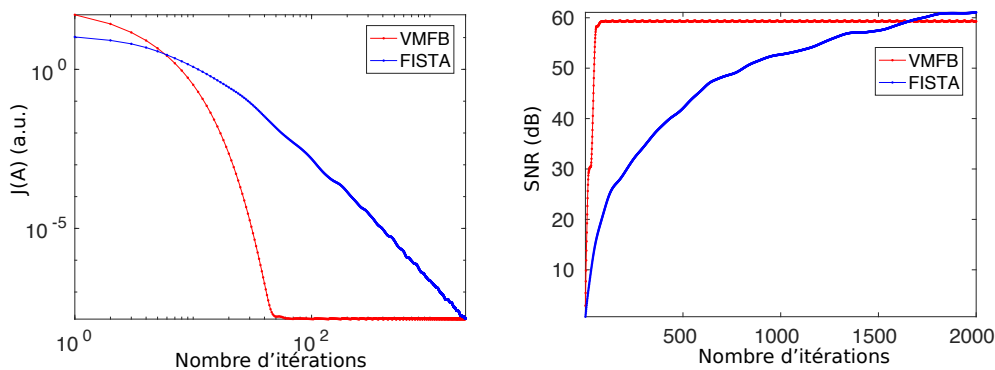


FIGURE 3.13 – Résultats pour  $\mathbf{M}_{nl}^2$  : à gauche, décroissance en échelle logarithmique des fonctionnelles  $J(\mathbf{A})$  générées par les algorithmes **VMFB** et **FISTA** - À droite, **SNR** au cours des itérations pour les algorithmes **VMFB** et **FISTA**.

due à la métrique, mais elle peut être aussi causée par un mauvais paramétrage de l'algorithme du point proximal. Il est important de noter que le point proximal influe sur la convergence globale. Lorsque l'on change la métrique de l'algorithme **VMFB**, il faut veiller à adapter le pas de descente du point proximal. En effet, comme la fonction objectif est non-convexe, on peut converger vers une solution différente selon les points proximaux intermédiaires et approximatifs que l'on cal-

cule. La figure 3.14 montre tout de même que le résidu de la reconstruction obtenue avec l'algorithme **VMFB** présente moins de granularité et a une dynamique similaire à celle de l'algorithme **FISTA**.

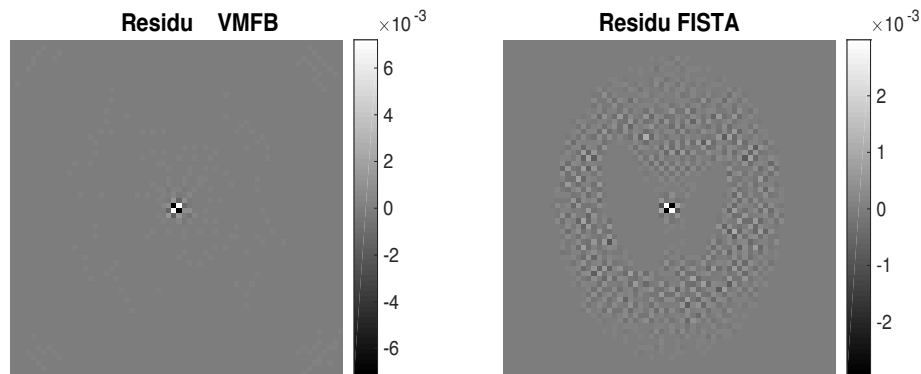


FIGURE 3.14 – Résultats pour  $M_{nl}^2$  : images résiduelles ( $64 \times 64$  pixels) pour les algorithmes **VMFB** et **FISTA**.

### Conclusion :

Les résultats obtenus avec une métrique confirment bien la théorie. Les métriques permettent une nette accélération de la convergence et permettent aussi d'avoir de bonnes reconstructions en peu d'itérations pour le cas non-linéaire sans bruit. Dans la suite nous allons voir les résultats que l'on obtient sur un jeu de données bruitées.

#### 3.2.1.5 Cas non-linéaire avec bruit

Les résultats présentés ici correspondent au cas où les données ont un bruit blanc gaussien d'un niveau choisi de manière à avoir un **SNR** de données en entrée à 30 dB. Nous avons conservé les mêmes paramètres que pour le cas sans bruit. La figure 3.15 illustre une fois de plus la rapidité

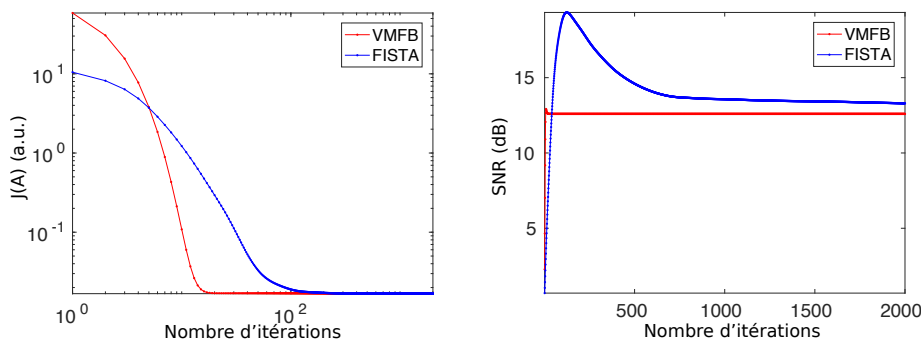


FIGURE 3.15 – Résultats pour  $M_{nl}^1$  : à gauche, décroissance en échelle logarithmique des fonctionnelles  $J(A)$  générées par les algorithmes **VMFB** et **FISTA** - A droite, **SNR** au cours des itérations pour les algorithmes **VMFB** et **FISTA**.

de convergence de l'algorithme **VMFB**. En effet, la fonctionnelle que génère l'algorithme décroît plus vite que celle de l'algorithme **FISTA**. Le **SNR** obtenu en sortie stagne rapidement pour l'algorithme **VMFB** à un niveau de 12,5 dB, mais il reste inférieur à celui de l'algorithme **FISTA**. Ce dernier continue de décroître et on peut penser qu'il finira par atteindre le même niveau de **SNR** que celui obtenu par l'algorithme **VMFB**, mais bien plus tard dans les itérations. Par ailleurs, lorsqu'on utilise la majorante  $M_{nl}^2$ , la fonctionnelle de l'algorithme **VMFB** décroît légèrement moins vite, mais reste toujours plus rapide que celle de l'algorithme **FISTA** comme on le voit en figure 3.17. Cependant, les deux **SNR** évoluent de manière similaire et il en est de même pour les images résiduelles en figures 3.16 et 3.18).

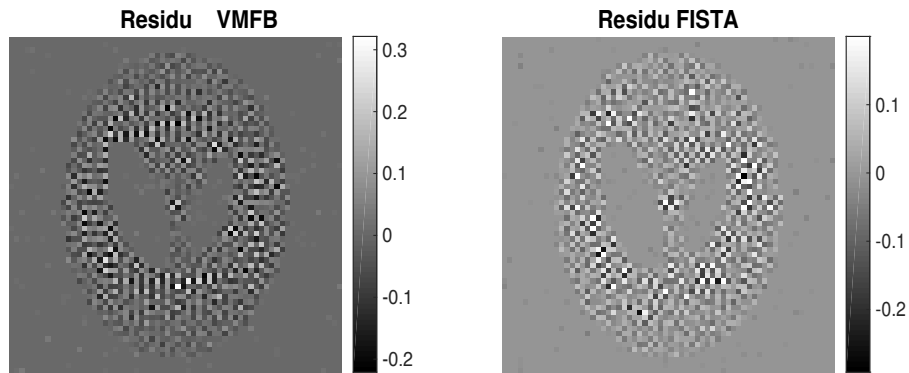


FIGURE 3.16 – Résultats pour  $M_{nl}^1$  : images résiduelles ( $64 \times 64$  pixels) pour les algorithmes **VMFB** et **FISTA**.

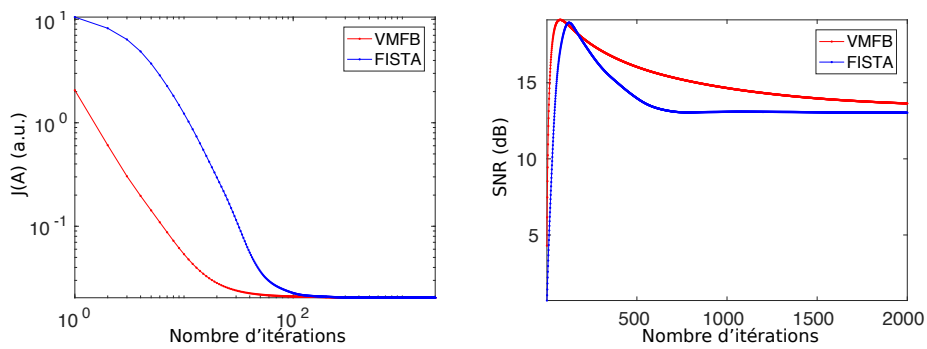


FIGURE 3.17 – Résultats pour  $M_{nl}^2$  : à gauche, décroissance en échelle logarithmique des fonctionnelles  $J(A)$  générées par les algorithmes **VMFB** et **FISTA** - A droite, **SNR** au cours des itérations pour les algorithmes **VMFB** et **FISTA**.

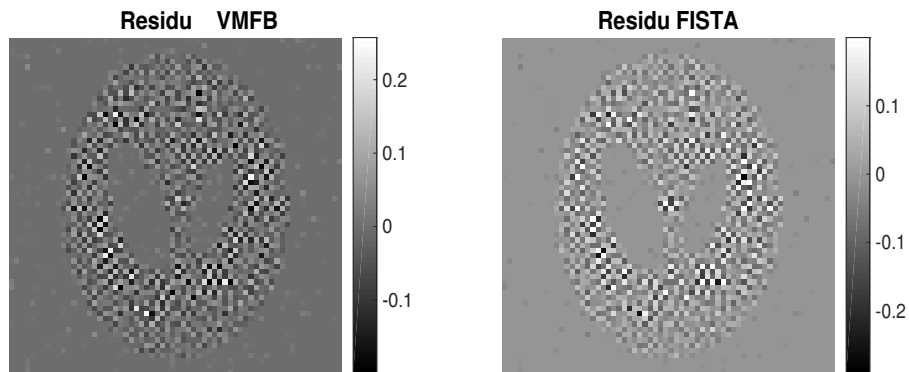


FIGURE 3.18 – Résultats pour  $M_{nl}^2$  : images résiduelles ( $64 \times 64$  pixels) pour les algorithmes **VMFB** et **FISTA**.



### 3.2.1.6 Conclusion

Les tests que nous avons menés montrent bien l'efficacité d'une descente de gradient proximal sur un problème non-linéaire de **TDM** d'absorption classique. D'autre part nous avons montré que lorsqu'une métrique est bien choisie elle permet à l'algorithme **VMFB** de converger en moins d'une centaine d'itérations et il demeure plus rapide que l'algorithme **FISTA** et ceci pour les deux métriques  $M_{nl}^1$  et  $M_{nl}^2$ .

Nous avons donc montré qu'un algorithme proximal peut résoudre un problème non-linéaire proche du spectral. En effet, l'algorithme **VMFB** s'avère être efficace si la métrique est assez pertinente et répond aux conditions citées en première en section 2.4.3 du chapitre 2. Dans la suite, nous allons tester nos méthodes pour résoudre le problème de **TDM** spectrale sur des données à deux dimensions.

## 3.2.2 Tests sur données polychromatiques 2D

Dans la première sous-section, nous avons démontré que les performances des algorithmes que l'on utilise correspondent à la théorie et sont plus élevées que celles des méthodes standards de l'état de l'art. Nous allons ici tester nos algorithmes sur des données polychromatiques en deux dimensions. L'objectif est de voir quel genre de résultats l'on peut obtenir en petite dimension et comment l'on peut étendre cela au cas tridimensionnel.

### 3.2.2.1 Cadre de test

Nous simulons numériquement un fantôme de contraste (figure 3.19) contenant une coupe d'un grand cylindre d'eau comprenant six tubes remplis d'agents de contraste répartis comme suit :

- trois tubes contiennent de l'yttrium en solution à différentes concentrations,
- deux contiennent de l'argent en solution à différentes concentrations,
- le dernier contient de l'iode en solution à une certaine concentration.

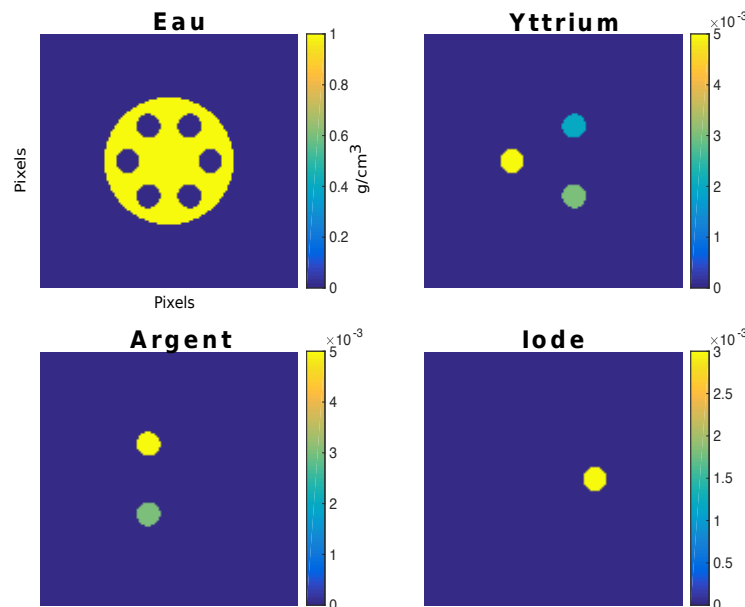


FIGURE 3.19 – Image originale ( $128 \times 128$  pixels) : répartition des composants au sein des tubes.

Remarquons dans un premier temps que l'opérateur de projection et rétroprojection  $S^t S$  agit comme un filtre passe-bas dont le gabarit fréquentiel est l'inverse du filtre **rampe**. Théoriquement,

ce filtre est à réponse impulsionnelle infinie, et pour effectuer une inversion parfaite il faut un support infini de l'image, ce qui est impossible en pratique. L'application du filtre sur des images avec un support de petite taille peut donc entraîner une perte d'information.

Une solution pour palier ce problème est d'augmenter la taille des images de travail par la méthode du *zero-padding* [GASQUET et WITOMSKI, 2000]. Cette méthode consiste à **compléter par des zéros** les bords de l'image afin d'augmenter le nombre de points d'analyse, le nombre de points de signal utile restant le même, ce qui ne change donc pas la résolution. Il s'agit en fait d'une interpolation en fréquence qui permet deux choses :

- augmenter la précision de la transformée de Fourier que l'on utilisera souvent pour calculer l'inverse du filtre conique,
- avoir un nombre total de points égal à une puissance de 2 et pouvoir ainsi exploiter les capacités de l'algorithme de **transformation de Fourier rapide** ou **FFT** (*Fast Fourier Transform*) [RAO et collab., 2010].

Notons toutefois que cette augmentation de taille n'entraîne pas une augmentation physique de la taille des pixels de l'image, et donc que 180 projections sont suffisantes dans ce cas.

#### Protocole 3.2.2.1: test sur le fantôme de contraste avec 4 composants

**Objet :** fantôme de contraste contenant : eau, yttrium, argent et iode (figure 3.19).

**Taille :**  $D = 128 \times 128$  pixels.

**Nombre de projections (angles) :** 180.

**Nombre de composants :**  $K = 4$ .

**Nombre d'acquisitions :**  $M = 5$ .

**Nombre d'échantillons en énergie :**  $N = 43$ .

**Nombre total de mesures :**  $P = 128 \times 180 = 2.3 \times 10^4$ . Illustration du sinogramme en figure 3.22.

**Source  $I_0(E)$  :** la géométrie est parallèle et l'énergie de la source est constante sur la bande d'énergie [13, 40] keV.

**Efficacité  $D(E)$  :** elle est constante pour tous les pixels du détecteur.

**Réponse du filtre  $F$  :** filtres passe-bande idéaux (voir figure 3.20) autour des discontinuités des couches  $K$  spécifiques aux agents de contraste présents :

$$\begin{aligned} F_1(E) &= \frac{1}{4} \mathbb{1}_{[13,17]}(E), \\ F_2(E) &= \frac{1}{4} \mathbb{1}_{[17,21]}(E), \\ F_3(E) &= \frac{1}{4.5} \mathbb{1}_{[21,25.5]}(E), \\ F_4(E) &= \frac{1}{7.7} \mathbb{1}_{[25.5,33.2]}(E), \\ F_5(E) &= \frac{1}{6.8} \mathbb{1}_{[33.2,40]}(E). \end{aligned}$$

**Dictionnaire des sections efficaces  $\Sigma$  :** sections efficaces d'interactions des composants de l'objet (illustration en figure 3.21).

Nous allons dans la suite présenter les résultats obtenus avec l'algorithme à région de confiance puis appliquer les algorithmes proximaux.

#### 3.2.2.2 Résultats obtenus avec l'algorithme à région de confiance

On souhaite résoudre ici le problème de **TDM** spectrale du cas 2D avec l'algorithme à région de confiance que l'on a présenté en sous-section 2.4.2 du chapitre 2. Le problème s'exprime comme suit :

$$\arg \min_{\mathbf{A} \in \mathbb{R}^{D \times K}} \{J(\mathbf{A}) = G_1(\mathbf{A}) + R(\mathbf{A})\},$$

où  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A})$  représente la contrainte de positivité ou bien  $R(\mathbf{A}) = 0$ .

Les tests menés avec les algorithmes à région de confiance avaient pour but de montrer deux choses :

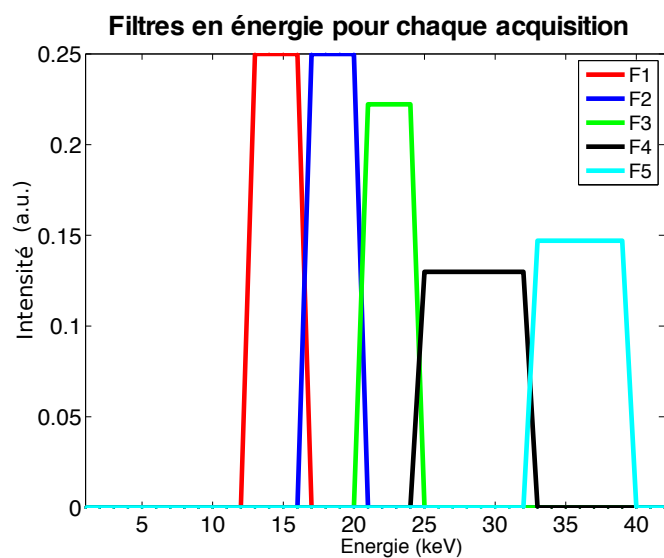


FIGURE 3.20 – Illustration des filtres passe-bande en énergie.

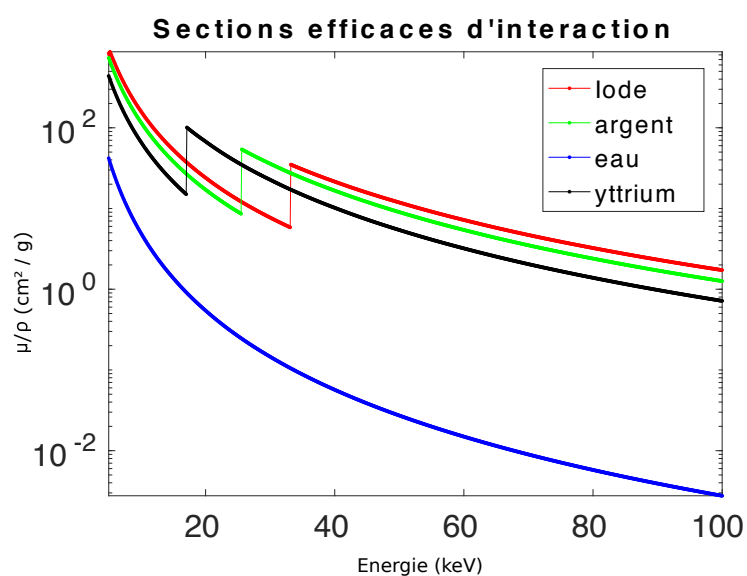


FIGURE 3.21 – Évolution des coefficients d'atténuation linéique atomiques de l'iode, l'eau, l'argent et l'yttrium.

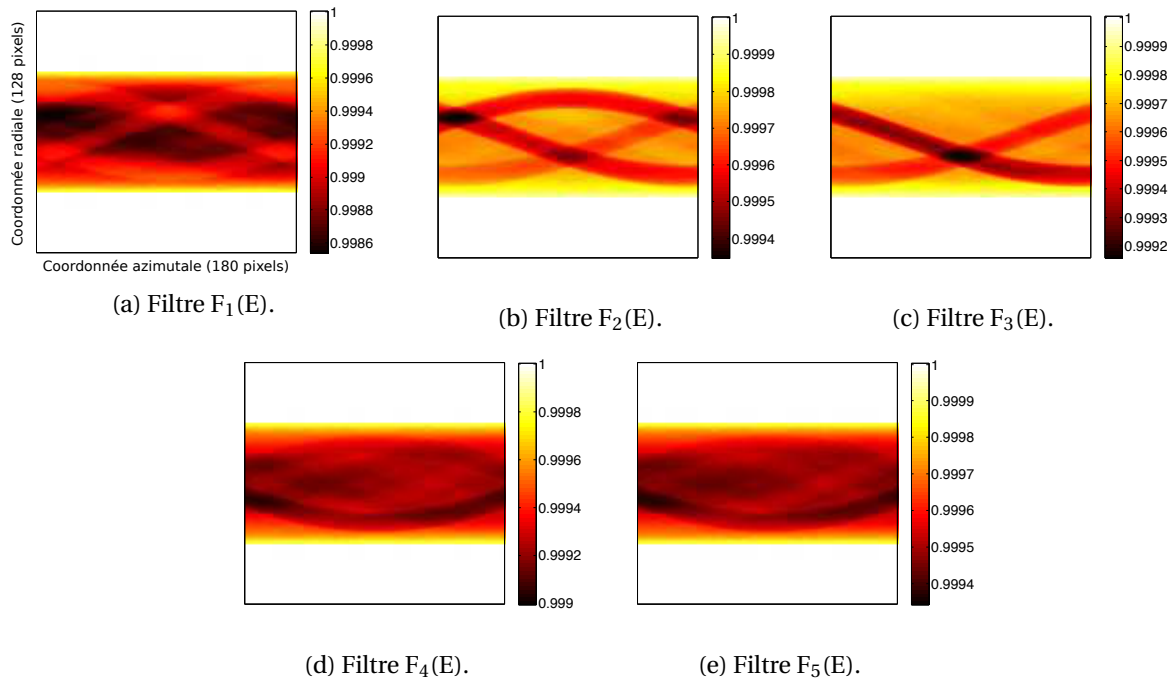


FIGURE 3.22 – Sinogrammes obtenus pour les différents filtres.

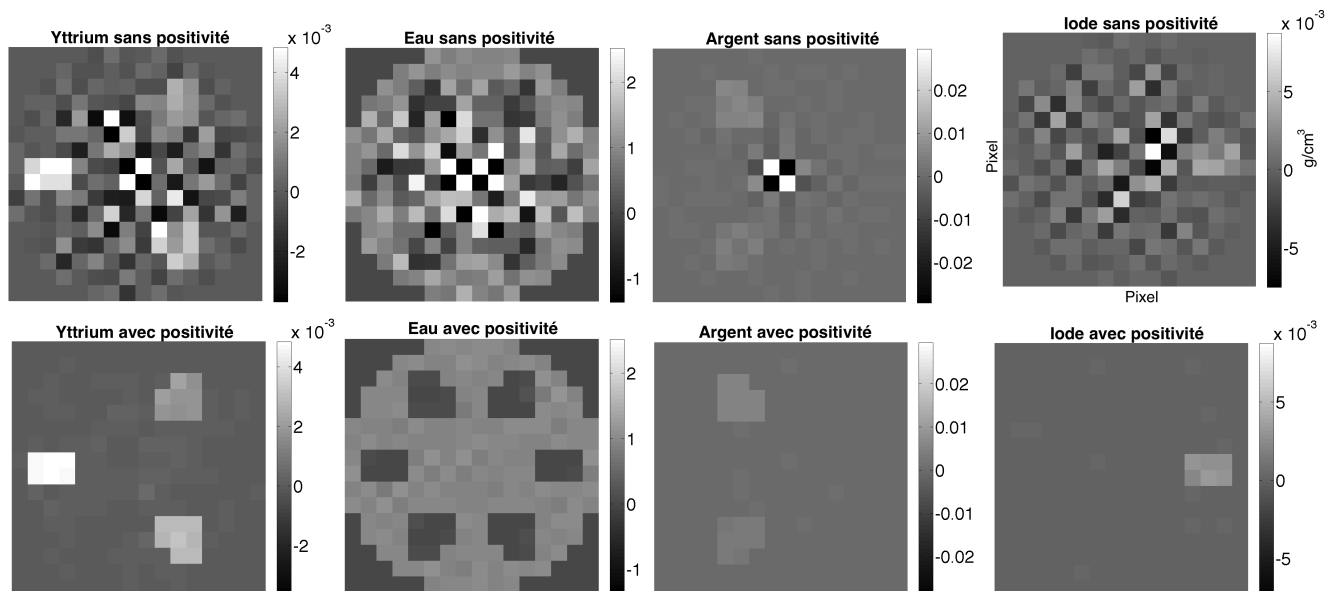


FIGURE 3.23 – Reconstitutions ( $16 \times 16$  pixels) de l'yttrium, de l'eau et de l'argent obtenues avec contrainte (figures en bas) et sans contrainte de positivité (figures en haut) sur un jeu de données avec un SNR en entrée de 20 dB.

- évaluer l'efficacité d'une descente de second ordre,
- évaluer la pertinence d'une régularisation.

La régularisation utilisée ici est seulement la positivité, car on ne peut pas incorporer d'autres types de contraintes avec ce genre d'algorithmes (voir sous-section 2.4.2 du chapitre 2). Nous avons délibérément réduit la taille de l'objet à  $16 \times 16$  pixels, car au delà de cette taille, le temps de calcul devenait beaucoup trop long.

Les paramètres de l'algorithme sont les suivants :

- **initialisation** : proche de la solution, *i.e.* la vérité de terrain bruitée,
- **norme maximale de la région** :  $\Delta = 100$ ,
- **critère d'arrêt** : nombre d'itérations égal à 150.

La version de l'algorithme que nous avons utilisée est celle employant la méthode de la résolution exacte discutée au paragraphe 2.4.2.1 du chapitre 2.

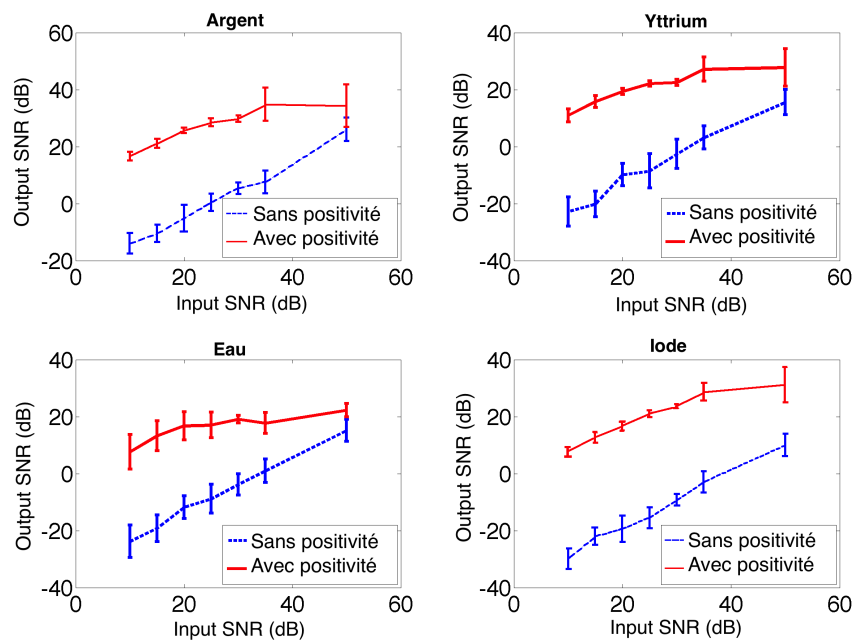


FIGURE 3.24 – Robustesse au bruit : évolution du **SNR** de la reconstruction en fonction du **SNR** des données pour l'argent, l'yttrium, l'eau et l'iode. Les courbes sont des moyennes sur 10 réalisations de bruit différentes pour chaque entrée.

Une étude de bruit a été menée pour évaluer la robustesse de la régularisation au bruit. Pour cela on lance l'algorithme sur un jeu de données bruitées avec dix réalisations différentes de bruit gaussien. La figure 3.23 montre les résultats des reconstructions obtenues pour deux cas : celui où on incorpore la contrainte de positivité et celui sans contrainte.

On remarque dans un premier temps que pour le cas où la contrainte de positivité est utilisée, l'algorithme reconstruit bien les cartes de composants à partir de mesures polychromatiques non-linéaires. En effet, on peut distinguer les 4 composants de la matrice **A**.

D'autre part, la contrainte de positivité permet un résultat avec un meilleur rapport signal sur bruit comme le montre la figure 3.24. Cette figure illustre l'évolution du **SNR** des données reconstruites en fonction de celui des données en entrée. L'évolution de ces courbes est linéaire pour les 4 composants, mais les résultats obtenus sont meilleurs lorsqu'on utilise une régularisation.

**Conclusion :**

Ce test montre bien que la séparation est possible à partir des données spectrales avec un al-

gorithme à région de confiance. De plus, l'incorporation de la contrainte de positivité est déterminante pour la qualité de la reconstruction. Mais comme nous l'avons mentionné à la sous-section 2.4.2 du chapitre 2, l'algorithme à région de confiance ne peut incorporer d'autres types de contraintes. Par ailleurs, la résolution qu'on utilise fait appel à la matrice hessienne à chaque itération, ce qui est très coûteux en temps de calcul et nous empêche de travailler sur de plus grandes dimensions. Dans la prochaine sous-section, nous verrons les résultats obtenus avec les algorithmes proximaux sur une plus grande taille et avec la régularisation  $l_1$ .

### 3.2.2.3 Résultats obtenus avec FISTA et VMFB

Dans la suite, nous comparerons les résultats de l'algorithme **VMFB** à ceux de l'algorithme **FISTA**. Pour ces tests, nous avons augmenté la taille du fantôme à  $128 \times 128$  pixels, car les algorithmes proximaux permettent de traiter les données en grande dimension. Les paramètres pour l'algorithme **FISTA** sont les suivants :

- **initialisation** : à zéro,
- **pas de descente** : l'inverse de la constante de majoration suivante :

$$L_{2D} = 2 \|\Sigma \mathbf{F} \mathbf{F}^t \Sigma^t\|_{L_{Op}}, \quad (3.39)$$

où  $L_{Op}$  est la norme de l'opérateur  $\mathbf{S}^t \mathbf{S}$  calculée d'après l'équation (3.29),

- **critère d'arrêt** : nombre d'itérations égal à 200.

Les paramètres pour l'algorithme **VMFB** sont les suivants :

- **initialisation** : à zéro,
- **métrique** : on utilise la métrique  $\mathbf{M}_1$ ,
- **pas de descente** : on multiplie seulement le gradient par le facteur correcteur  $\hat{\alpha} = 1.2$  calculé d'après l'équation (3.30),
- **paramètre de relaxation** : on fixe  $\lambda = 1$ , *i.e.* pas de relaxation.
- **critère d'arrêt** : nombre d'itérations égal à 200.

Nous avons incorporé deux régularisations dans ces tests : l'indicatrice des positifs et la norme  $l_1$ . On rappelle que ces contraintes s'écrivent de la manière suivante :

$$R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A}) + \sum_{k=1}^K \lambda_k \|\mathbf{A}_k\|_1.$$

Le choix des paramètres  $\lambda_k$  pour la régularisation a été déterminé de manière empirique suivant les valeurs moyennes des dynamiques des sinogrammes et selon l'amplitude du bruit :

$$\begin{aligned} \lambda_{\text{eau}} &= 1 \times 10^{-6.8}, \\ \lambda_{\text{yttrium}} &= 2.513 \times 10^{-6.8}, \\ \lambda_{\text{iode}} &= 0.086 \times 10^{-6.8}, \\ \lambda_{\text{argent}} &= 1.50 \times 10^{-6.8}. \end{aligned}$$

Le calcul du point proximal se fait à l'aide de l'algorithme **FISTA** en prenant pour paramètres :

- **initialisation** : à l'itéré  $\mathbf{A}_k$ ,
- **pas de descente** : l'inverse la norme de la métrique  $\mathbf{M}_1$  comme pas de descente du gradient,
- **projection** : l'opérateur de seuillage doux tel que défini en équation (2.35),
- **critère d'arrêt** : nombre d'itérations égal à 100.

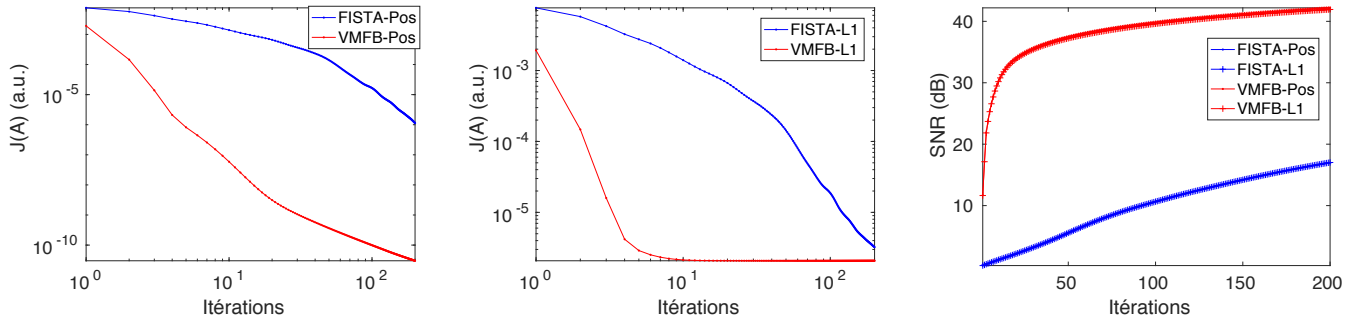


FIGURE 3.25 – Gauche : décroissance de la fonctionnelle  $J(\mathbf{A})$  avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A})$ . Centre : décroissance de la fonctionnelle  $J(\mathbf{A})$  avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A}) + \sum_{k=1}^K \lambda_k \|\mathbf{A}_k\|_1$ . Droite : évolution du **SNR** global sur les 4 composantes en fonction des itérations. Résultats obtenus sur des données non bruitées.

### Résultats dans un cas sans bruit

Dans un premier temps, nous lançons les algorithmes sur des données non bruitées. On peut observer en figure 3.25 que dans ce cas de figure aussi, le schéma proximal à métrique variable avec la métrique  $\mathbf{M}_1$  permet une convergence significativement plus rapide qu'un schéma **FISTA** et ceci indépendamment de la régularisation choisie. Cela confirme bien les premiers tests menés avec des données de **TDM** d'absorption classique dans la précédente sous-section.

Cependant, on remarque que les deux régularisations n'influent pas sur les performances comme

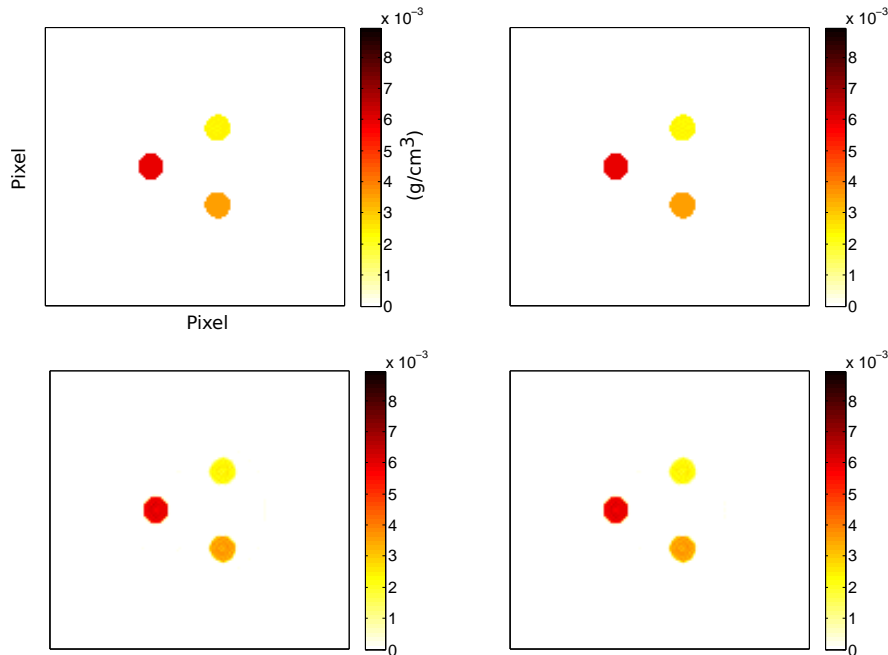


FIGURE 3.26 – Reconstructions ( $128 \times 128$ ) du composant yttrium générées par **VMFB** (en haut) et **FISTA** (en bas) : à gauche, avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A})$  — à droite, avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A}) + \sum_{k=1}^K \lambda_k \|\mathbf{A}_k\|_1$ . Résultats obtenus sur des données non bruitées.

on le constate avec le **SNR**. D'autre part, la régularisation  $l_1$  lisse un peu plus l'image et il y a moins de granularité comme on peut le constater en figure 3.26 avec le résidu de la reconstruction **FISTA**. Mais la dynamique la plus faible pour le résidu est obtenue en utilisant la positivité avec l'algorithme **VMFB**.

Résultats dans un cas avec bruit

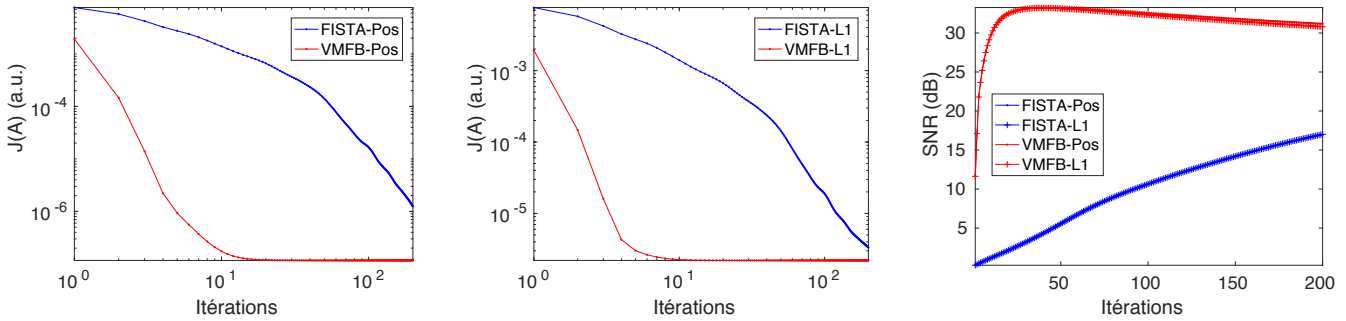


FIGURE 3.27 – Gauche : décroissance de la fonctionnelle  $J(\mathbf{A})$  avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A})$ . Centre : décroissance de la fonctionnelle  $J(\mathbf{A})$  avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A}) + \sum_{k=1}^K \lambda_k \|\mathbf{A}_k\|_1$ . Droite : évolution du SNR global sur les 4 composantes en fonction des itérations. Résultats obtenus sur des données bruitées.

L'expérience est faite avec un bruit de variance  $10^{-6}$  pour des données d'amplitudes comprises dans l'intervalle  $[10^{-3}, 1]$ . Dans ce cas de figure aussi, la convergence de l'algorithme **VMFB** est plus rapide que celle l'algorithme **FISTA** comme on le voit en figure 3.27. Comme pour le cas sans bruit, le type de régularisation ne semble pas influencer les performances de reconstruction.

Par ailleurs, la dynamique des images résiduelles est sensiblement la même pour chaque algorithme, on observe néanmoins que l'algorithme **VMFB** permet d'obtenir une meilleure reconstruction, car les inserts d'agent de contraste sont bien reconstruits, tandis que pour l'algorithme **FISTA** on voit sur les images en figure 3.28 que le centre des inserts n'est pas totalement reconstruit.

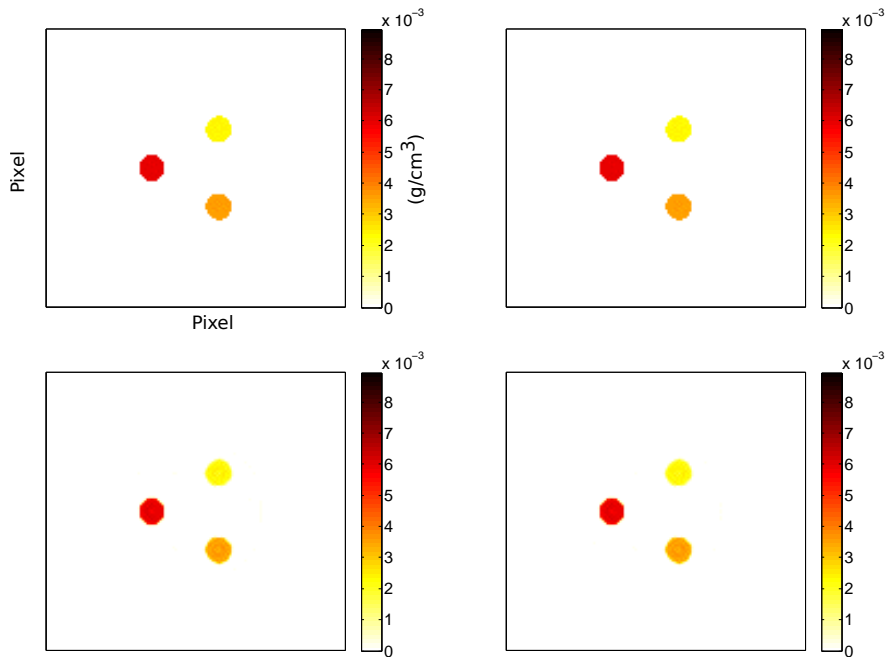


FIGURE 3.28 – Reconstructions ( $128 \times 128$ ) du composant yttrium générées par l'algorithme **VMFB** (en haut) et l'algorithme **FISTA** (en bas) : à gauche, avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A})$  — à droite, avec  $R(\mathbf{A}) = \chi_{\{\geq 0\}}(\mathbf{A}) + \sum_{k=1}^K \lambda_k \|\mathbf{A}_k\|_1$ . Résultats obtenus sur des données bruitées.



**Conclusion :**

Ces tests montrent que les performances des algorithmes restent les mêmes que les données soient bruitées ou non. En effet, l’algorithme **VMFB** converge beaucoup plus vite que l’algorithme **FISTA** et ceci quelle que soit la régularisation choisie. Notons toutefois que même si l’algorithme **FISTA** n’est pas un algorithme destiné à résoudre un problème non-convexe tel que le nôtre, il converge vers un minimum local dans les tests effectués.

Les fonctions de régularisation utilisées dans nos tests sont l’indicatrice des positifs et la norme  $l_1$ . Elles nous donnent des résultats sensiblement similaires dans le cas des données bruitées et non bruitées. D’autre part, choisir des hyperparamètres de régularisation  $\lambda_k$  pour la norme  $l_1$  s’est avéré être une tâche difficile, car les algorithmes étaient sensibles même aux plus petites variations de ces paramètres, ce qui rend les résultats de reconstruction mauvais en termes de **SNR** et de séparation.

Ces tests ont surtout permis de montrer que nos algorithmes peuvent résoudre le problème de **TDM** spectrale en 2D. Dans la prochaine sous-section, nous allons voir les résultats obtenus pour des données de **TDM** spectrale en 3D proches de la réalité.

**3.2.3 Tests sur donnée polychromatique 3D**

On présente dans cette sous-section les résultats obtenus avec des données spectrales simulées en 3 dimensions pour des objets de grandes tailles. Le problème que l’on veut résoudre ici est la minimisation de la fonction objectif suivante :

$$J(\mathbf{A}) = G_2(\mathbf{A}) + R(\mathbf{A}), \tag{3.40}$$

avec  $G_2(\mathbf{A})$  la fonction d’attache aux données pour le cas Poisson et  $R(\mathbf{A})$  la fonction contraignant la positivité.

L’objectif des tests de cette sous-section est de confronter nos algorithmes aux problématiques que l’on peut rencontrer avec des données **TDM** spectrales réelles. Pour nos tests, nous avons

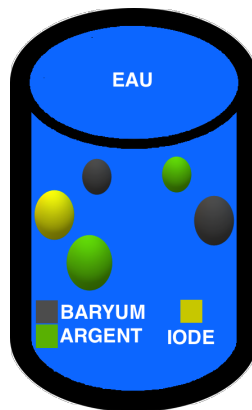


FIGURE 3.29 – Schéma d’un cylindre d’eau contenant différents agents de contraste à différentes concentrations pour le test des données de **TDM** spectrale en 3D.

simulé un fantôme de contraste en 3D contenant des inserts remplis par différents agents de contraste :

- deux inserts de nitrate d’argent ( $\text{AgNO}_3$ ) avec une concentration de 120 mg/mL d’argent,
- deux inserts de sulfate de baryum ( $\text{BaSO}_4$ ) avec une concentration de 80 mg/mL de baryum,
- un insert d’iomeron avec une concentration de 50 mg/mL d’iode.

Nous étudierons ce fantôme sur une hauteur représentant 21 voxels pour alléger les calculs. Cela représente 10 tranches au-dessus et en-dessous du plan médian. Par ailleurs, nous **fixons** les valeurs numérique de l'objet à 1 (présence de signal) ou 0 (absence de signal). Ceci nous permettra d'estimer la qualité de la quantification et de voir à quel point la séparation des composants est précise. Le protocole de test est reporté sur la fiche 3.2.3.1.

#### Protocole 3.2.3.1: test sur le fantôme de contraste en 3D avec 4 composants

**Objet :** fantôme de contraste 3D contenant : eau, iomeron, nitrate d'argent et sulfate de baryum (figure 3.29).

**Taille :**  $D = 128 \times 21 \times 128$  voxels.

**Nombre de projections (angles) :** 360.

**Nombre de composants :**  $K = 4$ .

**Nombre d'acquisitions :**  $M = 4$ .

**Nombre d'échantillons en énergie :**  $N = 36$ .

**Nombre total de mesures :**  $P = 128 \times 21 \times 360 = 9.6 \times 10^5$ . Illustration d'une projection en figure 3.31.

**Source  $I_0(E)$  :** la géométrie des faisceaux est conique (angle d'ouverture à 7 degrés) et le profil de la source provient d'une simulation d'un tube à anode de tungstène sur la bande d'énergie [10,55] keV.

**Distance source-objet et objet-détecteur :** 10 cm et 60 cm.

**Efficacité  $D(E)$  :** elle est égale à 1 (100%) pour tous les pixels du détecteur.

**Réponse du filtre  $F$  :** filtres passe-bande (voir figure 3.30) autour des discontinuités des couches  $K$  spécifiques aux agents de contraste présents.

**Dictionnaire des sections efficaces  $\Sigma$  :** sections efficaces d'interaction totale des composants de l'objet (illustration en figure 3.30).

Dans la suite, nous comparerons les résultats de l'algorithme **VMFB** à ceux de l'algorithme **FISTA**.

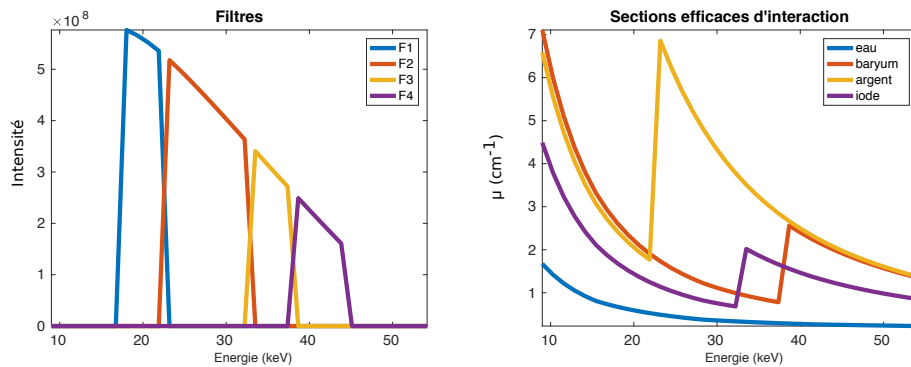


FIGURE 3.30 – À gauche, les filtres passe-bande en énergie – À droite, l'évolution du coefficient d'atténuation linéique pour l'iode, l'eau, l'argent et le baryum.

Les paramètres pour l'algorithme **FISTA** sont les suivants :

- **initialisation :** à zéro,
- **pas de descente :**

$$L_{3D} = 2 \|\Sigma F F^t \Sigma^t\|_{L_{op}}, \quad (3.41)$$

où  $L_{op}$  est l'estimation de la norme de l'opérateur  $S^t S$  calculée d'après l'équation (3.29),

- **critère d'arrêt :** nombre d'itérations égal à 200 (ou 1000 selon les cas).

Les paramètres pour l'algorithme **VMILAn** sont les suivants :

- **initialisation :** à zéro,
- **métrique :** on utilise la métrique  $M_1$  avec le filtre conique apodisé par une fenêtre de **Hamming**,

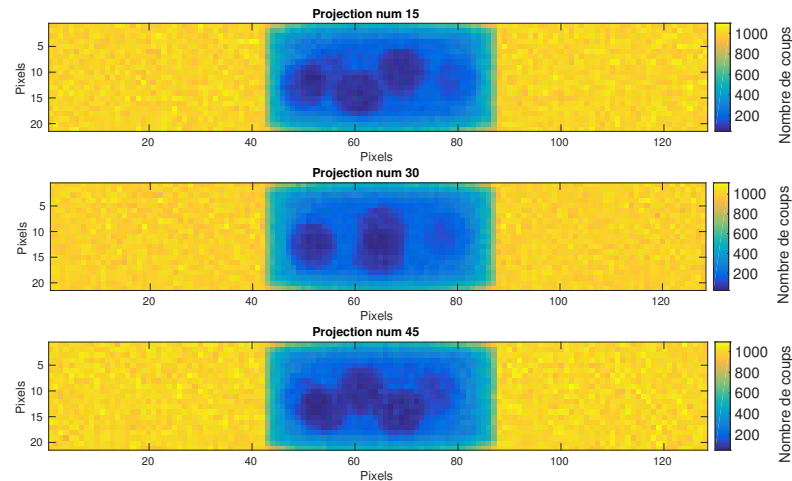


FIGURE 3.31 – Projections ( $128 \times 21$  pixels) numéro 15, 30 et 45 du fantôme de contraste obtenues pour la première acquisition. Acquisitions faites avec 1000 coups dans les blancs (pixels situés en dehors de l’objet projeté).

- **pas de descente** : on multiplie seulement le gradient par le facteur correcteur  $\hat{\alpha} = 1.2$  calculé d’après l’équation (3.30),
- **paramètre de relaxation** : déterminé selon la méthode décrite au paragraphe 2.4.3.5 du chapitre 2, avec un nombre d’essais plus petit que 10,
- **critère d’arrêt** : nombre d’itérations égal à 200 (ou 1000 selon les cas).

Nous n’avons que l’indicatrice des positifs pour ce test. Le calcul du point proximal se fait à l’aide de l’algorithme **FISTA** en prenant pour paramètres :

- **initialisation** : à l’itéré  $\mathbf{A}_k$ ,
- **pas de descente** : l’inverse de la norme de la métrique  $\mathbf{M}_1$  comme pas de descente du gradient,
- **projection** : l’opérateur de projection sur les positifs consistant à mettre les composantes négatives à zero,
- **critère d’arrêt** : nombre d’itérations égal à 100.

Nous allons présenter plusieurs résultats dans les paragraphes suivants. Le critère de qualité de nos résultats repose sur le **SNR** et sur la mesure qualitative des reconstructions. Dans un premier temps, nous allons regarder les résultats obtenus lorsqu’on laisse les algorithmes **FISTA** et **VMI-LAn** tourner sur plusieurs milliers d’itérations. Ceci permettra d’évaluer la stabilité de chaque algorithme et donnera une idée du comportement à long terme de ces derniers. Ce résultat est présenté pour des données sans bruit et avec un bruit de Poisson.

Nous présenterons par la même occasion les résultats obtenus pour différentes tranches de l’objet afin d’illustrer la robustesse des algorithmes au passage d’une géométrie 2D à une géométrie 3D.

Afin d’évaluer la robustesse de nos méthodes par rapport aux perturbations qui se présentent lors d’une acquisition, plusieurs études ont été menées. La première est une étude de bruit qui illustre la qualité des résultats obtenus en fonction du niveau de bruit des données en entrée (le bruit appliqué ici est un bruit de Poisson).

La seconde est une étude de perturbation de la matrice de filtre  $\mathbf{F}$ . En effet, en pratique cette matrice est estimée à partir des données du constructeur et de la connaissance du prototype PIXSCAN-FLI, il est donc important de mesurer la qualité des reconstructions en fonction de l’exactitude de la matrice  $\mathbf{F}$ .

### 3.2.3.1 Résultats obtenus pour des données spectrales 3D non bruitées

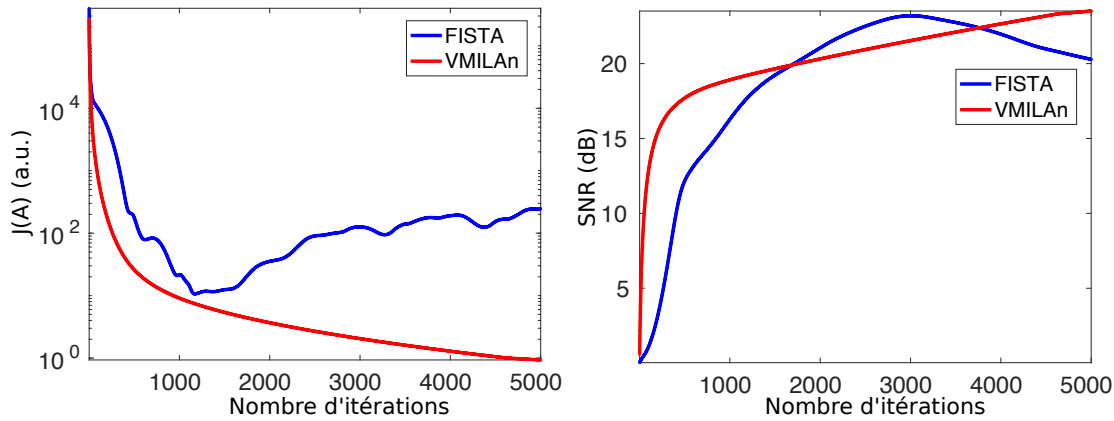


FIGURE 3.32 — À gauche, décroissance des fonctions objectifs en fonction du nombre d'itérations — À droite, évolution du **SNR** en fonction du nombre d'itérations.

Dans un premier temps, nous allons considérer le résultat de reconstruction sur des données parfaites. Dans ce cas de figure, nous laisserons les algorithmes **FISTA** et **VMILAn** faire un très grand nombre d'itérations (5000 itérations) afin d'étudier le comportement asymptotique de ceux-ci. La décroissance de la fonction objectif  $J(\mathbf{A})$  en figure 3.32 illustre bien ce qui est prédit par la théorie. En effet, comme nous l'avons dit en section 2.4 du chapitre 2, l'algorithme **FISTA** ne dispose d'aucune garantie de convergence lorsqu'on l'applique à des problèmes non-convexes, ce qui explique la décroissance de la fonction objectif de l'algorithme **FISTA** (courbe bleue), tandis que la convergence de l'algorithme **VMILAn** est garantie et que sa vitesse est plus grande que celle de l'algorithme **FISTA**, même dans le cas 3D, ce qui confirme l'utilité de la métrique.

Le résultat de la reconstruction et de la séparation est satisfaisant pour les deux algorithmes.

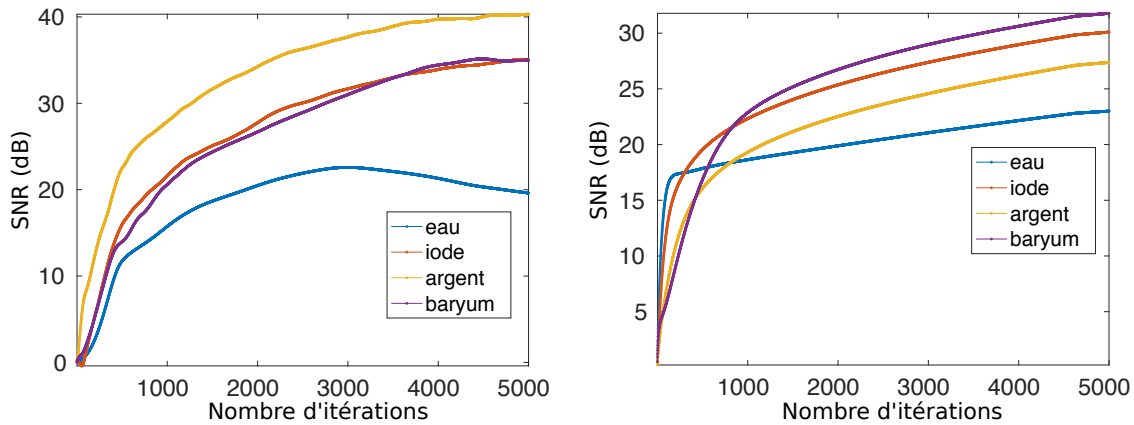


FIGURE 3.33 — Evolution du **SNR** des composants en fonction du nombre d'itérations des algorithmes **FISTA** (à gauche) et **VMILAn** (à droite).

En effet, comme on peut l'observer en figure 3.35, la séparation des 4 composants de l'objet est réussie et il n'y a aucune ambiguïté qui demeure dans les reconstructions. La quantification est très bonne pour les inserts contenant des agents de contraste reconstruits par l'algorithme **FISTA** (particulièrement l'insert contenant l'argent). En effet, on rappelle que les dynamiques dans notre objets sont la présence ou l'absence de signal (1 ou 0), et celle de l'argent est quasi-égale à 1 pour l'algorithme **FISTA** comme on le constate en figure 3.34.

D'autre part, l'algorithme **FISTA** dégrade le centre de l'eau comme on le constate en figure 3.35. Cette dégradation est la conséquence de la non-convergence de l'algorithme **FISTA** pour le problème de **TDM** spectrale.

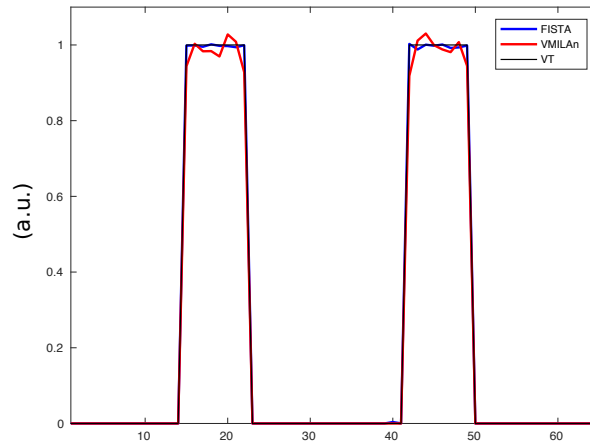


FIGURE 3.34 – Profils (64 pixels) des reconstructions de l’argent au plan médian pour les algorithmes **FISTA** et **VMILAn** comparés au profil de la vérité de terrain (VT sur la figure).

D’autre part, certains voxels des reconstructions obtenues avec l’algorithme **VMILAn** ont une dynamique surestimée comme on le constate en figure 3.35, mais cela reste proche de la réalité. Cette surestimation est due au moiré causé par le sous échantillonnage du filtre conique utilisé dans la métrique. Cet effet peut être atténué en augmentant l’échantillonnage ou en appliquant une fenêtre d’apodisation adaptée, qui atténuera davantage les hautes fréquences (une fenêtre de **Hann** ou de **Blackman** par exemple). De plus, le résultat de reconstruction montre bien que la métrique  $M_1$ , initialement désignée que pour les cas 1D et 2D, peut très bien être utilisée dans le cas 3D et donne des résultats satisfaisants.

La figure 3.33 illustre l’évolution du **SNR** des composants en fonction du nombre d’itérations. Comme nous l’avons mentionné précédemment, l’algorithme **FISTA** reconstruit en priorité les composants des agents de contraste avant de reconstruire l’eau, en commençant par l’argent comme on le voit sur la figure 3.33. Le **SNR** de l’eau se dégrade à partir de 3000 itérations et décroît progressivement, entraînant avec lui la décroissance du **SNR** global car les voxels de l’eau sont présents en grande quantité dans tout l’objet. L’algorithme **VMILAn** reconstruit les 4 composants de manière assez simultanée en mettant l’eau légèrement prioritaire.

En conclusion, les deux algorithmes **FISTA** et **VMILAn** se comportent comme le prédit la théorie et sont tous les deux capables de résoudre le problème de **TDM** spectrale dans le cas 3D. La séparation des composants est correctement exécutée par les deux algorithmes. Par ailleurs, on remarque que la quantification des inserts contenant les agents de contraste est légèrement meilleure pour l’algorithme **FISTA** (en particulier pour l’argent), mais celui-ci dégrade celle de l’eau. Dans le prochain paragraphe, nous allons présenter les résultats d’un test analogue, mais pour des données bruitées.

### 3.2.3.2 Résultats obtenus pour 1000 coups dans les pixels situés en dehors de l’objet projeté

Un test analogue au précédent a été mené, mais cette fois-ci en appliquant un bruit de Poisson aux données. Les données sont acquises dans ce cas pour un taux de comptage égal à 1000 coups dans les pixels situés en dehors de l’objet projeté, ce qui donne un **SNR** des données en entrée égal à 22.2 dB. C’est une statistique de comptage inférieure à celles obtenues pour des acquisitions que l’on peut faire sur le petit animal [PORTAL, 2018], le bruit est donc ici plus élevé. L’objectif est de voir si ce bruit impacte la convergence des algorithmes et la qualité de la reconstruction. Les ré-

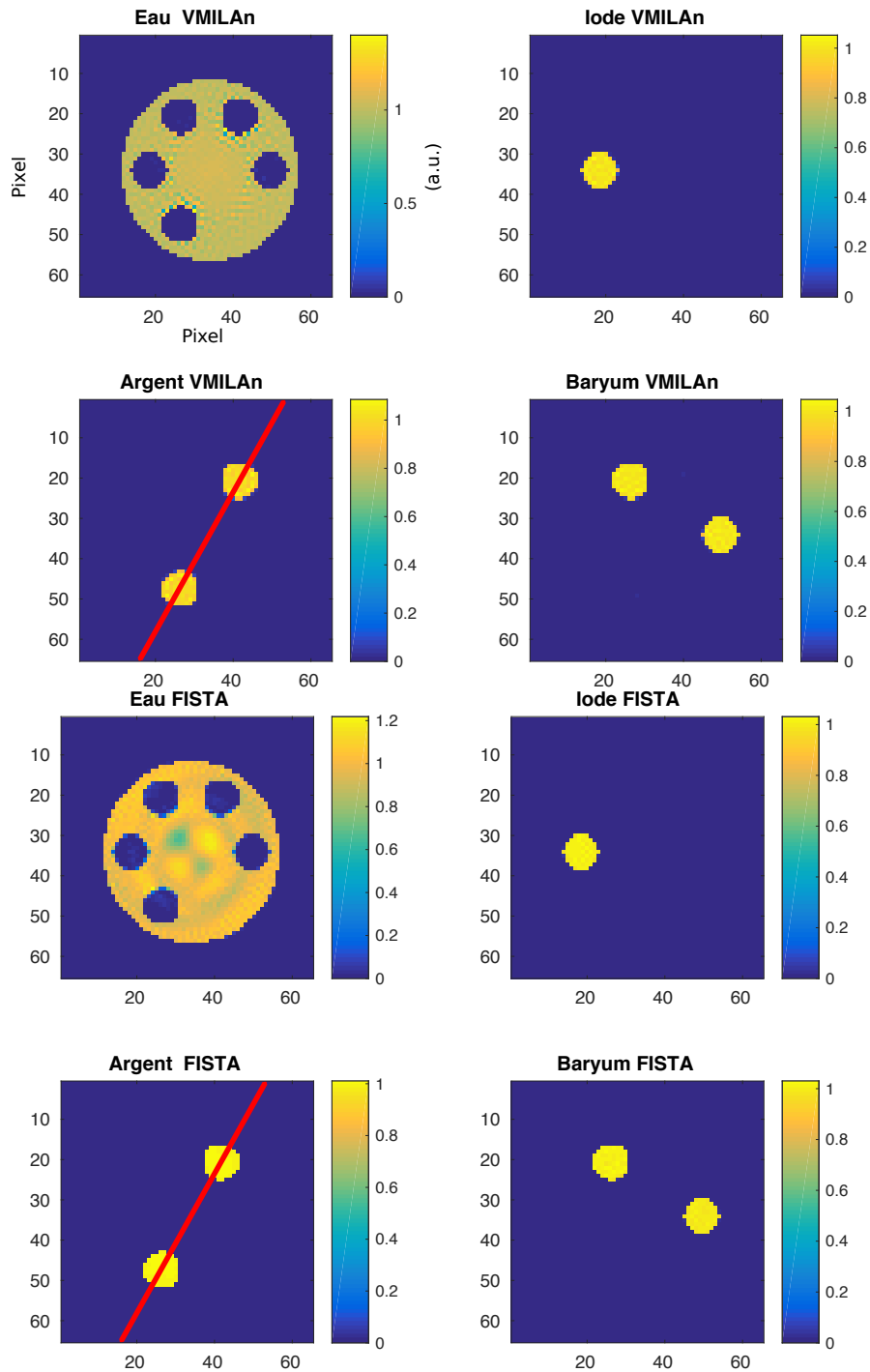


FIGURE 3.35 – Reconstruction du plan médian ( $64 \times 64$  pixels) obtenue avec les algorithmes **VMILAn** et **FISTA** avec des données spectrales 3D non bruitées.

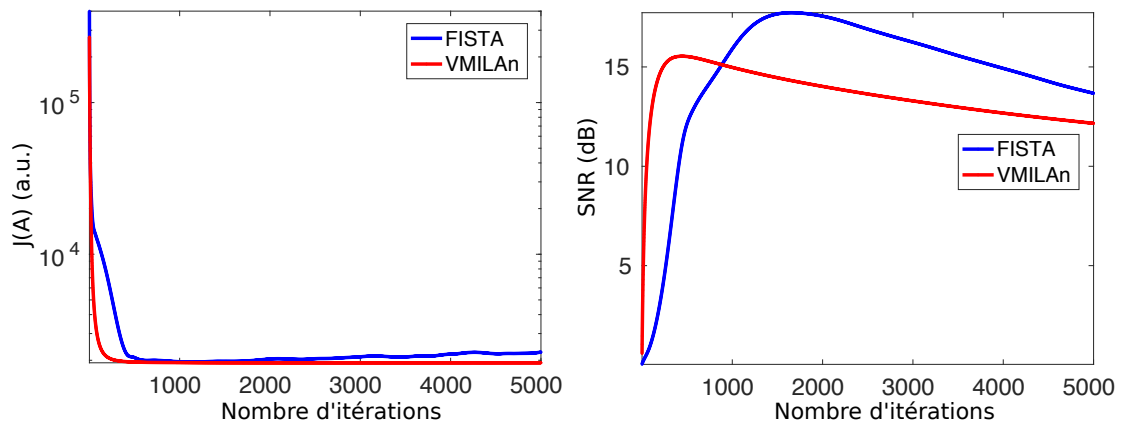


FIGURE 3.36 – A gauche, décroissance des fonctions objectifs en fonction du nombre d’itérations — A droite, évolution du **SNR** en fonction du nombre d’itérations.

sultats de décroissance de la fonction objectif  $J(A)$  et de l’évolution du **SNR** sont reportés en figure 3.36.

On voit là encore que l’incorporation d’une métrique permet à l’algorithme **VMILAn** de converger plus rapidement que l’algorithme **FISTA** et ceci de manière monotone. La qualité de la reconstruction est discutable si l’on regarde le **SNR**. En effet, **VMILAn** arrive à une valeur maximum du **SNR** (15.5 dB) au bout de 300 itérations tandis que l’algorithme **FISTA** arrive à une valeur maximum plus élevée (17 dB), mais plus tard à 1600 itérations.

Le bruit semble impacter un peu plus la qualité du **SNR** de l’algorithme **VMILAn**, mais les reconstructions obtenues sont similaires pour les deux algorithmes comme on peut le constater en figure 3.38. De plus, comme pour le cas sans bruit, l’algorithme **VMILAn** surestime certains voxels à cause du sous-échantillonnage du filtre conique. Notons que la quantification des composants est similaire pour les deux algorithmes sauf pour l’argent qui est mieux quantifié avec l’algorithme **FISTA** comme en témoigne les **SNR** en figure 3.37. Toutefois, le **SNR** de l’argent de l’algorithme **VMILAn** a une tendance croissante tandis que celui de l’algorithme **FISTA** stagne.

On peut conclure que les algorithmes se comportent de manière assez analogue au cas où les données sont non bruitées. L’objectif dans la suite est de quantifier la robustesse au bruit ainsi qu’aux perturbations que les données subissent lors d’une acquisition de **TDM** spectrale.

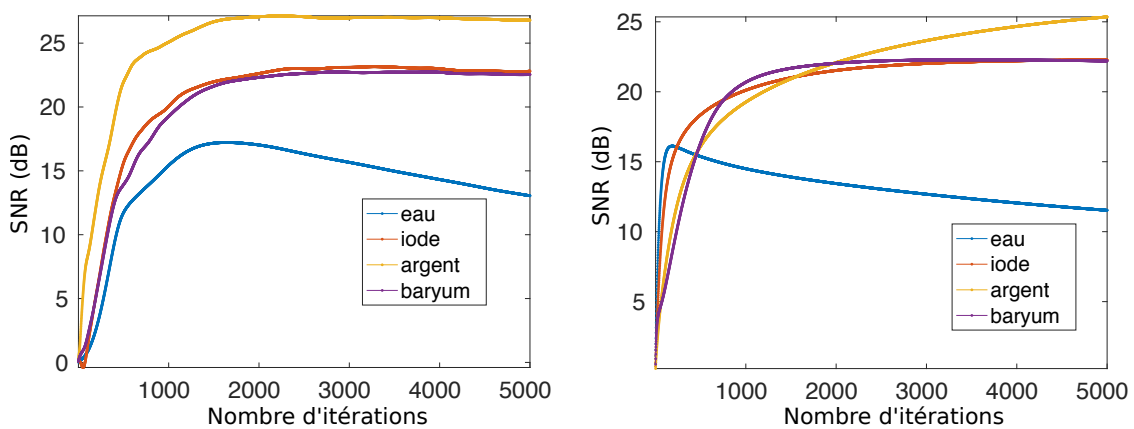


FIGURE 3.37 – Evolution du **SNR** des composants en fonction du nombre d’itérations des algorithmes **FISTA** (à gauche) et **VMILAn** (à droite).

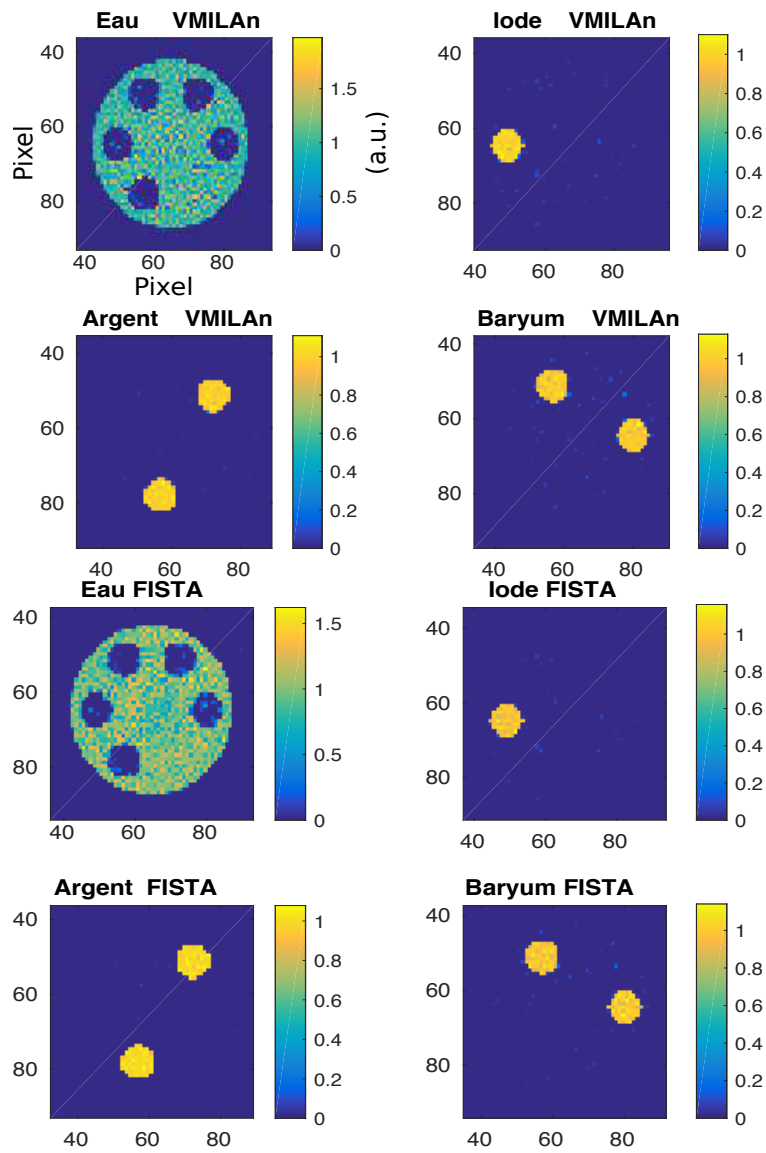


FIGURE 3.38 – Reconstruction du plan médian ( $64 \times 64$  pixels) obtenue avec les algorithmes **VMILAn** et **FISTA** avec des données 3D spectrale pour 1000 coups dans les pixels situés en dehors de l’objet projeté.



### 3.2.3.3 Remarque sur l'effet de l'ouverture du faisceau

Une des préoccupations du passage aux données 3D était la gestion des coefficients de correction pour la reconstruction des tranches hors du plan médian. Comme nous l'avons mentionné en sous-section 3.1.4, nous avons décidé de ne pas intégrer les coefficients de correction dans notre métrique  $\mathbf{M}_1$ . Ceci nous évite de complexifier le calcul de l'inverse de la métrique.

On reporte les résultats des reconstructions du premier plan au dessus du plan médian pour les algorithmes **VMILAn** et **FISTA** en figure 3.39. Ce plan correspond à la reconstruction de la première tranche en haut de la projection du cylindre en figure 3.31. Ce plan est en dehors du plan contenant les inserts d'agent de contraste, il n'y a donc que l'eau qui doit être présent dans les reconstructions.

On remarque alors que les résultats de reconstruction sont très similaires à ceux obtenus dans le plan médian que l'on a obtenu précédemment.

En effet, la reconstruction de l'eau obtenue avec l'algorithme **FISTA** est dégradé au centre, tandis que la reconstruction obtenue avec l'algorithme **VMILAn** est similaire avec la même dynamique que celle obtenue au plan médian. De plus, la dynamique des résidus présents dans les reconstructions des agents de contraste est plus faible pour l'algorithme **VMILAn** que pour l'algorithme **FISTA** comme on le constate en figure 3.39. Ceci montre bien que la métrique  $\mathbf{M}_1$  est suffisante pour que **VMILAn** puisse converger vers une solution acceptable. Nous verrons dans le paragraphe suivant que le résultat reste similaire même lorsque l'angle d'ouverture de la source par rapport au plan médian est élevé.

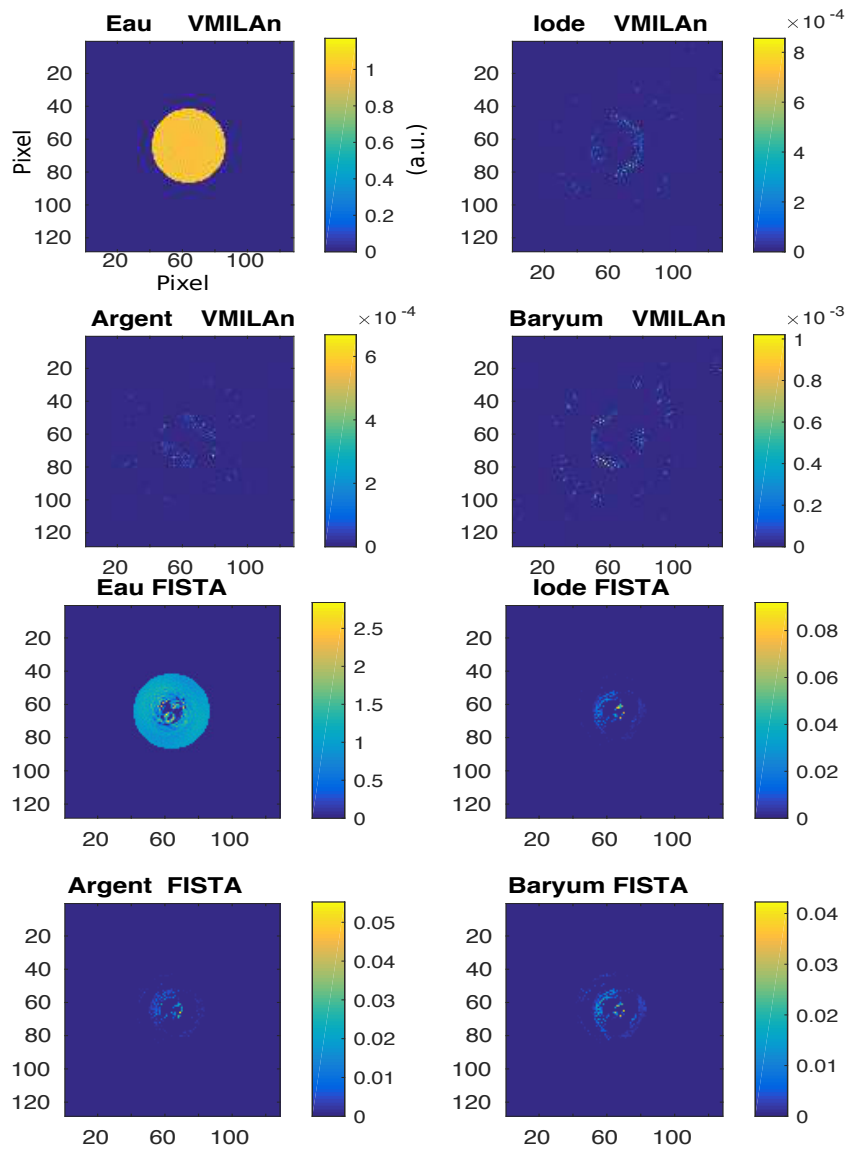


FIGURE 3.39 – Reconstruction de la tranche 1 au dessus du plan médian ( $128 \times 128$  pixels) obtenue avec les algorithmes **VMILAn** et **FISTA** pour des données spectrales 3D non bruitées.

### 3.2.3.4 Cas pour bruit plus élevé et une ouverture de faisceau plus grande

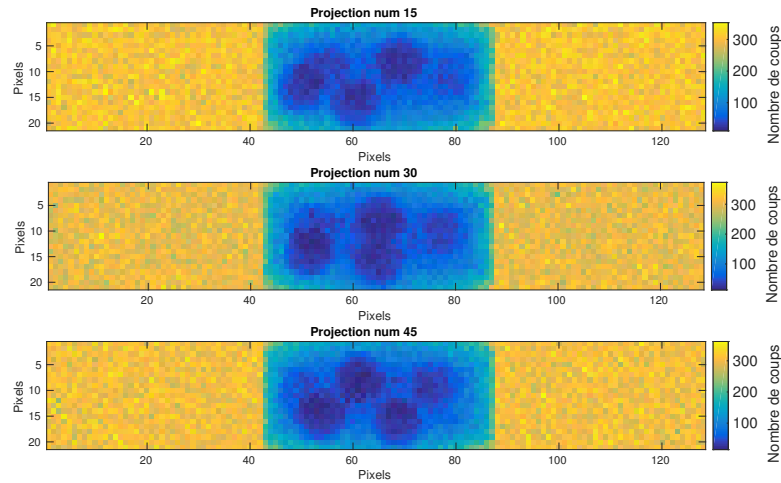


FIGURE 3.40 – Projections ( $128 \times 21$  pixels) numéro 15, 30 et 45 du fantôme de contraste obtenues pour la première acquisition. Acquisitions faites avec 300 coups dans les blancs (pixels situés en dehors de l’objet projeté).

Nous avons voulu tester ici la robustesse de notre méthode face à un bruit plus élevé et une ouverture de faisceau plus grande. La figure 3.40 montre les données obtenues lorsque 300 coups sont comptés dans les blancs (ce qui représente un **SNR** des données en entré de 15 dB) et lorsque l’angle d’ouverture de la source par rapport au plan médian est de 14 degrés, soit le double du cas précédent. Augmenter l’angle d’ouverture correspond en pratique à l’éloignement du détecteur par rapport à l’objet ou le rapprochement de la source par rapport à l’objet.

Nous avons utilisé 1000 itérations pour cette expérience, mais le comportement des fonctions

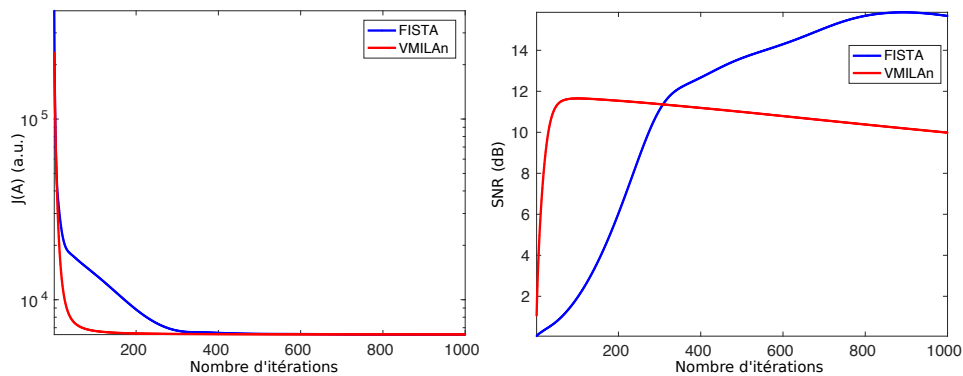


FIGURE 3.41 – A gauche, décroissance des fonctions objectifs en fonction du nombre d’itérations — A droite, évolution du **SNR** en fonction du nombre d’itérations.

objectifs reste le même comme on peut le voir en figure 3.41. En effet, la décroissance de la fonction objectif  $J(\mathbf{A})$  pour l’algorithme **VMILAn** se stabilise rapidement, tandis que celle obtenue avec l’algorithme **FISTA** met plusieurs centaines d’itérations avant de se stabiliser.

Le **SNR** généré par l’algorithme **VMILAn** atteint son maximum (11.8 dB) en moins de 100 itérations, puis décroît au fur et à mesure des itérations et est gouverné par le **SNR** de l’eau comme on le constate en figure 3.42. Le **SNR** généré par l’algorithme **FISTA** atteint son maximum à 900 itérations (16 dB) puis décroît lui aussi gouverné par le **SNR** de l’eau. On peut penser qu’il reproduira le même schéma que celui en figure 3.36.

D'autre part, la séparation obtenue avec les algorithmes **FISTA** et **VMILAn** est sans ambiguïté

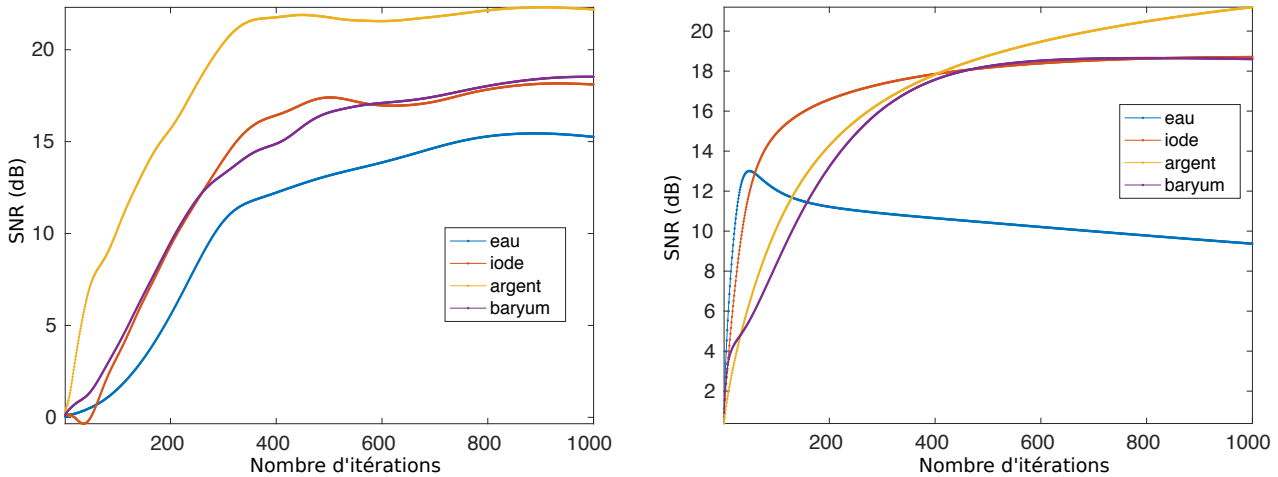


FIGURE 3.42 – Evolution du **SNR** des composants en fonction du nombre d'itérations des algorithmes **FISTA** (à gauche) et **VMILAn** (à droite).

comme on le constate en figure 3.43 et les 4 composants sont bien séparés. Les agents de contraste (iode, argent et baryum) sont assez bien estimés par les deux algorithmes. Mais on remarque tout de même une tendance croissante dans les **SNR** pour l'algorithme **VMILAn**, en particulier pour l'argent comme on le voit en figure 3.42. Remarquons aussi que la dynamique de l'eau est mieux estimée avec l'algorithme **FISTA** au terme des 1000 itérations comme on le constate en figure 3.43.

Par ailleurs, on voit sur la reconstruction de la tranche 1 obtenue avec l'algorithme **VMILAn** en figure 3.44 qu'une ouverture de faisceau plus grande n'impacte pas la reconstruction. Mais la reconstruction obtenue avec l'algorithme **FISTA** a une meilleure dynamique que celle obtenue avec l'algorithme **VMILAn** comme on peut le voir sur les profils en figure 3.45.

On peut conclure que la métrique  $M_1$  que l'on utilise dans l'algorithme **VMILAn** est efficace, même pour les cas où l'ouverture de faisceau est grande. D'autre part, l'algorithme **VMILAn** sépare les composants plus rapidement que l'algorithme **FISTA**, mais ce dernier semble être moins affecté par le bruit et quantifie mieux l'eau. On peut cependant penser que **FISTA** n'est pas encore arrivé à convergence et qu'il finira par dégrader l'eau comme nous l'avons constaté précédemment.

Nous voulons à présent mesurer la qualité de la reconstruction en fonction du niveau de bruit. Pour cela nous allons présenter les résultats d'une étude de bruit sur les deux algorithmes dans le prochain paragraphe.

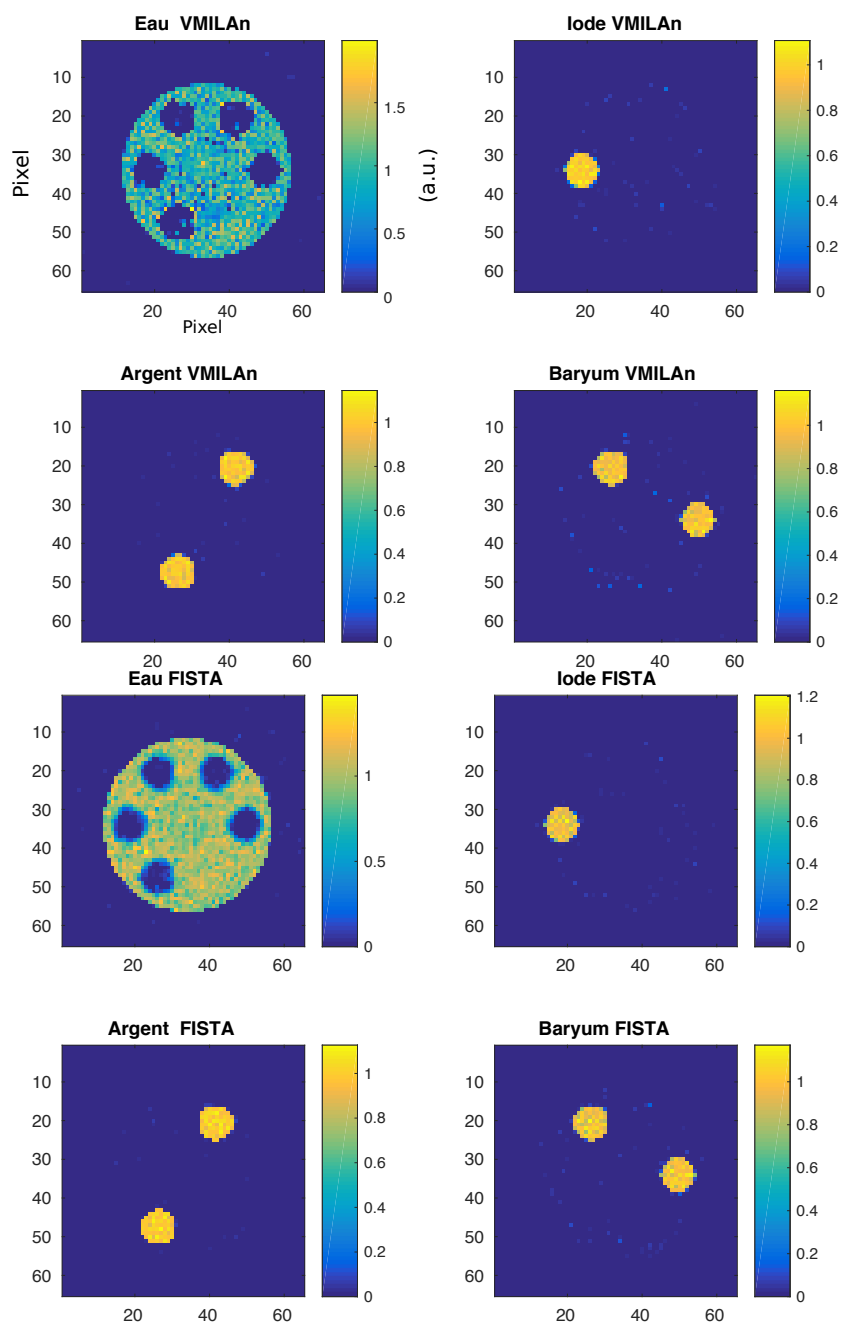


FIGURE 3.43 – Reconstruction du plan médian ( $64 \times 64$  pixels) obtenue avec les algorithmes **VMILAn** et **FISTA** pour données spectrales 3D avec 300 coups dans le blanc et une ouverture de faisceau de 14 degrés.

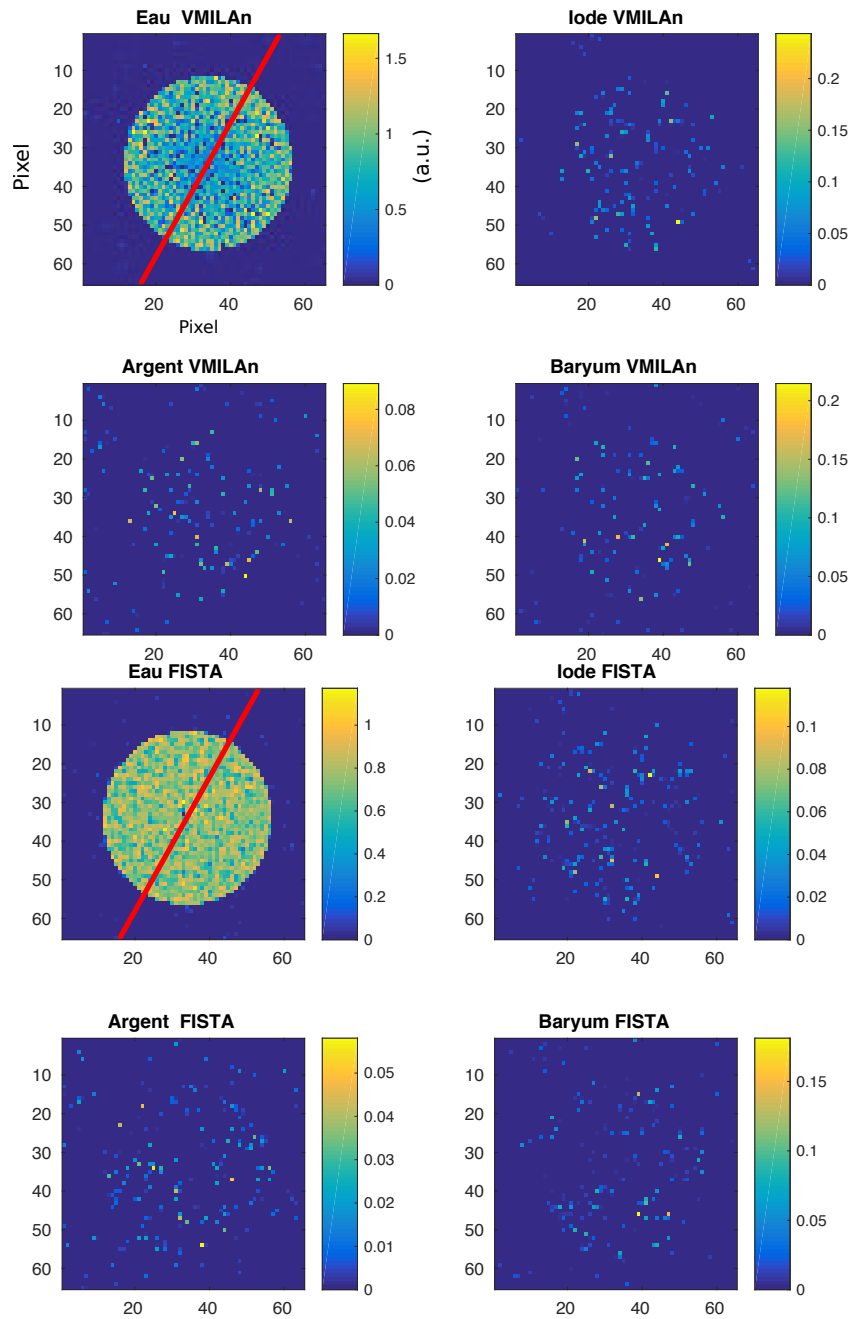


FIGURE 3.44 – Reconstruction de la tranche 1 au dessus du plan médian ( $64 \times 64$  pixels) obtenue avec les algorithmes **VMILAn** et **FISTA** pour données spectrales 3D avec 300 coups dans le blanc et une ouverture de faisceau de 14 degrés.

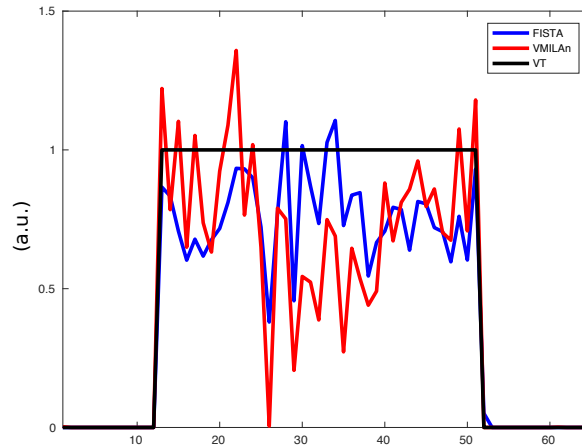


FIGURE 3.45 – Profiles (64 pixels) des reconstructions de l'eau de la première tranche de l'objet pour les algorithmes **FISTA** et **VMILAn** comparés au profil de la vérité de terrain (VT sur la figure).

### 3.2.3.5 Étude de bruit

Afin de tester la robustesse de nos méthodes face aux niveaux de bruit que l'on peut avoir lors d'une acquisition réelle, nous avons mené une étude de bruit sur 4 statistiques de comptage induisant 4 niveaux de bruit différents : 500 (19 dB), 1000 (22.2 dB), 2500 (26.2 dB) et 5000 (30 dB) coups dans le blanc. Pour chaque cas, nous avons simulé 5 réalisations de bruit aléatoire.

Nous mesurons le **SNR** global des reconstructions en sortie pour les algorithmes **FISTA** et **VMILAn** ainsi que le **SNR** de chaque composant. Nous calculons la moyenne des 5 **SNR** obtenus par statistique de comptage et nous regardons l'évolution de ces moyennes en fonction de l'évolution de la statistique de comptage. Nous calculons par ailleurs l'écart-type qui nous servira de barre d'erreur par rapport aux réalisations. Les algorithmes sont initialisés avec les mêmes paramètres que ceux décrits précédemment avec un critère d'arrêt à 1000 itérations. La figure 3.46 montre le

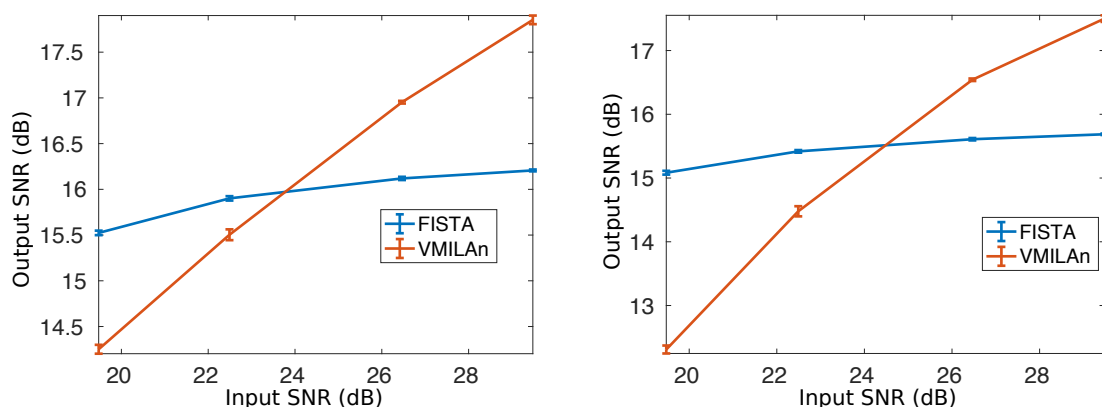


FIGURE 3.46 – Évolution de la moyenne du **SNR** global (à gauche) et de l'eau (à droite) en fonction du **SNR** en entrée.

résultat de l'étude sur le **SNR** global et le **SNR** de l'eau pour les algorithmes **VMILAn** et **FISTA**. On voit sur cette figure que plus le comptage est faible, plus le **SNR** global en sortie de l'algorithme **VMILAn** est faible. Ceci est expliqué par le fait que le bruit est amplifié par le filtre conique de la métrique  $M_1$ . Néanmoins, la qualité du **SNR** global augmente de manière linéaire en fonction de la statistique de comptage.

D'autre part, l'algorithme **FISTA** est moins sensible lorsque le **SNR** en entrée est faible, mais sa

qualité augmente de manière logarithmique comme on le constate en figure 3.46.

On voit aussi sur cette même figure le résultat de l'étude du **SNR** pour l'eau. On remarque immédiatement que l'allure de l'évolution du **SNR** en sortie de l'eau en fonction de la statistique de comptage est la même que celle du **SNR** global et cela pour les deux algorithmes.

Ceci est expliqué par le fait que l'eau est présente en grande quantité dans l'objet et comme le **SNR** dépend de la variance et donc du nombre de voxels, le **SNR** de l'eau influence le **SNR** global. On note à nouveau ici une évolution logarithmique pour l'algorithme **FISTA** et une évolution linéaire pour l'algorithme **VMILAn**.

On reporte les résultats pour les agents de contraste dans la figure 3.47. On remarque tout d'abord

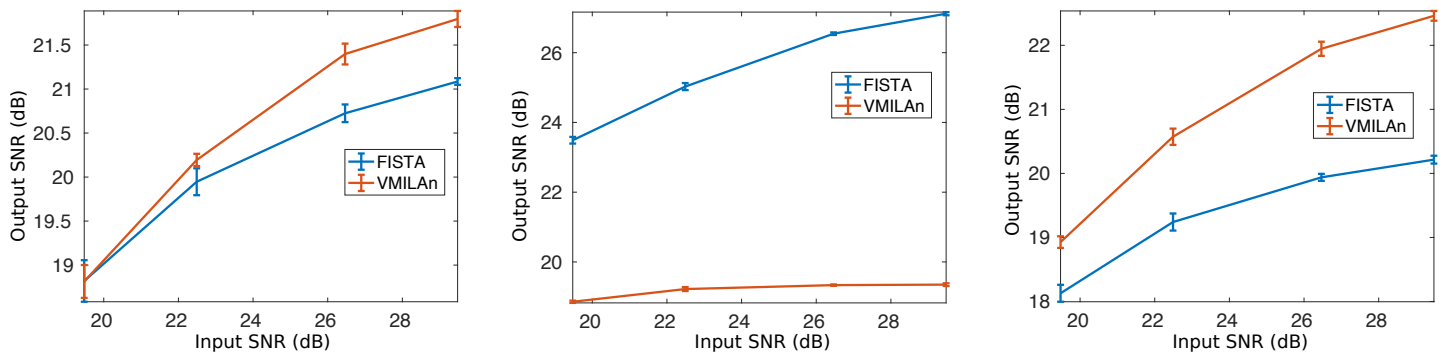


FIGURE 3.47 – Évolution de la moyenne du **SNR** de l'iode (à gauche), de l'argent (au milieu) et du baryum (à droite) en fonction du **SNR** en entrée.

que la qualité du **SNR** de la reconstruction du composant iode est similaire pour les deux algorithmes lorsque le **SNR** en entrée est bas. Celle-ci augmente en fonction du **SNR** en entrée avec un résultat légèrement meilleur pour l'algorithme **VMILAn** comme on le constate en figure 3.47. D'autre part, le **SNR** du baryum est plus élevé pour l'algorithme **VMILAn** que pour l'algorithme **FISTA**.

On remarque aussi que le **SNR** du composant argent est beaucoup plus élevé pour l'algorithme **FISTA** que pour l'algorithme **VMILAn**. Ceci a été constaté précédemment et dans la plus part des résultats obtenus, le composant argent est celui que l'algorithme **FISTA** reconstruit en priorité, même lorsque la statistique comptage est basse.

En conclusion cette étude a permis de montrer que les méthodes développées sont assez robustes aux faibles statistiques de comptage. Nous avons relevé que l'algorithme **FISTA** reconstruit en priorité le composant argent, tandis que **VMILAn** reconstruit les composants de manière assez similaire. De manière générale, on obtient de meilleures performances en termes de **SNR** avec l'algorithme **VMILAn** qu'avec l'algorithme **FISTA**, mais ce dernier est moins sensible que l'algorithme **VMILAn** lorsque le **SNR** en entrée est bas. La sensibilité au bruit de l'algorithme **VMILAn** est due au filtre conique, qui amplifie les hautes fréquences. Notons par ailleurs que la seule contrainte utilisée au cours de cette étude est la contrainte de positivité.

Une autre perturbation qui peut affecter les **SNR** en entrée est celle causée par une mauvaise estimation de la matrice des filtre **F**. Nous allons dans le prochain paragraphe présenter les résultats d'une étude de robustesse des deux algorithmes par rapport aux perturbations de cette matrice.

### 3.2.3.6 Robustesse aux perturbations engendrées par les filtres



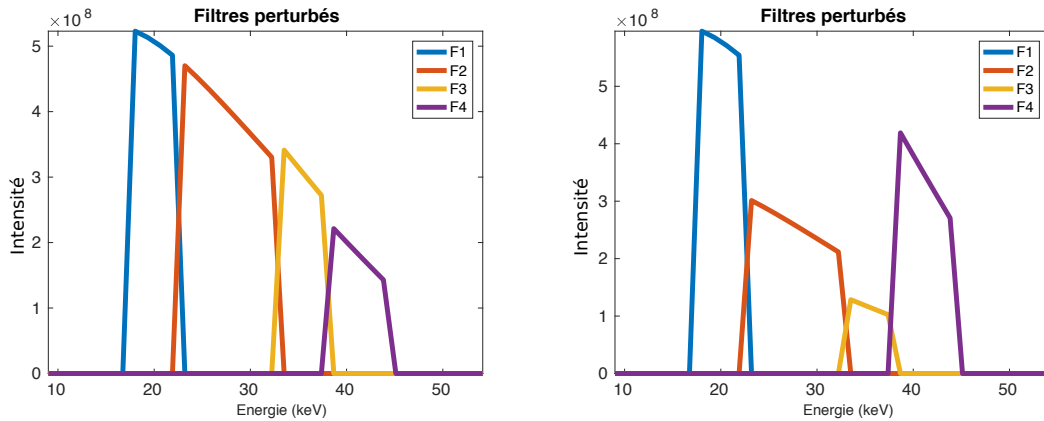


FIGURE 3.48 – Filtre en énergie perturbés par rapport aux filtres en énergie présentés en figure 3.30 — A gauche,  $\pm 10\%$ . A droite,  $\pm 40\%$ .

### Étude de perturbation de l'amplitude

On suppose dans nos approches que la matrice  $F$  est parfaitement connue, or celle-ci est souvent issue d'un processus d'estimation ou de calibration, ce qui rend en pratique sa connaissance inexacte ou approximative. L'objectif ici est de tester la robustesse de nos méthodes face aux perturbations du filtre qui peuvent être induites par une mauvaise estimation de celui-ci. Nous avons dans un premier temps perturbé l'amplitude des filtres de la matrice  $F$ . Cela peut se produire si l'intensité du flux de la source est mal estimé ou si l'épaisseur ou la pureté des filtres métalliques ne sont connues qu'approximativement.

Pour cela, on multiplie chaque colonne de la matrice  $F$  par un facteur qui augmente ou diminue d'un certain pourcentage l'amplitude des filtres. Un exemple de filtres perturbés est donné en figure 3.48.

Afin de quantifier l'impact de cette perturbation, nous mesurons le **SNR** pour des résultats de reconstructions obtenues à partir de 5 perturbations aléatoires pour 4 niveaux de perturbation :  $\pm 10\%$  (donne un **SNR** en entrée  $\approx 22$  dB),  $\pm 20\%$  (donne un **SNR** en entrée  $\approx 19$  dB),  $\pm 30\%$  (donne un **SNR** en entrée  $\approx 16$  dB) et  $\pm 40\%$  (donne un **SNR** en entrée  $\approx 12$  dB).

La figure 3.49 montre le résultat de l'étude sur le **SNR** global et celui de l'eau pour les algorithmes

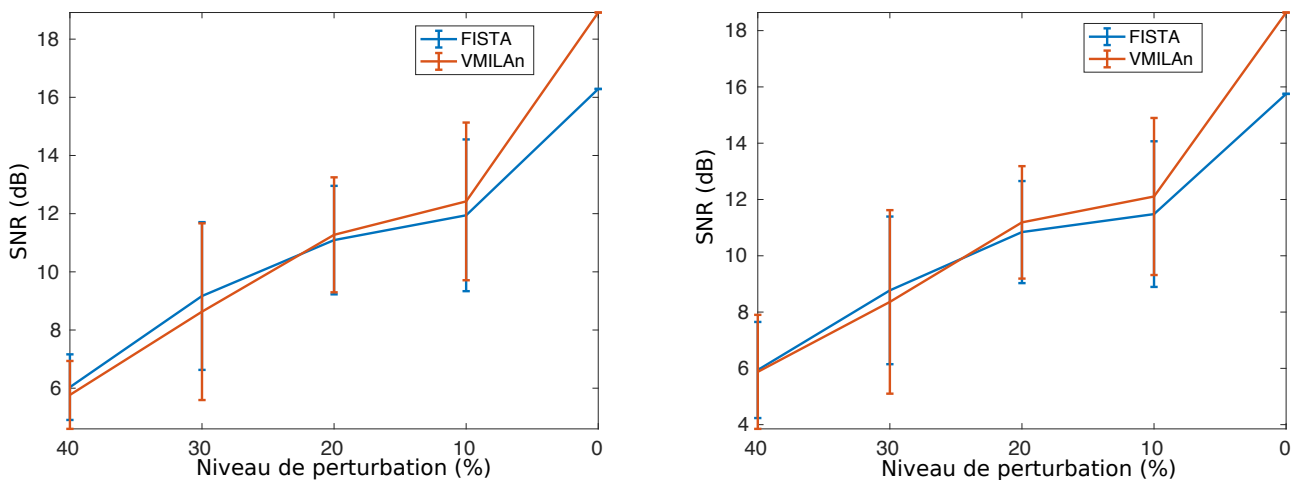


FIGURE 3.49 – Évolution des moyennes du **SNR** global (à gauche) et de l'eau (à droite) en fonction du niveau de perturbation de  $F$  (en %).

**VMILAn** et **FISTA**. On voit sur cette figure que l'évolution du **SNR** global en fonction du niveau

de perturbation est assez similaire pour les deux algorithmes. Les barres d'écart-types sont sensiblement les mêmes avec un léger avantage pour l'algorithme **FISTA**. La qualité du **SNR** global augmente de manière linéaire en fonction du niveau de perturbation.

On montre sur cette même figure les résultats de l'étude du **SNR** pour l'eau. On remarque encore une fois la similarité entre l'évolution du **SNR** en sortie de l'eau en fonction du niveau de perturbation et l'évolution du **SNR** global.

La figure 3.50 montre les résultats de l'étude du **SNR** pour l'iode, l'argent et le baryum. La qualité

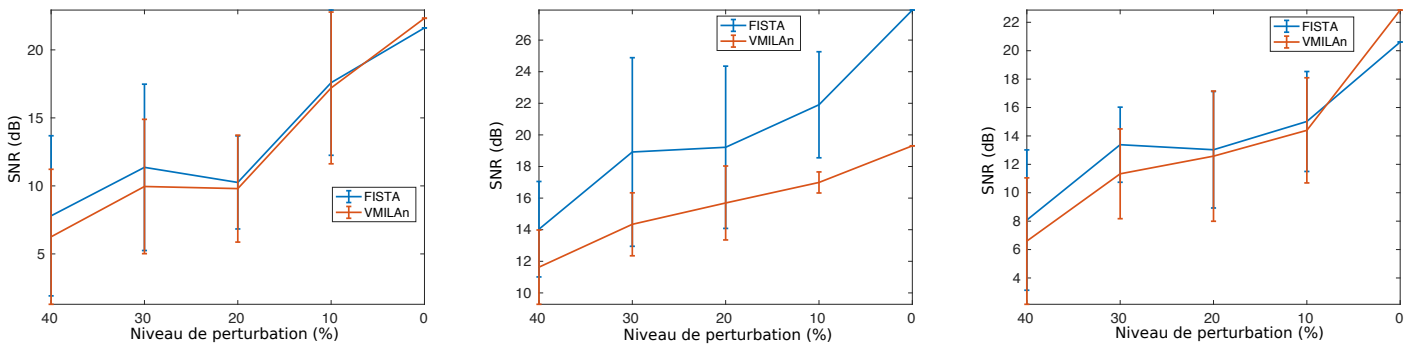


FIGURE 3.50 – Évolution des moyennes du **SNR** de l'iode (à gauche), de l'argent (au milieu) et du baryum (à droite) en fonction du niveau de perturbation de **F** (en %).

du **SNR** de la reconstruction du composant iode est similaire pour les deux algorithmes. Celle-ci augmente en fonction du niveau de perturbation avec un résultat légèrement meilleur pour l'algorithme **FISTA** comme on le constate en figure 3.50.

On remarque une fois de plus que le **SNR** du composant argent est beaucoup plus élevé pour l'algorithme **FISTA** que pour l'algorithme **VMILAn**. Néanmoins, les barres d'écart-types sont très grandes pour l'algorithme **FISTA** et on constate des variations allant jusqu'à 6 dB. D'autre part, le **SNR** du baryum est en moyenne légèrement plus élevé pour l'algorithme **FISTA** que l'algorithme **VMILAn**.

En conclusion, cette étude a permis de montrer que les méthodes développées sont assez sensibles aux perturbations de l'amplitude de la matrice **F**. Nous pouvons confirmer que l'algorithme **FISTA** reconstruit légèrement mieux les composants contenant de l'agent de contraste (particulièrement l'argent) tandis que **VMILAn** reconstruit les composants de manière assez similaire sans mettre en priorité un composant en particulier. De manière générale, on obtient de meilleures performances en termes de **SNR** avec l'algorithme **FISTA** qu'avec l'algorithme **VMILAn**, mais ce dernier présente des variations d'écart-types plus faibles que l'algorithme **FISTA**, qui est donc légèrement plus robuste aux variations. Nous allons présenter dans le prochain paragraphe les résultats d'une étude de robustesse d'un autre type de perturbation de la matrice des filtres **F**.

### Étude de perturbation du support

Il arrive parfois que les seuils en énergie de chaque pixel soient connus approximativement. En effet, lorsque l'on veut fixer un seuil en énergie  $s_k$  pour effectuer une acquisition, on calibre les pixels du détecteur pour que ceux-ci ne comptent les photons qu'à partir du seuil  $s_k$ . Or la calibration des pixels du détecteur n'est qu'approximative, comme nous le verrons au paragraphe 3.3.1.2, et tous les pixels ne seront pas calibrés exactement à la même valeur de seuil  $s_k$ . Ainsi, ces pixels mal calibrés compteront à partir d'une valeur de seuil plus grande ou plus petite que la valeur  $s_k$ , ce qui peut induire un changement de support des filtres.

Une manière de simuler cette perturbation est par exemple de mettre à zéro la première et/ou

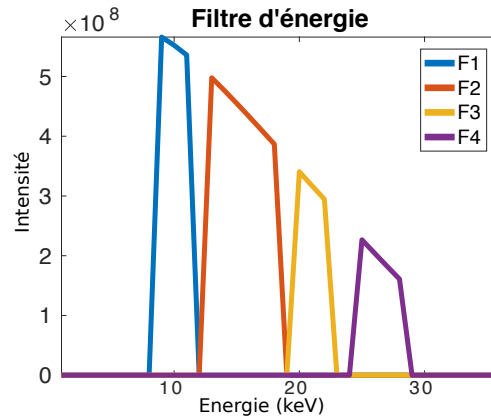


FIGURE 3.51 – Filtres en énergie sous-échantillonnés. Dans ces filtres, il y a moins d'information en énergie que pour ceux présentés en figure 3.30.

la dernière valeur des filtres. Un exemple de cette perturbation est en figure 3.51.

Nous lançons une expérience sur des données non bruitées dans un premier temps. En figure 3.52, on reporte les résultats des décroissances de la fonction objectif  $J(\mathbf{A})$  et l'évolution du **SNR** en fonction du nombre d'itérations des algorithmes **FISTA** et **VMILAn** pour deux cas : le cas où la matrice  $\mathbf{F}$  est perturbée et le cas normal. On remarque que cette perturbation ralentit la vitesse de convergence de la fonction objectif des deux algorithmes. Par ailleurs, le **SNR** global est lui aussi légèrement affecté :

- dans le cas normal, il atteint 18.1 dB pour **VMILAn** et 16 dB pour **FISTA**,
- dans le cas perturbé, il atteint 16.9 dB pour **VMILAn** et 14.9 db pour **FISTA**.

Afin de d'évaluer l'impact sur les différentes cartes de composants de la matrice  $\mathbf{A}$ , nous mesurons l'évolution du **SNR** des 4 composants reconstruites pour les deux cas. Le résultat est reporté en figure 3.53.

On remarque qu'il y a une légère diminution du **SNR** lorsque le filtre est perturbé, la plus prononcée étant celle de l'iode. Ce résultat est confirmé par les reconstructions en figures 3.54 et 3.56 où l'on voit bien que l'ambiguïté dans la carte de concentration contenant l'iode est plus prononcée qu'ailleurs lorsque le filtre est perturbé, contrairement au cas où le filtre ne l'est pas comme en figures 3.55 et 3.57. Remarquons aussi qu'il y a moins d'ambiguïté dans le composant baryum lorsque le filtre est perturbé dans les reconstructions obtenues avec **VMILAn**.

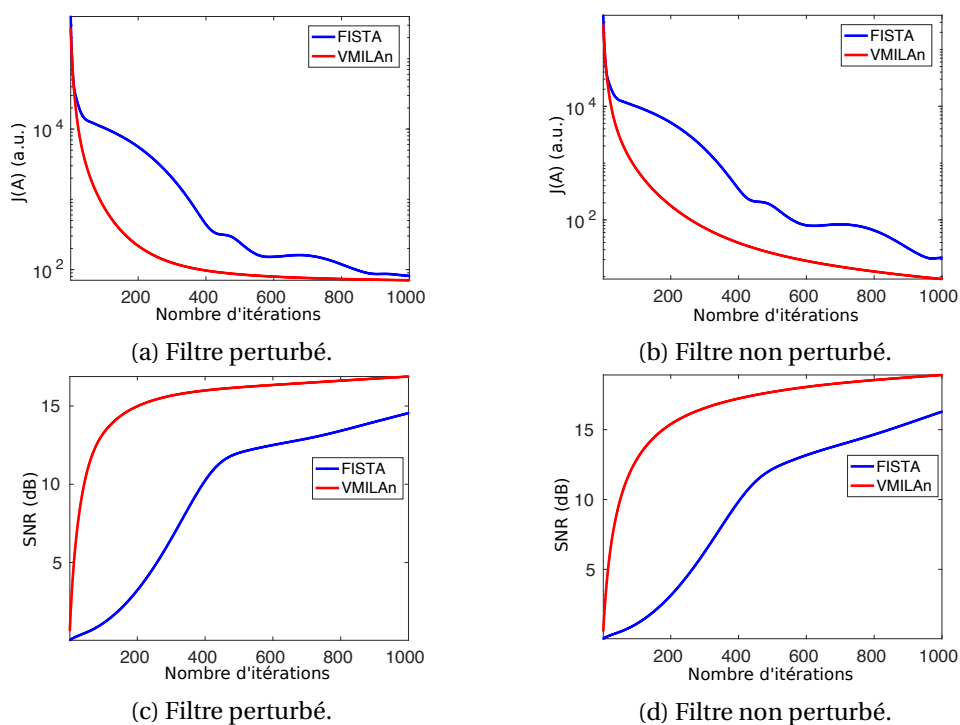


FIGURE 3.52 – Comparaison de l'évolution du **SNR** et de la fonction objectif  $J(A)$  en fonction du nombre d'itérations des algorithmes **FISTA** (en bleu) et **VMILAn** (en rouge).

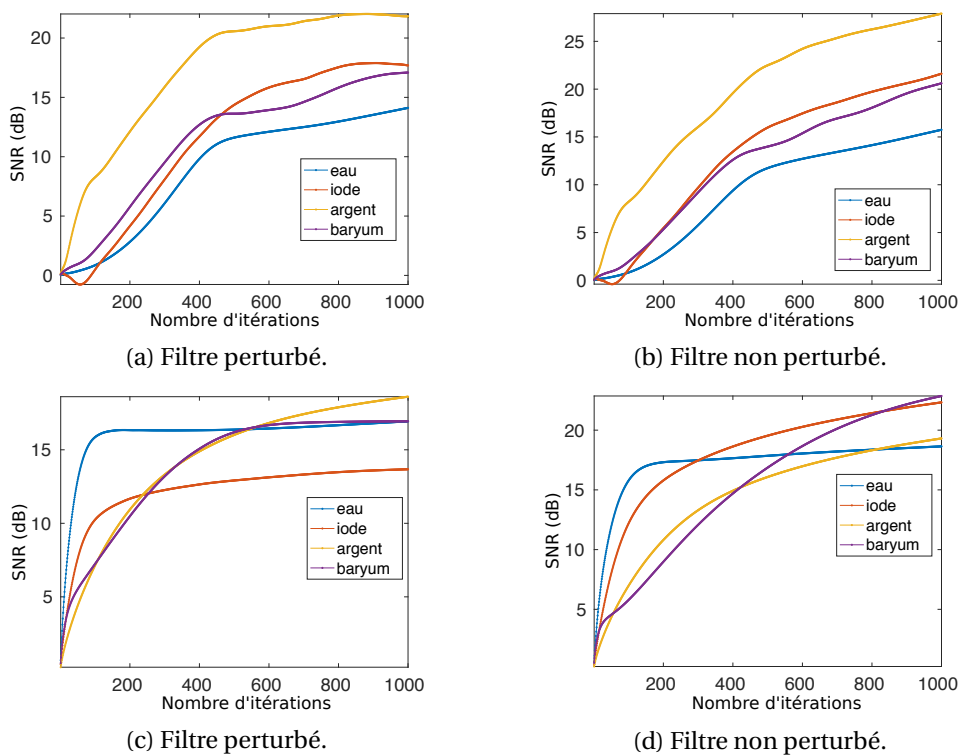


FIGURE 3.53 – Comparaison de l'évolution du **SNR** des 4 composantes en fonction du nombre d'itérations des algorithmes **FISTA** (en haut) et **VMILAn** (en bas).

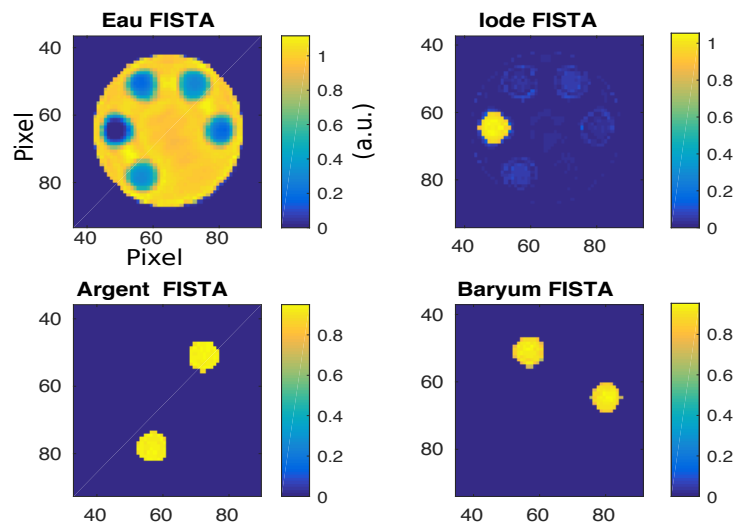


FIGURE 3.54 – Reconstruction du plan médian ( $64 \times 64$ ) obtenue avec **FISTA** pour données spectrales 3D avec la matrice **F** perturbée.

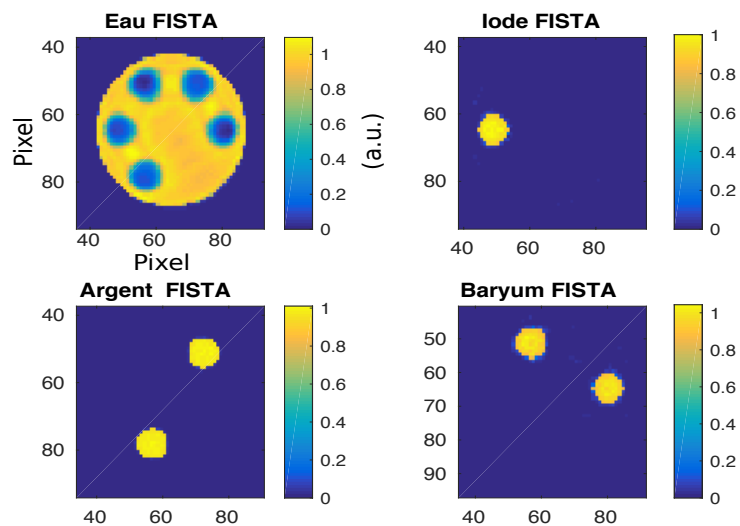


FIGURE 3.55 – Reconstruction du plan médian ( $64 \times 64$ ) obtenue avec **FISTA** pour données spectrales 3D avec la matrice **F** non perturbée.

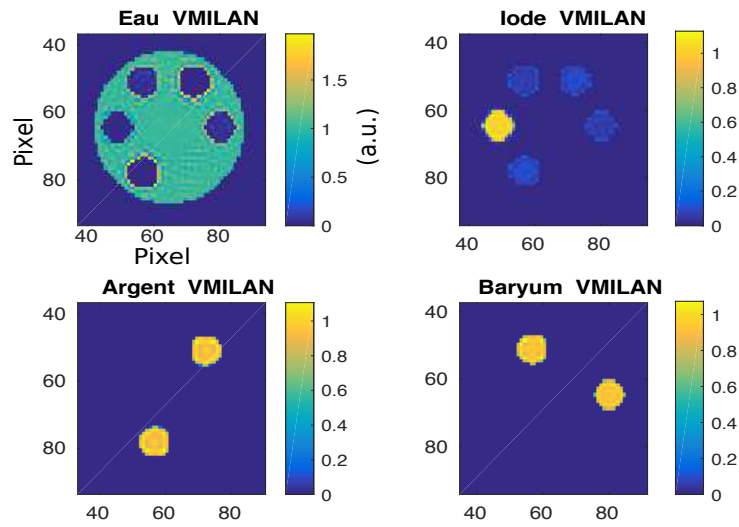


FIGURE 3.56 – Reconstruction du plan médian (64 × 64) obtenue avec **VMILAN** pour données spectrales 3D avec la matrice F perturbée.

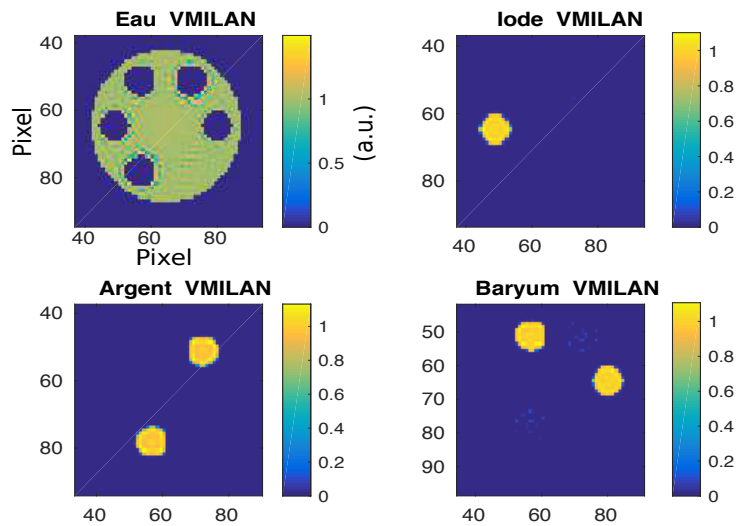


FIGURE 3.57 – Reconstruction du plan médian (64 × 64) obtenue avec **VMILAN** pour données spectrales 3D avec la matrice F non perturbée.

### 3.2.4 Conclusion sur les données simulées

Nous avons présenté dans cette section les tests sur des données simulées que nous avons menés de manière graduelle. En effet, si l'objectif est de tester nos algorithmes sur des données réelles, les tests menés jusqu'ici ont permis de s'assurer que nos méthodes fonctionnent pour des problèmes plus simples.

Dans un premier temps nous avons vu en sous-section 3.2.1 qu'on peut obtenir à l'aide d'algorithmes proximaux des résultats bien meilleurs que ce qu'une reconstruction standard telle que **FBP** peut faire. Nous avons vu que l'introduction d'une métrique dans une descente proximale permet d'accélérer significativement la convergence des algorithmes proximaux, particulièrement lorsque la métrique contient des informations précises de second ordre sur l'itéré en cours.

Ceci s'est confirmé avec la suite des tests en sous-sections 3.2.2 et 3.2.3. Nous avons vu que les algorithmes demeuraient robustes au bruit et plus important encore, que le changement de la géométrie du faisceau lors du passage du 2D au 3D n'influence pas les résultats de convergence bien que l'on garde la même paramétrisation des algorithmes. Plus précisément, la même métrique  $\mathbf{M}_1$  est utilisée dans tous les cas.

D'autre part, la perturbation de la matrice  $\mathbf{F}$  supposée connue et décrivant la réponse spectrale du système n'influe pas significativement sur le résultat de convergence lorsque cette perturbation est faible. Nous avons majoritairement décidé d'utiliser la positivité comme régularisation dans notre modèle. La contrainte  $l_1$  a été difficile à paramétrer à cause de la sensibilité des algorithmes aux variations des hyperparamètres de régularisation. Nous n'avons pas poussé l'étude plus loin concernant ces hyperparamètres pour nous concentrer sur la résolution du problème de **TDM** spectrale.

Enfin, le problème de séparation et reconstruction simultanée a bien été résolu pour le cas où les données sont simulées. Nous allons voir dans la suite les résultats que nous avons obtenus sur des données réelles acquises sur le prototype de micro-CT PIXSCAN-FLI.

## 3.3 Tests et résultats sur données réelles

Dans cette section, nous allons présenter les résultats obtenus avec des données réelles acquises sur le prototype de micro-CT PIXSCAN-FLI. Nous allons dans un premier temps décrire le protocole d'acquisition, puis nous présenterons les résultats obtenus avec deux jeux de données.

### 3.3.1 Acquisition de données le prototype PIXSCAN-FLI

#### 3.3.1.1 Interface d'acquisition

Nous allons décrire brièvement le logiciel permettant de contrôler l'ensemble du système lors d'une acquisition **TDM** avec le prototype PIXSCAN-FLI. L'acquisition des données se fait via le logiciel **PyXface** développé par l'équipe imXgam [DUPONT, 2014]. Ce logiciel offre une interface facile d'utilisation et son fonctionnement repose sur un certain nombre de protocoles de communication entre l'ordinateur d'acquisition et les éléments du système d'imagerie. Les principaux systèmes d'action sont :

- le contrôle moteurs (rotation, translations, etc),
- la synchronisation la prise d'image avec l'angle de rotation du support animal,
- le verrouillage de la porte principale,
- l'actionnement de l'obturateur (ou *shutter*) en sortie du faisceau,
- la communication avec la caméra XPAD3 pour le transfert des images
- le contrôle de la tension et de l'intensité de la source.

Ces communications passent par un protocole de transport d'information standardisé **MQTT**<sup>4</sup>, qui permet de centraliser la communication entre des objets connectés.

Le logiciel PyXface offre une interface comprenant trois fenêtres :

- une fenêtre **motors** permettant de gérer la position des différents actionneurs,
- une fenêtre **visio** permettant la prise et la visualisation d'images,
- la fenêtre principale **PyXface** permettant de régler les paramètres de la source, charger la calibration des seuils (*i.e.* la valeur des énergies des seuils) et définir une acquisition.

Les paramètres d'une acquisition sont les suivants :

- la tension, l'ampérage et la taille du point focal du faisceau,
- le choix du filtre métallique ou liquide en sortie du faisceau,
- les positions des actionneurs de la plateforme objet le pas de rotation,
- la table de **calibration**,
- le temps d'exposition par image et le nombre d'angles de rotation,
- le nombre de fonds blancs et de fonds noirs (dont nous détaillerons l'utilité au prochain paragraphe),
- le mode d'acquisition : image par image ou en rotation continue,
- le nom du fichier, l'emplacement de sauvegarde.

Dans la suite, nous allons décrire les différentes étapes de traitement permettant de lancer une acquisition et d'obtenir en fin de traitement des reconstructions tomographiques.

### 3.3.1.2 Calibrations du prototype PIXSCAN-FLI

#### Calibration de la géométrie

Afin d'obtenir le bon grandissement et des images reconstruites aux contours bien nets, nous devons connaître précisément les paramètres géométriques du système. Il existe plusieurs paramètres, parmi eux la distance de la caméra à la source et à l'objet, ou la distance de la source à l'objet. Il est impossible d'estimer ces distances de manière mécanique avec une précision suffisante en utilisant un appareil de métrologie. De plus, la position du point d'émission de la source est en général mal connu.

Une méthode basée sur la **caractérisation de la trajectoire elliptique** du centre d'une bille d'acier en rotation autour d'un axe vertical a été développée par l'équipe imXgam [BONISSENT et collab., 2009]. Cette bille métallique est incluse dans un fantôme cylindrique en mousse que l'on peut voir en figure 3.58. La **calibration géométrique** du prototype PIXSCAN-FLI est réalisée avec l'acquisition d'au moins trois acquisitions **TDM** de 360 projections du fantôme à bille positionné à des hauteurs différentes. Après avoir identifié le centre de la bille sur chaque projection, on peut tracer la trajectoire dans un plan de détection vertical virtuel comme on peut le voir en figure 3.58. La trajectoire de la bille étant décrite théoriquement par une ellipse, on peut retrouver les paramètres géométriques du système en résolvant un problème inverse [BONISSENT et collab., 2009] à l'aide d'un algorithme à région de confiance.

#### Fonds blancs et noirs

Les projections acquises lors d'une acquisition **TDM** doivent être traitées avant la reconstruction. En effet, les pixels défectueux des caméras induisent systématiquement des artefacts circulaires sur les images reconstruites si on ne les prend pas en compte. On parle de **pixels défectueux**

4. <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html>



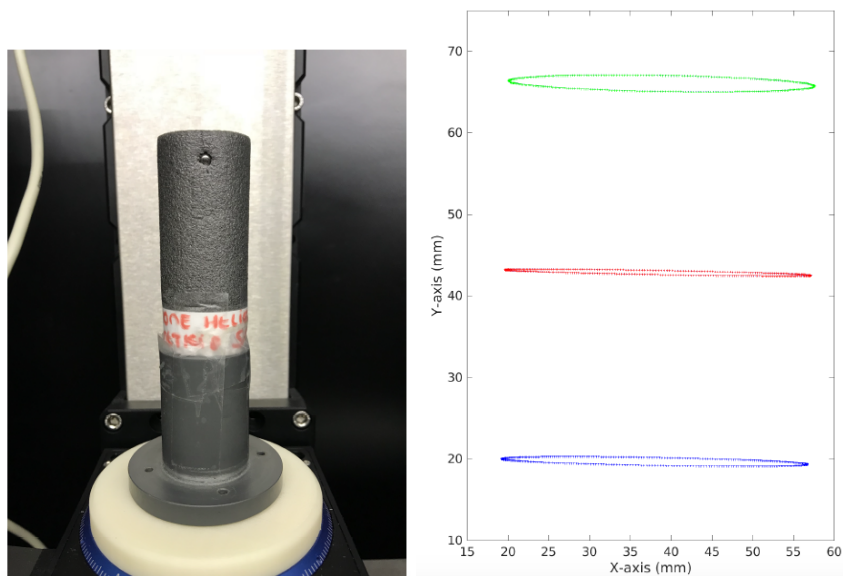


FIGURE 3.58 – A gauche, fantôme de la bille de calibration géométrique. A droite, trajectoires de la projection du centre de la bille dans un plan de détection vertical virtuel tracées à partir de trois acquisitions **TDM** pour trois positions verticales de la bille [PORTAL, 2018].

lorsque le taux de comptage du pixel s'éloigne trop de la moyenne du taux de comptage de l'image. Il peut s'agir :

- de pixels morts déconnectés de l'électronique,
- de pixels possédant un mauvais réglage de leur seuil,
- des pixels cachés par la barrette supérieure du fait de la géométrie tuilée de la caméra.

Afin d'identifier les pixels défectueux, chaque acquisition **TDM** est composée, en plus des projections de l'objet, de 50 fonds blancs et de 50 fonds noirs. Les fonds blancs sont des images acquises avec les mêmes paramètres d'acquisition utilisés pour l'objet, mais sans l'objet entre la source et la caméra. Les fonds noirs sont des images acquises avec la source éteinte.

L'acquisition des fonds blancs et noirs est faite systématiquement après l'acquisition **TDM** de l'objet pour être au plus près possible de l'état de la caméra pendant la prise des données. L'analyse de la moyenne des 50 fonds blancs va permettre d'identifier les pixels présentant un mauvais taux de comptage. Les pixels s'écartant du taux de comptage moyen des fonds blancs seront alors masqués. Les 50 fonds noirs vont permettre d'identifier les pixels présentant un signal en l'absence de rayons X. On masquera lors de cette étape les pixels comptant plus de 10 coups dans au moins un des fonds noirs.

Dans le cas d'une calibration des seuils à l'énergie d'un K-edge, ce dont nous discuterons dans le paragraphe suivant, les pixels mal calibrés lors du processus de calibration sont détectés par la procédure décrite ci-dessus puis masqués avant d'envoyer les données au programme de reconstruction des mesures.

La valeur des pixels masqués est remplacée par l'**interpolation linéaire** de leurs trois plus proches voisins lors du traitement des projections pré-reconstruction.

### Calibration du détecteur du prototype PIXSCAN-FLI

Comme nous l'avons déjà mentionné dans la sous-section 1.2.2 du chapitre 1, la caméra XPAD3 dispose d'un seuil de discrimination en énergie permettant de sélectionner les photons détectés. Ce seuil de discrimination peut être réglé et ajusté pour chaque pixel. Pour cela, nous disposons

de plusieurs registres pour régler des convertisseurs numérique-analogique ou **DAC** (*Digital to Analog Converter*) :

- le registre **ITH** pour *Intensity Threshold*, règle un **DAC** codé sur 8 bits (256 pas de réglage) et permet un réglage global autour d'une énergie donnée à l'**ensemble** des pixels d'un chip,
- le registre **DACL** pour *Digital to Analog Converter Local*, règle un **DAC** codé sur 6 bits (64 pas de réglage) et permet d'ajuster le seuil de chaque pixel pris séparément. C'est ce registre qui permet de corriger les différences de gain entre chaque pixel.

Une configuration de calibration de la caméra XPAD3 est constituée de deux fichiers : un fichier de configuration globale qui contiendra notamment les valeurs des registres globaux de chaque chip, et un fichier de configuration locale, appelé **table de calibration**, contenant les valeurs des **DACL** des pixels de chaque chip.

Le processus de calibration des seuils comprend deux étapes :

- la première consiste à effectuer un balayage des **ITH** avec **DACL** = 32 pour tous les pixels pour trouver une valeur **ITH** optimale pour chaque chip, ce qui permet un réglage grossier en énergie de chaque pixel,
- la seconde consiste à effectuer un balayage des **DACL** avec **ITH** optimal pour déterminer le **DACL** pour chaque pixel à **ITH** fixé, ce qui permet un réglage fin en énergie de chaque pixel.

Cette calibration peut être effectuée au ras-du-bruit (~ 5 keV), mais aussi aux énergies comprises entre 5 et 60 keV pour lesquelles l'amplificateur de charges de chaque pixel reste linéaire.

Dans le cas de l'imagerie d'absorption classique, les seuils de la caméra sont calibrés au ras-du-bruit. Cette calibration au ras-du-bruit est entièrement automatisée et se fait via l'interface du logiciel qui accompagne la caméra XPAD3. La réponse des pixels pouvant varier dans le temps en fonction de la température ou du vieillissement de l'électronique, la calibration est effectuée généralement une fois par mois pour utiliser la caméra de manière optimale.

Pour calibrer les seuils à une énergie donnée, une des méthodes est d'utiliser un faisceau de lumière synchrotron monochromatique [CASSOL et collab., 2013]. Cependant, nous ne disposons que d'une source polychromatique au CPPM. Pour cela, une autre méthode a été mise au point [CASSOL et collab., 2013]. Celle-ci consiste à utiliser le spectre du faisceau polychromatique filtré par un matériau présentant un K-edge à l'énergie que l'on cherche à calibrer. Dans le cas de la **TDM** spectrale, on utilise comme constituant du filtre métallique ou liquide l'élément que l'on cherche à discriminer sur les images spectrales (argent, iode, etc.).

On effectue un balayage des **DACL** qui va permettre de fixer le **DACL** à l'énergie du K-edge de l'élément constituant le filtre. Chaque pixel aura alors une réponse décrite par une courbe dont on voit un exemple en figure 3.59. On voit dans cette courbe **un point de cassure** dû à la présence du K-edge du filtre, puis un accroissement légèrement concave avant un fort accroissement suivi d'une retombée rapide. La discontinuité, qui survient pour des valeurs de **DACL** élevées, est due au partage de charge et au bruit.

Pour correctement calibrer le pixel, il faut identifier précisément le **point de cassure**. Pour cela, on modélise la réponse du pixel à l'aide de deux équations :

- une équation linéaire pour décrire la partie de la courbe avant la cassure (donc 2 paramètres à estimer),
- une équation du second degré pour décrire la courbe après la cassure (donc 3 paramètres à estimer). Ceci correspond à la droite qui longe la courbe après le point de cassure rouge sur la figure 3.59.

Ces paramètres sont estimés de manière itérative en résolvant là aussi un problème inverse. Les domaines de variation de ces paramètres de minimisation sont ajustables par l'expérimentateur de manière empirique selon les chips du détecteur. Cette étape est effectuée pour l'ensemble des

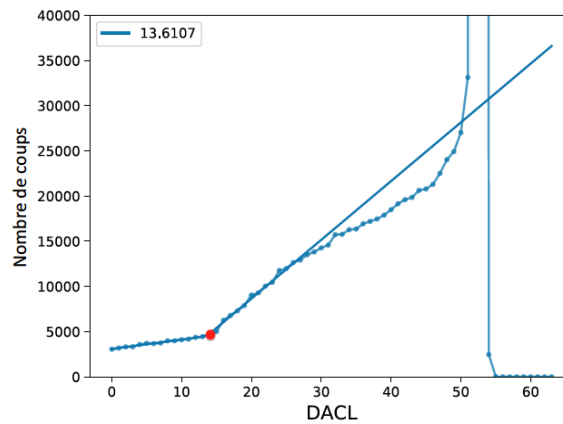


FIGURE 3.59 – Courbe résultant du balayage des **DACL** derrière un filtre d'argent pour un pixel de la caméra XPAD3. Le point rouge marque le point de cassure qui correspond à l'énergie du K-edge de l'élément dont est constitué le filtre.

pixels de la caméra afin de déterminer chaque **DACL** à l'énergie considérée. La difficulté de cette méthode réside dans le fait que les pixels du détecteur ne se comportent pas de manière homogène d'un chip à un autre et d'une barrette à une autre. Ainsi, il faut toujours vérifier l'uniformité et la pertinence de la calibration, ce qui peut représenter un temps de travail assez long, car la méthode n'est pas entièrement automatisée. On trouvera dans la section 2.3 de la thèse de Loriane Portal [PORTAL, 2018] plus de détails à ce sujet.

Afin de réaliser des acquisition spectrales avec le prototype PIXSCAN-FLI et d'évaluer les algorithmes développés sur ces données, nous avons calibré le détecteur au ras-du-bruit et au K-edge de l'argent (25.5 keV) et de l'iode (33.2 keV). A partir de la calibration à l'argent, nous avons déduit une calibration à 22 keV en ajoutant 7 pas de **DACL** à la calibration à 25.5 keV (l'énergie diminue lorsque le **DACL** augmente). Nous avons également utilisé une calibration à 36.7 keV en retirant 7 pas de **DACL** à la calibration de l'iode. Un pas de **DACL** équivaut à environ 0.5 keV [PORTAL, 2018], mais comme nous l'avons vu ci-dessus, tous les pixels n'ont pas la même réponse. Cela entraîne une perte de qualité des calibrations lorsqu'on n'est plus à une calibration obtenue précisément au un K-edge d'un filtre métallique.

On représente en figure 3.60 le fond blanc de la 4-ème barrette du détecteur XPAD3 pour les 6 acquisitions avec un seuil réglé au ras-du-bruit, puis à 22, 25.5, 29, 33.2 et 36.7 keV. En figure 3.61, les distributions des comptages du détecteur XPAD3 sont moyennées sur 50 fonds blancs que l'on obtient pour chaque acquisition, puis normalisées par le taux de comptage moyen afin de les centrer sur 1 et de pouvoir comparer leur écart-type. Le calcul des écart-types des distributions de chaque calibration confirme le fait que la qualité des calibrations est meilleure pour une calibration ras-du-bruit que pour une calibration au K-edge. Enfin, les calibrations déduites de celles à un K-edge par addition ou soustraction de pas de **DACL** sont à leur tour de moindre qualité par rapport aux calibrations effectuées au K-edge d'un filtre métallique comme on le voit en figure 3.61.

## Conclusion

Cette sous-section décrit les différentes étapes de mise en œuvre du prototype PIXSCAN-FLI. Ces étapes consistent à s'assurer que les données acquises seront exploitables et d'assez bonne qualité pour avoir des reconstructions. L'étape la plus importante est la calibration du détecteur, car la qualité de l'acquisition et la bonne connaissance de la matrice **F** dépendent directement de celle-ci. Dans la prochaine sous-section, nous allons présenter les résultats obtenus à partir des données de **TDM** spectrale acquises avec le prototype PIXSCAN-FLI.



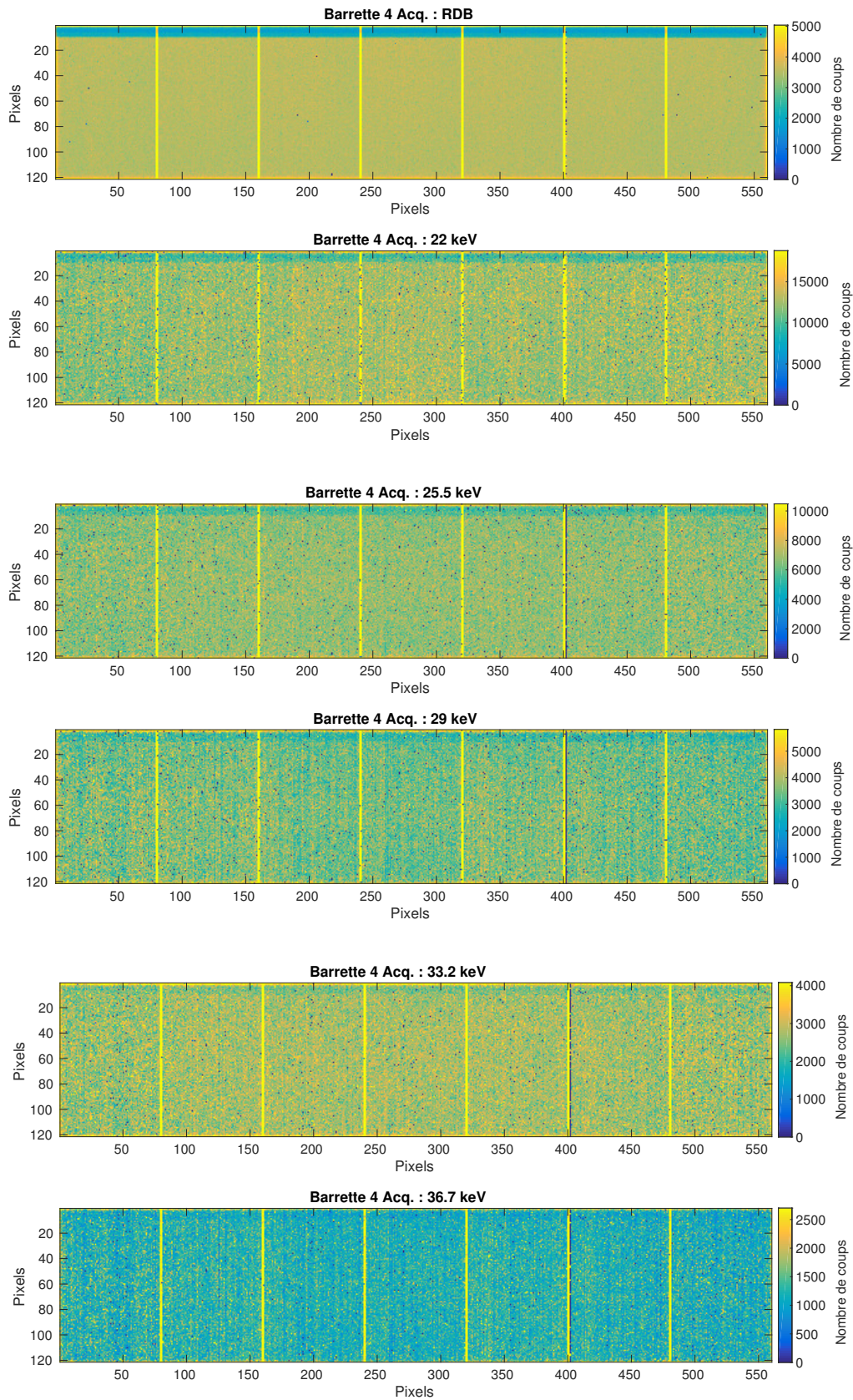


FIGURE 3.60 – Fonds blancs de la 4 ème barrette du détecteur XPAD3 avec un seuil réglé (de haut en bas) : au ras-du-bruit puis à 22, 25.5, 29, 33.2 et 36.7 keV.

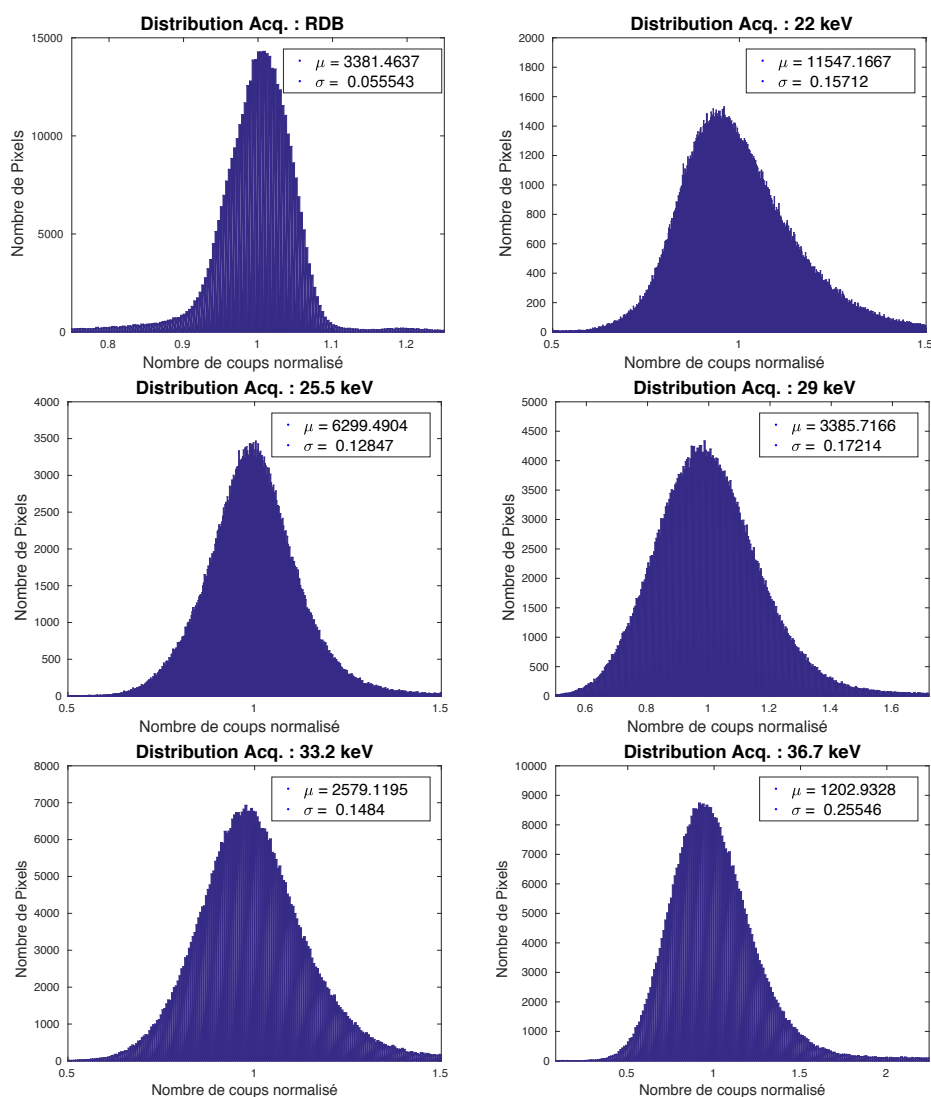


FIGURE 3.61 – Distribution moyenne des comptages du détecteur XPAD3 effectuée sur la moyenne de 50 fonds blancs aux acquisitions suivantes (de gauche à droite et de haut en bas) : au ras-du-bruit puis à 22, 25.5, 29, 33.2 et 36.7 keV. Chaque distribution a été normalisée par la moyenne des comptages sur le détecteur.

### 3.3.2 Résultats sur données réelles

Nous allons présenter dans cette sous-section les tests et les résultats obtenus sur des données réelles. Les données que nous allons traiter ne sont pas maîtrisées comme pour les données simulées. En particulier, le spectre de la source et l'efficacité des pixels du détecteur sont tous deux estimés ou simulés. Ce travail nous permettra d'évaluer les performances des méthodes développées afin de traiter les données spectrales au titre de preuve de concept obtenue dans un contexte préclinique.

#### 3.3.2.1 Fantôme à micro-tubes Eppendorf

##### Protocole 3.3.2.1: fantôme à micro-tubes Eppendorf avec 3 composants

**Objet :** fantôme rempli d'eau et de 6 inserts contenant 2 agents de contraste en solution à base d'argent et d'iode (figure 3.64).

**Taille :**  $D = 512 \times 16 \times 512$  voxels.

**Nombre de projections (angles) :** 720.

**Nombre de composants :**  $K = 3$ .

**Nombre d'acquisitions :**  $M = 5$ .

**Nombre d'échantillons en énergie :**  $N = 43$ .

**Nombre total de mesures :**  $P = 512 \times 32 \times 720 = 1.1 \times 10^7$ .

**Source  $I_0(E)$  :** le spectre d'un tube à anode de tungstène avec une tension fixée à 50 kV.

**Efficacité  $D(E)$  :** l'efficacité des pixels du XPAD3/Si a été simulée par Monte Carlo avec GATE [BRUNNER et collab., 2009; CASSOL et collab., 2015; JAN et collab., 2011].

**Filtre métallique (et leur réponse  $F(E)$ ) :** filtre d'aluminium, épaisseur 0.6 mm.

**Seuils d'acquisition :**  $s_1 = 22.0$  keV,  $s_2 = 25.5$  keV,  $s_3 = 29.0$  keV,  $s_4 = 33.2$  keV et  $s_5 = 36.7$  keV.

**Dictionnaire des sections efficaces  $\Sigma$  :** coefficient d'atténuation linéique (NIST XCOM database<sup>a</sup>) des composants suivants :

- iomeron 400 dilué à une concentration de 95 mg/mL d'iode,
- nitrate d'argent en solution avec une concentration de 120 mg/mL d'argent,
- de l'eau.

Illustration en figure 3.62.

a. <https://www.nist.gov/pml/xcom-photon-cross-sections-database>

La première expérience menée est sur un fantôme à micro-tubes de type **Eppendorf** conçu conjointement avec Loriane Portal. Dans le cadre de sa thèse sur la preuve de concept de séparation de composantes par soustraction au K-edge en utilisant le concept de pixels composites, Loriane Portal a obtenu des résultats sur ces données qui sont présentés dans sa thèse [PORTAL, 2018]. L'analyse de ces mêmes données vise à éprouver le bon comportement de nos méthodes sur des données réelles et à analyser leurs performances. Nous avons placé verticalement 6 micro-tubes de 0,2 mL dans un flacon hermétique rempli d'eau (photo en figure 3.64). Les 6 micro-tubes sont remplis avec des agents de contraste en solution préparés comme suit :

- 3 micro-tubes contenant des solutions d'argent à différentes concentrations : 11, 21.5 et 43 mg/mL préparées à partir d'une solution de nitrate d'argent,
- 3 micro-tubes contenant des solutions d'iode à différentes concentrations : 20, 30 et 40 mg/mL préparées à partir d'iomeron 400.

Après une première acquisition, il s'est avéré que le fantôme bougeait légèrement durant la rotation de la plateforme objet du prototype PIXSCAN-FLI sur laquelle il était posé. Le fond de ce fantôme a alors été équipé de trois petits aimants plats afin de le fixer au support objet du scanner. Ainsi, nous pouvons garantir sa rotation rigide avec le support objet, confirmée par l'analyse des données reconstruites.

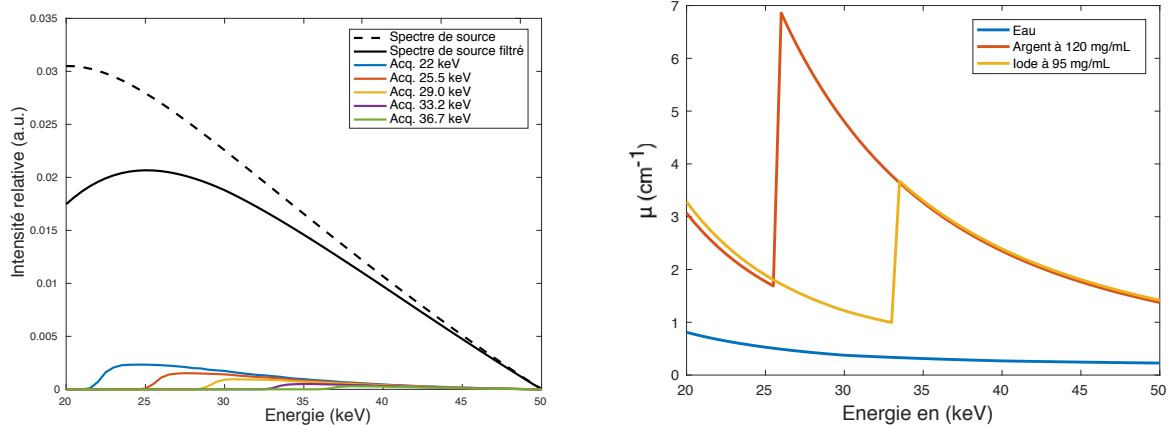


FIGURE 3.62 – A gauche, spectre de source (et sa version filtrée) et réponses de l’efficacité de détection pour les 5 acquisitions utilisées avec une tension appliquée de 50 kV prenant en compte le spectre d’énergie, de l’efficacité des pixels du détecteur estimée et du filtre métallique (0.6 mm d’aluminium) [CASSOL et collab., 2015]. A droite, les coefficients d’atténuation linéique de l’eau, de l’argent en solution à 120 mg/mL et de l’iode en solution à 95 mg/mL.

Afin d’illustrer l’impact des différents composants du prototype PIXSCAN-FLI sur la réponse spectrale du système complet, nous avons simulé cette réponse sur la plage d’énergie comprise entre 20 keV et 50 keV. On considère alors un spectre de source à anode de tungstène dont l’intégrale sur cette plage en énergie est **unitaire** (spectre normalisé à 1). La figure 3.62 montre surtout que la faible efficacité du Silicium aux énergies des acquisitions de l’expérience menée pour le fantôme à micro-tubes Eppendorf impacte fortement la réponse spectrale du système, bien plus que le filtrage par 0.6 mm d’aluminium.

La figure 3.63 est un zoom sur les réponses spectrales totales pour chaque acquisition. On compare sur cette figure les réponses spectrales du prototype PIXSCAN-FLI pour chacune des 5 acquisitions réelles. La principale différence entre les conditions de cette simulation et la précédente vient du fait que l’intensité du spectre utilisé diffère pour chaque acquisition, en raison d’un temps de pose différent notamment pour les acquisitions à 33.2 et 36.7 keV. L’intensité globale, en photons/keV, correspond au taux de comptage dans le blanc obtenu pour chaque acquisition.

Une coupe transverse du fantôme reconstruite à l’aide de l’algorithme **FDK** à partir de données

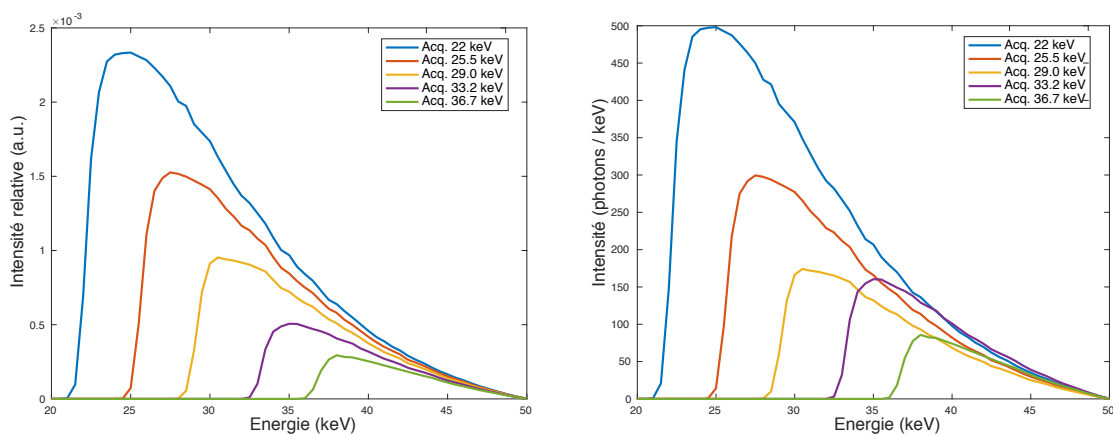


FIGURE 3.63 – A gauche, réponses de l’efficacité de détection idéale pour chaque acquisition. A droite, réponses de l’efficacité de détection réelle.



acquises avec la caméra XPAD3 calibrée au ras-du-bruit est présentée en figure 3.64. Une ana-

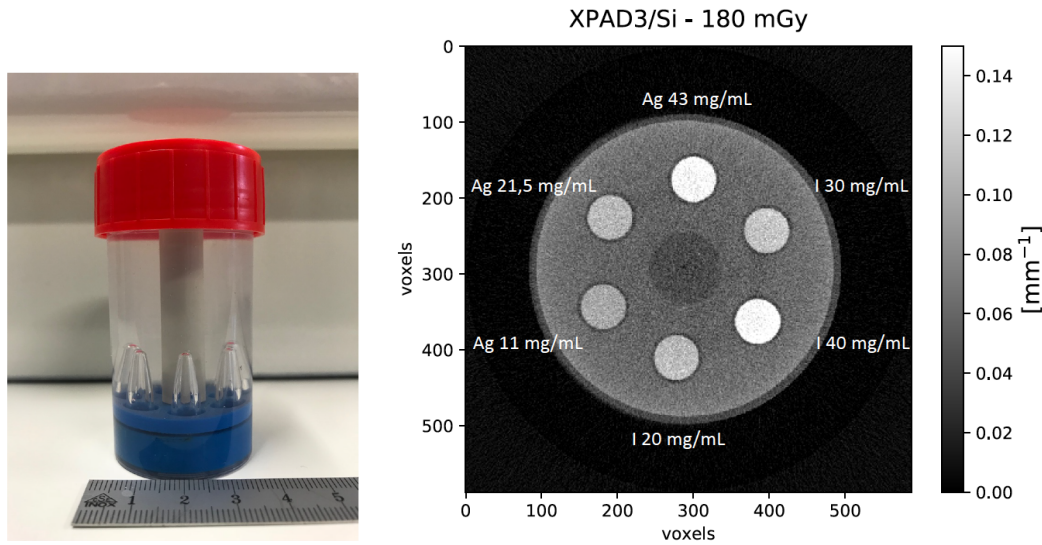


FIGURE 3.64 – Fantôme à micro-tubes Eppendorf : à gauche, une photo du fantôme, à droite, une coupe transverse de l’objet reconstruit avec l’algorithme **FDK** à partir de données acquises avec la camera XPAD3 calibrée au ras-du-bruit [PORTAL, 2018].

lyse visuelle de cette reconstruction **FDK** montre une très faible différence de contraste entre les inserts d’iode à 20 mg/mL et 30 mg/mL, de même qu’entre les inserts d’argent à 21.5 mg/mL et 11 mg/mL, contrairement au contraste entre les inserts d’iode 30 mg/mL et 40 mg/mL, qui est significativement plus important. Nous avons donc voulu dans un premier temps vérifier la fiabilité des concentrations des solutions préparées.

Pour ce faire, la première étape est de convertir les valeurs du coefficient d’absorption dans l’objet reconstruit, appelées **valeurs CT**, en **unités Hounsfield (HU)**. C’est une transformation affine des valeurs CT dans laquelle la densité de l’eau est définie à 0 HU et celle de l’air (apparenté au vide) à  $-1000$ . Ainsi, pour et pour toute valeur CT  $\mu_x$ , la valeur en HU correspondante est donnée par la formule suivante :

$$HU(\mu_x) = 1000 \times \frac{\mu_x - \mu_{\text{eau}}}{\mu_{\text{eau}}}, \quad (3.42)$$

où  $\mu_{\text{eau}}$  est le coefficient d’absorption linéaire de l’eau donné par la reconstruction en valeur CT.

Les valeurs de la reconstruction **FDK** converties en HU ont été reportées en fonction des concentrations des tubes pour chaque agent de contraste, les résultats sont présentés sur la figure 3.65. Il apparaît de façon évidente une non-linéarité en fonction des concentrations (malgré le fait de n’avoir que 3 points de mesure), qui est particulièrement marquée pour les tubes contenant de l’iode. L’hypothèse la plus probable est donc que les concentrations dans ces inserts ont été mal préparées lors de la conception du fantôme.

Afin d’évaluer ce problème, nous avons simulé analytiquement le fantôme de contraste contenant les concentrations supposées des solutions en iode et en argent. Pour cela, nous avons supposé que la calibration au ras-du-bruit utilisée correspondait à un seuil en énergie de 8 keV. Le fantôme simulé est de taille  $256 \times 16 \times 256$ . Une acquisition **TDM** tenant compte du spectre de la source à anode de tungstène, du filtre d’aluminium de 0.6 mm et de l’efficacité des pixels en silicium de la caméra XPAD3 a été simulée, puis une reconstruction **FDK** a été faite.

Les valeurs en HU des deux fantômes reconstruits, le mesuré et le simulé, ont pu être ainsi confrontées. Sur le graphique en figure 3.66, nous avons en ordonnée les valeurs HU simulées à partir du fantôme simulé ci-dessus et en abscisse les valeurs HU mesurées que nous avons reportées en



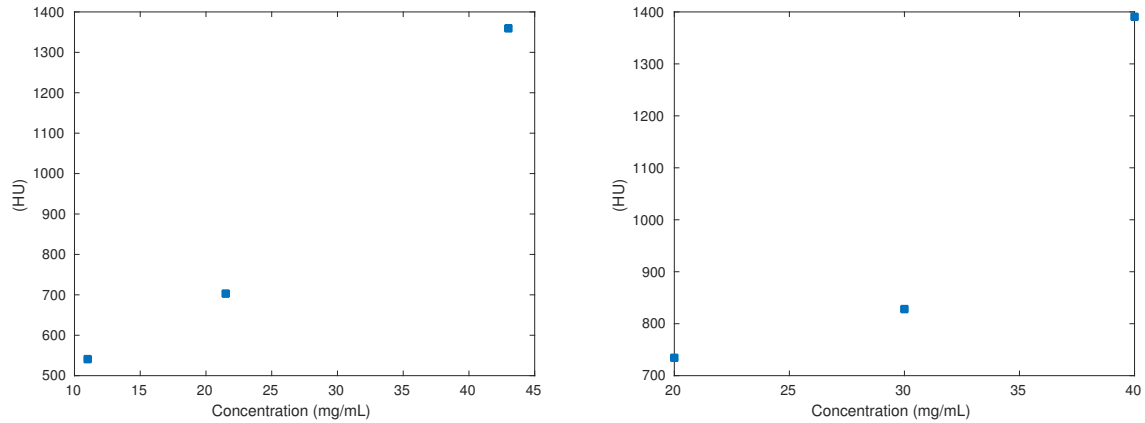


FIGURE 3.65 – Valeurs CT moyenne en HU en fonction de la concentration en argent (à gauche) et iode (à droite) pour l’acquisition au ras-du-bruit.

figure 3.65. Une régression linéaire (droites en pointillés sur la figure 3.66) montre une fois de plus un comportement non-linéaire, ce qui suggère fortement un mauvais contrôle des concentrations, ce dont il faudra tenir compte lors de la discussion des résultats.

Le protocole de test est reporté sur la fiche 3.3.2.1, qui présente les données utilisées pour la

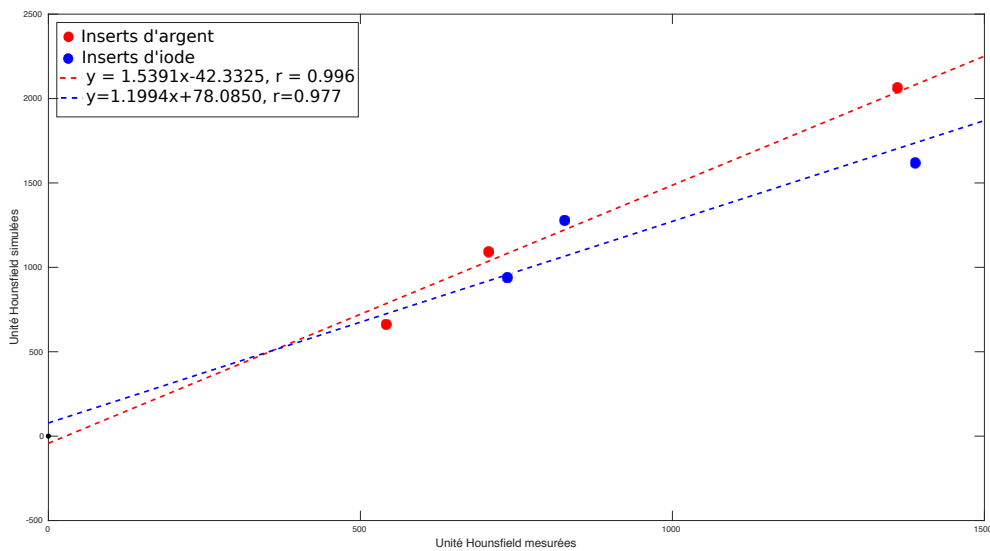


FIGURE 3.66 – Valeurs simulées moyenne en HU en fonction des valeurs réelles mesurées en HU.

séparation spectrale de ce fantôme. Le fantôme a été imagé suivant 6 calibrations différentes :

- au ras-du-bruit, avec un temps d’exposition de 400 ms/image correspondant à une statistique de comptage d’environ 3400 coups dans le fond blanc, mais cette acquisition n’a été utilisée que pour reconstruire le fantôme avec l’algorithme **FDK**,
- à un seuil en énergie estimé de 22 keV, avec un temps d’exposition de 5600 ms/image correspondant à une statistique de comptage d’environ 11000 coups dans le fond blanc,
- au K-edge de l’argent (25,5 keV), avec un temps d’exposition de 5600 ms/image correspondant à une statistique de comptage d’environ 6300 coups dans le fond blanc,
- à un seuil en énergie estimé de 29 keV , avec un temps d’exposition de 5600 ms/image correspondant à une statistique de comptage d’environ 3400 coups dans le fond blanc,
- au K-edge de l’iode (33,2 keV), avec un temps d’exposition de 8500 ms/image correspondant à une statistique de comptage d’environ 2600 coups dans le fond blanc.

- à un seuil en énergie estimé de 36.7 keV, avec un temps d'exposition de 8500 ms/image correspondant à une statistique de comptage d'environ 1200 coups dans le fond blanc.

Comme expliqué au paragraphe 3.3.1.2, nous sommes uniquement capables de calibrer le détecteur à des énergies correspondant à des K-edge d'agents de contraste. Connaissant la configuration de calibration à ces énergies, nous sommes capables de régler le détecteur à d'autres énergies en changeant le pas de **DACL** d'une même quantité pour tous les pixels (nominalement, 1 pas de **DACL** correspond à environ 0.5 keV). Cependant, des études menées au sein de l'équipe imXgam ont montré que les pixels de la caméra XPAD3 n'ont pas tous la même réponse lorsqu'on règle le **DACL**, ce qui conduit à des calibrations de moindre qualité.

Nous avons gardé les mêmes paramètres pour les algorithmes que dans le cas des données simulées. Les paramètres pour l'algorithme **FISTA** sont donc les suivants :

- **initialisation** : à zéro,
- **pas de descente** :

$$\gamma_{3D} = \frac{1}{2 \|\Sigma \mathbf{F} \mathbf{F}^t \Sigma^t\|_{Lop}}, \quad (3.43)$$

où  $L_{op}$  est l'estimation de la norme de l'opérateur  $\mathbf{S}^t \mathbf{S}$  calculée d'après l'équation (3.29),

- **critère d'arrêt** : nombre d'itérations égal à 500.

Les paramètres pour **VMILAn** sont les suivants :

- **initialisation** : à zéro,
- **métrique** : on utilise la métrique  $\mathbf{M}_1$ ,
- **pas de descente** : on multiplie seulement le gradient par le facteur correcteur  $\hat{\alpha} = 1.2$  calculé d'après l'équation (3.30),
- **paramètre de relaxation** : fixé à 1, donc pas de relaxation,
- **critère d'arrêt** : nombre d'itérations égal à 500.

Nous n'avons que l'indicatrice des positifs comme régularisation pour ce test. Le calcul du point proximal se fait à l'aide de l'algorithme itératif **FISTA** en prenant pour paramètres :

- **initialisation** : à l'itéré  $\mathbf{A}_k$ ,
- **pas de descente** : l'inverse de la norme de la métrique  $\mathbf{M}_1$  comme pas de descente du gradient,
- **projection** : l'opérateur de projection sur les positifs consistant à mettre les composantes négatives à zéro,
- **critère d'arrêt** : nombre d'itérations égal à 200 au minimum.

Le bon comportement des deux algorithmes **FISTA** et **VMILAn** est confirmé par les courbes de décroissance des fonctionnelles ainsi que la décroissance des résidus normalisés  $\frac{\|x_{n+1} - x_n\|}{\|x_n\|}$  en figure 3.67. Ces résidus n'ont pas de raison de décroître de manière monotone, il est donc normal d'observer des rebonds.

La fonctionnelle de **VMILAn** est décroissante alors que celle de **FISTA** n'est pas monotone mais semble néanmoins converger malgré la nature non-convexe de notre problème. Les deux fonctionnelles semblent se stabiliser numériquement rapidement à des valeurs proches, mais l'algorithme **VMILAn** atteint une valeur de fonctionnelle plus faible.

Il est délicat d'interpréter cette observation. La fonctionnelle étant non-convexe, les deux algorithmes peuvent prendre des chemins différents pour atteindre un point critique et il n'est pas sûr qu'ils convergent vers le même point. Si toutefois cela devait être le cas, alors cela signifierait que **FISTA** converge sur cet exemple plus lentement que **VMILAn**, mais cette différence de vitesse de convergence est moins marquante que celle observée pour les résultats en données simulées.

Nous avons voulu observer la différence de comportement des algorithmes au cours de la conver-

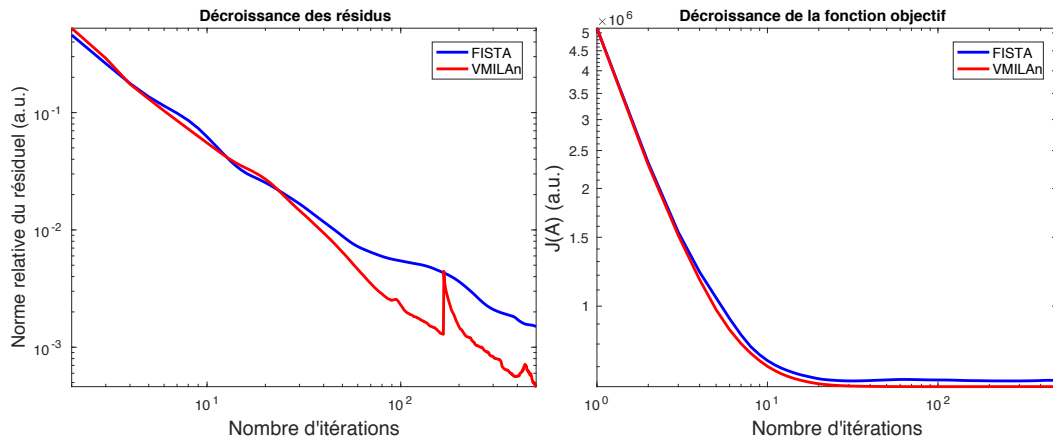


FIGURE 3.67 – Décroissance des résidus (à gauche)  $\|x_{n+1} - x_n\|$  et de la fonctionnelle  $J(\mathbf{A})$  (à droite) en fonction du nombre d'itérations pour les algorithmes **FISTA** et **VMILAn**.

gence. Pour cela, nous avons comparé les résultats finaux pour 500 itérations et les résultats intermédiaires pour 100 itérations. Les reconstructions donnent une séparation satisfaisante au bout de 100 d'itérations pour l'algorithme **VMILAn** comme on peut le constater en figure 3.68. Le composant iode est bien isolé et très contrasté avec son environnement comme on le constate en figure 3.68, mais le contenu des inserts d'iode est impacté par le bruit, ce qui explique la granularité des inserts. Le composant argent est également très bien isolé, même si on observe là aussi la présence de bruit dans les inserts sous forme d'artefacts circulaires. Ce bruit en artefact circulaire est typique d'une mauvaise connaissance de la courbe d'efficacité de chaque pixel du détecteur, qui peut donc être différente d'un pixel à son voisin. Il faut noter que cette observation est la principale source de bruit dans tous les résultats présentés.

D'autre part, même si la courbe de décroissance de l'algorithme **FISTA** est du même ordre de grandeur que celle de l'algorithme **VMILAn**, on constate que l'algorithme **FISTA** n'a absolument pas fini de séparer les composantes entre elles, en particulier du signal de la composante eau est encore fortement présent dans les deux composantes d'agents de contraste. On remarque tout de même que les images sont plus lisses et beaucoup moins impactées par le bruit.

Les résultats ne s'améliorent que très légèrement pour l'algorithme **VMILAn** au terme des 500 itérations comme on le constate en figure 3.69, ce qui montre bien que l'algorithme converge rapidement vers des résultats très intéressants en terme de **localisation** et **d'identification** d'agents de contraste après quelques dizaines d'itérations, et que les objets estimés ne sont que très peu modifiés après une centaine d'itérations. Les reconstructions obtenues avec l'algorithme **FISTA** donnent une meilleure séparation au terme des 500 itérations comparativement aux résultats obtenus pour 100 itérations, particulièrement pour le composant de l'argent comme on le voit en figure 3.69. Des artefacts circulaires restent très présents dans la composante d'iode pour la même raison que celles exposée pour **VMILAn** ci-dessus, *i.e.* la non homogénéité de l'efficacité des pixels.

On note toutefois que les images obtenues avec l'algorithme **FISTA** sont plus lisses et présentent moins de granularité que celles obtenues avec l'algorithme **VMILAn**. La principale explication tient dans la présence du filtre conique dans la métrique utilisée avec l'algorithme **VMILAn** à chaque itération. Même si elle permet d'accélérer la convergence de manière évidente au vu des résultats, son action contribue également à amplifier les hautes fréquences, et donc le bruit. Néanmoins, les séparations obtenues avec ces deux méthodes sont beaucoup moins bruitées que les reconstructions obtenues avec la méthode de séparation au **K-edge**. On remarque toute de même que la séparation obtenue avec cette méthode est sans ambiguïté, comme on le constate en figure 3.69.

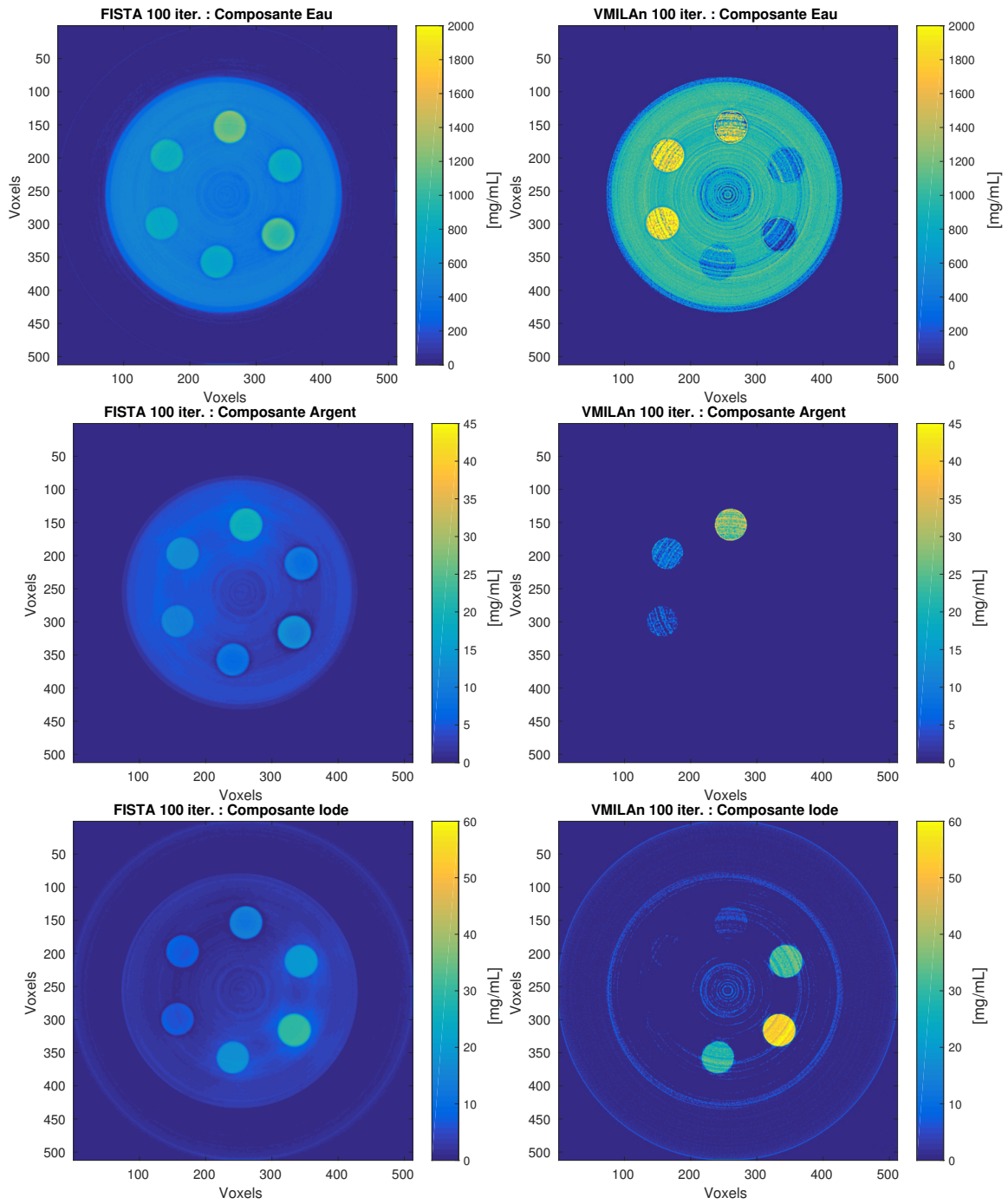


FIGURE 3.68 – Reconstruction ( $512 \times 512 \times 16$  voxels) et séparation par les algorithmes VMILAn (à droite) et FISTA (à gauche) pour 100 itérations sur le fantôme à micro-tubes Eppendorf.

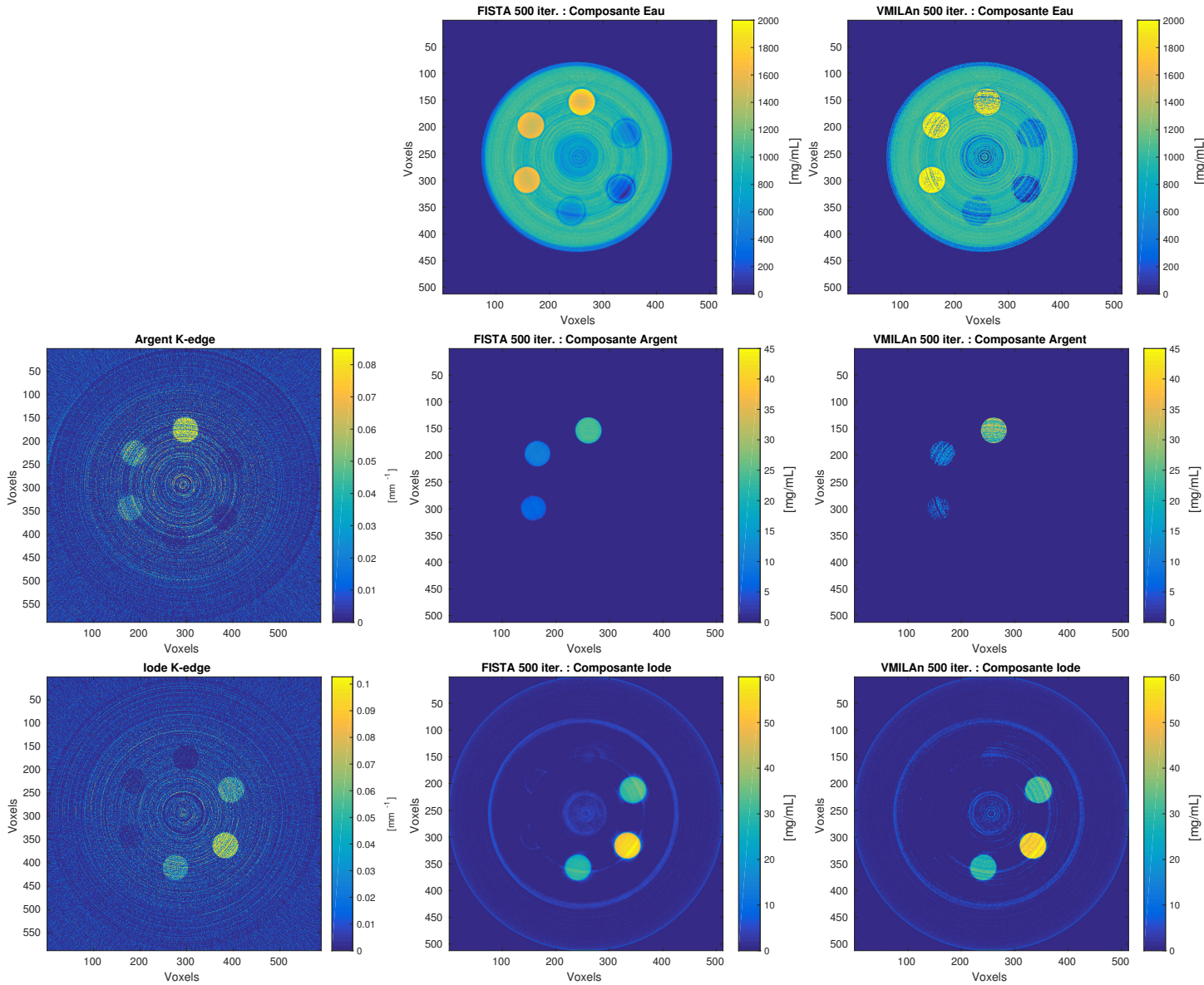


FIGURE 3.69 – Reconstruction ( $512 \times 512 \times 16$  voxels) et séparation par la méthode de séparation au **K-edge** (à gauche), les algorithmes **VMILAn** (à droite) et **FISTA** (au milieu) pour 500 itérations sur le fantôme à micro-tubes Eppendorf.

Pour pousser plus loin l'analyse de la qualité de reconstruction, on peut quantifier le contraste des images. Pour cela on utilise le **CNR** défini dans l'équation (2.50) au chapitre 2 suivante :

$$\text{CNR}_{\text{insert}} = \frac{|C_{\text{insert}} - C_{\text{région}}|}{\sqrt{\sigma_{\text{insert}}^2 + \sigma_{\text{région}}^2}}, \quad (3.44)$$

où  $C_{\text{insert}}$  et  $C_{\text{région}}$  sont les concentrations moyennes mesurées dans l'insert cible et dans la région voisine et  $\sigma_{\text{insert}}$  et  $\sigma_{\text{région}}$  sont les écarts-types respectifs. Nous comparons le **CNR** de chaque insert à celui de l'eau dans une région voisine dans l'image afin d'évaluer la qualité du contraste.

Les **CNR** sont reportées dans les tables 3.1 pour l'argent et 3.2 pour l'iode. On remarque que les **CNR** résultant de l'algorithme **FISTA** sont légèrement plus élevés mais présentent des barres d'erreurs plus grandes également. Ces **CNR** élevés sont le fruit d'une reconstruction lisse et robuste au bruit comme nous l'avons constaté dans les figures 3.68 et 3.69. En effet, les images étant lisses,



l'écart-type au dénominateur du **CNR** devient faible, augmentant ainsi la valeur du **CNR**. Notons par ailleurs que l'algorithme **VMILAn** n'atteint pas des valeurs de **CNR** aussi grande à cause du bruit, mais les barres d'erreur de celles-ci sont très faibles, ce qui montre la stabilité de l'algorithme.

D'autre part, les **CNR** mesurés ne témoignent pas de la réalité du contraste observé sur les images. Nous avons aussi mesuré le contraste pur des inserts en effectuant le calcul suivant :

$$CP_{\text{insert}} = |C_{\text{insert}} - C_{\text{région}}| \quad (3.45)$$

où  $C_{\text{insert}}$  et  $C_{\text{région}}$  sont les concentrations moyennes mesurées dans l'insert cible et dans la région voisine. On remarque que pour 100 itérations, le contraste pur pour l'algorithme **VMILAn** est plus élevé que celui obtenu avec l'algorithme **FISTA** comme on le constate en figure 3.70.

Pour ce qui concerne l'estimation des concentrations, les deux algorithmes semblent retrou-

Concentrations (mg/mL)	Ag 43		
Méthodes	K-edge	FISTA 500	VMILAn 500
CNR	1.245 ± 0.06	7.03 ± 0.69	2.54 ± 0.08
Concentration estimée (mg/mL)	x	23.31 ± 0.37	24.36 ± 0.15
Concentrations (mg/mL)	Ag 21.5		
Méthodes	K-edge	FISTA 500	VMILAn 500
CNR	0.673 ± 0.03	6.84 ± 1.18	1.15 ± 0.03
Concentration estimée (mg/mL)	x	9.91 ± 0.28	7.28 ± 0.07
Concentrations (mg/mL)	Ag 11		
Méthodes	K-edge	FISTA 500	VMILAn 500
CNR	0.634 ± 0.02	7.15 ± 2.11	0.69 ± 0.009
Concentration estimée (mg/mL)	x	6.61 ± 0.28	3.31 ± 0.02

TABLEAU 3.1 – Table d'études des régions d'intérêt pour les différents inserts d'argent pour la méthode du **K-edge** et les algorithmes **FISTA** et **VMILAn**.

Concentrations (mg/mL)	I 40		
Méthodes	K-edge	FISTA 500	VMILAn 500
CNR	1.51 ± 0.05	8.21 ± 0.82	7.63 ± 0.84
Concentration estimée (mg/mL)	x	52.64 ± 0.46	52.87 ± 0.48
Concentrations (mg/mL)	I 30		
Méthodes	K-edge	FISTA 500	VMILAn 500
CNR	1.00 ± 0.03	9.52 ± 1.03	7.1 ± 0.36
Concentration estimée (mg/mL)	x	31.51 ± 0.25	31.58 ± 0.17
Concentrations (mg/mL)	I 20		
Méthodes	K-edge	FISTA 500	VMILAn 500
CNR	0.907 ± 0.02	9.31 ± 0.81	6.86 ± 0.32
Concentration estimée (mg/mL)	x	28.47 ± 0.15	28.49 ± 0.13

TABLEAU 3.2 – Table d'études des régions d'intérêt pour les différents inserts d'iode pour la méthode du **K-edge** et les algorithmes **FISTA** et **VMILAn**.

ver les mêmes valeurs pour les inserts contenant de l'iode. On remarque que les valeurs trouvées pour les inserts I 40, I 30 et I 20 sont environ 52 mg/mL, 31 mg/mL et 28 mg/mL pour les deux algorithmes. Ces valeurs de concentrations correspondent mieux au contraste que l'on observe sur la reconstruction obtenue avec l'algorithme **FDK** en figure 3.64. Par ailleurs, la forte concentration contenue dans l'insert I 40 explique mieux la non linéarité évoquée précédemment (voir figure

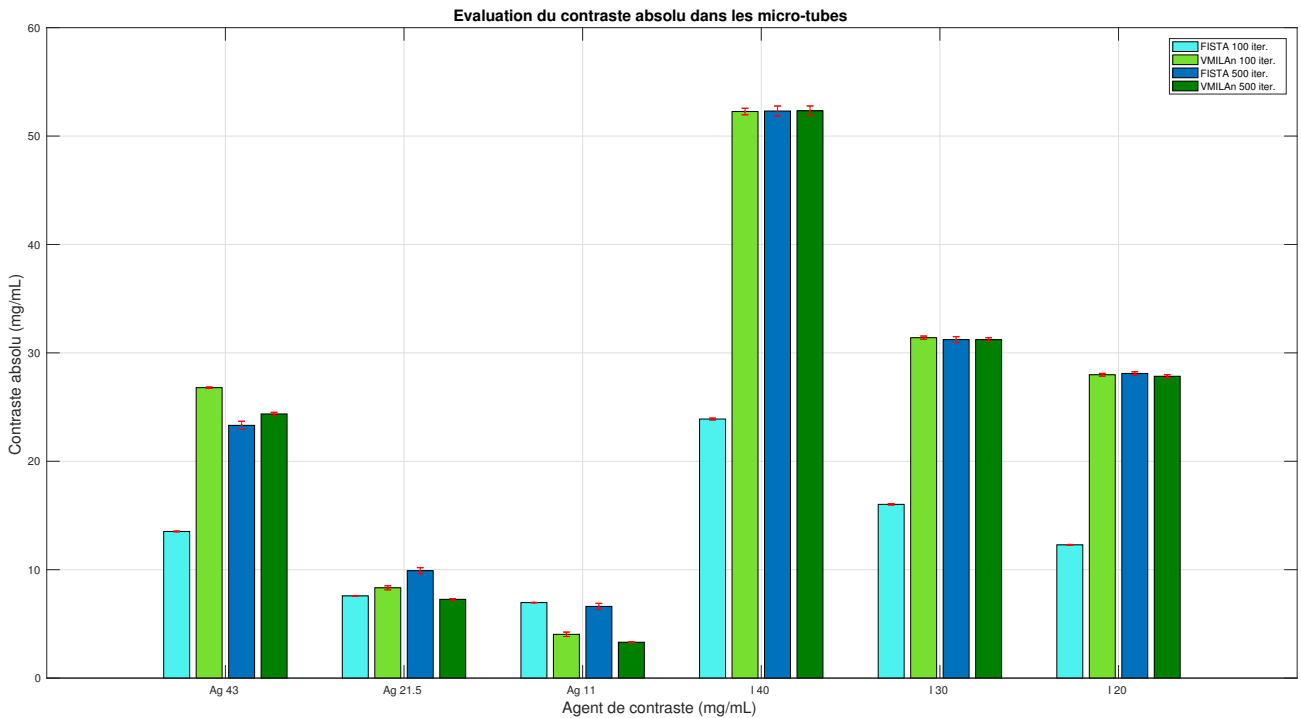


FIGURE 3.70 – Comparaison des contrastes pur des images spectrales calculés pour les différents inserts du fantôme à micro-tubes Eppendorf avec les algorithmes **FISTA** et **VMILAn** (pour 100 et 500 itérations).

3.66).

Si les deux algorithmes concordent pour les concentrations des inserts d'iode, il en va autrement pour les inserts d'argent. En effet, on ne retrouve une estimation similaire que pour l'insert Ag 43, à environ 23.3 mg/mL pour l'algorithme **FISTA** et 24.4 mg/mL pour l'algorithme **VMILAn**. On observe pour les inserts Ag 21.5 et Ag 11 une différence dans les estimations des algorithmes. Cette différence s'explique par la présence de bruit en artefacts circulaires dans ces inserts impactant fortement les reconstructions obtenues avec l'algorithme **VMILAn** comme on l'observe en figure 3.71.

On s'est également intéressé à quantifier la concentration de la quantité d'eau présente dans

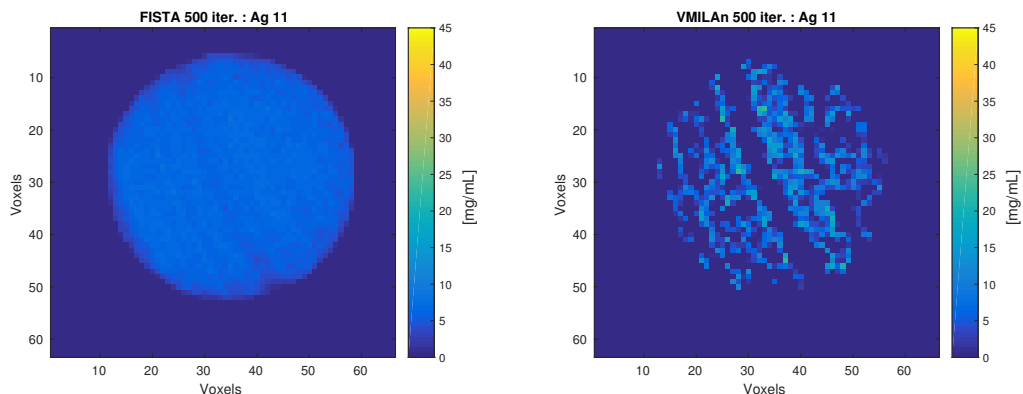


FIGURE 3.71 – Reconstruction ( $512 \times 512 \times 16$  voxels) de l'insert Ag 11 par les algorithmes **VMILAn** (à droite) et **FISTA** (à gauche) pour 500 itérations.

les inserts d'agent de contraste. Ces inserts ont été préparés à partir d'une solution d'eau, c'est pour cela que l'on observe une quantité d'eau présente dans les inserts sur les reconstructions de la composante eau. La figure 3.72 résume ces résultats. On observe que l'évolution de quantité

d'eau décroissante en fonction de la concentration est respectée par l'algorithme **VMILAn** sur les 6 inserts tandis que l'algorithme **FISTA** ne respecte pas cette tendance. De manière générale, les algorithmes ont une meilleure estimation pour les inserts d'iode comme on peut le voir en figure 3.72.

Enfin, nous avons voulu vérifier que les concentrations estimées par les algorithmes **FISTA** et

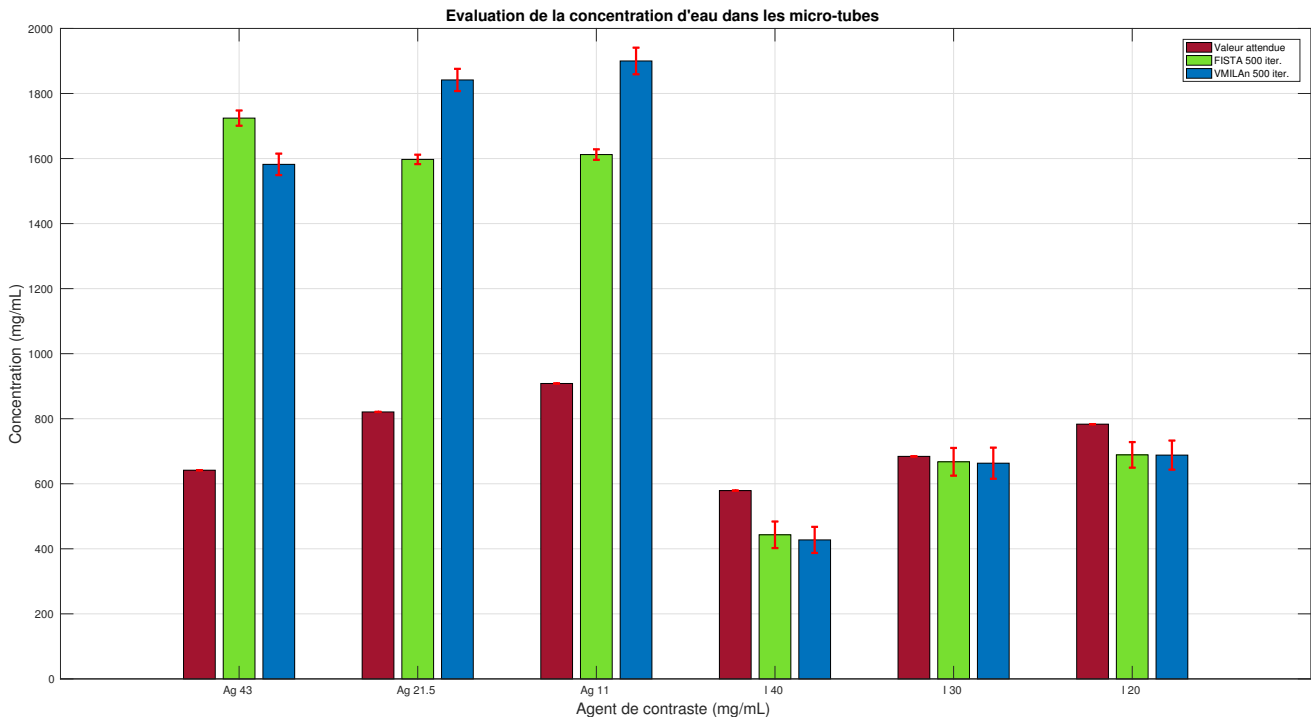


FIGURE 3.72 – Comparaison de la concentrations d'eau dans les inserts d'agent de contraste pour les algorithmes **FISTA** et **VMILAn** pour 500 itérations.

**VMILAn** permettait d'obtenir une linéarité entre les unités Hounsfield mesurées sur la reconstruction **FDK** au ras-du-bruit (de la figure 3.64) et celles mesurées sur la reconstruction **FDK** obtenue à partir d'une simulation d'un fantôme de contraste qui incorpore les concentrations estimées par les deux algorithmes.

Nous avons pour cela simulé un nouveau fantôme de contraste avec les concentrations suivantes en argent (resp. en iode) : 23 mg/mL, 10 mg/mL, 7 mg/mL (resp. 53 mg/mL, 32 mg/mL, 28 mg/mL). Ces concentrations ont été choisies sur la base des résultats de l'algorithme **VMILAn**, excepté pour la plus faible concentration en argent pour laquelle nous avons choisi la valeur suggérée par l'algorithme **FISTA**. Ce choix a été fait pour réduire l'impact du bruit des artefacts circulaires que nous avons observé pour ce micro-tube. La figure 3.73 présente le résultat attendu, à savoir que chaque agent de contraste présente une linéarité marquée (les coefficients de corrélation sont supérieurs à 0.99 dans les deux cas) entre les unités Hounsfield mesurées et les unités Hounsfield simulées.



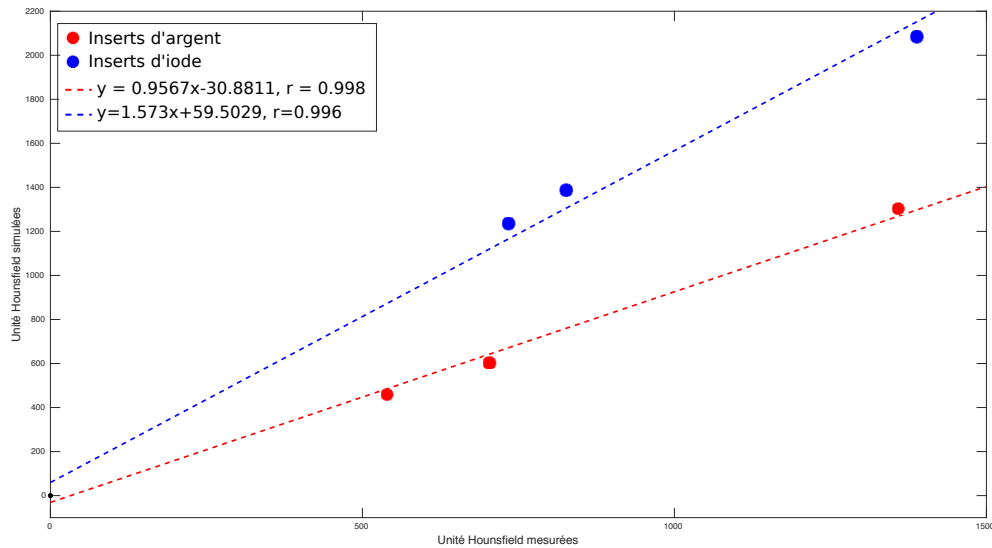


FIGURE 3.73 – Valeurs simulées moyenne en HU à partir des concentrations estimées par les algorithmes FISTA et VMILAn en fonction des valeurs réelles mesurées en HU.

### Changement de la matrice F

Nous avons vu au paragraphe 3.2.3.6 que le changement de filtre peut influencer le résultat de reconstruction. Nous voulons mesurer ici la différence de qualité si on ne tient pas compte du partage de charges. Ce phénomène est dû au fait qu'un photon interagissant dans le Silicium d'un pixel peut déposer une partie de son énergie dans le ou les pixels voisins, et ainsi introduire du bruit de mesure.

Il impacte donc les courbes d'efficacité des pixels du détecteur en fonction de l'énergie et peut induire une diminution de contraste comme cela a été montré dans les résultats de thèse de Loriane Portal [PORTAL, 2018]. Lorsque le partage de charge n'est pas modélisé, on obtient des profils de sources avec une nette discontinuité à l'énergie correspondant au seuil en énergie du pixel, comme on peut l'observer en figure 3.74.

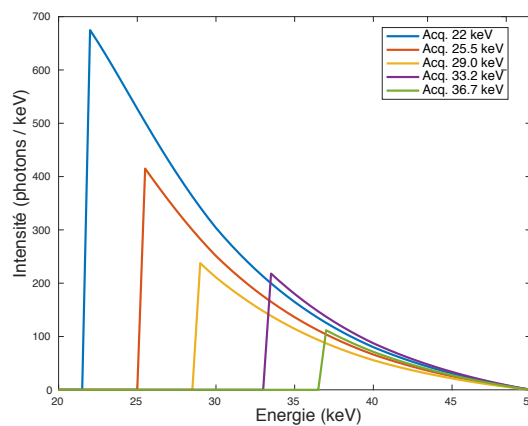


FIGURE 3.74 – Les courbes de réponse des filtres tenant compte de la tension appliquée (50 kV), du filtre métallique (0.6 mm d'aluminium), mais pas du partage de charges.

Concentrations (mg/mL)	Ag 43	
Méthodes	FISTA 500	VMILAn 500
CNR	$6.95 \pm 0.65$	$2.98 \pm 0.09$
Concentration estimée (mg/mL)	$22.13 \pm 0.33$	$23.72 \pm 0.04$
Concentrations (mg/mL)	Ag 21.5	
Méthodes	FISTA 500	VMILAn 500
CNR	$7.22 \pm 1.1$	$1.78 \pm 0.04$
Concentration estimée (mg/mL)	$9.85 \pm 0.3$	$9.65 \pm 0.09$
Concentrations (mg/mL)	Ag 11	
Méthodes	FISTA 500	VMILAn 500
CNR	$7.42 \pm 1.62$	$1.28 \pm 0.03$
Concentration estimée (mg/mL)	$7.03 \pm 0.29$	$6.27 \pm 0.06$

TABEAU 3.3 – Table d’études des régions d’intérêt pour les différents inserts d’argent pour les algorithmes **FISTA** et **VMILAn**.

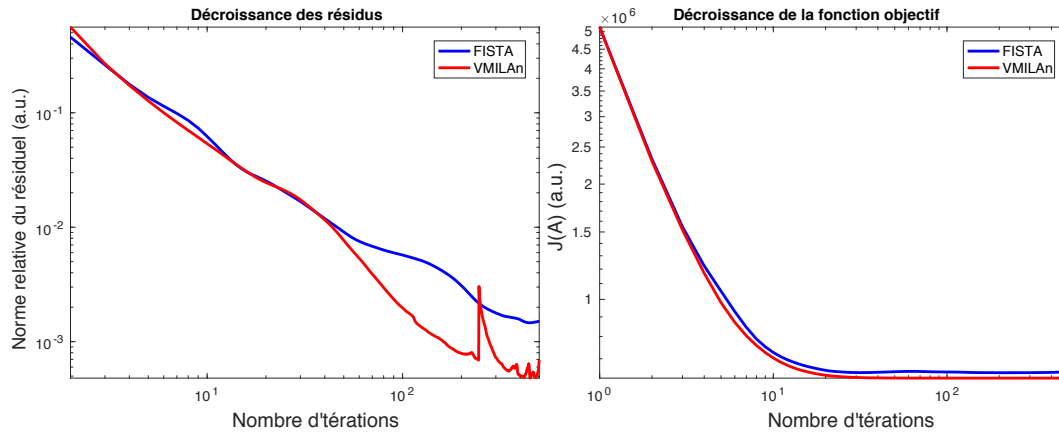


FIGURE 3.75 – Décroissance des résidus (à gauche)  $\|x_{n+1} - x_n\|$  et de la fonctionnelle  $J(\mathbf{A})$  (à droite) en fonction du nombre d’itérations pour les algorithmes **FISTA** et **VMILAn** lorsqu’on ne tient pas compte du partage de charges dans la matrice **F**.

Nous appliquons le même protocole d’expérience précédemment cité et observons des comportements très similaires des courbes de décroissance des résidus et de la fonctionnelle (figure 3.75). On observe également un résultat de reconstruction similaire au précédent mais avec une légère augmentation globale de qualité en terme de **CNR** (une augmentation moyenne d’une unité). On observe surtout cette différence pour les inserts d’argent (figure 3.76) reconstruits par l’algorithme **VMILAn**.

Les artefacts circulaires sont toujours présents dans les inserts, mais ils sont plus atténués que dans le cas précédent comme on le remarque en figure 3.76. De ce fait, les concentrations estimées, qui étaient alors différentes pour les deux algorithmes dans le cas précédent, sont similaires dans ce cas, comme on le voit au tableau 3.3.

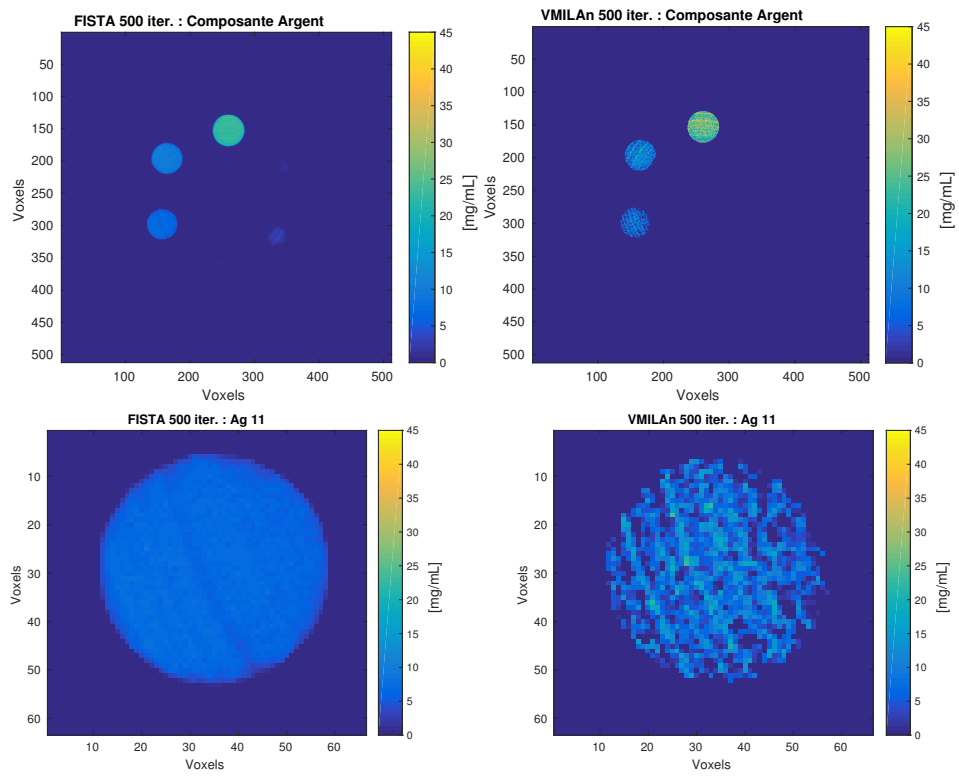


FIGURE 3.76 – Reconstruction ( $512 \times 512 \times 16$  voxels) et séparation de l'argent par les algorithmes **VMILAn** (à droite) et **FISTA** (à gauche) pour 500 itérations.

### 3.3.2.2 Fantôme torsadé

Une seconde expérience a été menée avec un fantôme torsadé formé de trois tubes en caoutchouc de 16 cm de long et de 3 mm de diamètre intérieur enroulés autour d'une potence en fibre de carbone. Ces données sont issues d'une expérience présentée dans [CASSOL et collab., 2013] dont les données ont été acquises sur le prototype PIXSCAN. Ce micro-tomodensitomètre spectral, prédécesseur du prototype PIXSCAN-FLI, est également équipé d'une caméra XPAD3/Si, mais utilise un tubes à anode de molybdène RTW-MCBM 65B-Mo (RTW, Berlin, Allemagne) dont le spectre théorique a été fourni par le constructeur. Chaque tube est rempli du composant suivant :

- un tube contenant de l'iomeron avec une concentration de 95 mg/mL d'iode,
- un tube contenant de sulfate du cuivre ( $\text{CuSO}_4$ ) avec une concentration de 1 mol/L,
- un tube contenant du nitrate d'argent ( $\text{AgNO}_3$ ) avec une concentration de 118 mg/mL d'argent.

Le protocole de test est reporté sur la fiche 3.3.2.2.



FIGURE 3.77 – Reconstruction 3D du fantôme torsadé [CASSOL et collab., 2013].

#### Protocole 3.3.2.2: test sur le fantôme torsadé avec 3 composants

**Objet :** fantôme torsadé contenant de l'iomeron, du nitrate d'argent et du sulfate de cuivre (figure 3.77).

**Taille :**  $D = 512 \times 16 \times 512$  voxels.

**Nombre de projections (angles) :** 360.

**Nombre de composants :**  $K = 3$ .

**Nombre d'acquisitions :**  $M = 3$ .

**Nombre d'échantillons en énergie :**  $N = 43$ .

**Nombre total de mesures :**  $P = 512 \times 32 \times 360 = 10 \times 10^6$ .

**Source  $I_0(E)$  :** spectre d'un tube à anode de molybdène avec une tension fixée à 50 kV.

**Efficacité  $D(E)$  :** l'efficacité des pixels du XPAD3/Si a été simulée par Monte Carlo avec GATE [BRUNNER et collab., 2009; CASSOL et collab., 2015; JAN et collab., 2011].

**Filtre métallique (et sa réponse  $F(E)$ ) :** filtre de cuivre avec une épaisseur de 100  $\mu\text{m}$  (figure 3.78).

**Seuils d'acquisition :**  $s_1 = 21.0$  keV,  $s_2 = 25.5$  keV,  $s_3 = 33.2$  keV.

**Dictionnaire des sections efficaces  $\Sigma$  :** sections efficaces d'interaction (NIST XCOM database<sup>a</sup>) des composants suivants :

- iomeron 400 dilué à une concentration de 95 mg/mL d'iode,
- nitrate d'argent en solution avec une concentration de 120 mg/mL d'argent,
- eau.

a. <https://www.nist.gov/pml/xcom-photon-cross-sections-database>

Nous avons initialisé nos algorithmes exactement de la même manière que pour l'expérience

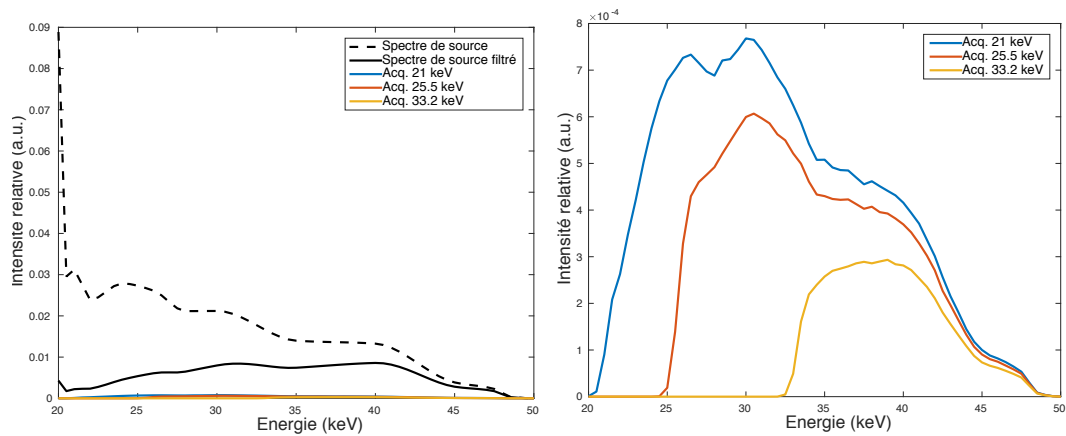


FIGURE 3.78 – A gauche, spectre de source (et sa version filtrée) et réponses de l'efficacité de détection pour les 3 acquisitions utilisées avec une tension appliquée de 50 kV, de l'efficacité des pixels du détecteur estimée et du filtre métallique (100  $\mu\text{m}$  de cuivre). A droite, zoom sur les réponses pour chaque acquisitions.

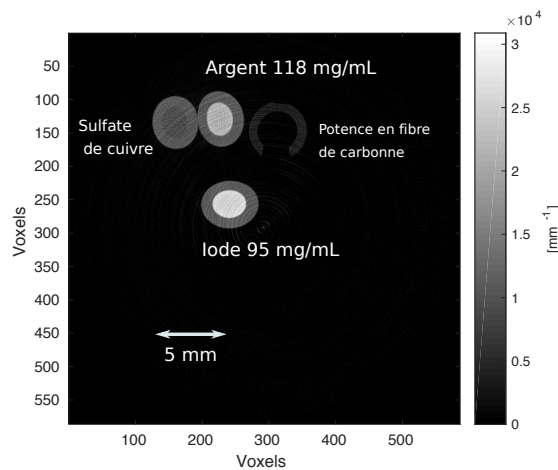


FIGURE 3.79 – Reconstruction par l'algorithme **FDK** de la coupe (582  $\times$  582 voxels) correspondant au plan médian acquise au seuil  $s_1$ .

sur le fantôme à micro-tubes Eppendorf. On reconstruit l'objet à partir d'une acquisition faite au ras-du-bruit à l'aide de l'algorithme **FDK**, qui nous servira de support d'exemple. La figure 3.79 montre la reconstruction du plan médian obtenue à partir de l'algorithme **FDK**. On y voit les différents composants de l'objet, mais on ne peut distinguer l'iode de l'argent.

Le bon comportement des deux algorithmes **FISTA** et **VMILAn** est à nouveau confirmé par les courbes de décroissance des fonctionnelles ainsi que la décroissance des résidus normalisés en figure 3.80. Les deux fonctionnelles semblent se stabiliser numériquement rapidement, mais l'algorithme **FISTA** atteint ici une valeur de fonctionnelle plus faible. Cependant, cette différence de vitesse de convergence ne s'interprète pas au niveau de ce que l'on observe sur les reconstructions.

En effet, les résultats intermédiaires pour 100 itérations montrent que même si la valeur de la fonctionnelle de l'algorithme **FISTA** est plus faible que celle de **VMILAn**, les reconstructions obtenues avec l'algorithme **VMILAn** donnent une séparation qui est plus satisfaisante au bout de 100 itérations. Des ambiguïtés demeurent sur les reconstructions de l'algorithme **FISTA**, comme on peut le constater en figure 3.81.

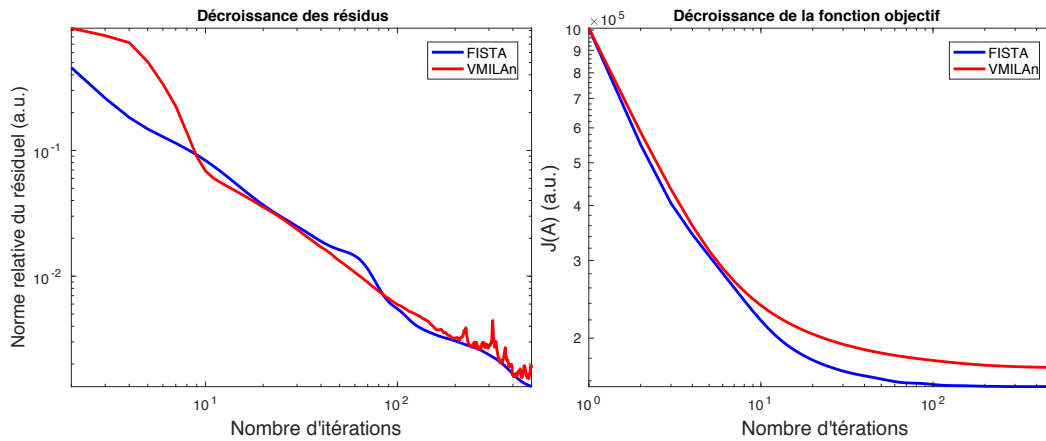


FIGURE 3.80 – Décroissance des résidus (à gauche)  $\|x_{n+1} - x_n\|$  et de la fonctionnelle  $J(\mathbf{A})$  (à droite) en fonction du nombre d'itérations pour les algorithmes **FISTA** et **VMILAn**.

Sur les résultats de **VMILAn**, les agents de contraste sont bien isolés et très contrastés avec leurs environnements, comme on le constate en figure 3.81, mais le contenu du tube d'iode est là aussi impacté par le bruit sous forme d'artefacts circulaires. D'autre part, comme pour les résultats précédents, les images reconstruites par l'algorithme **FISTA** sont plus lisses et beaucoup moins impactées par le bruit.

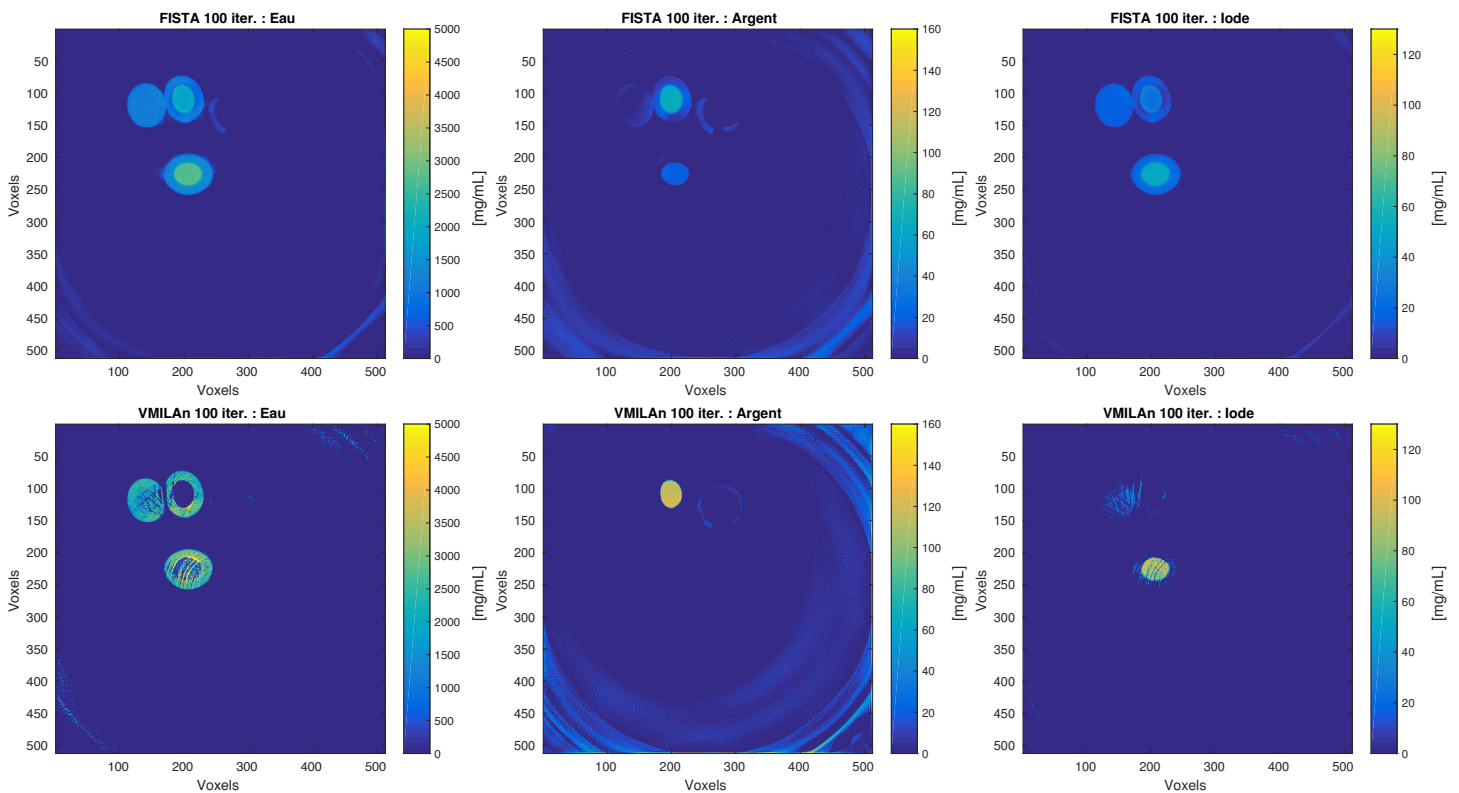


FIGURE 3.81 – Reconstruction ( $512 \times 512 \times 16$  voxels) et séparation par les algorithmes **VMILAn** (en bas) et **FISTA** (en haut) pour 100 itérations sur le fantôme torsadé.

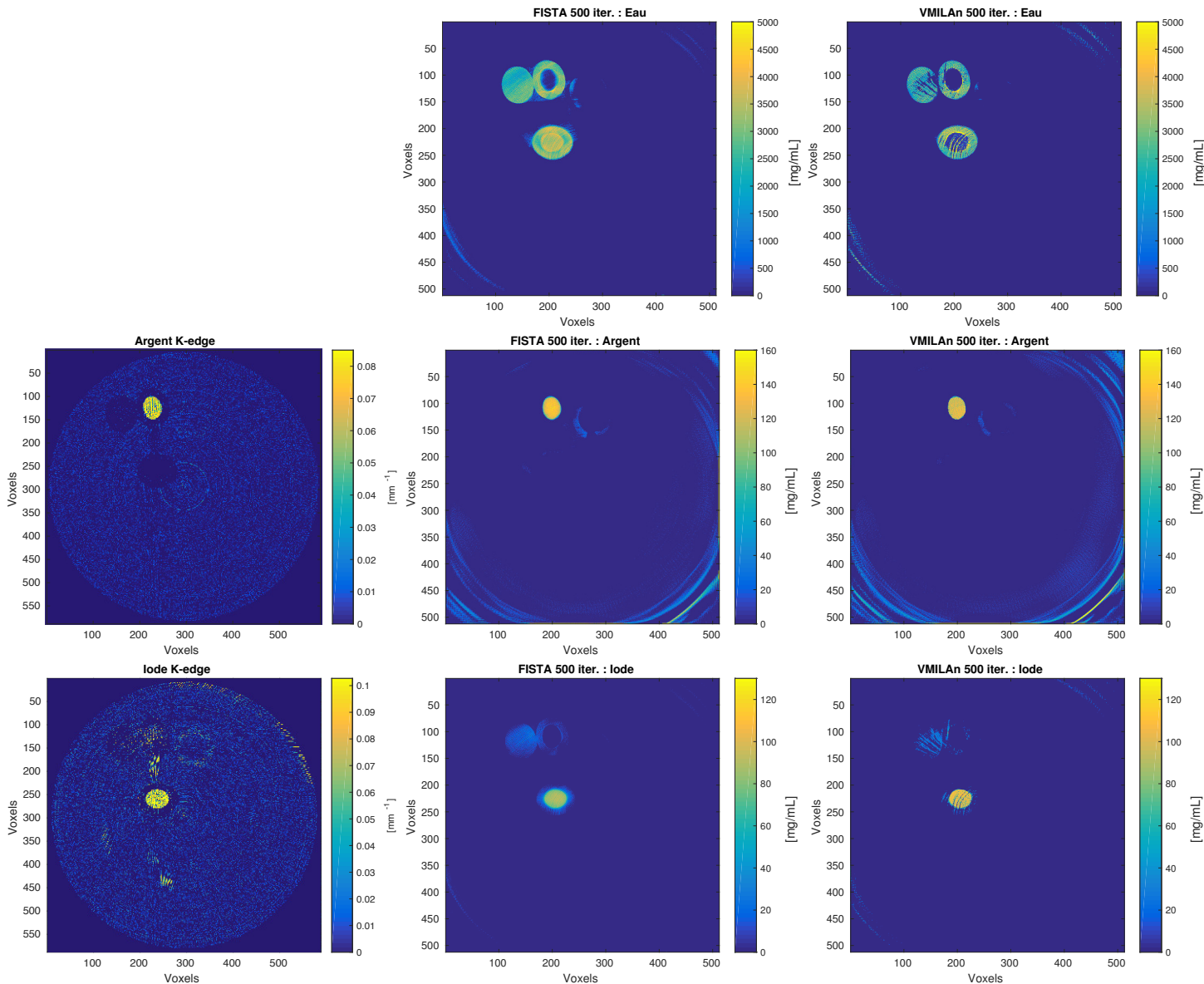


FIGURE 3.82 – Reconstruction ( $512 \times 512 \times 16$  voxels) et séparation par la méthode de séparation au **K-edge** (à gauche), les algorithmes **FISTA** (au milieu) **VMILAn** (à droite) pour 500 itérations sur le fantôme torsadé.

On obtient des résultats analogues à ceux présentés sur le fantôme à micro-tubes Eppendorf pour 500 itérations. En effet, on observe de légères améliorations pour l'algorithme **VMILAn** et une meilleure séparation pour l'algorithme **FISTA** comparativement aux résultats obtenus pour 100 itérations, comme on le constate en figure 3.82. Les artefacts circulaires restent très présents dans la composante d'iode et dans la composante eau contenant le sulfate de cuivre et les parois des tubes en caoutchouc.

D'autre part, les images obtenues avec l'algorithme **FISTA** sont plus lisses et présentent moins de granularité que celles obtenues avec l'algorithme **VMILAn**, mais les contours des objets sur les images sont moins nets, ce qui laisse penser que l'algorithme **FISTA** n'a pas encore convergé. L'algorithme **FISTA** semble reconstruire d'abord les signaux basses fréquences : on obtient des images plus lisses, mais sans détails précis, tandis que le filtre conique dans la métrique utilisée pour l'algorithme **VMILAn** amplifie les hautes fréquences et donne des images plus détaillées, mais bruitées comme on le voit en figure 3.83. Notons toutefois que dans ces résultats, les séparations obtenues avec ces deux méthodes sont beaucoup moins bruitées que les reconstructions obtenues avec la méthode de séparation au **K-edge**, comme on le constate en figure 3.82. On re-

Concentrations (mg/mL)	Ag 118	
Méthodes	FISTA 500	VMILAn 500
CNR	$4.86 \pm 0.74$	$3.71 \pm 0.17$
Concentration estimée (mg/mL)	$107.75 \pm 9.3$	$107.10 \pm 7.52$
Concentrations (mg/mL)	I 95	
Méthodes	FISTA 500	VMILAn 500
CNR	$5.71 \pm 1.35$	$3.28 \pm 0.7$
Concentration estimée (mg/mL)	$66.62 \pm 11.72$	$83.64 \pm 3.6$

TABLEAU 3.4 – Table d’études des régions d’intérêt pour les tubes d’argent et d’iode pour les algorithmes **FISTA** et **VMILAn** sur le fantôme torsadé.

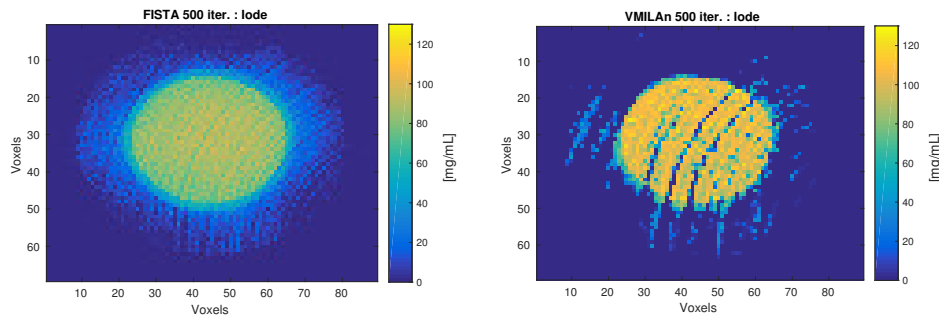


FIGURE 3.83 – Reconstruction ( $512 \times 512 \times 16$  voxels) de l’argent du fantôme torsadé par les algorithmes **FISTA** (à gauche) et **VMILAn** (à droite) et pour 500 itérations.

porte dans la table 3.4 les valeurs des **CNR** et des concentrations obtenues pour les deux algorithmes. Comme supposé, les **CNR** résultant de l’algorithme **FISTA** sont légèrement plus élevés que les **CNR** résultant de l’algorithme **VMILAn**. Notons tout de même que le contraste pur des reconstructions obtenus par l’algorithme **VMILAn** est meilleur que celui obtenu avec l’algorithme **FISTA**, comme on le voit en figure 3.84.

Par ailleurs, les concentrations d’iode retrouvées par l’algorithme **VMILAn** semblent plus proches de celles attendues que l’algorithme **FISTA**. Cela illustre bien le fait que l’algorithme **FISTA** n’a pas fini de converger. Cependant, les deux algorithmes semblent retrouver les mêmes valeurs pour l’argent comme on le constate dans la table 3.4.

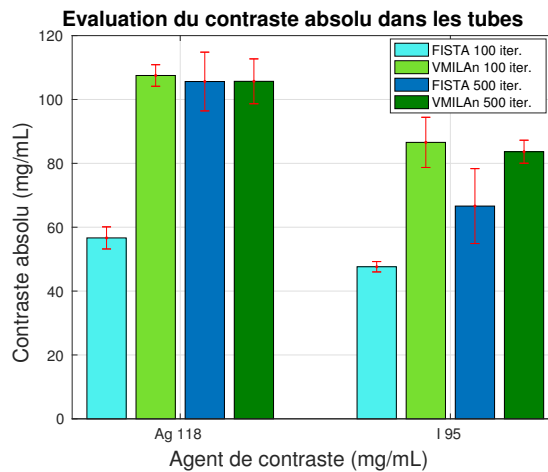


FIGURE 3.84 – Comparaison des contrastes purs des images spectrales calculés pour les différents inserts du fantôme torsadé avec les algorithmes **FISTA** et **VMILAn** (pour 100 et 500 itérations).



## Conclusion

Nous avons présenté dans ce chapitre les méthodes développées au cours de ce travail de thèse ainsi que les résultats obtenus avec ces méthodes.

On rappelle brièvement au début de la section 3.1 le problème de **TDM** spectrale que l'on souhaite résoudre ainsi que les fonctions objectifs à minimiser. Nous montrons que ces fonctions respectent les hypothèses que l'on avait énoncées au chapitre 2, ce qui nous permet donc de pouvoir utiliser les algorithmes présentés dans ce même chapitre. Nous discutons dans la fin de la section 3.1 de la métrique, incorporant le filtre conique, développée spécifiquement pour notre problème de **TDM** spectrale.

On trouve en section 3.2 les tests et les résultats obtenus sur des données simulées. Ces tests sont présentés de manière graduelle en commençant par des tests sur le problème de **TDM** d'absorption classique et de petite dimension jusqu'au problème 3D de **TDM** spectrale 3D à grande dimension. Les résultats de ces tests montrent que les algorithmes présentés au chapitre 2 résolvent bien le problème de séparation et reconstruction simultanées en **TDM** spectrale.

Par ailleurs, les résultats montrent aussi que les méthodes développées sont assez robustes aux différentes perturbations qui peuvent se produire lors d'une acquisition réelle en **TDM** spectrale.

Enfin, la section 3.3 présente les tests et les résultats obtenus sur des données réelles acquises sur le prototype PIXSCAN-FLI et son prédécesseur le prototype PIXSCAN. On présente dans un premier temps la démarche à suivre pour avoir une acquisition **TDM** spectrale sur le prototype PIXSCAN-FLI. Ces étapes, telle que la calibration du détecteur par exemple, sont très importantes pour garantir des données de bonne qualité.

On présente les résultats obtenus sur deux jeux de données de **TDM** spectrale et on montre que nos méthodes sont efficaces pour bien séparer un objet contenant trois composants (eau, iode et argent). Ce chapitre conclut donc le travail de cette thèse et montre bien que le problème de **TDM** spectrale peut être résolu efficacement avec des algorithmes à descente proximale pouvant incorporer des métriques. Nous discuterons en conclusion finale des perspectives d'avenir des méthodes développées durant ce travail de thèse.

### 3.4 Références

- F. ACTON. 1990, *Numerical Methods that Work*, MAA spectrum, Mathematical Association of America, ISBN 9780883854501. URL <https://books.google.fr/books?id=cGnSMGSE5Y4C>. 124
- S. BECKER and J. FADILI. 2012, «A quasi-newton proximal splitting method», *Advances in Neural Information Processing Systems* 25, p. 2618–2626, URL <http://papers.nips.cc/paper/4523-a-quasi-newton-proximal-splitting-method.pdf>. 121
- S. BONETTINI, *et al.* 2017, «On the convergence of a linesearch based proximal-gradient method for nonconvex optimization», *Inverse Probl.*, vol. 33, n° 5, p. 055 005. 125
- A. BONISSENT *et al.* 2009, «A geometrical calibration method for the PIXSCAN micro-CT scanner», *J. Instrum.*, vol. 4, doi :10.1088/1748-0221/4/07/P07016, p. P07 016. URL <http://hal.in2p3.fr/in2p3-00446028>. 166
- J. F. BONNANS, J. C. GILBERT, C. LEMARÉCHAL and C. A. SAGASTIZÁBAL. 1995, «A family of variable metric proximal methods», *Math. Program.*, vol. 68, n° 1, doi :10.1007/BF01585756, p. 15–47, ISSN 1436-4646. URL <https://doi.org/10.1007/BF01585756>. 121
- F. C. BRUNNER, J. C. CLEMENS, C. HEMMER and C. MOREL. 2009, «Imaging performance of the hybrid pixel detectors XPAD3-S», *Phys. Med. Biol.*, vol. 54, n° 6, p. 1773–1789, doi :10.1088/0031-9155/54/6/024. 172, 186
- F. CASSOL *et al.* 2013, «First K-edge imaging with a micro-CT based on the XPAD3 hybrid pixel detector», *IEEE T. Nucl. Sci.*, vol. 60, p. 103–108, doi :10.1109/TNS.2012.2217753. 168, 186
- F. CASSOL *et al.* 2015, «K-edge imaging with the XPAD3 hybrid pixel detector, direct comparison of CdTe and Si sensors», *Phys. Med. Biol.*, vol. 60, n° 14, p. 5497–5511, doi :10.1088/0031-9155/60/14/5497. 172, 173, 186
- G. H.-G. CHEN and R. T. ROCKAFELLAR. 1997, «Convergence rates in forward–backward splitting», *SIAM J. Optimiz.*, vol. 7, n° 2, doi :10.1137/S1052623495290179, p. 421–444, ISSN 1052-6234. URL <http://dx.doi.org/10.1137/S1052623495290179>. 121
- P. L. COMBETTES and B. C. VŪ. 2014, «Variable metric forward–backward splitting with applications to monotone inclusions in duality», *Optimiz.*, vol. 63, n° 9, p. 1289–1318. 121
- DUPONT, M. 2014, «Tomographie spectrale à comptage de photons : développement du prototype PIXSCAN et preuve de concept», thèse de doctorat, Aix Marseille University. URL <http://www.theses.fr/2014AIXM4011/document>. 165
- J. FESSLER. 2017, «Image reconstruction». URL <http://web.eecs.umich.edu/~fessler/book/>. 123
- C. GASQUET and P. WITOMSKI. 2000, «Analyse de Fourier et applications : filtrage, calcul numérique, ondelettes», Sciences sup, Dunod, ISBN 9782100050185. URL <https://books.google.fr/books?id=WeuQAAAACAAJ>. 136
- S. JAN *et al.* 2011, «GATE V6 : a major enhancement of the GATE simulation platform enabling modelling of CT and radiotherapy», *Phys. Med. Biol.*, vol. 56, n° 4, p. 881–901. 172, 186
- L. A. PARENTE, P. A. LOTITO and M. V. SOLODOV. 2008, «A class of inexact variable metric proximal point algorithms», *SIAM J. Optimiz.*, vol. 19, n° 1, p. 240–260. 121
- A. PERELLI, M. A. LEXA, A. CAN and M. E. DAVIES. 2016, «Denoising message passing for X-ray computed tomography reconstruction», *CoRR*, vol. abs/1609.04661. URL <http://arxiv.org/abs/1609.04661>. 123, 131

- L. PORTAL. 2018, «Étude de la tomographie à comptage de rayons X avec des pixels hybrides en Si et en CdTe et application au suivi longitudinal du carcinome hépatocellulaire chez la souris», thèse de doctorat, Aix-Marseille Univ. 147, 167, 169, 172, 174, 183
- S. RAMANI and J. A. FESSLER. 2012, «A splitting-based iterative algorithm for accelerated statistical X-ray CT reconstruction», *IEEE T. Med. Imaging*, vol. 31, n° 3, p. 677–688, ISSN 0278-0062, doi : 10.1109/TMI.2011.2175233. 123
- K. R. RAO, D. N. KIM and J.-J. HWANG. 2010, «Fast Fourier Transform - Algorithms and Applications», 1<sup>re</sup> éd., Springer Publishing Company, Incorporated, ISBN 1402066287, 9781402066283. 136
- S. RIT, *et al.* 2014, «The reconstruction toolkit (RTK), an open-source cone-beam CT reconstruction toolkit based on the insight toolkit (ITK)», *J. Phys.*, vol. 489, n° 1, p. 012 079. URL <http://stacks.iop.org/1742-6596/489/i=1/a=012079>. 125
- R. D. SCHAFER. 1966, «An introduction to nonassociative algebras», Academic Press New York. 119
- L. A. SHEPP and B. F. LOGAN. 1974, «The Fourier reconstruction of a head section», *IEEE T. Nucl. Sci.*, vol. 21, n° 3, p. 21–43, ISSN 0018-9499, doi :10.1109/TNS.1974.6499235. 126
- C. ZHANG *et al.* 2016, «Low-dose CT reconstruction via L1 dictionary learning regularization using iteratively reweighted least-squares», *BioMed. Eng. OnLine*, vol. 15, n° 1, doi :10.1186/s12938-016-0193-y, p. 66. URL <https://doi.org/10.1186/s12938-016-0193-y>. 129



# Conclusion générale

Dans cette thèse, nous avons développé **des méthodes itératives de reconstruction et séparation simultanées de données en TDM spectrale** dans le but d'analyser la structure interne d'un objet. Les principales contributions de ce travail sont axées sur les points suivants :

- identifier des méthodes théoriques capables de résoudre le problème de **TDM spectrale**. En effet, ce problème est un problème inverse mal posé et non linéaire se traduisant en un problème d'optimisation non convexe qui regroupe deux sous-problèmes : un problème de reconstruction tomographique et un problème de séparation non aveugle de sources,
- adapter ces méthodes pour le traitement de données acquises en **TDM spectrale** et les tester sur des données simulées et sur des données réelles.

Pour atteindre ces objectifs, nous avons commencé par l'analyse de la physique permettant de décrire la tomographie par transmission de rayons X à partir de la loi d'atténuation de Beer-Lambert. Celle-ci fait apparaître un **coefficient d'atténuation linéique**, qui dépend à la fois de l'énergie des photons et de la composition atomique du milieu traversé. Le modèle d'acquisition prend en compte ceci afin de mieux séparer les composants de l'objet de la manière suivante :

$$Y = e^{-SA\Sigma F} \quad (3.46)$$

où

- $Y \in \mathbb{R}^{P \times M}$  est la **matrice de données**,
- $F \in \mathbb{R}^{N \times M}$  est le **dictionnaire de filtrage en énergie**,
- $S \in \mathbb{R}^{P \times D}$  est l'**opérateur de la transformée** à rayons X,
- $A \in \mathbb{R}^{D \times K}$  est la **matrice des cartes des concentrations** que l'on souhaite reconstruire,
- $\Sigma \in \mathbb{R}^{K \times N}$  est le **dictionnaire des sections efficaces** d'interaction.

Ce modèle est non-linéaire, ce qui induit un problème inverse mal posé se traduisant en un problème d'optimisation non convexe et de grande dimension. Afin d'augmenter la qualité des résultats de reconstruction, nous avons employé des contraintes décrivant les informations *a priori* dont on dispose. Ces contraintes s'écrivent sous forme de fonctions de régularisation dans le modèle, qui, bien qu'importantes, complexifient l'obtention d'une solution de par leur caractère non-différentiable.

Nous avons vu que pour résoudre ce type de problème, une simple descente de gradient n'est pas suffisante. Nous avons donc introduit les notions de **sous-différentiabilité** et d'**opérateur proximal**, qui permettent de minimiser une fonction même lorsque celle-ci est non-différentiable. L'autre outil important sur lequel nous avons fondé notre travail est l'**inégalité de Kurdyka-Łojasiewicz**, qui donne une garantie de solution locale même lorsque le problème est non-convexe. Ces outils nous ont permis d'introduire les algorithmes proximaux et leur version accélérée incorporant une **métrique**, variable ou non, qui permettent une convergence rapide.

Nous avons adapté ces méthodes pour résoudre le problème de **TDM spectrale** en construisant une métrique spécifique à partir de la hessienne de la fonction d'attache aux données associée

au problème de **TDM** spectrale. Les tests ont été menés en deux temps. Dans un premier temps, nous avons évalué nos méthodes sur des données simulées afin d'avoir une idée de la qualité des reconstructions accessibles pour ce type de données et de comparer ces résultats à ceux des méthodes classiques de l'état de l'art. Nous avons effectué ces tests d'abord avec des données en deux dimensions, puis avec des données en trois dimensions en intégrant de manière progressive les contraintes physiques relatives à la **TDM** spectrale. Les résultats obtenus sur les données simulées ont montré que nos méthodes convergent vers une solution de bonne qualité dans tous les cas étudiés et qu'ils sont de plus robustes face à la plupart des perturbations qui peuvent survenir lors d'une acquisition.

Cette étude a permis d'être mieux préparé au passage des données simulées vers les données réelles. Ces dernières obtenues pour un fantôme à micro-tubes Eppendorf ont clairement montré que les algorithmes à descente proximale **FISTA** et **VMILAn** que nous avons développés convergent vers une solution. En effet, on obtient une séparation et une reconstruction simultanée à l'aide des méthodes employées et cela avec un faible nombre d'itérations des algorithmes, ce qui constituait deux objectifs fondamentaux de cette thèse. De plus, l'analyse qualitative et quantitative de nos résultats a montré que les reconstructions obtenues sont meilleures que celles obtenues avec la méthode de soustraction au **K-edge** et que la quantification des constituants du fantôme était plutôt fidèle à la réalité. Ces résultats sur données réelles sont donc le point fort de cette thèse.

De nombreuses pistes d'amélioration sont possibles directement dans la continuité de ce travail de thèse. Dans un premier temps, l'adaptation de la **métrique** aux données réelles. La métrique est le paramètre clé qui permet d'accélérer la convergence d'un algorithme à descente proximale. Le choix de cette métrique est crucial pour l'efficacité de la méthode. Celle que nous avons utilisée incorpore le **filtre conique**. Il est efficace, car il permet d'éviter de lourds calculs d'inversions de matrices, mais est sensible au bruit, ce qui peut diminuer la qualité des résultats obtenus comme nous l'avons constaté sur les résultats présentés au chapitre 3. Une solution que l'on a utilisée est l'apodisation de ce filtre avec une fenêtre de **Hamming**, mais d'autres fenêtres peuvent être plus performantes.

Par ailleurs, nous avons fait le choix de conserver une métrique constante. La variabilité de la métrique au cours des itérations peut contribuer à accélérer la convergence. On peut par exemple utiliser l'approximation de Gauss-Newton de la hessienne relative au problème de reconstruction. Il faut bien évidemment pouvoir inverser cette métrique à chaque itération de l'algorithme, ce qui constitue une véritable problématique au vu de la grande dimension des données.

Un autre point essentiel à la qualité de reconstruction est le choix de la **régularisation**. La contrainte utilisée le plus souvent lors des tests est la contrainte naturelle de **positivité** induite par la physique. Cette contrainte donne des résultats corrects en terme de qualité, mais ne peut résoudre le problème d'amplification de bruit.

L'autre régularisation utilisée est la norme  $l_1$  qui est une contrainte de **parcimonie**. Nous avons obtenu des résultats similaires à la contrainte de positivité, mais avec l'effort supplémentaire de devoir trouver de manière empirique les bons hyperparamètres de régularisation. Adapter la contrainte de parcimonie au problème de la **TDM** spectrale en utilisant un algorithme proximal à métrique variable serait donc intéressant pour la suite.

Une autre régularisation assez utilisée en traitement d'image et qui permet d'obtenir de bons résultats en terme de **SNR** est la **variation totale (TV)**. Cette régularisation a été abandonnée durant ce travail de thèse, car son incorporation dans les algorithmes proximaux à métrique a été difficile.

Ce travail a permis aussi d'établir l'importance de la connaissance des **filtres** et de l'**efficacité** des pixels du détecteur. En effet, nous avons vu au chapitre 3 que la mauvaise estimation de cette matrice entraîne des ambiguïtés et une séparation partielle des composants de l'objet. Par ailleurs, la mauvaise **calibration** des pixels du détecteur entraîne une modification du support des filtres et

peut impacter les résultats.

Une idée pour améliorer cela est d'utiliser la méthode développée dans [Sidky, 2017], qui consiste à introduire un paramètre supplémentaire permettant d'estimer cette matrice au cours des itérations. D'autre part, la méthode employée au CPPM pour calibrer les pixels du détecteur repose essentiellement sur la vérification manuelle de l'uniformité de la calibration des pixels, ce qui représente un temps de travail assez long. Il serait donc intéressant d'élaborer une stratégie entièrement automatisée et qui permettrait une calibration robuste des pixels du détecteur.

Pour terminer, on peut relever encore deux points importants à discuter à propos des méthodes développées dans ce travail de thèse :

- **L'applicabilité clinique** : la première question que l'on peut se poser est l'applicabilité de la méthode dans le domaine clinique. Comme nous l'avons mentionné plus haut, le problème de **TDM** spectrale est en partie un problème de séparation de sources. Pour avoir une reconstruction de bonne qualité, il faut avoir au moins autant d'acquisitions que de composants à séparer. Or effectuer plusieurs acquisitions peut s'avérer nocif pour la santé de l'homme. Une des solutions à ce problème est l'acquisition en **pixels composites** [Kronland-Martinet, 2015, Portal, 2018], une méthode basée sur l'interpolation et qui permet d'avoir une séparation, mais reste encore très sensible au bruit.
- **L'applicabilité numérique** : les algorithmes de reconstruction utilisés dans le domaine clinique permettent une reconstruction presque instantanée tandis que les méthodes que l'on présente ne sont efficaces qu'après plusieurs itérations. Donc lorsqu'il s'agit de traiter des données de très grande taille, le temps de reconstruction peut devenir très important. La question d'optimisation algorithmique se pose donc afin de diminuer ce potentiel temps de calcul.

Ces deux points peuvent être les principaux facteurs freinant le transfert de la technologie vers la clinique. Il est certain que ces problématiques doivent être abordés dans le futur proche. Cependant, avec l'amélioration constante des puissances de calcul, gageons que les algorithmes à descente proximale offrent des perspectives très intéressantes et robustes au bruit pour les applications clinique et préclinique en **TDM** spectrale.





# Annexe A

## Annexes

### Sommaire

---

<b>A.1 Démonstrations</b> . . . . .	<b>II</b>
A.1.1 Coupe centrale . . . . .	II
A.1.2 Rétroprojection filtrée . . . . .	II
A.1.3 Rétroprojection filtrée de faisceau en éventail . . . . .	IV
A.1.4 Démonstration sous problème de minimisation en sous-espace à deux di- mensions . . . . .	VIII
A.1.5 Démonstration passage au proximal ISTA . . . . .	IX
A.1.6 Démonstration $G_1(A)$ analytique réelle pour le cas gaussien . . . . .	IX
A.1.7 Démonstration $G_2(A)$ analytique réelle pour le cas Poisson . . . . .	X
A.1.8 Calcul du gradient et de la hessienne de $G(A)$ pour le cas Gaussien . . . . .	XI
A.1.9 Calcul du gradient et de la hessienne de $G(A)$ pour le cas Poisson . . . . .	XV
A.1.10 Démonstration filtre conique . . . . .	XVIII
<b>A.2 Références</b> . . . . .	<b>XXII</b>

---

## A.1 Démonstrations

### A.1.1 Coupe centrale

#### Théorème A.1.1.1: Coupe centrale

La transformée de Fourier de la projection  $p_\theta$  vérifie  $\forall \theta \in [0, \pi[$  et  $\forall \xi \in \mathbb{R}$

$$\mathcal{F}[p_\theta](\xi) = \mathcal{F}[f](\xi \cos \theta, \xi \sin \theta), \quad (\text{A.1})$$

où  $\mathcal{F}$  est l'opérateur de transformée de Fourier.

#### Preuve

Pour tout  $\theta$  fixé,  $p_\theta$  est dans  $L^2(\mathbb{R})$  et sa transformée de Fourier est donnée par :

$$\mathcal{F}[p_\theta](\xi) = \int_{\mathbb{R}} p_\theta(t) e^{-it\xi} dt.$$

En remplaçant  $p_\theta$  par sa définition 1.22 on obtient :

$$\mathcal{F}[p_\theta](\xi) = \int_{\mathbb{R}} \left( \int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy \right) e^{-it\xi} dt.$$

On utilise le théorème de Fubini pour échanger l'ordre des intégrales :

$$\mathcal{F}[p_\theta](\xi) = \int \int_{\mathbb{R}} \left( \int_{\mathbb{R}} f(x, y) \delta(x \cos \theta + y \sin \theta - t) e^{-it\xi} dt \right) dx dy.$$

Or la fonction de dirac  $\delta$  est définie comme suit

$$\delta(x) = \begin{cases} +\infty & \text{si } x = 0, \\ 0 & \text{sinon,} \end{cases}$$

donc le terme dans l'intégrale  $\int_{\mathbb{R}} f(x, y) \delta(x \cos \theta + y \sin \theta - t) e^{-it\xi} dt$  est nul partout sauf au point  $t = x \cos \theta + y \sin \theta$ , d'où :

$$\mathcal{F}[p_\theta](\xi) = \int \int_{\mathbb{R}} f(x, y) e^{-i\xi(x \cos \theta + y \sin \theta)} dx dy = \mathcal{F}[f](\xi \cos \theta, \xi \sin \theta)$$

■

### A.1.2 Rétroprojection filtrée

#### Théorème A.1.2.1: Rétroprojection filtrée

Soit  $p_\theta(t)$  la projection de  $f$  au point  $t$  et à l'angle  $\theta$ . Alors  $\forall (x, y) \in \mathbb{R}^2$

$$f(x, y) = \int_0^\pi (p_\theta \star h)(x \cos \theta + y \sin \theta) d\theta \quad (\text{A.2})$$

où  $\star$  dénote le produit de convolution et  $h$  est tel que  $\hat{h}(\xi) = |\xi|$ ,  $\forall \xi \in \mathbb{R}$ .  $h$  est appelé le **filtre rampe**.

**Preuve**

On rappelle que la transformée de Fourier inverse pour le cas bidimensionnel est :

$$f(x, y) = \int_{\mathbb{R}^2} \hat{f}(\boldsymbol{\omega}) e^{i\langle \boldsymbol{\omega}, (x, y) \rangle} d\boldsymbol{\omega} = \int_{\mathbb{R}} \int_{\mathbb{R}} \hat{f}(\omega) e^{i(\omega_1 x + \omega_2 y)} d\omega_1 d\omega_2.$$

Pour passer en coordonnées polaires on pose  $(\omega_1, \omega_2) = (\xi \cos \theta, \xi \sin \theta)$ , en appliquant ce changement de variables et en remarquant que  $\langle \boldsymbol{\omega}, (x, y) \rangle = \xi(x \cos \theta + y \sin \theta)$  on obtient :

$$f(x, y) = \int_0^{+\infty} \int_0^{2\pi} \hat{f}(\xi \cos \theta, \xi \sin \theta) e^{i\xi(x \cos \theta + y \sin \theta)} \xi d\theta d\xi.$$

On a montré précédemment que  $\hat{p}_\theta(\xi) = \hat{f}(\xi \cos \theta, \xi \sin \theta)$ , on remplace et on obtient :

$$f(x, y) = \int_0^{+\infty} \left( \int_0^{2\pi} \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta \right) \xi d\theta d\xi. \quad (\text{A.3})$$

On décompose alors l'intégrale entre parenthèses :

$$\int_0^{2\pi} \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta = \int_0^\pi \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta + \int_\pi^{2\pi} \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta,$$

et on applique le changement de variable  $\theta' = \theta - \pi$  à la seconde partie, on obtient alors :

$$\int_0^{2\pi} \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta = \int_0^\pi \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta + \int_0^\pi \hat{p}_{\theta'+\pi}(\xi) e^{i\xi(x \cos(\theta'+\pi) + y \sin(\theta'+\pi))} d\theta'.$$

Par commodité, nous continuerons par écrire  $\theta$  au lieu de  $\theta'$ . En se servant du fait que  $\cos(x + \pi) = -\cos(x)$  et  $\sin(x + \pi) = -\sin(x)$  on trouve que  $\hat{p}_{\theta+\pi}(\xi) = \hat{p}_\theta(-\xi)$ , de même que  $x \cos(\theta + \pi) + y \sin(\theta + \pi) = -(x \cos \theta + y \sin \theta)$ . Avec cela l'intégrale précédente devient :

$$\int_0^{2\pi} \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta = \int_0^\pi \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\theta + \int_0^\pi \hat{p}_\theta(-\xi) e^{i(-\xi)(x \cos \theta + y \sin \theta)} d\theta.$$

On réinjecte ce résultat dans (A.3) puis on utilise le théorème de Fubini pour inverser les intégrales, on obtient alors :

$$f(x, y) = \int_0^\pi \left( \int_0^{+\infty} \xi \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\xi + \int_0^{+\infty} \xi \hat{p}_\theta(-\xi) e^{i(-\xi)(x \cos \theta + y \sin \theta)} d\xi \right) d\theta.$$

On applique le changement de variable  $\xi \rightarrow -\xi$  à la seconde partie :

$$f(x, y) = \int_0^\pi \left( \int_0^{+\infty} \xi \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\xi - \int_0^{-\infty} (-\xi) \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\xi \right) d\theta,$$

on obtient alors :

$$f(x, y) = \int_0^\pi \left( \int_{-\infty}^{+\infty} |\xi| \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\xi \right) d\theta. \quad (\text{A.4})$$

Pour la suite, on pose  $\mathcal{F}$  l'opérateur de la transformée de Fourier et son inverse  $\mathcal{F}^{-1}$ . On remarque que  $\int_{-\infty}^{+\infty} |\xi| \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\xi$  est la transformée de Fourier inverse de la fonction  $|\xi| \hat{p}_\theta(\xi)$  évaluée au point  $(x \cos \theta + y \sin \theta)$ . On peut donc écrire que :

$$\int_{-\infty}^{+\infty} |\xi| \hat{p}_\theta(\xi) e^{i\xi(x \cos \theta + y \sin \theta)} d\xi = \mathcal{F}^{-1} [|\cdot| \hat{p}_\theta(\cdot)] (x \cos \theta + y \sin \theta). \quad (\text{A.5})$$

Par ailleurs, on sait que la transformée de Fourier d'un produit de convolution de deux fonctions est égale au produit des transformées de Fourier de ces dernières. Ici on a le produit de  $\hat{p}_\theta(\cdot)$  avec la fonction  $|\cdot|$ , on peut donc écrire :

$$|\cdot| \hat{p}_\theta(\cdot) = \mathcal{F} \left[ \mathcal{F}^{-1} [|\cdot|] \star p_\theta(\cdot) \right],$$

ou encore

$$\mathcal{F}^{-1}[|\cdot|\hat{p}_\theta(\cdot)] = \mathcal{F}^{-1}[|\cdot|] \star p_\theta(\cdot).$$

En réinjectant ce résultat (A.5) dans on obtient :

$$\int_{-\infty}^{+\infty} |\xi| \hat{p}_\theta(\xi) e^{i\xi(x\cos\theta+y\sin\theta)} d\xi = (\mathcal{F}^{-1}[|\cdot|] \star p_\theta(\cdot))(x\cos\theta+y\sin\theta).$$

On pose alors  $h = \mathcal{F}^{-1}[|\cdot|]$ , autrement dit  $h$  est la transformée de Fourier inverse du filtre rampe. On obtient le résultat final en insérant le résultat précédent dans (A.4) :

$$f(x, y) = \int_0^\pi p_\theta \star h(x\cos\theta+y\sin\theta) d\theta,$$

où  $\hat{h}(\xi) = |\xi| \forall \xi \in \mathbb{R}$ .

■

### A.1.3 Rétroprojection filtrée de faisceau en éventail

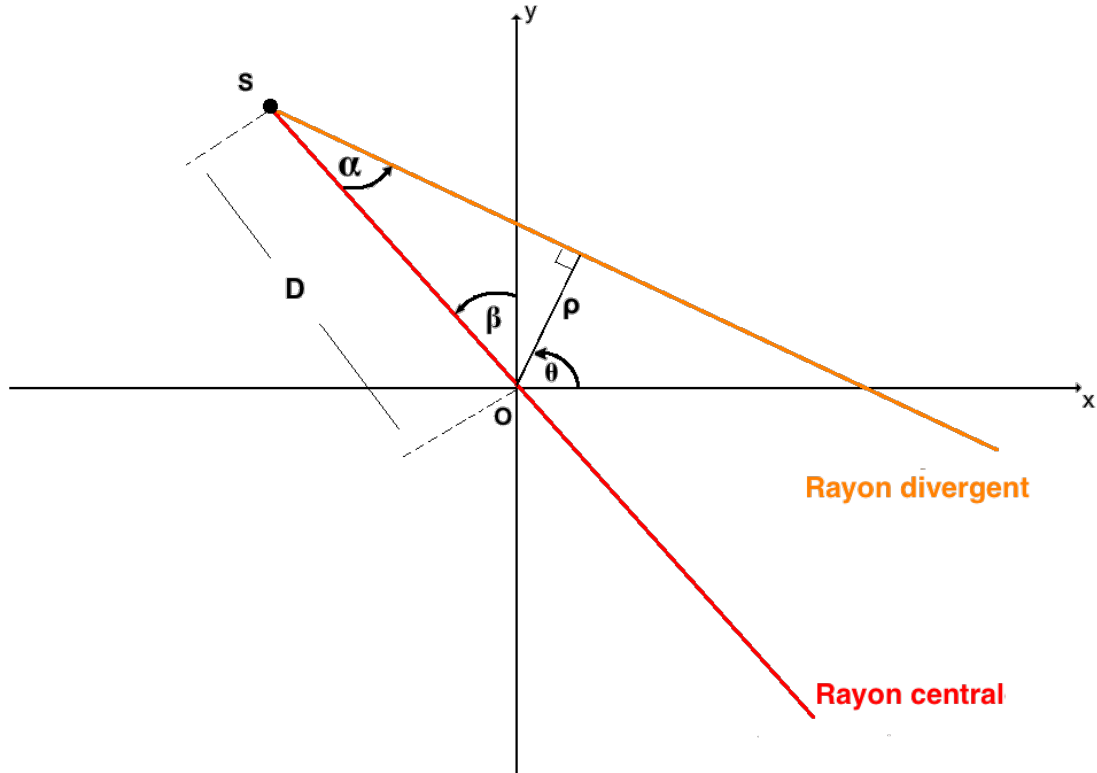


FIGURE A.1 – Illustration de la géométrie en éventail. En rouge le rayon central qui passe par le centre du détecteur. Le rayon orange est le rayon divergent qui forme un angle  $\alpha$  avec le rayon central. C'est ce rayon divergent que l'on souhaite caractériser.

Nous allons raisonner à partir de la figure A.1. On note  $D$  la distance source-origine [SO], c'est le rayon du cercle que décrit la source en rotation autour de l'objet. On prend alors un rayon quelconque depuis la source. On peut représenter ce rayon de plusieurs manière comme par exemple ici en coordonnées polaires  $(\rho, \theta)$ . On peut aussi choisir de le représenter par rapport au rayon central. En effet, ceci est l'idée de base de la reconstruction pour la géométrie divergente : tenir compte des changements géométriques par rapport au rayon central afin d'avoir une formule similaire à la rétroprojection filtrée en faisceau parallèle. Pour cela on considère deux angles  $\beta$  et  $\alpha$ .  $\beta$  est l'angle de rotation que décrit la droite (SO) par rapport à l'axe des ordonnées  $y$ .  $\alpha$  est l'angle

entre le rayon central et le rayon divergent étudié.

Sur la figure A.1 on peut voir qu'il existe une relation de passage entre ces deux représentations. En effet, le côté de longueur  $\rho$  est opposé à l'angle  $\alpha$  dans un triangle rectangle pour écrire

$$\rho = D \sin(\alpha). \quad (\text{A.6})$$

D'autre part, par complémentarité des angles on a

$$\theta = \beta + \alpha. \quad (\text{A.7})$$

L'idée maintenant est d'adapter la formule de rétroprojection filtrée en géométrie parallèle à notre situation, et pour cela nous utilisons la formule de reconstruction (A.2)

$$f(x, y) = \int_0^\pi (p_\theta \star h)(x \cos \theta + y \sin \theta) d\theta, \quad (\text{A.8})$$

en réécrivant le produit de convolution comme suit

$$f(x, y) = \int_0^\pi \int_{-\infty}^{+\infty} p_\theta(\rho) h(x \cos \theta + y \sin \theta - \rho) d\rho d\theta. \quad (\text{A.9})$$

Montrons maintenant l'égalité suivante

$$\int_\pi^{2\pi} \int_{-\infty}^{+\infty} p_\theta(\rho) h(x \cos \theta + y \sin \theta - \rho) d\rho d\theta = \int_0^\pi \int_{-\infty}^{+\infty} p_\theta(\rho) h(x \cos \theta + y \sin \theta - \rho) d\rho d\theta. \quad (\text{A.10})$$

On applique dans un premier temps le changement de variable  $\theta' = \theta - \pi$ , on obtient alors :

$$\int_0^\pi \int_{-\infty}^{+\infty} p_{\theta'+\pi}(\rho) h(x \cos(\theta' + \pi) + y \sin(\theta' + \pi) - \rho) d\rho d\theta'. \quad (\text{A.11})$$

Par commodité, nous continuerons par écrire  $\theta$  au lieu de  $\theta'$ . En se servant du fait que  $\cos(x + \pi) = -\cos(x)$  et  $\sin(x + \pi) = -\sin(x)$ , nous allons simplifier les expressions dans l'intégrale précédente. On rappelle dans un premier temps que d'après le lemme (1.23), la projection  $p_{\theta+\pi}(\rho)$  peut s'écrire

$$p_{\theta+\pi}(\rho) = \int_{\mathbb{R}} f(\rho \cos(\theta + \pi) - r \sin(\theta + \pi), \rho \sin(\theta + \pi) + r \cos(\theta + \pi)) dr. \quad (\text{A.12})$$

En utilisant les propriétés trigonométriques ainsi que le changement de variable  $r' = -r$ , on tombe sur

$$p_{\theta+\pi}(\rho) = \int_{\mathbb{R}} f(-\rho \cos(\theta) - r' \sin(\theta), -\rho \sin(\theta) + r' \cos(\theta)) dr'. \quad (\text{A.13})$$

Donc  $p_{\theta+\pi}(\rho) = p_\theta(-\rho)$ . Par ailleurs la fonction  $h$  est une fonction de filtrage **paire**, ainsi

$$h(x \cos(\theta + \pi) + y \sin(\theta + \pi) - \rho) = h(-x \cos(\theta) - y \sin(\theta) - \rho) = h(x \cos(\theta) + y \sin(\theta) + \rho). \quad (\text{A.14})$$

Ainsi l'équation (A.11) devient

$$\int_0^\pi \int_{-\infty}^{+\infty} p_\theta(-\rho) h(x \cos(\theta) + y \sin(\theta) + \rho) d\rho d\theta. \quad (\text{A.15})$$

En appliquant alors le changement de variable  $\rho' = -\rho$ , on tombe sur l'égalité (A.10). Nous pouvons ainsi réécrire (A.9) comme suit

$$f(x, y) = \frac{1}{2} \int_0^{2\pi} \int_{-\infty}^{+\infty} p_\theta(\rho) h(x \cos \theta + y \sin \theta - \rho) d\rho d\theta, \quad (\text{A.16})$$

puis on passe en coordonnées polaires en effectuant le changement de variable suivant

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r \cos(\phi) \\ r \sin(\phi) \end{pmatrix}.$$

En injectant cela dans (A.16) et en utilisant des propriétés trigonométriques simples on obtient

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \int_{-\infty}^{+\infty} p_\theta(\rho) h(r \cos(\theta - \phi) - \rho) d\rho d\theta. \quad (\text{A.17})$$

Afin de représenter le faisceau divergent on effectue alors le changement de variable utilisant (A.6) et (A.7). Ce changement de variable induit la matrice jacobienne suivante

$$\begin{pmatrix} 1 & 1 \\ D \cos(\alpha) & 0 \end{pmatrix},$$

ce qui donne l'égalité

$$d\rho d\theta = D \cos(\alpha) d\alpha d\beta. \quad (\text{A.18})$$

Avec ce changement et en simplifiant les bornes, l'équation (A.17) devient

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \int_{-\infty}^{+\infty} p_{(\alpha+\beta)}(D \sin(\alpha)) h(r \cos(\alpha + \beta - \phi) - D \sin(\alpha)) D \cos(\alpha) d\alpha d\beta. \quad (\text{A.19})$$

Afin d'alléger la notation on posera pour la suite

$$p(\alpha, \beta) = p_{(\alpha+\beta)}(D \sin(\alpha)).$$

On se basera maintenant sur la figure A.2 pour montrer comment simplifier l'équation (A.19). On

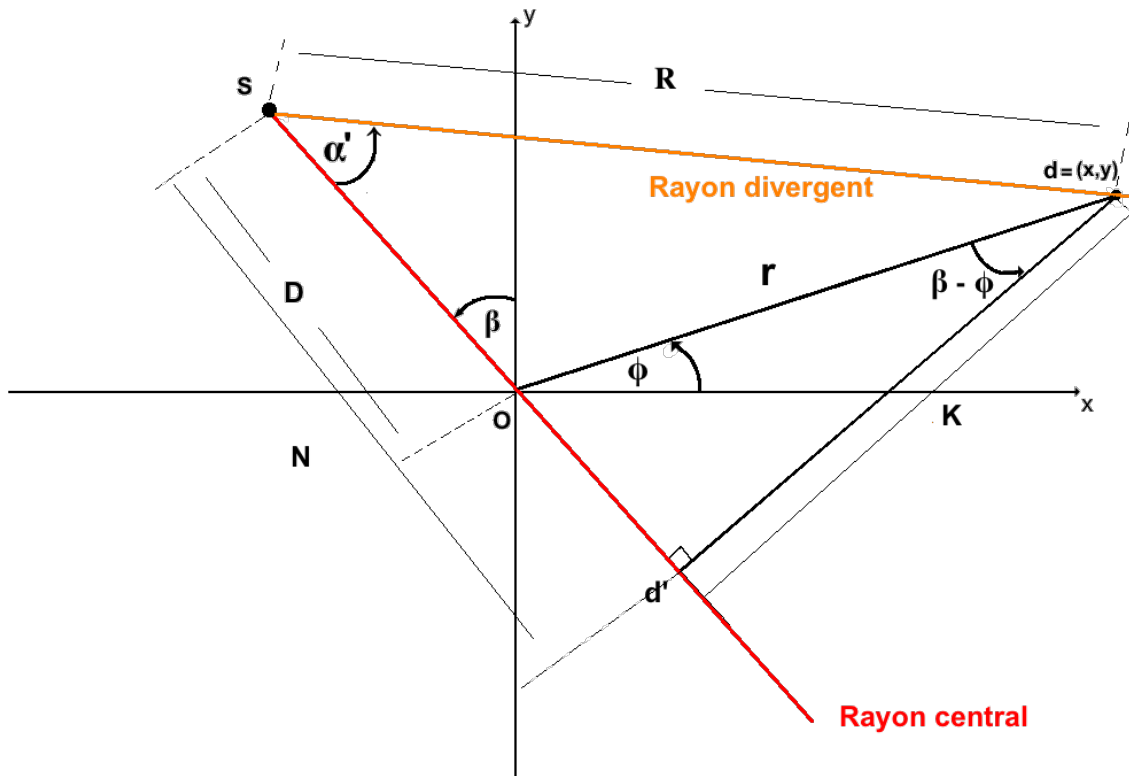


FIGURE A.2 – Illustration de la géométrie en éventail.

considère un point  $d$  de coordonnées  $(x, y)$  dans l'image. Ce point peut également être représenté en coordonnées polaire  $(r, \phi)$ . On considère le rayon divergent passant par  $d$  avec un angle d'ouverture  $\alpha'$  par rapport au rayon central.

L'idée ici est de caractériser l'angle  $\alpha'$  en fonction de  $\beta$ . On note  $R$  la longueur du segment  $[Sd]$ .

On trace le projeté orthogonal  $d'$  du point  $d$  sur le rayon central, on notera  $K$  la longueur du segment  $[dd']$ . On peut facilement montrer par complémentarité des angles que l'angle formé par les

segments  $[dO]$  et  $[dd']$  mesure  $\beta - \phi$ . Par ailleurs, on notera  $N$  la longueur du segment  $[Sd']$ .

En considérant les propriétés géométriques des triangles rectangles on peut calculer les longueurs  $K$  et  $N$  de la manière suivante

$$\begin{aligned} K &= r \cos(\beta - \phi) = R \sin(\alpha') \\ N &= r \sin(\beta - \phi) + D = R \cos(\alpha'). \end{aligned}$$

On peut maintenant réécrire l'argument de  $h$  dans l'équation (A.19)

$$\begin{aligned} r \cos(\alpha + \beta - \phi) - D \sin(\alpha) &= r \cos(\beta - \phi) \cos(\alpha) - r \sin(\beta - \phi) \sin(\alpha) - D \sin(\alpha) \\ &= r \cos(\beta - \phi) \cos(\alpha) - (r \sin(\beta - \phi) + D) \sin(\alpha), \end{aligned}$$

que l'on peut remplacer en utilisant les égalités des distances  $K$  et  $N$  précédentes

$$\begin{aligned} r \cos(\beta - \phi) \cos(\alpha) - (r \sin(\beta - \phi) + D) \sin(\alpha) &= R \sin(\alpha') \cos(\alpha) - R \cos(\alpha') \sin(\alpha) \\ &= R \sin(\alpha' - \alpha). \end{aligned}$$

On remarque que  $R$  et  $\alpha'$  sont tous deux dépendants des paramètres  $r, \phi, \beta$  et  $D$ . En remplaçant dans (A.19) on obtient

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \int_{-\infty}^{+\infty} p(\alpha, \beta) h(R \sin(\alpha' - \alpha)) D \cos(\alpha) d\alpha d\beta. \quad (\text{A.20})$$

Résonnons maintenant sur le filtre  $h$ , on rappelle que celui-ci est défini de la manière suivante

$$h(t) = \frac{1}{\pi} \int_{-\pi T}^{+\pi T} |\omega| e^{i2\omega t} d\omega, \quad (\text{A.21})$$

où  $T$  est la fréquence de coupure du filtre, donc

$$h(R \sin(\alpha' - \alpha)) = \frac{1}{\pi} \int_{-\pi T}^{+\pi T} |\omega| e^{i2\omega(R \sin(\alpha' - \alpha))} d\omega. \quad (\text{A.22})$$

On pose alors le changement de variable suivant

$$\omega' = \frac{\omega R \sin(\alpha' - \alpha)}{(\alpha' - \alpha)},$$

pour obtenir l'égalité suivante

$$h(R \sin(\alpha' - \alpha)) = \frac{1}{\pi} \int_{-\pi T'}^{+\pi T'} |\omega'| \left( \frac{\alpha' - \alpha}{R \sin(\alpha' - \alpha)} \right)^2 e^{i2\omega'(\alpha' - \alpha)} d\omega' = \left( \frac{\alpha' - \alpha}{R \sin(\alpha' - \alpha)} \right)^2 h(\alpha' - \alpha). \quad (\text{A.23})$$

En réinjectant cela dans (A.20) on obtient

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \int_{-\infty}^{+\infty} D \cos(\alpha) p(\alpha, \beta) \left( \frac{\alpha' - \alpha}{R \sin(\alpha' - \alpha)} \right)^2 h(\alpha' - \alpha) d\alpha d\beta. \quad (\text{A.24})$$

On pose alors

$$\begin{aligned} H(\alpha' - \alpha) &= \left( \frac{\alpha' - \alpha}{\sin(\alpha' - \alpha)} \right)^2 h(\alpha' - \alpha) \\ Q(\alpha, \beta) &= D \cos(\alpha) p(\alpha, \beta), \end{aligned}$$

et c'est ainsi qu'on retrouve la formule (1.27) :

$$f(r, \phi) = \frac{1}{2} \int_0^{2\pi} \frac{1}{R^2} \int_{-\infty}^{+\infty} Q(\alpha, \beta) H(\alpha' - \alpha) d\alpha d\beta. \quad (\text{A.25})$$

### A.1.4 Démonstration sous problème de minimisation en sous-espace à deux dimensions

On souhaite minimiser

$$m_k(p) = f(x_k) + \nabla f(x_k)^t p + \frac{1}{2} p^t B_k p, \quad (\text{A.26})$$

sous la contrainte que

$$p = \alpha d_1 + \beta d_2,$$

avec  $(\alpha, \beta) \in \mathbb{R}^2$ ,  $\|p\| \leq \Delta$ . On rappelle que  $d_1$  et  $d_2$  sont définis comme suit :

$$d_1 = -B_k^{-1} \nabla f(x_k), \quad d_2 = -\frac{\|\nabla f(x_k)\|^2}{\nabla f(x_k)^t B_k \nabla f(x_k)} \nabla f(x_k). \quad (\text{A.27})$$

En réécrivant (A.26) on obtient alors

$$m_k(\alpha, \beta) = f(x_k) + \nabla f(x_k)^t (\alpha d_1 + \beta d_2) + \frac{1}{2} (\alpha d_1 + \beta d_2)^t B_k (\alpha d_1 + \beta d_2). \quad (\text{A.28})$$

On développe alors les expressions puis on factorise par le vecteur  $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ . On obtient ainsi

$$m_k(\alpha, \beta) = f(x_k) + (\nabla f(x_k)^t d_1 \quad \nabla f(x_k)^t d_2) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} + \frac{1}{2} (\alpha \quad \beta) \begin{pmatrix} d_1^t B_k d_1 & d_1^t B_k d_2 \\ d_1^t B_k d_2 & d_2^t B_k d_2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

On pose alors

$$G = \begin{pmatrix} \nabla f(x_k)^t d_1 \\ \nabla f(x_k)^t d_2 \end{pmatrix} \quad A = \begin{pmatrix} d_1^t B_k d_1 & d_1^t B_k d_2 \\ d_1^t B_k d_2 & d_2^t B_k d_2 \end{pmatrix}.$$

La contrainte  $\|p\| \leq \Delta$  se développe également :

$$\|\alpha d_1 + \beta d_2\| \leq \Delta,$$

ce qui donne en développant et en factorisant par le vecteur  $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$

$$(\alpha \quad \beta) \begin{pmatrix} d_1^t d_1 & d_1^t d_2 \\ d_1^t d_2 & d_2^t d_2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \leq \Delta.$$

On pose alors

$$D = \begin{pmatrix} d_2^t d_2 & d_2^t d_1 \\ d_2^t d_1 & d_1^t d_1 \end{pmatrix} \quad \tilde{\theta} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

Ainsi le problème (A.28) devient

$$m_k(\tilde{\theta}) = f(x_k) + G^t \tilde{\theta} + \frac{1}{2} \tilde{\theta}^t A \tilde{\theta}, \quad (\text{A.29})$$

sous contrainte

$$\tilde{\theta}^t D \tilde{\theta} \leq \Delta.$$

On peut réécrire l'expression pour avoir une contrainte dépendant uniquement de la variable étudiée en posant

$$\theta = \sqrt{D} \tilde{\theta}.$$

Remarquons alors que D est semi-définie positive et qu'on peut appliquer la racine carrée. On obtient ainsi la formulation suivante :

$$\tilde{m}_k(\theta) = f(x_k) + \tilde{G}^t \theta + \frac{1}{2} \theta^t \tilde{A} \theta, \quad (\text{A.30})$$

sous contrainte que  $\|\theta\| \leq \Delta$  et où

- $\tilde{G}^t = G^t (\sqrt{D})^{-1}$ ,
- $\tilde{A} = (\sqrt{D})^{-1} A (\sqrt{D})^{-1}$ ,

■



### A.1.5 Démonstration passage au proximal ISTA

On souhaite réécrire la mise à jour suivante :

$$x_{k+1} = \text{prox}_{\gamma_k, \lambda, \|\cdot\|_1} (x_k - \gamma_k \nabla h(x_k)). \quad (\text{A.31})$$

D'après la définition du point proximal (2.23) on a

$$x_{k+1} = \arg \min_{x \in \mathbb{R}^n} \left\{ \lambda \|x\|_1 + \frac{1}{2\gamma_k} \|x - (x_k - \gamma_k \nabla h(x_k))\|^2 \right\}. \quad (\text{A.32})$$

Sachant qu'on peut écrire

$$\|U\|^2 = U^t U, \quad \forall U \in \mathbb{R}^N$$

on développe le terme au carré de (A.32) pour obtenir

$$\begin{aligned} x_{k+1} &= \arg \min_{x \in \mathbb{R}^n} \left\{ \lambda \|x\|_1 + \frac{1}{2\gamma_k} (\|x\|^2 - 2x_k^t x + \|x_k\|^2 + 2\gamma_k \nabla h(x_k)^t x - 2\gamma_k \nabla h(x_k)^t x_k + \gamma_k^2 \|\nabla h(x_k)\|^2) \right\} \\ &= \arg \min_{x \in \mathbb{R}^n} \left\{ \lambda \|x\|_1 + \frac{1}{2\gamma_k} (\|x - x_k\|^2 + 2\gamma_k \nabla h(x_k)^t (x - x_k) + \gamma_k^2 \|\nabla h(x_k)\|^2) \right\} \\ &= \arg \min_{x \in \mathbb{R}^n} \left\{ \lambda \|x\|_1 + \frac{1}{2\gamma_k} \|x - x_k\|^2 + \nabla h(x_k)^t (x - x_k) + \frac{\gamma_k}{2} \|\nabla h(x_k)\|^2 \right\}. \end{aligned}$$

Les termes constants n'interviennent pas dans la recherche d'un minimiseur. Ainsi, on peut remplacer le terme  $\frac{\gamma_k}{2} \|\nabla h(x_k)\|^2$  par  $h(x_k)$  dans l'égalité précédente pour obtenir l'équation obtenue en (2.90)

$$x_{k+1} = \arg \min_{x \in \mathbb{R}^n} \left\{ h(x_k) + \nabla h(x_k)^t (x - x_k) + \frac{1}{2\gamma_k} \|x - x_k\|^2 + \lambda \|x\|_1 \right\}. \quad (\text{A.33})$$

■

### A.1.6 Démonstration $G_1(\mathbf{A})$ analytique réelle pour le cas gaussien

On Souhaite démontrer la proposition suivante :

#### Théorème A.1.6.1: propriété de $G_1(\mathbf{A})$

Le terme d'attache aux données  $G_1(\mathbf{A})$  pour le cas du bruit gaussien

$$G_1(\mathbf{A}) = \frac{1}{2\sigma^2} \|\text{vec}(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\|_2^2,$$

est une fonction analytique réelle, par conséquent d'après la proposition donnée en (2.38),  $G_1(\mathbf{A})$  satisfait l'inégalité de **Kurdyka-Łojasiewicz**.

On rappelle que  $\mathbf{Y} \in \mathbb{R}^{P \times M}$ ,  $\mathbf{F} \in \mathbb{R}^{N \times M}$ ,  $\mathbf{S} \in \mathbb{R}^{P \times D}$ ,  $\mathbf{A} \in \mathbb{R}^{D \times K}$  et  $\Sigma \in \mathbb{R}^{K \times N}$ . Nous allons utiliser ici la **norme de Frobenius**, c'est la norme euclidienne standard lorsque l'on considère des matrices de taille  $P \times M$  comme des collections de PM scalaires. Nous avons ainsi

$$G_1(\mathbf{A}) = \|\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}\|_{fro}^2 = \sum_{p=1}^P \sum_{m=1}^M (y_{pm} - (e^{-\mathbf{SA}\Sigma}\mathbf{F})_{pm})^2 = \sum_{p=1}^P \sum_{m=1}^M \left( Y_{pm} - \sum_{n=1}^N (e^{-\mathbf{SA}\Sigma})_{pn} F_{nm} \right)^2. \quad (\text{A.34})$$

$G_1(\mathbf{A})$  s'écrit sous la forme d'une double somme de termes au carré. En effet, le théorème (3.1) dans [LANG, 1977] assure que la multiplication par un scalaire, le produit et la somme de fonctions analytiques est aussi une fonction analytique. On ne considère alors dans le terme

$$Y_{pm} - \sum_{n=1}^N (e^{-\mathbf{SA}\Sigma})_{pn} F_{nm},$$

que l'expression qui dépend de  $\mathbf{A}$ . Ainsi, il suffit de montrer que le terme  $(e^{-\mathbf{SA}\Sigma})_{pn}$  dans l'équation (A.34) est une fonction analytique pour montrer que  $G_1(\mathbf{A})$  est analytique. On considère alors la fonction  $g_{pn}(\mathbf{A})$  que l'on définit comme suit :

$$g_{pn}(\mathbf{A}) = (e^{-\mathbf{SA}\Sigma})_{pn} = \exp(-(\mathbf{SA}\Sigma)_{pn}). \quad (\text{A.35})$$

La fonction  $\exp : \mathbb{R} \rightarrow \mathbb{R}$  est une fonction entière dont le rayon de convergence est infini, elle est donc analytique réelle. D'autre part, le terme  $(\mathbf{SA}\Sigma)_{pn}$  peut se réécrire sous la forme suivante

$$(\mathbf{SA}\Sigma)_{pn} = \sum_{d=1}^D \sum_{k=1}^K S_{pd} \mathbf{A}_{dk} \Sigma_{kn}. \quad (\text{A.36})$$

C'est une fonction linéaire en  $\mathbf{A}$ , elle est donc développable en série analytique réelle dont le rayon de convergence est aussi infini. D'après le théorème (3.4) dans [LANG, 1977], la composé de ces deux fonctions est aussi une fonction analytique. Ainsi,  $g_{pn}(\mathbf{A})$  est une fonction analytique réelle, donc le terme  $Y_{pm} - \sum_{n=1}^N (e^{-\mathbf{SA}\Sigma})_{pn} F_{nm}$  est aussi une fonction analytique réelle par somme et multiplication par des scalaires. On peut alors conclure que le terme d'attache aux données  $G_1(\mathbf{A})$ , pour le cas où le bruit supposé est gaussien, est une fonction analytique et par conséquent il satisfait l'inégalité de **Kurdyka-Łojasiewicz**. ■

### A.1.7 Démonstration $G_2(\mathbf{A})$ analytique réelle pour le cas Poisson

On Souhaite démontrer la proposition suivante :

#### Théorème A.1.7.1: propriété de $G_2(\mathbf{A})$

*Le terme d'attache aux données  $G(\mathbf{A})$  pour le cas du bruit de Poisson*

$$G_2(\mathbf{A}) = \sum_{p,m} (e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m} - Y_{p,m} \log((e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m}),$$

*est une fonction analytique réelle, par conséquent d'après la proposition donnée en (2.38),  $G_2(\mathbf{A})$  satisfait l'inégalité de **Kurdyka-Łojasiewicz**.*

Comme pour le cas Gaussien précédent, il faut montrer que  $G_2(\mathbf{A})$  se compose de fonctions analytiques. On peut réécrire le terme  $G_2(\mathbf{A})$  comme suit

$$G_2(\mathbf{A}) = \sum_{p,m} (e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m} - \sum_{p,m} Y_{p,m} \log((e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m}). \quad (\text{A.37})$$

On ne tient alors compte que des termes qui dépendent de  $\mathbf{A}$ . On a vu en équation (A.36) que le terme  $g_{pn}(\mathbf{A}) = (\mathbf{SA}\Sigma)_{pn}$  est une fonction analytique réelle donc la somme  $\sum_{p,m} (e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m}$  est aussi une fonction analytique réelle. Remarquons par ailleurs qu'en TDM spectrale, toutes les quantités étudiées sont positives. Donc  $(e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m} > 0$  pour tout  $p \in \{1, \dots, P\}$  et  $m \in \{1, \dots, M\}$ . D'autre part remarquons que

$$\log((e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m}) = \log\left(\left(e^{-\mathbf{SA}\Sigma} \frac{\mathbf{F}}{\max \mathbf{F}}\right)_{p,m} \max \mathbf{F}\right) \quad (\text{A.38})$$

$$= \log\left(\left(e^{-\mathbf{SA}\Sigma} \frac{\mathbf{F}}{\max \mathbf{F}}\right)_{p,m}\right) + \log(\max \mathbf{F}), \quad (\text{A.39})$$

où  $\max \mathbf{F} = \max\{F_{nm} | n = 1 \dots N \text{ et } m = 1 \dots M\}$ . De cette manière on garantit que le logarithme du terme  $(e^{-\mathbf{SA}\Sigma} \mathbf{F})_{p,m}$  peut s'écrire en série entière convergente car  $0 < (e^{-\mathbf{SA}\Sigma} \frac{\mathbf{F}}{\max \mathbf{F}})_{p,m} \leq 1$ , c'est donc une fonction analytique. On peut alors conclure que le terme d'attache aux données  $G_2(\mathbf{A})$ , pour le cas où le bruit supposé est de Poisson, est une fonction analytique et par conséquent il satisfait l'inégalité de **Kurdyka-Łojasiewicz**. ■

### A.1.8 Calcul du gradient et de la hessienne de $G(\mathbf{A})$ pour le cas Gaussien

Nous allons donner ici le détail de calcul qui permet d'avoir l'expression du gradient et de la hessienne de la fonction d'attache aux données pour le cas où le bruit est supposé gaussien :

$$G(\mathbf{A}) = \frac{1}{2\sigma^2} \|\mathbf{vec}(\mathbf{Y} - e^{-\mathbf{SA}\Sigma\mathbf{F}})\|^2.$$

Dans un premier temps nous allons donner quelques définitions de base de calcul matriciel.

#### A.1.8.1 Prérequis

##### Définition A.1.8.1: définitions de base

Nous considérons ici  $\mathbf{X}$  et  $\mathbf{Y}$  deux matrices et nous désignons le produit matriciel par  $\mathbf{XY}$ .

► Le produit de **Frobenius** de deux matrices de même taille, comme le produit **scalaire** :

$$\mathbf{X} \cdot \mathbf{Y} = \sum_{i,j} \mathbf{X}_{ij} \mathbf{Y}_{ij}.$$

► Le produit de **Hadamard** de deux matrices de même taille est le produit par **éléments** (ou produit point par point) :

$$(\mathbf{X} \odot \mathbf{Y})_{ij} = \mathbf{X}_{ij} \mathbf{Y}_{ij}.$$

► De même qu'on peut définir la division de **Hadamard** :

$$(\mathbf{X} \oslash \mathbf{Y})_{ij} = \frac{\mathbf{X}_{ij}}{\mathbf{Y}_{ij}}.$$

► Le produit de **Kronecker** de deux matrices de tailles quelconques est le produit de **tenseur**. Pour deux matrices  $\mathbf{X} \in \mathbb{R}^{n \times m}$  et  $\mathbf{Y} \in \mathbb{R}^{p \times q}$ , le produit de Kronecker est la matrice  $(\mathbf{X} \otimes \mathbf{Y}) \in \mathbb{R}^{np \times mq}$  définie par blocs successifs de taille  $p \times q$  de la manière suivante :

$$(\mathbf{X} \otimes \mathbf{Y}) = \begin{pmatrix} \mathbf{X}_{11}\mathbf{Y} & \cdots & \mathbf{X}_{1m}\mathbf{Y} \\ \vdots & \ddots & \vdots \\ \mathbf{X}_{n1}\mathbf{Y} & \cdots & \mathbf{X}_{nm}\mathbf{Y} \end{pmatrix}.$$

► On définit **diag**( $\mathbf{V}$ ), pour  $\mathbf{V} \in \mathbb{R}^n$ , comme la matrice  $\in \mathbb{R}^{n \times n}$  dont la diagonale est composée des éléments de  $\mathbf{V}$  :

$$\mathbf{diag}(\mathbf{V}) = \begin{pmatrix} V_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & V_n \end{pmatrix}.$$

► Pour  $\mathbf{X} \in \mathbb{R}^{n \times m}$ , on note **vec**( $\mathbf{X}$ )  $\in \mathbb{R}^{nm}$ , la forme vectorisée de la matrice  $\mathbf{X}$ .

► On définit  $\mathbb{1}_{n \times m}$  comme la matrice de taille  $n \times m$  contenant 1 partout :

$$\mathbb{1}_{n \times m} = \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix}.$$

Les propriétés suivantes seront par la suite utiles au calcul matriciel.

#### Proposition A.1.8.1: propriétés de calcul

- (i) Associativité du produit scalaire et du produit de Hadamard :

$$\mathbf{X} \cdot (\mathbf{Y} \odot \mathbf{Z}) = (\mathbf{X} \odot \mathbf{Y}) \cdot \mathbf{Z}.$$

- (ii) Pseudo-commutativité du produit scalaire :

$$\mathbf{X} \cdot (\mathbf{Y}\mathbf{Z}) = (\mathbf{Y}^t \mathbf{X}) \cdot \mathbf{Z} = (\mathbf{X}\mathbf{Z}^t) \cdot \mathbf{Y}.$$

- (iii) Le produit de Hadamard vectorisé de deux matrices de même taille peut se réécrire comme suit :

$$\text{vec}(\mathbf{X} \odot \mathbf{Y}) = \text{vec}(\mathbf{X}) \odot \text{vec}(\mathbf{Y}) = \text{diag}(\text{vec}(\mathbf{X})) \text{vec}(\mathbf{Y}).$$

- (iv) Le produit matriciel vectorisé de matrices peut se réécrire comme suit :

$$\text{vec}(\mathbf{Y}\mathbf{X}\mathbf{Z}) = (\mathbf{Z}^t \otimes \mathbf{Y}) \text{vec}(\mathbf{X}).$$

- (v) On déduit de (iv) que pour une matrice  $\mathbf{X}$  de taille  $n \times m$  (resp.  $n \times p$ ) et une matrice  $\mathbf{Y}$  de taille  $p \times n$  que :

$$\text{vec}(\mathbf{Y}\mathbf{X}) = (\mathbf{I}_m \otimes \mathbf{Y}) \text{vec}(\mathbf{X}) \quad \text{resp.} \quad \text{vec}(\mathbf{X}\mathbf{Y}) = (\mathbf{Y}^t \otimes \mathbf{I}_n) \text{vec}(\mathbf{X}),$$

où  $\mathbf{I}_m$  (resp.  $\mathbf{I}_n$ ) est la matrice identité de taille  $m \times m$  (resp.  $n \times n$ ).

Nous allons dans la suite donner quelques rappels de propriété de dérivation dont on se servira pour nos calculs.

#### Proposition A.1.8.2: propriétés de dérivation

- (a) Soit deux matrices  $\mathbf{X} \in \mathbb{R}^{n \times m}$   $\mathbf{Y} \in \mathbb{R}^{m \times p}$ . La dérivée du produit matriciel s'écrit

$$d(\mathbf{X}\mathbf{Y}) = d(\mathbf{X})\mathbf{Y} + \mathbf{X}d(\mathbf{Y}).$$

Ceci reste vrai pour le produit scalaire et le produit de Hadamard.

- (b) Si une fonction  $f$  est appliquée aux éléments d'une matrice  $\mathbf{X}$  alors on exprime la dérivée matricielle comme suit :

$$df(\mathbf{X}) = f'(\mathbf{X}) \odot d\mathbf{X},$$

où  $f'$  est la dérivée de  $f$ .

- (c) Identification :

- (c1) Si  $df(\mathbf{X}) = \mathbf{A} \cdot d\mathbf{X}$  alors la matrice **jacobienne** est

$$\mathbf{J}f = \mathbf{A}.$$

- (c2) Si  $d^2f(\mathbf{X}) = (\text{vec}(d\mathbf{X}))^t \mathbf{B}(\text{vec}(d\mathbf{X}))$  alors la matrice **hessienne** est

$$\frac{1}{2}(\mathbf{B} + \mathbf{B}^t).$$

### A.1.8.2 Jacobien

Nous voulons dans ce cas calculer le gradient de la fonction d'attache aux données pour le cas du bruit gaussien suivante :

$$G_1 : \mathbb{R}^{D \times K} \rightarrow \mathbb{R}^+$$

$$\mathbf{A} \mapsto \|\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}\|^2$$

avec  $\mathbf{Y} \in \mathbb{R}^{P \times M}$ ,  $\mathbf{S} \in \mathbb{R}^{P \times D}$ ,  $\mathbf{A} \in \mathbb{R}^{D \times K}$ ,  $\Sigma \in \mathbb{R}^{K \times M}$ ,  $\mathbf{F} \in \mathbb{R}^{N \times M}$ . On peut réécrire  $G$  comme le produit scalaire de deux expressions :

$$G_1(\mathbf{A}) = (\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}) \cdot (\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}).$$

On dérive alors  $G_1$  en utilisant **(a)**, et avec la commutativité du produit scalaire on obtient :

$$dG_1(\mathbf{A}) = 2(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}) \cdot d(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}).$$

La dérivé est une application linéaire et la matrices des données  $\mathbf{Y}$  ne dépend pas de  $\mathbf{A}$  donc :

$$dG_1(\mathbf{A}) = -2(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}) \cdot d(e^{-\mathbf{SA}\Sigma}\mathbf{F}).$$

On utilise **(a)** sur le terme à droite du produit scalaire :

$$dG_1(\mathbf{A}) = -2(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F}) \cdot d(e^{-\mathbf{SA}\Sigma})\mathbf{F}.$$

L'objectif est d'arrivé à une forme identifiable, comme celle mentionnée en **(c1)**. On utilise donc **(ii)** pour isoler le terme dérivé :

$$dG_1(\mathbf{A}) = -2(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \cdot d(e^{-\mathbf{SA}\Sigma}).$$

La fonction exponentielle est appliquée à chaque élément, on utilise alors **(b)** sur le second terme :

$$dG_1(\mathbf{A}) = 2(\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \cdot (e^{-\mathbf{SA}\Sigma} \odot d(\mathbf{SA}\Sigma)).$$

En utilisant **(i)** on isole le terme dérivé :

$$dG_1(\mathbf{A}) = 2((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \cdot d(\mathbf{SA}\Sigma).$$

On utilise à nouveau **(a)** :

$$dG_1(\mathbf{A}) = 2((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \cdot \mathbf{S}d(\mathbf{A})\Sigma.$$

On applique **(ii)** deux fois sur le terme à droite pour obtenir :

$$dG_1(\mathbf{A}) = 2\mathbf{S}^t((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma})\Sigma^t \cdot d(\mathbf{A}).$$

Ainsi nous obtenons une expression qu'on identifie à l'aide de **(c1)** :

$$\nabla G_1(\mathbf{A}) = 2\mathbf{S}^t \left( (\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma} \right) \Sigma^t. \quad (\text{A.40})$$

### A.1.8.3 Hessienne

Comme pour le cas précédent, l'objectif du calcul est d'obtenir une forme identifiable comme en (c2). Dans ce cas, nous allons utiliser la forme vectorisée :

$$\begin{aligned} d^2 G_1(\mathbf{A}) &= d(dG(\mathbf{A})) \\ &= \mathbf{vec}(d(\nabla_{\mathbf{A}} G(\mathbf{A})))^t \mathbf{vec}(d(\mathbf{A})) \\ &= \mathbf{vec}(d[2\mathbf{S}^t((\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}]\Sigma^t))^t \mathbf{vec}(d(\mathbf{A})). \end{aligned}$$

On utilise (a) pour dériver cette expression :

$$d^2 G_1(\mathbf{A}) = \mathbf{vec}(2\mathbf{S}^t d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}]\Sigma^t)^t \mathbf{vec}(d(\mathbf{A})).$$

On reconnaît ici la forme  $\mathbf{vec}(XYZ)$ , on utilise alors (iv) pour obtenir :

$$d^2 G_1(\mathbf{A}) = (2(\Sigma \otimes \mathbf{S}^t) \underbrace{\mathbf{vec}(d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}])^t}_{:=d^2 f_1}) \mathbf{vec}(d(\mathbf{A})). \quad (\text{A.41})$$

On ne développera que le terme dans l'accolade et on utilise (a) pour le dériver :

$$\mathbf{vec}(d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}]) = \mathbf{vec}(d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t] \odot e^{-\mathbf{SA}\Sigma} + (\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot d[e^{-\mathbf{SA}\Sigma}]).$$

En utilisant la linéarité de la vectorisation, on obtient :

$$\mathbf{vec}(d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}]) = \underbrace{\mathbf{vec}(d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t] \odot e^{-\mathbf{SA}\Sigma})}_{:=d^2 f_1} + \underbrace{\mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t \odot d[e^{-\mathbf{SA}\Sigma}])}_{:=d^2 f_2}.$$

Dans la suite nous allons procéder par étape pour calculer les deux termes  $d^2 f_1$  et  $d^2 f_2$ .

1 ► Premier terme :

$$d^2 f_1 = \mathbf{vec}(d[(\mathbf{Y} - e^{-\mathbf{SA}\Sigma})\mathbf{F})\mathbf{F}^t] \odot e^{-\mathbf{SA}\Sigma}).$$

On utilise (a) sur le terme de gauche pour isoler  $d(\mathbf{A})$  :

$$d^2 f_1 = \mathbf{vec}(-d[e^{-\mathbf{SA}\Sigma}]\mathbf{F}\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}).$$

On utilise (b) pour dériver l'exponentielle :

$$d^2 f_1 = \mathbf{vec}((e^{-\mathbf{SA}\Sigma} \odot d[\mathbf{SA}\Sigma])\mathbf{F}\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}).$$

On utilise (a) une fois de plus pour dériver le terme à gauche

$$d^2 f_1 = \mathbf{vec}((e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F}\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}).$$

On reconnaît alors l'expression  $\mathbf{vec}(\mathbf{X} \odot \mathbf{Y})$ , donc avec (iii) on obtient :

$$d^2 f_1 = \mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))\mathbf{vec}((e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F}\mathbf{F}^t).$$

On utilise (v) sur la partie droite de l'expression :

$$d^2 f_1 = \mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))(\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_p)\mathbf{vec}(e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma).$$

On utilise à nouveau (iii) :

$$d^2 f_1 = \mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))(\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_p)\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))\mathbf{vec}(\mathbf{S}d(\mathbf{A})\Sigma).$$

Enfin, avec (iv) on obtient :

$$d^2 f_1 = \underbrace{\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))(\mathbf{F}\mathbf{F}^t \otimes \mathbf{I}_p)\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))}_{:=\phi_1}(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d(\mathbf{A})).$$

2 ► Second terme :

$$d^2 f_2 = \mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot d[e^{-\mathbf{SA}\Sigma}]).$$

On utilise (b) et (a) pour dériver l'exponentielle et le terme précédent :

$$d^2 f_2 = \mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot (-e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)).$$

Le produit de Hadamard est comutatif, ainsi on peut regrouper les deux premier terme et en utilisant (iii) on obtient :

$$d^2 f_2 = \mathbf{diag}(\mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot -e^{-\mathbf{SA}\Sigma}))\mathbf{vec}(\mathbf{S}d(\mathbf{A})\Sigma).$$

Finalement, on utilise (iv) pour obtenir :

$$d^2 f_2 = \underbrace{\mathbf{diag}(\mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot -e^{-\mathbf{SA}\Sigma}))}_{:=\phi_2}(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d(\mathbf{A})).$$

On revient alors à l'expression (A.41) :

$$d^2 G_1(\mathbf{A}) = (2(\Sigma \otimes \mathbf{S}^t)(\phi_1 + \phi_2)\mathbf{vec}(d(\mathbf{A}))^t \mathbf{vec}(d(\mathbf{A})).$$

Ce qui donne :

$$d^2 G_1(\mathbf{A}) = 2\mathbf{vec}(d(\mathbf{A}))^t [(\phi_1 + \phi_2)^t (\Sigma \otimes \mathbf{S}^t)^t] \mathbf{vec}(d(\mathbf{A})).$$

On déduit alors avec (c2) que la hessienne est :

$$\mathbf{H}_A G_1(\mathbf{A}) = [(\phi_1 + \phi_2)^t (\Sigma \otimes \mathbf{S}^t)^t + (\Sigma \otimes \mathbf{S}^t)(\phi_1 + \phi_2)].$$

**Remarque :**

$$(\phi_1 + \phi_2) = (\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma})))(\mathbf{F}\mathbf{F}^t \otimes \mathbf{I})\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))(\Sigma^t \otimes \mathbf{S}) + \mathbf{diag}(\mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot -e^{-\mathbf{SA}\Sigma}))(\Sigma^t \otimes \mathbf{S}).$$

On factorise par  $(\Sigma^t \otimes \mathbf{S})$  et on pose :

$$\Phi = (\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma})))(\mathbf{F}\mathbf{F}^t \otimes \mathbf{I})\mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma})) + \mathbf{diag}(\mathbf{vec}((\mathbf{Y} - e^{-\mathbf{SA}\Sigma}\mathbf{F})\mathbf{F}^t \odot -e^{-\mathbf{SA}\Sigma})). \quad (\text{A.42})$$

On obtient :

$$(\phi_1 + \phi_2) = \Phi(\Sigma^t \otimes \mathbf{S}).$$

On sait que  $(\mathbf{X} \otimes \mathbf{Y})^t = (\mathbf{X}^t \otimes \mathbf{Y}^t)$ , donc  $\Phi$  est symétrique et on obtient :

$$(\phi_1 + \phi_2)^t = (\Sigma^t \otimes \mathbf{S})^t \Phi^t = (\Sigma \otimes \mathbf{S}^t)\Phi.$$

On déduit alors :

$$\mathbf{H}_A G_1(\mathbf{A}) = 2(\Sigma \otimes \mathbf{S}^t) \Phi (\Sigma^t \otimes \mathbf{S}). \quad (\text{A.43})$$

■

## A.1.9 Calcul du gradient et de la hessienne de $G(\mathbf{A})$ pour le cas Poisson

### A.1.9.1 Gradient

On s'intéresse ici à la fonction d'attache aux données pour cas de bruit de Poisson. Cette fonction s'écrit :

$$G_2 : \mathbb{R}^{D \times K} \rightarrow \mathbb{R}^+ \\ \mathbf{A} \mapsto (e^{-\mathbf{SA}\Sigma}\mathbf{F} - \mathbf{Y} \odot \log(e^{-\mathbf{SA}\Sigma}\mathbf{F})) \cdot \mathbf{W}$$

avec :  $\mathbf{Y} \in \mathbb{R}^{P \times M}$ ,  $\mathbf{S} \in \mathbb{R}^{P \times D}$ ,  $\mathbf{A} \in \mathbb{R}^{D \times K}$ ,  $\Sigma \in \mathbb{R}^{K \times M}$ ,  $\mathbf{F} \in \mathbb{R}^{N \times M}$  and  $\mathbf{W} \in \mathbb{R}^{P \times M}$ .  $\mathbf{W}$  est une matrice de pondération, dans le cas pratique nous supposons que celle si est l'identité. On dérive  $G$  avec (a) et avec la commutativité du produit scalaire on obtient :

$$dG_2(\mathbf{A}) = \mathbf{W} \cdot d(e^{-\mathbf{SA}\Sigma}\mathbf{F} - \mathbf{Y} \odot \mathbf{log}(e^{-\mathbf{SA}\Sigma}\mathbf{F})).$$

La linéarité de la dérivée  $d(\cdot)$  donne :

$$dG_2(\mathbf{A}) = \mathbf{W} \cdot (d(e^{-\mathbf{SA}\Sigma}\mathbf{F}) - d(\mathbf{Y} \odot \mathbf{log}(e^{-\mathbf{SA}\Sigma}\mathbf{F}))).$$

Les fonctions log et exp sont appliquées aux éléments, on utilise alors (a) et (b) :

$$dG_2(\mathbf{A}) = \mathbf{W} \cdot ((e^{-\mathbf{SA}\Sigma} \odot -\mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F} - \mathbf{Y} \odot ((\mathbb{1}_{P \times M} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) \odot (e^{-\mathbf{SA}\Sigma} \odot -\mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F})),$$

où  $\mathbb{1}_{P \times M}$  est une matrice de taille  $P \times M$  contenant des 1 et  $\oslash$  iest la division de Hadamard. On a alors :

$$dG_2(\mathbf{A}) = \mathbf{W} \cdot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) \odot (e^{\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F} - (e^{\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F}).$$

On obtient en factorisant :

$$dG_2(\mathbf{A}) = \mathbf{W} \cdot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M}) \odot (e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F}.$$

On utilise (i) :

$$dG_2(\mathbf{A}) = (\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \cdot (e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma)\mathbf{F},$$

puis (ii) :

$$dG_2(\mathbf{A}) = (\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \cdot (e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d(\mathbf{A})\Sigma).$$

On utilise à nouveau (i) :

$$dG_2(\mathbf{A}) = ((\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \cdot \mathbf{S}d(\mathbf{A})\Sigma,$$

puis finalement (ii) :

$$dG_2(\mathbf{A}) = \mathbf{S}^t ((\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \Sigma^t \cdot d(\mathbf{A}).$$

Par l'identification en c1, on déduit :

$$\nabla G_2(\mathbf{A}) = \mathbf{S}^t ((\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \Sigma^t \quad (\text{A.44})$$

### A.1.9.2 Hessienne

Comme pour le cas du bruit gaussien, on souhaite obtenir une forme identifiable comme en (c2) pour avoir la hessienne dans le cas du bruit de Poisson. On utilise alors la forme vectorisée des matrices :

$$\begin{aligned} d^2 G_2(\mathbf{A}) &= d(dG_2(\mathbf{A})) \\ &= \mathbf{vec}(d[\mathbf{J}_A G_2(\mathbf{A})])^t \mathbf{vec}(d[\mathbf{A}]) \\ &= \mathbf{vec}(d[\mathbf{S}^t ((\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \Sigma^t])^t \mathbf{vec}(d[\mathbf{A}]). \end{aligned}$$

On utilise (a) pour dériver l'expression :

$$d^2 G_2(\mathbf{A}) = \mathbf{vec}(\mathbf{S}^t d[(\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \Sigma^t])^t \mathbf{vec}(d[\mathbf{A}]).$$

A l'aide de la propriété de la transposée de matrice et de (iv) on obtient :

$$d^2 G_2(\mathbf{A}) = \underbrace{\mathbf{vec}(d[(\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}) \Sigma^t]}_{:=\Psi_A} (\Sigma^t \otimes \mathbf{S}) \mathbf{vec}(d[\mathbf{A}]). \quad (\text{A.45})$$

Nous développerons uniquement le terme dans l'accolade, nous utilisons (a) pour le dériver :

$$\Psi_A = \underbrace{\mathbf{vec}(d[(\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma})]}_{:=d^2 g_1(\mathbf{A})} + \underbrace{(\mathbf{W} \odot ((\mathbf{Y} \oslash e^{-\mathbf{SA}\Sigma}\mathbf{F}) - \mathbb{1}_{P \times M})) \mathbf{F}^t \odot d[e^{-\mathbf{SA}\Sigma}]}_{:=d^2 g_2(\mathbf{A})}.$$



**1 ► Premier terme :**

On utilise **(iii)**, **(v)** et **(a)** :

$$d^2 g_1(\mathbf{A}) = \mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))(\mathbf{F} \otimes \mathbf{I})\mathbf{vec}(d[\mathbf{W} \odot ((\mathbf{Y} \otimes e^{-\mathbf{SA}\Sigma})\mathbf{F}) - \mathbb{1}_{\mathbf{P} \times \mathbf{M}}])).$$

On pose  $\mathbf{D}_1 = \mathbf{diag}(\mathbf{vec}(e^{-\mathbf{SA}\Sigma}))$ . On remarque que  $\mathbf{W}$  et  $\mathbb{1}_{\mathbf{P} \times \mathbf{M}}$  sont des termes constants, donc :

$$d^2 g_1(\mathbf{A}) = \mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{vec}(\mathbf{W} \odot d[(\mathbf{Y} \otimes e^{-\mathbf{SA}\Sigma})\mathbf{F}])).$$

On utilise alors **(a)** et **(b)** :

$$d^2 g_1(\mathbf{A}) = \mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{vec}(\mathbf{W} \odot \mathbf{Y} \odot (e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d[\mathbf{A}]\Sigma)\mathbf{F} \odot (e^{-\mathbf{SA}\Sigma}\mathbf{F})^2)).$$

Avec **(iii)** on obtient :

$$d^2 g_1(\mathbf{A}) = \mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{diag}(\mathbf{vec}((\mathbf{W} \odot \mathbf{Y}) \odot (e^{-\mathbf{SA}\Sigma}\mathbf{F})^2))\mathbf{vec}((e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d[\mathbf{A}]\Sigma)\mathbf{F})).$$

On pose alors  $\mathbf{D}_2 = \mathbf{diag}(\mathbf{vec}((\mathbf{W} \odot \mathbf{Y}) \odot (e^{-\mathbf{SA}\Sigma}\mathbf{F})^2))$ . On obtient avec **(v)** :

$$d^2 g_1(\mathbf{A}) = \mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{D}_2(\mathbf{F}^t \otimes \mathbf{I})\mathbf{vec}(e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d[\mathbf{A}]\Sigma).$$

Puis avec **(iii)** et **(iv)** :

$$d^2 g_1(\mathbf{A}) = \mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{D}_2(\mathbf{F}^t \otimes \mathbf{I})\mathbf{D}_1(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d[\mathbf{A}]).$$

**2 ► Second terme :**

$$d^2 g_2(\mathbf{A}) = \mathbf{vec}((\mathbf{W} \odot ((\mathbf{Y} \otimes e^{-\mathbf{SA}\Sigma})\mathbf{F}) - \mathbb{1}_{\mathbf{P} \times \mathbf{M}}))\mathbf{F}^t \odot d[e^{-\mathbf{SA}\Sigma}])).$$

On utilis **(iii)**, **(a)** et **(b)** :

$$d^2 g_2(\mathbf{A}) = \mathbf{diag}(\mathbf{vec}((\mathbf{W} \odot (\mathbb{1}_{\mathbf{P} \times \mathbf{M}} - \mathbf{Y} \otimes e^{-\mathbf{SA}\Sigma})\mathbf{F}))\mathbf{F}^t))\mathbf{vec}(e^{-\mathbf{SA}\Sigma} \odot \mathbf{S}d[\mathbf{A}]\Sigma).$$

Puis, avec **(iii)** et **(iv)** :

$$d^2 g_2(\mathbf{A}) = \mathbf{diag}(\mathbf{vec}((\mathbf{W} \odot (\mathbb{1}_{\mathbf{P} \times \mathbf{M}} - \mathbf{Y} \otimes e^{-\mathbf{SA}\Sigma})\mathbf{F}))\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma}))(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d[\mathbf{A}]).$$

On pose alors :

$$\mathbf{D}_3 = \mathbf{diag}(\mathbf{vec}((\mathbf{W} \odot (\mathbb{1}_{\mathbf{P} \times \mathbf{M}} - \mathbf{Y} \otimes e^{-\mathbf{SA}\Sigma})\mathbf{F}))\mathbf{F}^t \odot e^{-\mathbf{SA}\Sigma})).$$

On injecte ces résultats dans l'expression (A.45) :

$$\begin{aligned} d^2 G_2(\mathbf{A}) &= \Psi_{\mathbf{A}}^t(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d[\mathbf{A}]) \\ &= (d^2 g_1(\mathbf{A}) + d^2 g_2(\mathbf{A}))^t(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d[\mathbf{A}]). \end{aligned}$$

Ceci donne :

$$d^2 G_2(\mathbf{A}) = \mathbf{vec}(d[\mathbf{A}])^t(\Sigma \otimes \mathbf{S}^t)(\mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{D}_2(\mathbf{F}^t \otimes \mathbf{I})\mathbf{D}_1 + \mathbf{D}_3)(\Sigma^t \otimes \mathbf{S})\mathbf{vec}(d[\mathbf{A}]).$$

On peut alors conclure avec **(c2)** que la hessienne est :

$$\mathbf{H}_{\mathbf{A}}G_2(\mathbf{A}) = (\Sigma \otimes \mathbf{S}^t)(\mathbf{D}_1(\mathbf{F} \otimes \mathbf{I})\mathbf{D}_2(\mathbf{F}^t \otimes \mathbf{I})\mathbf{D}_1 + \mathbf{D}_3)(\Sigma^t \otimes \mathbf{S}). \quad (\text{A.46})$$

■

### A.1.10 Démonstration filtre conique

Dans allons donner ici la démonstration permettant de faire le lien entre le **filtre conique** (ou l'inverse du **filtre rampe**) et l'opérateur de **projection-rétroprojection**  $S^t S$  qui apparaît dans la métrique permettant d'accélérer la convergence des algorithmes proximaux.

Dans le cas continu, nous avons vu précédemment que la **projection**  $P_f$ ,  $\forall \theta \in [0, \pi[$  et  $\forall t \in \mathbb{R}$ , d'une fonction  $f$  le long d'un rayon noté  $\Delta_{t,\theta}$  est définie par :

$$P_f : [0, \pi] \times \mathbb{R} \rightarrow \mathbb{R} \\ (\theta, t) \mapsto P_f(\theta, t) = p_\theta(t)$$

où

$$p_\theta(t) = \int_{\Delta_{t,\theta}} f(v) dv = \int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy. \quad (\text{A.47})$$

On définit alors l'opérateur linéaire de la **rétroprojection** pondérée par

$$V_\omega : L^2([0, \pi] \times \mathbb{R}) \rightarrow L^2(\mathbb{R}^2) \\ u \mapsto V_\omega u(x, y) = \int_0^\pi \omega(\theta) u(\theta, x \cos \theta + y \sin \theta) d\theta$$

où  $\omega : \mathbb{R} \rightarrow \mathbb{R}$  est une fonction poids  $\pi$ -périodique et bornée, ce qui garantit donc que  $V_\omega$  est bien défini. Remarquons que lorsque  $\omega = 1$ ,  $V_1 = P^*$  est la rétroprojection comme définie en A.1.2.

#### Théorème A.1.10.1: filtre conique

Soit  $f \in L^2(\mathbb{R}^2)$ . On pose  $T_\omega = V_\omega P$ , où  $P$  est la transformée définie en (A.47). On définit par  $h$  :

$$\forall x, y \in \mathbb{R}^* \quad h(x, y) = \frac{\omega(\arctan(\frac{y}{x} + \frac{\pi}{2}) \ [\pi])}{\sqrt{x^2 + y^2}},$$

qui peut aussi se réécrire en version **polaire signée**

$$h^{p.s.}(\theta, t) = \frac{\omega(\theta + \frac{\pi}{2} \ [\pi])}{|t|}, \quad (\text{A.48})$$

où  $z \ [\pi]$  est le reste de la division euclidienne de  $z$  par  $\pi$ . Alors

$$T_\omega f = f \star h, \quad (\text{A.49})$$

où  $\star$  dénote le produit de convolution.

Avant de donner la démonstration du théorème, nous aurons besoin du lemme suivant :

#### Lemme A.1.10.1: polaire signée

$$\int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) dx dy = \int_0^\pi \int_{\mathbb{R}} |t| f(t \cos \theta, t \sin \theta) dt d\theta. \quad (\text{A.50})$$

#### Preuve lemme :

On pose  $(x, y) = (t \cos \theta, t \sin \theta)$  sur la partie gauche de l'égalité afin de passer en coordonnées polaires, en appliquant ce changement de variables on obtient :

$$\int_{\mathbb{R}} \int_{\mathbb{R}} f(x, y) dx dy = \int_0^{2\pi} \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta.$$

On décompose alors la première intégrale :

$$\int_0^{2\pi} \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta = \int_0^\pi \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta + \int_\pi^{2\pi} \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta.$$

On applique le changement de variable  $\theta \rightarrow \theta - \pi$  à la seconde partie, on obtient alors :

$$\int_0^{2\pi} \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta = \int_0^\pi \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta + \int_0^\pi \int_0^{+\infty} t f(-t \cos \theta, -t \sin \theta) dt d\theta.$$

On applique ensuite le changement de variable  $t \rightarrow -t$  à cette même partie

$$\int_0^{2\pi} \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta = \int_0^\pi \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta - \int_0^\pi \int_{-\infty}^0 t f(t \cos \theta, t \sin \theta) dt d\theta,$$

d'où

$$\int_0^{2\pi} \int_0^{+\infty} t f(t \cos \theta, t \sin \theta) dt d\theta = \int_0^\pi \int_{\mathbb{R}} |t| f(t \cos \theta, t \sin \theta) dt d\theta$$

■

**Remarques :** Soit  $h(x, y) \in L^2(\mathbb{R}^2)$ , on pose  $\forall \theta \in [0, 2\pi]$  et  $t \in [0, +\infty[$  pour effectuer le changement en variable **polaire** :

$$h^p(\theta, t) = h(t \cos \theta, t \sin \theta)$$

On dit alors que changement de variable est **polaire signé** lorsque  $\theta \in [0, \pi]$  et  $t \in ]-\infty, +\infty[$ , on le note  $h^{p.s.}(\theta, t)$ .

### Démonstration du théorème filtre conique

(i) Dans un premier temps nous allons montrer que l'égalité (A.49) est vraie au point  $(0, 0)$ . En utilisant l'équation (1.23) on a d'une part

$$\begin{aligned} \mathbf{T}_\omega f(0, 0) &= \mathbf{V}_\omega \mathbf{P}_f(\theta, 0) = \int_0^\pi \omega(\theta) \mathbf{P}_f(\theta, 0) d\theta \\ &= \int_0^\pi \omega(\theta) \int_{\mathbb{R}} f(-r \sin \theta, r \cos \theta) dr d\theta. \end{aligned}$$

L'objectif est d'arriver à la même égalité en partant du produit de convolution  $f * h$ . Par définition du produit de convolution nous avons

$$(f * h)(0, 0) = \int_{\mathbb{R}} \int_{\mathbb{R}} f(-x, -y) h(x, y) dx dy.$$

En passant en coordonnées polaires signées (voir (A.50)) on obtient

$$(f * h)(0, 0) = \int_0^\pi \int_{\mathbb{R}} |r| f(-r \cos \theta, -r \sin \theta) h^{p.s.}(\theta, r) dr d\theta,$$

ce qui donne en remplaçant par la valeur de  $h^{p.s.}$

$$(f * h)(0, 0) = \int_0^\pi \int_{\mathbb{R}} |r| f(-r \cos \theta, -r \sin \theta) \frac{\omega(\theta + \frac{\pi}{2} \lfloor \pi \rfloor)}{|r|} dr d\theta.$$

On applique alors le changement de variable  $r \rightarrow -r$

$$(f * h)(0, 0) = \int_0^\pi \int_{\mathbb{R}} f(r \cos \theta, r \sin \theta) \omega(\theta + \frac{\pi}{2} \lfloor \pi \rfloor) dr d\theta,$$

puis le changement  $\theta \rightarrow \theta + \frac{\pi}{2}$

$$(f * h)(0, 0) = \int_{\frac{\pi}{2}}^{\frac{3\pi}{2}} \int_{\mathbb{R}} f(r \sin \theta, -r \cos \theta) \omega(\theta \lfloor \pi \rfloor) dr d\theta.$$

En remarquant que lorsque  $\theta \in [\frac{\pi}{2}, \pi]$  on a  $\theta - \pi \in [-\frac{\pi}{2}, 0]$  et lorsque  $\theta \in ]\pi, \frac{3\pi}{2}]$  on a  $\theta - \pi \in ]\pi, \frac{3\pi}{2}]$ , on sépare alors l'intégrale

$$(f * h)(0, 0) = \int_{\frac{\pi}{2}}^{\pi} \int_{\mathbb{R}} f(r \sin \theta, -r \cos \theta) \omega(\theta) dr d\theta + \int_{\pi}^{\frac{3\pi}{2}} \int_{\mathbb{R}} f(r \sin \theta, -r \cos \theta) \omega(\theta - \pi) dr d\theta.$$

On applique ensuite le changement de variable  $r \rightarrow -r$  à la première partie et  $\theta \rightarrow \theta - \pi$  à la seconde partie, on obtient ainsi

$$(f * h)(0, 0) = \int_{\frac{\pi}{2}}^{\pi} \int_{\mathbb{R}} f(-r \sin \theta, r \cos \theta) \omega(\theta) dr d\theta + \int_0^{\frac{\pi}{2}} \int_{\mathbb{R}} f(-r \sin \theta, r \cos \theta) \omega(\theta) dr d\theta,$$

ce qui donne au final

$$(f * h)(0, 0) = \int_0^{\pi} \int_{\mathbb{R}} \omega(\theta) f(-r \sin \theta, r \cos \theta) dr d\theta.$$

On a donc bien

$$\mathbf{T}_{\omega} f(0, 0) = (f * h)(0, 0).$$

(ii) Afin de démontrer l'égalité en tout point il faut montrer que l'opérateur linéaire  $\mathbf{T}_{\omega}$  commute avec les translations. On pose  $\forall x_0, y_0 \in \mathbb{R}$  l'opérateur de translation

$$\mathcal{T}_{(x_0, y_0)} f(x, y) = f(x - x_0, y - y_0).$$

On veut montrer l'égalité suivante pour tout  $(x_0, y_0) \in \mathbb{R}^2$  :

$$\mathcal{T}_{(x_0, y_0)}(\mathbf{T}_{\omega} f) = \mathbf{T}_{\omega}(\mathcal{T}_{(x_0, y_0)} f).$$

1. Nous avons d'un côté

$$\mathcal{T}_{(x_0, y_0)}(\mathbf{T}_{\omega} f)(x, y) = \mathbf{T}_{\omega} f(x - x_0, y - y_0),$$

ce qui donne

$$\mathcal{T}_{(x_0, y_0)}(\mathbf{T}_{\omega} f)(x, y) = \int_0^{\pi} \omega(\theta) P f(\theta, (x - x_0) \cos \theta + (y - y_0) \sin \theta) d\theta.$$

2. Calculons maintenant la deuxième partie

$$\mathbf{T}_{\omega}(\mathcal{T}_{(x_0, y_0)} f)(x, y) = \int_0^{\pi} \omega(\theta) P(\mathcal{T}_{(x_0, y_0)} f)(\theta, x \cos \theta + y \sin \theta) d\theta. \quad (\text{A.51})$$

Or

$$P(\mathcal{T}_{(x_0, y_0)} f)(\theta, x \cos \theta + y \sin \theta) = \int_{\mathbb{R}} \int_{\mathbb{R}} (\mathcal{T}_{(x_0, y_0)} f)(u, v) \delta(u \cos \theta + v \sin \theta - x \cos \theta - y \sin \theta) du dv,$$

ce qui donne en translatant  $f$

$$P(\mathcal{T}_{(x_0, y_0)} f)(\theta, x \cos \theta + y \sin \theta) = \int_{\mathbb{R}} \int_{\mathbb{R}} f(u - x_0, v - y_0) \delta((u - x_0) \cos \theta + (v - y_0) \sin \theta - x \cos \theta - y \sin \theta) du dv.$$

On applique alors le changement de variable  $(u, v) = (u - x_0, v - y_0)$

$$P(\mathcal{T}_{(x_0, y_0)} f)(\theta, x \cos \theta + y \sin \theta) = \int_{\mathbb{R}} \int_{\mathbb{R}} f(u, v) \delta((u + x_0 - x) \cos \theta + (v + y_0 - y) \sin \theta) du dv,$$

cela donne

$$P(\mathcal{T}_{(x_0, y_0)} f)(\theta, x \cos \theta + y \sin \theta) = \int_{\mathbb{R}} \int_{\mathbb{R}} f(u, v) \delta(u \cos \theta + v \sin \theta - [(x - x_0) \cos \theta + (y - y_0) \sin \theta]) du dv.$$

On reconnaît alors que

$$P(\mathcal{T}_{(x_0, y_0)} f)(\theta, x \cos \theta + y \sin \theta) = P f(\theta, (x - x_0) \cos \theta + (y - y_0) \sin \theta).$$

On remplace alors dans l'équation (A.51) :

$$\mathbf{T}_{\omega}(\mathcal{T}_{(x_0, y_0)} f)(x, y) = \int_0^{\pi} \omega(\theta) P f(\theta, (x - x_0) \cos \theta + (y - y_0) \sin \theta) d\theta.$$

Nous avons donc bien montré l'égalité

$$\mathcal{T}_{(x_0, y_0)}(\mathbf{T}_\omega f) = \mathbf{T}_\omega(\mathcal{T}_{(x_0, y_0)} f).$$

Donc l'opérateur  $\mathbf{T}_\omega$  commute avec les translations. Nous pouvons à l'aide de cette propriété montrer l'égalité en tout point. Soient  $(x_0, y_0) \in \mathbb{R}^2$ , la définition du produit de convolution donne

$$(f \star h)(-x_0, -y_0) = \int_{\mathbb{R}} \int_{\mathbb{R}} f(-x_0 - x, -y_0 - y) h(x, y) dx dy,$$

ce qui peut s'écrire

$$(f \star h)(-x_0, -y_0) = \int_{\mathbb{R}} \int_{\mathbb{R}} (\mathcal{T}_{x_0, y_0} f)(-x, -y) h(x, y) dx dy.$$

On a par conséquent l'égalité

$$(f \star h)(-x_0, -y_0) = (\mathcal{T}_{x_0, y_0} f \star h)(0, 0).$$

Or nous avons montré l'égalité en  $(0, 0)$ , on a

$$(\mathcal{T}_{x_0, y_0} f \star h)(0, 0) = \mathbf{T}_\omega(\mathcal{T}_{x_0, y_0} f)(0, 0),$$

et comme on a vu que l'opérateur  $\mathbf{T}_\omega$  commute avec les translations on a

$$\mathbf{T}_\omega(\mathcal{T}_{x_0, y_0} f)(0, 0) = \mathcal{T}_{x_0, y_0}(\mathbf{T}_\omega f)(0, 0),$$

ce qui donne

$$\mathcal{T}_{x_0, y_0}(\mathbf{T}_\omega f)(0, 0) = \mathbf{T}_\omega f(-x_0, -y_0).$$

On a donc montré que  $\forall (x_0, y_0) \in \mathbb{R}^2$

$$(f \star h)(-x_0, -y_0) = \mathbf{T}_\omega f(-x_0, -y_0).$$

Ils sont donc égaux en tout point. Ainsi

$$\mathbf{T}_\omega f = f \star h.$$

**Remarques :** Lorsque  $\omega = 1$  on a  $\mathbf{T}_1 = P^*P$  et donc  $\mathbf{T}_1 f = P^*P f = f \star h$ . Or le produit de convolution se ramène à un produit point à point dans l'espace de Fourier. Ainsi, dans notre cas de figure on obtient que

$$\mathbf{S}^t \mathbf{S} f = \mathcal{F}^{-1} \left[ \mathcal{F}[f] \circledast \mathcal{F}[h] \right],$$

où  $\mathcal{F}$  est l'opérateur de transformée de Fourier. Il devient alors simple d'inverser  $\mathbf{S}^t \mathbf{S}$  en inversant  $\mathcal{F}[h]$  :

$$(\mathbf{S}^t \mathbf{S})^{-1} f = \mathcal{F}^{-1} \left[ \mathcal{F}[f] \oslash \mathcal{F}[h] \right],$$

où  $\oslash$  est la division de Hadamard. Appliquer l'inverse de l'opérateur  $\mathbf{S}^t \mathbf{S}$  revient juste à effectuer un produit de convolution dans le domaine réel et donc un produit de Hadamard dans le domaine de Fourier. C'est une opération de filtrage, et on appelle  $\mathcal{F}[h]$  **filtre conique** (en anglais *cone filter*).

■

## A.2 Références

LANG, S. 1977, *Complex analysis*, Addison-Wesley Pub. Co Reading Mass. [IX](#), [X](#)

