Amélioration qualitative et quantitative de reconstruction TEP sur plate-forme graphique

Awen Autret

▶ **To cite this version:**

Awen Autret. Amélioration qualitative et quantitative de reconstruction TEP sur plate-forme graphique. Imagerie médicale. Télécom Bretagne; Université de Bretagne Occidentale, 2015. Français. NNT : . tel-01272743

HAL Id: tel-01272743
https://hal.science/tel-01272743

Submitted on 11 Feb 2016

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

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

THÈSE / Télécom Bretagne
sous le sceau de l'Université européenne de Bretagne
pour obtenir le grade de Docteur de Télécom Bretagne
En accréditation conjointe avec l'Ecole Doctorale Sicma
Mention : Sciences et Technologies de l'Information et de la Communication

présentée par
Awen Autret
préparée dans le département Image & traitement de l'information
Laboratoire Latim

Amélioration qualitative et quantitative de reconstruction TEP sur plate-forme graphique

Thèse soutenue le 3 décembre 2015
Devant le jury composé de :

Vincent Rodin
Professeur, Université de Bretagne Occidentale / président

Michel Defrise
Professeur, Université Libre de Bruxelles / rapporteur

Claude Comtat
Directeur de recherche, IMIV – CEA - Orsay / rapporteur

Julien Bert
Ingénieur de recherche, Latim - CHRU Brest / examinateur

Olivier Strauss
Maître de conférences (HDR), Limm - Université Montpellier II / examinateur

Dimitris Visvikis
Directeur de Recherche, Latim - Inserm / directeur de thèse

Frédéric Lamare
Ingénieur de recherche, Incia - CHU Bordeaux / invité

N° d'ordre : 2015telb0363

Sous le sceau de l'Université européenne de Bretagne

Télécom Bretagne

En accréditation conjointe avec l'Ecole Doctorale Sicma

Amélioration qualitative et quantitative de reconstruction TEP sur plate-forme graphique

---

Thèse de Doctorat

Mention : Maths - STIC (sciences et technologies de l'information et de la communication)

Présentée par **Awen AUTRET**

Département Image et Traitement de l'Information

Laboratoire de Traitement de l'Information Médicale - INSERM UMR 1101

Directeur de thèse : Dimitris VISVIKIS

Soutenue le 3 décembre 2015

Jury :

Rapporteurs :	M. **Michel Defrise**, *Professeur, Department of Nuclear Medicine, Universitair Ziekenhuis Brussel*
	M. **Claude Comtat**, *Physicien, Laboratoire IMIV, CEA, Orsay*
Examinateurs :	M. **Dimitris Visvikis**, *Directeur de Recherche, LaTIM, INSERM, Brest*
	M. **Vincent Rodin**, *Professeur, Lab-STICC, Université de Bretagne Occidentale*
	M. **Julien Bert**, *Ingénieur de recherche, LaTIM, CHRU, Brest*
	M. **Olivier Strauss**, *Maître de Conférences, LIRMM, CNRS, Montpellier*
Invité :	M. **Frédéric Lamare**, *Ingénieur de recherche, INCIA, CHU Bordeaux*

Résumé

La tomographie par émission de positons (TEP) est une modalité d'imagerie médicale fonctionnelle, incontournable pour le diagnostic et le suivi thérapeutique, couramment employée dans plusieurs branches de la médecine. En imagerie TEP, la quantification précise de la répartition du traceur impose de prendre en compte, dans le processus de reconstruction, l'intégralité des nombreux phénomènes physiques intervenant pendant l'acquisition, dans l'objet imagé aussi bien que dans le détecteur du scanner. La modélisation précise de tous ces effets nécessite une puissance de calcul considérable qui n'est atteinte aujourd'hui que par des grappes de serveurs de calcul aux coûts d'achat et d'entretien importants. Depuis le milieu des années 2000, les processeurs graphiques (*GPU*), rendus programmables par les fabricants de cartes graphiques, ont permis d'accéder à des puissances de calcul comparables à celles d'un *cluster* de plusieurs centaines de cœurs, le tout dans un ordinateur de bureau classique et pour un coût dérisoire. L'objectif de cette thèse a été de proposer des méthodes permettant de tendre vers une reconstruction TEP qualitative et quantitative dans des temps compatibles avec les applications en routine clinique en exploitant la puissance de calcul des *GPU*.

En reconstruction TEP itérative, les étapes de projection et rétroprojection modélisant la réponse du détecteur à l'aide d'un projecteur, peuvent avoir un coût en temps de calcul important, même avec une implémentation *GPU*. Dans un ordinateur de bureau, il est possible d'intégrer jusqu'à 8 *GPU* et plus encore sur certaines plates-formes professionnelles. Dans la première partie de cette thèse nous avons développé une nouvelle approche de distribution de la reconstruction sur une architecture multi-*GPU*. Cette méthode permet de tirer parti efficacement de la puissance de tous les *GPU* tout en limitant les temps de communication entre *GPU*.

Nous avons ensuite proposé un nouveau projecteur (*IRIS*), basé sur un échantillonnage aléatoire d'une modélisation de la fonction de réponse intrinsèque du détecteur dérivée de mesures effectuées par simulation Monte-Carlo (SMC). Ce projecteur permet d'estimer à la volée la réponse du détecteur en tenant compte de tous les effets intercristaux et intracristaux. Nous avons montré que ce projecteur apporte une amélioration de la qualité d'image tout en conservant un coût de calcul raisonnable pour une utilisation clinique.

Ensuite, nous avons effectué une étude comparative du projecteur *IRIS* avec un projecteur basé sur une matrice de la réponse du système stockée, précalculée à l'aide de SMC. Notre projecteur permet d'égaler, voire de dépasser les performances de ce second projecteur en matière de qualité d'image, tout en nécessitant moins d'espace mémoire pour stocker le modèle ainsi que des temps de simulation pour la construction du modèle et de reconstruction inférieurs.

L'utilisation de radionucléides comme le ^{68}Ga ou le ^{82}Rb progresse, parce qu'ils ne nécessitent pas de disposer d'un accélérateur de particules dans l'hôpital. Cependant, ces marqueurs émettent des positons dont l'énergie cinétique initiale moyenne est très importante, ce qui implique un parcours des positons important et donc une dégradation de la résolution spatiale des images reconstruites. Dans ce contexte, nous avons proposé une nouvelle méthode de correction du parcours du positon

basée sur une convolution, avec comme particularité d'utiliser une SMC simplifiée et accélérée sur *GPU* pour effectuer cette convolution. Cette approche permet de prendre en compte de manière précise les hétérogénéités des matériaux constitutifs du corps du patient, permettant d'améliorer la quantification de la concentration du traceur, en particulier aux interfaces entre tissus de densités très différentes et dans les tissus peu denses.

Abstract

Positron emission tomography (PET) is a functional medical imaging modality fundamental in various fields of medicine for diagnosis and therapeutic monitoring. In PET imaging, the accurate quantization of the distribution of the radioactive tracer in the patient's body requires to take into account, in the reconstruction process, of all the effects that occur in the system, including the patient and the scanner. The accurate modeling of all these effects requires a large amount of computing power, which is currently only reachable with an expensive computing cluster. Since the middle of the last decade, graphic processing units (GPU) were made programmable by their manufacturers, which made them suitable for general purpose computing. The GPU are designed for a high computing power using a massively parallel processing architecture. Using the GPUs allows to have in a single computer a similar computing power as that of a cluster with hundreds of cores. The overall aim of this thesis was to propose and develop methods that capitalize on GPUs to allow a quantitative and qualitative PET reconstruction with times compatible with the clinical context.

In iterative PET reconstruction, the forward and backward projection steps that models the detector response have a significant computing cost, even with a GPU implementation. As we can easily fit more than one GPU in a single computer and up to 8 GPUs in some professional platforms, we proposed a new way to scatter the reconstruction process across multiple GPUs that distributes the reconstruction volume instead of the list-mode data. We showed that this approach allows to exploit better the available computing power while keeping a low communication time between the GPUs.

We also proposed two new projectors, called IRIS, based on a random sampling of the intrinsic detector response functions using a model built based on data obtained from Monte Carlo simulations (MCS). These projectors can compute during the reconstruction on the fly the detector response including all the physical and geometrical effects in the detector crystals. We compared them with other on the fly projectors and showed that they surpass them in term of image quality and resolution while having a smaller computing cost. We then compared the IRIS projectors with a projector based on a stored projection matrix built from MCS before the reconstruction. The IRIS projectors provided similar or better results than the stored matrix approach while being faster and less memory intensive.

The use of radionuclides such as ^{68}Ga or ^{82}Rb as tracer has been growing in the recent years because they don't require to have an expensive particle accelerator in the hospital. However, these tracers emit high energy positrons that lead to an important positron range effect and thus to a loss of resolution in the reconstructed images. In this context, we proposed a new approach to correct this effect that allows a fine modeling of the matter heterogeneities in the patient's body. This method is based on a convolution of the reconstructed image before processing the forward projection and uses a simplified MCS accelerated with a GPU implementation that computes on the fly the local kernels, specifics to each voxel neighborhood. We have shown that a reconstruction incorporating this method provides a good recovery of the tracer distribution, even in the soft tissues, without introducing introducing artifacts as other methods do.

Table des matières

Résumé . . . . . **i**

Abstract . . . . . **iii**

Table des matières . . . . . **vi**

Liste des figures . . . . . **x**

Liste des tableaux . . . . . **xi**

Introduction **1**

Chapitre 1 Principes généraux de la tomographie par émission de positons **3**

1.1 Introduction 5

1.2 Principes physiques d'une acquisition en TEP 6

1.3 Chaîne de traitement des événements détectés et formation des coïncidences 17

1.4 La reconstruction en tomographie par émission de positons 22

1.5 Modélisation du système et corrections 33

1.6 Conclusion 44

Chapitre 2 Accélération de la reconstruction TEP sur plate-forme multi-*GPU* **47**

2.1 Introduction 48

2.2 Le calcul sur *GPU* 48

2.3 Reconstruction LM-OSEM mono-*GPU* 55

2.4 Reconstruction multi-*GPU* avec fractionnement des sous-ensembles de données *list-mode* 57

2.5 Reconstruction multi-*GPU* avec fractionnement du volume de reconstruction 60

2.6 Étude d'évaluation 66

2.7 Résultats 67

2.8 Discussion et conclusion 72

Chapitre 3 Projecteur intégrant un modèle complet de la réponse du détecteur **75**

3.1 Introduction 77

3.2 Modélisation de la réponse du système 77

3.3 Projecteurs multiligne avec un modèle complet de la réponse du détecteur 88

3.4 Implémentations des projecteurs 95

3.5 Étude d'évaluation 95

3.6 Discussion et conclusion 109

Chapitre 4 Étude comparative de matrices système précalculées et calculées à la volée 111
4.1 Introduction . . . 112
4.2 Méthode $S(MC)^2PET$ basée sur un matrice système stockée . . . 112
4.3 Étude d'évaluation . . . 117
4.4 Discussion et conclusion . . . 126

Chapitre 5 Correction du parcours du positon par simulation sur *GPU* . . . 129
5.1 Introduction . . . 130
5.2 Estimation du parcours du positon . . . 131
5.3 Correction du parcours du positon . . . 132
5.4 Proposition d'une nouvelle méthode de correction du parcours du positon . . . 135
5.5 Étude d'évaluation . . . 140
5.6 Résultats . . . 144
5.7 Discussion et conclusion . . . 153

Conclusion et perspectives . . . 155

Annexe A Définitions . . . 159
A.1 Glossaire . . . 159
A.2 Liste des symboles . . . 161

Annexe B Communications . . . 163

Annexe C Détails sur l'implémentation *GPU* avec *CUDA* . . . 165
C.1 Définition de la taille des blocs . . . 165
C.2 Exemple d'implémentation : la convolution 1D . . . 166

Bibliographie . . . 187

Liste des figures

1.1 Schéma du processus complet de la TEP. 5
1.2 Le glucose et ^{18}F-FDG 6
1.3 Désintégration β^+ 8
1.4 Annihilation positron-électron. 8
1.5 Énergies cinétiques des positons émis par différents traceurs. 9
1.6 Distance parcourue par des positons jusqu'à annihilation. 10
1.7 Erreur induite par la non-colinéarité des photons d'annihilation. 11
1.8 Sections efficaces des photons avec l'eau pour les cinq interactions physiques possibles. 11
1.9 Reconstruction avec et sans correction d'atténuation. 13
1.10 Diffusion dans le patient. 13
1.11 Principe de détection des photons d'annihilation. 14
1.12 Effets de parallaxe et diffusion dans le détecteur 16
1.13 Schéma des traitements appliqués aux événements détectés pour former les coïncidences. 18
1.14 Fausses coïncidences. 19
1.15 Représentation de l'information de temps de vol. 20
1.16 Système de coordonnées d'un sinogramme. 22
1.17 Illustration d'un scanner 2D et des plans reconstruits. 23
1.18 Paramétrisation de la transformée de Radon bidimensionnelle. 24
1.19 Illustration du théorème de la coupe centrale. 25
1.20 Comparaison de reconstructions analytiques et itérative. 27
1.21 Mise en forme algébrique du problème de la reconstruction. 28
1.22 Méthode des projections de Kaczmarz. 29
1.23 Problème surdéterminé avec la méthode des projections de Kaczmarz. 29
1.24 Comparaison de reconstruction avec et sans sous-ensembles. 32
1.25 Différentes sophistications des modélisations de la matrice système. 34
1.26 Sinogrammes des coïncidences vraies et fortuites. 39
1.27 Comparaison des sinogrammes des coïncidences vraies et diffusées. 41
1.28 Réduction des effets de parallaxe en utilisant la profondeur d'interaction. 44

2.1 Comparaison des puissances et bande passante mémoire des *CPU* et *GPU*. 49
2.2 Illustration des différences entre architecture *CPU* et architecture *GPU*. 51
2.3 Exemple de *kernel* pour additionner deux vecteurs. 54
2.4 Exemple d'exécution d'un *kernel* avec les allocations et copies mémoires nécessaires. 54
2.5 Diagramme de la reconstruction LM-OSEM. 56
2.6 Modélisation de la SRM par une distribution Gaussienne 2D invariante. 56

2.7 Comparaison d'implémentation *CPU* et *GPU* de la projection et rétroprojection. . . . 57
2.8 Diagramme de la reconstruction avec fractionnement des sous-ensembles de données. 59
2.9 Schéma des données traitées par chaque *GPU* avec la méthode de fractionnement des sous-ensembles. . . . 59
2.10 Diagramme de la reconstruction avec fractionnement du volume. . . . 61
2.11 Schéma des données traitées par chaque *GPU* avec la méthode du volume de reconstruction. . . . 62
2.12 Découpe du volume de reconstruction à taille équivalente. . . . 63
2.13 Coupe sagittale d'une estimation de la charge de travail en chaque voxel. . . . 63
2.14 Estimation de la charge de travail par coupe. . . . 64
2.15 Principe de la découpe du volume de reconstruction à charge de travail équivalente. . 64
2.16 Fantôme NEMA NU-2 2001. . . . 66
2.17 Comparaison d'une coupe transverse du fantôme NEMA IEC NU 2-2001 reconstruit avec les méthodes de parallélisation. . . . 68
2.18 Profils dans les images reconstruites à l'itération 30. . . . 68
2.19 Image de différence entre les reconstructions parallélisées en la reconstruction non parallélisée. . . . 68
2.20 Différence moyenne entre les reconstructions parallélisées et la reconstruction non parallélisée. . . . 69
2.21 Facteur d'accélération des différentes méthodes de parallélisation. . . . 70
2.22 Temps de communication par itération des différentes méthodes de parallélisation. . 71
2.23 Fraction du temps passé à communiquer sur le temps total pour les différentes méthodes de parallélisation. . . . 72

3.1 Illustration de la *CDRF*. . . . 78
3.2 Modèles linéiques de la CDRF. . . . 79
3.3 Représentation bidimensionnelle de la modélisation de la CDRF par un cylindre. . . . 81
3.4 Sections de CDRF mesurées par SMC. . . . 81
3.5 Paramétrisation des positions le long de LOR avec le projecteur Gaussien$_{variant}$ 83
3.6 Paramétrisation des angles d'incidence pour le calcul des *IDRF*. . . . 84
3.7 Exemple d'une *IDRF* mesurée avec la définition de [Lecomte *et al.*, 1984] 84
3.8 Définition alternative de la IDRF de [Gonzalez *et al.*, 2011] 85
3.9 Principe d'estimation de la *CDRF* avec le projecteur *IRIS*. . . . 89
3.10 Diagrammes de la projection et de la rétroprojection avec le projecteur *IRIS*. . . . 90
3.11 Exemple d'un ensemble d'*IDRF*$_{3D}$ mesurées. . . . 91
3.12 Production d'échantillons d'une distribution décrite par un histogramme. . . . 92
3.13 Distributions interne et externe de *IDRF*$_{3D}$ le long de l'axe du cristal pour des angles d'incidences différents. . . . 93
3.14 Coupe suivant le plan $x' = 0$ de *IDRF*$_{3D}$ avec des angles d'incidences différents. . . . 94
3.15 Tables des paramètres du modèle du projecteur Gaussien$_{variant}$. . . . 97
3.16 Tableaux des paramètres estimés du modèle du projecteur *IRIS*$_{analytique}$. . . . 98

3.17 Contraste dans une petite sphère chaude en fonction du bruit dans le fond pour des nombres variables de lignes accumulées par *LOR* avec les projecteurs *IRIS* et Chen. . . 100
3.18 Contraste en fonction du bruit dans les images reconstruites du fantôme NEMA IEC NU 2-2001. 101
3.19 Reconstructions du fantôme composé de quatre sources ponctuelles. 104
3.20 Résolution des images reconstruites en fonction des itérations estimées en différentes positions du champ de vue. 105
3.21 Coupes transverses des cartes d'activités des fantômes anthropomorphiques simulés. 105
3.22 Reconstructions du fantôme $NCAT_1$. 107
3.23 Reconstructions du fantôme $NCAT_1$. 108

4.1 Symétries dans l'échantillonnage du champ de vue. 114
4.2 Symétries du détecteur et d'échantillonnage du champ de vue. 115
4.3 Volume source utilisé dans la simulation Monte-Carlo pour estimer la matrice système. 115
4.4 Indexation des *LOR* 116
4.5 Schématisation de la méthode de stockage de matrice creuse, exploitée avec la méthode *S(MC)^{2}PET* pour la matrice système. 117
4.6 Fantôme Jaszczak utilisé pour l'étude du recouvrement du contraste. 119
4.7 Schéma d'une coupe transverse du fantôme Derenzo. 119
4.8 Schéma du fantôme NEMA NU-4 2008 utilisé pour évaluer la qualité d'image des scanners précliniques. 120
4.9 Contrastes mesurés dans les reconstructions du fantôme Jaszczak. 121
4.10 Images reconstruites du fantôme Jaszczak à niveau de bruit équivalent. 122
4.11 Contraste en fonction du bruit, mesurés dans les reconstructions du fantôme NEMA. . 123
4.12 Coupes transverses du fantôme NEMA en deux positions différentes, dans les images reconstruites avec un même niveau de bruit. 124
4.13 Mesures de la résolution sur les fantômes Derenzo et NEMA. 124
4.14 Coupes transverses des images reconstruites du fantôme Derenzo. 125

5.1 Pertes d'énergie fractionnelle, des électrons et positons, par unité de longueur, associées aux différents effets physiques, en fonction de l'énergie. 132
5.2 Diagramme du processus de simulation du positons avec la méthode *MCC-PR*. 137
5.3 Spectres d'énergie cinétique des positons émis par des traceurs communs, mesurés avec *GATE*. 138
5.4 Histogrammes des variations des paramètres des positons le long d'un pas dans trois matériaux. 139
5.5 Fantôme pour l'évaluation de l'espace de phase. 140
5.6 Fantôme F_{prof} pour une étude visuelle de la simulation et de la correction du parcours du positon. 141
5.7 Fantôme F_{cont} pour l'étude du contraste en fonction du bruit dans les reconstructions. 142
5.8 Fantôme F_{res} dédié à l'évaluation de la résolution dans les images reconstruites. 142

5.9 Histogrammes de la distance parcourue par les positons jusqu'à annihilation, simulés avec *GATE* et *MCC-PR*. 144
5.10 Trajectoires de 20 positons simulés avec *GATE* et *MCC-PR*. 145
5.11 Histogramme des positions et directions suivant l'axe z des positons des espaces de phase. 145
5.12 Distributions des annihilations des positons émis par du ^{18}F et du ^{82}Rb dans les fantômes F_{prof} et F_{cont}. 146
5.13 Profils dans les distributions des annihilations des positons émis par du ^{18}F et du ^{82}Rb dans les fantômes F_{prof} et F_{res}. 147
5.14 Coupes frontales des reconstructions du fantôme F_{prof}. 149
5.15 Profil dans les reconstructions du fantôme F_{prof}. 149
5.16 Courbes du contraste, en fonction du bruit. 150
5.17 Coupes transverses des reconstructions des données ^{82}Rb du fantôme F_{cont}. 151
5.18 Mesures de la résolution sur le fantôme F_{res}. 152
5.19 Profils passant par les sources ponctuelles du fantôme F_{res} dans les reconstructions des données *back-to-back*, ^{18}F et ^{82}Rb. 153

C.1 Comparaison des temps d'exécution d'une convolution implémenté sur *CPU* et sur *GPU*. 169

Liste des tableaux

1.1 Liste d'isotopes les plus communs en TEP 7

2.1 Liste des microarchitectures *GPU* NVIDIA entre 2006 et 2015 et quelques-unes de leurs spécificités. 51
2.2 Temps d'exécution des reconstructions multi-*GPU*. 72

3.1 Nombre d'itérations pour atteindre un niveau de $Bruit_{CV}$ de 30 %. 102
3.2 Temps de reconstruction des différents projecteurs. 102
3.3 Contrastes mesurés dans les fantômes anthropomorphiques. 106

4.1 Temps de reconstruction du fantôme NEMA, par itération. 125
4.2 Quantités de mémoire et temps de simulation nécessaires aux différents modèles. . . . 126

5.1 Temps de simulation du parcours d'un million de positons émis par du ^{18}F et du ^{82}Rb dans les trois fantômes étudiés, simulés avec *GATE* et avec la méthode *MCC-PR*. 148
5.2 Temps nécessaire à chaque itération pour corriger le parcours du positon. 153

Introduction

La tomographie par émission de positons (TEP) est une modalité d'imagerie médicale permettant de visualiser spatialement l'activité métabolique associée à une fonction de l'organisme, en mesurant la concentration volumique d'une molécule marquée par un isotope radioactif émetteur de positons. Elle est employée dans plusieurs branches de la médecine, pour le diagnostic et le suivi thérapeutique, où elle permet de révéler des anomalies locales du métabolisme.

Dans le chapitre 1 nous aborderons l'ensemble des principes de la TEP. Nous commencerons par introduire les principes de l'acquisition et les différents phénomènes qui peuvent venir perturber les informations acquises. Ensuite, nous développerons les méthodes permettant de reconstruire les images de la répartition du traceur dans les tissus du patient à partir des données acquises par le scanner. Nous verrons d'abord des méthodes de reconstruction analytiques, qui ont été historiquement les premières à être utilisées en TEP, puis nous présenterons des méthodes itératives, aujourd'hui utilisées, et les avantages qu'elles apportent par rapport aux méthodes analytiques. Nous aborderons aussi les différentes méthodes utilisées pour corriger les effets qui perturbent les données acquises.

En reconstruction TEP l'ensemble des corrections indispensables pour obtenir une reconstruction qualitative et quantitative nécessite des puissances de calcul très importantes, qui imposerait de disposer d'un large *cluster* de calcul pour obtenir une reconstruction s'exécutant dans des temps compatibles avec les applications cliniques. En pratique, il n'est pas envisageable d'utiliser de telles solutions de calcul pour des raisons de coût d'achat et d'entretien, ce qui impose de négliger ou de simplifier certaines corrections, au prix d'une moins bonne qualité d'image. Depuis quelques années, la puissance de calcul massivement parallèle des processeurs graphiques ou *graphics processing units* en anglais (*GPU*), normalement dédiée au traitement de données graphiques, a été rendue utilisable pour tous types de calculs. Depuis, plusieurs implémentations *GPU* de la reconstruction en TEP ont déjà été développées. Bien que très puissant par rapport à un microprocesseur ou *central processing unit* en anglais (*CPU*), un unique *GPU* n'est pas toujours suffisant pour obtenir des temps de reconstruction compatibles avec les applications cliniques. Dans un ordinateur de bureau, il est courant de pouvoir y intégrer quelques *GPU* et jusqu'à une dizaine dans des ordinateurs dédiés aux jeux vidéo ou dans certaines solutions professionnelles. Dans le chapitre 2 nous introduirons les spécificités du calcul sur *GPU* et nous présenterons une nouvelle méthode générique permettant de distribuer les reconstructions TEP itératives sur des systèmes multi-*GPU*.

L'obtention d'une reconstruction en TEP qui soit qualitative et quantitative, implique de prendre en compte tous les effets intervenant pendant le processus d'acquisition. On trouve des solutions satisfaisantes pour la plupart d'entre eux, comme la diffusion, les temps-morts, les coïncidences fortuites ou encore l'atténuation, mais les effets physiques et géométriques associés au détecteur sont souvent négligés lorsqu'un projecteur est utilisé pour calculer la réponse du détecteur à la volée. Dans le chapitre 3 nous introduirons un nouveau projecteur permettant de modéliser précisément et à la volée la réponse associée aux effets physiques et géométriques intervenant dans le détecteur. Dans

ce chapitre, nous évaluerons les performances de l'approche proposée par rapport à d'autres projecteurs calculant aussi la réponse du détecteur à la volée. Dans le chapitre 4 nous effectuerons une étude comparative de notre méthode face à un projecteur basé sur une réponse du détecteur stockée, estimée par simulation Monte-Carlo (SMC) avant la reconstruction.

Un autre effet souvent négligé est le parcours du positon. En effet, avec le traceur le plus couramment employé, le fluor 18, la distance parcourue par les positons est en moyenne largement inférieure à la résolution du détecteur. D'autres traceurs sont associés à un parcours du positon plus important, mais sont employés dans des contextes où des solutions approximatives suffisent. Par exemple, le parcours du positon associé à l'oxygène 15, employé en neurologie, peut être modélisé par une fonction d'étalement invariante en raison de l'homogénéité des tissus cérébraux. Cependant, depuis quelques années, l'utilisation de radionucléides comme le gallium 68 ou le rubidium 82, se développe pour des raisons de coût comme l'ont noté [Jødal *et al.*, 2014]. Ces isotopes émettent des positons beaucoup plus énergétiques que le ^{18}F qui parcourt donc en moyenne plus de distance avant de s'annihiler, ce qui entraîne une dégradation importante des images reconstruites. De plus, ils peuvent être utilisés dans des contextes où les tissus sont très hétérogènes, comme dans le thorax, par exemple. Dans le chapitre 5 nous présenterons une nouvelle méthode de simulation rapide du parcours du positon afin de corriger cet effet dans la reconstruction. Nous évaluerons son impact sur la qualité des images données avec une reconstruction intégrant cette simulation.

Chapitre 1

Principes généraux de la tomographie par émission de positons

1.1 Introduction **5**
1.2 Principes physiques d'une acquisition en TEP **6**
1.2.1 Le traceur radioactif 6
1.2.2 Désintégration β^+ et émission du positon 7
1.2.3 Annihilation du positon 8
1.2.3.1 Parcours du positon avant annihilation 9
1.2.3.2 Non-colinéarité des photons d'annihilation 10
1.2.4 Parcours des photons d'annihilation jusqu'au détecteur 10
1.2.5 Détection des photons d'annihilation 14
1.2.5.1 Principes 14
1.2.5.2 Variations de la sensibilité 15
1.2.5.3 Effet de parallaxe 16
1.2.5.4 Diffusion dans le détecteur 16
1.2.5.5 Résolution énergétique limitée 17
1.2.5.6 La profondeur d'interaction 17
1.3 Chaîne de traitement des événements détectés et formation des coïncidences . . . **17**
1.3.1 Principes 17
1.3.2 Temps-mort 18
1.3.3 Coïncidences fortuites 19
1.3.4 Le temps de vol 19
1.3.5 Les modes de stockage des coïncidences 20
1.3.5.1 *List-mode* 20
1.3.5.2 Mode histogramme 21
1.3.5.3 Mode sinogramme 21
1.4 La reconstruction en tomographie par émission de positons **22**
1.4.1 Reconstruction analytique 22
1.4.1.1 Reconstruction 2D 23
1.4.1.2 Reconstruction 3D 26
1.4.1.3 Limites des approches analytiques 26
1.4.2 Reconstruction itérative 27
1.4.2.1 Reconstruction algébrique 28

1.4.2.2 Reconstruction basée sur un modèle statistique de Poisson . . . 29
1.4.2.3 Bruit dans les images reconstruites . . . 32
1.5 Modélisation du système et corrections . . . **33**
1.5.1 La matrice système . . . 33
1.5.2 Effets physiques et géométriques liés au détecteur . . . 35
1.5.3 Normalisation . . . 36
1.5.4 Atténuation . . . 37
1.5.5 Parcours du positon . . . 38
1.5.6 Coïncidences fortuites . . . 39
1.5.7 Temps-mort . . . 40
1.5.8 Diffusion . . . 41
1.5.9 Intégration des effets modélisés . . . 43
1.5.10 Prise en compte de l'information de temps de vol . . . 43
1.5.11 Prise en compte de la DOI ou de la POI . . . 44
1.6 Conclusion . . . **44**

1.1 Introduction

La TEP est une modalité d'imagerie permettant de visualiser l'activité d'une fonction métabolique spécifique dans le corps d'un patient en mesurant la répartition d'un traceur radioactif dans ses tissus. La fonction observée dépend de la molécule traçante injectée au patient qui en fonction de sa composition chimique, va intervenir dans des métabolismes spécifiques. La détection de ce traceur radioactif repose sur une chaîne d'événements physiques et de traitements électroniques et informatiques dont le point de départ est l'émission de positons par un traceur composé de molécules marquées par un isotope radioactif, injecté au patient. Les positons s'annihilent rapidement avec les électrons présents dans le corps du patient et forment des paires de photons γ possédant chacun la même énergie de 511 kilo électron-volts (eV). Les photons issus d'une même annihilation se propagent sur une même ligne, mais en sens opposés. Ils sont ensuite détectés en deux points du scanner et une série de traitements électroniques permet de déterminer si ces deux photons proviennent bien de la même annihilation. Lorsque cette vérification est validée, on obtient ce que l'on appelle une coïncidence. Cette coïncidence indique qu'une annihilation s'est produite quelque part le long d'une ligne passant par les positions de détection dans le scanner des deux photons γ. On appelle cette ligne virtuelle une ligne de réponse ou *line of response* en anglais (*LOR*). L'examen se poursuit quelques dizaines de minutes, pendant lesquelles plusieurs dizaines ou centaines de millions de coïncidences sont détectées et enregistrées. Chaque coïncidence ne donne pas la position d'une annihilation de manière précise mais seulement qu'elle s'est produite le long de la *LOR*. À partir de la liste des coïncidences détectées, on fait appel à un algorithme de reconstruction pour obtenir les images de la répartition spatiale du traceur dans le corps du patient. La figure 1.1 montre un résumé schématique de l'ensemble du processus de la TEP.

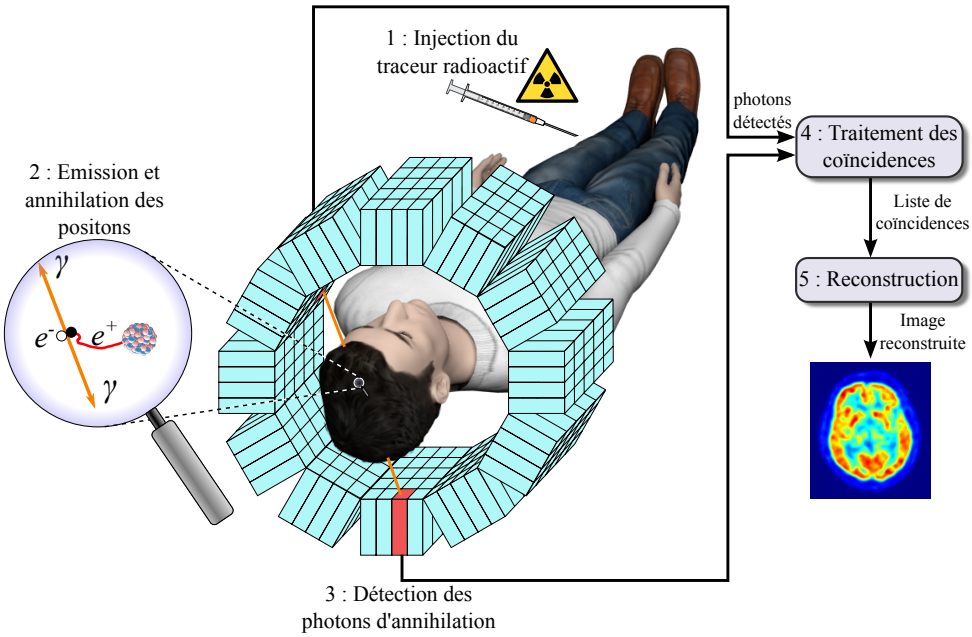


FIGURE 1.1 – Schéma du processus complet de la TEP.

Dans ce chapitre, une première partie sera dédiée à l'introduction des principes physiques géné-

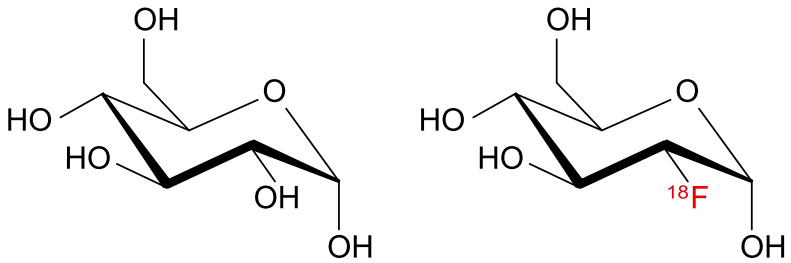
raux de l'émission de positons, suivis des principes de détection des photons d'annihilations et du traitement des données détectées permettant de former les coïncidences. Une seconde partie sera, quant à elle, dédiée à la reconstruction des images de la répartition spatiale du traceur dans le corps du patient.

1.2 Principes physiques d'une acquisition en TEP

Un examen TEP commence par l'injection d'un traceur radioactif émetteur de particules β^+, aussi appelé positon, dans le corps du patient. Les positons émis s'annihilent avec les électrons présents dans la matière environnante en formant des paires de photons qui se propagent à environ 180° l'un de l'autre. La détection de ces paires de photons permet de localiser les lieux d'émissions et par conséquent la concentration du traceur en chaque point du corps du patient. Cette section détaille les étapes de ce processus.

1.2.1 Le traceur radioactif

La solution injectée au patient en début d'examen contient un traceur qui est composé d'une molécule dont au moins un de ses atomes ou un de ses groupements d'atomes (exemple : hydroxyle) est substitué par un isotope instable émetteur de particules β^+ aussi appelées positons. En fonction de sa formule chimique, le traceur est impliqué dans des métabolismes spécifiques. Un traceur très communément utilisé en routine clinique pour le diagnostic en cancérologie est le fluorodésoxyglucose (^{18}F-FDG). C'est un analogue du glucose, c'est-à-dire qu'il possède la même formule chimique à l'exception d'un groupement hydroxyle qui a été remplacé par un atome de fluor 18 (^{18}F), comme on peut le voir sur la figure 1.2.



(a) Glucose (b) ^{18}F-FDG

FIGURE 1.2 – Classiquement utilisé en oncologie, le ^{18}F-FDG est une molécule de glucose dont un groupement hydroxyle a été remplacé par un atome de ^{18}F, un isotope instable du fluor qui se désintègre en en oxygène 18 en émettant un positon dans 97% des cas.

La variation de concentration de cette molécule dans les tissus du patient est proportionnelle à la consommation en glucose. Les tumeurs sont révélées par une consommation anormalement élevée en glucose.

Les radioisotopes utilisés en TEP sont produits principalement en bombardant une cible avec un faisceau de particules chargées accélérées dans un accélérateur linéaire ou un cyclotron. Ils peuvent aussi être obtenus par séparation chimique d'un radionucléide père à longue demi-vie, du radionucléide fils à courte demi-vie, avec des méthodes de chromatographie sur couche mince. Par exemple,

le Germanium 68 (^{68}Ge) se désintègre en gallium 68 (^{68}Ga) avec une demi-vie de 271 jours, ce qui permet une production continue pendant plusieurs mois de ^{68}Ga dont la demi-vie est de seulement 68 minutes, sans avoir à disposer d'un accélérateur de particules dans l'hôpital. Moins d'une dizaine d'isotopes émetteurs de positons sont utilisés comme marqueur en TEP, bien que d'autres isotopes moins conventionnels soient étudiés [Jødal *et al.*, 2014]. Le tableau 1.1 en montre une liste non-exhaustive, avec certaines des propriétés associées et certaines applications. Ces isotopes permettent de marquer un nombre illimité de molécules intervenant dans des fonctions métaboliques variées. Pour cette raison, la TEP est un outil apportant des informations capitales pour le diagnostic de pathologies dans plusieurs branches de la médecine, comme en oncologie, en neurologie ou encore en cardiologie.

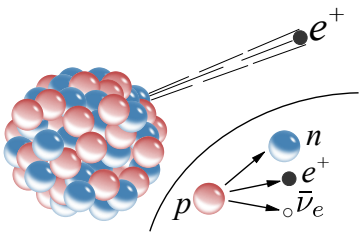
TABLEAU 1.1 – Liste des isotopes les plus communs en TEP et certaines de leurs propriétés et applications [Jødal *et al.*, 2014].

Isotope	Énergie cinétique maximale des positons (keV)	Demi-vie (minutes)	Parcours moyen du positon dans l'eau (mm)	Applications
Fluor 18 (^{18}F)	635	110	0,66	Mesure du métabolisme du glucose (^{18}F-FDG)
Carbone 11 (^{11}C)	970	20	1,26	Marquage de neurotransmetteurs
Oxygène 15 (^{15}O)	1720	2	2,965	Mesure du débit sanguin cérébral
Azote 13 (^{13}N)	1190	10	1,73	Marquage de molécules d'ammoniac pour l'imagerie de perfusion myocardique
Brome 76 (^{76}Br)	3440	960	5,0	Mesure de l'innervation cardiaque
Gallium 68 (^{68}Ga)	1899	68	3,559	Nombreux marquages de molécules possibles pour des applications variées
Rubidium 82 (^{82}Rb)	3378	1.3	7.491	Mesure de la perfusion myocardique

1.2.2 Désintégration β^+ et émission du positon

En TEP la localisation du traceur est possible grâce à la détection des photons γ rayonnants du patient. Mais à la différence de la tomographie d'émission monophotonique (TEMP) ces photons ne sont pas émis directement par le traceur radioactif, ils résultent de l'annihilation des positons émis lors de la désintégration β^+ du noyau de l'isotope instable présent dans le traceur. Cette désintégration, schématisée dans la figure 1.3, repose sur la conversion d'un proton du noyau en neutron

en émettant un neutrino électronique (ν_e) ainsi qu'un positon. Le positon est une particule d'antimatière, un antiélectron, c'est-à-dire qu'il possède les mêmes propriétés physiques qu'un électron à l'exception de sa charge électrique qui vaut +1 au lieu de -1. L'équation 1.1 montre le cas ^{18}F où la transition d'un proton en neutron convertit l'atome de fluor 18 en atome d'oxygène 18 stable. Le neutrino, étant quasiment indétectable, n'est pas utilisé en TEP.



$$^{18}\mathrm{F} \rightarrow {}^{18}\mathrm{O} + e^+ + \nu_e \tag{1.1}$$

FIGURE 1.3 – La désintégration β^+ repose sur la conversion d'un proton en neutron donnant lieu à l'émission d'un neutrino électronique et d'un positon.

1.2.3 Annihilation du positon

Le positon émis par la désintégration β^+ récupère une partie de l'énergie de transition du proton en neutron sous la forme d'énergie cinétique. Cette énergie lui permet de parcourir quelques millimètres, voire quelques centimètres, pendant lesquels les interactions avec la matière traversée vont le ralentir jusqu'à se trouver quasiment au repos. Lorsque sa vitesse est suffisamment faible le positon peut se combiner et s'annihiler avec un électron, donnant naissance à une paire de photons γ colinéaires et de sens opposé (dans le référentiel du barycentre du couple positon-électron), chacun avec une énergie égale à 511 keV (figure 1.4).

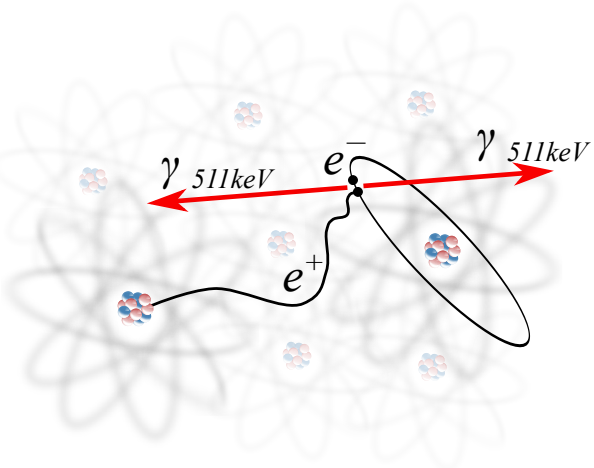


FIGURE 1.4 – Émission d'un positon et annihilation avec un électron d'un atome de la matière environnante après un parcours de quelques millimètres à quelques centimètres.

1.2.3.1 Parcours du positon avant annihilation

En TEP l'information que l'on cherche à visualiser est la répartition de traceur dans l'espace. Cependant, la détection des photons γ d'annihilations donne une information sur la position d'annihilation d'un positon, ce qui ne correspond pas exactement à celle de l'émission de ce positon et donc à celle du traceur. Ainsi, plus la distance entre la position d'émission (celle du traceur) et la position d'annihilation (celle de la détection) est importante, plus la résolution spatiale des images reconstruites est détériorée. On appelle cet effet le parcours du positon.

Cette distance n'est pas une valeur fixe et varie considérablement en fonction du traceur utilisé et du matériau dans lequel le positon se propage. L'impact du traceur provient de la variation de l'énergie de transition proton-neutron qui est transmise au positon. Pour un radionucléide donné, cette énergie suit un spectre spécifique. Les spectres d'énergies cinétiques des positons émis par des radioisotopes communs sont présentés dans la figure 1.5. On peut voir que le ^{18}F a un spectre qui ne permet pas l'émission de positon avec une énergie supérieure à 600 keV, alors que le ^{82}Rb a un spectre qui s'étale jusqu'à plus de 3.5 MeV. Le tableau 1.1 montre que cela impacte directement le parcours du positon moyen dans un matériau donné.

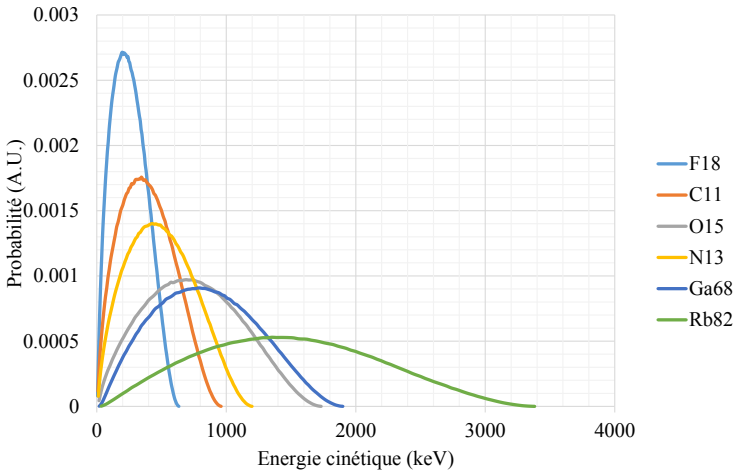
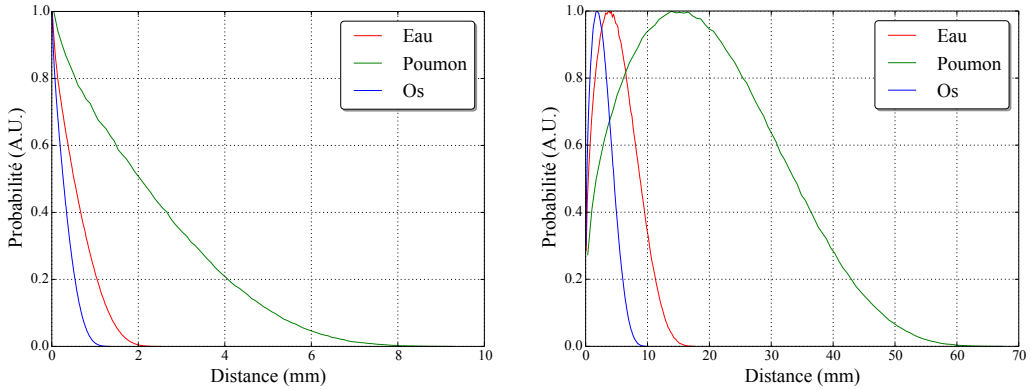


FIGURE 1.5 – Énergies cinétiques des positons émis par différents traceurs.

Dans le cas des molécules marquées au ^{18}F, comme le ^{18}F-FDG, le parcours du positon est en général négligé car il est largement inférieur à la résolution intrinsèque des scanners TEP cliniques actuels. Par contre, avec d'autres isotopes comme le ^{82}Rb les positons peuvent parcourir plus d'un centimètre avant de s'annihiler, ce qui n'est plus négligeable face à la résolution du scanner et entraîne une dégradation significative des images reconstruites. La densité de la matière dans laquelle se propage un positon a aussi un impact important sur la distance parcourue avant qu'il ne s'annihile. La figure 1.6 montre les distributions des distances entre la position d'émission et celle d'annihilation des positons émis par du ^{18}F et du ^{82}Rb. Ces distributions ont été estimées dans trois matériaux de densités différentes, de l'eau (densité moyenne), du poumon (densité faible) et de l'os (densité élevée).



(a) ^{18}F (b) ^{82}Rb

FIGURE 1.6 – Distance parcourue par des positons jusqu'à annihilation pour (a) le ^{18}F et (b) le ^{82}Rb dans de l'eau, du poumon et de l'os.

1.2.3.2 Non-colinéarité des photons d'annihilation

Lors d'une annihilation, le couple électron-positon possède un barycentre qui n'est généralement pas au repos. La loi de conservation du moment cinétique impose alors à la paire de photons d'annihilations de posséder le même moment cinétique que le couple électron-positon juste avant annihilation. Pour une annihilation avec un centre de gravité électron-positon au repos, la paire de photons est émise avec un angle de 180°, garantissant un moment cinétique nul. En pratique, ce centre de gravité n'est pas au repos, comme cela a été mesuré dans [Colombino *et al.*, 1965]. Les travaux de [Levin et Hoffman, 1999] ont estimé l'angle de non-colinéarité suit est distribué aléatoirement selon un loi normale avec une largeur à mi-hauteur ou *full width at half maximum* en anglais (*FWHM*) de 0,25°. Cette non-colinéarité implique une imprécision sur la localisation de l'annihilation, qui ne se trouve pas exactement sur la ligne reliant les deux points de détection des photons γ mais légèrement à côté, comme cela est représenté dans la figure 1.7. L'incertitude qui en résulte possède une *FWHM* de $0.0022 \times d_s$ au centre du champ de vue d'un scanner dont le diamètre vaut d_s. Cet effet est souvent négligé en raison de son impact relativement mineur face à la résolution intrinsèque du scanner. Par exemple, il implique une incertitude dont la *FWHM* est de 2,2 mm au centre du champ de vue d'un scanner de 1 mètre de diamètre.

1.2.4 Parcours des photons d'annihilation jusqu'au détecteur

Les photons, issus de l'annihilation du positon avec un électron, vont se propager dans la matière qui compose le patient et son environnement (lit, vêtements, parties du scanner . . .) avant d'atteindre le détecteur. Les photons γ interagissent de cinq manières différentes avec la matière. Leurs sections efficaces dans l'eau, en fonction de l'énergie du photon, sont représentées dans la figure 1.8. La sections efficaces est une grandeur physique associée la probabilité d'interaction d'une particule (ici des photons) pour une réaction donnée.

La création de paires fait référence à la production d'un couple particule-antiparticule, nécessitant une énergie minimale au moins deux fois supérieure à l'énergie de masse au repos de la paire générée. La paire la plus légère qui puisse être générée est celle constituée d'un électron et d'un posi-

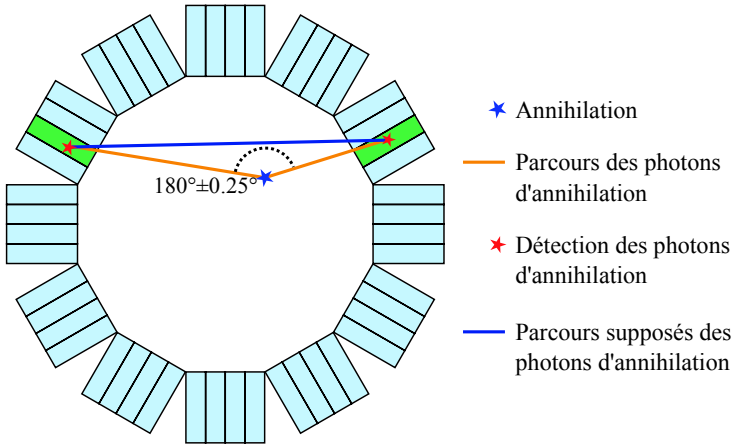


FIGURE 1.7 – Illustration de l'erreur induite par la non-colinéarité des photons d'annihilation. Le segment bleue représente la ligne où on suppose que l'annihilation s'est produite et les segments orange montre les parcours des photons d'annihilation.

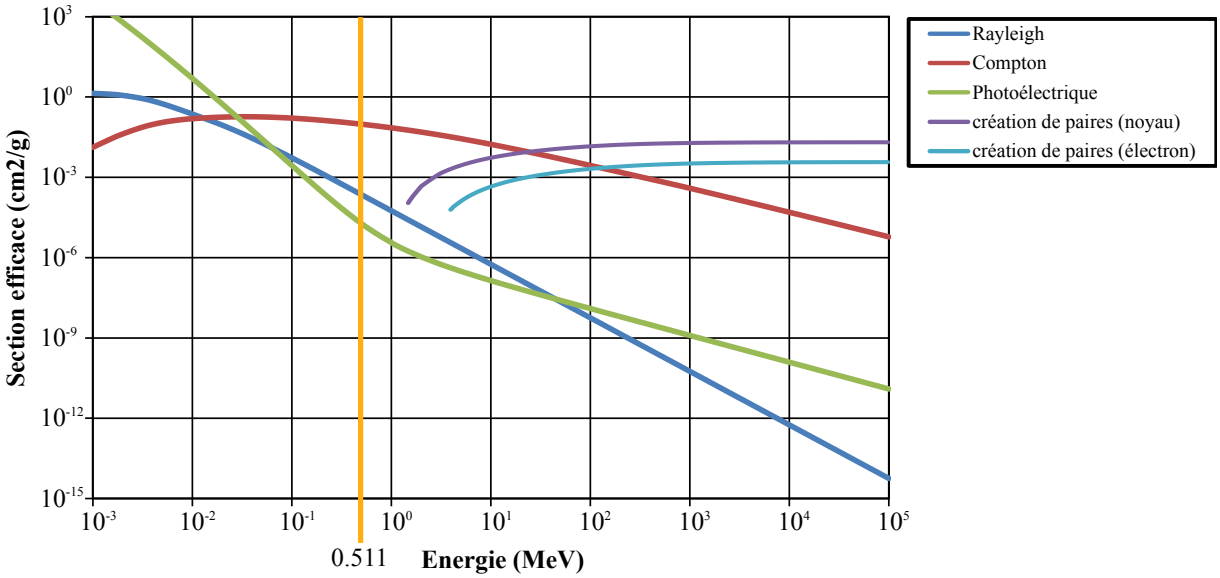


FIGURE 1.8 – Sections efficaces des photons avec l'eau pour les cinq interactions physiques possibles. La ligne orange verticale indique 511 keV qui est l'énergie initiale des photons d'annihilation. On voit qu'à cette énergie la section efficace de l'effet Compton est entre deux et trois ordres de grandeur plus importante que celle des effets photoélectrique et Rayleigh.

ton dont l'énergie de masse au repos est de 1,022 MeV, c'est à dire le double de l'énergie des photons d'annihilation. La création de paires n'intervient donc pas en TEP. Les trois autres effets, la diffusion Compton, la diffusion Rayleigh et l'effet photoélectrique, interviennent pour les photons à 511 keV, bien que l'effet Compton soit prédominant pour cette énergie et dans les matériaux biologiques.

La diffusion Compton naît de la collision d'un photon avec un électron libre d'un atome. Cette diffusion entraîne l'éjection de l'électron de l'atome (qui se retrouve ionisé) et la diffusion du photon avec perte d'énergie. L'énergie résultant E peut être calculée en appliquant les principes de conserva-

tion de l'énergie et de la quantité de mouvement et est donnée par l'équation suivante :

$$E = \frac{E_0}{1 + \alpha(1 - \cos\theta)} \tag{1.2}$$

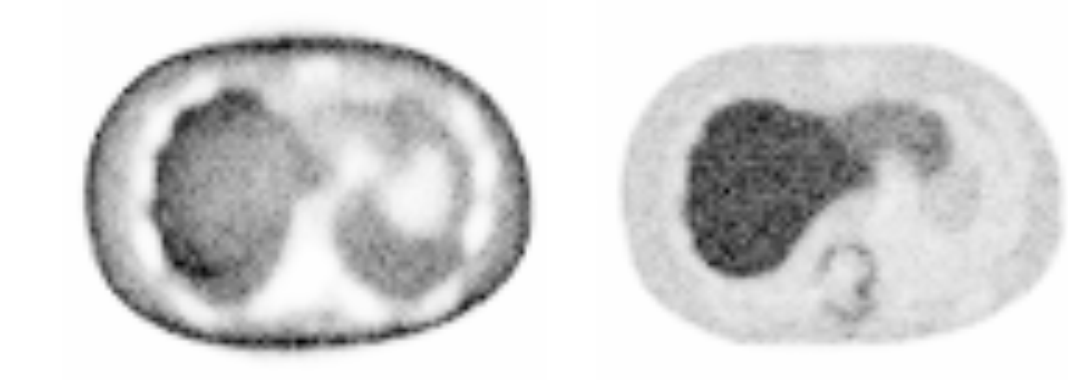
Plus l'angle de diffusion est important et plus l'énergie perdue par le photon est importante. Par conséquent les photons diffusés, avec un angle trop important ou plusieurs fois, peuvent atteindre le détecteur avec une énergie résultante trop faible pour être détectée, ont dit alors qu'ils sont atténués.

L'effet photoélectrique se caractérise par l'absorption complète d'un photon par un électron qui se trouve éjecté de l'atome. Les photons ne peuvent arracher un électron que si leur énergie est supérieure à l'énergie de liaison de celui-ci. Pour les atomes légers, principaux composants de la matière organique, l'énergie de liaison des électrons étant faible face à celle des photons d'annihilation, ceux-ci leur semblent libres, c'est donc l'effet Compton qui prédomine. En revanche, pour les atomes lourds, comme le plomb, l'énergie de liaison des électrons des couches internes est beaucoup plus importante et dans ce cas l'effet photoélectrique prédomine. Dans le corps du patient, les absorptions par effet photoélectrique se produisent principalement dans les os en raison de la présence de calcium et de phosphore.

La diffusion Rayleigh est due à la polarisation d'un atome ou d'une molécule par le champ électrique du photon incident, formant un dipôle électrostatique rayonnant une onde électromagnétique, qui constitue le rayonnement diffusé. La perte d'énergie du photon suit l'équation 1.2 mais cette fois la particule diffusante est soit un atome, soit une molécule. Le calcul du coefficient α ne se fait plus avec la masse d'un électron, mais la masse d'un atome ou d'une molécule. Pour un électron et une énergie de 511 keV la valeur du coefficient α est de l'ordre de grandeur de 1, tandis que pour un atome de carbone ou d'oxygène cette valeur est de l'ordre de 10^{-8}. Le second terme du dénominateur de l'équation 1.2 étant au maximum du même ordre de grandeur que α, il est alors lui aussi de l'ordre de 10^{-8}, ce qui est négligeable face à 1. La diffusion Rayleigh se fait donc avec une perte d'énergie négligeable et n'importe quel photon diffusé par ce processus restent détectables, quel que soit l'angle de diffusion.

Ces trois types d'interactions causent deux effets, l'atténuation et la diffusion. Une paire de photons d'annihilation est considérée comme atténuée si au moins un des deux photons a perdu trop d'énergie pour être détecté. Nous avons vu que seul l'effet Compton et l'effet photoélectrique implique une perte d'énergie (totale dans le deuxième cas). Ce sont donc ces deux effets qui sont responsables de l'atténuation. En fonction de l'épaisseur et du type de matière que les photons d'annihilations vont devoir traverser, une fraction variable d'entre eux vont être atténués (diffusés ou absorbés) le long de leurs trajets jusqu'au détecteur. Les annihilations qui se produisent à la surface du corps et dont les photons d'annihilation se propagent dans des directions tangentes à cette surface ont peu de matière à traverser et sont donc surreprésentées par rapport à ceux émis au centre du corps. Cela va avoir pour effet de surestimer la concentration de traceur à la surface de l'objet imagé, on peut l'observer dans la figure 1.9a. On appelle cela l'effet de peau. Plus généralement, cette atténuation implique une variation de la sensibilité de chaque paire de cristaux en fonction de la distribution de la matière dans le champ de vue du scanner, qui vient s'ajouter à la sensibilité intrinsèque de cette

paire. L'effet Compton étant prédominant dans les tissus organiques à l'énergie de 511 keV, c'est principalement lui qui est responsable de l'atténuation.



(a) (b)

FIGURE 1.9 – Coupe transverse de reconstructions (a) sans et (b) avec correction d'atténuation. Sans correction d'atténuation, l'activité en périphérie du corps est surestimée.

La détection d'une coïncidence dont au moins un des deux photons d'annihilation a été diffusé peut entraîner des erreurs importantes, comme représenté dans la figure 1.10. On appelle ce type d'événement une coïncidence diffusée. Cette diffusion résulte de l'effet Compton et la diffusion Ray-

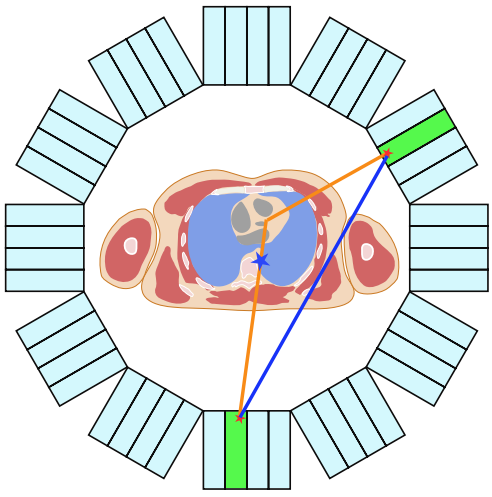


FIGURE 1.10 – Les photons γ (trajectoire en orange) peuvent être diffusés par le corps du patient avant d'atteindre le détecteur et conduire à la détection d'une coïncidence selon la *LOR* (en bleue) passant loin de la position d'émission du positon.

leigh. Nous avons vu que la diffusion Compton est la source principale d'atténuation. Cependant, pour des angles de diffusion faibles, les photons conservent une énergie suffisante pour être détecté comme des photons non diffusés en raison de la résolution énergétique limitée du détecteur, que nous présenterons dans le paragraphe 1.2.5.5. La diffusion est un problème majeur, car les coïncidences diffusées représentent entre 40% et 60% des coïncidences détectées [Thompson, 1993] et l'angle de diffusion des photons est aléatoire.

1.2.5 Détection des photons d'annihilation

1.2.5.1 Principes

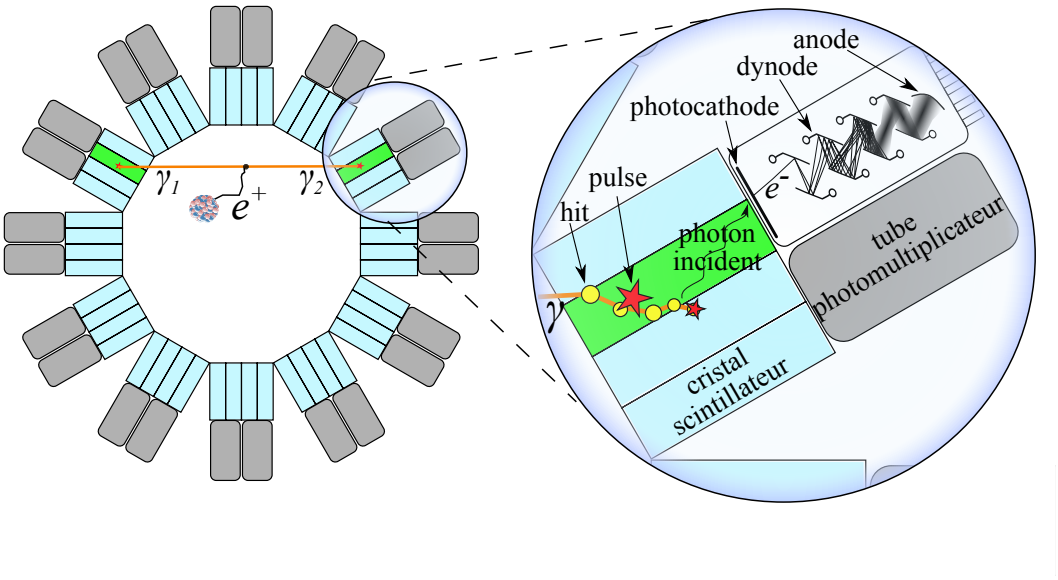


FIGURE 1.11 – La détection des photons d'annihilation repose sur deux étapes, la conversion des photons d'annihilation en photon optique par les cristaux scintillateurs et la conversion des photons optiques en signal électrique par les TPM ou SiPM.

Le détecteur prend la forme d'un cylindre placé autour du patient afin de détecter les photons γ issus de l'annihilation des positons dans le corps du patient. Cet anneau, représenté dans la figure 1.11, est composé de deux couches. La première est formée de blocs de cristaux scintillateurs convertissant les photons d'annihilation en photons optiques ou ultraviolets par fluorescence. Les cristaux communément utilisés sont inorganiques, par exemple le $LuSiO_5$ (LSO) et le Gd_2SiO_5 (GSO). Dans un solide, les électrons ne peuvent avoir des énergies que dans des intervalles spécifiques que l'on appelle des bandes. La bande de valence est celle située juste en dessous du niveau de Fermi, qui est l'énergie maximale des électrons lorsque le solide est à 0 kelvin. La bande de conduction est celle qui est juste au-dessus du niveau de Fermi. La conversion d'un photon γ par les cristaux repose sur l'excitation d'un électron de la bande de valence, par effet Compton ou photoélectrique, qui se voit transférer vers la bande de conduction en laissant un trou dans la bande de valence. Si l'électron est transféré dans la bande de conduction avec peu d'énergie, il va former une pseudo-particule qu'on appelle un exciton. Cet exciton est composé d'une paire électron-trou faiblement liée, qui peut errer dans le réseau cristallin jusqu'à être capturé par un centre d'impureté. Ce dernier va se désexciter rapidement en émettant de la lumière de scintillation. C'est la composante rapide de la scintillation. Pour les électrons émis dans la bande de valence avec plus d'énergie, les électrons et les trous vont

être capturés par des centres d'impuretés et exciter des états métastables inaccessibles aux excitons. La désexcitation retardée de ces états métastables entraîne aussi l'émission d'un rayonnement de scintillation. C'est une composante lente de la scintillation.

Les photons de scintillation ainsi générés se propagent jusqu'à la seconde couche du détecteur, composée de tubes photomultiplicateurs (TPM) ou photomultiplicateurs en silicium (SiPM) permettant de convertir les photons optiques en signaux électriques mesurables. Le principe de fonctionnement des TPM et SiPM repose en premier lieu sur l'absorption des photons optiques par effet photoélectrique, au niveau d'une photocathode constituée de métal ou de semi-conducteur, engendrant une émission d'électrons. Cependant, ceux-ci créent un signal électrique trop faible pour qu'on puisse le détecter. Pour amplifier ce signal, une série de dynodes est placée entre la photocathode, chargée négativement, et l'anode, chargée positivement. De la photocathode vers l'anode, chaque dynode est chargée plus positivement que la précédente. L'électron arraché va être accéléré en direction de la première dynode par le champ électrique généré par la différence de potentiel. La collision avec la dynode va engendrer l'émission de plusieurs électrons qui, à leur tour, vont être accélérés vers la dynode suivante. Ce phénomène va se répéter de dynode en dynodes, générant à chaque fois plus d'électrons jusqu'à l'anode où l'ensemble des électrons va être absorbé en créant un courant électrique mesurable. La géométrie de la chaîne de dynodes est construite de telle sorte que le nombre d'électrons émis entre chaque dynode augmente de manière exponentielle. Par exemple, si à chaque dynode, un électron en arrache en moyenne 5 électrons et que la chaîne contient 12 dynodes, 1 seul électron émis de la photocathode va produire $\approx 10^8$ électrons au niveau de l'anode. Dans un SiPM l'amplification se fait par effet d'avalanche, c'est-à-dire qu'un champ électrique intense est appliqué à un semi-conducteur et à chaque fois qu'un électron percute un de ses atomes, plusieurs électrons sont arrachés puis accélérés par le champ électrique. Le même processus se répète avec les électrons ainsi générés jusqu'à l'anode.

Lorsqu'un photon γ est totalement absorbé par un cristal, le rayonnement de scintillation résultant est composé d'un ensemble de photons dont la somme des énergies est proportionnelle à l'énergie initiale du photon γ. Le courant généré par un TPM ou SiPM pour un photon est proportionnel à son énergie. Le courant généré par tous les photons de scintillation d'un photon γ incident permet d'estimer son énergie et donc de discriminer les photons d'annihilation à 511 keV des autres (photons diffusés ou produits par d'autres processus qu'une annihilation électron-positon).

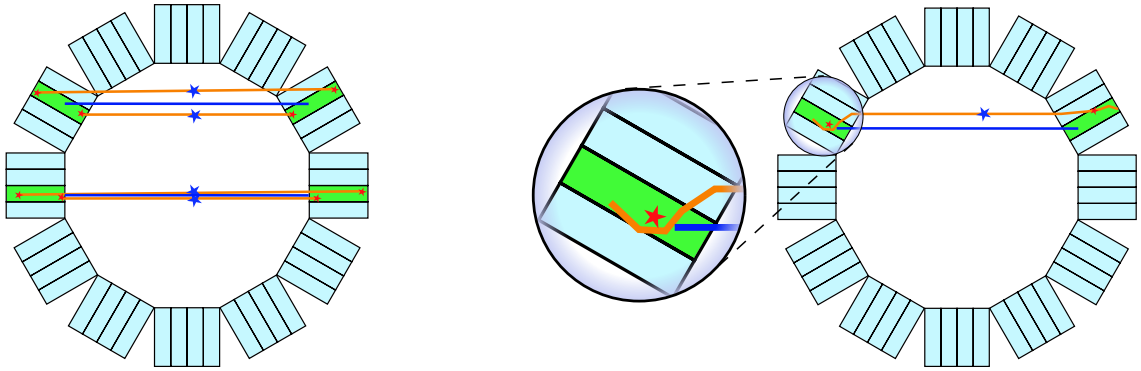
1.2.5.2 Variations de la sensibilité

Chaque couple de cristaux du détecteur possède une sensibilité propre qui peut être source d'erreurs de quantification dans les reconstructions. Cette variation est due à deux effets. Premièrement, la variation de sensibilité intrinsèque aux deux cristaux, liée à la position des cristaux par rapport aux TPM, aux gains des TPM ainsi qu'à des variations physiques dans les cristaux. Cette valeur est susceptible d'évoluer avec le temps est nécessite d'être mesurée périodiquement. Deuxièmement, la variation de sensibilité géométrique liée à la géométrie du détecteur. Un couple de cristaux donné va avoir une certaine sensibilité géométrique qui dépend de l'orientation relative des cristaux et des parties du détecteur (cristaux voisins et septa) pouvant atténuer les photons. Cette deuxième com-

posante est moins susceptible d'évoluer dans le temps.

1.2.5.3 Effet de parallaxe

Afin de détecter efficacement les photons d'annihilation, l'anneau de cristaux placé autour du patient est conçu de telle sorte qu'il absorbe la majorité des photons l'atteignant. Pour répondre à cette contrainte, il y a deux solutions, créer des cristaux scintillateurs dans un matériau très atténuant pour les photons γ ou avoir un détecteur épais. Les cristaux classiques comme le LSO ou le GSO absorbent 50% des photons en un peu plus de 1 centimètre [Nikolopoulos *et al.*, 2006]. Pour que le scanner ait une sensibilité suffisante, les cristaux sont très allongés, typiquement 2 cm pour les scanners corps entier et 1 cm pour les scanner pré cliniques, par rapport aux autres dimensions, comprises entre 6 et 1 mm. Pour les paires de cristaux détectant des *LOR* passant loin du centre du scanner, l'allongement des cristaux implique que les *LOR* peuvent être comprises dans un intervalle de positions plus grand, comme illustré sur la figure 1.12a. C'est ce qu'on appelle l'effet de parallaxe, qui est responsable d'une plus grande incertitude sur la position d'une *LOR* détectée sur le bord du champ de vue qu'une *LOR* détectée au centre. L'effet de parallaxe est aussi appelé pénétration intercristaux, parce qu'il résulte de la pénétration des photons γ dans le détecteur, de telle sorte qu'ils traversent plusieurs cristaux avant d'être détectés.



(a) Effet de parallaxe (b) Diffusion dans le détecteur

FIGURE 1.12 – En (a) la longueur des cristaux entraîne des effets de parallaxe lorsque l'on s'éloigne du centre du champ de vue. En (b) la diffusion des photons γ dans le détecteur peut entraîner un décalage de la *LOR* détectée.

1.2.5.4 Diffusion dans le détecteur

Lorsqu'un photon γ atteint un cristal scintillateur, celui-ci peut subir en général plusieurs diffusions Compton avant d'être absorbé par effet photoélectrique ou d'en sortir. Chaque interaction avec le cristal émet des photons optiques convertis en un signal électrique dont l'intensité est proportionnelle à l'énergie perdue par le photon γ dans la diffusion. L'ensemble de ces interactions, et des émissions de photons optiques associées, se produisent dans un délai si court que le détecteur ne perçoit que la somme de toutes ces interactions. Il est alors possible qu'un photon, lors de son parcours, interagisse avec plusieurs cristaux et dévie significativement de sa trajectoire initiale, comme le montre la figure 1.12b. On nomme cet effet la diffusion intercristaux, parce qu'elle résulte de la dif-

fusion des photons γ dans le détecteur, qui les dévie et les conduit à être détectés dans des cristaux ne se trouvant pas sur leurs trajectoires initiales.

1.2.5.5 Résolution énergétique limitée

Le système composé d'un cristal scintillateur et d'un TPM ou SiPM a une résolution énergétique limitée. L'erreur associée dépend de l'énergie mesurée et est distribuée suivant une loi normale. Pour des cristaux en GSO avec des TPM, elle a une *FWHM* qui se situe autour de 15% à 511 keV [Balcerzyk *et al.*, 2000]. Comme nous l'avons dit, la discrimination des photons d'annihilation directs et des photons diffusés ou produits par d'autres processus physiques est faite grâce à l'énergie mesurée qui doit être de 511 keV. Étant donnée la résolution énergétique limitée, les énergies mesurées des photons d'annihilation directs vont être distribuées autour de 511 keV et donc se mélanger avec les énergies des autres photons. Une fenêtre d'énergie est fixée afin de discriminer les photons d'annihilation directs des autres. Les seuils de cette fenêtre doivent être choisis avec précaution pour sélectionner le plus de photons d'annihilation directs et le moins de photons diffusés ou générés par d'autres processus physiques.

1.2.5.6 La profondeur d'interaction

La profondeur d'interaction ou *depth-of-interaction* en anglais (*DOI*) correspond à la position le long de l'axe du cristal du barycentre de toutes les émissions de photons de scintillation générés par un photon γ. Sans position d'interaction ou *position-of-interaction* en anglais (*POI*) la seule information de localisation du photon est la position du cristal. La *POI* quant à elle donne la position complète dans le cristal. La forme allongée des cristaux entraîne des effets de parallaxe importants lorsqu'on s'éloigne du centre du champ de vue du scanner.

Actuellement, il n'existe aucun scanner commercial permettant de mesurer cette information et seulement quelques prototypes le peuvent [Moses et Derenzo, 1994, Moses *et al.*, 1995, Ling *et al.*, 2007].

1.3 Chaîne de traitement des événements détectés et formation des coïncidences

1.3.1 Principes

Les signaux électriques générés par les TPM ou SiPM subissent une série de traitements afin de déterminer quand un photon d'annihilation a été détecté et quand deux photons détectés sont issus d'une même annihilation. On distingue quatre niveaux dans les événements détectés. Premièrement, les *hits* qui sont les interactions uniques des photons γ avec les cristaux. Les détecteurs actuels, de par leur résolution temporelle et le temps de relaxation des cristaux scintillateurs, ne permettent pas de mesurer chacun de ces événements séparément. On commence cependant à voir émerger des prototypes de scanner permettant d'avoir des informations sur les *hits* [Raczyński *et al.*, 2014]. Deuxièmement, les *pulses* qui correspondent aux sommes des *hits* produits dans un même cristal

et dans un laps de temps suffisamment court pour qu'on puisse considérer qu'ils aient été générés par le même photon γ. Troisièmement, les *singles* qui sont les sommes des *pulses* produites parmi un groupe de cristaux défini. Ils permettent de détecter les photons interagissant avec plusieurs cristaux. Ce sont généralement ces événements qui sont détectés par le scanner. Enfin, les coïncidences qui sont constituées de deux *singles* dont les énergies sont comprises dans la fenêtre d'énergie et ayant été détectés dans un intervalle de temps donné, qu'on appelle la fenêtre de coïncidence et qui correspond à la résolution temporelle du système. Cette chaîne de détection et de traitements est schématisée dans la figure 1.13.

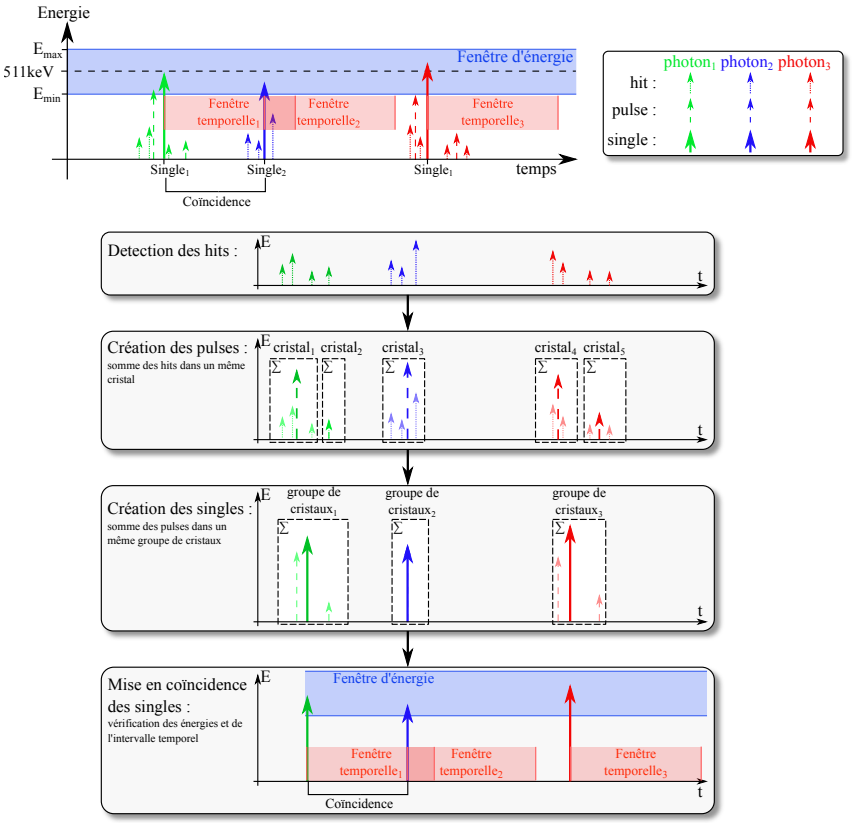


FIGURE 1.13 – Schéma des traitements appliqués aux événements détectés pour former les coïncidences. Ici, trois photons (une couleur par photon) génèrent des *hits* qui, sommés par cristaux, forment des *pulses* qui, à leur tour sommées par blocs de cristaux, forment des *singles*. Quand un single est détecté, la fenêtre de coïncidence s'ouvre et tout autre *single* détecté pendant cet intervalle de temps lui est associé pour former une coïncidence, si les deux *singles* rentrent dans la fenêtre d'énergie.

1.3.2 Temps-mort

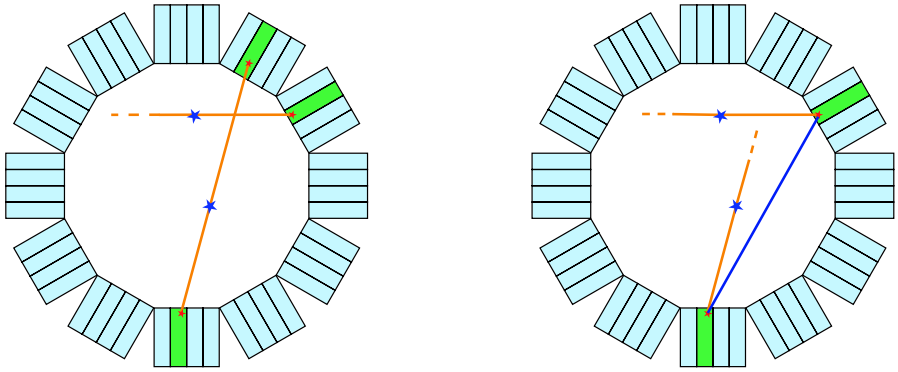
Tous les systèmes de détection ont une vitesse limite avec laquelle les événements détectés peuvent être traités. Ainsi, lors d'une acquisition en TEP, l'électronique qui traite les impulsions détectées a une fréquence de comptage maximale qui est de l'ordre du MHz. Lorsqu'une impulsion est détectée, le système reste inactif pendant un certain laps de temps avant de pouvoir à nouveau détecter une impulsion. Une deuxième source de temps-mort est due au temps de décroissance de scintillation des cristaux. En effet, lorsqu'un photon γ interagit avec un cristal scintillateur, des photons optiques

sont émis pendant un certain temps qui dépend du type de cristal utilisé. Ce temps est appelé période réfractaire. Ainsi, lorsque plusieurs photons atteignent un même bloc de cristaux pendant cette période réfractaire, il n'est pas possible de séparer les impulsions. On dit que le système est paralysé.

Le temps-mort a pour effet de saturer le système, d'autant plus que la fréquence d'apparition des impulsions sera importante, réduisant le nombre de coïncidences détectées.

1.3.3 Coïncidences fortuites

Dans la chaîne de traitements permettant de former les coïncidences, il est possible que deux positons (ou plus) soient émis dans un intervalle de temps suffisamment court pour être détectés dans la même fenêtre de coïncidence. Dans un tel cas de figure, plusieurs scénarios sont possibles. Si, comme dans la figure 1.14a, plus de deux photons d'annihilation sont détectés, on obtient une coïncidence fortuite multiple. Il est facile de discriminer ce type de coïncidences simplement parce qu'elles impliquent plus de deux détections. Cependant, il existe un second type de coïncidences fortuites qui lui ne peut pas être discriminé, les coïncidences fortuites simples, illustré dans la figure 1.14b. Avec une coïncidence de ce type, deux photons γ provenant de deux annihilations distinctes sont détectés en coïncidence tandis que les deux autres photons ne sont pas détectés, à cause de l'atténuation ou de la sensibilité imparfaite du détecteur.



(a) Coïncidence fortuite multiple (b) coïncidence fortuite simple

FIGURE 1.14 – Les deux types de fausses coïncidences générées par la production et l'annihilation de deux positons dans la même fenêtre temporelle sont (a) les coïncidences multiples où plus de deux photons sont détectes et (b) les coïncidences fortuites où deux photons provenant de deux annihilations différentes sont détectés.

1.3.4 Le temps de vol

L'amélioration de la résolution temporelle des scanners TEP a permis, en plus d'améliorer le taux de coïncidences fortuites détectées, de mesurer l'écart de temps d'arrivée des photons d'annihilations sur le détecteur, ce qu'on appelle le temps de vol ou *time-of-flight* en anglais (*TOF*). Connaissant la vitesse de propagation des photons γ il est possible à partir de cette information de *TOF* de déterminer la position de l'annihilation le long de la *LOR* avec l'équation suivante :

$$X = \frac{x_1 + x_2}{2} + c\frac{t_1 - t_2}{2} \tag{1.3}$$

où, X est la position de l'annihilation sur la ligne liant les cristaux dont les positions sont x_1 et x_2, ayant détecté les photons d'annihilation respectivement à l'instant t_1 et t_2. c est la vitesse de la lumière dans le vide, qu'on suppose être la vitesse de propagation des photons d'annihilation.

Cette information, avec une résolution temporelle suffisante, permettrait de se passer du processus de reconstruction pour obtenir les images de la répartition spatiale du traceur. La résolution spatiale de la position d'annihilation le long de la *LOR* est liée à la résolution temporelle par l'équation suivante :

$$\Delta x = \frac{c}{2}\Delta t \tag{1.4}$$

où, Δx est la *FWHM* spatiale, Δt est la *FWHM* temporelle et c la vitesse de la lumière. Ces informations sont représentées dans la figure 1.15. Le prototype testé dans [Miller *et al.*, 2014] atteint une résolution de 307 picosecondes qui donne une résolution spatiale de 46 mm. Cependant, cela reste insuffisant pour se passer du processus de reconstruction.

Tant que la résolution temporelle des scanners TEP sera supérieure à quelques dizaines de picosecondes, il ne sera pas possible de se passer de la reconstruction.

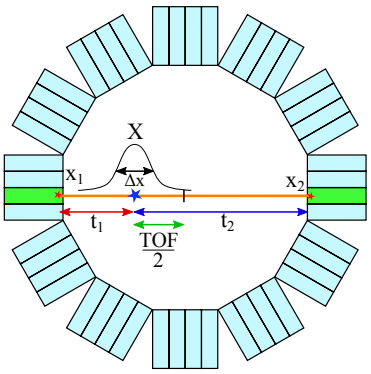


FIGURE 1.15 – Représentation de l'information de *TOF*. Le *TOF* mesure la différence de temps d'arrivée dans le détecteur des deux photons d'annihilation.

1.3.5 Les modes de stockage des coïncidences

Les coïncidences détectées durant une acquisition TEP sont stockées en vue de reconstruire la répartition de traceur dans le patient. Plusieurs modes existent avec chacun, différents avantages et inconvénients.

1.3.5.1 *List-mode*

Le stockage en *list-mode* (mode liste) est un format de données brutes où les informations des coïncidences détectées sont stockées séquentiellement dans un fichier dont la taille va croître avec le temps d'acquisition. Ce mode a l'avantage de conserver l'intégralité des données disponibles sans perte comme l'échantillonnage temporel, pouvant être utile aux corrections des phénomènes dynamiques, comme les mouvements respiratoires. Il permet aussi de conserver l'échantillonnage spatial, permettant une meilleure résolution de la reconstruction [Rahmim *et al.*, 2004]. Les informations

comme la *DOI*, le *TOF* et l'énergie d'intéraction ne sont pas dégradées, ce qui permet de les exploiter sans perte de précision. En revanche, les fichiers associés à ce format ont l'inconvénient d'être de tailles variables et d'être volumineux. Ces dernières années, l'augmentation des capacités de stockage et de traitement des ordinateurs, a permis à ce mode de se développer.

1.3.5.2 Mode histogramme

Le mode histogramme prend la forme d'un long vecteur à une dimension dont chaque coefficient est associé à un couple de cristaux et correspond au nombre de coïncidences détectées par celui-ci. Ce mode a l'avantage de fournir un fichier de taille fixe qui dépend du nombre de paires de cristaux. Un tel fichier a une taille importante mais peut facilement être compressé en ne stockant pas les coefficients nuls, qui sont généralement très nombreux. Par contre, l'ensemble des informations annexes comme le *TOF*, l'énergie d'interaction, la *DOI* et la *POI*, pouvant servir à certaines corrections ou certaines améliorations des images reconstruites doivent être échantillonnées ce qui peut induire des pertes d'information (à moins de respecter le théorème de Nyquist-Shannon). De plus, il est nécessaire d'ajouter une dimension à l'histogramme pour chaque information supplémentaire prise en compte, ce qui induit une augmentation de la taille du fichier histogramme.

1.3.5.3 Mode sinogramme

Le mode sinogramme est différent des deux modes précédents par le fait, qu'une fois construit, il ne dépend plus de la géométrie du scanner. Par ailleurs, il permet une visualisation des données sans reconstruction volumique. Un sinogramme se présente sous la forme d'un ensemble de projections dont la valeur en chaque pixel représente le nombre de *LOR* détectées dans un certain intervalle angulaire centré en θ, φ et un certain intervalle de position centré en s, z_{coupe}, qu'on appelle *bin*, comme défini dans la figure 1.16 pour un sinogramme 3D. La construction d'un sinogramme se fait en passant par une étape nommée *binning*.

Ce mode a l'avantage de fournir une visualisation des données acquises, ainsi qu'un fichier de taille fixe et généralement plus compact que les deux modes précédents. En revanche, il a l'inconvénient d'induire une perte de résolution plus ou moins importante en fonction de la taille des *bins* et il ne permet pas la conservation sans perte de toutes les informations liées à chaque coïncidence. Chaque information supplémentaire, comme le *TOF*, l'instant d'occurrence des coïncidences ou l'énergie d'interaction, doit être échantillonnée et nécessite l'ajout d'une dimension supplémentaire au sinogramme qui peut conduire à une consommation importante de mémoire.

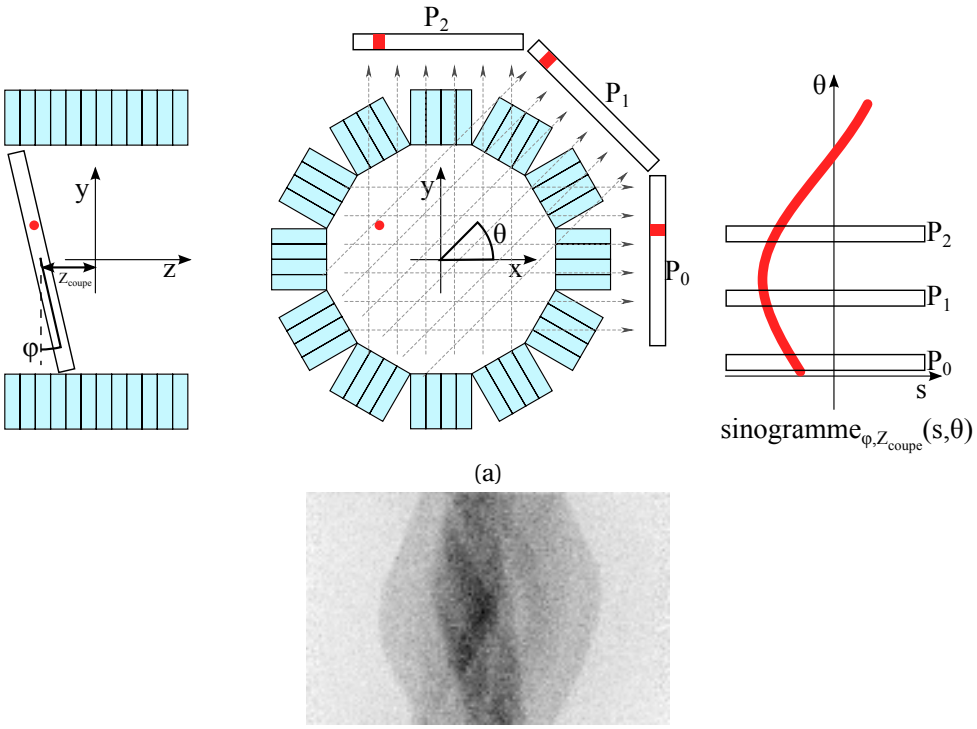


FIGURE 1.16 – (a) représente le système de coordonnées d'un sinogramme et (b) un exemple de sinogramme pour une position z_{coupe} et un angle φ donnés.

1.4 La reconstruction en tomographie par émission de positons

Les données mesurées par le scanner TEP, qu'elles soient en *list-mode*, histogramme ou sinogramme, nécessitent de passer par une étape de reconstruction permettant d'obtenir une estimation de la distribution spatiale du traceur dans le champ de vue du scanner, exploitable par les cliniciens. Le problème de la reconstruction en TEP est un sujet qui, depuis les années 1970 [Brownell *et al.*, 1971], a été à l'origine d'un grand nombre de publications et d'algorithmes. Dans cette partie, nous allons commencer par présenter des algorithmes de reconstruction analytiques qui proposent une inversion directe des équations du système. Ensuite, nous présenterons des algorithmes itératifs, qui répondent mieux aux contraintes de la TEP que les approches analytiques. Pour finir, nous verrons comment les différents effets intervenants pendant l'acquisition des données peuvent être pris en compte dans la reconstruction pour fournir des images reconstruites qui soient qualitatives et quantitatives.

1.4.1 Reconstruction analytique

Historiquement, le problème de la reconstruction en TEP a, dans un premier temps, été résolu avec des méthodes analytiques. Tous ces algorithmes reposent sur l'hypothèse que chaque *pixel*, des sinogrammes mesurés, donne une mesure de l'intégrale de la fonction mathématique décrivant la répartition spatiale du traceur radioactif, le long de la ligne de l'espace associée à ce *pixel*.

1.4.1.1 Reconstruction 2D

Les premiers scanner TEP fonctionnaient en mode 2D, c'est-à-dire que seules les coïncidences formées entre deux cristaux d'un même anneau du détecteur ou deux anneaux adjacents, étaient considérées comme valides. Pour limiter le nombre de coïncidences ne satisfaisant pas cette contrainte, des septa étaient placés entre les anneaux de détection, permettant de bloquer les photons d'annihilation dont la direction s'éloigne du plan perpendiculaire à l'axe du scanner, comme le montre la figure 1.17. Les coïncidences d'un même anneau permettent de former ce qu'on appelle un sinogramme droit, quant à celles entre deux anneaux adjacents, elles forment un sinogramme quasi droit.

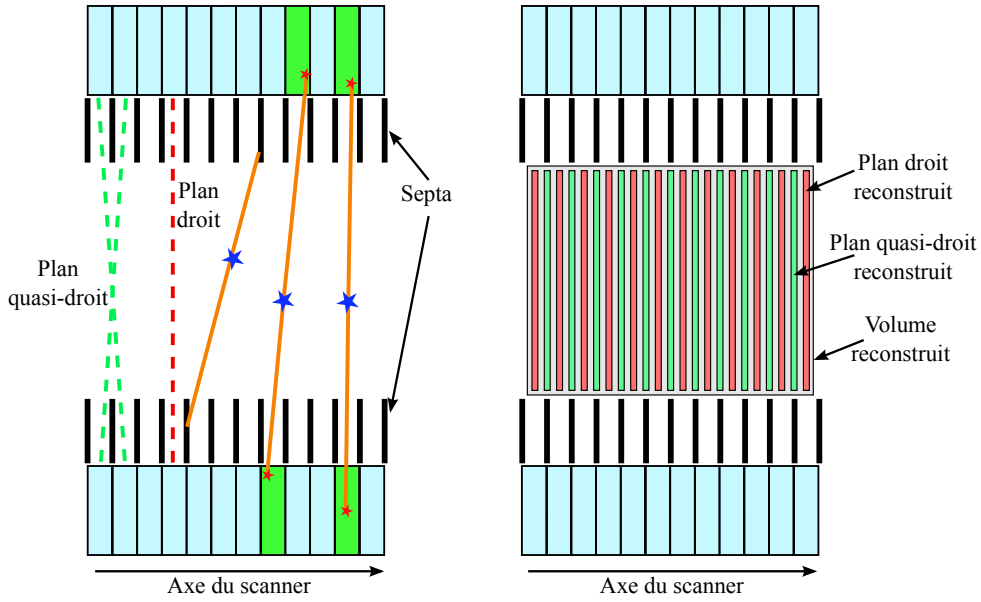
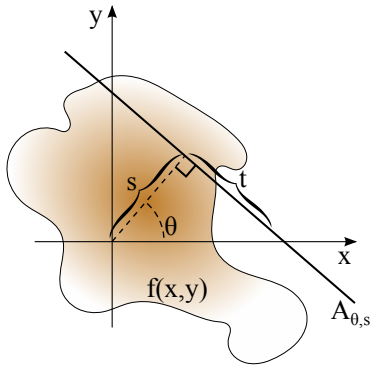


FIGURE 1.17 – Illustration, à gauche, d'un scanner 2D avec des septa bloquant les photons trop obliques. À droite, les images 2D reconstruites avec les sinogrammes des plans droits et quasi droits sont empilées pour former la reconstruction complète.

Dans ce contexte, la reconstruction d'un volume tridimensionnel se ramène à un ensemble de reconstructions bidimensionnelles indépendantes. Chaque sinogramme droit ou quasi droit est reconstruit indépendamment, fournissant chacun une coupe du volume final. Le volume est ensuite obtenu par empilement des coupes reconstruites, comme schématisé dans la figure 1.17.

Le problème de la reconstruction d'une fonction d'un plan à partir de ses intégrales le long de toutes les lignes du plan a été résolu mathématiquement en 1917 par [Radon, 1917], bien avant l'apparition des premiers tomographes. Cette reconstruction repose sur la transformée de Radon inverse.

Nous allons dans un premier temps mettre en forme mathématiquement le problème. La valeur de projection p, mesurée pour un *bin* donné, est la somme de toutes les annihilations produites le long de la ligne de l'espace associée à ce *bin*. Si on note f la fonction décrivant la concentration en traceur dans la coupe (x, y), la projection p suivant une ligne L définie par un angle θ et une distance s à l'origine est égale à la somme de l'activité le long de cette ligne ou à l'intégrale de la concentration en traceur, à un facteur multiplicatif près et en considérant le processus d'émission comme continu et non bruité. Si on pose $A_{\theta,s}$, la ligne du plan (x, y) dont l'équation paramétrique est donnée ci-dessous :



(a)

$$A_{\theta,s} : \begin{cases} x(t) = t\sin\theta + s\cos\theta \\ y(t) = -t\cos\theta + s\sin\theta \end{cases} \tag{1.5}$$

(b)

FIGURE 1.18 – Paramétrisation de la transformée de Radon bidimensionnelle.

où s est la distance de la ligne à l'origine et θ son angle avec l'axe y. La valeur d'une projection est donc donnée par l'intégrale suivante :

$$p(s,\theta) = \int_{A_{\theta,s}} f(x(t),y(t))\,\mathrm{d}t \tag{1.6}$$

$$= \int_{-\infty}^{+\infty} f((t\sin\theta + s\cos\theta, -t\cos\theta + s\sin\theta))\,\mathrm{d}t \tag{1.7}$$

La transformée de Fourier 1D selon s d'une projection p_θ pour un angle θ fixé prend la forme suivante :

$$P_\theta(\nu_s) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(t\sin\theta + s\cos\theta, -t\cos\theta + s\sin\theta)\,e^{-2\pi i\nu_s s}\,\mathrm{d}s\mathrm{d}t \tag{1.8}$$

$$= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)\,e^{-2\pi i\nu_s(x\cos\theta + y\sin\theta)}\,\mathrm{d}x\mathrm{d}y \tag{1.9}$$

On peut constater ici qu'on obtient la transformée de Fourier bidimensionnelle de f. On a alors :

$$P_\theta(\nu_s) = F(\nu_x,\nu_y), avec \begin{cases} \nu_x = \nu_s\cos\theta \\ \nu_y = \nu_s\sin\theta \end{cases} \tag{1.10}$$

La fonction P_θ est alors égale à la transformée de Fourier 2D de f le long d'une ligne passant par l'origine et d'angle θ avec l'axe ν_x. On appelle théorème de la coupe centrale cette équivalence entre la transformée de Fourier 1D d'une projection d'une fonction et une coupe de la transformée de Fourier 2D de cette même fonction. Ce théorème est illustré par la figure 1.19. On peut constater, avec ce théorème, que la transformée de Fourier d'une projection, pour un angle donné, fournit un échantillon de la transformée de Fourier du signal à reconstruire.

À partir du théorème de la coupe centrale, on peut imaginer une méthode de reconstruction directe. Dans un premier temps, on échantillonnerait F, la transformée de Fourier 2D de f, avec les transformées de Fourier 1D des projections p_θ. Dans un second temps, F serait inversée en appliquant la transformée de Fourier inverse 2D, pour obtenir l'estimation de f [Lewitt, 1983]. Les projections étant en nombre limité, elles fournissent un échantillonnage partiel et irrégulier de F dans l'es-

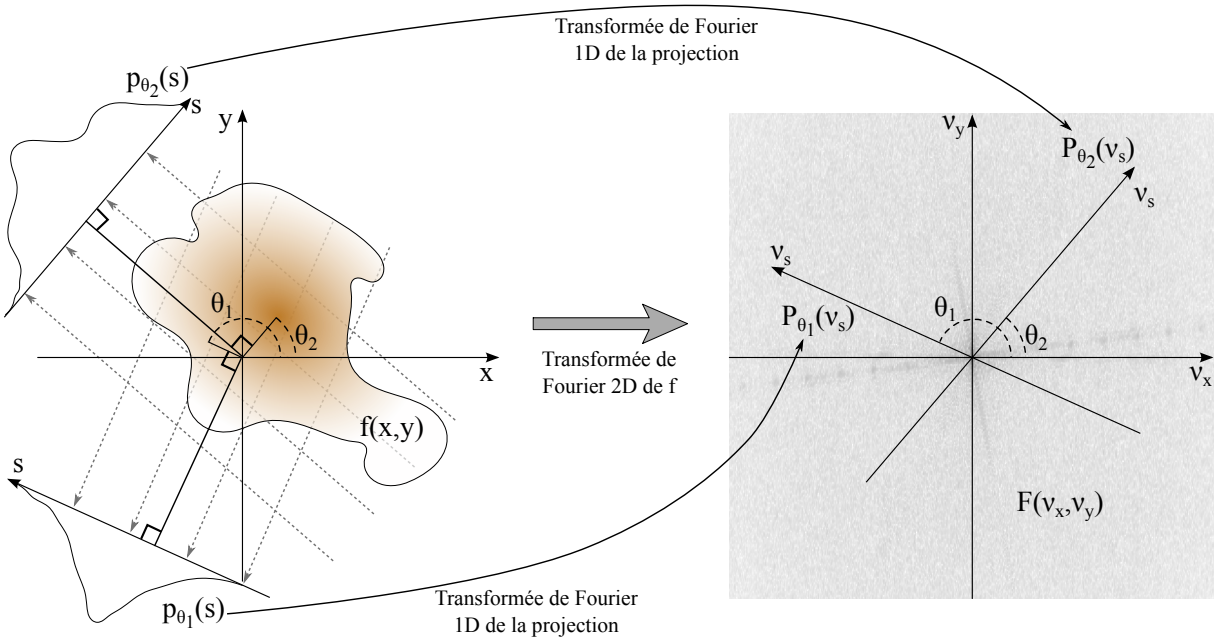


FIGURE 1.19 – Illustration du théorème de la coupe centrale. La transformée de Fourier 1D d'une projection de f, suivant l'ensemble des droites du plan ayant un angle θ par rapport l'axe y, est égale à une coupe de F, la transformée de Fourier 2D de f, passant par l'origine avec un angle θ par rapport à l'axe ν_x.

pace des fréquences (ν_x, ν_y). Il est possible d'interpoler F dans une grille d'échantillonnage régulière, afin de procéder à la transformée de Fourier inverse, mais cette approche ne permet pas d'obtenir une bonne qualité d'image [O'sullivan, 1985].

Une autre approche plus populaire consiste à rétroprojeter dans le domaine image les projections préfiltrées [Kak et Slaney, 1988] dans le domaine de Fourier, c'est la rétroprojection filtrée ou *filtered back projection* en anglais (FBP). En repartant de la transformée de Fourier 2D inverse de F, on a :

$$f(x,y) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \mathrm{F}(\nu_x,\nu_y)\, \mathrm{e}^{2\pi \mathrm{i}(\nu_x x+\nu_y y)}\, \mathrm{d}\nu_x \mathrm{d}\nu_y \tag{1.11}$$

En effectuant le changement de variable, pour passer en coordonnées polaires, $\nu_x = \nu_s \cos\theta, \nu_y = \nu_s \sin\theta$, qui donne $\mathrm{d}\nu_x \mathrm{d}\nu_y = |det\left(\begin{smallmatrix}\cos\theta & -\nu_s\sin\theta \\ \sin\theta & \nu_s\cos\theta\end{smallmatrix}\right)| \mathrm{d}\nu_s \mathrm{d}\theta = \nu_s \mathrm{d}\nu_s \mathrm{d}\theta$, on obtient :

$$f(x,y) = \int_0^{2\pi}\int_0^{+\infty} \mathrm{F}(\nu_s\cos\theta, \nu_s\sin\theta)\, \mathrm{e}^{2\pi \mathrm{i}(\nu_s\cos\theta x+\nu_s\sin\theta y)}\, \nu_s \mathrm{d}\nu_s \mathrm{d}\theta \tag{1.12}$$

En utilisant le théorème de la coupe centrale, on obtient alors :

$$\begin{aligned} f(x,y) &= \int_0^{2\pi}\int_0^{+\infty} \mathrm{P}(\nu_s,\theta)\, \mathrm{e}^{2\pi \mathrm{i}(\nu_s\cos\theta x+\nu_s\sin\theta y)}\, \nu_s \mathrm{d}\nu_s \mathrm{d}\theta & (1.13)\\ &= \int_0^{\pi}\int_{-\infty}^{+\infty} |\nu_s| \mathrm{P}(\nu_s,\theta)\, \mathrm{e}^{2\pi \mathrm{i}(\nu_s\cos\theta x+\nu_s\sin\theta y)}\, \mathrm{d}\nu_s \mathrm{d}\theta & (1.14)\\ &= \int_0^{\pi} p'(s,\theta) \mathrm{d}\theta & (1.15) \end{aligned}$$

où $p'(s,\theta)$ est la transformée de Fourier inverse de $|\nu_s|\mathrm{P}(\nu_s,\theta)$, ou encore, le produit de convolution

de $p(s,\theta)$ par la transformée de Fourier inverse du filtre rampe $\mathscr{F}^{-1}(|\nu_s|)$.

1.4.1.2 Reconstruction 3D

Les premiers scanners étaient cantonnés à un mode de fonctionnement 2D principalement parce que les puissances de calcul disponibles à l'époque étaient insuffisantes pour traiter des 3D. Le placement de septa permet de réduire mécaniquement le nombre de coïncidences qui ne sont pas dans des plans droits ou quasi droits, mais ils réduisent par la même occasion la sensibilité globale du scanner ce qui implique des temps d'acquisition plus longs ou des doses de traceur radioactif plus importantes pour obtenir une statistique de données équivalente. En effet, la sensibilité générale d'un scanner fonctionnant en mode 2D est de l'ordre de 0.5% (une coïncidence détectée pour 200 émissions de positons) quand celle d'une scanner 3D peut être supérieure à 3% [Bailey, 1992]. Le fonctionnement en mode 3D entraîne aussi une augmentation de la fraction de coïncidences diffusées, de l'ordre de 15% en 2D et 40% en 3D pour une source linéaire dans un cylindre d'eau de 20cm de diamètre [Bailey, 1992]. La sensibilité du scanner étant plus importante avec le mode 3D, plus de de coïncidences fortuites sont détectées. Cependant, le taux global du nombre de coïncidences fortuites sur le nombre de coïncidences vraies reste en général plus favorable en mode 3D qu'en mode 2D [Bailey *et al.*, 1991].

Par le passé, les reconstructions directes des sinogrammes 3D étaient un enjeu en matière de temps d'exécution. Pour répondre à cette problématique, de nombreuses méthodes ont été proposées afin de réduire la dimensionnalité du sinogramme afin de revenir à un cas 2D à partir d'un sinogramme 3D. On qualifie ce mode de reconstruction 2D$\frac{1}{2}$. L'étape permettant de passer d'un sinogramme 3D à un sinogramme 2D s'appelle *rebinning*. et quelques-unes de ces méthodes les plus populaires sont le *single-slice rebinning* (*SSRB*) [Daube-Witherspoon et Muehllehner, 1987], *multi-slice rebinning* (*MSRB*) [Lewitt *et al.*, 1994], *fourier simple averaging* (*FOSA*) [Stark *et al.*, 1981] et le *Fourier rebinning* (*FORE*) [Defrise, 1995, Defrise *et al.*, 1997, Tanaka et Amo, 1998, Matej *et al.*, 1998, Liow *et al.*, 2000].

Les reconstructions utilisant directement le sinogramme 3D sont qualifiées de *fully 3D*. La méthode de la FBP peut être étendue au cas 3D en repartant de l'équation 1.11, en ajoutant la dimension axiale à f et en effectuant un passage en coordonnées sphériques. Cependant, seul un scanner couvrant complètement l'angle solide 4π stéradian permet d'exploiter cette adaptation directe de la FBP au cas 3D. Plusieurs approches ont été proposées pour résoudre ce problème, comme des FBP avec des filtres spécifiques [Defrise *et al.*, 1989] ou des méthodes d'estimation des données manquantes [Kinahan et Rogers, 1989, Ben Bouallègue *et al.*, 2007].

1.4.1.3 Limites des approches analytiques

Les méthodes de reconstruction analytiques s'appuient généralement sur le filtrage des projections 1D ou 2D pour les reconstructions 2D ou 3D respectivement. Cependant, ces filtres ont comme caractéristique d'amplifier les hautes fréquences et par conséquent le bruit. Pour limiter ce bruit dans la reconstruction, une fenêtre d'apodisation est en général appliquée au filtre, comme une fenêtre de Hann [Colsher, 1980]. Plus la fréquence de coupure de cette fenêtre est basse, plus le bruit est réduit,

mais au prix d'une réduction de la résolution, comme la montre la figure 1.20.

Les méthodes analytiques supposent que les projections sont des mesures d'intégrales le long des droites de l'espace de la fonction à reconstruire. Cette supposition exclut toute modélisation des différents effets intervenant pendant le processus d'acquisition. Par conséquent, il est nécessaire, soit de précorrigés les projections, soit de corriger les images postreconstruction, de ces différents effets. Les méthodes de reconstruction itératives, présentées par la suite, ne souffrent pas de cette limite, ce qui permet d'obtenir une meilleure qualité d'image, comme on peut le voir sur la figure 1.20.

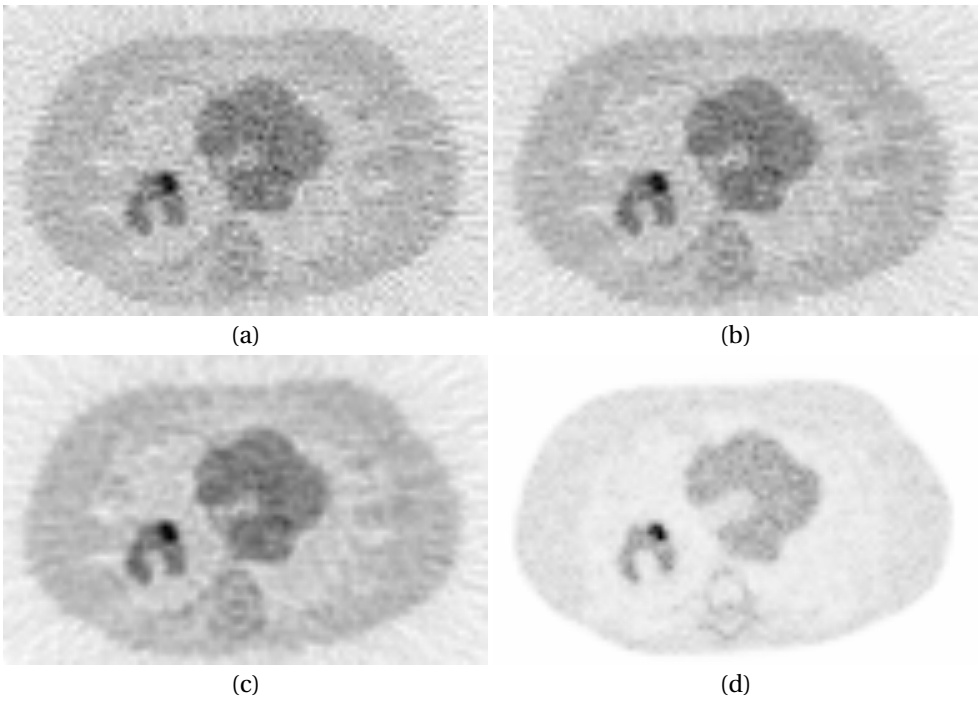


FIGURE 1.20 – Comparaison de reconstructions en FBP avec (a) sans réduction des hautes fréquences, (b) une réduction des hautes fréquences avec un filtre de Hann appliqué à partir de 70 % de la fréquence la plus haute, (c) une réduction à partir de 40 % et (d) une reconstruction itérative.

Les méthodes de reconstruction analytiques ont l'avantage de fournir une reconstruction rapide et en une seule étape. Cependant, en TEP les projections étant peu nombreuses et très bruitées, les images reconstruites analytiquement souffrent de nombreux artefacts bien moins présents dans une reconstruction itérative. Ensuite, ces approches ne permettent pas une reconstruction directe des données en *list-mode* ou histogramme, or la conversion en sinogramme induit des pertes de résolution.

1.4.2 Reconstruction itérative

Contrairement aux méthodes de reconstruction analytiques, les méthodes itératives posent le problème de la reconstruction sous une forme discrétisée spatialement, comme le montre la figure 1.21. Ici, on note A la matrice de réponse du système ou *system response matrix* en anglais (*SRM*) dont la valeur d'un élément a_{ij} correspond à la probabilité qu'une émission de positon dans le *voxel* j conduise à une détection selon le couple de cristaux i, communément appelée *LOR*. La valeur en

chaque *voxel* du volume f est égale à la quantité de traceur dans ce *voxel*. Ici, les reconstructions 2D et 3D se formulent de la même manière, et il n'y a donc pas de différence d'un point de vue algorithmique.

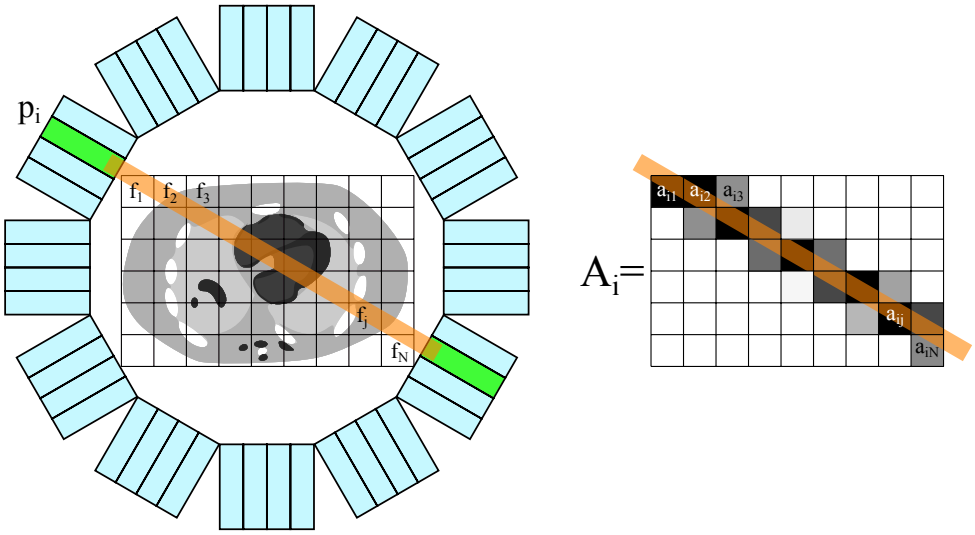


FIGURE 1.21 – Mise en forme algébrique du problème de la reconstruction. La fonction décrivant la répartition de traceur est discrétisée.

1.4.2.1 Reconstruction algébrique

En reconstruction algébrique, le système est supposé linéaire, c'est-à-dire que les projections p et la répartition de traceur f sont liées par l'équation suivante :

$$p = \mathrm{A}f + b \tag{1.16}$$

Un vecteur de bruit b est ajouté pour prendre en compte le bruit présent sur les projections. La méthode des projections de Kaczmarz [Kaczmarz, 1937], permettant de résoudre itérativement un système de N équations à N inconnues, a été à l'origine d'une méthode de reconstruction appelée *algebraic reconstruction technique* (*ART*). En négligeant le bruit, l'équation 1.21 peut être mise sous la forme du système suivant :

$$\begin{cases} p_1 = & a_{11}f_1 + a_{12}f_2 + \cdots + a_{1\mathrm{N}}f_\mathrm{N} \\ p_2 = & a_{21}f_1 + a_{22}f_2 + \cdots + a_{2\mathrm{N}}f_\mathrm{N} \\ \vdots & \\ p_\mathrm{M} = & a_{\mathrm{M}1}f_1 + a_{\mathrm{M}2}f_1 + \cdots + a_{\mathrm{MN}}f_\mathrm{N} \end{cases} \tag{1.17}$$

On constate que chaque équation du système définit un hyperplan dans l'espace des reconstructions. Cet algorithme consiste à projeter orthogonalement successivement sur ces hyperplans le vecteur image $f^{(k)}$, comme représenté pour un cas très simplifié dans la figure 1.22.

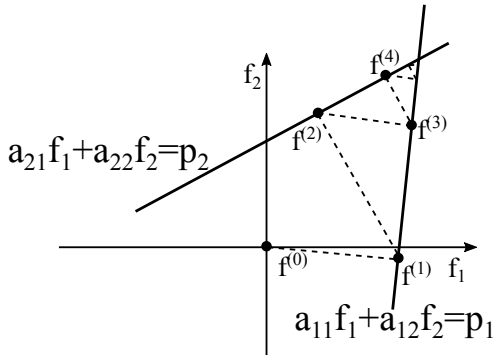


FIGURE 1.22 – La méthode des projections de Kaczmarz qui consiste à projeter orthogonalement successivement la reconstruction sur les hyperplans définis par les projections mesurées.

Une itération de cet algorithme prend la forme suivante :

$$f^{(k)} = f^{(k-1)} - \frac{\mathrm{A}_i f^{(k-1)} - p_i}{\|\mathrm{A}_i\|^2} \mathrm{A}_i \tag{1.18}$$

où $\mathrm{A}_i = (a_{i1}\, a_{i2} \cdots a_{i\mathrm{N}})$ est la partie de la *SRM* A associée à la *LOR* i.

Si le système est surdéterminé, c'est-à-dire que M > N, et que les projections sont bruitées, il n'y aura généralement pas de solution unique et l'estimation de la solution va osciller au voisinage de l'intersection des hyperplans, comme on peut le voir dans la figure 1.23. Cet algorithme a été introduit pour la reconstruction tomographique dans [Gordon *et al.*, 1970], et a montré son efficacité dans [Herman et Meyer, 1993].

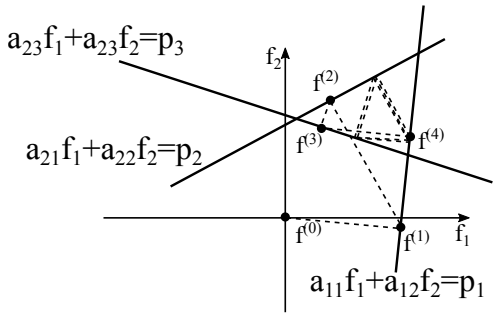


FIGURE 1.23 – Problème avec la méthode des projections de Kaczmarz lorsque le système est surdéterminé et que les projections sont bruitées.

Il existe plusieurs variantes à cet algorithme, dont les principales sont : S-ART (simultaneous ART), M-ART (multiplicative ART), SIRT (simultaneous iterative reconstruction technique). En pratique, ces algorithmes sont peu efficaces parce qu'ils négligent le bruit, très présent en TEP.

1.4.2.2 Reconstruction basée sur un modèle statistique de Poisson

En TEP, l'émission d'un positon, c'est-à-dire une désintégration radioactive, se produit à une fréquence moyenne donnée et indépendante du temps écoulé depuis l'événement précédent. Ainsi, le nombre d'émissions de positons dans un *voxel* est une variable aléatoire suivant une loi de Poisson. La probabilité $\mathrm{P}(f_j)$ que f_j émissions de positons se produisent dans le *voxel* j pendant un certain

laps de temps (par exemple, le temps de l'examen) est de la forme :

$$\mathrm{P}(f_j) = \frac{\overline{f_j}^{f_j}}{f_j!}\,\mathrm{e}^{\overline{f_j}} \tag{1.19}$$

où $\overline{f_j}$ est le nombre moyen d'émissions de positons dans le *voxel* j pendant le laps de temps considéré. Le système étant considéré comme linéaire, la relation entre les valeurs moyennes des projections et de la répartition de traceur est la suivante :

$$\overline{p} = \mathrm{A}\overline{f} \tag{1.20}$$

ou :

$$\overline{p_i} = \sum_{j=1}^{\mathrm{N}} a_{ij}\overline{f_j} \tag{1.21}$$

On note $n(j,i)$ la variable aléatoire décrivant le nombre de positons émis dans le *voxel* j qui sont détectés selon la *LOR* i. Cette variable aléatoire correspond à f_j amoindri de $\mathrm{P}(j,i)$, la probabilité de détecter une coïncidence selon la *LOR* i sachant qu'un positon a été émis dans le *voxel* j, ce qui correspond aux coefficients a_{ij} de la matrice système A. Par conséquent, $n(j,i)$ suit aussi une loi de Poisson de moyenne :

$$\overline{n(j,i)} = \mathrm{P}(j,i)\overline{f_j} = a_{ij}\overline{f_j} \tag{1.22}$$

Le nombre total d'émissions détectées selon la *LOR* i peut être reformulé comme ceci :

$$p_i = \sum_{j=1}^{\mathrm{M}} n(j,i) \tag{1.23}$$

Ainsi, on voit que p_i est une somme de variables aléatoires de Poisson indépendantes. Par conséquent, c'est aussi une variable aléatoire suivant une loi de Poisson. La vraisemblance de f prend la forme suivante :

$$\mathrm{L}(f) = \prod_{i=1}^{\mathrm{M}} \mathrm{P}(p_i|f) = \prod_{i=1}^{\mathrm{M}} \frac{\overline{p_i}^{p_i}}{p_i!}\,\mathrm{e}^{\overline{p_i}} \tag{1.24}$$

À partir de cette expression de la vraisemblance, plusieurs algorithmes ont été proposés [Qi et Leahy, 2006] afin de trouver l'estimation $\widehat{f}$ maximisant le logarithme de la vraisemblance :

$$\widehat{f}_{\mathrm{MV}} = \underset{f}{arg\,max}\{log(\mathrm{L}(f))\} \tag{1.25}$$

Dans ce contexte, l'algorithme *maximum likelihood expectation maximization* (*ML-EM*) introduit par [Dempster *et al.*, 1977, Shepp et Vardi, 1982, Vardi *et al.*, 1985] est très couramment utilisé en raison de sa simplicité d'implémentation et de son efficacité. Il prend la forme suivante :

$$f_j^{(k+1)} = \frac{f_j^{(k)}}{\sum_{i=1}^{\mathrm{M}} a_{ij}} \sum_{i=1}^{\mathrm{M}} a_{ij} \frac{p_i}{\sum_{j'=1}^{\mathrm{N}} a_{ij'} f_{j'}^{(k)}} \tag{1.26}$$

où k désigne l'indice de l'itération. Cet algorithme et ses variantes sont les plus répandues en TEP. Il a

été démontré dans [Lange *et al.*, 1984] que l'algorithme *ML-EM* converge vers l'unique maximum de la fonction de vraisemblance.

Une variante de cet algorithme reposant sur une découpe des projections en sous-ensembles a été proposée dans [Hudson et Larkin, 1994], afin d'accélérer la convergence de l'algorithme *ML-EM* et donc de réduire le nombre d'itérations nécessaires. Cette variante, appelée *ordered subset expectation maximization* (*OSEM*), repose sur l'équation suivante :

$$f_j^{(k,l+1)} = \frac{f_j^{(k,l)}}{\sum_{i=1}^{M} a_{ij}} \sum_{i \in S_l} a_{ij} \frac{p_i}{\sum_{j'=1}^{N} a_{ij'} f_{j'}^{(k,l)}} \tag{1.27}$$

où l est l'indice du sous-ensemble dans les L sous-ensembles, S_l est le sous-ensemble de projections. Une itération est complétée lorsque les L sous-ensembles ont été traités. Lorsqu'un seul sous-ensemble est utilisé, cet algorithme est équivalent à *ML-EM*. Il a été démontré que cet algorithme ne converge pas vers le maximum de vraisemblance [Browne et De Pierro, 1996]. Cependant, en pratique, on observe que l'utilisation de L sous-ensembles accélère la convergence L fois, comme on peut le voir dans la figure 1.24.

Les algorithmes précédents considèrent des projections p_i ayant chacune comme valeur la somme des événements détectés selon un couple de cristaux ou un *bin*, suivant que les données soient en mode histogramme ou sinogramme. Une variante de l'algorithme *ML-EM* a été proposée afin d'effectuer des reconstructions avec les données au format *list-mode* sans conversion préalable [Parra et Barrett, 1998]. Cet algorithme, *list-mode expectation maximization* (*LM-EM*), prend la forme suivante :

$$f_j^{(k+1)} = \frac{f_j^{(k)}}{\sum_{i=1}^{M} a_{ij}} \sum_{i \in LM} a_{ij} \frac{1}{\sum_{j'=1}^{N} a_{ij'} f_{j'}^{(k)}} \tag{1.28}$$

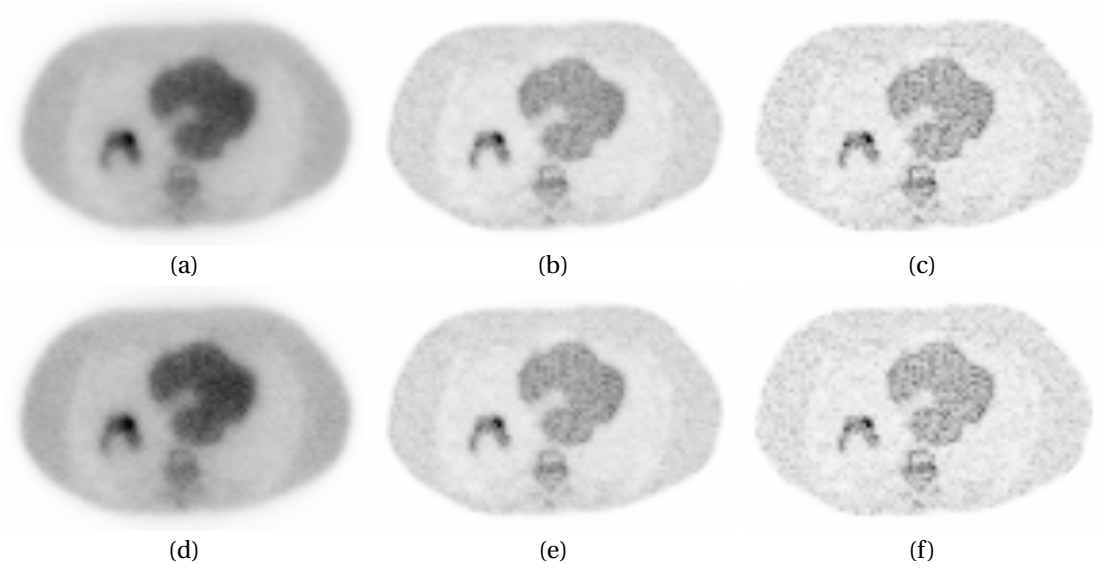
où LM est la liste des indices des *LOR* détectées. Il a été démontré, comme pour l'algorithme *ML-EM*, que l'algorithme *LM-EM* converge vers l'unique maximum de la fonction de vraisemblance [Parra et Barrett, 1998].

Comme pour l'algorithme *ML-EM*, une version de *LM-EM* avec des sous-ensembles a été proposée dans [Reader *et al.*, 1998] pour accélérer la convergence, c'est l'algorithme *list-mode ordered subset expectation maximization* (*LM-OSEM*).

$$f_j^{(k,l+1)} = \frac{f_j^{(k,l)}}{\sum_{i=1}^{M} a_{ij}} \sum_{i \in LM_l} a_{ij} \frac{1}{\sum_{j'=1}^{N} a_{ij'} f_{j'}^{(k,l)}} \tag{1.29}$$

où LM_l est la liste des indices des *LOR* détectées dans le sous-ensemble l. Une comparaison entre cette reconstruction et l'algorithme *LM-EM* sans sous-ensemble est faites dans la figure 1.24. La reconstruction *LM-OSEM* peut se décomposer en quatre étapes. La première étape, la projection est obtenue en multipliant la reconstruction à l'itération courante par la *SRM*. Elle peut être vue comme une simulation du processus d'acquisition TEP où la distribution de traceur imagée par le scanner est son estimation à l'itération courante. L'étape suivante, la correction, inverse la projection estimée afin de former la projection d'erreur. Ensuite, la rétroprojection permet d'obtenir l'image d'erreur en

multipliant la projection d'erreur par la transposée de la *SRM*. Enfin, la mise à jour permet d'obtenir la reconstruction à l'itération suivante en multipliant la reconstruction actuelle par l'image d'erreur divisée par l'image de normalisation.



(a) (b) (c)

(d) (e) (f)

FIGURE 1.24 – La première ligne de cette figure présente des images reconstruites avec l'algorithme *LM-EM* et 10 itérations (a), 40 itérations (b) et 100 itérations (c). La seconde ligne de la figure montre les images reconstruites avec l'algorithme *LM-OSEM* avec 10 sous-ensembles et 1 itération (d), 4 itérations (e) et 10 itérations (f).

En pratique, certaines déviations de la statistique des projections par rapport au modèle de Poisson sont induites par des non-linéarités du système (temps-mort), ce qui a pour effet d'introduire des erreurs à la reconstruction [Daniel et Fessler, 2000].

1.4.2.3 Bruit dans les images reconstruites

Avec les algorithmes de la famille de *ML-EM*, la convergence vers le maximum de vraisemblance est accompagnée d'une augmentation du niveau de bruit dans l'image reconstruite [Wilson *et al.*, 1994], comme on peut le constater sur la figure 1.24. Une méthode permettant de limiter le bruit dans l'image finale consiste à limiter le nombre d'itérations. Cependant, cette méthode ne permet pas un contrôle précis du bruit final, car l'évolution du niveau de bruit en fonction du nombre d'itérations est complexe et dépend de la distribution reconstruite [Barrett *et al.*, 1994]. Une autre approche, très couramment utilisée sur les systèmes cliniques, consiste à reconstruire avec un nombre d'itérations fixe et d'appliquer un filtre de débruitage à l'estimation finale. Il est très courant d'utiliser un filtre reposant sur une convolution par un noyau Gaussien dont la taille correspond à la résolution du scanner. Il existe d'autres approches basées sur un filtrage dans le domaine des ondelettes [Boussion *et al.*, 2009], les moyennes non locales [Chan *et al.*, 2010] et bien d'autres. Une autre manière de contrôler le niveau de bruit est d'incorporer un terme de régularisation à la vraisemblance, exprimée par l'équation 1.24, et de maximiser la distribution *a posteriori* (MAP), donnée par le théorème Bayes :

$$\mathrm{P}(f|p) = \frac{\mathrm{P}(p|f)\mathrm{P}(f)}{\mathrm{P}(p)} \tag{1.30}$$

Avec la reconstruction MAP [Hebert et Leahy, 1989], on cherche une estimation $\widehat{f}_{\text{MAP}}$ maximisant la vraisemblance de p :

$$\begin{aligned}\widehat{f}_{\text{MAP}} &= \underset{f}{arg\,max}\{log(\text{L}(p))\} \\ &= \underset{f}{arg\,max}\{log(\text{L}(f)) + log(\text{P}(f))\}\end{aligned} \tag{1.31}$$

où $\text{P}(p)$ peut être supprimé parce que les projections sont connues. $\text{P}(f)$ est une densité choisie de telle sorte qu'elle pénalise les solutions ne satisfaisant pas des hypothèses *a priori* fixées. Un grand nombre de critères de pénalités ont été proposés, certains contraignant la reconstruction de l'activité à être continue par morceaux [Mumcuoglu *et al.*, 1996b], d'autres utilisant l'information anatomique (tomodensitométrie (TDM) ou imagerie par résonance magnétique (IRM)) pour générer une contrainte [Somayajula *et al.*, 2005]. Ces approches fonctionnent en général très bien pour les études sur fantôme où la distribution d'activité est effectivement continue par morceaux et/ou corrélée avec les structures anatomiques. Cependant, dans le cas des données cliniques, ces *a priori* sont souvent non vérifiés et peuvent conduire à des erreurs.

1.5 Modélisation du système et corrections

Nous avons vu dans la section 1.2 que de nombreux effets interviennent durant le processus d'acquisition en TEP. Une reconstruction quantitative, dans le sens où elle permet de mesurer de manière absolue la concentration en traceur en chaque *voxel*, nécessite de prendre en compte l'intégralité de ces effets. Il a été montré de nombreuses fois que l'absence de correction de l'atténuation et des coïncidences fortuites et diffusées implique d'importantes erreurs dans les images reconstruites [Hoffman *et al.*, 1979, Huang *et al.*, 1979, Hoffman *et al.*, 1981, Mazziotta *et al.*, 1981, Hoffman *et al.*, 1982, Casey et Hoffman, 1986, Cherry et Huang, 1995, Zaidi *et al.*, 2004]. Par ailleurs, la qualité des images reconstructrices dépend directement de la précision de la modélisation des différentes composantes du système.

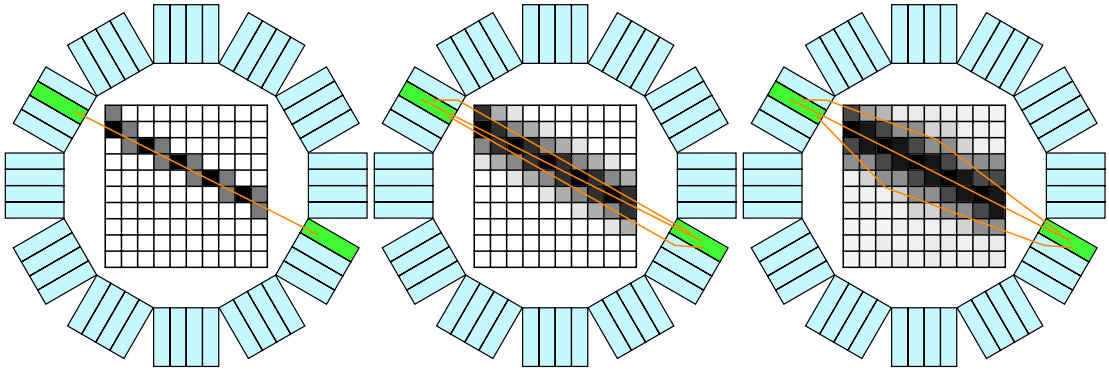
Dans cette section, nous présentons des modélisations des différents effets impliqués dans le système d'acquisition (comprenant le scanner et le patient) présentés dans la section 1.2. Nous verrons aussi comment ces méthodes s'insèrent dans un processus de reconstruction itérative, afin de corriger ces effets et ainsi obtenir une reconstruction quantitative. Nous verrons lesquelles de ces approches apportent une solution satisfaisante et lesquelles laissent encore des possibilités d'amélioration, ce qui nous permettra d'énoncer clairement la problématique de ce travail de thèse.

1.5.1 La matrice système

Toutes les méthodes de reconstruction itérative se basent sur une *SRM* afin de modéliser la réponse du système pour effectuer les opérations de projection et de rétroprojection. L'utilisation d'une matrice pour modéliser le système suppose que celui-ci est linéaire, ce qui est faux pour les effets de temps-mort et les coïncidences fortuites. Il est toutefois possible de les estimer avec des approches qui seront abordées dans la suite de ce document, puis d'ajouter ces composantes aux projections estimées avec la *SRM* tenant compte des autres effets. Une précorrection de ces effets est aussi envi-

sageable mais n'est pas à privilégier parce qu'elle impliquerait une perte du caractère poissonien des données.

L'ensemble des autres effets peut être inclus dans la *SRM*, mais pour des raisons de temps de calcul et d'espace mémoire il existe différents degrés de modélisation. En effet, la *SRM* est une matrice de très grandes tailles, de l'ordre de 10^{14} éléments pour un scanner TEP Philips Allegro/GEMINI. Dans la figure 1.25a on considère les cristaux comme des détecteurs ponctuels parfaits et aucun effet dans l'objet imagé n'est modélisé. Un tel modèle a l'avantage de pouvoir être calculé rapidement à la volée. Ensuite, on peut intégrer un modèle de la réponse du détecteur, ce que a pour conséquence d'étaler la ligne, comme illustré dans la figure 1.25b. Avec ce type de modélisation, il est plus difficile d'estimer les coefficients de la *SRM* à la volée, mais une préestimation et un stockage reste envisageable grâce au caractère creux de cette matrice. Pour finir, on peut intégrer tous les effets liés à l'objet et au détecteur, ce qui a pour effet d'élargir encore plus la ligne, comme l'illustre la figure 1.25c. Ce dernier modèle n'est pas utilisable en pratique, car il nécessite la construction d'une *SRM* spécifique à chaque nouvel objet imagé. Il existe cependant des simplifications où le milieu objet est considéré comme étant homogène, permettant l'utilisation d'une *SRM* unique, mais même dans un tel cas, il est difficile de stocker cette matrice parce qu'elle possède un grand nombre de coefficients non nuls.



(a) Modèle linéique (b) Modèle incluant le détecteur (c) Modèle incluant le détecteur et l'objet

FIGURE 1.25 – Pour la *LOR* considérée, les niveaux de gris indiquent la probabilité de l'émission du positon en chaque *voxel* donnée par les différentes sophistications des modélisations de la matrice système. Le blanc correspond à une probabilité faible et le noir une probabilité élevée.

Pour simplifier la complexité de la *SRM* A, une approche consiste à la décomposer en un produit de matrices modélisant chacune une partie du système. L'ensemble de ces matrices de dimensionnalité plus faible, ou plus creuses, est plus facile à stocker ou à estimer à la volée. Une décomposition proposée dans [Qi *et al.*, 1998] est la suivante :

$$\mathrm{A} = \mathrm{A}_{sens\ det}\mathrm{A}_{flou\ det}\mathrm{A}_{att}\mathrm{A}_{geom}\mathrm{A}_{positon} \tag{1.32}$$

où $\mathrm{A}_{sens\ det} \in \mathbb{R}^{\mathrm{M}\times\mathrm{M}}$ est une matrice diagonale composée des coefficients sensibilité des *LOR*. $\mathrm{A}_{flou\ det} \in \mathbb{R}^{\mathrm{M}\times\mathrm{M}}$ est une matrice modélisant le flou dans le détecteur dû à la diffusion inter cristaux et aux erreurs de lecture des TPM. $\mathrm{A}_{att} \in \mathbb{R}^{\mathrm{M}\times\mathrm{M}}$ est une matrice diagonale contenant les facteurs d'atténuation pour chaque paire de cristaux. $\mathrm{A}_{geom} \in \mathbb{R}^{\mathrm{M}\times\mathrm{N}}$ est la matrice permettant de passer de l'espace image

à l'espace des projections en tenant compte de la géométrie du détecteur. $A_{positon} \in \mathbb{R}^{N\times N}$ est une matrice modélisant le flou dans l'espace image dû au parcours du positon. Cette décomposition permet de séparer les composantes du système qui dépendent du scanner, qui sont fixes, de celles qui dépendent du patient et varient donc pour chaque examen. Cette décomposition n'intègre pas de matrices modélisant la diffusion et la non-colinéarité des photons d'annihilation bien qu'il soit aussi possible de les intégrer. Une autre décomposition, proposée dans [Reader *et al.*, 2002], est donnée par l'équation suivante :

$$A = A_{proj} A_{geom} A_{im} \tag{1.33}$$

où $A_{proj} \in \mathbb{R}^{M\times M}$ combine les effets intervenants dans l'espace des projections, A_{geom} est équivalent à A_{geom} dans l'équation précédente et $A_{im} \in \mathbb{R}^{N\times N}$ combine les effets intervenants dans le domaine image, dont la non-colinéarité des photons d'annihilation et la diffusion. Cependant, il est plus correct d'intégrer la non-colinéarité des photons d'annihilation dans la matrice A_{geom}, comme il en a été discuté dans [Rahmim *et al.*, 2008a]. Dans cette thèse, nous nous baserons sur la décomposition suivante :

$$A = A_{sens\ det} A_{att} A_{det} A_{positon} \tag{1.34}$$

où A_{det} combine la matrice A_{geom} et la matrice $A_{flou\ det}$.

1.5.2 Effets physiques et géométriques liés au détecteur

Nous avons vu dans le paragraphe 1.2.5.3 que pour détecter efficacement les photons d'annihilation les détecteurs ont une conception qui implique de fort effets de parallaxe sur les bords du champ de vue. Certaines conceptions géométriques différentes ont été proposées dans le but de supprimer les effets de parallaxe dus à la pénétration inter cristaux [Braem *et al.*, 2004]. Cependant, à l'heure actuelle aucun scanner clinique n'est équipé de ce type de conception. De plus, ces scanners ne permettent pas d'éviter les erreurs causées par la diffusion inter cristaux.

Ces effets sont modélisés dans la reconstruction par la matrice A_{det}, modélisant la réponse du scanner. Elle peut être estimée en amont de la reconstruction, par des approches empiriques [Selivanov *et al.*, 2000, Frese *et al.*, 2003, De Bernardi *et al.*, 2003, Lee *et al.*, 2004, Zhou et Qi, 2011], ou analytiques [Schmitt *et al.*, 1988, Qi *et al.*, 1998, Strul *et al.*, 2003, Staelens *et al.*, 2004], ou encore basée sur des SMC [Mumcuoglu *et al.*, 1996a, Veklerov *et al.*, 1998, Alessio *et al.*, 2006, Stute, 2010]. Dans toutes ces approches, la matrice est stockée pour être utilisée au moment de la reconstruction, ce qui impose d'utiliser des méthodes de compression tirant parti des symétries du système de détection et de l'aspect creux de la matrice.

Il existe aussi des approches permettant d'estimer à la volée cette composante de la réponse du système [Siddon, 1985, Pratx *et al.*, 2006, Chen et Glick, 2007, Cui *et al.*, 2010, Pratx et Levin, 2011, Cui *et al.*, 2011b, Bert et Visvikis, 2011, Cui *et al.*, 2011a], mais aucune d'entre elles n'intègrent un modèle précis du détecteur, comme par exemple la diffusion intercristaux.

La modélisation de la réponse du détecteur dans la reconstruction se fait à travers la matrice notée A_{geom} (voir section précédente). Cette matrice permet d'effectuer les deux étapes clés de tous algorithmes de reconstruction itératifs, c'est-à-dire la projection et la rétroprojection.

La projection est l'opération permettant de passer de l'espace image à l'espace des projections ou des données, tandis que la rétroprojection permet de passer de l'espace des projections à l'espace image. La projection se calcule en multipliant le vecteur image par A_{det}. Quant à la rétroprojection, elle se calcule en multipliant un vecteur de projections par la transposée de A_{det}. Il existe deux stratégies pour calculer la projection ou la rétroprojection, l'une dite *voxel-driven* et l'autre dite *LOR-driven* (ou *pixel-driven* si la reconstruction est basée sur le format sinogramme).

Les approches *LOR-driven* et *pixel-driven* consistent à effectuer les projection et rétroprojection en partant d'une LOR ou d'un *pixel* du sinogramme et d'utiliser les coefficients de A_{det} associés à cette LOR ou ce *pixel*. L'approche *voxel-driven*, quant à elle, traite la projection et la rétroprojection *voxel* par *voxel*. En TEP *list-mode*, l'approche *LOR-driven* est préférée à l'approche *voxel-driven* parce que les données de projection n'étant pas organisées, avec l'approche *voxel-driven*, il serait nécessaire de parcourir l'ensemble des données de projection pour chaque *voxel* afin de déterminer quelles sont les *LOR* interagissant avec ce *voxel*.

1.5.3 Normalisation

La normalisation vise à corriger les données acquises de la variation de sensibilité, présentée dans le paragraphe 1.2.5.2. En pratique, cette correction est appliquée en amont de la reconstruction en divisant les projections mesurées par les coefficients des sensibilité associées.

Le moyen le plus direct pour estimer tous les coefficients de sensibilité est d'effectuer l'acquisition d'une source radioactive uniforme. La sensibilité associée à un couple de cristaux est alors proportionnelle au nombre de coïncidences détectées par celui-ci. Cette méthode directe a de nombreux désavantages. Les coïncidences diffusées requièrent un coefficient de normalisation différent des vraies [Ollinger, 1995], en raison de leur énergie réduite et de leur plus grande fenêtre d'angles d'incidence. Or, seuls les coefficients de coïncidences vraies sont calculés par la méthode directe. De plus, la concentration d'activité de la source uniforme doit être réduite pour éviter les phénomènes de saturation du détecteur causé par le temps-mort. Ceci impose une grande durée d'acquisition (plusieurs dizaines d'heures) afin d'obtenir une qualité statistique satisfaisante. Il est aussi nécessaire de prendre en compte l'atténuation de la source utilisée. Enfin, les éléments constituant les appareils d'imagerie voient leur comportement varier avec le temps. Il est donc nécessaire de recalculer régulièrement les coefficients de normalisation. De ce fait, le long temps d'acquisition est un frein à l'utilisation de ces techniques en pratique clinique.

Une méthode, proposée par [Defrise *et al.*, 1991], permet d'améliorer les propriétés statistiques de la normalisation tout en réduisant les temps d'acquisition. Elle se base sur la décomposition des effets de normalisation en une série de composantes indépendantes représentant chacune une source particulière de non-uniformité. Bien que cette décomposition donne lieu à de nombreux facteurs à estimer, ceux-ci peuvent tous être calculés en imageant quelques distributions simples. Ainsi, l'acquisition d'une source linéaire tournante permet de calculer les facteurs correctifs relatifs aux effets géométriques, tandis que l'acquisition d'un cylindre uniforme tient compte des variations de sensibilité des cristaux [Badawi *et al.*, 2000].

1.5.4 Atténuation

La correction de l'effet due à l'atténuation des photons d'annihilation dans l'objet imagé, présentée dans la section 1.2.4, nécessite la détermination de l'atténuation subie par chaque *LOR* en fonction des objets présents dans le champ de vue du scanner (patient, table d'examen, blindage ...).

Historiquement, les premières approches développées pour mesurer l'atténuation reposaient sur l'utilisation du détecteur du scanner TEP et d'une source externe, du type ^{68}Ge [Meikle *et al.*, 1995], tournant autour du patient. Ces méthodes possèdent cependant plusieurs inconvénients majeurs. D'une part, la mesure de l'atténuation et la mesure de l'activité interfèrent entre elles parce qu'elles utilisent le même détecteur [Kinahan *et al.*, 1998]. D'autre part, l'estimation de l'atténuation est entachée d'un bruit statistique élevé.

Aujourd'hui, la méthode la plus courante est basée sur l'utilisation d'un scanner TDM accolé au scanner TEP. Le scanner TDM fournit des images anatomiques de résolution élevée avec un faible niveau de bruit, un temps d'acquisition court et sans interaction avec le scanner TEP [Kinahan *et al.*, 1998]. De nombreux systèmes TEP/TDM combinent en une seule machine les deux scanners. L'image anatomique donnée par le scanner TDM donne les valeurs en chaque *voxel* de l'atténuation des rayons X. Celles-ci sont données en unités Hounsfield, qui est une échelle utilisée par tous les scanners TDM cliniques. Dans cette échelle, l'eau possède une valeur de 0 et l'air de -1000. La conversion d'une atténuation $\mu_{x\ \mathrm{TDM}}$ en unités Hounsfield UH_x est donnée par l'expression suivante :

$$\mathrm{UH}_x = 1000\frac{\mu_{x\ \mathrm{TDM}} - \mu_{eau\ \mathrm{TDM}}}{\mu_{eau\ \mathrm{TDM}}} \tag{1.35}$$

où $\mu_{eau\ \mathrm{TDM}}$ correspond à l'atténuation de l'eau en TDM. Les valeurs d'atténuation fournies par la TDM sont obtenues avec des photons dont l'énergie est répartie de manière continue entre 10 et 100 keV, soit bien moins que les 511 keV des photons d'annihilation. Il est alors nécessaire d'effectuer une conversion de ces coefficients d'atténuation. La conversion, proposée dans [Kinahan *et al.*, 1998], convertit les coefficients d'atténuation des images TDM avec une mise à l'échelle deux fois linéaire. Les tissus moins atténuant que l'os sont considérés comme étant similaire à de l'eau, c'est-à-dire où l'effet Compton prédomine largement sur l'effet photoélectrique, et leur coefficients d'atténuation peuvent donc être convertis avec un même facteur d'échelle. Pour les tissus autant ou plus atténuant que l'os, l'effet photoélectrique n'est plus négligeable, on utilise donc un autre facteur d'échelle. La conversion est donnée par l'équation suivante :

$$\mu_{x\ 511keV} = \begin{cases} \frac{\mathrm{UH}_x+1000}{1000}\mu_{eau\ 511keV} & ,si\ \mathrm{UH}_x <= 300 \\ \frac{\mathrm{UH}_x+1000}{1000}\frac{\mu_{eau\ \mathrm{TDM}}}{\mu_{os\mathrm{TDM}}}\mu_{os\ 511keV} & ,si\ \mathrm{UH}_x > 300 \end{cases} \tag{1.36}$$

où $\mu_{eau\ 511keV}$ et $\mu_{os\ 511keV}$ sont, respectivement, les coefficients d'atténuation à 511keV de l'eau et de l'os, $\mu_{eau\ \mathrm{TDM}}$ et $\mu_{os\ \mathrm{TDM}}$ sont les mêmes coefficients d'atténuation, mais pour la TDM. Cette méthode de conversion permet d'obtenir une estimation de l'atténuation à 511keV suffisamment précise pour corriger efficacement l'atténuation dans les données TEP [Nakamoto *et al.*, 2002].

Une nouvelle génération de systèmes combinés TEP/IRM se développe ses dernières années [Martinez-Moller *et al.*, 2009, Schreibmann *et al.*, 2010, Keereman *et al.*, 2010, Hofmann *et al.*, 2011]. L'IRM, par

rapport à la TDM, a pour avantages de n'émettre aucun rayonnement ionisant ainsi que de fournir des images supplémentaires pouvant être utiles au diagnostic. Cependant, les images IRM ne fournissent pas une mesure de l'atténuation et la conversion des niveaux de gris des images IRM en coefficient d'atténuation est une problématique complexe [Visvikis *et al.*, 2014]. Par exemple, les os et l'air apparaissent avec les mêmes niveaux de gris.

Une autre approche prometteuse, proposée dans [Defrise *et al.*, 2012], consiste à estimer et corriger l'atténuation pendant la reconstruction en utilisant l'information de *TOF*, disponibles sur les scanners TEP récents. Cette méthode a l'avantage de fournir une estimation de l'atténuation pour l'énergie des photons d'annihilation et de fournir une correction de l'atténuation sans aucune donnée autre que celles fournies par le scanner TEP.

1.5.5 Parcours du positon

Le parcours du positon (voir section 1.2.3.1), peut dégrader les images reconstruites, en particulier lorsque le traceur émet des positons avec une énergie cinétique élevée.

Une solution proposée a été de réduire la distance parcourue par le positon à l'aide de champs magnétique puissant [Hammer *et al.*, 1994, Wirrwar *et al.*, 1997]. Cette méthode permet de réduire efficacement la distance que parcourent les positons dans les directions orthogonales au champ magnétique, mais pas dans sa direction, le long de laquelle la force de Lorentz est nulle. Cette solution est complexe et coûteuse à mettre en œuvre matériellement parce qu'elle nécessite un champ magnétique très intense, mais est présente intrinsèquement dans les systèmes combinés TEP/IRM.

D'autres approches s'appuient sur une déconvolution des projections avant la reconstruction [Derenzo, 1986, Haber *et al.*, 1990]. Le filtre de déconvolution ne considère cependant qu'un parcours du positon moyen, et de plus, il amplifie le bruit déjà présent dans les projections. Cette méthode n'est compatible qu'avec le mode sinogramme.

La grande majorité des méthodes de correction du parcours du positon s'appuient sur une modélisation de ce parcours dans le processus de reconstruction par la convolution de l'image reconstruite avec un noyau modélisant le flou induit par cet effet, juste avant d'effectuer la projection. Le noyau de convolution est généralement estimé pour une distribution de matière homogène (un seul matériau) [Bai *et al.*, 2003, Bai *et al.*, 2005, Palmer *et al.*, 2005, Ruangma *et al.*, 2006, Rahmim *et al.*, 2008b, Cal-González *et al.*, 2009, Jødal *et al.*, 2012], posant des problèmes dans les milieux avec une densité très différente du milieu choisi. D'autres méthodes utilisent des noyaux de convolution pour différents milieux [Rahmim *et al.*, 2008a, Alessio et MacDonald, 2008, Lehnert *et al.*, 2011]. Cette méthode pose des problèmes au niveau des interfaces entre matériaux. Une autre approche repose sur la construction d'une *SRM* complète, par SMC [Moreau *et al.*, 2014], modélisant le système ainsi que les effets dépendants de l'objet imagé. La *SRM* ainsi construite intègre la composante de parcours du positon. Il est cependant très coûteux en temps de calcul, de construire une telle *SRM*. Il n'est pas envisageable de répéter cette étape pour chaque nouvel objet imagé.

Pour des traceurs, comme le ^{18}F-FDG, où l'énergie cinétique des positons est faible, le parcours du positon est quasiment toujours négligé. Sinon, dans les systèmes cliniques, c'est un noyau de convolution Gaussien modélisant la fonction d'étalement du point ou *point spread function* en an-

glais (*PSF*) du scanner qui intègre aussi un parcours de positon moyen.

1.5.6 Coïncidences fortuites

Les coïncidences fortuites, formées par deux photons d'annihilations détectés dans la même fenêtre de temps, mais provenant d'annihilations distinctes, ont été introduites dans la sous-section 1.3.3. Les coïncidences fortuites peuvent être précorrigées par la soustraction aux projections acquises de la quantité de coïncidences fortuites estimée. Toutefois, afin de préserver la statistique de Poisson des données, les quantités de coïncidences fortuites estimées sont incluses dans les projections estimées par le projecteur.

Si on observe le sinogramme des coïncidences fortuites par rapport au sinogramme des coïncidences vraies, voir figure 1.26, on remarque que la distribution des coïncidences fortuites est relativement homogène et sans corrélation évidente avec le sinogramme de coïncidences vraies. Il a été montré dans [Hoffman *et al.*, 1981] que le taux de moyen de coïncidences fortuites r_{ij}, à un instant t dans la *LOR* associée aux détecteurs i et j, est :

$$r_{ij}(t) = \tau s_i(t) s_j(t) \tag{1.37}$$

où τ est la largeur de la fenêtre de coïncidences, $s_i(t)$ et $s_j(t)$ les taux moyens de photons détectés par les détecteurs i et j. On peut observer, avec cette relation, que le taux de coïncidences fortuites va augmenter avec le carré de l'activité dans l'objet imagé. L'activité injectée dans le patient doit donc être suffisante pour enregistrer une quantité raisonnable de coïncidences, nécessaire pour obtenir une reconstruction avec une qualité d'image correcte, mais sans être trop importante afin d'éviter un taux de coïncidences fortuites trop important et de minimiser la dose de radiation reçue par le patient.

(a) (b)

FIGURE 1.26 – Sinogrammes des (a) coïncidences vraies et (b) coïncidences fortuites. La répartition de coïncidences fortuites est quasiment uniforme et ne semble pas être corrélée avec le sinogramme des coïncidences vraies.

Le sinogramme des coïncidences fortuites peut être estimé à partir du nombre de photons uniques détectés dans les détecteurs du scanner pendant l'acquisition en intégrant l'équation 1.37 sur la durée de l'acquisition T [Williams *et al.*, 2003]. En négligeant la décroissance radioactive du traceur pendant l'acquisition, le taux de coïncidences fortuites R total est donné par l'équation :

$$\mathrm{R}_{ij} = \frac{\tau}{\mathrm{T}} \mathrm{S}_i \mathrm{S}_j \tag{1.38}$$

où S_i et S_j sont les nombres totaux de photons détectés dans les détecteurs i et j respectivement. Le nombre de photons uniques détectés pendant une acquisition étant 10 à 100 fois plus élevés que le nombre de coïncidences, cette méthode permet d'estimer le taux de coïncidences fortuites de manière moins bruitée qu'une mesure directe [Stearns *et al.*, 2003].

Une autre méthode très populaire est la méthode de la fenêtre retardée (MFR). Elle consiste, comme son nom l'indique, à retarder la fenêtre de coïncidence τ d'une durée retard de l'ordre de 60 nanosecondes. Les coïncidences détectées dans cette fenêtre sont forcément des coïncidences fortuites puisqu'une coïncidence primaire ne peut pas être constituée de photons dont les temps d'arrivée sont séparés par un intervalle de temps aussi important [Hoffman *et al.*, 1981]. La méthode de la fenêtre retardée est une mesure directe des coïncidences fortuites naturellement affectée par le temps-mort et le recouvrement des *pulses*, elle est donc plus exacte que la méthode des photons uniques mais aussi plus bruitée. Des techniques de réduction de variance basées sur la redondance des données mesurées par la MFR et l'équation 1.38 ont cependant été développées, permettant de réduire significativement le bruit de cette estimation [Casey et Hoffman, 1986, Badawi *et al.*, 1999].

1.5.7 Temps-mort

L'effet du temps-mort, présenté dans la sous-section 1.3.2, est une conséquence de la vitesse de traitement limitée de l'électronique du système et la période réfractaire des blocs de cristaux. Si on note n et m les taux respectifs de photons incidents et détectés dans un bloc donné et τ_{bloc} la période réfractaire, la relation liant m à n est la suivante [Knoll, 2010] :

$$m = n\,\mathrm{e}^{-n\tau_{bloc}} \tag{1.39}$$

La relation entre n, le taux de coïncidences à l'entrée du système électronique, et m, le taux de coïncidences réellement traitées, est la suivante [Knoll, 2010] :

$$m = \frac{n}{1 + n\tau_{elec}} \tag{1.40}$$

où τ_{elec} le temps pendant lequel le système est inactif après la détection d'une coïncidence.

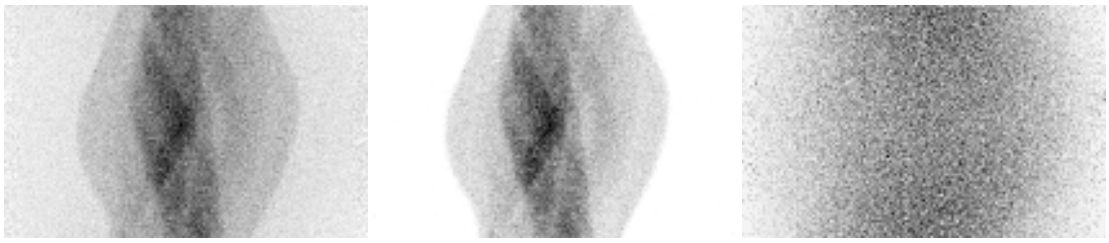
Concernant le temps-mort lié à la période réfractaire des blocs de cristaux, on observe que l'augmentation du taux de photons incidents lorsqu'elle tend vers l'infini, fait tendre vers zéro le taux de photons détectés [Eriksson *et al.*, 1994, Moisan *et al.*, 1997]. Le temps-mort lié à l'électronique, quant à lui, fait tendre le taux de coïncidences détectées vers $1/\tau_{elec}$ quand le taux de coïncidences en entrer du système tend vers l'infini.

Corriger le temps-mort des données TEP nécessite d'estimer la valeur de τ_{bloc} pour le scanner considéré (τ_{elec} est généralement connu exactement), ce qui est généralement fait en ajustant le modèle 1.39 à la courbe du nombre de photons détectés par le système en fonction de l'activité imagée lors de l'acquisition d'un fantôme cylindrique dont l'activité décroit dans le temps [Tai *et al.*, 1997]. À partir des modèles 1.39 et 1.40, des facteurs de correction du temps-mort tm_{ij} sont ensuite calculés pour chaque *LOR* du scanner en multipliant les fractions de temps-mort dans les blocs contenant les détecteurs i et j et en multipliant le tout par la fraction de temps-mort dans le système électro-

nique [Eriksson *et al.*, 1994, Moisan *et al.*, 1997].

1.5.8 Diffusion

Les coïncidences diffusées, présentées dans le paragraphe 1.2.4, sont un problème majeur en TEP où elles représentent typiquement de 40% à 60% des coïncidences détectées, comme on peut le visualiser sur les sinogrammes de la figure 1.27.



(a) (b) (c)

FIGURE 1.27 – Sinogramme des coïncidences promptes (vraies+diffusées) (a) comparé aux sinogrammes des coïncidences vraies (b) et des coïncidences diffusées (c) séparés.

Cet effet ayant un impact très important sur la qualité des images reconstruites, de très nombreuses méthodes de correction ont été proposées. La correction de la diffusion se fait en soustrayant aux projections une composante de diffusion estimée. Il existe un grand nombre de méthodes pour estimer la diffusion, nous allons en aborder quelques-unes.

Si on observe la distribution des coïncidences diffusées dans le sinogramme de la figure 1.27c on se rend compte que celle-ci suit une évolution lente dans la direction transaxiale. Une solution proposée a alors été d'ajuster une fonction analytique simple, parabole [Karp *et al.*, 1990] ou Gaussienne [Cherry et Huang, 1995], à la distribution des coïncidences détectées en dehors du patient (appelé la queue), qui ne peuvent qu'être des coïncidences diffusées (une fois les coïncidences fortuites corrigées) étant donné qu'il n'y a pas d'activité en dehors du patient. Cette méthode a l'avantage d'être rapide, mais n'est pas précise lorsque la distribution de traceur n'est pas très homogène.

En TEP, comme nous l'avons vu précédemment, une fenêtre d'énergie dont la largeur est fixée en fonction de la résolution énergétique du scanner, est utilisée pour détecter les photons d'annihilations primaires. Mais nous avons vu qu'elle contenait environ 50% de photons diffusés. Une méthode pour estimer le sinogramme des coïncidences diffusées concise à définir deux nouvelles fenêtres d'énergie, une avec des seuils hauts FH où il n'y a que des coïncidences vraies, une avec des seuils plus bas FB où il y aura une majorité de coïncidences diffusées [Bendriem *et al.*, 1998]. Les sinogrammes des FH et FB sont ensuite fortement lissés (ce qui n'entraîne pas de pertes significatives grâce aux variations lentes dans le sinogramme des diffusées) pour réduire le bruit. Le sinogramme de coïncidences diffusées est alors obtenu en soustrayant le sinogramme de FH à FB. Une méthode similaire basée sur l'utilisation de trois fenêtres d'énergie a aussi été proposée [Shao *et al.*, 1994]. Ces méthodes ont l'avantage de prendre en compte la distribution réelle du traceur, mais sont fortement biaisées [Harrison *et al.*, 1991, Adam *et al.*, 1998] du fait qu'elles estiment la distribution des coïncidences diffusées dans la fenêtre d'énergie principale en observant cette distribution dans d'autres fenêtres d'énergies où les angles de diffusions et les nombres de diffusions n'ont pas les mêmes dis-

tributions.

La méthode la plus populaire à l'heure actuelle repose sur une estimation analytique du sinogramme des coïncidences lorsqu'il n'y a qu'une seule diffusion, c'est l'algorithme *single scatter simulation* (*SSS*) de Watson [Watson *et al.*, 1996, Watson, 2000, Watson, 2007]. Cette méthode repose sur l'utilisation d'une estimation de l'activité (reconstruction sans correction de la diffusion) et de la formule de Klein-Nishina. En effet, aux énergies considérées (autour de 511 keV), la probabilité qu'un photon d'énergie E_0 soit diffusé d'un angle θ par un seul électron est donnée par la formule de [Klein et Nishina, 1929] :

$$\frac{d\sigma}{d\Omega}(E_0,\theta) = \frac{r_0^2}{2}\frac{1}{(1+\alpha(1-\cos\theta))^2}\left(1+\cos^2\theta+\frac{\alpha^2(1-\cos\theta)^2}{1+\alpha(1-\cos\theta)}\right) \tag{1.41}$$

où $r_0 = 2,81794 \times 10^{-15}$ cm est le rayon classique d'un électron et $\alpha = \frac{E_0}{m_e c^2}$ avec m_e la masse d'un électron et c est la vitesse de la lumière dans le vide et Ω est l'angle solide de diffusion. La probabilité totale qu'un photon d'énergie E_0 soit diffusé par un seul électron est donnée par l'intégrale de l'équation 1.41 sur l'angle solide complet 4π stéradian. On obtient alors la probabilité suivante :

$$\begin{aligned}\sigma(E_0) &= \int_{4\pi}\frac{d\sigma}{d\Omega}(E_0,\theta)d\Omega \\ &= 2\pi r_0^2\left(\frac{1+\alpha}{\alpha^3}\left(\frac{2\alpha(1+\alpha)}{1+2\alpha}-\ln(1+2\alpha)\right)+\frac{\ln(1+2\alpha)}{2\alpha}-\frac{1+3\alpha}{(1+2\alpha)^2}\right)\end{aligned} \tag{1.42}$$

En utilisant ces equations, l'estimation du taux de coïncidences diffusées une seule fois S^{AB}, parmi les coïncidences détectées par le couple de cristaux A et B, est donnée par l'équation suivante :

$$S^{AB} = \int_{V_s} dV_s \left(\frac{\sigma_{AS}\sigma_{BS}}{4\pi R_{AS}^2 R_{BS}^2}\right)\frac{\mu}{\sigma_c}\frac{d\sigma_c}{d\Omega}[I^A + I^B] \tag{1.43}$$

où

$$I^A = \epsilon_{AS}\epsilon'_{BS}e^{-(\int_S^A \mu ds+\int_S^B \mu' ds)}\int_S^A f ds \tag{1.44}$$

$$I^B = \epsilon'_{AS}\epsilon_{BS}e^{-(\int_S^A \mu' ds+\int_S^B \mu ds)}\int_S^B f ds \tag{1.45}$$

La section efficace différentielle $\frac{d\sigma_c}{d\Omega}$ et la section efficace totale σ_c sont calculée avec la formule de Klein-Nishina et son intégration, données par les équations 1.41 et 1.42 respectivement. Dans ces équations, V_s représente le volume diffusant, S la position de diffusion, σ_{AS} et σ_{BS} sont les sections efficaces géométriques des détecteurs A et B pour des photons γ incidents suivant les lignes AS et BS, ϵ_{AS} et ϵ_{BS} sont les efficacités des détecteurs A et B pour des photons γ d'énergie 511 keV incidents suivant les lignes AS et BS, R_{AS} et R_{BS} sont les distances du détecteur A au point de diffusion S et du détecteur B au point de diffusion S, μ est le coefficient d'atténuation linéique pour des photons γ à 511 keV, f est la densité d'émission. Les variables pourvues d'un prime sont évaluées avec l'énergie du photon diffusé.

Cette méthode est efficace pour les répartitions de traceur hétérogène et est rapide d'exécution.

Cependant, elle nécessite une étape de mise à l'échelle du sinogramme des coïncidences diffusées, utilisant la distribution des coïncidences dans les queues du sinogramme (*tail-fit*), qui peut être instable lorsque la statistique dans ces queues est faible (scanner avec un faible rayon relativement à l'objet imagé). De plus, des erreurs peuvent être introduites par le manque de modélisation des coïncidences diffusées multiples, c'est-à-dire quand au moins un des photons d'annihilation est diffusé plus de une fois ou quand les deux photons sont diffusés au moins une fois.

Une approche, qui se popularise ces dernières années, repose sur l'utilisation de SMC pour estimer les fractions de coïncidences diffusées. Certaines méthodes utilisent une SMC accéléré sur *GPU* intégrant les effets Compton et photoélectrique pour simuler la propagation des photons d'annihilation dans le corps du patient, [Gaens *et al.*, 2013, Kim *et al.*, 2014]. Ces approches ont l'avantage de simuler tous les types de coïncidences (les non diffusées et les diffusées simples et multiples) et il n'est donc pas nécessaire de faire une mise à l'échelle du sinogramme des coïncidences diffusée sur les queues du sinogramme. Aujourd'hui, les puissances de calcul disponibles permettent de profiter de ce type de corrections basé sur des SMC. Il existe aussi une approche, introduite par [Guérin et El Fakhri, 2011], corrigeant la diffusion en intégrant dans la reconstruction l'information d'énergie des photons des coïncidences.

1.5.9 Intégration des effets modélisés

Il existe plusieurs manières pour intégrer à la reconstruction les coefficients de correction estimés avec les méthodes qui viennent d'être présentées. Avec toutes les corrections, l'algorithme *LM-OSEM* présenté dans l'équation 1.29 peut prend la forme suivante :

$$f_j^{(k,l+1)} = \frac{f_j^{(k,l)}}{\sum_{i=1}^{\mathrm{M}} a_{det_{ij}}} \sum_{i \in \mathrm{LM}_l} a_{det_{ij}} \frac{1}{q_i} \tag{1.46}$$

où $a_{det_{ij}}$ est le coefficient de la matrice associée à la réponse du détecteur pour le *voxel* j et la *LOR* i et q_i la projection calculée avec l'équation suivante :

$$q_i = sens_i \; scat_i + rnd_i - tm_i + att_i \left(\sum_{j=1}^{\mathrm{N}} a_{det_{ij}} \sum_{j'=1}^{\mathrm{N}} a_{positon_{jj'}} \; f_{j'}^{(k,l)} \right) \tag{1.47}$$

où i indique l'indice de la *LOR*, att_i est son coefficient d'atténuation, $sens_i$ celui de sensibilité, $scat_i$ celui de diffusion, rnd_i celui associé aux coïncidences fortuites et tm_i celui du temps-mort. $a_{positon_{jj'}}$ est le coefficient de la matrice modélisant le parcours du positon qui donne la probabilité qu'un positon émis dans la *voxel* j' vienne s'annihiler dans le *voxel* j.

1.5.10 Prise en compte de l'information de temps de vol

Dans la sous-section 1.3.4 nous avons vu que certain scanner récent permettait de mesurer la différence de temps de vol des photons d'annihilation permettant alors de connaître la position de l'annihilation le long de la *LOR*. Cette information peut être exploitée au moment de la reconstruction pour améliorer la qualité des images reconstruites [Moses, 2003, Surti *et al.*, 2006, Karp *et al.*, 2008,

Surti, 2015].

Il est possible de l'intégrer en mode sinogramme, en ajoutant une dimension à ce dernier. Chaque bin du sinogramme contient alors des sous-bins pour différents intervalles de positions le long de la *LOR*. Ensuite, il est nécessaire de construire une nouvelle *SRM* définissant la réponse associée à chaque sous-bin de ce sinogramme.

Avec le format de donnée en *list-mode*, l'information de temps de vol peut être exploitée au moment de la reconstruction lors des projections et rétroprojections en pondérant les coefficients de la *SRM* avec une distribution Gaussienne centrée sur la position d'annihilation donnée par le temps de vol et dont la *FWHM* est liée par la résolution temporelle du scanner par l'équation 1.4 [Popescu *et al.*, 2004, Popescu et Lewitt, 2004, Groiselle et Glick, 2004, Cui *et al.*, 2011a].

1.5.11 Prise en compte de la DOI ou de la POI

La forme allongée des cristaux détecteurs implique des erreurs de parallaxes. Il est alors possible de diminuer cet effet en utilisant la *POI* dans le cristal pour mieux localiser les *LOR*, comme le montre la figure 1.28, et ainsi améliorer la qualité des images reconstructives [MacDonald et Dahlbom, 1998].

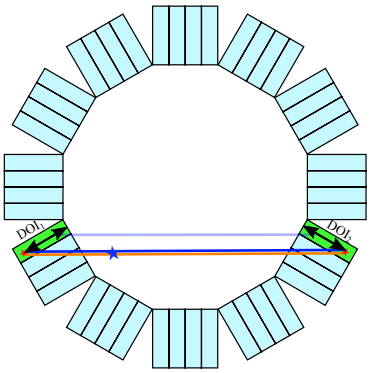


FIGURE 1.28 – Réduction des effets de parallaxe en utilisant la profondeur d'interaction.

La *DOI* et la *POI* peuvent facilement être exploité lors de la construction d'un sinogramme. En effet, ces positions peuvent être exploitées pour déterminer plus précisément le *pixel* du sinogramme impacté par une *LOR* donnée.

Concernant le format de données en *list-mode*, si la reconstruction repose sur une *SRM* stockée, il est alors nécessaire d'échantillonner les cristaux en sous-cristaux, pour différents intervalles de *DOI*. Cela a pour effet d'agrandir la taille de la *SRM*, ce qui rend cette approche difficilement envisageable. Avec un projecteur calculant à la volée les coefficients de la *SRM* il est possible de redéfinir un modèle analytique dépendant de l'information de *DOI*.

1.6 Conclusion

De nombreuses méthodes satisfaisantes existent pour ce qui est de la modélisation des effets d'atténuation, de la sensibilité variable des *LOR*, de la diffusion des photons d'annihilation, du temps-mort et des coïncidences fortuites affectant les données acquises par un scanner TEP. Pour tous les

effets présentés dans ce chapitre, il existe des méthodes d'estimation qui sont à la fois peu biaisées et peu bruitées. En effet, les facteurs d'atténuation peuvent être estimés de manière exacte (biais faible) et précise (variance faible) à partir de cartes tomodensitométriques acquises avec des scanners combinés TEP-TDM aujourd'hui extrêmement courants. Les facteurs de normalisation corrigeant la sensibilité peuvent être estimés avec une très bonne précision en utilisant des méthodes de factorisation ou en acquérant des fantômes de normalisation pendant une longue durée. De telles mesures ne dépendent pas du patient et doivent donc être faites une seule fois. La distribution spatiale des coïncidences fortuites est généralement mesurée par la méthode de la fenêtre retardée, qui est peu biaisée et peu bruitée grâce à l'utilisation de méthodes de réduction de variance prenant en compte la redondance de telles mesures. Le temps-mort peut être calculé précisément en utilisant des modèles mathématiques du comportement du temps-mort dans les blocs de cristaux et dans le système électronique. L'estimation des paramètres de ces modèles est faite une fois pour toute et de manière peu bruitée en utilisant des acquisitions longues de distributions simples. Cependant, certaines corrections ne sont pas satisfaisantes ou présentent des limites avec les évolutions actuelles, de formats et de matériels. La correction de diffusion avec l'algorithme *SSS* présente comme inconvénient de ne pas modéliser les coïncidences diffusées multiples et de nécessiter une étape de mise à l'échelle avec les données mesurées, instable en cas de faible statistique des données détectées en dehors du patient (scanners avec faible rayon, patients obèses). Cependant, en exploitant la puissance de calcul des *GPU*, il est aujourd'hui possible de modéliser précisément cet effet avec des SMC dans des temps compatibles avec les applications en clinique, comme le propose la méthode *multiple scatter simulation* (*MSS*) de [Kim *et al.*, 2014].

Depuis quelques années, le format *list-mode* s'impose face au mode sinogramme grâce à l'augmentation des capacités de stockage et à l'augmentation des puissances de calcul des ordinateurs, nécessaire pour une reconstruction *list-mode* rapide. Ce format, à la différence du mode sinogramme, permet de conserver sans dégradation des échantillonnages spatiaux et temporels fournis par le scanner, ce qui permet d'obtenir une meilleure résolution spatiale des images reconstruites [Rahmim *et al.*, 2005] et une meilleure correction des effets dynamiques, comme les mouvements respiratoires [Livieratos *et al.*, 2005, Lamare *et al.*, 2007]. Il permet également le stockage brut d'informations spécifiques à chaque coïncidence, comme le *TOF*, la *DOI* et l'énergie d'interaction. Cependant, ce format implique des temps de reconstruction plus importants que les méthodes basées sur le mode sinogramme, en particulier si on cherche à corriger l'ensemble des effets intervenant durant l'acquisition des données, ce qui est nécessaire pour une reconstruction qualitative et quantitative. L'utilisation de *clusters* de calcul est indispensable si on souhaite conserver des temps de reconstruction acceptables en routine clinique. Depuis le milieu des années 2000, les *GPU* ont été rendus programmables, ce qui a permis de les utiliser pour exécuter tout type de traitements, différents de celui de données graphiques pour lequel ils sont créés. Des *GPU* peuvent avoir une puissance de calcul équivalente à celle d'un *cluster* de calcul composé de plusieurs dizaines de machines. Plusieurs implémentations *GPU* de la reconstruction en TEP *list-mode* ont déjà été proposées. Cependant, l'utilisation d'un seul *GPU* n'est pas toujours suffisante. Néanmoins, il est possible d'intégrer plusieurs *GPU* dans un seul ordinateur. Nous proposons dans le chapitre 2 une méthode permettant de distribuer une reconstruction TEP *list-mode* sur une plate-forme multi-*GPU* permettant d'exploiter au mieux

de la puissance de calcul disponible.

Concernant la modélisation de la réponse du détecteur liée à sa physique et à sa géométrie, on trouve principalement des méthodes basées sur une préestimation et un stockage de sa matrice de réponse. Elles se basent sur des mesures empiriques, des SMC ou de modèles analytiques. Ces approches présentent cependant comme lourd inconvénient de devoir stocker l'immense matrice de réponse du détecteur, ce qui nécessite d'exploiter des méthodes de compression qui peuvent conduire à des pertes à cause des différentes approximations faites pour réduire la taille de la matrice. Malgré la compression, les matrices résultantes occupent généralement plusieurs dizaines de gigaoctets. Une reconstruction rapide avec ce type de matrices implique de disposer d'une plate-forme de reconstruction ayant des mémoires rapides et de grandes capacités. Une autre limite de cette approche est son manque de flexibilité, c'est-à-dire que, une fois la matrice estimée et stockée, les dimensions du champ de vue et des *voxels* ne peuvent plus être modifiées. Une autre approche repose sur des projecteurs qui permettent de calculer à la volée la réponse du détecteur ce qui permet de s'affranchir des problèmes de stockage et de flexibilité. De plus, cette approche permet d'exploiter plus efficacement les *GPU* que les méthodes basées sur une matrice stockée, qui est caractérisée par une grande puissance de calcul, mais une faible quantité de mémoire. Le temps de calcul à la volée de la réponse du détecteur impacte directement le temps de reconstruction, il est donc important qu'il soit aussi rapide que possible. Dans ce contexte, seules des méthodes analytiques sont envisageables. Plusieurs projecteurs ont déjà été proposés pour répondre à ce problème. Actuellement, les méthodes les plus précises reposent sur l'utilisation de fonctions Gaussiennes pour modéliser la réponse du détecteur. Cependant, ce type de distribution ne permet pas une modélisation précise de l'ensemble des effets physiques et géométriques, particulièrement de la diffusion intercristaux. Dans le chapitre 3 nous proposons un nouveau projecteur permettant de répondre à la problématique qui est d'estimer rapidement une réponse du système qui modélise l'ensemble des effets associés au détecteur du scanner. Une étude comparative, de ce projecteur avec une reconstruction basée sur une *SRM* préestimée par SMC et stockée, est présentée dans le chapitre 4.

Le parcours du positon est un effet souvent négligé, et généralement à juste titre puisque pour les isotopes comme le ^{18}F, il implique une perte de résolution inférieure à la résolution intrinsèque des scanners cliniques. Les isotopes injectés aux patients doivent posséder une demi-vie suffisamment brève pour qu'ils se désintègrent principalement pendant l'examen TEP et pas après, ce qui implique qu'ils doivent pouvoir être produits à proximité du lieu d'examen et qu'ils ne peuvent pas être stockés longtemps. Les isotopes communs sont produits par des accélérateurs de particules coûteux, qui se trouvent nécessairement proches des scanners pour la raison que nous venons d'évoquer. Cependant, l'utilisation de radionucléides tels que le ^{68}Ga ou le ^{82}Rb, produit par séparation chimique (chromatographie sur couche mince) de leur radionucléide père dont la demi-vie est compatible avec un stockage et un transport sur de longues distances, permet de se passer de cet accélérateur. Cependant, ces radionucléides émettent des positons beaucoup plus énergétiques, ce qui entraîne une dégradation importante des images reconstruites, en particulier lorsque le milieu objet est très hétérogène. Le chapitre 5 présente une méthode de simulation et de correction du parcours du positon basée sur une SMC simplifiée implémentée sur *GPU*, qui permet de tenir compte précisément des hétérogénéités de l'objet.

Chapitre 2

Accélération de la reconstruction TEP sur plate-forme multi-*GPU*

2.1 Introduction **48**
2.2 Le calcul sur *GPU* **48**
2.2.1 Historique 48
2.2.2 Différences entre architectures *CPU* et *GPU* 50
2.2.3 Spécificités de l'architecture *GPU* NVIDIA et de *CUDA* 51
2.2.3.1 Les processeurs de flux ou *streaming multiprocessors* 51
2.2.3.2 Les mémoires 52
2.2.3.3 Les opérations atomiques 53
2.2.3.4 L'implémentation 53
2.2.4 Limites du calcul sur *GPU* 55
2.3 Reconstruction LM-OSEM mono-*GPU* **55**
2.4 Reconstruction multi-*GPU* avec fractionnement des sous-ensembles de données *list-mode* **57**
2.4.1 Principes 58
2.4.2 Communication et occupation mémoire 60
2.4.3 En résumé 60
2.5 Reconstruction multi-*GPU* avec fractionnement du volume de reconstruction . . . **60**
2.5.1 Principes 61
2.5.2 Morceaux de volume de taille équivalente 61
2.5.3 Charge de travail équilibrée 62
2.5.4 Communication et occupation mémoire 65
2.5.5 En résumé 65
2.6 Étude d'évaluation **66**
2.6.1 Jeux de données simulés 66
2.6.2 Reconstructions 66
2.6.3 Matériels 67
2.7 Résultats **67**
2.7.1 Erreurs de reconstructions dues à la parallélisation 67
2.7.2 Temps d'exécution 69
2.8 Discussion et conclusion **72**

2.1 Introduction

En TEP, l'utilisation d'algorithmes de reconstruction itératifs commence seulement à se développer dans les appareils d'imagerie clinique pour des raisons de coût important en matière de temps de calcul, bien qu'ils fournissent des reconstructions de meilleure qualité [Lubberink *et al.*, 2004]. En routine clinique, où le temps est une denrée précieuse, cette évolution a été rendue possible grâce à l'augmentation constante de la puissance de calcul des *CPU* et par l'utilisation de *clusters*. Depuis maintenant dix ans, les processeurs graphiques (*GPU*), normalement dédiés aux traitements de données graphiques comme les images des jeux vidéo, ont été rendus capables d'exécuter tous types de calculs par les constructeurs. Depuis, ils sont devenus la solution privilégiée pour les problèmes de calcul intensif en raison de leur bon rapport puissance de calcul sur coût. Un seul ordinateur utilisant des *GPU* pourrait remplacer un *clusters* utilisés actuellement.

Dans le processus de reconstruction en TEP, les étapes de projection et la rétroprojection sont très coûteuses en calculs. Dans ce contexte, de nombreuses approches utilisant l'architecture *GPU* pour accélérer ces deux étapes ont été proposées. Certaines approches sont basées sur un projecteur calculant à la volée la réponse du système [Pratx *et al.*, 2006, Pratx *et al.*, 2009, Cui *et al.*, 2010, Cui *et al.*, 2011a, Cui *et al.*, 2011b, Cui *et al.*, 2011a, Bert et Visvikis, 2011, Ha *et al.*, 2012, Kinouchi *et al.*, 2012, Nassiri *et al.*, 2015] et d'autres sur une *SRM* stockée [Zhou et Qi, 2011]. Avec une reconstruction *list-mode* incluant toutes les corrections utiles, comme par exemple une modélisation précise de la réponse du détecteur, les temps de reconstruction restent encore trop importants pour une utilisation en routine clinique, malgré la puissance de calcul des *GPU*. Cependant, il est possible aujourd'hui d'intégrer jusqu'à 8 *GPU* dans un seul ordinateur grand public et plus encore dans des solutions professionnelles. La puissance combinée nous semble prometteuse pour réduire les temps de reconstruction, notamment avec l'objectif d'intégrer l'ensemble des corrections nécessaires en reconstruction TEP. Cependant, très peu d'approches permettant de paralléliser une reconstruction *LM-OSEM* sur une plate-forme multi-*GPU* ont été proposées [Cui *et al.*, 2011c, Cui *et al.*, 2013]. Dans ce chapitre, nous proposons une nouvelle méthode de parallélisation de reconstruction *LM-OSEM* sur une plate-forme multi-*GPU*. Cette méthode est générique et ne dépend pas du type de correction ni de la méthode de calcul de la réponse du système. Pour une raison de simplicité, une reconstruction avec un projecteur calculant à la volée la réponse du détecteur sera utilisé dans ce chapitre.

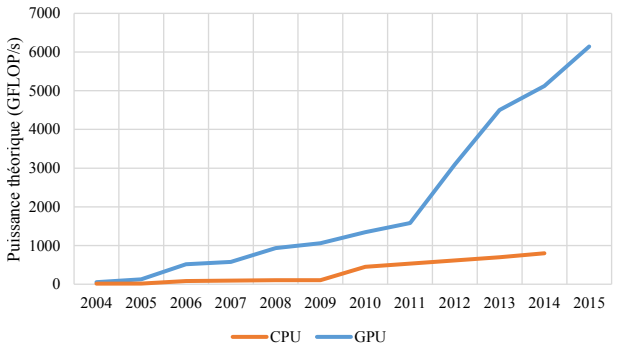
2.2 Le calcul sur *GPU*

2.2.1 Historique

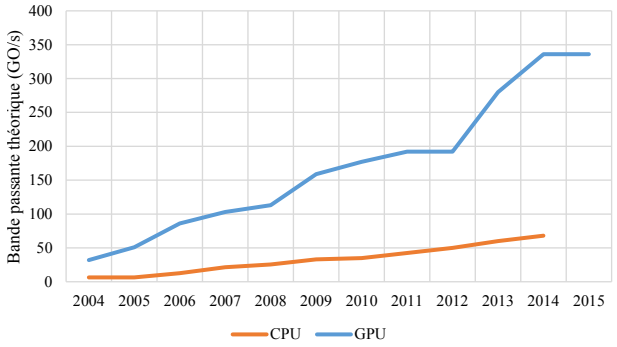
Les *CPU*, inventé en 1971 par Intel, ont pour objectif principal de traiter séquentiellement et aussi rapidement que possible des séries d'instructions. Pour répondre à ce problème, deux approches permettent d'améliorer les performances des *CPU*. La première consiste à augmenter le nombre de traitements qui peuvent être exécutés à chaque cycle de calcul, grâce l'intégration d'unités dédiées à certains calculs spécifiques ou à l'augmentation de la taille des mémoires cache réduisant les temps-morts dus aux communications avec la mémoire vive. La seconde approche vise à augmenter

le nombre de cycles de calcul effectués à chaque seconde, dicté par l'augmentation de la fréquence du *CPU*. Depuis plusieurs années, les fréquences des *CPU* ont atteint une limite et le nombre d'instructions exécutées à chaque cycle n'a que peu évolué. Depuis le début des années 2000, une troisième piste d'évolution des *CPU* a permis de poursuivre l'augmentation de la puissance de calcul en multipliant le nombre d'unités de calcul des *CPU*, qu'on appellent des cœurs. Cette évolution tire profit des systèmes d'exploitation multitâche où de nombreux programmes indépendants s'exécutent en parallèle. L'ajout de cœurs ne permet cependant pas d'augmenter la puissance de calcul lorsqu'il s'agit d'exécuter une tâche unique de manière totalement séquentielle.

Parallèlement au développement des *CPU*, les *GPU*, unités de calcul initialement dédiées à l'affichage graphique, ont connu un essor important dans le domaine du calcul scientifique avec le développement du calcul générique sur processeurs graphiques ou *general-purpose computing on graphics processing units* en anglais (*GPGPU*) à partir du milieu des années 2000. Cette évolution a été rendue possible par la possibilité de programmer ces *GPU* qui auparavant utilisaient des fonctions fixes. Leur succès vient de leur puissance de calcul et de leur bande passante mémoire bien supérieures à ce qui est disponible avec n'importe quel *CPU*, comme le montre la figure 2.1. Cette tendance n'a fait que s'amplifier durant les dix dernières années permettant aux *GPU* de devenir la solution privilégiée pour résoudre les problèmes de calculs intensifs [Nickolls et Dally, 2010].



(a) Puissance de calcul



(b) Bande passante mémoire

FIGURE 2.1 – Comparaison des puissances (a) et bande passante mémoire (b) des *CPU* et *GPU* entre 2004 et 2015.

À ses débuts, le calcul sur *GPU* ne disposait d'aucun outil de développement dédié et les développeurs étaient contraints de détourner des mécanismes créés à l'origine pour le rendu 3D. Les unités de calcul dédiées à l'exécution de programmes chargés de la gestion de la lumière en images de syn-

thèse qu'on appelle les *shaders* unifiés, pouvaient être utilisés pour exécuter des calculs génériques. L'architecture Tesla de NVIDIA a été la première à implémenter les *shaders* unifiés en 2006. Depuis, le travail des développeurs a été facilité par l'introduction de plusieurs interfaces de programmation ou *application programming interfaces* en anglais (*API*) dédiées à la programmation *GPU* : *compute unified device architecture* (*CUDA*) de NVIDIA en 2007, *open computing language* (*OpenCL*) du groupe Khronos proposée par Apple en 2008 et DirectCompute de Microsoft en 2009. Chacune présente des avantages et des inconvénients. DirectCompute ne fonctionne qu'avec les systèmes d'exploitation Windows à partir de la version 7 et avec les *GPU* compatibles avec DirectX 10 et 11. *CUDA* est *multi plate-forme* mais ne fonctionne qu'avec les *GPU* de marque NVIDI. Cette *API* est la pionnière concernant le *GPGPU*. Pour cette raison, elle est très utilisée et très bien documentée. OpenCL est aussi *multi plate-forme* et fonctionne sur toutes marques de *GPU*, bien qu'une implémentation optimisée pour une certaine plate-forme soit généralement incompatible avec les autres plates-formes.

En raison de sa maturité et de son importante communauté de développeurs, nous avons choisi d'utiliser l'*API CUDA*, et par conséquent l'architecture des *GPU* NVIDIA, pour l'ensemble de nos développements de programmes exécutés sur *GPU*.

2.2.2 Différences entre architectures *CPU* et *GPU*

Un *thread* représente une série d'instructions appliquée à un ensemble de données. Les *CPU* sont des unités de calcul généralistes consacrées à l'exécution de tous types de tâches. À l'origine mono-cœur, les *CPU* possèdent une architecture du type « une instruction unique pour une donnée unique » ou *single instruction single data* en anglais (*SISD*) et ne peuvent exécuter qu'un seul *thread* à un instant donné (deux avec la technologie *Hyper-Threading* de Intel). Sur un ordinateur classique, plusieurs dizaines de *threads* semblent s'exécuter en même temps, mais cette simultanéité d'exécution n'est que virtuelle. En réalité, le processeur alterne plusieurs milliers de fois par seconde entre chaque *thread*. Les processeurs multi-cœurs peuvent être vu comme plusieurs *CPU* autonomes intégrés sur une même puce, capables d'exécuter chacun un *thread* indépendemment des autres *CPU*. On nomme ce type d'architecture comme étant « des instructions multiples pour des données multiples » ou *multiple instructions multiple data* en anglais (*MIMD*).

Les *GPU* quant à eux sont des puces dédiées au calcul pour le rendu 3D. Dans ce contexte, les valeurs de chaque *pixel* de l'écran sont généralement obtenues par une série de traitements fixée et appliquée à différentes régions du champ de vue. Pour répondre efficacement à cette problématique, les constructeurs de *GPU* ont développés des puces à l'architecture de type « une instruction unique pour des données multiples » ou *single instruction multiple data* en anglais (*SIMD*), qui permettent d'exécuter en même temps plusieurs milliers de *threads* avec une série d'instruction identique, mais sur des données différentes.

La figure 2.2 schématise les différences entre un *CPU* et *GPU*. Chaque cœur possède son propre registre d'instruction, qui est une mémoire permettant de stocker les instructions à exécuter, et peut pour cette raison exécuter un *thread* différent d'un autre cœur. Dans un *GPU* les cœurs sont regroupés dans des processeurs de flux ou *streaming multiprocessors* en anglais (*SM*). Chacun de ces *SM* possède son propre registre d'instruction, les cœurs d'un même *SM* exécutent donc la même série

d'instruction à un instant donné, mais peuvent exécuter une série d'instruction différentes des cœurs d'un autre *SM*.

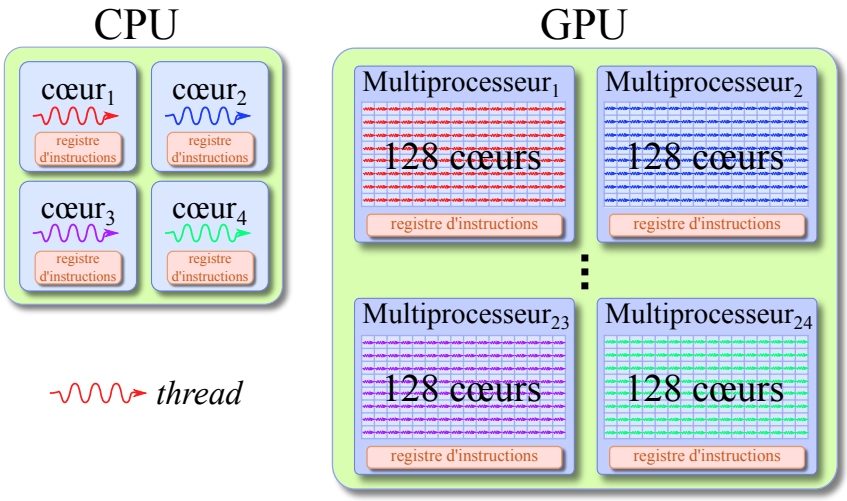


FIGURE 2.2 – Illustration des différences entre architecture *CPU* et architecture *GPU*. Chaque cœur d'un *CPU* est capable d'exécuter un *thread* (voir deux avec la technologie *Hyper-Threading* de Intel). Un *GPU* NVIDIA actuel est composé de quelques dizaines de *SM* (24 sur une architecture GM200) contenant chacun 128 cœurs pouvant exécuter chacun un *thread* à un instant donné, donc des milliers sur l'ensemble du *GPU*.

2.2.3 Spécificités de l'architecture *GPU* NVIDIA et de *CUDA*

L'architecture interne d'un *GPU* se divise en *SM* qui sont des grappes de cœurs exécutant les mêmes instructions. Cependant, à l'intérieur d'un *SM*, il existe de nombreux mécanismes qu'il est nécessaire de connaître pour implémenter efficacement un programme s'exécutant sur *GPU*.

TABLEAU 2.1 – Liste des microarchitectures des *SM* des *GPU* NVIDIA entre 2006 et 2015 et quelques-unes de leurs spécificités.

Microarchitecture	Cœurs par *SM*	Quantité de mémoire partagée par *SM*	nombre de registres par *SM*	nombre maximal de registres par *thread*	nombre maximal de *threads* résidents par *SM*	nombre maximal de blocs résidents par *SM*
Tesla (2006)	16	16 ko	8k à 16k	128	768	8
Fermi (2010)	32	48 ko	32k	63	1024	8
Kepler (2012)	192	48 ko	64k à 128k	63 à 255	1536	16
Maxwell (2014)	128	64 ko	64k	255	2048	32

2.2.3.1 Les processeurs de flux ou *streaming multiprocessors*

Depuis la première génération de *GPU* permettant d'utiliser *CUDA* (en 2006), la microarchitecture des *SM* n'a pas changée en profondeur. Les évolutions ont concerné le nombre de cœurs par *SM* et les capacités des différentes mémoires, comme on peut le voir sur le tableau 2.1. Les différentes générations de *SM* peuvent héberger des nombres variables de *threads* et de blocs de *threads*. Les

blocs sont des groupes de *threads* qui partagent de la mémoire (la mémoire partagée) et peuvent être synchronisé.

2.2.3.2 Les mémoires

Les *GPU* possèdent plusieurs types de mémoires avec des latences, bandes passantes, durées de vie et visibilités variables. On peut en distinguer une grande variété de mémoires, chacune étant dédiées à un certain type d'application.

D'abord, les registres sont des mémoires permettant de stocker les variables locales à un *thread*. C'est la mémoire la plus rapide, elle n'est pas gérée par le développeur, c'est au moment de la compilation que le nombre de registres associé à chaque *thread* est déterminé. Cette mémoire n'est visible qu'à partir du *thread* et sa durée de vie est égale à celle du *thread*. Ensuite, la mémoire partagée est accessible en lecture et en écriture par tous les *threads* d'un même bloc mais pas par le *CPU* et a une durée de vie égale à celle du bloc. C'est une mémoire hébergée par le cache de niveau L1 qui a donc une faible latence et une grande bande passante, mais une faible capacité (voir le tableau 2.1). Elle peut permettre de communiquer entre les *threads* d'un même bloc.

Ensuite, la mémoire globale est celle qui a la plus grande capacité (plusieurs Go) mais c'est aussi celle avec la latence la plus importante et la bande passante la plus faible. Elle est gérée manuellement par le développeur et elle sert généralement à stocker les données à traiter. Elle est accessible par tous les *threads* et par le *CPU* et a une durée de vie égale à celle du programme. Elle utilise la mémoire cache pour accélérer les accès redondants (voir les mémoires cache).

Ensuite, la mémoire locale, comme les registres, stocke des variables locales au *thread* et est gérée automatique par le compilateur. Elle est hébergée dans la mémoire globale et possède donc une bande passante et une latence similaire à la mémoire globale. Cette mémoire est utilisée lorsque le nombre maximal de registres par *thread* (voir le tableau 2.1) est atteint, ou pour les tableaux alloués dynamiquement dans les *threads*. Elle utilise la mémoire cache pour accélérer les accès redondants.

Ensuite, la mémoire constante est accessible en lecture seule et est hébergée dans la mémoire globale mais est aussi chargée en partie en mémoire cache constante. Elle est donc rapide, mais de petite taille. Elle peut être utilisée manuellement et est aussi exploitée automatiquement par le compilateur pour passer des arguments aux *threads*. Lorsque les données chargées dans cette mémoire ont une taille plus importante que la taille de la mémoire cache constante, seule une partie de ces données est chargée dans la mémoire cache en fonction des besoins du programme (gestion par le compilateur). Chaque fois que des données de la mémoire constante doivent être transférées vers la mémoire cache constante, la lecture se fait dans la mémoire globale avec la latence et la bande passante qui lui sont associées. La durée de vie de cette mémoire est égale à celle du programme.

Ensuite, la mémoire de texture est hébergée par la mémoire globale. Elle est accessible en lecture seule et les lectures se font à travers les unités de texture qui implémentent matériellement certaines interpolations. Cette mémoire est gérée manuellement et a une durée de vie égale à celle du programme. La mémoire cache est aussi utilisée pour accélérer les accès redondants.

Ensuite, les mémoires caches de texture, constant, de niveau L1 et de niveau L2 sont des mémoires locales à un *SM* pour les trois premières et globale à tous les *SM* pour la dernière. Elles sont gérées

automatiquement par le compilateur et permettent de stocker certaines valeurs de la mémoire globale, locale, constante ou de texture. Elles sont très rapides et possèdent une faible latence. Pour les mémoires globale, locale et de texture, les valeurs lues plusieurs fois sont chargées en mémoire cache afin d'accélérer toutes les relectures. La mémoire cache constante, quant à elle, charge systématiquement une partie de la mémoire constante, accélérant les accès sur les éléments chargés.

Parmi toutes ces mémoires, seules quatre d'entre elles sont gérées manuellement par le développeur, la mémoire partagée, la mémoire globale, la mémoire constante et la mémoire de texture. Avec les anciennes générations de *GPU*, la mémoire de texture présentait comme avantage, par rapport à la mémoire globale, d'utiliser les mémoires cache pour accélérer les lectures redondantes, mais aujourd'hui la mémoire globale utilise aussi ce mécanisme. La mémoire de texture représente quand même un avantage pour effectuer des interpolations linéaires, qui sont implémentées matériellement. En résumé, le développeur doit utiliser la mémoire globale pour toutes les données volumineuses, la mémoire constante pour les paramètres ou de petits tableaux, la mémoire de texture pour effectuer des interpolations linéaires et la mémoire partagée pour des petits tableaux dont les valeurs doivent être lues et modifiées un très grand nombre de fois.

2.2.3.3 Les opérations atomiques

Lorsque des données sont traitées en parallèle par plusieurs *threads* il peut arriver ce que l'on appelle une collision mémoire. Ce phénomène se produit lorsque au moins deux *threads* tentent de lire dans une même case mémoire puis d'y écrire en fonction de la valeur lue. On peut prendre comme exemple l'incrémentation. Si deux *threads* tentent d'incrémenter au même moment une valeur qui se trouve dans la même case mémoire, ceux-ci vont d'abord lire la valeur présente, par exemple 0 puis l'incrémenter de 1 et écrire le résultat de cette opération. Sans collision mémoire, la valeur finale devrait être de 2, mais comme les deux *threads* lisent exactement au même moment ils récupèrent tous les deux la valeur 0, qu'ils vont incrémenter de 1 pour écrire finalement tous les deux 1. Comme résultat la case mémoire va contenir la valeur 1 à la place de 2. Pour prévenir ce genre de problème, *CUDA* possède des fonctions qui sont dites atomiques qui permettent de verrouiller une adresse mémoire pour qu'elle ne soit accessible que par un *thread* à la fois. Lorsque plusieurs *threads* essayent d'y accéder en même temps, ceux-ci sont placés dans une file attente pour effectuer leur opération chacun leur tour.

2.2.3.4 L'implémentation

Dans une implémentation avec l'*API CUDA* on distingue trois types de fonctions. Nous avons d'abord le type hôte qui utilise le qualificatif *host* qui désigne les fonctions exécutées sur le *CPU*. Ce sont des fonctions C/C++ tout à fait classiques. Ensuite, les fonctions que l'on appelle *kernels* et qui sont désignées par le qualificatif *global*. Ces fonctions sont appelées par le *CPU* mais s'exécutent sur le *GPU*. Ce sont ces fonctions qui sont exécutées en parallèle. Enfin, les fonctions *device* désignées par le qualificatif du même nom, sont des fonctions exécutées sur le *GPU* qui ne peuvent être appelées que par d'autres fonctions *devices* ou par des *kernels*.

La fonction ci-dessous montre un exemple d'implémentation de la fonction faisant la somme des

vecteurs A et B dans C.

```
__global__ void somme(float* A, float* B, float* C, int taille)
{
    int i = blockIdx.x * blockDim.x + threadIdx.x;
    if(i<taille)        C[i] = A[i] + B[i];
}
```

FIGURE 2.3 – Exemple de *kernel* pour additionner deux vecteurs.

La fonction équivalente en C/C++ ferait une boucle sur l'indice i pour traiter chaque élément de la somme. Ici, la fonction somme n'est exécutée que pour un seul indice ou élément de la somme. Sur le *GPU*, à la différence du *CPU*, chaque élément de la somme est traité par un *thread*, mais un nombre de *threads* égal à la taille des vecteurs est exécuté. Afin que chaque *thread* traite un élément différent de la somme, celui-ci calcule son indice (première ligne de la fonction) et traite l'élément correspondant. L'indice absolu du *thread* n'est pas disponible directement et il faut le calculer à partir de son indice de *thread* "threadIdx.x" dans le bloc, de l'indice de son bloc "blockIdx.x" et de la taille du bloc "blockDim.x". Le nombre de *threads* que le *GPU* doit exécuter est fixé par le développeur en donnant une taille des blocs (comprise entre 1 et 1024 de manière générale), et un taille de grille aussi grande que nécessaire, qui correspond au nombre de blocs de *threads*. Le nombre de *threads* qui seront exécutes par le *GPU* est égal au produit de la taille des blocs par la taille de la grille. Une explication détaillée de la méthode à suivre pour fixer la taille des blocs est donnée en annexe C.

Précédemment, nous avons vu que les *GPU* possèdent plusieurs types de mémoires. Aucune de ces mémoires n'est accessible directement depuis le *CPU*. De même, les mémoires *CPU* ne sont pas accessibles depuis le *GPU*. Il existe heureusement des fonctions permettant de transférer des données de la mémoire *CPU* vers la mémoire *GPU* et inversement, mais elles ne peuvent être appelées que depuis des fonctions *hosts*. Il est aussi nécessaire d'allouer de la mémoire sur le *GPU* avant de pouvoir transférer les données. Les allocations et copies mémoires nécessaires à l'exécution de notre précédent exemple, sont présentées dans la figure 2.4.

```
    // Allocation de l'espace mémoire nécessaire sur le GPU
    float *A_gpu, *B_gpu, *C_gpu;
    cudaMalloc(A_gpu, sizeof(float)*taille);
    cudaMalloc(B_gpu, sizeof(float)*taille);
    cudaMalloc(C_gpu, sizeof(float)*taille);
    // Copie des données de la mémoire CPU vers la mémoire GPU
    cudaMemcpy(A_gpu, A, sizeof(float)*taille, cudaMemcpyHostToDevice);
    cudaMemcpy(B_gpu, B, sizeof(float)*taille, cudaMemcpyHostToDevice);
    // Calcul de la somme sur le GPU
    somme<<<taille_grille, taille_bloc>>>(A_gpu, B_gpu, C_gpu, taille);
    // Copie des données de la mémoire GPU vers la mémoire CPU
    cudaMemcpy(C, C_gpu, sizeof(float)*taille, cudaMemcpyDeviceToHost);
```

FIGURE 2.4 – Exemple d'exécution d'un *kernel* avec les allocations et copies mémoires nécessaires à l'envoi et à la récupération des données sur la mémoire du *GPU*.

Les trois premières allocations permettent de créer les vecteurs A, B et C dans la mémoire du *GPU*. Ensuite, les deux copies envoient les données contenues dans les vecteurs A et B sur le *CPU* vers la

mémoire du *GPU*, dans A_{GPU} et B_{GPU}. Ensuite, le *kernel* est exécuté. Pour finir, la copie permet de récupérer sur la mémoire du *CPU* le résultat de la somme de A et B, qui est stocké sur la mémoire *GPU* dans C_{GPU}. Une fois ces traitements effectués, il faut évidemment libérer les espaces mémoire alloués, ce qui n'est pas présenté dans cet exemple.

2.2.4 Limites du calcul sur *GPU*

Les facteurs d'accélération obtenus par un portage sur *GPU* peuvent varier de manière importante pour différentes fonctions. La capacité d'un algorithme à être parallélisé est le point le plus important. Cependant, même pour des algorithmes parallélisables les facteurs d'accélération peuvent varier de manière importante. Il est difficile d'estimer les gains d'une implémentation *GPU* et il faut généralement réaliser l'implémentation pour connaître les bénéfices réels.

Pour tirer pleinement profit de la puissance d'un *GPU*, il est généralement nécessaire de régler finement différents paramètres dépendant de l'architecture du *GPU* considéré ou d'utiliser des mécanismes spécifiques, ce qui peut réduire son efficacité sur d'autres architectures voir même le rendre incompatible. Plusieurs types de mémoires peuvent être exploitées pour accélérer une implémentation. Cependant, chacune de ces mémoires n'est adaptée qu'à certains contextes bien spécifiques. De plus, leurs caractéristiques ont évolué depuis les débuts de CUDA et évoluent toujours. Plusieurs études montrent ces limites intrinsèques aux *GPU* actuels [Vuduc *et al.*, 2010, Malits *et al.*, 2012].

Une contrainte importante pour le développeur vient de l'impossibilité d'appeler dans un *kernel* ou une fonction *device*, des fonctions de librairies externes C/C++. Il est donc nécessaire d'implémenter toutes les fonctions utiles au programme. Toutefois, de plus en plus de projets se développent autour de l'*API CUDA*, ce qui permet de trouver de plus en plus de librairies compatibles.

2.3 Reconstruction LM-OSEM mono-*GPU*

La reconstruction *LM-OSEM*, présenté dans le paragraphe 1.4.2.2, comporte quatre étapes principales exécutées séquentiellement et répétées en boucle qui peuvent être mise sous la forme du diagramme présenté dans la figure 2.5.

Dans notre implémentation de l'algorithme *LM-OSEM* chacune de ces étapes est effectuée par un *kernel* spécifique. L'ensemble des données nécessaires à la reconstruction sont chargées sur la mémoire du *GPU* avant de commencer la reconstruction. Comme nous l'avons expliqué précédemment, un *GPU* est un processeur permettant l'exécution en parallèle de plusieurs milliers de traitements ou *threads*. Pour profiter de la puissance disponible, il est donc nécessaire de découper les calculs en un ensemble de tâches indépendantes qui pourront alors être exécutées en parallèle sans avoir à communiquer. Les tâches de correction et de mise à jour sont de simples multiplications ou division de vecteurs, terme à terme. Pour ces opérations, chaque *thread* traite un des termes de ces calculs, c'est-à-dire une *LOR* pour la correction et un *voxel* pour la mise à jour. Pour les opérations de projection et rétroprojection, nous avons choisi de traiter une *LOR* par *thread*. Ici, nous avons utilisé un projecteur qui modélise la réponse du détecteur associée à une *LOR* par un tube dont chaque section transversale est une distribution Gaussienne 2D invariante dans le champ de vue, dont la taille est

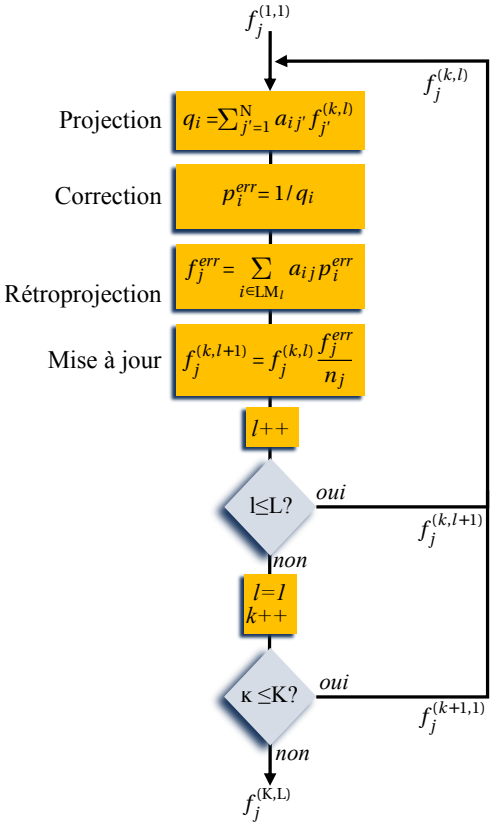


FIGURE 2.5 – Diagramme de la reconstruction *LM-OSEM*.

fixée à partir de mesures de la *PSF* du détecteur [Pratx *et al.*, 2006, Cui *et al.*, 2010, Cui *et al.*, 2011c]. Ce projecteur est illustré sur la figure 2.6.

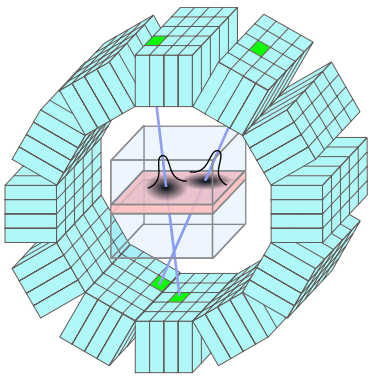


FIGURE 2.6 – La modélisation de la partie de la SRM associée au détecteur pour une *LOR* est réalisée par la construction d'un tube de *voxels* dont chaque coupe transversale est une distribution Gaussienne 2D dont les variances sont fixées par la *PSF* du détecteur. Ce principe est ici représenté pour deux *LOR* dans une coupe du champ de vue.

Nous avons choisi ce projecteur pour cette étude parce que l'utilisation de fonctions Gaussiennes pour modéliser la réponse du détecteur est une approche commune et relativement simple à implémenter, tout en nécessitant une charge de travail pour le *GPU* qui est classique pour un projecteur de ce type. Il existe une implémentation optimisée de ce projecteur utilisant la mémoire partagée du

GPU [Cui *et al.*, 2011a], mais elle est limitée à des tailles de volume faibles et est complexe à mettre en œuvre. Dans ce chapitre, nous nous basons sur une implémentation *GPU* qui est un portage direct de l'implémentation *CPU* de ce projecteur. La répartition de la charge de travail sur le *GPU* s'opère en assignant à chaque *thread* une *LOR*, comme le montre la figure 2.7. Pour fournir la projection associée à une *LOR*, pendant la reconstruction, un *thread* va parcourir la *LOR* en accumulant les valeurs des *voxels* de chaque coupe pondérées par la distribution Gaussienne 2D. La rétroprojection d'une *LOR* par un *thread* consiste à incrémenter les *voxels* du volume de rétroprojection par l'erreur de projection (noté p_i^{err} dans la figure 2.5) pondérée par la même distribution Gaussienne 2D qu'à la projection.

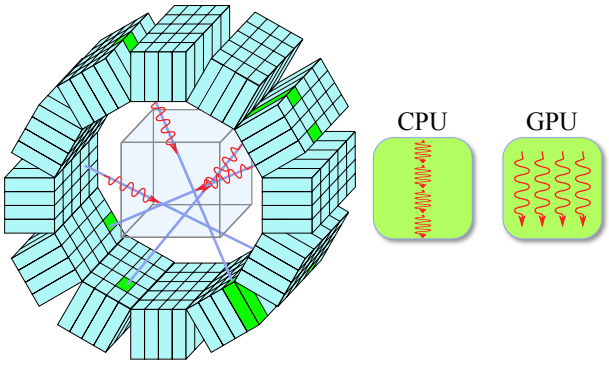


FIGURE 2.7 – Représentation de l'exécution des opérations de projection et rétroprojection sur *CPU* et sur *GPU*. Chaque *thread* du *GPU* traite une seule *LOR* tandis que sur le *CPU* toutes les *LOR* sont traitées séquentiellement par un seul *thread*.

Tous les *threads* rétroprojetant les erreurs de projection dans le même volume, il est possible que deux (ou plus) d'entre eux tentent d'incrémenter la valeur en un même *voxel* au même moment. Pour prévenir ce problème, nous utilisons donc les fonctions atomiques de *CUDA*. Cependant, cette solution peut être préjudiciable aux performances. En effet, si un grand nombre de *threads* tentent d'accéder à un même *voxel* au même moment, l'accès va être séquentialisé et aura pour conséquence la mise en attente de nombreux *threads*. En pratique, le nombre de *voxels* étant grand face au nombre de *threads* s'exécutant à un instant donné, il est très peu probable qu'un grand nombre d'entre eux tente d'accéder au même *voxel* au même moment. Pour cette raison, l'utilisation des fonctions atomiques n'a pas un impact significatif sur les performances, dans ce cas.

2.4 Reconstruction multi-*GPU* avec fractionnement des sous-ensembles de données *list-mode*

Jusqu'à présent, l'ensemble des méthodes de parallélisation de la reconstruction en TEP se fondent sur une répartition des données d'entrées aux *GPU*. C'est l'approche la plus intuitive et c'est aussi celle qui est utilisée pour la parallélisation de la reconstruction tomographique, qui est d'ailleurs celle utilisée pour la parallélisation sur *GPU* (voir section 2.3).

Avant l'apparition du *GPGPU* de nombreuse implémentations parallélisées sur *clusters*, de la reconstruction en TEP avaient déjà été proposées. Les plates-formes PARAPET [Labbé *et al.*, 1999],

HeinzelCluster [Vollmar *et al.*, 2000, Vollmar *et al.*, 2002] et d'autres implémentations [Chen *et al.*, 1991, Shattuck *et al.*, 2002] ont été développées pour exécuter des reconstructions itératives sur des *clusters* de calcul. Dans toutes ces implémentations, la répartition de la charge de travail se fait par le découpage du sinogramme sur les différentes machines constituant le *cluster* de calcul.

Des implémentations parallélisées des reconstructions *list-mode* ont aussi déjà été proposées. L'implémentation de [Wang *et al.*, 2006] permet une reconstruction multi-*CPU* basée sur un découpage des données *list-mode* avec information du *TOF* et un équilibrage de la charge de travail en estimant la charge associée à chaque *LOR*. Il existe plusieurs implémentations multi-*GPU* de reconstructions *list-mode* pour le TEP [Cui *et al.*, 2011c, Cui *et al.*, 2013] et la tomographie proton [Karonis *et al.*, 2013], utilisée en protonthérapie pour évaluer la dose déposée. Encore une fois, toutes ces implémentations se basent sur un découpage des données en entrée, le fichier *list-mode* ici.

2.4.1 Principes

L'ensemble des méthodes de parallélisation des reconstructions itératives *list-mode* se basent sur un fractionnement des données d'entrée, c'est-à-dire du fichier *list-mode*. Nous allons voir ici comment l'algorithme *LM-OSEM* (voir section 1.4.2.2) est implémenté avec cette approche de parallélisation. L'idée générale de cette méthode repose sur la découpe des sous-ensembles de données *list-mode* traités à chaque itération de l'algorithme *LM-OSEM*.

Le principe général d'une reconstruction sur un *cluster* avec cette parallélisation est représenté dans l'organigramme 2.8. On retrouve les principales étapes de la reconstruction *LM-OSEM* présentée dans la sous-section 2.3, la projection, la correction, la rétroprojection et la mise à jour notées respectivement P, C, B et U, exécutées en boucle pour chaque sous-ensemble et chaque itération. À ces quatre étapes de calcul s'ajoutent deux étapes de communication entre les différents *GPU* ou nœuds du *cluster*, la répartition des morceaux des sous-ensembles de données *list-mode* et du volume reconstruit à l'itération précédente (D) et le rassemblement des volumes d'erreurs (R). La première étape, la répartition consiste à découper équitablement le sous-ensemble de données *list-mode* et à transmettre chacun de ces morceaux à un *GPU* spécifique. Ensuite, chaque morceau du sous-ensemble est traité indépendamment pour les étapes de projection, correction et rétroprojection. À la fin de ces trois étapes, chaque *GPU* a généré un volume d'erreurs qui est associé à son morceau de sous-ensemble. Tous les volumes d'erreurs sont transmis à un *GPU* hôte qui les somme, afin d'obtenir le volume d'erreur global associé au sous-ensemble complet. Ce *GPU* procède ensuite à la mise à jour du volume de reconstruction en le multipliant par le volume d'erreur, lui-même divisé par le volume de normalisation.

La charge de travail est divisée sur les N *GPU* disponibles par la division du nombre de *LOR* à traiter par un facteur N, comme représenté sur la figure 2.9 avec 2 *GPU* et 4 *LOR*. La charge de travail associée à une *LOR* n'est pas une quantité constante et en pratique, elle dépend principalement du nombre de *voxels* qui compose le tube Gaussien associé, ou à sa longueur d'intersection avec le volume de reconstruction, si on considère que les sections de toutes les *LOR* varient peu. On pourrait alors imaginer que certains *GPU* se voient assigner des morceaux du sous-ensemble dont la charge de travail associée est plus importante que ceux assignés à d'autres *GPU*. Dans un tel contexte, cer-

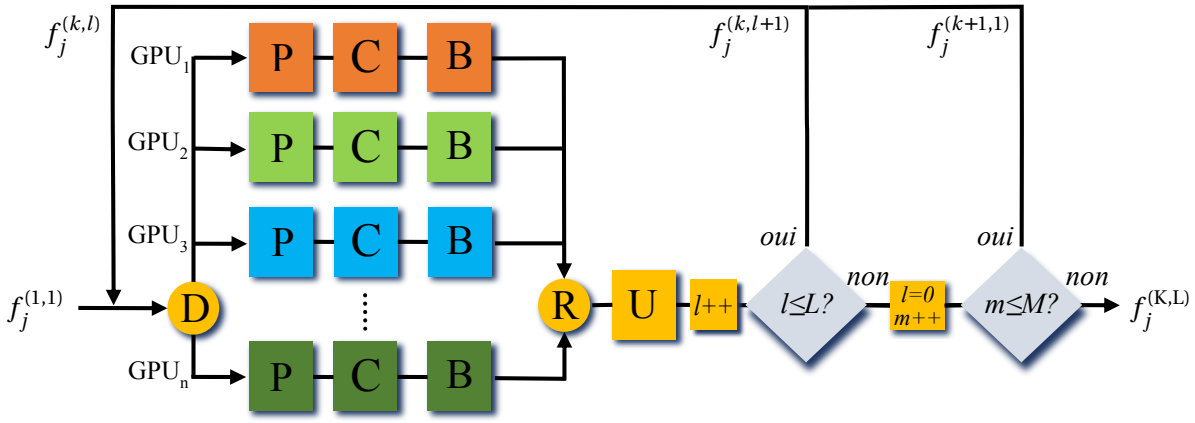


FIGURE 2.8 – Organigramme de la reconstruction *LM-OSEM* avec la parallélisation basée sur le fractionnement des sous-ensembles de données *list-mode* sur N *GPU*. Les étapes de projection, correction, rétroprojection et mise à jour sont notées P, C, B et U respectivement. Deux étapes de communication s'ajoutent, la répartition des morceaux de sous-ensembles notée D, et le rassemblement des volumes de rétroprojection, noté R. L'indice du sous-ensemble est noté l et l'indice d'itération m.

tains *GPU* mettront plus de temps à traiter leur morceau du sous-ensemble et imposeront aux autres *GPU* de les attendre à chaque communication. La vitesse globale de la reconstruction est contrainte de suivre la vitesse du *GPU* le plus lent. En pratique, les *LOR* dans un fichier *list-mode* sont organisées de manière aléatoire, et il en est de même dans les sous-ensembles et morceaux de sous-ensembles. Pour cette raison, la charge de travail variable associée aux *LOR* est répartie équitablement sur les morceaux de sous-ensembles de données *list-mode* des *GPU*.

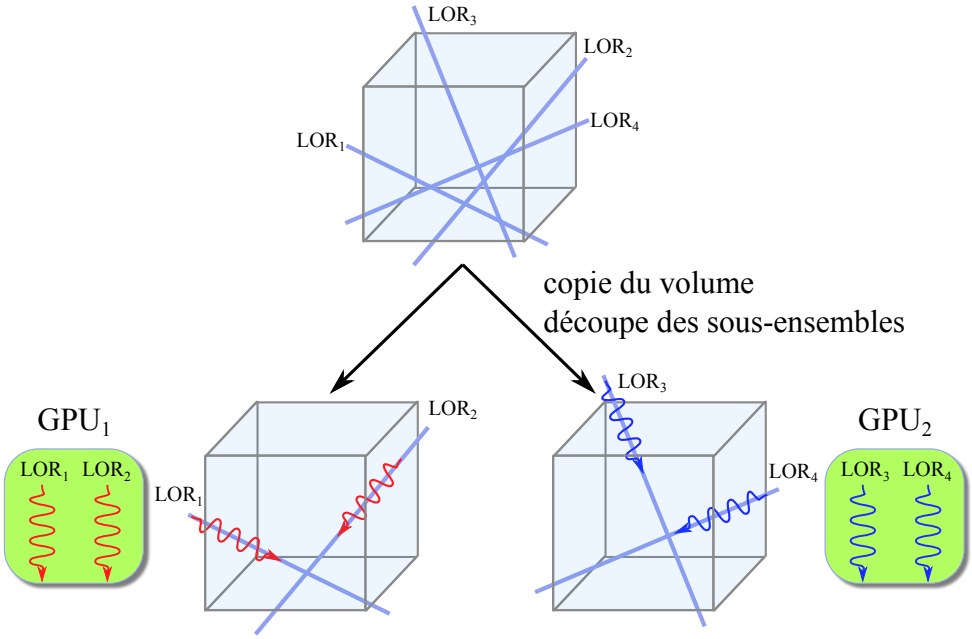


FIGURE 2.9 – Schéma des données traitées par chaque *GPU* avec la méthode de fractionnement des sous-ensembles. Chaque *GPU* traite un morceau du sous-ensemble de données *list-mode* sur une copie complète du volume de reconstruction.

2.4.2 Communication et occupation mémoire

Pour une reconstruction sur N *GPU*, chacun d'eux est lié à un cœur de *CPU* qui lui envoie les commandes à exécuter et qui récupère les données dans la mémoire du *GPU* pour effectuer les communications. Chaque couple *GPU* et cœur de *CPU* forme un nœud du *cluster*. Dans cette liste de *GPU*, l'un d'eux est désigné comme *maitre* pour commander les étapes de communication et de mise à jour, et les autres comme *esclaves*. Le système *message passing interface* (*MPI*) est utilisé pour communiquer les données entre les *GPU*. À chaque boucle, pour l'étape de répartition, $N-1$ copie du volume de reconstruction sont transmises par le *maitre* vers les autres *GPU*. Les étapes suivantes sont traitées indépendamment sur chaque *GPU* avec la même implémentation que celle décrite dans la section 2.3. Ensuite, après la rétroprojection, les *GPU esclaves* transmettent les volumes d'erreurs au *maitre*, qui les somme avec son propre volume d'erreur, et effectue la mise à jour. Ainsi, pour une itération avec L sous-ensembles, c'est $2(N-1)L$ fois la taille mémoire du volume de reconstruction qui est transmise.

Concernant l'utilisation mémoire, chaque *GPU* conserve deux volumes complets, le volume de reconstruction à l'itération courante et le volume d'erreur. Il stocke aussi les morceaux de sous-ensembles de données *list-mode* qui sont chargées une seule fois au moment de l'initialisation de la reconstruction. Si on note V la taille mémoire du volume de reconstruction, L la taille mémoire des données *list-mode* et P celle des projections associées à chaque *LOR*, la mémoire occupée sur un *GPU* est égale à $2V + \frac{L+P}{N}$.

2.4.3 En résumé

La méthode de parallélisation de la reconstruction par fractionnement des sous-ensembles de données *list-mode* permet de répartir efficacement la charge de travail sur les différents *GPU*. Certaine *LOR* nécessite une charge de travail plus importante, mais l'organisation aléatoire du fichier *list-mode* permet de répartir équitablement les *LOR* associées à des charges de travail élevées. Cette méthode permet de réduire l'occupation mémoire associée aux données *list-mode* mais pas l'occupation mémoire associée au volume de reconstruction. Avec cette approche, il est nécessaire de communiquer un volume de reconstruction complet pour chaque *GPU* à chaque boucle. Plus le nombre de sous-ensembles est important et plus le nombre de boucle par itération l'est aussi. Donc la quantité de données à transmettre pendant la reconstruction est proportionnelle au nombre de sous-ensembles multiplié par le nombre d'itérations. Cela peut limiter les performances si le volume de reconstruction contient beaucoup de *voxels* et/ou si la reconstruction utilise beaucoup de sous-ensembles.

2.5 Reconstruction multi-*GPU* avec fractionnement du volume de reconstruction

Dans cette section, nous proposons une nouvelle approche de parallélisation de la reconstruction *LM-OSEM* sur une plate-forme multi-*GPU* qui repose sur la réduction de la portion de champ de vue traitée par chaque *GPU*. Cette approche peut cependant être adaptée à d'autres algorithmes itératifs.

Le but de cette méthode de parallélisation est de répartir équitablement la charge de travail et l'occupation mémoire sur les différents *GPU*, tout en minimisant le volume de données communiquées à chaque itération entre eux.

2.5.1 Principes

La découpe du volume de reconstruction se fait en amont de la reconstruction et chaque *GPU* reçoit dans un premier temps sa portion de volume. Ensuite, tous les *GPU* procèdent à la projection du sous-ensemble dans son intégralité et transmettent leurs résultats vers le *GPU maitre* qui va les sommer afin de procéder à la correction. L'erreur de projection ainsi générée est transmise vers tous les *GPU* où elle est rétroprojetée sur chaque morceau de volume d'erreur avant de procéder à la mise à jour. À la fin de la dernière itération, tous les morceaux de volume sont transmis au *GPU maitre* et recombinés pour former le volume reconstruit complet. Ce flux de travail est représenté par un organigramme dans la figure 2.10.

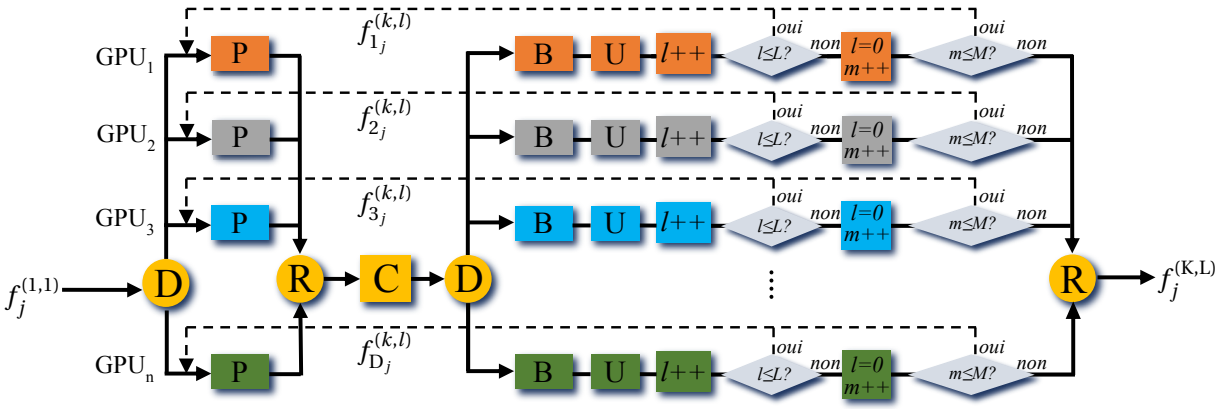


FIGURE 2.10 – Organigramme de la reconstruction *LM-OSEM* avec la parallélisation basée sur le fractionnement du volume de reconstruction sur N *GPU*. L'étape de projection est effectuée par chaque *GPU* sur le sous-ensemble complet, chacun sur sa partition de volume. Les projections générées sont transmises au *GPU maitre* pour être sommées, effectuer la correction avant de transmettre les erreurs de projection aux *GPU esclaves*. Ensuite, la rétroprojection et la mise à jour sont effectuées indépendamment. L'indice du sous-ensemble est noté l et l'indice d'itération m.

Comme le montre la figure 2.11, chaque *GPU* n'héberge qu'une fraction du volume à reconstruire et l'intégralité du sous-ensemble. La réduction de la charge de travail est apportée par la diminution de la quantité de *voxels* à traiter pour une *LOR* donnée.

À la différence de la méthode de parallélisation basée sur le fractionnement des sous-ensembles de données *list-mode* présentée dans la section 2.4, cette méthode repose sur une division du volume de reconstruction, dont nous proposons deux approches de découpe.

2.5.2 Morceaux de volume de taille équivalente

Nous avons développé deux stratégies de découpage du volume de reconstruction. La première est la plus simple, elle consiste en une découpe en volumes de taille équivalente. Il existe un grand nombre de manières de découper un volume suivant ses trois axes. La manière dont le volume est découpé a une importance primordiale car la charge de travail associée à chaque région du champ

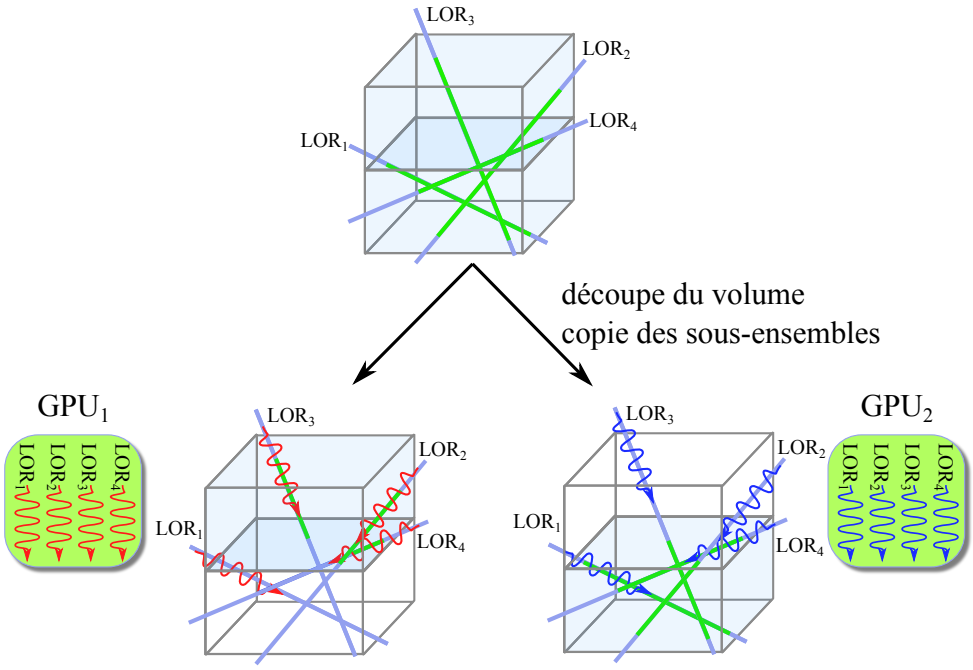


FIGURE 2.11 – Schéma des données traitées par chaque *GPU* avec la méthode du volume de reconstruction. Chaque *GPU* traite le sous-ensemble de données *list-mode* complet, mais sur un morceau du volume de reconstruction. La charge de travail associé à une *LOR* est représentée par le segment vert clair, plus il est long et plus la charge de travail est importante.

de vue varie de manière importante et, comme nous l'avons vu précédemment, la vitesse globale de la reconstruction est dictée par la vitesse du *GPU* le plus lent (celui qui a la charge de travail la plus importante).

La première approche de découpe que nous avons choisi consiste à diviser le volume en sous-volumes dont les proportions suivant chaque axe restent le plus équilibré possible, comme représenté dans la figure 2.12. C'est-à-dire que, lorsque cela est possible, les découpes se répartissent équitablement sur les trois axes de l'espace (par exemple, une découpe sur chaque axe avec 8 *GPU*). Cette approche permet de mieux répartir la charge de travail qu'une découpe qui serait faite toujours suivant le même axe et qui amènerait à des portions de volume plates sur les bords du champ de vue où peut de *LOR* passent, et donc avec une faible charge de travail associée. Pour certains nombres de *GPU* (par exemple, un nombre premier), il n'est pas possible de faire autrement que de couper le volume suivant un même axe. De plus, même lorsque cette découpe est possible, l'activité n'est pas répartie de manière homogène dans le champ de vue. Il y aura donc forcément des régions du volume traversées par un plus grand nombre de *LOR* et en conséquence liées à un charge de travail plus importante.

Par la suite, nous appellerons cette méthode de parallélisation volume(taille).

2.5.3 Charge de travail équilibrée

Pour remédier à la mauvaise répartition de la charge de travail associée à une découpe en volumes de même taille, nous proposons une découpe en sous-volumes associés à une même charge de travail. Pour effectuer cette découpe, il est nécessaire dans un premier temps d'estimer la charge

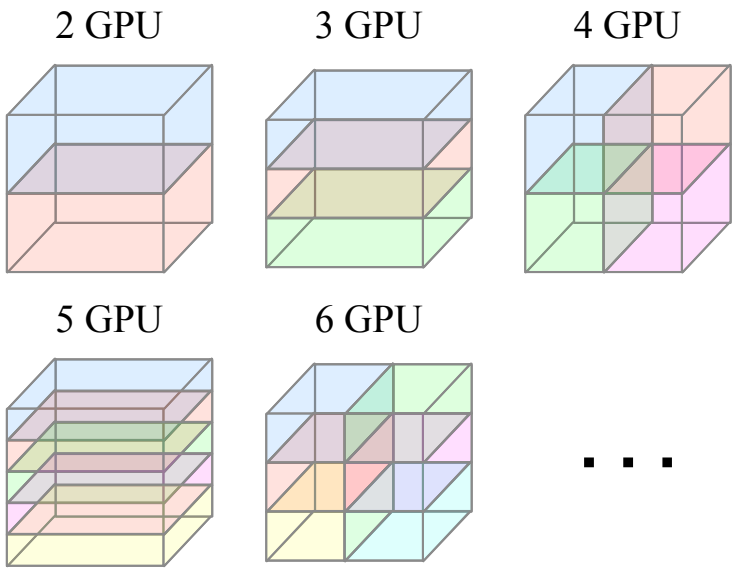


FIGURE 2.12 – Découpe du volume de reconstruction à taille équivalente pour 2 à 6 *GPU*. Chaque couleur est allouée à un *GPU* spécifique.

de travail associée à chaque région du champ de vue. La charge de travail associée à un *voxel* est proportionnelle au nombre de fois où il est affecté par la réponse d'une *LOR* du jeu de données considéré. Pour simplifier l'estimation, nous considérons que cette valeur est aussi proportionnelle au nombre de *LOR* traversant ce *voxel*. Il est alors possible d'estimer cette charge pour chaque *voxel*, comme cela est représenté dans la figure 2.13. La source utilisée dans ce cas était particulièrement homogène, on remarque que les variations de charge de travail sont principalement dues aux variations spatiales de la sensibilité du détecteur. Il n'est cependant pas nécessaire d'estimer cette charge de travail en

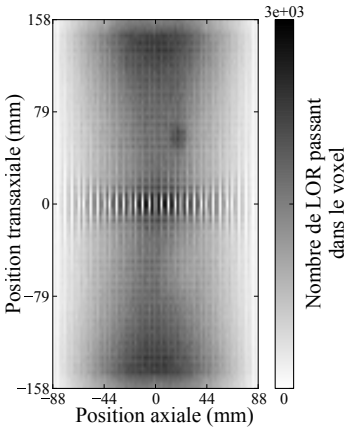
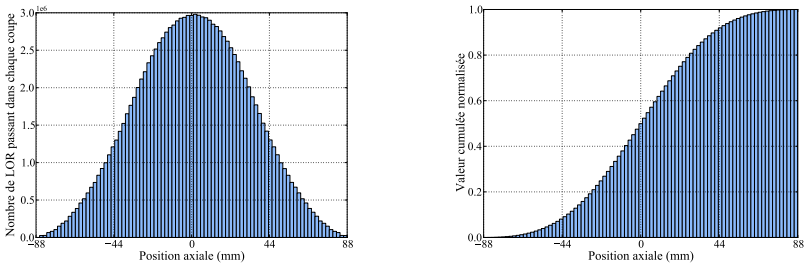


FIGURE 2.13 – Coupe sagittale d'une estimation de la charge de travail. La valeur en chaque *voxel* représente le nombre de *LOR* passant par ce *voxel*.

chaque *voxel*. Il serait d'ailleurs compliqué d'exploiter cette répartition de charge de travail 3D pour définir une découpe du champ de vue. En fixant une direction de découpe, il suffit d'estimer la charge de travail pour les coupes du volume perpendiculaires à la direction choisie. De manière arbitraire, nous fixons la direction axiale comme axe de découpe. On obtient ainsi une charge de travail 1D dont on peut voir un exemple dans la figure 2.14a, très rapide à estimer. La découpe doit être faite

de telle sorte que chaque morceau de volume couvre un intervalle axial dont l'intégrale de la courbe de travail (l'aire sous la courbe) soit égale à l'intégrale complète divisée par le nombre de *GPU*. Pour trouver cette découpe, nous pouvons utiliser la somme cumulée et normalisée des coefficients de la fonction de la charge de travail, représentée dans la figure 2.14b. Avec cette fonction, la charge de travail normalisée d'un intervalle axial est donnée directement par la différence de la charge de travail cumulée en fin d'intervalle et de la charge de travail cumulée en début d'intervalle. Par exemple pour le $i^{ième}$ *GPU* parmi N, les positions axiales de début et de fin du morceau de volume sont les positions sur la courbe 2.14b ayant les valeurs en ordonné les plus proches de $\frac{i-1}{N}$ et $\frac{i}{N}$ respectivement. Cette procédure de découpe est schématisée dans la figure 2.15.



(a) (b)

FIGURE 2.14 – (a) la charge de travail de chaque coupe transverse du champs de vue et (b) la charge de travail cumulée.

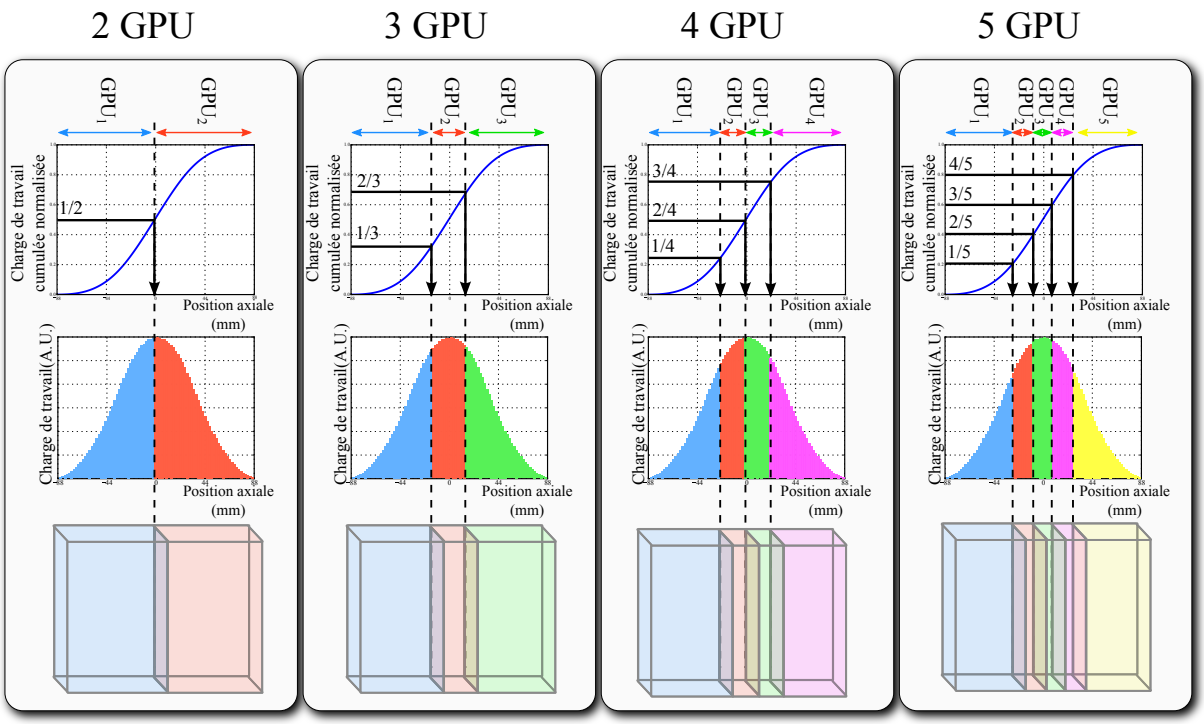


FIGURE 2.15 – Schéma de la découpe du volume de reconstruction à charge de travail équivalente pour 2 à 5 *GPU*. Les positions de découpe sont définies par une valeur de la charge de travail cumulée normalisée égale à un multiple de 1/N, N étant le nombre de *GPU*.

Par la suite, cette méthode de parallélisation sera nommée volume(travail).

2.5.4 Communication et occupation mémoire

Comme pour la reconstruction avec fractionnement des sous-ensembles de données *list-mode*, chaque *GPU* est associé à un cœur de *CPU* formant un nœud, où le cœur *CPU* contrôle le *GPU* et communique avec les autres cœurs *CPU* en utilisant la technologie *MPI*. Un des *GPU* est aussi désigné comme *maître* et les autres comme *esclaves*.

À chaque boucle de l'algorithme *LM-OSEM* toutes les communications portent sur les valeurs des projections. Le mode de découpe du volume n'a donc pas d'influence sur la quantité de données échangées pendant les étapes de communication. À la différence de la méthode de fractionnement des sous-ensembles, la quantité de données transmises pour une itération est constante relativement au nombre de sous-ensembles, sachant que l'ajout de sous-ensembles réduit la taille de chaque sous-ensemble. Ainsi, pour une itération d'une reconstruction sur N *GPU*, on compte 2N fois la taille mémoire des projections qui sont transmises.

La méthode de découpe du volume va avoir un impact sur la quantité de mémoire utilisée sur chaque *GPU*. Pour la méthode de découpe basée sur une répartition équitable en terme de taille des morceaux de volumes, chaque *GPU* aura une même occupation mémoire qui sera de $3\frac{V}{N}$ pour les morceaux des volumes de reconstruction, d'erreur et de normalisation, et de $L+P$ pour les données *list-mode* et les projections. La valeur totale est donc de $3\frac{V}{N}+L+P$. Avec la méthode de fractionnement basée sur la répartition de la charge de travail, la quantité de mémoire associée aux données *list-mode* et aux projections est toujours $L+P$. Cependant, étant donné les découpes variables du volume, il n'est pas possible de connaître à l'avance la consommation de mémoire associée aux volumes. Il est seulement possible d'en connaître la valeur moyenne, qui est aussi de $3\frac{V}{N}$.

2.5.5 En résumé

La première méthode de parallélisation proposée dans cette section permet de reconstruire des volumes très grands lorsque beaucoup de *GPU* sont utilisés. Cependant, cette approche peut poser des problèmes de temps d'itération irréguliers entre *GPU* et donc une mauvaise exploitation de la puissance de calcul. En effet, en fonction de la répartition de l'activité dans le champ de vue et de la sensibilité du scanner, la charge de travail peut être très irrégulièrement répartie entre les *GPU*. La seconde approche proposée dans cette section permet de corriger ce problème, mais implique aussi une moins bonne répartition de la charge mémoire. Cette approche devrait permettre de corriger le problème du temps de reconstruction, mais distribue moins équitablement la charge mémoire. Cependant, la répartition de la charge mémoire est moins critique que celle de la charge de travail parce qu'aujourd'hui les tailles des mémoires des *GPU* sont largement suffisantes pour stocker les volumes et les jeux de données nécessaires aux reconstructions. Un avantage de ces deux méthodes par rapport à la méthode fractionnant les sous-ensembles est que la quantité de données communiquées à chaque itération n'augmente pas si plus de sous-ensembles sont utilisés.

2.6 Étude d'évaluation

2.6.1 Jeux de données simulés

Les méthodes présentées ci-dessus ont été évaluées en utilisant un ensemble de données *list-mode*. Nous avons utilisé la plate-forme de SMC *Geant4 application for tomographic emission* (*GATE*) [Jan *et al.*, 2011] pour simuler de manière réaliste le scanner TEP Allegro/GEMINI de Philips, avec le modèle validé dans [Lamare *et al.*, 2006]. La simulation intègre les effets Compton et photoélectrique avec le modèle standard et la diffusion Rayleigh avec le modèle Livermore. Le jeu de données a été obtenu en simulant un fantôme NEMA IEC NU 2-2001 [NEMA, 2001], représenté dans la figure 2.16. Ce fantôme se compose d'un cylindre principal de 30 cm de diamètre, percé en son centre d'un trou cylindrique de 5 cm et rempli d'eau moyennement active. Dans ce cylindre sont placées sur le périmètre d'un cercle de 11,4 cm de diamètre, 6 sphères de 10, 13, 17, 22, 28 et 37 mm de diamètres. Les quatre sphères les plus petites sont remplies d'eau 8 fois plus active que celle placée dans le cylindre principal. Les deux sphères les plus grandes sont remplies d'eau non-active. Le mode de simulation *back-to-back* a été utilisé, c'est-à-dire que les photons d'annihilation sont parfaitement colinéaires et qu'ils sont émis là où est généré le positon. Cela permet d'accélérer la simulation en se passant de la simulation du parcours des positons. Seules les coïncidences vraies ont été enregistrées. Le jeu de données obtenu est un fichier *list-mode* de 12 millions de coïncidences.

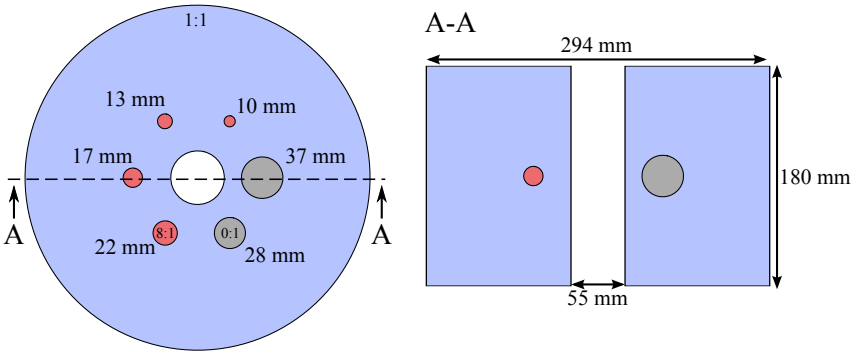


FIGURE 2.16 – Fantôme NEMA IEC NU-2 2001 composé d'un cylindre d'eau moyennement active creux avec un diamètre extérieur de 294 mm, un diamètre intérieur de 55 mm et 180 mm de longueur. À l'intérieur, sur un cercle de 57 mm dans le plan central du scanner, sont placées six sphères de 10, 13, 17, 22, 28, 37 mm. Les quartes plus petites contiennent de l'eau active avec un concentration huit fois plus élevée en traceur que l'eau du cylindre. Les deux plus grosses sphères contiennent de l'eau inactive.

2.6.2 Reconstructions

Une première reconstruction *LM-OSEM* avec 1 sous-ensemble, 100 itérations, des *voxels* de $1 \times 1 \times 1$ mm^3 et un champ de vue axial de 176 mm et $317 \times 317 mm^2$ transaxial, a été exécutée pour comparer les images reconstruites avec différentes parallélisations. La reconstruction non parallélisée a été exécutée sur 1 *GPU* et les reconstructions parallélisées sur 14 *GPU*. Les volumes de différences entre les reconstructions parallélisées et la reconstruction non parallélisée ont été calculés de la manière

suivante :

$$f_{diff_j} = \frac{|f_{parallélisée_j} - f_{non\ parallélisée_j}|}{f_{non\ parallélisée_j}} \tag{2.1}$$

où $f_{parallélisée_j}$ et $f_{non\ parallélisée_j}$ sont les valeurs du *voxel j* dans les volumes reconstruits respectivement avec et sans la parallélisation multi-*GPU*. La différence moyenne a été calculée pour chaque itération de la manière suivante :

$$\overline{f_{diff}} = \frac{\sum_{j=1}^{M} f_{différence_j}}{M} \tag{2.2}$$

où M est le nombre de *voxels* dans le volume.

L'étude des performances des différentes approches de parallélisation a été faite avec deux tailles de *voxel* différentes, $1 \times 1 \times 1$ mm^3 et $2 \times 2 \times 2$ mm^3 et trois nombres de sous-ensembles différents, 4, 8 et 16. Le champ de vue axial a été fixé à 176 mm et le champ de vue transaxial à $565 \times 565 mm^2$. Les reconstructions parallélisées ont été exécutées avec un nombre de *GPU* allant de 2 à 24. Chaque temps mesuré correspond au temps nécessaire pour effectuer une itération. Toutes ces mesures ont été répétées trois fois et moyennées.

2.6.3 Matériels

Les reconstructions ont été exécutées sur un *cluster* composé de trois machines hébergeant 10, 8, et 6 *GPU*, connectées ensemble en Gigabit Ethernet. Toutes les reconstructions utilisant au plus 10 *GPU* ont été exécutées sur la même machine possédant 10 *GPU*, afin de bénéficier de la bande passante plus important des connections PCI Express entre les *GPU*. Le modèle de carte graphique utilisé est la Nvidia GeForce GTX 690 qui intègre deux *GPU*, chacun possédant 1536 cœurs à 915 MHz et une mémoire de 2 gigaoctets. Les facteurs d'accélération des reconstructions sont mesurés en faisant le rapport du temps de la reconstruction mono-*GPU* sur le temps de la reconstruction multi-*GPU*.

2.7 Résultats

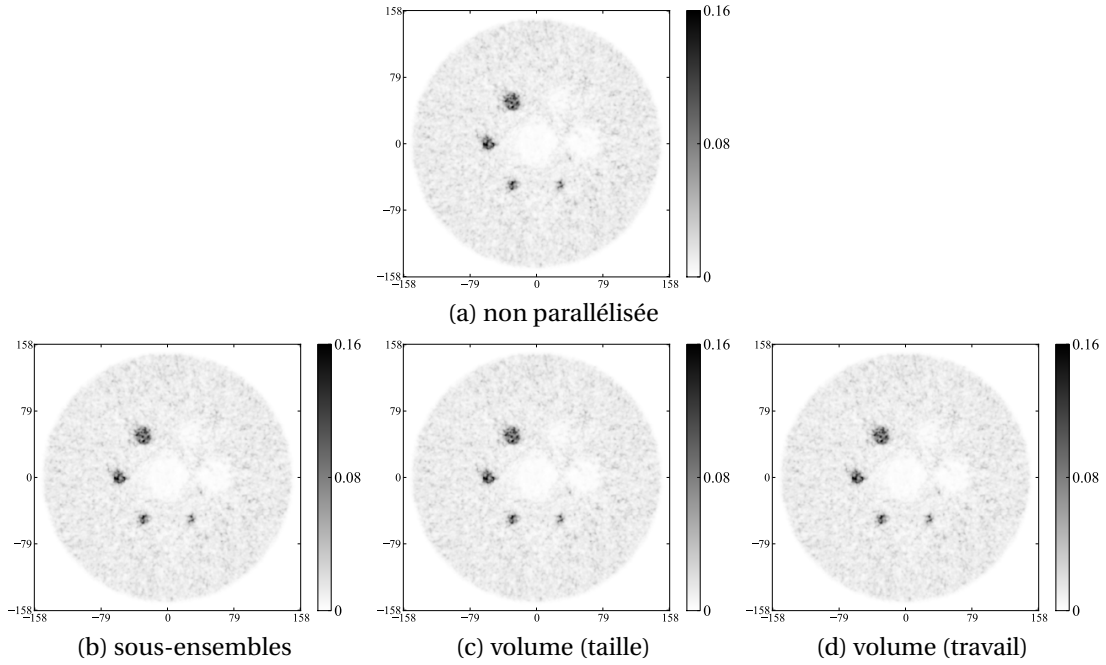
2.7.1 Erreurs de reconstructions dues à la parallélisation

Dans la figure 2.17 sont comparées les images reconstruites du fantôme NEMA IEC NU 2-2001 après 30 itérations, sans parallélisation et avec les trois méthodes de parallélisations. Visuellement, ces images sont parfaitement identiques.

En observant le profil horizontal passant par le milieu de ces images 2.18 on constate que les quatre courbes se superposent parfaitement.

Avec les images reconstruites et les profils, il n'est pas possible de distinguer de différence entre une reconstruction non parallélisée et une parallélisée. Si on observe dans la figure 2.19 les images de différences en pourcentage calculées à partir de l'équation 2.1, on constate quelques différences, mais elles ne dépassent pas plus de 0,1 % dans tous les cas pour cette coupe, et se concentrent principalement en dehors de l'objet, là où les valeurs d'activités sont proches de 0.

La figure 2.20 montre l'évolution de la différence moyenne sur tout le volume de reconstruction



(a) non parallélisée

(b) sous-ensembles (c) volume (taille) (d) volume (travail)

FIGURE 2.17 – Coupe transverse du fantôme NEMA IEC NU 2-2001 à l'itération 30 reconstruit sur un seul *GPU* (a), sur 14 *GPU* avec les parallélisations de sous-ensembles (b), volume avec morceaux de même taille (c) et volume avec la même charge de travail par morceau (d).

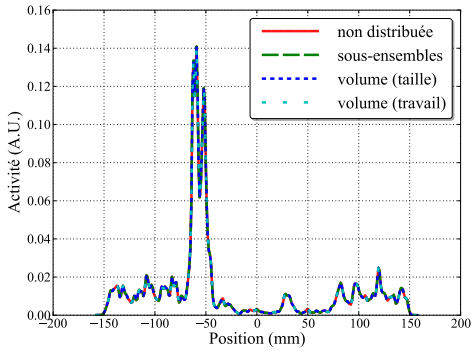
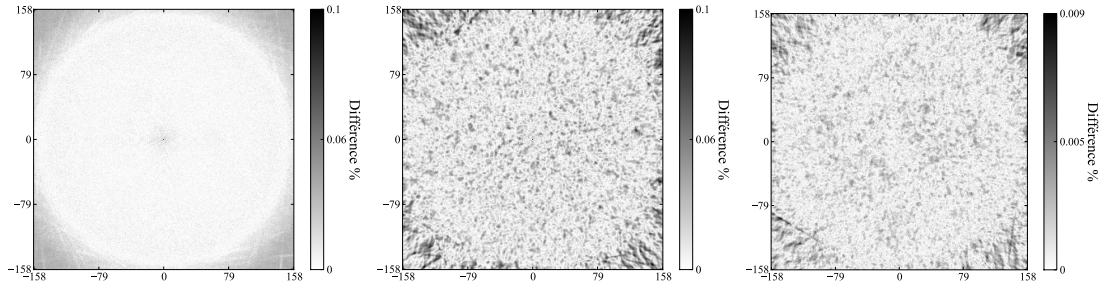


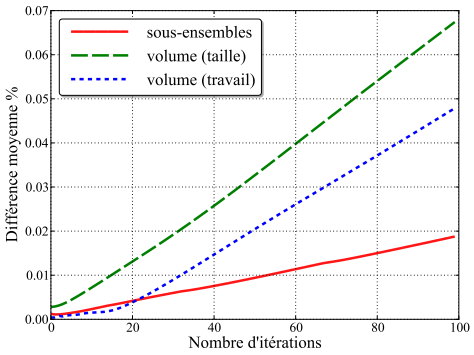
FIGURE 2.18 – Profils dans les images reconstruites avec et sans parallélisation à l'itération 30. Les profils se superposent parfaitement.



(a) sous-ensembles (b) volume (taille) (c) volume (travail)

FIGURE 2.19 – Image de la différence en pourcentage entre les reconstructions parallélisées et la reconstruction non parallélisée.

telle qu'elle est définie par l'équation 2.2. Cette différence moyenne ne cesse de croître, mais dans tous les cas, elle reste inférieure à 0,07 % pour la centième itération. Pour la dernière itération, la méthode de fractionnement de sous ensemble donne la différence la plus faible avec environ 0,02 %, la méthode de parallélisation volume(taille) donne la différence la plus élevée avec un peu moins de 0,07 % et la méthode de parallélisation volume(travail) donne une différence intermédiaire avec un peu moins de 0,05 %.



(a)

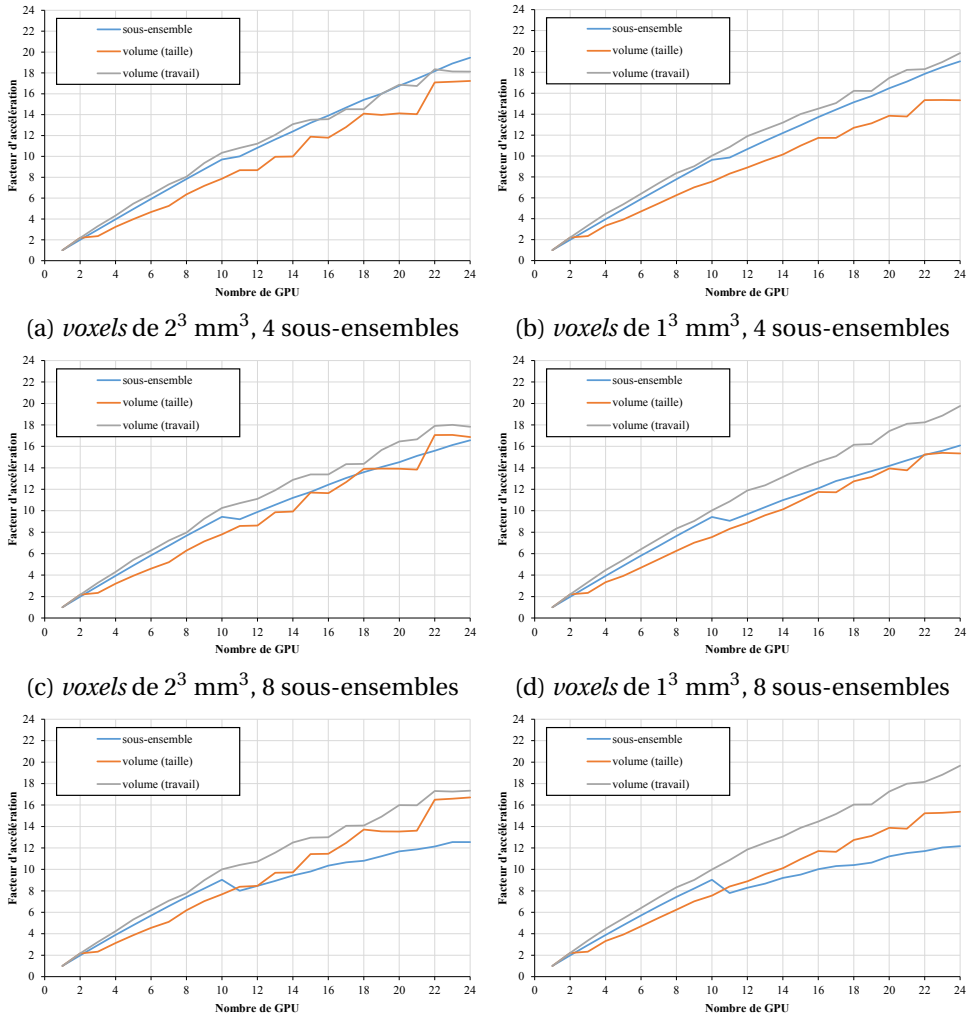
FIGURE 2.20 – Différence moyenne en pourcentage entre les reconstructions parallélisées et la reconstruction non parallélisée.

Les différences de reconstruction mesurées proviennent des erreurs lorsque l'on somme deux nombres codés en virgule flottante, c'est-à-dire lorsque l'on rassemble les volumes d'erreurs avec la méthode de fractionnement des sous-ensembles et lorsque l'on rassemble les projections avec les méthodes de fractionnement de volume. On ne peut pas qualifier ces différences d'erreurs, car même la reconstruction mono-*GPU* souffre aussi de ces erreurs de somme lorsqu'elle effectue les projections et rétroprojections. Ces différences restent très faibles relativement aux valeurs des *voxels* et peuvent donc être négligées.

2.7.2 Temps d'exécution

La figure 2.21 représente les facteurs d'accélération des différentes méthodes de parallélisation multi-*GPU* de la reconstruction *LM-OSEM* avec les deux tailles de *voxel* et les trois nombres de sous-ensembles différents. Dans tous les cas, la deuxième méthode de fractionnement du volume est celle qui apporte soit le meilleur facteur d'accélération soit un facteur d'accélération égale à la méthode de fractionnement des sous-ensembles. La première méthode de fractionnement des volumes fournit quant à elle le moins bon facteur d'accélération, sauf pour les nombres de sous-ensembles le plus élevé. La variation du nombre de sous-ensembles n'a pas d'impact sur la quantité d'information échangée pendant une itération avec les méthodes de parallélisation basées sur la découpe du volume, à la différence de la méthode de découpe des sous-ensembles. On constate particulièrement cette différence pour les nombres de *GPU* supérieurs à 10 car dans notre cas, certaines communications doivent se faire à travers la connexion Ethernet.

Si on observe le temps de communication pour chaque itération, présenté dans la figure 2.22, les méthodes utilisant la découpe du volume ont des temps de communication qui ne varient qu'en



(a) *voxels* de 2^3 mm^3, 4 sous-ensembles (b) *voxels* de 1^3 mm^3, 4 sous-ensembles

(c) *voxels* de 2^3 mm^3, 8 sous-ensembles (d) *voxels* de 1^3 mm^3, 8 sous-ensembles

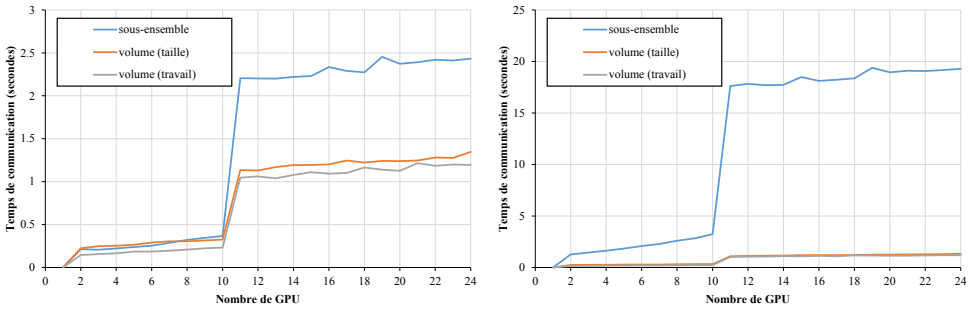
(e) *voxels* de 2^3 mm^3, 16 sous-ensembles (f) *voxels* de 1^3 mm^3, 16 sous-ensembles

FIGURE 2.21 – Facteur d'accélération des différentes méthodes de parallélisation avec 4, 8, et 16 sous-ensembles et des tailles de *voxels* de 2^3 mm^3 et 1^3 mm^3.

fonction du nombre de *GPU* et non pas de la taille des *voxels*, ni du nombre de sous-ensembles. En effet, les communications, pour ces deux méthodes, portent sur les projections et ne dépendent donc que du nombre de *LOR* dans le fichier *list-mode*. À l'inverse, la parallélisation basée sur les sous-ensembles a un coût de communication dépendant aussi du nombre de *GPU* mais aussi du nombre de sous-ensembles et du nombre de *voxels* (donc de leur taille à champ de vue fixe).

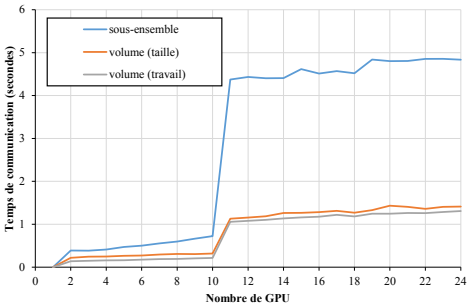
On peut alors constater, dans la figure 2.23 qui représente la fraction du temps de communication sur le temps total de reconstruction, que le temps de communication devient non négligeable avec la méthode de fractionnement des sous-ensembles. Avec des *voxels* de 1^3 mm^3 et 16 sous-ensembles, le temps de communication représente 80 % du temps de reconstruction sur 24 *GPU*, bloquant le facteur d'accélération à 12. Alors qu'avec la deuxième méthode de fractionnement du volume, le temps de communication représente moins de 2 % du temps de reconstruction, et le facteur d'accélération atteint 19,5.

Dans tous les cas, on constate que la méthode de parallélisation volume(taille) ne fournit pas de

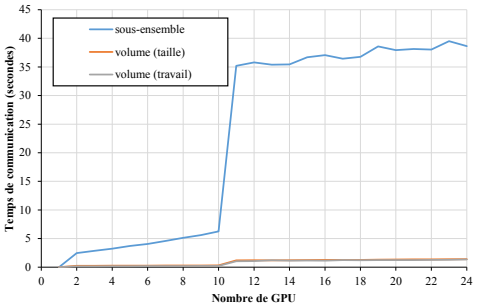


(a) *voxels* de 2^3 mm^3, 4 sous-ensembles

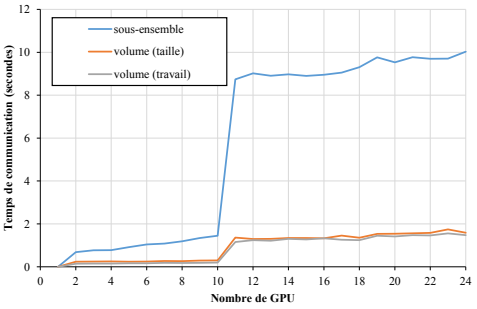
(b) *voxels* de 1^3 mm^3, 4 sous-ensembles



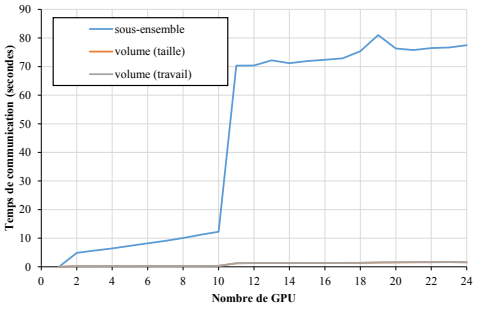
(c) *voxels* de 2^3 mm^3, 8 sous-ensembles



(d) *voxels* de 1^3 mm^3, 8 sous-ensembles



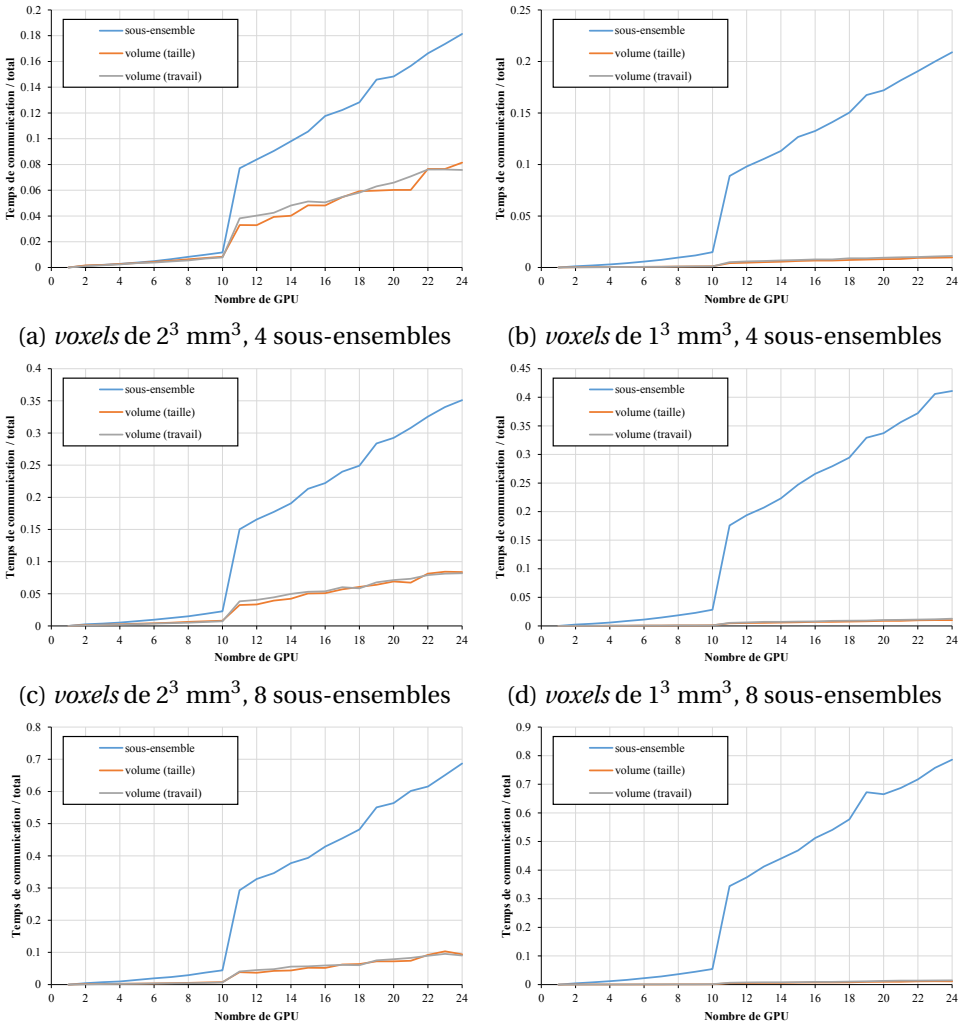
(e) *voxels* de 2^3 mm^3, 16 sous-ensembles



(f) *voxels* de 1^3 mm^3, 16 sous-ensembles

FIGURE 2.22 – Temps de communication par itération des différentes méthodes de parallélisation avec 4, 8, et 16 sous-ensembles et des tailles de *voxels* de 2^3 mm^3 et 1^3 mm^3.

facteurs d'accélérations aussi bons que les deux autres approches, de plus très variables en fonction du nombre de *GPU*. Comme nous l'avons vu précédemment, la charge de travail varie fortement dans le champ de vue, ce qui implique que pour certaines découpes des morceaux de sous-volumes sont associés à une charge de travail bien supérieure que d'autres et comme nous l'avons vu, la vitesse global de la reconstruction est contrainte par la vitesse du *GPU* le plus lent parce qu'au moment des communications tous les autres *GPU* doivent attendre le résultat de son travail. Nous pouvons voir dans le tableau 2.2 les variations de temps nécessaires pour exécuter une itération en excluant les temps de communication pour une reconstruction parallélisée sur 3 *GPU*. La méthode basée sur la découpe des sous-ensembles distribue très équitablement la charge de travail avec 0,15 % de différence entre le *GPU* le plus rapide et le plus lent. Par contre, la parallélisation volume(taille) donne une variation très importante de 89 %. Notre approche de répartition de la charge permet d'améliorer grandement ce déséquilibre avec une variation de seulement 2 %.



(a) *voxels* de 2^3 mm^3, 4 sous-ensembles

(b) *voxels* de 1^3 mm^3, 4 sous-ensembles

(c) *voxels* de 2^3 mm^3, 8 sous-ensembles

(d) *voxels* de 1^3 mm^3, 8 sous-ensembles

(e) *voxels* de 2^3 mm^3, 16 sous-ensembles

(f) *voxels* de 1^3 mm^3, 16 sous-ensembles

FIGURE 2.23 – Fraction du temps passé à communiquer sur le temps total pour les différentes méthodes de parallélisation avec 4, 8, et 16 sous-ensembles et des tailles de *voxels* de 2^3 mm^3 et 1^3 mm^3.

TABLEAU 2.2 – Temps d'exécution en secondes d'une itération d'une reconstruction sur 3 *GPU* avec des *voxels* de 1^3 mm^3 et 4 sous-ensembles. La première colonne indique le temps du *GPU* le plus rapide, la deuxième indique le temps du plus lent tandis que la dernière donne la différence en pourcentage entre ces deux temps.

Parallélisation	Minimum (secondes)	Maximum (secondes)	Différence ($\frac{max-min}{min}$)
sous-ensembles	634	635	0,15%
volume(taille)	429	814	89%
volume(travail)	551	562	2,0%

2.8 Discussion et conclusion

Dans ce chapitre, nous avons proposé deux méthode de parallélisation de la reconstruction en TEP qui se basent sur un fractionnement du volume de reconstruction. Ces méthodes ont été comparées à l'approche classique qui repose sur un fractionnement des sous-ensembles de données *list-*

mode. Dans le contexte étudié, la méthode de parallélisation permettant d'obtenir les meilleurs facteurs d'accélération est celle où le volume de reconstruction est fractionné en morceaux dont la taille est définie de telle sorte que la charge de travail soit la plus équilibrée possible entre le *GPU*. Avec cette méthode, nous avons réussi à atteindre des facteurs d'accélérations variant entre 18 et de 20 avec 24 *GPU*, tandis que la méthode classique oscillait entre 12 et 19 et la parallélisation volume(taille) entre 15 et 17. La méthode de parallélisation classique a l'avantage d'être simple et de répartir très efficacement la charge de travail sur tous les *GPU*, permettant de minimiser les temps d'attente. En effet, nous avons obtenu une variation de seulement 0,15% avec cette méthode alors que la première parallélisation par fractionnement de volume donnait une variation de 89% et 2 % avec la seconde méthode. Cependant, les coûts de communication associés à cette méthode deviennent très importants lorsque l'on augmente le nombre de sous-ensembles et la taille du champ de vue ou que l'on augmente le nombre de *voxels*, ce qui n'est pas le cas avec les deux méthodes proposées. La première méthode de parallélisation avec fractionnement du volume a l'inconvénient de très mal répartir la charge de travail, il est alors fréquent pendant la reconstruction que l'ensemble des *GPU* soit contraint d'attendre un seul *GPU* qui n'a pas encore terminé ses traitements. La deuxième méthode de parallélisation avec fractionnement du volume permet de corriger ce problème et garde l'avantage du faible temps de communication qui est lié à la taille du fichier *list-mode* et au nombre de *GPU*. Ces deux avantages sont particulièrement intéressants pour une reconstruction en une passe (grand nombre de sous-ensembles pour une seule itération), haute résolution avec un grand champ de vue.

Dans la suite de ce manuscrit, toutes les reconstructions multi-*GPU* se basent sur la méthode de parallélisation volume(travail).

Chapitre 3

Projecteur intégrant un modèle complet de la réponse du détecteur

3.1 Introduction **77**
3.2 Modélisation de la réponse du système **77**
3.2.1 Estimation directe des fonctions de réponse en coïncidence du détecteur . 78
3.2.1.1 Mesures empiriques 78
3.2.1.2 Simulation Monte-Carlo 78
3.2.1.3 Modèles analytiques 79
3.2.1.3.1 Modèles géométriques 79
3.2.1.3.2 Modélisation par des fonctions Gaussiennes 81
3.2.2 Estimation indirecte avec les fonctions de réponse intrinsèque du détecteur 83
3.2.2.1 Définitions 83
3.2.2.2 Estimation par convolution 86
3.2.2.3 Méthodes multiligne 87
3.2.3 En résumé 87
3.3 Projecteurs multiligne avec un modèle complet de la réponse du détecteur **88**
3.3.1 Estimation multiligne des *CDRF* 89
3.3.2 Estimation des $IDRF_{3D}$ 90
3.3.3 Modèles des $IDRF_{3D}$ et production des échantillons aléatoires 91
3.3.3.1 Utilisation brute des $IDRF_{3D}$ estimées 91
3.3.3.2 Modèle analytique des $IDRF_{3D}$ 92
3.4 Implémentations des projecteurs **95**
3.5 Étude d'évaluation **95**
3.5.1 Construction des modèles des projecteurs 95
3.5.1.1 Jeu de données 95
3.5.1.2 Projecteurs Gaussiens 96
3.5.1.3 Projecteurs *IRIS* 96
3.5.2 Reconstructions 97
3.5.3 Estimation du nombre minimal de lignes nécessaire aux projecteurs multiligne 97
3.5.4 Évaluation de la qualité d'image 100
3.5.5 Temps de reconstruction 102
3.5.6 Évaluation de la résolution 103
3.5.7 Évaluations avec des fantômes anthropomorphiques 104

3.5.7.1 Fantômes et jeux de données 104
3.5.7.2 Reconstructions et facteurs de mérite 105
3.5.7.3 Résultats 106
3.6 Discussion et conclusion **109**

3.1 Introduction

Nous avons vu, dans la section 1.5, que les algorithmes de reconstruction itératifs se basent sur une modélisation matricielle où le système est généralement décomposé en un ensemble d'effets, chacun modélisé par une matrice spécifique. Une décomposition commune, consiste à regrouper les effets intervenant dans le domaine des données (projections ou données *list-mode*) d'une part, et les effets intervenant dans le domaine image d'autre part. La transition entre ces deux domaines est dévouée à une troisième matrice, que l'on nomme matrice système, modélisant la réponse du détecteur avec les effets physiques et géométriques associés ainsi que la non-colinéarité des photons d'annihilation. Cette matrice système permet d'effectuer les deux étapes clés que sont la projection et la rétroprojection. En TEP, on appelle "projecteur" l'opérateur qui effectue les projections et rétroprojections.

La modélisation précise de l'ensemble des effets intervenant dans le système d'acquisition des données par le projecteur est primordiale lorsqu'il s'agit d'obtenir une reconstruction qui soit qualitative mais aussi quantitative. [Gifford *et al.*, 2000, Tong *et al.*, 2010] ont montré que l'utilisation d'un modèle précis de la réponse du détecteur dans le processus de reconstruction permet d'obtenir des images de meilleure qualité qu'avec un modèle approximatif ou une correction post-reconstruction des effets associés à la réponse du détecteur dans le domaine image.

3.2 Modélisation de la réponse du système

De très nombreuses méthodes ont été proposées afin d'intégrer un modèle de la réponse du système dans la reconstruction en TEP. La méthode la plus directe consiste à estimer à l'aide de mesures empiriques ou avec des SMC l'ensemble de la matrice système et de la stocker pour l'exploiter ensuite pendant la reconstruction. En pratique, cette approche brute est complexe à mettre en œuvre pour des champs de vue comptant un grand nombre de *voxels* et de *LOR* possibles, en raison de la taille de la matrice système qui est égale au produit de ces deux nombres. En TEP 2D ou 2D$\frac{1}{2}$ ce type de méthode a été utilisé, mais en TEP 3D les dimensions de la matrice système nécessitent généralement plusieurs dizaines de téraoctets d'espace mémoire, difficilement compatibles avec les solutions de stockage actuelles et incompatible avec une reconstruction rapide à cause des bandes passantes mémoire nécessaire pour lire rapidement un tel volume de données.

Pour répondre à ce problème, beaucoup d'approches ont été proposées, parmi lesquelles on peut distinguer deux stratégies. La première repose sur le stockage de la matrice système à l'aide de méthodes de compression, en ayant préestimée celle-ci auparavant. La deuxième stratégie repose sur une estimation à la volée des coefficients de la matrice système, pendant la reconstruction avec un opérateur, appelé projecteur. Toutes les méthodes utilisées pour préestimer la matrice système ne sont pas utilisables comme projecteur. En effet, elles peuvent nécessiter des mesures empiriques ou des SMC, incompatibles avec un calcul à la volée pendant le processus de reconstruction.

Dans cette section, nous ferons un bref état de l'art des différentes méthodes permettant d'estimer la matrice de la réponse du système.

3.2.1 Estimation directe des fonctions de réponse en coïncidence du détecteur

Pendant le processus de reconstruction, les opérations de projection et de rétroprojection n'ont pas besoin de l'intégralité de la matrice système pour chaque *LOR* traitée, mais seulement d'une ligne de cette matrice. Une ligne de cette matrice est associée à une paire de cristaux détecteurs et donne la probabilité en chaque *voxel* du champ de vue qu'une émission de positon dans ce *voxel* soit détectée selon cette paire de cristaux. Pour une paire de cristaux cette ligne de la matrice système est appelée la fonction de réponse en coïncidence du détecteur ou *coincidence detector response function* en anglais (*CDRF*) (aussi appelée *coincidence aperture function* dans [Lecomte *et al.*, 1984]), et donne la représentation physique de la *LOR* associée, comme l'illustre la figure 3.1.

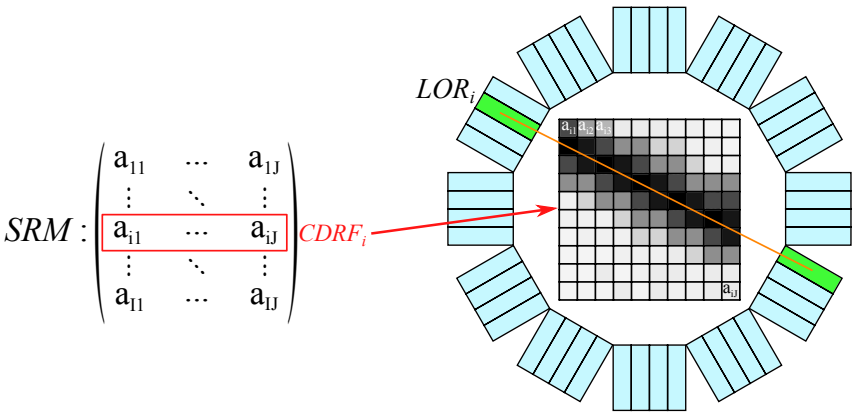


FIGURE 3.1 – Illustration du lien entre les *CDRF* et la matrice système. Une *CDRF* donne la représentation physique d'une *LOR*.

3.2.1.1 Mesures empiriques

Une méthode précise pour estimer les *CDRF* consiste à les mesurer directement. Cette approche, explorée de nombreuses fois [Selivanov *et al.*, 2000, Frese *et al.*, 2003, De Bernardi *et al.*, 2003, Lee *et al.*, 2004, Panin *et al.*, 2006, Sureau *et al.*, 2008, Alessio *et al.*, 2010, Yao *et al.*, 2012], a l'avantage de fournir une matrice système très réaliste qui intègre l'ensemble des effets physiques, géométriques et électroniques du détecteur, mais aussi les défauts intrinsèques au détecteur utilisé. L'inconvénient principal de ce type de méthodes vient de la mesure qui doit être effectuée à l'aide d'un dispositif automatisé déplaçant très précisément une source radioactive en différentes positions du champ de vue. Pour garantir une bonne qualité statistique de la matrice système mesurée, il est nécessaire de conserver la source en chaque position pendant un temps suffisant. Le temps total d'acquisition pour construire ce type de matrice est important, généralement de l'ordre d'une semaine [Panin *et al.*, 2006]).

3.2.1.2 Simulation Monte-Carlo

Une autre approche d'estimation de la matrice système repose sur l'utilisation de SMC [Veklerov *et al.*, 1988, Mumcuoglu *et al.*, 1996a, Qi *et al.*, 1998, Veklerov *et al.*, 1998, Rafecas *et al.*, 2004, Alessio *et al.*, 2006, Moreau *et al.*, 2014]. Comme avec une mesure empirique, l'ensemble des effets physiques

et géométriques sont modélisés précisément. La plate-forme logicielle GATE [Jan *et al.*, 2011] est couramment employée pour les SMC réalistes en imagerie nucléaire et radiothérapie. Ces approches nécessitent des temps de calcul importants, mais peuvent être accélérées en utilisant des méthodes de réduction de variance, des *clusters* de calcul ou des cartes graphiques [Bert *et al.*, 2013]. Avec les puissances de calcul disponibles actuellement, il est possible de construire une matrice système avec ce type d'approches en quelques jours lorsque l'on dispose d'un cluster de calcul possédant quelques dizaines de *CPU*.

Ce type d'estimation des *CDRF* pourrait être utilisé comme projecteur pendant la reconstruction. Cependant, les temps de simulation nécessaires pour une estimation peu bruités des *CDRF* sont aujourd'hui beaucoup trop important pour envisager des reconstructions dans des temps compatibles avec les contraintes en clinique.

3.2.1.3 Modèles analytiques

Une troisième méthode de construction de la matrice système s'appuie sur l'utilisation de modèles analytiques. Ce type d'approches a l'avantage d'être généralement moins coûteux en temps de calcul que les méthodes utilisant des SMC et elles ne nécessitent pas de mesures longues et précises avec des dispositifs spécifiques comme les méthodes basés sur des mesures empiriques. En revanche, peu de modèles analytiques intègrent une modélisation précise de l'ensemble des effets physiques et géométriques du détecteur du scanner.

3.2.1.3.1 Modèles géométriques

Un modèle commun des *CDRF* est de les approximer par de simple ligne. On distingue principalement deux types de projecteurs utilisant un modèle linéique, ceux traçant des lignes binaires, c'est-à-dire avec des coefficients de *CDRF* valant soit zéro soit un, et ceux traçant des lignes plus lisses avec des méthodes d'anticrénelage. Ces deux approches sont représentées dans la figure 3.2.

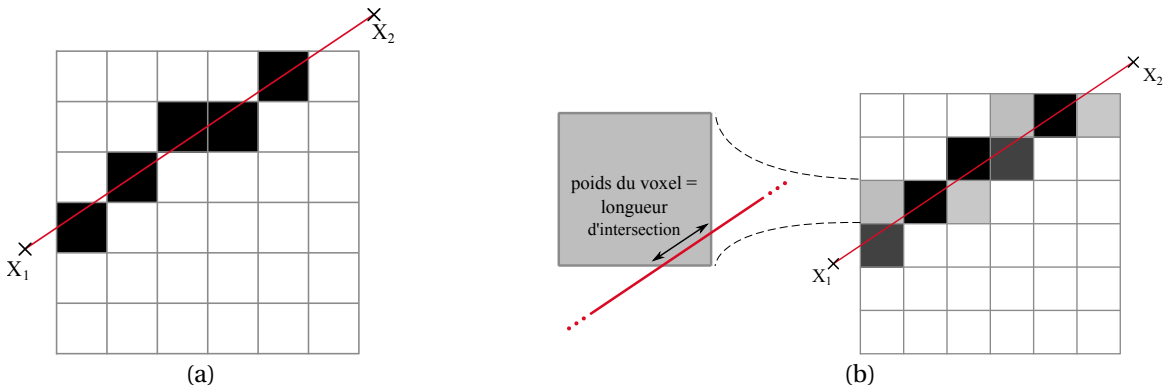


FIGURE 3.2 – Modèles linéiques de la *CDRF*. (a) montre un modèle linéique binaire de *CDRF* et (b) montre un modèle linéique donné par l'algorithme de Siddon, où la valeur de la *CDRF* en un *voxel* est égale à la longueur d'intersection de la *LOR* et du *voxel*.

Les lignes binaires peuvent être tracés avec des algorithmes de type [Bresenham, 1965] ou des algorithmes plus rapides comme le *Digital Differential Analyzer* (*DDA*) de [Mayorov, 1964] ou une

version optimisée comme celle proposée par [Bert et Visvikis, 2011]. L'algorithme de [Siddon, 1985] est la méthode la plus populaire lorsqu'il s'agit de tracer des lignes avec un anticrénelage. Avec cette méthode, le coefficient d'un *voxel* d'une *CDRF* est proportionnel à la longueur d'intersection de la *LOR* avec le *voxel*. On peut en trouver une version itérative plus rapide dans [Jacobs *et al.*, 1998]. Avec cet algorithme, la valeur en un *voxel* d'une *CDRF* est égale à la longueur de l'intersection de la *LOR* et du *voxel*.

Dans le contexte d'une matrice système stockée, les approches linéiques ont comme avantage principal de réduire le nombre d'éléments non nuls, ce qui permet de réduire l'espace mémoire nécessaire à son stockage en exploitant des méthodes de stockage de matrices creuses. L'avantage d'une matrice système plus creuse peut aussi résider en une accélération de la reconstruction grâce à la réduction du nombre de *voxels* à traiter pour chaque *LOR*. Cependant, ce modèle n'intègre aucun effet physique ou géométrique du système de détection. Pour remédier à ce problème, des méthodes les modélisent à l'aide de convolutions appliquées dans l'espace des projections, comme on peut le trouver dans [Rahmim *et al.*, 2008a, Zhou et Qi, 2011]. Cependant, les convolutions dans l'espace des projections ne peuvent pas être appliquées au format *list-mode* et ne permettent pas de modéliser les variations d'une *CDRF* le long de l'axe de la *LOR*.

Il y a encore quelques années, seuls les projecteurs linéiques permettaient d'estimer les *CDRF* à la volée suffisamment rapidement pour être compatible avec une application clinique. Cependant, ils ne modélisent aucun effet physique ou géométrique et ne permettent donc pas d'obtenir des reconstructions qualitatives et quantitatives. Ce type de projecteur a aussi l'inconvénient de fournir des images dont la qualité est fortement liée à la taille des *voxels*. En effet, ce type de projecteur trace des lignes dans une grille de *voxels* sans prendre en compte leurs dimensions physiques. La taille des *voxels* est en général fixée par des mesures de la *PSF* du détecteur.

Un moyen de contourner le problème de la variation de la largeur des *CDRF* calculées par le projecteur en fonction de la taille des *voxels*, repose sur l'utilisation d'un modèle volumique des *CDRF*. Le modèle volumique le plus simple est un tube homogène, qui peut être de section rectangulaire [Smith *et al.*, 2003] ou de section elliptique [Johnson *et al.*, 1995, Schretter, 2006, Lougovski *et al.*, 2014]. Dans les deux cas, la valeur en un *voxel* d'une *CDRF* est égale au volume d'intersection du *voxel* avec le tube, comme représenté dans la figure 3.3.

Communément, les dimensions des tubes sont fixées, soit par la taille des cristaux, soit par la *PSF* du détecteur. Ce type de projecteur permet de supprimer la dépendance de la *CDRF* à la taille des *voxels*, qu'ont les projecteurs linéiques. Comme les modèles linéiques, ces modèles volumiques ne permettent pas de modéliser précisément ni la géométrie ni la physique du détecteur.

Un dernier type de modèles géométriques utilise l'angle solide de vue des cristaux pour estimer les coefficients des *CDRF*, comme ceux proposés par [Chen *et al.*, 1991, Terstegge *et al.*, 1996]. Ces projecteurs permettent de modéliser la composante géométrique de la réponse du détecteur, mais pas la pénétration et la diffusion dans les cristaux.

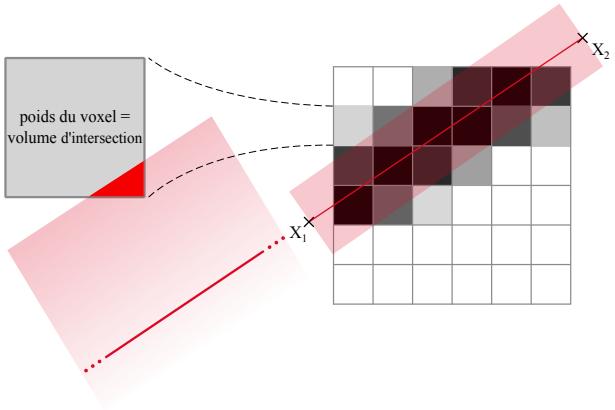
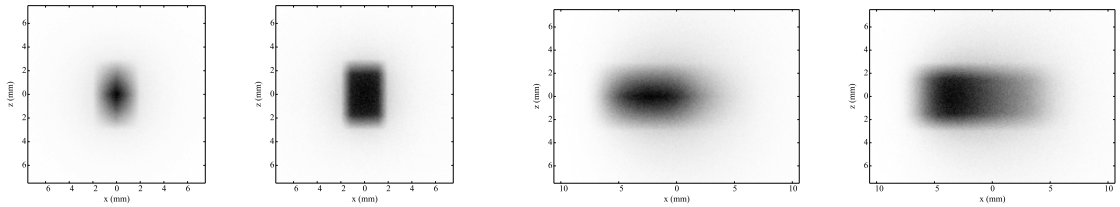


FIGURE 3.3 – Représentation bidimensionnelle de la modélisation de la *CDRF* par un cylindre. La valeur de la *CDRF* en un *voxel* est égale au volume d'intersection du volume cylindrique associé à la *LOR* avec le *voxel*.

3.2.1.3.2 Modélisation par des fonctions Gaussiennes

Le modèle précédent modélise les *CDRF* par des volumes binaires, ce qui est loin de représenter la réalité physique de cette fonction. Les effets de parallaxes et la diffusion dans les cristaux donnent une forme complexe à la *CDRF*, comme on peut le voir sur les mesures obtenues par SMC dans la figure 3.4. Dans le but de modéliser ces variations, certains projecteurs utilisent des fonctions comme la fonction Gaussienne.



(a) LOR centrale, coupe au centre de la LOR

(b) LOR centrale, coupe à 220mm du centre de la LOR

(c) LOR à 250 mm du centre du champ de vue, coupe au centre de la LOR

(d) LOR à 250 mm du centre du champ de vue, coupe à 110 mm du centre de la LOR

FIGURE 3.4 – Sections de *CDRF* mesurées par SMC.

Les *CDRF* varient pour différentes positions de la *LOR* dans le champ de vue, mais aussi le long de l'axe de la *LOR*, comme on peut le voir dans la figure 3.4. Plusieurs projecteurs modélisent les coupes transversales des *CDRF* par des distributions normales bivariées. Certains de ces projecteurs utilisent une distribution unique pour modéliser toutes les coupes de toutes les sections des *CDRF*, comme dans [Selivanov *et al.*, 2000, Pratx *et al.*, 2006, Cui *et al.*, 2010, Cui *et al.*, 2011c, Cui *et al.*, 2011a]. Les deux variances de ce modèle sont fixées à partir de mesures de la *PSF* moyenne du détecteur. Dans la suite de ce manuscrit, on appellera ce projecteur Gaussien$_{constant}$. D'autres projecteurs utilisent des distributions normales dont les paramètres varient, afin de modéliser les variations des *CDRF*, comme dans [Selivanov *et al.*, 2000, Pratx *et al.*, 2009]. Les *LOR* qui ne passent pas au centre du champ de vue ont une *CDRF* asymétrique, comme on peut le voir sur les figures 3.4c et 3.4d. Afin de modéliser cette asymétrie, certains projecteurs utilisent des distributions Gaussiennes asymétriques,

c'est-à-dire avec des variances différentes d'un côté et de l'autre du maximum de la distribution pour chacun des deux axes, qui varient dans le champ de vue. Ce type de projecteur est utilisé dans [Cui *et al.*, 2011b, Ha *et al.*, 2012, Ha *et al.*, 2013]. Dans ce manuscrit, nous nous baserons sur le projecteur décrit dans [Cui *et al.*, 2011b], que nous appellerons le projecteur Gaussien$_{variant}$. Avec ce projecteur, chaque coupe de *CDRF* est modélisée par quatre variances, deux pour chaque axe, et une pour chaque côté du centre de la distribution. Le modèle utilisé avec ce projecteur suppose que le système de détection est invariant par rotation autour de l'axe principal du détecteur, par translation suivant ce même axe et par rotation suivant l'axe perpendiculaire. Les quatre variances ne varient donc qu'en fonction de deux paramètres S et P, représentés dans la figure 3.5. Avec ce modèle, la valeur de la *CDRF* a_i associée à une *LOR* i donnée, est donnée par l'équation suivante :

$$a_i(x,z) = \begin{cases} c(s,p)\mathrm{e}^{-\frac{1}{2}\left(\frac{x-x_0}{\sigma_{x1}(s,p)}\right)^2-\frac{1}{2}\left(\frac{z-z_0}{\sigma_{z1}(s,p)}\right)^2} & ,si\ x <= x_0\ et\ z <= z_0 \\ c(s,p)\mathrm{e}^{-\frac{1}{2}\left(\frac{x-x_0}{\sigma_{x1}(s,p)}\right)^2-\frac{1}{2}\left(\frac{z-z_0}{\sigma_{z2}(s,p)}\right)^2} & ,si\ x <= x_0\ et\ z > z_0 \\ c(s,p)\mathrm{e}^{-\frac{1}{2}\left(\frac{x-x_0}{\sigma_{x2}(s,p)}\right)^2-\frac{1}{2}\left(\frac{z-z_0}{\sigma_{z1}(s,p)}\right)^2} & ,si\ x > x_0\ et\ z <= z_0 \\ c(s,p)\mathrm{e}^{-\frac{1}{2}\left(\frac{x-x_0}{\sigma_{x2}(s,p)}\right)^2-\frac{1}{2}\left(\frac{z-z_0}{\sigma_{z2}(s,p)}\right)^2} & ,si\ x > x_0\ et\ z > z_0 \end{cases} \tag{3.1}$$

où x et z sont les directions perpendiculaires à la *LOR* i, x étant la direction dans le plan perpendiculaire à l'axe principale du détecteur et z la direction perpendiculaire à x, x_0 et z_0 sont les positions du point d'intersection de la *LOR* avec le plan (x,z). $\sigma_{x1}(s,p)$, $\sigma_{x2}(s,p)$ sont les variances du modèle Gaussien, dépendantes de s et de p, suivant l'axe x pour les positions en x, respectivement inférieure à x_0 et supérieures à x_0. $\sigma_{z1}(s,p)$ et $\sigma_{z2}(s,p)$ sont les variances du modèles suivant l'axe z. $c(s,p)$ est un facteur de normalisation permettant à toutes les coupes d'avoir la même intégrale. Ce facteur est calculé avec l'équation suivante :

$$c = \frac{2}{\pi\left(\sigma_{x1}\sigma_{z1}+\sigma_{x1}\sigma_{z2}+\sigma_{x2}\sigma_{z1}+\sigma_{x2}\sigma_{z2}\right)} \tag{3.2}$$

Dans [Cui *et al.*, 2011b], les quatre variances décrivant ce modèle sont estimées à partir de reconstruction de sources ponctuelles placées à différents endroits du champ de vue, ce qui permet de construire des tableaux donnant les valeurs de ces variances pour toutes les positions (s,p) possibles. Toutefois, il est aussi possible de déterminer ces variances en différentes positions (s,p), en adaptant le modèle 3.1 sur des *CDRF* estimées par SMC. C'est cette seconde approche que nous retiendrons dans la suite de ce document.

Les modèles des *CDRF* utilisant des fonctions Gaussiennes ont principalement été développés pour être utilisé comme projecteur pendant la reconstruction et non pas pour préestimer celles-ci. La capacité d'un projecteur de ce type à modéliser toutes les composantes des *CDRF* tient dans la capacité de la fonction modèle (ici la Gaussienne) à pouvoir mimer la forme de la véritable *CDRF*. En pratique, on remarque que les queues de la distribution Gaussienne décroissent trop rapidement (kurtosis trop faible) pour modéliser la composante due à la diffusion intercristaux. Des méthodes utilisant d'autres fonctions pour modéliser la forme des *CDRF* ont aussi été proposées. Par exemple, [Taschereau *et al.*, 2011] proposent un modèle où chaque coupe transversale des *CDRF* sont construite en

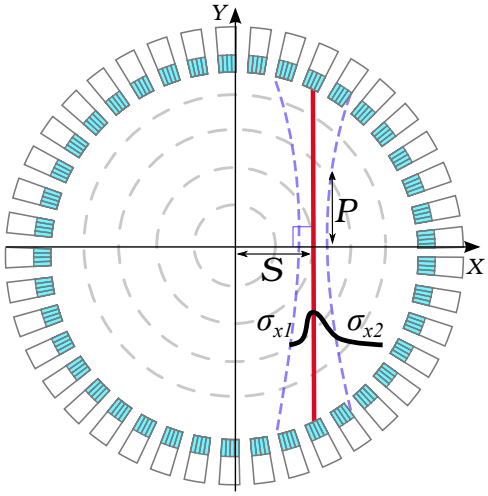


FIGURE 3.5 – Paramétrisation des positions le long de *LOR* avec le projecteur Gaussien$_{variant}$. Chaque *LOR* est caractérisée par sa position S qui est sa distance avec le centre du champ de vue du scanner. P désigne la position le long de l'axe de la *LOR* considérée.

convoluant une fonction rectangle (porte 2D) avec une fonction Gaussienne 2D. Les paramètres de ces fonctions sont optimisés pour faire correspondre le modèle à des coupes de *CDRF* mesurée par SMC. Cette méthode a été utilisée pour préestimer une matrice système stockée. Avec ce modèle, les coupes des *CDRF* modélise mieux la composante associée à la forme rectangulaire des cristaux, par rapport aux modèles purement Gaussiens. Cependant, la composante associée à la diffusion inter-cristaux n'est pas modélisée précisément pour les mêmes raisons qu'avec les projecteurs Gaussiens.

3.2.2 Estimation indirecte avec les fonctions de réponse intrinsèque du détecteur

3.2.2.1 Définitions

Nous venons de voir que plusieurs méthodes de modélisation de la réponse du détecteur tentent de décrire directement les *CDRF*. Cependant, d'autres méthodes utilisent plutôt les fonctions de réponse intrinsèques du détecteur ou *intrinsic detector response functions* en anglais (*IDRF*) (aussi appelées *intrinsic aperture functions* [Lecomte *et al.*, 1984]), pour ensuite calculer les *CDRF* à l'aide d'une relation qui les lie. Dans cette sous-section, nous allons voir comment se définisent ces *IDRF* et quel est leur lien avec les *CDRF*.

Dans [Lecomte *et al.*, 1984] ainsi que dans [Pratx et Levin, 2011], l'*IDRF* définit la probabilité de détecter un photon d'annihilation ayant comme angles d'incidences θ et φ et passant par un point de coordonnées (y', z') dans le plan perpendiculaire à la direction de propagation du photon qui passe par O le centre de la face du cristal. La définition de ce référentiel est représentée dans la figure 3.6. Avec cette définition, l'*IDRF* est une fonction de quatre variables, les angles θ et φ et les positions y' et z'. Avec la définition que nous venons de présenter, la *IDRF* pour un angle d'incidence (θ, φ) donné sera nommée $IDRF_{2D}$ dans le reste de ce manuscrit.

Pour une *LOR* donnée, la distance entre les deux cristaux détecteurs est généralement très grande relativement à la taille des cristaux. Pour cette raison, l'intervalle d'angle d'incidence des photons d'annihilation est très étroit, quelques degrés. [Lecomte *et al.*, 1984] font alors l'approximation que

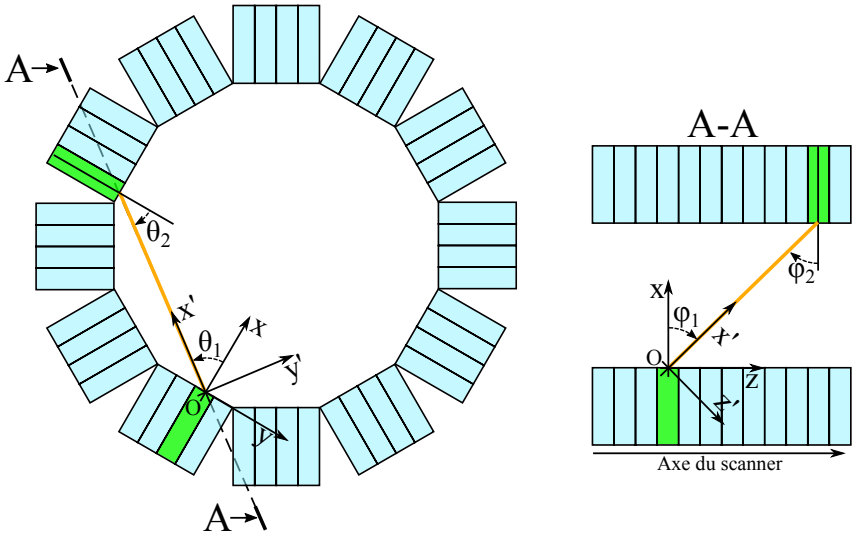


FIGURE 3.6 – Paramétrisation des angles d'incidence pour le calcul des *IDRF*. L'angle d'incidence θ est défini dans le plan (xOy), relativement à la direction x, sachant que x est la direction parallèle à l'axe du cristal et y la direction perpendiculaire à x et à z, l'axe du scanner. L'angle φ est, quant à lui, défini dans le plan (xOz), relativement à la direction x. x', y' et z' sont obtenus par rotation des directions x, y et z d'un angle θ autour de l'axe z puis φ autour de l'axe y.

pour une paire de cristaux donnée tous les photons arrivent avec le même angle d'incidence, si on considère un seul des deux cristaux. Cela permet d'associer à chaque *CDRF* deux $IDRF_{2D}$, une pour chaque cristal. Chacune de ces $IDRF_{2D}$ peut être estimée à l'aide d'une source collimatée balayant le plan (y', z') avec un angle d'incidence (θ, φ), et en mesurant le rapport du nombre de photons détectés sur le nombre de photons émis. La figure 3.7 donne un exemple d'une *IDRF* mesurée.

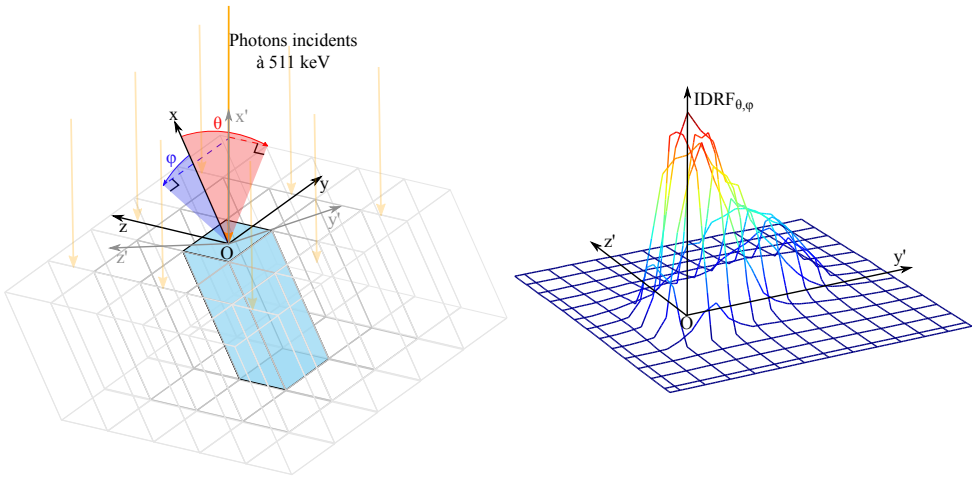


FIGURE 3.7 – Exemple d'une *IDRF* mesurée avec la définition de [Lecomte *et al.*, 1984].

[Pratx et Levin, 2011] proposent la relation suivante entre une *CDRF* a_i de la *LOR* i, et ses deux

IDRF, $\mathrm{IDRF}_{\theta_1\varphi_1}$ et $\mathrm{IDRF}_{\theta_2\varphi_2}$:

$$a_i(x',y',z') = \int_{-\pi/2}^{\pi/2}\int_{-\pi/2}^{\pi/2} \mathrm{IDRF}_{\theta_1\varphi_1}\left(y'+x'\tan\theta, z'+x'\tan\varphi\right) \\ \mathrm{IDRF}_{\theta_2\varphi_2}\left(y'-(\mathrm{D}-x')\tan\theta, z'-(\mathrm{D}-x')\tan\varphi\right)\mathrm{d}\theta\mathrm{d}\varphi \tag{3.3}$$

où D est la distance séparant les deux cristaux.

L'$IDRF_{2D}$ n'est pas la seule définition de la *IDRF* que l'on peut trouver dans la littérature. Il existe une définition alternative, que nous nommerons ici l'$IDRF_{3D}$. Celle-ci part de la constatation que pour n'importe quelle coïncidence vraie détectée, la position d'annihilation est placée sur une droite qui passe aussi par la position de la première diffusion dans le détecteur de chacun des deux photons d'annihilation. Ceci n'est évidemment vrai que si on néglige la non-colinéarité des photons d'annihilation. Si on note X la position de l'annihilation et X_1 et X_2 les positions des premières diffusions, celles-ci sont liées par l'équation suivante :

$$\mathrm{X} = \frac{1}{2}(1-t)\mathrm{X}_1 + \frac{1}{2}(1+t)\mathrm{X}_2 \tag{3.4}$$

où t est un nombre compris entre -1 et 1. Les densités de probabilité des points X_1 est X_2 définissent les $IDRF_{3D}$ des deux cristaux. Cette relation est représentée dans la figure 3.8.

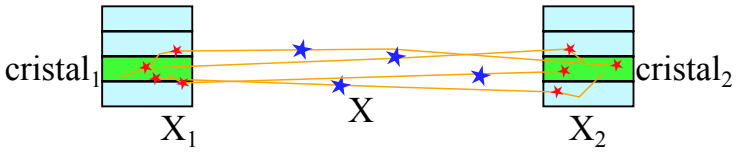


FIGURE 3.8 – Définition alternative de l'*IDRF* de [Gonzalez *et al.*, 2011]. Les points X sont les positions des annihilations détectées par les cristaux 1 et 2 dont la densité de probabilité est la *CDRF*. Les points Y_1 et Y_2 sont les positions de la première diffusion dans le détecteur des photons détectés, dont les densités de probabilité sont les *IDRF* des cristaux 1 et 2.

Pour une *LOR* i donnée, la densité de probabilité de X est liée à la *CDRF*. En effet, l'intégrale de la densité de X dans le voxel j donne la probabilité $\mathrm{P}(j|i)$ qu'une détection dans la *LOR* i soit causée par une émission dans j. Un élément de la *CDRF* donne la probabilité $\mathrm{P}(i|j)$ qu'une émission dans le *voxel* j soit détectée dans la *LOR* i. Le théorème de Bayes fournit la relation suivante :

$$\mathrm{P}(i|j) = \frac{\mathrm{P}(j|i)\mathrm{P}(i)}{\mathrm{P}(j)} \tag{3.5}$$

où $\mathrm{P}(j)$ est la probabilité qu'une émission se produise en j, qui est une constante valant $\frac{1}{\mathrm{N}}$ avec N le nombre de *voxels* dans le champ de vue, étant donné qu'on considère que chaque *voxel* possède la même probabilité d'émission. $\mathrm{P}(\mathrm{L}i)$ est la probabilité de détecter une émission suivant la *LOR* i, c'est-à-dire la sensibilité S_i de cette *LOR*. Ce qui donne :

$$a_{i,j} = \mathrm{P}(j|i)\mathrm{S}_i\mathrm{N} \tag{3.6}$$

où $a_{i,j}$ est le coefficient de la *CDRF* de la *LOR* i associé au *voxel* j.

Avec cette définition, à chaque *CDRF* est associée une paire de $IDRF_{3D}$, ce qui implique qu'il y a deux fois plus de $IDRF_{3D}$ que de *CDRF*. Cependant, comme pour l'$IDRF_{2D}$, on peut supposer que pour un cristal d'une paire donnée, tous les photons qui y sont détectés et qui appartiennent à des coïncidences vraies détectées par ces cristaux, ont le même angle d'incidence dans le cristal. Avec cette approximation, l'$IDRF_{2D}$ est la projection parallèle suivant l'angle d'incidence de la *LOR* de l'$IDRF_{3D}$. Cela permet de n'avoir qu'une $IDRF_{3D}$ par angle d'incidence possible, à la place de deux $IDRF_{3D}$ par *CDRF*, ce qui réduit drastiquement leur nombre, en fonction de l'échantillonnage des angles d'incidence θ et φ. L'$IDRF_{3D}$ est une fonction des trois positions spatiales (x', y', z') dans le repère du cristal, qui est donnée pour un angle d'incidence (θ, φ) de la *LOR* dans le cristal, comme défini dans la figure 3.6.

En partant des définitions des *IDRF* et de leurs relations avec les *CDRF* (equation 3.4), on peut imaginer pouvoir dériver une expression analytique des *CDRF* à partir d'une expression analytique des *IDRF*. Cette approche a été examinée par [Gonzalez *et al.*, 2011] avec des $IDRF_{3D}$ modélisées par des fonctions portes ou Gaussiennes. Cependant, des solutions ont été trouvées seulement pour un cas particulier, c'est-à-dire lorsque les deux cristaux se font face et sont alignés. Cette approche n'a pour l'instant pas pu être exploitée dans une reconstruction parce qu'aucune solution complète n'a été trouvée.

3.2.2.2 Estimation par convolution

Le modèle proposé dans [Lecomte *et al.*, 1984] calcule chaque *CDRF* en effectuant la convolution dans l'espace image des deux $IDRF_{2D}$ des cristaux associés. Dans cette étude, les $IDRF_{2D}$ ont été estimées pour plusieurs angles d'incidences par des SMC ou analytiquement en utilisant les coefficients d'atténuation linéaire du matériau composant les cristaux. Les $IDRF_{2D}$ obtenues par SMC modélisent l'ensemble de la réponse du détecteur, dont les effets géométriques et la diffusion dans les cristaux. Toutefois, cette approche n'a pas été exploitée pour modéliser la réponse du détecteur dans la reconstruction mais pour estimer de manière théorique la résolution spatiale associée à certains agencement géométriques des cristaux dans le scanner.

Une approche, relativement similaire a été proposée par [Yamaya *et al.*, 2005]. Ce projecteur utilise des convolutions des $IDRF_{2D}$, calculées analytiquement à partir de l'atténuation linéaire des cristaux, pour estimer les *CDRF*. Ce modèle de $IDRF_{2D}$ ne modélise pas la diffusion dans les cristaux. À la différence du modèle de [Lecomte *et al.*, 1984], la *CDRF* est supposée invariante dans la direction longitudinale de la *LOR*. Les *CDRF* ainsi calculées sont stockées et échantillonnées pendant la reconstruction pour générer les *CDRF* à la volée. Les $IDRF_{2D}$ estimée intègrent l'ensemble des effets intra et intercristaux. Cependant, l'invariance de la *CDRF* le long de la *LOR* ne permet pas une modélisation précise de la réponse du détecteur partout dans le champ de vue.

À partir de l'équation 3.4 on peut envisager de calculer une *CDRF* par convolution des $IDRF_{3D}$. Cette approche n'a pas été explorée dans la littérature.

3.2.2.3 Méthodes multiligne

Un projecteur décrit dans [Chen et Glick, 2007] propose d'estimer les *CDRF* en accumulant des lignes dont les positions des points de départ et d'arrivée varient aléatoirement avec un modèle d'$IDRF_{3D}$. Ce dernier néglige les interactions des photons dans les cristaux voisins du cristal détecteur et est invariant par rapport à l'angle d'incidence de la *LOR*. Toutes les $IDRF_{3D}$ sont modélisées par une distribution qui est uniforme à l'intérieur du cristal dans le plan perpendiculaire à son axe et exponentielle décroissante le long de son axe. Le seul paramètre ce ce modèle est celui de la décroissance exponentielle, qui est estimé par SMC pour des photons dont l'angle d'incidence est nul. Aucune distribution n'est modélisée dans les cristaux voisins, ce projecteur ne peut donc pas modéliser les effets intercristaux. Dans la suite de ce manuscrit, nous désignerons cette méthode comme le projecteur de Chen.

Un autre modèle, présenté dans [Moehrs *et al.*, 2008], approxime la *CDRF* par l'accumulation de plusieurs lignes pondérées par des coefficients donnés par un modèle analytique de *IDRF* qui ne modélise que l'atténuation linéaire des cristaux. Cette approche ne modélise pas la diffusion intercristaux.

[Stute *et al.*, 2011] proposent un modèle des $IDRF_{3D}$ où les effets intracristaux et intercristaux sont séparés. La composante intracristal est modélisée par une fonction 3D qui ne fournit que la composante de la $IDRF_{3D}$ qui se trouve à l'intérieur du cristal considéré. La composante intercristaux est modélisée par un tableau donnant pour tous les cristaux dans le voisinage du cristal considéré, la probabilité d'intéraction du photon avec le cristal. L'$IDRF_{3D}$ complète d'un cristal, c'est-à-dire avec les effets intra et intrecristaux, est supposée être donnée en répétant la fonction 3D modélisant les effets intracristal sur tous les cristaux du voisinage, en la pondérant à chaque fois par la valeur correspondante du tableau modélisant les effets intercristaux. L'estimation d'une *CDRF* repose sur le tracé d'une multitude de lignes avec l'algorithme de Siddon, chacune reliant deux échantillons des $IDRF_{3D}$ complètes des deux cristaux et pondérée par le produit des valeurs de ces deux échantillons. Avec cette approche, un seul modèle des effets intracristal est estimé par SMC, pour tous les cristaux et est échantillonné irrégulièrement, c'est-à-dire plus finement sur la face frontale du cristal que sur sa face extérieure. Le tableaux fournissant les interactions intercristaux sont estimés par SMC pour tous les cristaux d'un secteur de cristaux. Cette méthode estime l'intégralité de la réponse du détecteur, mais le modèle des *IDRF* ne varie pas en fonction de l'angle d'incidence des photons dans les cristaux, ce qui ne permet pas de prendre en compte précisément toutes ses composantes. De plus, cette méthode a été utilisée pour préestimer les *CDRF* et non pour estimer à la volée pendant la reconstruction, parce que chaque *CDRF* était construite en accumulant 56 millions de lignes, ce qui nécessite des temps de calcul importants.

3.2.3 En résumé

Actuellement, il est possible d'estimer précisément la matrice système avec les méthodes de SMC qui simulent de manière réaliste l'ensemble du détecteur, ce qui n'était pas envisageable il y a quelques dizaines d'années. Cependant, le stockage de la matrice système reste encore un problème majeur avec les mémoires actuelles, tant en matière de capacité que de bande passante. Ce problème est

d'autant plus important avec le développement des reconstructions *fully 3D*, *list-mode* et des corrections qui augmentent encore la taille de la matrice système. Dans ce contexte, beaucoup de méthodes de compression ont été proposées. La majorité de celles-ci exploite les symétries du détecteur et l'aspect creux de la matrice système, dont on peut trouver des exemples dans [Lazaro *et al.*, 2005, Matthieu, 2014], mais il existe d'autres approches comme celle de [Yao *et al.*, 2012] où seules quelques *CDRF* sont stockées et organisées par rapport aux angles d'incidence des *LOR* dans les cristaux. On trouve aussi des approches maximisant le caractère creux de la matrice système en simplifiant son modèle, ce qui permet de réduire l'espace mémoire nécessaire à son stockage, comme dans [Zhou et Qi, 2011], au prix d'une modélisation moins précise du système. Une autre problématique liée au stockage d'une matrice système précalculée est le temps de reconstruction qui peut être important. En effet, même compressée, il est rare que la matrice système soit suffisamment compacte pour être stockée intégralement en mémoire vive, ce qui implique de nombreuses lectures sur le disque dur dont les bandes passantes sont bien plus faibles. Un autre facteur de ralentissement provient de la décompression de la matrice système qui peut demander des puissances de calculs importantes. La faible flexibilité est une autre limite des méthodes basées sur une matrice préestimée. En effet, une fois celle-ci construite, il n'est plus possible de modifier la taille des *voxels* ni le champ de vue. De plus, à moins de construire plusieurs matrices systèmes, il n'est pas possible d'intégrer certaines informations comme la *DOI* et l'énergie d'interaction.

L'utilisation de projecteurs pour calculer à la volée les *CDRF* pendant la reconstruction permet de corriger l'ensemble de ces problèmes de flexibilité, de stockage et de temps de calcul. Toutefois, la calcul à la volée est incompatible avec certaines méthodes d'estimation trop couteuses en temps de calcul, comme par exemple une SMC. Les modèles linéiques des *CDRF* permettent une estimation très rapide mais aussi très approximative, en ne modélisant aucun des effets physiques et géométriques associés au détecteur. Les modèles analytiques basés sur des fonctions Gaussiennes permettent une estimation moins rapide, mais permettent de modéliser les effets associés à la géométrie du détecteur. Cependant, les effets intercristaux ne sont pas modélisés précisément avec ces projecteurs. Le projecteur multiligne de Chen utilise un modèle des *IDRF* pour estimer les *CDRF*. L'utilisation de simples lignes permet un calcul rapide des *CDRF*, mais le modèle des *IDRF* utilisé n'intègre pas de modèle des effets intercristaux, ni la variation de la *IDRF* en fonction des angles d'incidence des photons. Finalement, aucun projecteur n'intègre un modèle complet et précis des effets physiques et géométriques associés au détecteur du scanner. Dans cette thèse, nous proposons un projecteur multiligne basé sur une modélisation complète des $IDRF_{3D}$, intégrant les effets intracristal et intrecritaux, ainsi que les variations en fonction de l'angle d'incidence de la *lor* dans le cristal.

3.3 Projecteurs multiligne avec un modèle complet de la réponse du détecteur

Dans cette section, nous proposons un nouveau projecteur multiligne permettant d'estimer pendant la reconstruction les *CDRF* en modélisant l'ensemble des composantes physiques et géométriques du détecteur. Les positions des lignes accumulées pour construire une *CDRF* sont générées

aléatoirement avec des modèles des $IDRF_{3D}$. Ce projecteur, appelé *iterative random IDRF sampling* (*IRIS*), à été décliné en deux versions. L'une appelée $IRIS_{mesure}$, utilise des $IDRF_{3D}$ échantillonnées qui ont été estimées par SMC, l'autre appelée $IRIS_{analytique}$, modélise les $IDRF_{3D}$ avec un modèle analytique dont les paramètres sont dérivés des mesures effectuées par SMC.

3.3.1 Estimation multiligne des *CDRF*

Avec la définition de l'$IDRF_{3D}$ donnée dans l'équation 3.4, trouver une expression analytique pour les *CDRF* à partir de celle des $IRDF_{3D}$ n'est pas triviale, particulièrement avec des modèles des réalistes et pour tous les angles d'incidences possibles.

La relation 3.4 donne une relation entre des échantillons aléatoires de Y_1, Y_2, t et X. Avec cette relation, il est possible de générer des échantillons de X à partir d'échantillons de Y_1, Y_2 et t. La *CDRF* étant liée à la densité de probabilité de X par l'équation 3.6, il est possible de l'estimer en construisant l'histogramme 3D d'un échantillon suffisamment grand de X et en connaissant la sensibilité de la *LOR* considérée. Le projecteur *IRIS*, se base sur cette observation. De plus, avec ce projecteur nous exploitons le fait que la variable aléatoire t est équirépartie dans l'intervalle $]-1,1[$, ce qui permet d'incrémenter tous les *voxels* de l'histogramme 3D qui se trouvent le long des droites reliant les échantillons Y_1 et Y_2, et donc d'estimer la densité de probabilité de X avec moins d'échantillons de Y_1 et Y_2. L'estimation de la *CDRF* a_i d'une *LOR* i donnée à la position $X = (x, y, z)$, à partir de deux ensembles de N points Y_1 et Y_2 générés avec les modèles des deux $IDRF_{3D}$ associées à cette *LOR*, peut être formulée de la manière suivante :

$$a_i(X) = \frac{1}{N}\sum_{i=1}^{N}\int_{-1}^{1}\delta\left(X - \frac{1}{2}(1-t)Y_{1i} + \frac{1}{2}(1+t)Y_{2i}\right)dt \tag{3.7}$$

où δ est la distribution de Dirac. En pratique, l'estimation d'une *CDRF* se fait en générant des échantillons de points Y_1 et Y_2 avec les modèles des $IDRF_{3D}$, puis en traçant toutes les lignes qui les connectent. Le résultat de l'accumulation de toutes ces lignes fournit l'estimation de la *CDRF*. Ce principe est représenté dans la figure 3.9.

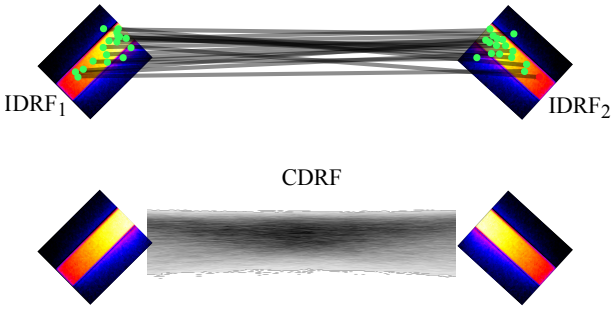
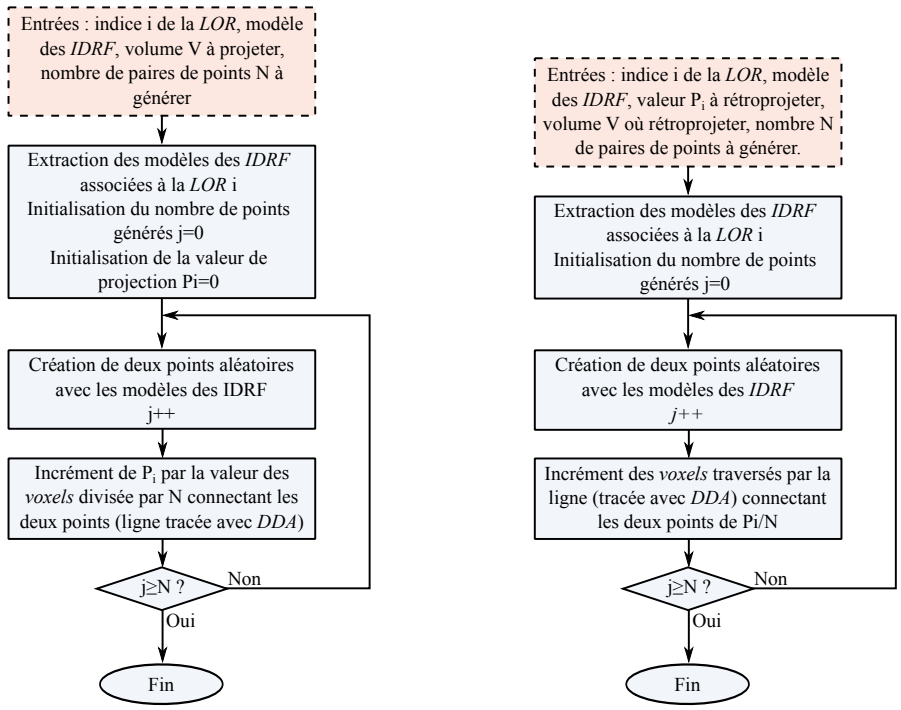


FIGURE 3.9 – Principe d'estimation de la *CDRF* avec le projecteur *IRIS*. Les deux $IDRF_{3D}$ sont utilisées pour générer des échantillons de points. Chaque point d'une $IDRF_{3D}$ est appairé avec un point généré avec l'autre $IDRF_{3D}$. L'estimation de la *CDRF* résulte de l'accumulation des droites connectant les paires de points générés avec les $IDRF_{3D}$.

Le calcul de la projection ou de la rétroprojection de la valeur P_i avec la méthode *IRIS* se décom-

pose de la manière suivante. Dans un premier temps, les modèles des $IDRF_{3D}$ sont sélectionnes en fonction de la *LOR i* traitée et le nombre de lignes N accumulées pour construire la *CDRF* est fixé. Ensuite, deux points sont générés avec les modèles des $IDRF_{3D}$. La ligne connectant ces deux points est construite avec l'algorithme *DDA* [Bert et Visvikis, 2011] dans le champ de vue voxélisé. Pour la projection, la valeur P_i est incrémentée de la somme de la valeur des *voxels* de la ligne divisée par N. Pour la rétroprojection, les *voxels* de la ligne sont incrémentés de $\frac{P_i}{N}$. Ensuite, une nouvelle paire de points aléatoire est générée et une nouvelle ligne est tracée et ainsi de suite N fois. Ces deux procédures sont schématisées par des diagrammes dans la figure 3.10.



(a) Projection (b) Rétroprojection

FIGURE 3.10 – Diagrammes de (a) la projection et de (b) la rétroprojection avec les projecteurs *IRIS*.

3.3.2 Estimation des $IDRF_{3D}$

Les $IDRF_{3D}$ estimées sont des volumes voxélisés placés dans le repère du cristal, c'est-à-dire dans le repère (x', y', z') défini dans la figure 3.6. Elles contiennent la distribution des positions de la première interaction des photons détectés et ayant un angle d'incidence dans un certain intervalle (θ, φ). Ces angles sont échantillonnés régulièrement. Nous avons vu dans la section 1.3.1 que les détecteurs des scanners TEP ne permettent pas de mesurer séparément chaque interaction des photons détectés, ce qui rend impossible une mesure expérimentale de l'$IDRF_{3D}$, à la différence de l'$IDRF_{2D}$. Pour estimer ces distributions, nous avons utilisé des SMC avec *GATE* [Jan *et al.*, 2011], qui permettent de simuler le détecteur de manière réaliste et d'avoir accès à l'historique complet des interactions des photons détectés. Nous avons simulé un modèle du scanner avec une source remplissant l'intégralité du champ de vue, avec les sorties des *hits* (les interactions des photons dans le détecteur) et des coïncidences vraies. Parmi les *hits*, sont conservés seulement ceux qui correspondent à la première

interaction de chacun des photons de la liste de coïncidences. Pour chacun d'eux, l'angle d'incidence (θ, φ) de la *LOR* est calculé et, dans la $IDRF_{3D}$ correspondante, le *voxel* où se situe le *hit* est incrémenté d'un. Une fois toutes les données de simulation traitées, chaque $IDRF_{3D}$ est normalisée de telle sorte que la somme de ses *voxels* vaille un. Une représentation du résultat de cette estimation est donnée dans la figure 3.11.

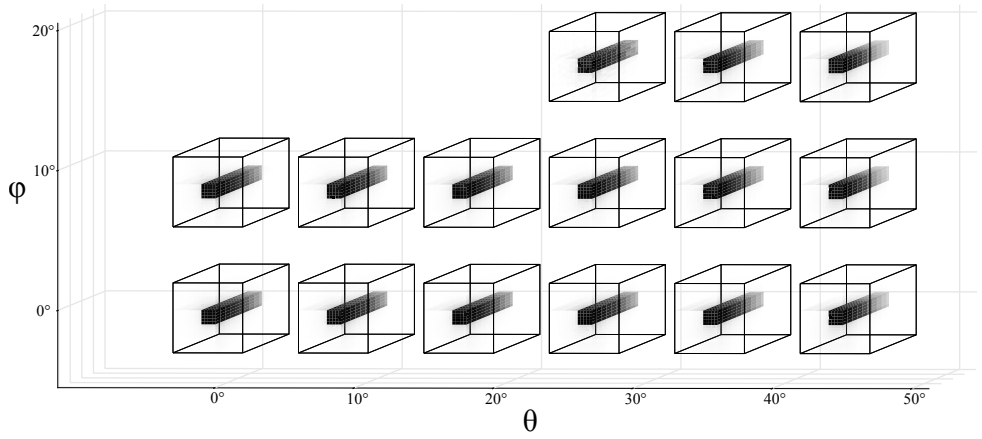


FIGURE 3.11 – Exemple d'un ensemble de $IDRF_{3D}$ estimées par SMC, obtenu avec un modèle du scanner TEP GEMINI de Philips. Chaque pavé contient un volume qui est une estimation d'une $IDRF_{3D}$. Pour certains angles, aucune $IDRF_{3D}$ n'apparaît parce qu'aucune *LOR* n'est détectée avec de tels angles d'incidence.

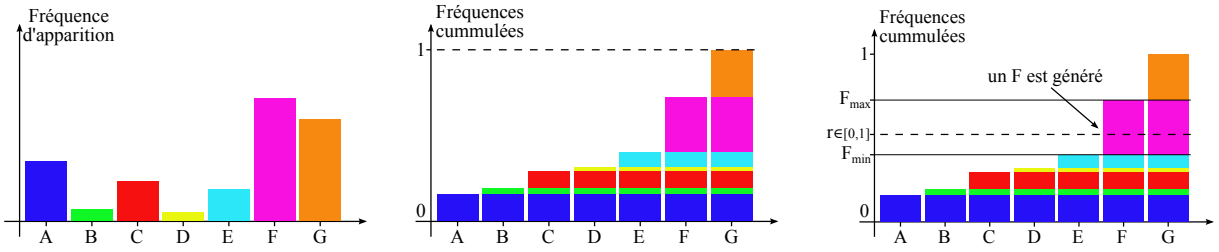
3.3.3 Modèles des $IDRF_{3D}$ et production des échantillons aléatoires

Nous avons vu comment les projecteurs *IRIS* estiment une *CDRF* à partir d'échantillons de points aléatoires, distribués suivant les $IDRF_{3D}$ associées. Afin d'exploiter cette méthode, nous avons donc besoin de générer ces points aléatoires. Dans ce contexte, nous avons développé deux modèles des $IDRF_{3D}$, qui permettent de générer facilement et rapidement des échantillons aléatoires.

3.3.3.1 Utilisation brute des $IDRF_{3D}$ estimées

La première méthode consiste à utiliser directement les $IDRF_{3D}$ voxélizées, estimées avec la procédure décrite précédemment. Cette méthode repose sur la création d'échantillons aléatoires suivant une certaine distribution en utilisant une estimation de sa fonction de répartition, représentée dans la figure 3.12. Pour une $IDRF_{3D}$ donnée, nous convertissons le problème 3D en un problème 1D. C'est-à-dire que chacun de ces *voxel* n'est plus décrit par ses coordonnées (x', y', z') mais par un indice qui lui est spécifique. La valeur d'un *voxel* donne la probabilité d'apparition de son indice 1D. Nous allons donc chercher à générer ces indice 1D, que nous pouvons ensuite reconvertir en position (x', y', z'). Pour générer ces indices 1D nous construisons d'abord leur histogramme cumulé et normalisé qui nous donne une estimation de leur fonction de répartition. Celle-ci nous donne une table de conversion permettant de passer d'une distribution uniforme dans l'intervalle $[0, 1]$ à la distribution décrite par l'histogramme. Ainsi, en générant un nombre aléatoire r entre $[0, 1]$, il suffit de trouver dans l'histogramme cumulé la classe qui contient cette valeur. Une fois la classe trouvée, celle-ci fournit l'indice du *voxel* et dont la position où sera généré le point aléatoire. Afin d'éviter que les positions générées

soient toujours placées au centre des *voxels* $IDRF_{3D}$, on déplace les points aléatoirement à l'intérieur de ceux-ci. Lors de la reconstruction avec les projecteurs *IRIS*, pour chaque *LOR*, il est nécessaire de générer plusieurs points aléatoires. Il est donc crucial que la recherche de la classe de l'histogramme cumulé qui inclue la valeur r, soit la plus rapide possible. L'histogramme cumulé est un tableau de valeurs croissantes, ce qui nous permet d'opter pour une méthode de recherche dichotomique, qui a l'avantage d'être simple d'implémentation et d'avoir une complexité en O(log(n)).



(a) Histogramme de la distribution

(b) Histogramme cumulé ou fonction de répartition

(c) Production d'un échantillon

FIGURE 3.12 – Production d'échantillons d'une distribution décrite par un histogramme. À partir d'un histogramme (a) nous donnant les fréquences d'apparition des différents événements possibles, nous calculons l'histogramme cumulé (b) qui nous permet de transformer des échantillons d'une distribution uniforme dans l'intervalle $[0,1]$ en échantillon de notre distribution (c) en trouvant dans quelle classe de l'histogramme est compris ce nombre.

Dans la suite de ce document, nous noterons $IRIS_{mesure}$ le projecteur qui utilise cette méthode de génération des points aléatoires.

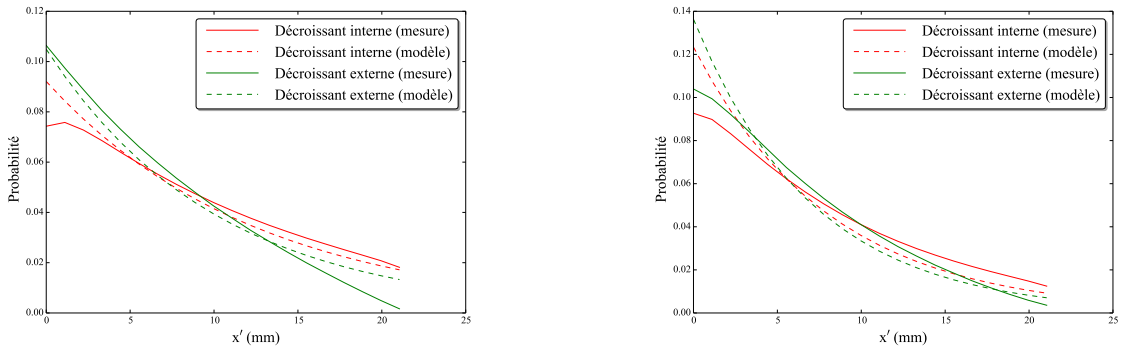
L'utilisation directe des $IDRF_{3D}$ nécessite de les stocker, or en fonction des échantillonnages de (x', y', z') et (θ, φ), cela peut nécessiter une quantité de mémoire importante. De plus, la création des points aléatoires implique de nombreuses lectures dans ces données, ce qui peut augmenter les temps de reconstruction. De plus, les $IDRF_{3D}$ estimées peuvent être bruitées si un nombre insuffisant de données obtenues par SMC, sont utilisées pour les construire, ce qui peut conduire à des estimations imprécises des *CDRF* .

3.3.3.2 Modèle analytique des $IDRF_{3D}$

Pour générer des points aléatoires qui suivent les distributions décrites par les $IDRF_{3D}$, rapidement, sans bruit et sans besoin de stockage excessif, nous avons développé un modèle analytique. Nous avons dérivé ce modèle des observations des $IDRF_{3D}$ estimées par SMC. Une contrainte de ce modèle était d'utiliser seulement des fonctions de lois de probabilité pour lesquelles il est facile de générer des échantillons. Les paramètres de ce modèle sont estimés pour chaque $IDRF_{3D}$ et stockés dans des tableaux pour être exploités au moment de la reconstruction.

En observant les $IDRF_{3D}$, on peut constater une discontinuité nette entre la distribution à l'intérieur du cristal et celle dans les cristaux voisins. Cette partie intérieure provient de la pénétration et de la diffusion intracristal, tandis que la distribution extérieur provient de la diffusion intercristaux. Nous avons choisi de modéliser ces deux parties séparément. Un paramètre P du modèle donne le pourcentage que représente la distribution dans le cristal par rapport à l'ensemble. Au moment de générer un point aléatoire, cette valeur permet de déterminer ci celui-ci sera dans le cristal ou dans

un de ses voisins. Pour cela, il suffit de générer un nombre r distribué uniformément dans $[0,1]$ et si $r < \mathrm{P}$ le point sera dans le cristal, sinon, il sera dans un voisin.



(a) $(\theta, \varphi) = (0°, 0°)$ (b) $(\theta, \varphi) = (45°, 0°)$

FIGURE 3.13 – Distributions interne et externe de $IDRF_{3D}$ le long de l'axe du cristal pour des angles d'incidences d'incidence (a) nul et (b) de $\theta = 45°$ et $\varphi = 0°$.

La figure 3.13 montre la somme selon x' et y' des distributions intérieurs et extérieurs de deux $IDRF_{3D}$. Pour ces deux parties du modèle, la distribution des points suivant x' est modélisée par une loi exponentielle de paramètre $\lambda_{x'\,int}$ pour la partie intérieure de la $IDRF_{3D}$ et $\lambda_{x'\,ext}$ pour la partie dans les cristaux voisins. La conversion d'un nombre r distribué uniformément dans $[0,1]$ vers une distribution exponentielle de paramètre λ, commençant en x'_0 (face avant du cristal) et finissant en x'_1 (face arrière du cristal) est donnée par la méthode de la transformée inverse et prend la forme suivante :

$$x' = \frac{-\log\left(r(1-\mathrm{e}^{-(x'_1-x'_0)\lambda}) + \mathrm{e}^{-(x'_1-x'_0)\lambda}\right)}{\lambda} + x'_0 \tag{3.8}$$

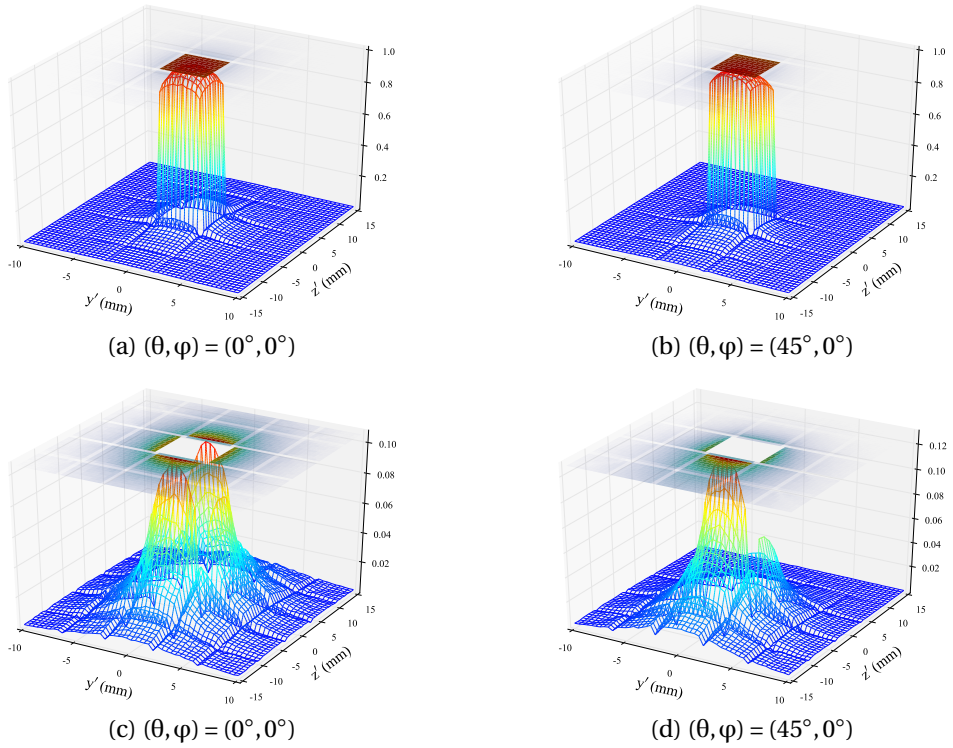
Ce modèle n'est pas totalement en accord avec les données obtenues par SMC. Il permet cependant de modéliser globalement la décroissance dans la profondeur du détecteur et de générer des échantillons aléatoires rapidement.

La figure 3.14 montre des coupes (y', z') pour $x' = 0$, de deux $IDRF_{3D}$. Concernant les positions (y', z') à l'intérieur du cristal, nous utilisons une simple distribution uniforme dans l'intervalle défini par les dimensions du cristal. À l'extérieur du cristal, la distribution est plus complexe comme on peut le voir sur les figures 3.14c et 3.14d. Pour cette distribution, nous utilisons une loi exponentielle 2D définie de la manière suivante :

$$\mathrm{IDRF}_{x'}(y', z') = \exp\left(-\sqrt{\left(\lambda_{y'}(y' - \mu_{y'})\right)^2 + \left(\lambda_{z'}(z' - \mu_{z'})\right)^2}\right) \tag{3.9}$$

où $\lambda_{y'}$ et $\lambda_{z'}$ sont les paramètres de la loi exponentielle 2D, $\mu_{y'}$ et $\mu_{z'}$ sont les décalages du centre de la distribution.

En utilisant un changement de variable en coordonnées polaires et la méthode de la transformée inverse, on obtient la relation qui permet de générer des points aléatoires suivant la distribution exponentielle de l'équation 3.9 à partir de deux nombres r et β uniformément répartis dans l'intervalle



(a) $(\theta, \varphi) = (0°, 0°)$ (b) $(\theta, \varphi) = (45°, 0°)$

(c) $(\theta, \varphi) = (0°, 0°)$ (d) $(\theta, \varphi) = (45°, 0°)$

FIGURE 3.14 – Coupe suivant le plan $x' = 0$ de la $IDRF_{3D}$ avec un angle d'incidence nul en (a) et (c) et avec $\theta = 45°$ et $\varphi = 0°$ en (b) et (d). (a) et (b) montrent les distributions complètes, tandis que (c) et (d) présentent seulement les distributions à l'extérieur du cristal.

$[0, 1]$. Cette relation prend la forme suivante :

$$\begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} -\frac{\log(1-\sqrt{r})}{\lambda_{z'} sqrt2} \cos(\beta 2\pi) + \mu_{y'} \\ -\frac{\log(1-\sqrt{r})}{\lambda_{z'} sqrt2} \sin(\beta 2\pi) + \mu_{z'} \end{pmatrix} \tag{3.10}$$

Avec cette méthode, des points peuvent être générés à l'intérieur du cristal. Pour éviter ce problème, une méthode de rejet est utilisée, c'est-à-dire qu'on répète la génération du point tant que celui-ci se trouve à l'intérieur du cristal.

En résumé, le modèle des $IDRF_{3D}$ se décompose en deux parties, une à l'intérieur et l'autre à l'extérieur du cristal, avec une proportion P de premières interactions à l'intérieur. La décroissance des interactions le long de l'axe du cristal est modélisée par une loi exponentielle de paramètre $\lambda_{x'\ int}$ à l'intérieur et $\lambda_{x'\ ext}$ à l'extérieur du cristal. Dans le plan orthogonal à l'axe du cristal, la distribution est uniforme à l'intérieur du cristal et suit une loi exponentielle 2D à l'extérieur du cristal. La loi exponentielle 2D est définie par les paramètres $\lambda_{z'}$, dans la direction de l'axe du scanner, et $\lambda_{y'}$, dans la direction perpendiculaire à l'axe du cristal et du scanner. Le centre de cette distribution est excentré de $\mu_{z'}$ et $\mu_{y'}$. Ces sept paramètres sont estimés, pour toutes les $IDRF_{3D}$ mesurées, par une optimisation minimisant la distance quadratique entre le modèle et les données mesurées. Les résultats sont stockés dans des tables pour être exploités au moment de la reconstruction. Dans la suite de ce document, nous noterons $IRIS_{analytique}$ le projecteur *IRIS* basé sur ce modèle analytique des $IDRF_{3D}$.

3.4 Implémentations des projecteurs

Les six projecteurs *IRIS*, de Siddon, de Chen, Gaussien$_{constant}$ et Gaussien$_{variant}$ ont été implémentés sur *CPU* et sur *GPU*. Le projecteur de Siddon, a été implémenté dans sa version itérative (iSiddon), proposée par [Zhao et Reader, 2002], plus rapide que la version originale. Avec les deux projecteurs Gaussiens, le tracé des *CDRF* se fait coupe par coupe avec la méthode de [Sportelli *et al.*, 2011]. Cette méthode consiste, dans un premier temps, à choisir la direction du volume qui est la plus parallèle à la direction de la *LOR*. Ensuite, en parcourant le volume coupe par coupe selon cette direction, on recherche le *voxel* où se trouve l'intersection avec la *LOR*. En partant de ce *voxel*, on étend itérativement la coupe dans les deux directions, en calculant en chaque *voxel* la valeur de la *CDRF* avec le modèle Gaussien associé. Cette croissance s'arrête lorsque la distance dans une des deux directions dépasse $3 \times \sigma$. Les coupes du volume n'étant généralement pas parfaitement perpendiculaires à la *LOR*, il est nécessaire d'appliquer des facteurs correctifs aux σ des modèles. Dans une des deux directions du plan de la coupe, si on note θ l'angle de la *LOR* avec cet axe du plan, la valeur corrigée est $\sigma' = \frac{\sigma}{sin(\theta)}$. Les projecteurs *IRIS* et le projecteur de Chen utilisent une construction des *CDRF* par l'accumulation de plusieurs. Celles-ci sont tracées avec l'implémentation de [Bert et Visvikis, 2011] de l'algorithme *DDA*, qui permet de générer des lignes binaires très efficacement. Les nombres aléatoires sont générés, avec le générateur congruentiel linéaire de [Lehmer, 1951], pour les implémentations *CPU*, et avec le générateur Xorshift de [Marsaglia, 2003], pour les implémentations *GPU*. Les implémentations *GPU* de tous les projecteurs traitent toujours une *LOR* avec un *thread*.

3.5 Étude d'évaluation

Dans cette section, nous avons évalué les projecteurs *IRIS*. Leurs performances ont été comparées avec celles d'autres approche, comme le projecteur linéique de Siddon décrit dans le paragraphe 3.2.1.3.1, les projecteurs Gaussien$_{constant}$ et Gaussien$_{variant}$ décrits dans le paragraphe 3.2.1.3.2,et le projecteur multiligne de Chen présenté dans la sous-section 3.2.2.3.

L'ensemble des données utilisées dans cette étude ont été obtenues par SMC sur la plate-forme *GATE* du scanner TEP Allegro/GEMINI de Philips, avec le modèle validé par [Lamare *et al.*, 2006]. Le mode *back-to-back* a été utilisé pour l'émission des photons d'annihilation afin d'accélérer les simulations, et de supprimer le parcours du positon et la non-colinéarité des photons d'annihilation qui ne sont pas modélisés avec les méthodes évaluées. Les effets physiques Compton, photoélectrique, Rayleigh et l'électroionisation ont été modélisés avec le modèle standard.

3.5.1 Construction des modèles des projecteurs

3.5.1.1 Jeu de données

Tous les modèles, des projecteurs utilisés dans cette étude, ont été construits à partir du même jeu de données. Ce jeu de données a été obtenu en simulant une source cylindrique homogène de $80cm$ de diamètre et $20cm$ de long, placée au centre du champ de vue du scanner et alignée avec son axe. L'activité dans la source a été fixée à une valeur faible (1 Becquerel (Bq)) afin d'éviter d'enregistrer

des coïncidences fortuites, qui ne sont pas modélisées par les projecteurs. Le matériau dans la source a été défini comme étant du vide, dans le but d'éviter l'atténuation et d'enregistrer des coïncidences diffusées, que les projecteurs ne modélisent pas. Cela permet également d'accélérer la simulation en réduisant le nombre d'interactions physiques à simuler et en augmentant la fraction de coïncidences détectées par annihilations produites. Nous avons exécuté cette simulation jusqu'à obtenir un jeu de données de 1 milliard de coïncidences vraies, ce qui a pris 20 heures sur 100 cœurs. Pour chaque coïncidence détectée, nous avons enregistré, la position de l'émission des photons d'annihilation, la position de la *LOR* et la position de la première interaction dans le détecteur des deux photons d'annihilation, dans le repère du cristal associé.

3.5.1.2 Projecteurs Gaussiens

Le projecteur Gaussien$_{variant}$ présenté dans le paragraphe 3.2.1.3.2 modélise les coupes des *CDRF* par des distributions Gaussiennes asymétriques définies par quatre variances, σ_{z1} et σ_{z2} dans la direction z de l'axe du scanner, σ_{x1} et σ_{x2} dans la direction x perpendiculaire à la direction de la *LOR* et à l'axe du scanner.

Pour estimer les paramètres de ce modèle, nous avons construit un ensemble d'images, chacune correspondant à un certain intervalle de positions (S,P), telles qu'elles sont définies dans la figure 3.5. Les positions (S,P) ont été échantillonnées régulièrement. Avec notre jeu de données, nous avons calculé pour chaque coïncidence la position (S,P), et mis la position de l'émission des photons d'annihilation dans le repère (x,z), où z est l'axe du scanner et x l'axe perpendiculaire à la *LOR* et à z. Dans la coupe correspondant à la position (S,P) calculée, le *pixel* correspond à la position (x,z) est incrémenté. Une fois toutes les coïncidences traitées, nous avons estimé pour chaque coupe les paramètres de la distribution Gaussienne 2D asymétrique, définie par l'équation 3.1, qui minimisent la distance quadratique avec les valeurs dans la coupe mesurée. Le scanner étant symétrique par rapport à l'origine en S et en P, nous avons construit les coupes des *CDRF* seulement pour les valeurs positives de S et P.

Nous avons construit 25×25 groupes de positions (S,P) avec des intervalles de positions de 10 mm en S et 10 mm en P. Chaque coupe de *CDRF* a été estimée en construisant les histogrammes des positions des émissions de photons d'annihilation, avec des classes de $0.2 \times 0.2 mm^2$. Le résultat de l'estimation des paramètres du modèle du projecteur Gaussien$_{variant}$ sont présenté dans la figure 3.15. On peut voir, sur cette figure, que pour les valeurs élevées de S la distribution en x s'élargit, ce qui est cohérent avec l'augmentation des erreurs de parallaxe qu'on observe dans ces régions du champ de vue.

Le projecteur Gaussien$_{constant}$ utilise des variances σ_x et σ_z qui ne varient pas. Il est donc important de définir des valeurs qui conviennent en moyenne pour tout le champ de vue. Nous avons fixé la *FWHM* en x à 4,3 mm et en z à 6,3 mm.

3.5.1.3 Projecteurs *IRIS*

Avec le jeu de données, nous avons estimé 12 $IDRF_{3D}$ avec le protocole présenté dans la sous-section 3.3.2. Celles-ci sont réparties régulièrement pour des intervalles (θ,φ) de $7,5° \times 7,5°$ avec θ

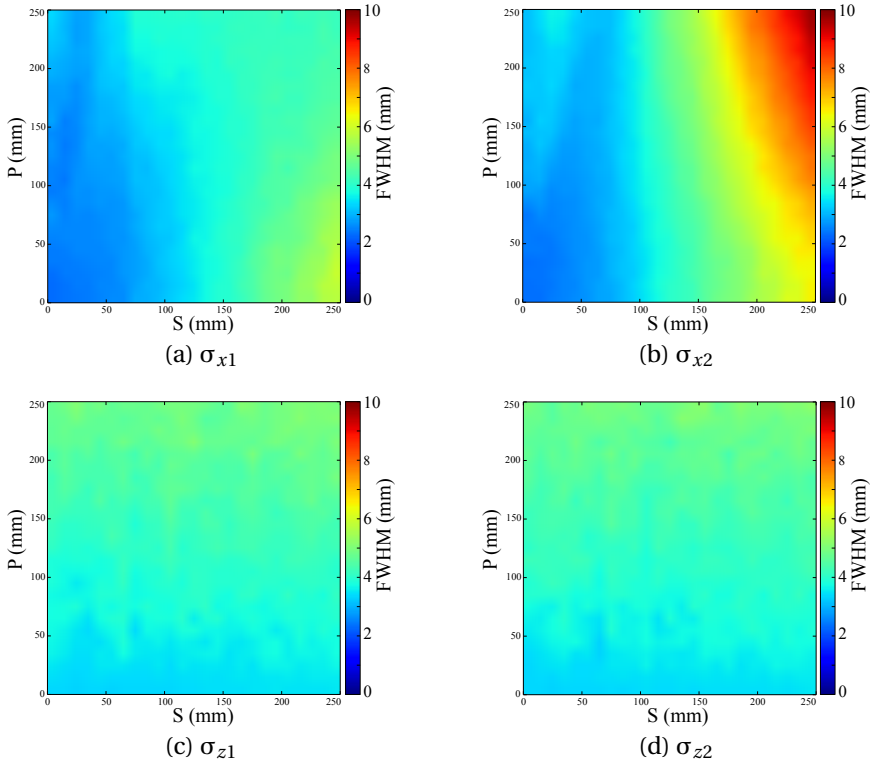


FIGURE 3.15 – Tables des paramètres du modèle du projecteur Gaussien$_{variant}$. Chacune de ces images donne une des variances, du modèle Gaussien asymétrique, en fonction de la position S de la *LOR* et de la position P dans la *LOR*.

variant de 0° à 45° et φ de 0° à 15°.

Ensuite, nous avons estimé les tableaux des valeurs des sept paramètres du modèle analytique du projecteur $IRIS_{analytique}$ avec le protocole présenté dans la sous-section 3.3.3.2. Ces tables sont présentées dans la figure 3.16.

3.5.2 Reconstructions

Toutes les reconstructions de cette étude ont été effectuées avec l'algorithme *LM-EM*, présenté dans le paragraphe 1.4.2.2, avec 100 itérations et trois tailles de *voxels* différentes, 4^3 mm^3 qui est la taille standard pour le scanner Allegro/GEMINI, ainsi que 2^3 mm^3 et 1^3 mm^3 afin d'évaluer les bénéfices des projecteurs en matière de résolution.

3.5.3 Estimation du nombre minimal de lignes nécessaire aux projecteurs multiligne

Les projecteurs *IRIS* et le projecteur de Chen estiment les *CDRF* de chaque *lor* en accumulant un certain nombre de lignes. Afin de fixer ce nombre, nous avons évalué la qualité d'images reconstruites avec des nombres de lignes accumulées allant de 1 à 1024, afin de déterminer à partir de quel nombre la qualité d'image n'évolue plus. Pour évaluer cette qualité d'image, nous avons utilisé le fantôme NEMA IEC NU 2-2001 (voir sous-section 2.6.1 du chapitre 2). Nous avons réalisé deux simulations de ce fantôme sur la plate-forme *GATE*, afin de produire deux jeux de données *list-mode* de 10 millions de coïncidences vraies chacun et indépendants l'un de l'autre, c'est-à-dire que les simulations ont été

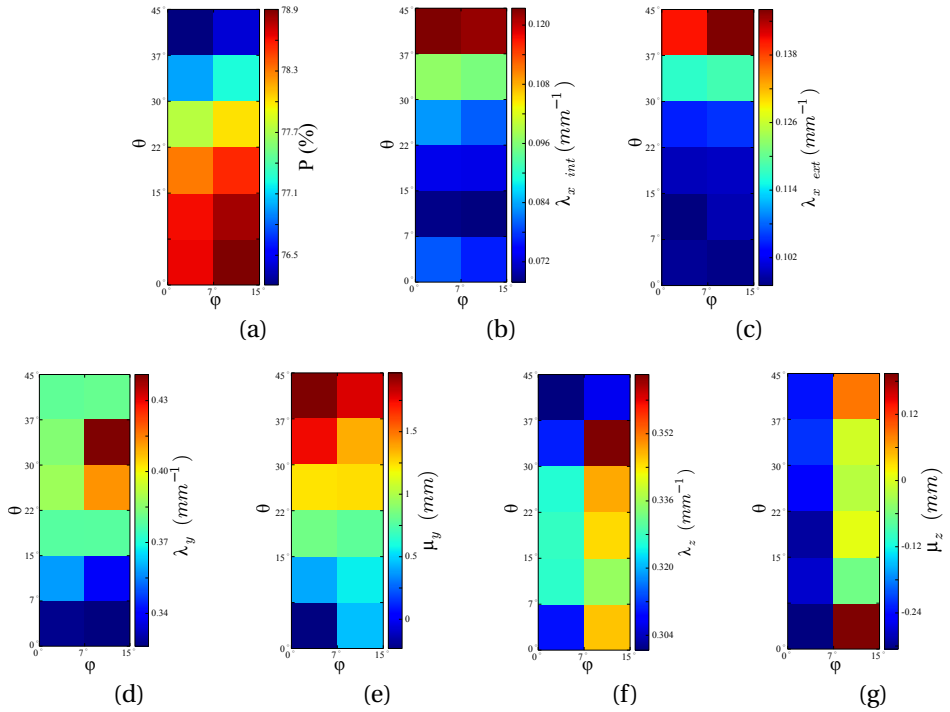


FIGURE 3.16 – Tableaux des paramètres estimés du modèle du projecteur $IRIS_{analytique}$ en fonction des angles d'incidences (θ, φ). (a) donne les fractions du nombre de points à l'intérieur du cristal sur le nombre total de points (dans le cristal et dans les voisins). (b) et (c) donnent les paramètres des lois exponentielles, respectivement, à l'intérieur et à l'extérieur du cristal. (e) et (g) donnent les positions du centre de la distribution exponentielle 2D et (d) et (f) les paramètres de cette loi.

initialisées avec des graines différentes pour le générateur de nombres aléatoires.

Dans les images reconstruites avec ces deux jeux de données, nous avons évalué à chaque itération deux critères, le contraste et le bruit. Les deux reconstructions ont permis d'estimer le bruit avec la méthode de [Lodge *et al.*, 2010]. Cette méthode se base sur une mesure de l'écart type entre deux répliques indépendantes d'un même signal, ici les reconstructions des deux jeux de données *list-mode*. Le bruit est estimé avec l'équation suivante :

$$Bruit_{CV} = 100 \times \frac{1}{S\sqrt{2}} \sum_{i}^{S} \frac{dsd_i}{a_i} \tag{3.11}$$

où a_i est la valeur moyenne, des *voxels* de la région d'intérêt dans la coupe i, S est le nombre total de coupes, dsd_i est l'écart type estimé avec l'équation suivante :

$$dsd_i = \sqrt{\frac{n\sum_j d_j{}^2 - (\sum_j d_j)^2}{n(n-1)}} \tag{3.12}$$

où d_j est la différence entre le *voxel* j d'une des deux reconstructions avec le même *voxel* j de l'autre reconstruction et n correspond au nombre de *voxels* dans la région d'intérêt dans la coupe i. Pour estimer ce $Bruit_{CV}$, nous avons défini une région d'intérêt cylindrique creuse dans le fond du fantôme, ayant le même axe que le fantôme, de 220 mm de diamètre extérieur, 80 mm de diamètre intérieur

et 30 mm de long. Le contraste a été mesuré dans les quatre sphères chaudes (activité élevée) et les deux sphères froides (aucune activité), relativement à l'activité mesurée dans le fond (dans la même région d'intérêt que pour l'estimation du bruit) avec la formule suivante :

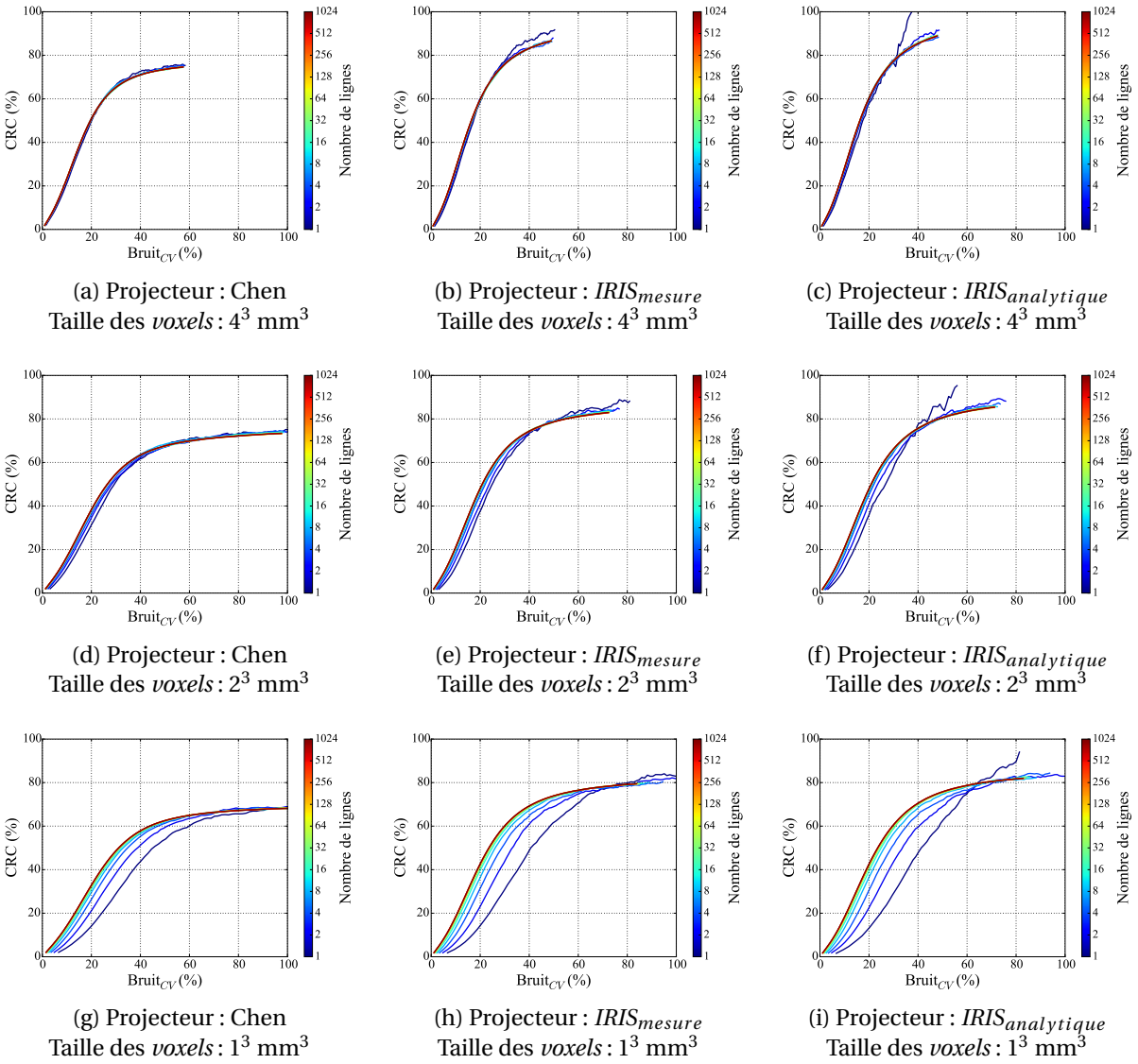
$$\mathrm{CRC} = \frac{\overline{r_c} - \overline{r_f}}{\overline{r_f}} \tag{3.13}$$

où, $\overline{r_c}$ est la valeur moyenne des *voxels* contenus dans la sphère considérée et $\overline{r_f}$ est la valeur moyenne des *voxels* dans le fond. Les valeurs de CRC seront exprimées en pourcentage de la valeur optimale, donnée par la source utilisée dans la SMC. Ce critère a été évalué à chaque itération sur seulement une des deux reconstructions. La qualité d'image optimale est obtenue avec un CRC de 100 % et un $Bruit_{\mathrm{CV}}$ nul.

La figure 3.17 montre l'évolution du CRC en fonction du $Bruit_{\mathrm{CV}}$ au cours des 100 itérations pour les trois projecteurs et les trois tailles de *voxels*, le CRC ayant été mesuré dans la plus petite sphère chaude (10 mm de diamètre).

Avec les plus petits nombres de lignes, on peut constater beaucoup d'instabilités sur les courbes, en particulier avec le projecteur $IRIS_{analytique}$. En observant les images reconstruites on se rend compte qu'avec une seule ligne par *LOR*, beaucoup de grandes valeurs apparaissent sur les bords du fantôme parce que les *CDRF* varient entre chaque itération et même entre la projection et la rétroprojection. En effet, les *CDRF* estimées avec les projecteurs *IRIS* et Chen ne sont jamais deux fois identiques parce qu'elles reposent sur une accumulation de lignes aléatoires. Toutefois, ce problème disparaît dès que le nombre de lignes augmente. Avec les *voxels* de 4^3 mm^3, à partir de 4 lignes accumulées, toutes les courbes se superposent. Plus la taille des *voxels* réduit plus le nombre de lignes minimal, pour que les courbes se superposent, augmente. On peut aussi noter que les courbes du projecteur de Chen se superposent pour des nombres plus faibles de lignes accumulées. Ce projecteur ne génère des échantillons qu'à l'intérieur du cristal, ce qui réduit la zone des positions possibles et par conséquent le volume des *CDRF* estimée. Ayant un volume plus faible, un nombre inférieur de lignes sera nécessaire pour le remplir. Le projecteur $IRIS_{mesure}$ peut quant à lui générer des positions dans l'ensemble de l'$IDRF_{3\mathrm{D}}$ mesurée, c'est-à-dire ici dans 5×5 cristaux, ce qui est bien plus étendu qu'avec le projecteur de Chen. Le projecteur $IRIS_{analytique}$ utilise un modèle analytique qui n'a pas de limite, il peut donc en théorie générer des échantillons infiniment éloignés du cristal, ce qui explique qu'il nécessite un plus grand nombre de lignes accumulées que les deux autres projecteurs. Le cas nécessitant le plus de lignes est donc celui du projecteur $IRIS_{analytique}$ avec des *voxels* de 1^3 mm^3. Pour un nombre de lignes accumulées supérieur ou égal à 16, toutes les courbes se superposent, ce qui indique que ce nombre suffit à atteindre le rapport CRC sur $Bruit_{\mathrm{CV}}$ maximal pour les mesures effectuées.

Par conséquent, pour toutes les reconstructions produites avec les projecteurs de Chen, et *IRIS*, nous avons fixé à 16 le nombre de lignes accumulées par *LOR*.



(a) Projecteur : Chen
Taille des *voxels* : 4^3 mm^3

(b) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 4^3 mm^3

(c) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 4^3 mm^3

(d) Projecteur : Chen
Taille des *voxels* : 2^3 mm^3

(e) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 2^3 mm^3

(f) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 2^3 mm^3

(g) Projecteur : Chen
Taille des *voxels* : 1^3 mm^3

(h) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 1^3 mm^3

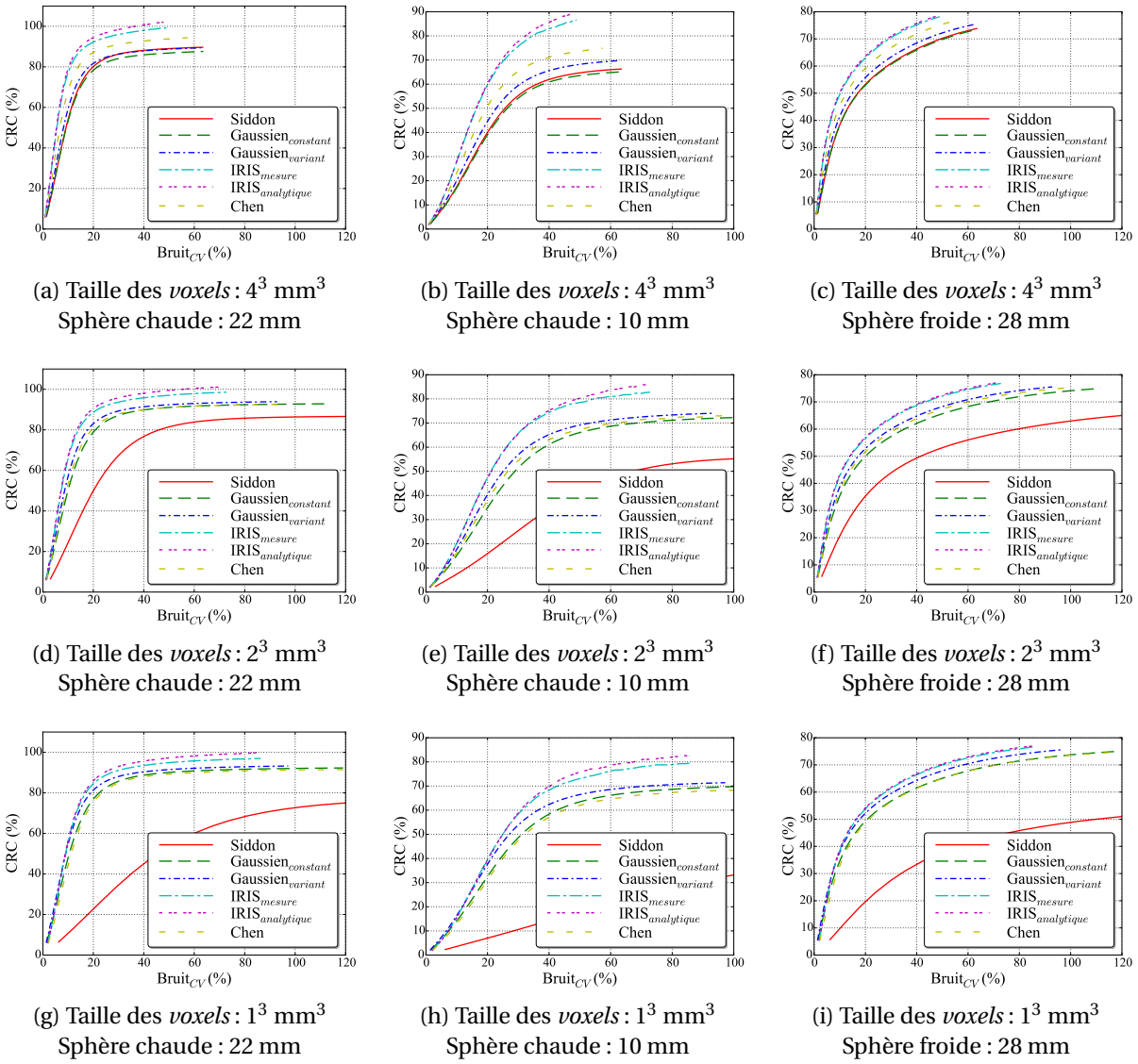
(i) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 1^3 mm^3

FIGURE 3.17 – Contraste dans la plus petite sphère chaude (10 mm de diamètre) en fonction du bruit dans le fond, mesuré dans les reconstructions du fantôme NEMA IEC NU 2-2001 avec les projecteurs de Chen, $IRIS_{mesure}$ et $IRIS_{analytique}$ et des nombres de lignes accumulées par *LOR* variant de 1 à 1024.

3.5.4 Évaluation de la qualité d'image

Le protocole présenté dans la section précédente a été utilisé pour évaluer la qualité d'image obtenue avec les six projecteurs différents. Le CRC a été estimé dans deux sphères chaudes, de 22 mm et 10 mm de diamètre, et une sphère froide, de 28 mm de diamètre. Le $Bruit_{\mathrm{CV}}$ a été estimé dans la même région d'intérêt que précédemment. Ces résultats sont présentés dans la figure 3.18.

Dans tous les cas, à un niveau de bruit donné, les deux projecteurs *IRIS* donnent les valeurs de contraste les plus élevées, avec un avantage plus marqué pour la plus petite sphère chaude. Le projecteur de Siddon fournit des résultats honorables lorsque la taille des *voxels* est fixée à 4^3 mm^3, mais il fournit aussi les plus mauvais résultats avec des *voxels* plus petits, ce qui est la conséquence de l'incapacité du projecteur de Siddon à moduler la largeur de la *CDRF*, celle-ci étant toujours fixée par la



(a) Taille des *voxels* : 4^3 mm^3 Sphère chaude : 22 mm

(b) Taille des *voxels* : 4^3 mm^3 Sphère chaude : 10 mm

(c) Taille des *voxels* : 4^3 mm^3 Sphère froide : 28 mm

(d) Taille des *voxels* : 2^3 mm^3 Sphère chaude : 22 mm

(e) Taille des *voxels* : 2^3 mm^3 Sphère chaude : 10 mm

(f) Taille des *voxels* : 2^3 mm^3 Sphère froide : 28 mm

(g) Taille des *voxels* : 1^3 mm^3 Sphère chaude : 22 mm

(h) Taille des *voxels* : 1^3 mm^3 Sphère chaude : 10 mm

(i) Taille des *voxels* : 1^3 mm^3 Sphère froide : 28 mm

FIGURE 3.18 – Contraste en fonction du bruit, au cours des itérations, dans les images reconstruites du fantôme NEMA IEC NU 2-2001 pour les six projecteurs comparés, avec des tailles de *voxels* de 4^3 mm^3, 2^3 mm^3 et 1^3 mm^3. Le contraste a été mesuré dans deux sphères chaudes, une de 22 mm de diamètre et l'autre de 10 mm de diamètre, et une sphère froide de 28 mm de diamètre.

taille des *voxels*. Pour les petites tailles de *voxels* l'estimation des *CDRF* est beaucoup trop fine. Les projecteurs Gaussiens et le projecteur de Chen fournissent des résultats très similaires dans tous les cas, sauf pour la taille des *voxels* de 4^3 mm^3 ou le projecteur de Chen apporte un meilleur contraste.

Généralement, le nombre d'itération d'une reconstruction est limité pour que le bruit ne prenne pas le dessus sur les informations contenues dans l'image. Sur la figure 3.18 on observe que le niveau de bruit n'augmente pas à la même vitesse. Visuellement, un niveau de $Bruit_{CV}$ de 30 % semble être un compromis acceptable entre bruit et signal dans la plupart des reconstructions que nous avons observées. Dans le tableau 3.1 nous pouvons voir, pour chaque reconstruction, le nombre d'itérations qui ont été nécessaires pour atteindre le niveau de $Bruit_{CV}$ de 30 %.

Comme observé sur les courbes du contraste en fonction du bruit, on peut voir que les projec-

TABLEAU 3.1 – Nombre d'itérations pour atteindre un niveau de $Bruit_{CV}$ de 30 % dans le fond du fantôme NEMA IEC NU 2-2001.

Projecteur	4^3 mm^3	2^3 mm^3	1^3 mm^3
Siddon	28	10	4
$\text{Gaussien}_{constant}$	28	22	22
$\text{Gaussien}_{variant}$	32	27	26
Chen	40	25	21
$IRIS_{mesure}$	52	35	30
$IRIS_{analytique}$	54	37	31

teurs *IRIS* permettent d'effectuer un plus grand nombre d'itérations avant d'atteindre un niveau de bruit fixé et ainsi d'atteindre un meilleur contraste. Le projecteur de Siddon quant à lui ne permet pas d'opérer un grand nombre d'itérations avant d'atteindre le bruit de 30 %, particulièrement avec *voxels* de petites tailles. Les projecteurs Gaussien et le projecteur de Chen permettent d'effectuer des nombres d'itérations proches.

3.5.5 Temps de reconstruction

Nous avons mesuré les temps de reconstruction associés aux différents projecteurs, avec les trois tailles de *voxels* précédemment utilisées 4^3 mm^3, 2^3 mm^3 et 1^3 mm^3. Le jeu de données utilisé est celui basé sur le fantôme NEMA IEC NU 2-2001, qui a été présenté dans la sous-section 3.5.4. Les reconstructions ont été exécutées, sur *CPU* et sur *GPU*, le *CPU* étant un Intel Xeon E5-2680 à 2.7GHz et le *GPU* celui d'une carte Nvidia GTX 980 Ti à 1GHz. Ces résultats sont présentés dans le tableau 3.2.

TABLEAU 3.2 – Temps de reconstruction en secondes, avec les différents projecteurs, sur *CPU* et sur *GPU*. Ils correspondent au temps nécessaire pour effectuer une itération de l'algorithme *LM-EM* avec un jeu de données *list-mode* du fantôme NEMA IEC NU 2-2001 d'un million de coïncidences.

Projecteur	*CPU*			*GPU*		
	4^3 mm^3	2^3 mm^3	1^3 mm^3	4^3 mm^3	2^3 mm^3	1^3 mm^3
Siddon	4,9	9,6	32	0,02	0,17	0,59
$\text{Gaussien}_{constant}$	38	121	973	0,90	3,5	34
$\text{Gaussien}_{variant}$	56	215	1507	0,28	5,7	53
Chen	27	47	140	0,14	2,1	6,6
$IRIS_{mesure}$	28	46	100	0,18	2,1	6,3
$IRIS_{analytique}$	29	49	160	0,17	2,0	6,0

CPU : Intel Xeon E5-2680 à 2.7GHz
GPU : NVIDIA GTX 980 Ti à 1GHz

Les projecteurs Gaussiens sont les plus lents, suivis des projecteurs *IRIS* et Chen, dix fois plus rapides, puis du projecteur de Siddon, encore dix fois plus rapide. Le *GPU* permet d'accélérer les différents projecteurs d'un facteur compris entre 20 et 30 fois. Les temps d'exécution des projecteurs

IRIS ne dépassent pas 10 fois le temps d'exécution d'un projecteur rapide comme le projecteur de Siddon. De plus, l'utilisation d'un *GPU* rend ce type de projecteurs totalement compatible avec les applications en clinique.

3.5.6 Évaluation de la résolution

Pour évaluer la résolution, nous avons simulé un fantôme constitué de quatre sources ponctuelles distribuées le long de l'axe x dans le plan perpendiculaire à l'axe du scanner, et passant au centre du champ de vue. Les sources sont placées aux positions x = 0 mm, x = 66 mm, x = 133 mm et x = 200 mm afin d'observer les effets dus aux erreurs de parallaxe lorsque l'on s'éloigne du centre du champ de vue du scanner. Aucun fantôme matériel n'a été inséré dans cette simulation. Le jeu de données obtenu est composé de 2 millions de coïncidences vraies.

À chaque itération, nous avons mesuré dans la coupe centrale des images reconstruites, la largeur d'une distribution Gaussienne ajustée, par la méthode des moindres carrées, aux profils passant le long de l'axe x par chacune des quatre sources ponctuelles.

Il faut noter que l'estimation de la résolution avec une source ponctuelle dans un fond sans activité reconstruite avec les algorithmes du type *ML-EM*, peut entraîner une sous-estimation de la résolution, comme mentionné par [Moehrs *et al.*, 2008]. Cependant, dans cette étude nous nous intéressons à évaluer les projecteurs de manière relative, les uns par rapport aux autres.

Dans la figure 3.19 nous pouvons voir une coupe des images reconstruites de notre fantôme composé de quatre sources ponctuelles, à l'itération 100 et pour des *voxels* de $1^3 mm3$. Le projecteur de Siddon donne dans tous les cas une reconstruction de la source ponctuelle qui est beaucoup plus large que les autres projecteurs. On peut aussi remarquer une légère dégradation de la résolution après avoir atteint son minimum. Cette dégradation est une conséquence de la taille trop faible des *voxels* pour ce projecteur, ce qui entraîne des problèmes d'échantillonnage du champ de vue, avec certain *voxel* qui ne sont dans aucune *CDRF*. Ce problème entraîne l'apparition de motifs dans l'image, qui s'amplifient avec les itérations, ce qui peut réduire la résolution. Au centre du champ de vue, les cinq autres projecteurs semblent fournir des résolutions similaires. Le projecteur Gaussien$_{constant}$ donne une reconstruction qui s'étale de plus en plus à mesure qu'on s'éloigne du centre du champ de vue. Les projecteurs Gaussien$_{variant}$, de Chen et *IRIS* fournissent quant à eux des reconstructions qui varient très peu en fonction de la position dans le champ de vue. Mais, le projecteur Gaussien$_{variant}$ semble être celui qui fournit les résolutions les plus homogènes.

La figure 3.20 montre l'évolution de la résolution estimée sur les quatre sphères. Ces résultats confirment ce que nous avons pu observer sur les images reconstruites. On voit bien ici qu'avec le projecteur de Siddon, la résolution sature rapidement vers une *FWHM* de 2,7 mm au centre du champ de vue et 5,5 mm à 200 mm du centre du champ de vue. Le projecteur Gaussien$_{constant}$ donne une résolution de 1 mm au centre du champ de vue, similaire à ce qu'on obtient avec les autres projecteurs, mais elle se dégrade rapidement en s'éloignant du centre du champ de vue jusqu'à atteindre 4,5 mm. Les projecteurs Gaussient$_{variant}$, de Chen, et *IRIS* fournissent des résolutions très similaires, avec encore un très léger avantage pour le projecteur Gaussien. Avec ces projecteurs, on obtient une résolution d'environ 1 mm au centre et de 2 mm à 200 mm du centre du champ de vue.

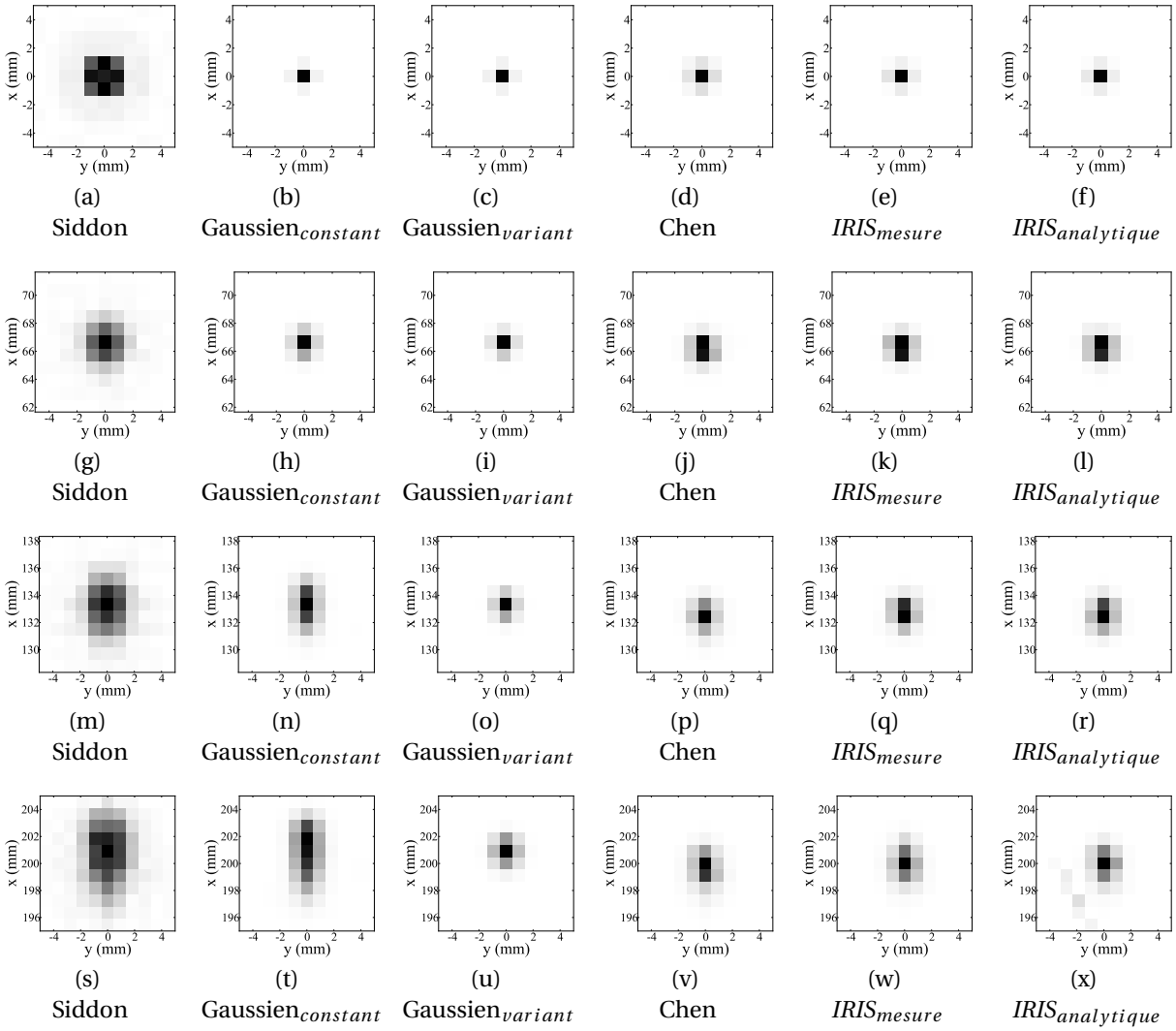
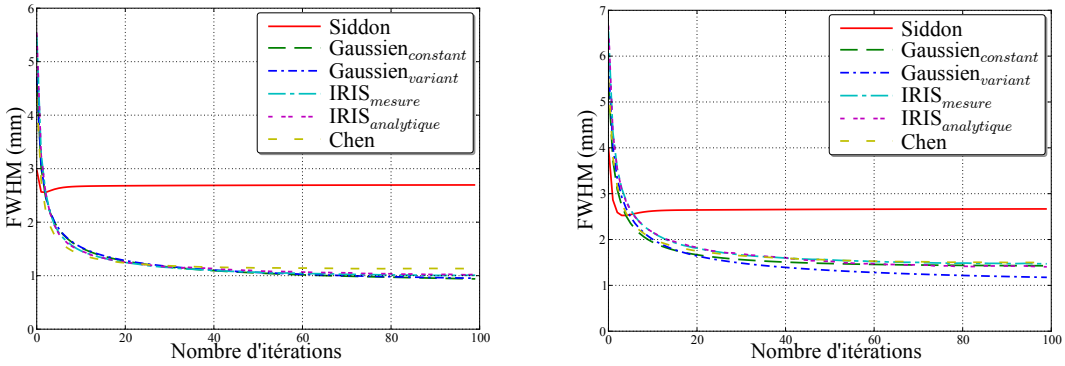


FIGURE 3.19 – Reconstructions du fantôme composé de quatre sources ponctuelles avec une taille des *voxels* de 1^3 mm^3. La première ligne montre la source placée au centre du champ de vue, la deuxième ligne la source en $x = 66$ mm, la troisième ligne la source en $x = 133$ mm et la dernière ligne la source en $x = 200$ mm

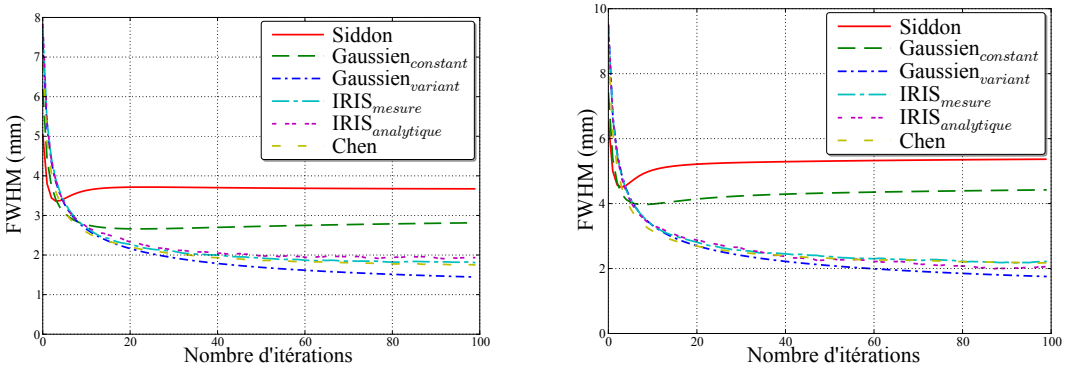
3.5.7 Évaluations avec des fantômes anthropomorphiques

3.5.7.1 Fantômes et jeux de données

Afin d'évaluer les différents projecteurs dans un contexte plus réaliste, nous avons aussi construit des jeux de données basés sur des fantômes anthropomorphiques. Ces deux fantômes, dont les sources sont présentées dans la figure 3.21, ont été simulés avec *GATE* et des cartes d'activité et de matériaux toutes deux voxélisées. Le premier fantôme, qu'on appellera NCAT$_1$, contient une tumeur pulmonaire hétérogène avec deux niveaux d'activité dont le rapport de concentration vaut 1,8. Le second fantôme, qu'on nommera NCAT$_2$, contient deux tumeurs hépatiques sphériques de 20 mm et 36 mm de diamètre, dont la concentration d'activité est 2,42 fois celle présente dans le foie. Ces deux fantômes ont été simulés, jusqu'à obtenir des jeux de données *list-mode* de 15 millions de coïncidences vraies.

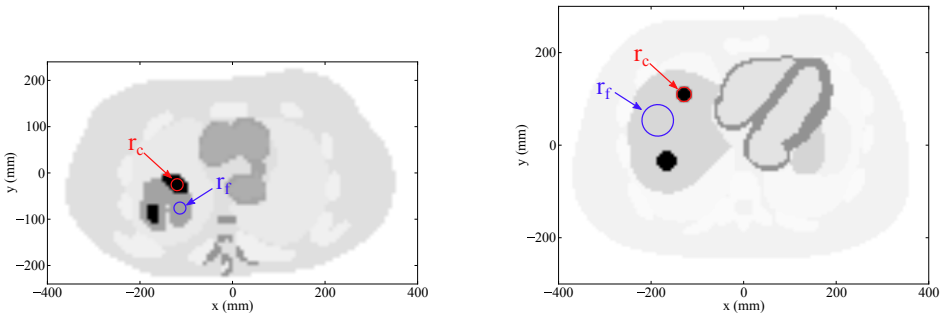


(a) Position source : $x = 0$ mm (b) Position source : $x = 66$ mm



(c) Position source : $x = 133$ mm (d) Position source : $x = 200$ mm

FIGURE 3.20 – Résolution des images reconstruites en fonction des itérations estimées en différentes positions du champ de vue.



(a) NCAT$_1$ (b) NCAT$_2$

FIGURE 3.21 – Coupes transverses des cartes d'activités des fantômes anthropomorphiques simulés. Le contraste dans le fantôme (a) est évalué dans la tumeur hétérogène en mesurant le CRC de l'activité mesurée dans une région d'intérêt active r_c de 12 mm de diamètre, relativement à une région d'intérêt moins active r_f, elle aussi de 12 mm de diamètre. Le contraste dans le fantôme (b) est mesuré entre la région d'intérêt active r_c de 20 mm de diamètre, et une région d'intérêt r_f de 40 mm de diamètre, dans le foie.

3.5.7.2 Reconstructions et facteurs de mérite

Les fantômes ont été reconstruits avec les nombres d'itérations donnés dans le tableau 3.1, pour obtenir un niveau de $Bruit_{CV}$ fixe de 30 %. Les images ont été évaluées par une analyse visuelle dans un premier temps, puis par une mesure du contraste, calculé avec l'équation 3.13 dans les régions d'intérêts définies dans la figure 3.21.

3.5.7.3 Résultats

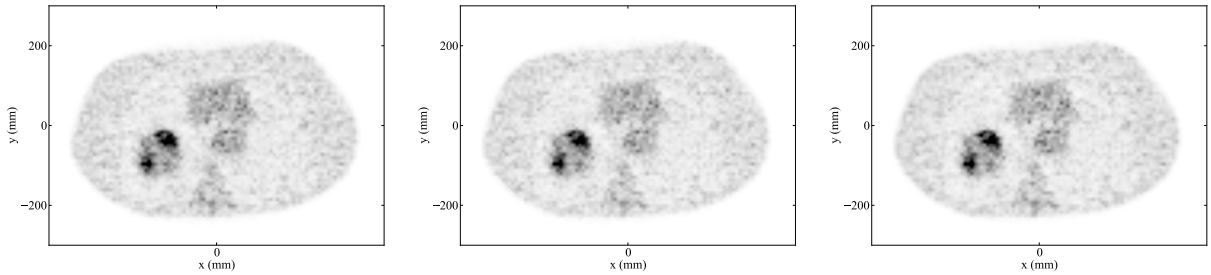
Les figures 3.22 et 3.22 montrent les coupes des reconstructions obtenues avec les six projecteurs et deux tailles de *voxels*, 4^3 mm^3 et 1^3 mm^3. Avec le fantôme NCAT$_1$, les hétérogénéités de la tumeur semblent plus contrastées avec les projecteurs *IRIS* et Chen. La réduction de la taille des *voxels* permet de faire apparaître des détails un peu plus nettement, comme le nécrose au milieu de la tumeur. Cependant, avec le projecteur de Siddon, le contraste est très réduit et on peut observer un "trou" au milieu de l'image qui est la conséquence de *CDRF* trop fines, qui n'atteignent jamais ces *voxels*.

Avec le fantôme NCAT$_2$, la structure du myocarde est plus nette qu'avec les projecteurs Gaussien$_{variant}$, *IRIS* et Chen, bien qu'on ne puisse pas observer de différences notables entre eux. Comme avec le fantôme précédent, l'utilisation de *voxels* de plus petite taille permet de reconstruire certains détails plus finement, comme la fine paroi sur la partie basse du cœur. À l'inverse, avec le projecteur de Siddon, l'utilisation de petits *voxels* réduit le contraste. Ici, on ne constate pas de trou dans l'image parce que la coupe affichée n'a pas été extraite à la même position le long de l'axe du scanner.

Le tableau 3.3 montre les valeurs de contrastes obtenues dans ces reconstructions. Les projecteurs *IRIS* fournissent les meilleurs contrastes, avec des valeurs de 5 % à 10 % plus élevées qu'avec le projecteur Gaussien$_{variant}$. Le projecteur *IRIS*$_{analytique}$ permet d'obtenir des contrastes très légèrement supérieurs au projecteur *IRIS*$_{mesure}$. Les projecteurs *IRIS* sont les seuls à fournir systématiquement des contrastes supérieurs à 90 %, tandis que le projecteur de Siddon fournit des contrastes toujours inférieurs à cette valeur.

TABLEAU 3.3 – Contrastes mesurés dans les fantômes anthropomorphiques reconstruits avec les nombres d'itérations donnés dans le tableau 3.1.

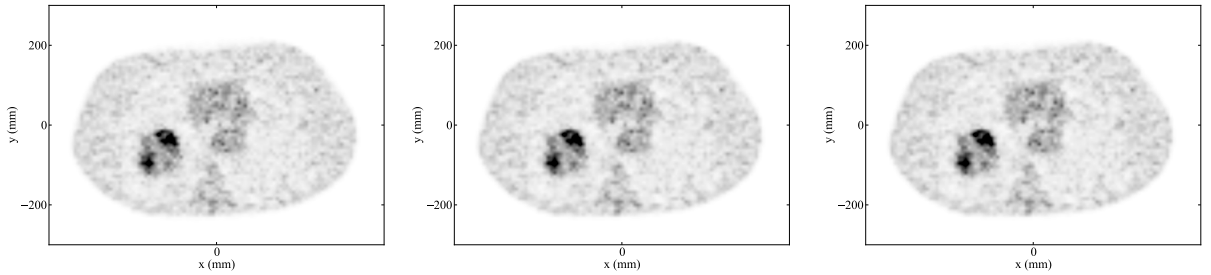
Projecteur	NCAT$_1$			NCAT$_2$		
	4^3 mm^3	2^3 mm^3	1^3 mm^3	4^3 mm^3	2^3 mm^3	1^3 mm^3
Siddon	89%	72%	39%	86%	62%	29%
Gaussien$_{constant}$	83%	88%	89%	85%	87%	85%
Gaussien$_{variant}$	87%	89%	90%	92%	92%	89%
Chen	92%	88%	88%	93%	88%	83%
IRIS$_{mesure}$	99%	93%	94%	99%	97%	92%
IRIS$_{analytique}$	102%	95%	95%	102%	100%	94%



(a) Projecteur : Siddon
Taille des *voxels* : 4^3 mm^3

(b) Projecteur : Gaussien$_{constant}$
Taille des *voxels* : 4^3 mm^3

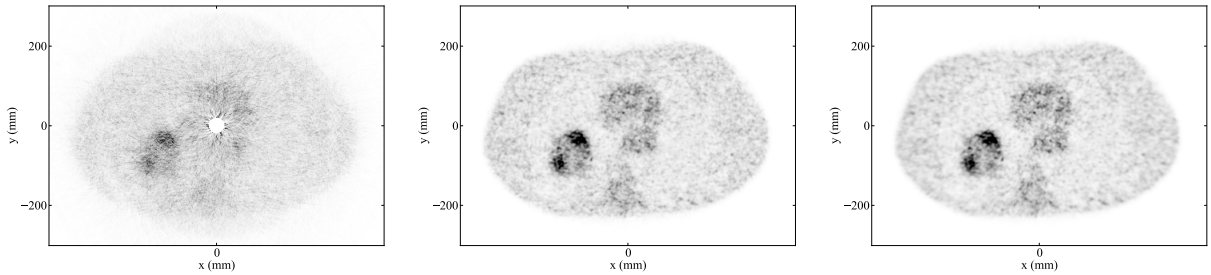
(c) Projecteur : Gaussien$_{variant}$
Taille des *voxels* : 4^3 mm^3



(d) Projecteur : Chen
Taille des *voxels* : 4^3 mm^3

(e) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 4^3 mm^3

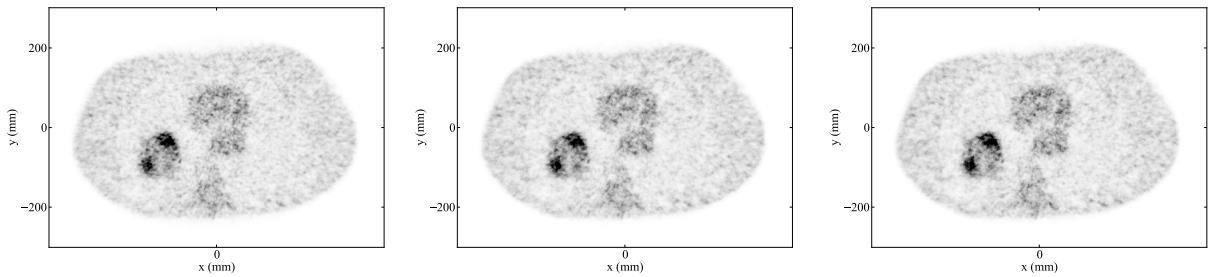
(f) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 4^3 mm^3



(g) Projecteur : Siddon
Taille des *voxels* : 1^3 mm^3

(h) Projecteur : Gaussien$_{constant}$
Taille des *voxels* : 1^3 mm^3

(i) Projecteur : Gaussien$_{variant}$
Taille des *voxels* : 1^3 mm^3

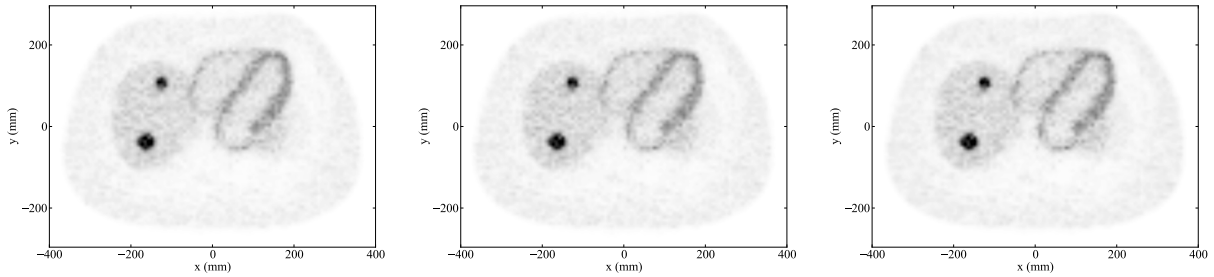


(j) Projecteur : Chen
Taille des *voxels* : 1^3 mm^3

(k) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 1^3 mm^3

(l) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 1^3 mm^3

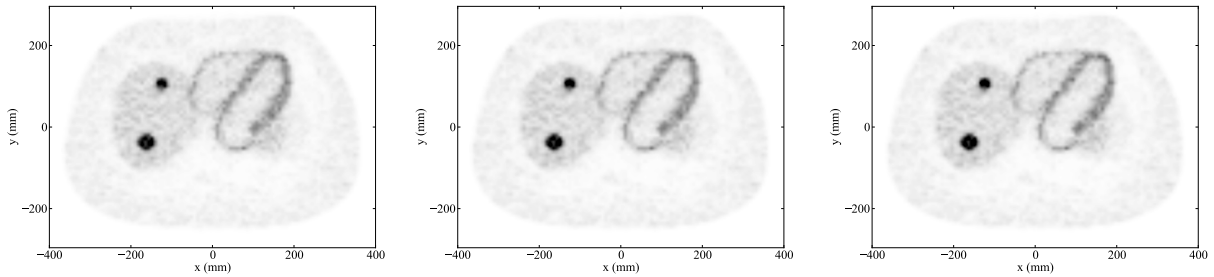
FIGURE 3.22 – Reconstructions du fantôme NCAT$_1$ avec des *voxels* de 4^3 mm^3 et 1^3 mm^3.



(a) Projecteur : Siddon
Taille des *voxels* : 4^3 mm^3

(b) Projecteur : Gaussien$_{constant}$
Taille des *voxels* : 4^3 mm^3

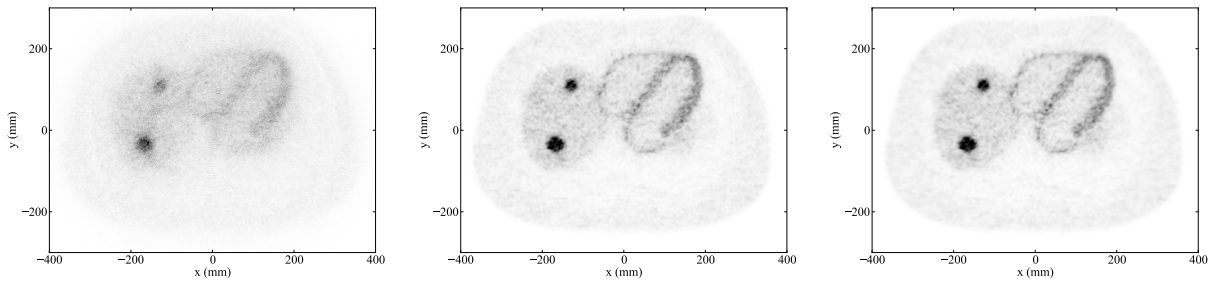
(c) Projecteur : Gaussien$_{variant}$
Taille des *voxels* : 4^3 mm^3



(d) Projecteur : Chen
Taille des *voxels* : 4^3 mm^3

(e) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 4^3 mm^3

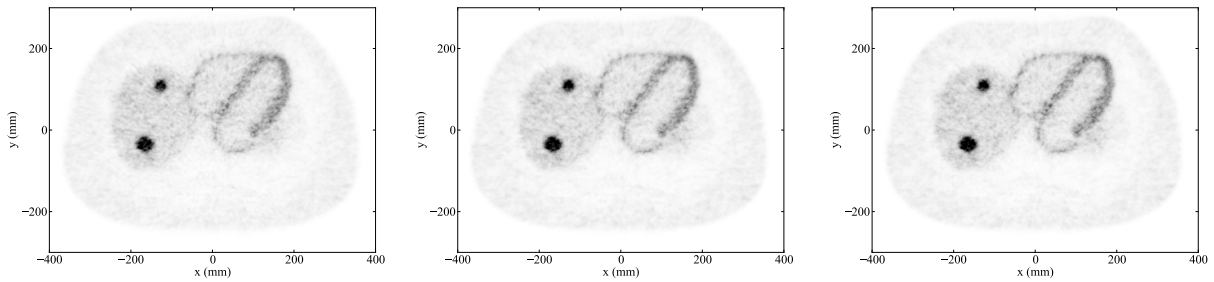
(f) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 4^3 mm^3



(g) Projecteur : Siddon
Taille des *voxels* : 1^3 mm^3

(h) Projecteur : Gaussien$_{constant}$
Taille des *voxels* : 1^3 mm^3

(i) Projecteur : Gaussien$_{variant}$
Taille des *voxels* : 1^3 mm^3



(j) Projecteur : Chen
Taille des *voxels* : 1^3 mm^3

(k) Projecteur : $IRIS_{mesure}$
Taille des *voxels* : 1^3 mm^3

(l) Projecteur : $IRIS_{analytique}$
Taille des *voxels* : 1^3 mm^3

FIGURE 3.23 – Reconstructions du fantôme NCAT$_1$ avec des *voxels* de 4^3 mm^3 et 1^3 mm^3.

3.6 Discussion et conclusion

Les reconstructions itératives en TEP modélisant la réponse du détecteur, utilisent soit une matrice système préestimée et stockée, soit un projecteur qui calcule cette réponse à la volée pendant la reconstruction. Cette première approche présente comme avantage de permettre d'utiliser des méthodes très coûteuses en temps de calcul, parce que l'étape d'estimation de la réponse du système n'est pas réalisée pendant la reconstruction. Cependant, il faut ensuite stocker cette estimation, ce qui est impossible sans effectuer certaines approximations et sans utiliser de méthodes de compression, ce qui peut induire des pertes de qualité des images reconstruites. L'accès aux éléments de la matrice système peut aussi être ralenti par la compression appliquée à ces données, ce qui peut réduire la vitesse de reconstruction. Un autre problème des matrices système stockées est leur manque de flexibilité. C'est-à-dire qu'une fois construite, il n'est plus possible de modifier le champ de vue ou la taille des *voxels*. Dans ce chapitre, nous nous sommes intéressés à l'utilisation de projecteurs, ce qui permet de résoudre tous les problèmes de stockage et de flexibilité. Il y a quelques années encore, ce type de projecteur était contraint d'utiliser des modèles grossiers afin de conserver des temps de reconstruction raisonnables. Aujourd'hui, avec le calcul sur *GPU*, nous avons accès à des puissances de calcul qui permettent d'envisager l'utilisation de modèles plus élaborés, plus précis et plus coûteux en calcul, en conservant des temps de reconstruction compatibles avec les applications cliniques.

Dans ce chapitre, nous avons introduit deux nouveaux projecteurs calculant à la volée la matrice de réponse du détecteur, permettant d'effectuer les opérations de projection et de rétroprojection nécessaires aux algorithmes de reconstruction itératifs. Ces projecteurs, $IRIS_{mesure}$ et $IRIS_{analytique}$, sont basés sur l'échantillonnage aléatoire des $IDRF_{3D}$ et l'accumulation d'une multitude de lignes simples pour estimer la réponse du détecteur en modélisant l'ensemble des effets physiques et géométriques. $IRIS_{mesure}$ utilise des histogrammes 3D comme modèles des $IDRF_{3D}$, tandis que $IRIS_{analytique}$ utilise un modèle analytique de ces fonctions. Nous avons pu voir que ces projecteurs permettent d'obtenir une qualité d'image supérieure à celle obtenue par les projecteurs Gaussiens, qui sont l'état de l'art actuel dans ce contexte. Le modèle Gaussien permet de modéliser la variation de la réponse du détecteur dans le champ de vue, mais ne modélise pas précisément les effets intercristaux telle la diffusion. Nous avons aussi intégré dans notre évaluation le projecteur de Chen, sur lequel nous nous sommes basés pour développer les projecteurs *IRIS*. Ce projecteur utilise aussi une stratégie multiligne, mais il ne modélise que les effets liés à la géométrie et à la pénétration intracristal et pas la diffusion ou la pénétration intrecristaux. Nos projecteurs ont permis d'obtenir des contrastes plus de 20 % supérieurs aux projecteurs Gaussiens, tout en fournissant une résolution équivalente et un temps de reconstruction sur *GPU* 10 fois plus court. En étant seulement 10 fois moins rapide que le projecteur de Siddon, les projecteurs *IRIS* permettent d'obtenir une résolution deux fois inférieure et des contrastes largement supérieurs (entre 15 % et 60 %). En effet, le projecteur de Siddon ne modélise aucun effet physique ou géométrique. Le projecteur $IRIS_{analytique}$ surpasse très légèrement le projecteur $IRIS_{mesure}$ en matière de contraste. Cela s'explique par la présence de bruit dans les histogrammes 3D qui estiment les $IDRF_{3D}$. En effet, nous avons utilisé 1 milliard de coïncidences pour les construire, ce qui n'est pas suffisant pour estimer avec un bruit faible la composante de diffusion intercritaux. Le modèle du projecteur $IRIS_{analytique}$ est dérivé de ces $IDRF_{3D}$, mais étant analytique,

il ne souffre pas de ce bruit.

Concernant les temps de reconstruction, les projecteurs Gaussiens pourraient être accélérés en réduisant la distance d'arrêt du tracé des coupes des *CDRF*, définie dans la sous-section 3.4, ce qui aurait pour effet de réduire le nombre de *voxels* dans les *CDRF* estimées, donc de réduire la charge de travail, mais aussi le nombre d'accès mémoire. Il faudrait cependant évaluer l'impact d'un tel changement sur la qualité des images reconstruites. Le projecteur de Chen, proche dans le principe aux projecteurs *IRIS*, se base sur un modèle de $IDRF_{3D}$ très simple qui ne dépend que d'un seul paramètre. Malgré cette simplicité, ce projecteur fournit des résultats très proches de ceux obtenus avec le projecteur Gaussien$_{variant}$, dont les paramètres sont bien plus nombreux, l'implémentation et l'estimation du modèle beaucoup plus complexes et qui est aussi beaucoup plus coûteux en temps de calcul.

Un point qui n'a pas été présenté est la capacité à intégrer des informations supplémentaires comme le *TOF* ou la *DOI* dans les projecteurs. De manière générale, l'intégration du *TOF* est toujours possible en appliquant des pondérations aux *voxels* le long de la *LOR*. La *DOI* présente plus de problèmes. En effet, la réponse du système varie en fonction de la *DOI*, et pas seulement en translation. Avec un projecteur Gaussien, il faudrait construire un modèle spécifique pour prendre en compte cette information. Avec les projecteurs *IRIS*, il est relativement facile de modifier le modèle des $IDRF_{3D}$ pour prendre en compte l'information de *POI*. En effet, cette information donnerait une profondeur plus probable pour les points générés avec les *IDRF*.

Dans cette étude, nous avons négligé l'impact de la non-colinéarité des photons d'annihilation. Une perspective de développement des projecteurs *IRIS* serait d'intégrer la non-colinéarité à l'équation 3.4 liant les $IDRF_{3D}$ et la *CDRF*. Connaissant la distribution de l'angle de non-colinéarité, donnée dans [Levin et Hoffman, 1999], il serait alors possible de générer à la volée des *CDRF* modélisant cette non-colinéarité, en plus de la géométrie du détecteur et de la diffusion des photons dans le détecteur.

Chapitre 4

Étude comparative de matrices système précalculées et calculées à la volée

4.1 Introduction **112**
4.2 Méthode *S(MC)²PET* basée sur un matrice système stockée **112**
4.2.1 Estimation de la matrice système par Simulation Monte-Carlo 112
4.2.2 Estimation des coefficients de la matrice système et stockage creux 115
4.3 Étude d'évaluation **117**
4.3.1 Modèles de la réponse du système 118
4.3.2 Fantôme Jaszczak 118
4.3.3 Fantôme Derenzo 118
4.3.4 Fantôme NEMA NU-4 2008 119
4.3.5 Reconstructions 119
4.3.6 Facteurs de mérite 120
4.3.7 Résultats 121
4.3.7.1 Contraste et bruit 121
4.3.7.2 Résolution 123
4.3.7.3 Temps de reconstruction 124
4.4 Discussion et conclusion **126**

4.1 Introduction

Les travaux présentés dans ce chapitre ont été réalisés en collaboration avec Matthieu Moreau du centre de recherche en cancérologie Nantes-Angers (INSERM UMR 982) et Thomas Carlier du centre hospitalier universitaire de Nantes, qui ont développé la reconstruction modélisant la réponse du système avec une matrice stockée, qui est évalué dans ce chapitre.

Dans le chapitre précédent, nous avons proposé le projecteur *IRIS*, permettant d'estimer à la volée les coefficients de la matrice système et modélisant l'intégralité des effets liés au détecteur. Nous avons pu voir que ce type de projecteurs, où les coefficients de la matrice système sont calculés à la volée, coexiste avec une autre approche, où la matrice système est préestimée puis stockée pour être ensuite utilisée au moment de la reconstruction. Avec ce second type d'approche, il est possible d'envisager d'utiliser des méthodes plus précises et complexes, qui peuvent nécessiter des temps de calcul importants, parce que l'estimation de la matrice système est faite une seule fois, séparément de la reconstruction. Cependant, ces méthodes posent des problèmes de stockage et de temps de lecture de la matrice système, à cause de sa taille colossale, ainsi que des problèmes de flexibilité. L'utilisation d'un projecteur pendant la reconstruction permet de corriger ces problèmes, mais cela implique de trouver des méthodes qui soient à la fois précises, pour obtenir une bonne qualité d'image, et rapides, pour que les reconstructions puissent s'exécuter dans des temps raisonnables pour une application clinique. Précédemment, nous avons montré que les projecteurs *IRIS* permettent d'obtenir des reconstructions dont la résolution et le contraste sont autant, voire plus élevés qu'avec les projecteurs à la volée de l'état de l'art, tout en conservant des temps de reconstruction compatibles avec les applications en routine clinique. Dans ce chapitre, nous allons voir une étude comparative des projecteurs *IRIS* avec la méthode proposée par [Matthieu, 2014] qui se fonde sur une matrice système préestimée par SMC et ensuite stockée. Cette étude s'appuie sur un modèle du scanner préclinique INVEON de Siemens.

4.2 Méthode *S(MC)^2 PET* basée sur un matrice système stockée

4.2.1 Estimation de la matrice système par Simulation Monte-Carlo

La méthode appelée *system Matrix Computation by Monte Carlo simulations in PET* (*S(MC)^2 PET*) et proposée par [Matthieu, 2014], utilise une SMC dite complète pour estimer les coefficients de la matrice système. C'est-à-dire que l'ensemble du détecteur est modélisé dans une SMC intégrant tous les effets physiques intervenant pendant le processus d'acquisition. Cette méthode de calcul permet de construire une matrice système qui modélise très précisément l'ensemble des effets associés au détecteur, comme les effets de parallaxes et les diffusions intercristaux et intracristal. Cependant, la précision de cette estimation est directement liée à celles des modèles du détecteur et de la chaîne de détection, ainsi que des modèles des effets physiques utilisés dans la simulation.

Le principe de cette estimation repose sur la simulation d'une source homogène, de positons ou de paire de photons *back-to-back*, remplissant l'ensemble du champ de vue du scanner, voxélisée avec la taille de *voxel* que l'on souhaite utiliser pour les reconstructions, et d'un milieu objet vide ou

non, suivant ce que l'on souhaite modéliser dans cette matrice. Pour chaque *LOR i* détecté, connaissant le *voxel j* d'émission du positon ou de la paire de photons γ *back-to-back*, le coefficient a_{ij} de la matrice système est incrémentée de 1.

La qualité de la matrice système estimée dépend de deux paramètres. Premièrement, sa sophistication qui traduit la quantité de phénomènes physiques inclus dans la SMC. Le second paramètre est sa qualité statistique, qui traduit la variance de chaque élément de cette matrice qui dépend de la quantité de coïncidences détectées, proportionnelle au nombre d'événements simulés. Une estimation non bruitée nécessiterait théoriquement un nombre infini d'événements simulés, ce qui n'est évidemment pas possible en pratique. La vitesse de convergence de l'estimation est en $\frac{1}{\sqrt{n}}$, n étant le nombre d'événements simulés. En pratique, il est difficile de déterminer le nombre d'événements minimal qu'il est nécessaire de simuler pour obtenir une estimation de la matrice système avec une bonne qualité statistique.

On distingue généralement trois niveaux de sophistication de la matrice système. Dans le premier, seul le détecteur est modélisé, dans le second en plus du détecteur, le parcours du positon et la non-colinéarité des photons d'annihilation sont modélisés, le dernier modèle intègre en plus l'atténuation, la diffusion et peut aussi intégrer les coïncidences fortuites et le temps-mort. Le premier niveau de sophistication a été développé, en TEP 2D par [Veklerov *et al.*, 1988] et en TEP 3D par [Rafecas *et al.*, 2004, Cabello et Rafecas, 2012]. On peut trouver dans [Ortuno *et al.*, 2010] une estimation de la matrice système avec le second niveau de sophistication. Des matrices systèmes utilisant le troisième niveau de sophistication ont été développées dans [Yao *et al.*, 2012] avec un milieu objet homogène et dans [Matthieu, 2014] avec un milieu objet hétérogène. Cette seconde matrice système a été développée dans le but de corriger les nombreux phénomènes physiques liés à l'imagerie TEP à l'iode 124. Avec le troisième niveau de sophistication, en raison de la diffusion et des coïncidences fortuites, un événement émis de n'importe quel *voxel* peut être détecté suivant quasiment n'importe quelle *LOR*. Par conséquent, une telle matrice système n'est pas creuse, ce qui pose d'importants problèmes de stockage. Les méthodes de second et troisième niveaux de sophistication utilisant un milieu objet hétérogène ont l'avantage de fournir une estimation de la matrice système adaptée à l'objet imagé, mais cela nécessite aussi de construire à chaque fois cette nouvelle matrice système. Cependant, la SMC servant à estimer cette matrice nécessite typiquement plusieurs jours d'exécution sur un *cluster* de plusieurs centaines de cœurs, ce qui n'est pas compatible avec une application dans un contexte clinique.

Ici, nous ne nous intéressons qu'à modéliser la matrice qui modélise la réponse du détecteur, c'est-à-dire une matrice système de premier niveau de sophistication. Pour cela, la simulation utilise un milieu objet vide, pour éviter toute diffusion et atténuation, ainsi que le mode d'émission de photons *back-to-back*, pour s'affranchir du parcours du positon. On suppose que les effets intervenant au sein de l'objet peuvent être modélisés par d'autres approches. De plus, nous considérons que nous nous trouvons dans le domaine linéaire du détecteur, c'est-à-dire sans temps-mort et sans coïncidences fortuites, ce qui implique que la source utilisée pour la simulation émette suffisamment lentement pour éviter tout phénomène de saturation. On suppose que ces effets peuvent être précorrigés sur les données acquises.

Comme nous l'avons vu, la qualité statistique de l'estimation de la matrice système est liée au

nombre d'événements simulés. Étant donné le nombre important d'éléments dans la matrice et la convergence lente des méthodes de Monte-Carlo, l'estimation d'une matrice système ayant une bonne qualité statistique nécessite de simuler un grand nombre d'événements, ce qui implique des temps de simulation très importants. Afin de réduire ceux-ci, il est possible d'exploiter les symétries du détecteur se combinant avec les symétries d'échantillonnage spatial du champ de vue.

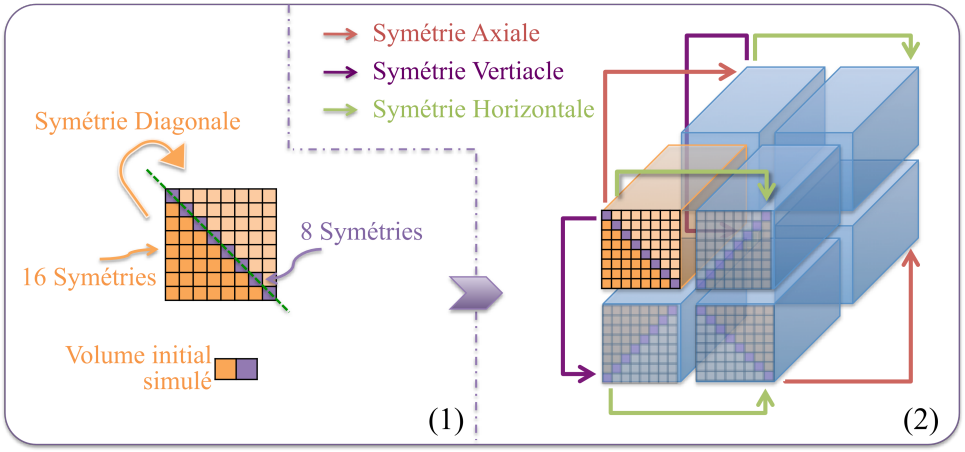


FIGURE 4.1 – Schéma des symétries de l'échantillonnage du champ de vue. Le champ de vue est échantillonné par un volume parallélépipédique voxélisé centré dans le champ de vue. On peut trouver 4 symétries qui en se combinant fournissent 16 symétries pour tous les *voxels* à l'exception des *voxels* diagonaux qui n'en admettent que 8. [Matthieu, 2014]

Dans cette étude, nous nous basons sur le scanner INVEON de Siemens, composé de secteurs de cristaux répétés 16 fois circulairement. On considère que deux *LOR* sont identiques s'il existe une transformation qui superpose ces deux *LOR* et qui ne modifie pas le détecteur. Avec la géométrie de ce détecteur, nous avons 16 secteurs angulaires identiques, ce qui donne donc une symétrie par rotation d'un seizième de tour. Le détecteur est symétrique par rapport à n'importe quel plan passant par le centre d'un secteur et l'axe central du détecteur, ce qui nous amène à 32 symétries. À cela, on peut ajouter la symétrie par rapport au plan perpendiculaire à l'axe du scanner passant en son centre, ce qui nous amène à 64 symétries. Pour pouvoir être exploitée, chacune de ces symétries doit se coupler avec une symétrie de l'échantillonnage spatial du champ de vue. Comme pour les *LOR*, deux *voxels* sont identiques s'il existe une transformation qui permet de les superposer sans modifier la grille d'échantillonnage. L'échantillonnage du champ de vue possède quatre symétries, représentées dans la figure 4.1. Trois de ces symétries se construisent par rapport aux plans centraux perpendiculaires à chacun des trois axes du volume, et une quatrième symétrie se construit par rapport au plan diagonal au volume et passant par son centre, ce qui nous donne un total de 16 symétries en les combinant. Ceci n'est valable que parce que le milieu objet est homogène (vide), ce qui ne serait pas forcément le cas avec un milieu hétérogène. Ces 16 symétries sont compatibles avec celles du détecteur, comme on peut le voir sur la figure 4.2.

Afin d'exploiter ces symétries dans la SMC, le volume source des paires de photons *back-to-back* n'occupe qu'un seizième du champ de vue, comme illustré dans la figure 4.3. L'ensemble des *voxels* de cette source contiennent la même activité. Le fantôme, quant à lui, est vide parce que la matrice estimée ne tient pas compte des effets liés au milieu objet.

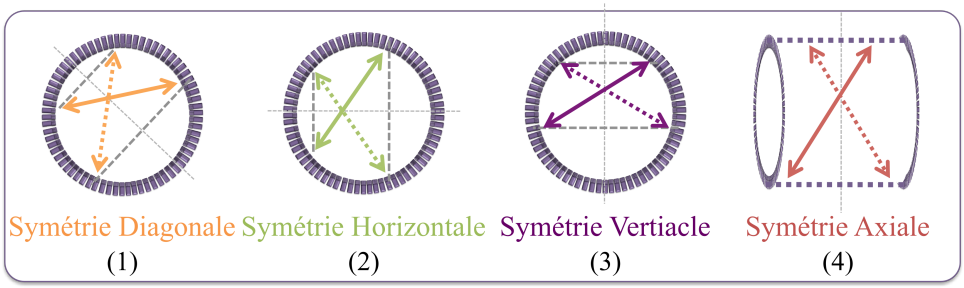


FIGURE 4.2 – Représentation des quatre symétries se combinant à la fois avec celles de l'échantillonnage du champ de vue et avec celles de la géométrie du détecteur de l'INVEON. [Matthieu, 2014]

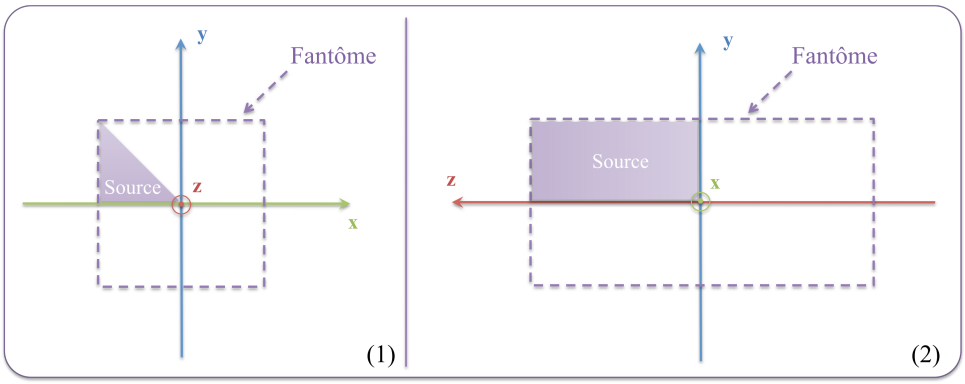


FIGURE 4.3 – Volume source utilisé dans la SMC pour estimer la matrice système, n'occupant qu'un seizième de champ de vue grâce aux symétries. Le fantôme est vide dans notre cas. [Matthieu, 2014]

4.2.2 Estimation des coefficients de la matrice système et stockage creux

Le fichier *list-mode* issu de la simulation est dans un premier temps traité afin de la nettoyer des éventuelles coïncidences fortuites. Chaque *LOR* est associée à un indice spécifique calculé à partir de l'indice des deux cristaux défini par anneau et indice d'anneau, comme représenté dans la figure 4.4. Si on note C_1 l'indice du premier cristal dans l'anneau A_1 et C_2 l'indice du second cristal dans l'anneau A_2, leurs indices sont calculés de la manière suivante :

$$\begin{aligned} L_1 &= C_1 + A_1 N_{anneau} \\ L_2 &= C_2 + A_2 N_{anneau} \end{aligned} \tag{4.1}$$

où N_{anneau} est le nombre de cristal par anneau. L'indice globale i de la *LOR* est quant à lui calculé de la manière suivante :

$$i = max(L_1, L_2) + min(L_1, L_2) N_{cristaux} \tag{4.2}$$

où $N_{cristaux}$ est le nombre total de cristaux dans le détecteur. Les fonctions max et min retournent la valeur maximale et minimale des valeurs données et elles permettent ici d'éviter de dupliquer les *LOR* identiques, celle joignant L_1 à L_2 et celle joignant L_2 à L_1.

Pour chaque coïncidence, à partir de l'indice i de la *LOR* et connaissant l'indice j du *voxel* source de la paire de photons d'annihilation, la probabilité a_{ij} de la matrice système est incrémentée de la valeur $\frac{1}{N_{coïncidences}}$, où $N_{coïncidences}$ est le nombre de coïncidences dans le jeu de données *list-mode*.

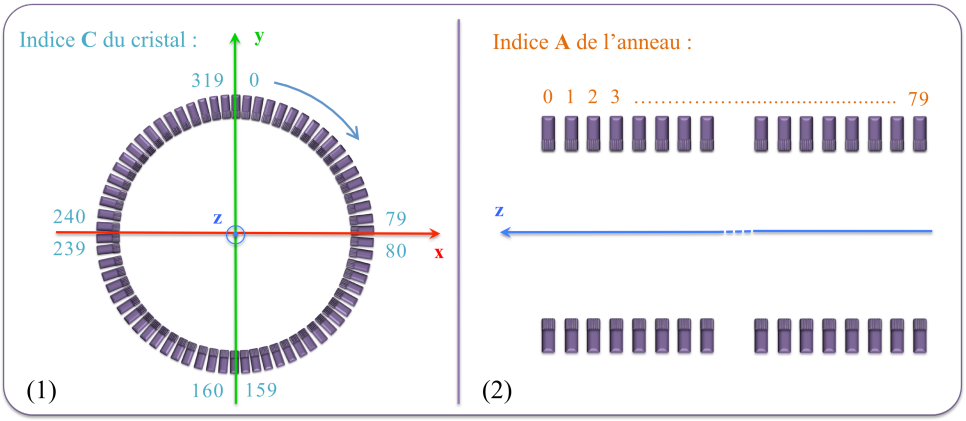


FIGURE 4.4 – Indexation des *LOR*. [Matthieu, 2014]

Évidemment, étant donné la forme de la source, seule un seizième de la matrice système est calculée, mais en exploitant les symétries, tous les coefficients manquants peuvent être retrouvés dans ce seizième de matrice.

L'approche *S(MC)*^{2}*PET* exploite une méthode de stockage de matrice creuse, introduite par [Lazaro *et al.*, 2005] pour la reconstruction en TEMP, pour stocker la matrice système. Avec cette méthode, la matrice se remplit au fur et à mesure de la lecture du fichier *list-mode*. Ce mode de stockage est schématisé dans la figure 4.5. Cette matrice creuse est constituée d'une liste de cellules, chacune assignée à un *voxel* du champ de vue. Chacune de ces cellules contient à son tour une autre liste et un nombre qui indique la taille de cette liste. Au début de la lecture du fichier *list-mode*, les nombres valent zéro et les listes sont vides. Pour un *voxel* j donné, chaque cellule de sa liste contient deux valeurs, l'indice i d'une *LOR* et le coefficient a_{ij} de la matrice système. Lors de la lecture du fichier *list-mode*, à chaque fois qu'une paire de photons d'annihilation émis du *voxel* j considéré est détectée suivant une *LOR* i, deux traitements sont possibles. Si la liste du *voxel* ne contient pas de cellule associée à la *LOR* d'indice i, une nouvelle cellule est ajoutée à sa liste, la valeur de a_{ij} est initialisée avec la valeur $\frac{1}{N_{coïncidences}}$ et le nombre indiquant la taille de la liste est incrémenté de un. Sinon, si la liste contient déjà une cellule pour cette *LOR*, la valeur a_{ij} de cette cellule est incrémentée de la valeur $\frac{1}{N_{coïncidences}}$.

Le facteur de compression apporté par ce mode de stockage, par rapport au stockage brut de la matrice système, est inversement proportionnel au nombre d'éléments non-nuls présents dans la matrice.

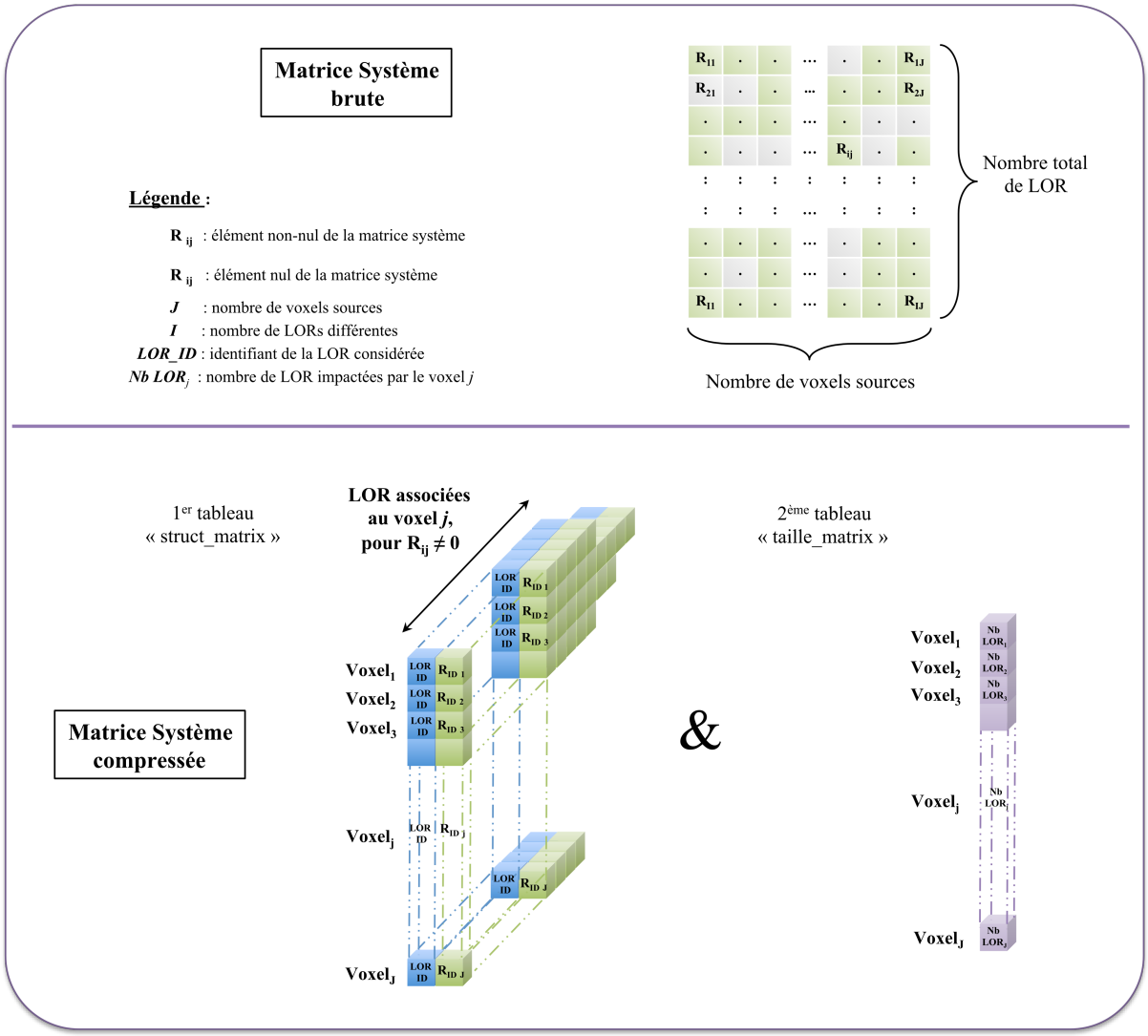


FIGURE 4.5 – Schématisation de la méthode de stockage de matrice creuse exploitée avec la méthode *S(MC)*^{2} *PET*. [Matthieu, 2014]

4.3 Étude d'évaluation

Dans cette section nous avons effectué une étude de la méthode *S(MC)*^{2} *PET* face à des projecteurs afin d'évaluer les caractéristiques (contraste, résolution) des images reconstruites avec ces différentes approches et de comparer leurs temps de reconstruction. Les projecteurs utilisés dans cette étude sont les deux projecteurs *IRIS* qui modélisent l'ensemble des effets physiques et géométriques de la réponse du détecteur comme la méthode *S(MC)*^{2} *PET*, le projecteur de Siddon, qui ne modélise quant à lui aucun effet, et un projecteur Gaussien$_{constant}$ qui modélise la *PSF* du détecteur au centre du champ de vue.

L'ensemble de cette étude se base sur des données obtenues par SMC effectuées dans la plateforme *GATE* avec un modèle du scanner préclinique INVEON de siemens, validé par [Anizan, 2010, Matthieu, 2014]. Les processus physiques qui ont été simulés sont, l'effet photoélectrique, la diffusion Compton et l'électroionisation modélisés avec le modèle standard, ainsi que l'effet Rayleigh, modé-

lisé avec le modèle Penelope.

4.3.1 Modèles de la réponse du système

La matrice système de l'approche *S(MC)²PET* a été estimée avec trois niveaux de qualité statistique, que l'on nommera $S(MC)^2PET_{low}$, $S(MC)^2PET_{medium}$ et $S(MC)^2PET_{high}$, respectivement pour la plus faible statistique, la statistique moyenne et la statistique la plus élevée. La matrice $S(MC)^2PET_{low}$ a été construite avec un jeu de données de $5,68 \times 10^9$ coïncidences vraies, $S(MC)^2PET_{medium}$ avec $1,14 \times 10^{10}$ coïncidences vraies et $S(MC)^2PET_{high}$ avec $2,34 \times 10^{10}$ coïncidences vraies. Les SMC qui ont servi à générer les données utilisées pour construire les matrices systèmes ont nécessité 24, 34 et 60 heures de calcul, respectivement pour $S(MC)^2PET_{low}$, $S(MC)^2PET_{medium}$ et $S(MC)^2PET_{high}$, sur un cluster de 336 cœurs, composé d'Intel Xeon E5645 cadencés à 2,4 GHz.

Trois projecteurs ont été utilisés pour modéliser la réponse du système à la volée dans le processus de reconstruction. Le projecteur de Siddon, présenté dans le paragraphe 3.2.1.3.1, le projecteur $\text{Gaussien}_{constant}$, vu dans le paragraphe 3.2.1.3.2 et les projecteurs $IRIS_{mesure}$ et $IRIS_{analytique}$ présentés dans la section 3.3. Le projecteur de Siddon ne modélise aucun effet associé au détecteur, il n'y a donc aucun paramètre spécifique à estimer pour celui-ci. Le projecteur $\text{Gaussien}_{constant}$ a été utilisé avec une *FWHM* de 1,64 mm, mesurée par [Goertzen *et al.*, 2012], qui modélise la *PSF* du détecteur de l'INVEON. Les modèles des projecteurs *IRIS* ont été construits en se basant sur un jeu de données *list-mode* de 10 milliards de coïncidences vraies, avec un échantillonnage des angles (θ, φ) de $7,5° \times 7,5°$ dans un intervalle de $[0°, 45°] \times [0°, 45°]$. La SMC qui a permis de construire les modèles des projecteurs *IRIS* a nécessité 27 heures de simulations sur un cluster 336 cœurs composé d'Intel Xeon E5645 cadencés à 2,4 GHz. Avec les deux projecteurs *IRIS*, pour chaque *LOR*, 16 lignes aléatoires ont été tracées pour estimer la *CDRF*.

4.3.2 Fantôme Jaszczak

Le fantôme Jaszczak a été conçu pour étudier les CRC. Il est composé d'un cylindre d'eau de 30 mm de diamètre et 20 mm de long avec une activité que l'on définit comme étant celle du fond. Dans ce cylindre sont placés quatre cylindres d'eau de 4 mm de diamètre et 20 mm de long, avec des niveaux d'activité différents, pour faire varier le contraste par rapport au fond. Dans les quatre inserts cylindriques, un d'eux a une valeur d'activité nulle, tandis que les autres ont des activités deux fois, trois fois et quatre fois plus importantes que dans le fond. Ce fantôme est schématisé dans la figure 4.6. Ce fantôme a été simulé dans le modèle du scanner INVEON sous *GATE*. Le jeu de données *list-mode* obtenu, composé de $1,9 \times 10^8$ de coïncidences promptes, a été converti en mode histogramme.

4.3.3 Fantôme Derenzo

Afin d'évaluer les performances en matière de résolution, nous avons utilisé un fantôme du type Derenzo, constitué d'un corps cylindrique de plexiglas de 30 mm de diamètre, perforé par six groupes de cylindres avec des diamètres allant de 1,6 mm à 2,6 mm. Chacun de ces cylindres est séparé de ses voisins d'une distance au moins égale à son diamètre. Tous les trous cylindriques sont remplis d'eau ayant la même concentration d'activité. Ce fantôme est schématisé dans la figure 4.7. Nous avons

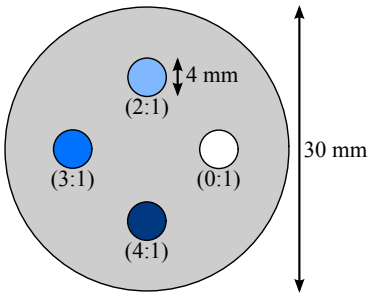


FIGURE 4.6 – Fantôme Jaszczak utilisé pour l'étude du recouvrement du contraste.

simulé ce fantôme et avons obtenu un jeu de données en mode histogramme construit avec $1,9 \times 10^8$ coïncidences promptes détectées.

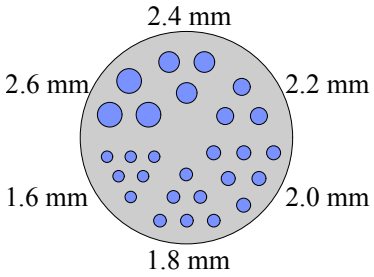


FIGURE 4.7 – Schéma d'une coupe transverse du fantôme Derenzo utilisé pour évaluer la résolution.

4.3.4 Fantôme NEMA NU-4 2008

Enfin, nous avons utilisé un fantôme NEMA préclinique pour évaluer la qualité des images reconstruites. Ce fantôme est composé en deux parties. D'un côté, c'est un cylindre de 33,5 mm de diamètre rempli d'eau active dans lequel baignent, sur la moitié de sa longueur, deux cylindres non-actifs de 8 mm de diamètre, l'un rempli d'eau et l'autre d'air. La seconde partie du fantôme est constituée d'un cylindre de plexiglas de 33,5 mm de diamètre, percé sur sa longueur de cinq trous de 1, 2, 3, 4 et 5 mm de diamètre, rempli de la même eau active que le large cylindre d'eau. Ce fantôme est schématisé dans la figure 4.8. Ce fantôme a été simulé avec le modèle du scanner INVEON, pour obtenir un jeu de données *list-mode* de $1,57 \times 10^8$ coïncidences promptes, utilisé pour construire les projections en mode histogramme.

4.3.5 Reconstructions

L'ensemble des reconstructions ont été effectuées avec l'algorithme *ML-EM*, présenté dans le paragraphe 1.4.2.2, avec des données en mode histogramme. Un champ de vue de reconstruction de $40 \times 40 \times 80$ mm^3, aligné avec le centre du scanner et dont la plus grande dimension correspond à l'axe du scanner, a été utilisé avec des *voxels* de $0.8 \times 0.8 \times 0.8$ mm^3. Aucune correction d'atténuation n'a été effectuée en raison de son faible impact en imagerie TEP préclinique. Toutes les reconstructions ont été effectuées avec 100 itérations et avec les différentes matrices *S(MC)^2 PET* et les trois projecteurs.

Les reconstructions effectuées avec la méthode *S(MC)^2 PET* ont été exécutées avec un cœur d'un

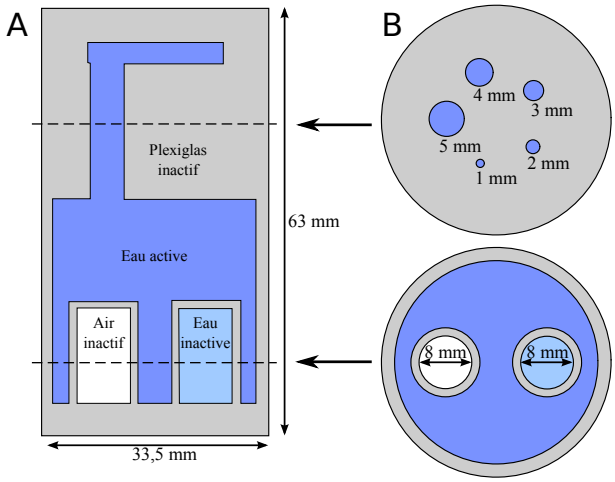


FIGURE 4.8 – Schéma du fantôme NEMA NU-4 2008 utilisé pour évaluer la qualité d'image des scanners précliniques.

CPU Intel Xeon E5645 cadencé à 2.4GHz. Les reconstructions effectuées avec les projecteurs ont toutes été exécutées sur un cœur d'un *CPU* Intel Xeon E5-2680 cadencé à 2.7GHz, ainsi que sur un *GPU* NVIDIA GTX 980 Ti cadencé à 1GHz. Les temps de reconstruction sont donnés par itération, pour les reconstructions du fantôme NEMA.

4.3.6 Facteurs de mérite

À chaque itération le CRC, calculé avec l'équation 3.13, est estimé dans le fantôme Jaszczak, dans les quatre petits cylindres, par rapport à l'activité reconstruite dans le fond. Celui-ci est défini dans quatre régions d'intérêts cylindriques de 4 mm de diamètre aussi longues que le fantôme, placées chacune entre deux des petits cylindres. Les valeurs de CRC sont normalisées par la valeur théorique à atteindre. Le $Bruit_{SD}$ est évalué dans la région d'intérêt définie dans le fond, avec l'équation suivante :

$$Bruit_{SD} = \frac{\sqrt{\overline{(r_b - \overline{r_b})^2}}}{\overline{r_b}} \tag{4.3}$$

où r_b est le vecteur contenant les valeurs des *voxels* du fond.

Le contraste est mesuré dans le fantôme NEMA, dans les cinq petits cylindres, de la manière suivante :

$$Contraste = \frac{\overline{r_c}}{\overline{r_b}} \tag{4.4}$$

où $\overline{r_c}$ est la valeur moyenne des *voxels* dans le petit cylindre et $\overline{r_b}$ la valeur moyenne des *voxels* dans une région d'intérêt de 20 mm de diamètre et de 10 mm de long placée dans le large cylindre d'eau active. Le $Bruit_{SD}$ est aussi estimé dans cette zone.

Sur le fantôme Derenzo, la résolution est évaluée dans un premier temps par une analyse visuelle, en mesurant un profil passant sur les cylindres de 1,6 et 1,8 mm de diamètre, dans l'image reconstruite avec 100 itérations. Dans un second temps, un profil sur le cylindre de 1,6 mm est mesuré sur les images reconstruites à chaque itération et une distribution Gaussienne est ajustée sur chaque pique

du profil (qui correspondent aux cylindres). À une itération donnée, les *FWHM* des Gaussiennes ajustées sont moyennes pour obtenir l'estimation de la résolution. Une procédure similaire a été utilisée pour estimer la résolution avec le fantôme NEMA. Cette fois, c'est une distribution Gaussienne 2D qui a été ajustée sur le cylindre de 1 mm de diamètre dans une coupe des images reconstruites. L'estimation de la résolution est calculée en moyennant les deux *FWHM* de la Gaussienne 2D.

4.3.7 Résultats

4.3.7.1 Contraste et bruit

Les valeurs de CRC mesurées sur le fantôme Jaszczak sont présentées dans la figure 4.9. Sur ces résultats, on peut voir que, quel que soit le taux de contraste original, le projecteur de Siddon fournit le moins bon CRC pour un niveau de $Bruit_{SD}$ donné, il est suivi de la matrice $S(MC)^2 PET_{low}$, à son tour suivi du projecteur $IRIS_{mesure}$, puis de la matrice $S(MC)^2 PET_{medium}$. Les projecteurs $Gaussien_{constant}$ et $IRIS_{analytique}$ donnent quant à eux des résultats similaires. Finalement, la matrice $S(MC)^2 PET_{high}$ surpasse d'environ 5 % les autres méthodes dans tous les cas, sauf pour le cylindre froid, où elle donne un résultat équivalent au projecteur $Gaussien_{constant}$. Le projecteur $IRIS_{mesure}$ donne des résultats légèrement inférieurs à la matrice $S(MC)^2 PET_{medium}$, tandis que le projecteur $IRIS_{analytique}$ surpasse légèrement cette dernière.

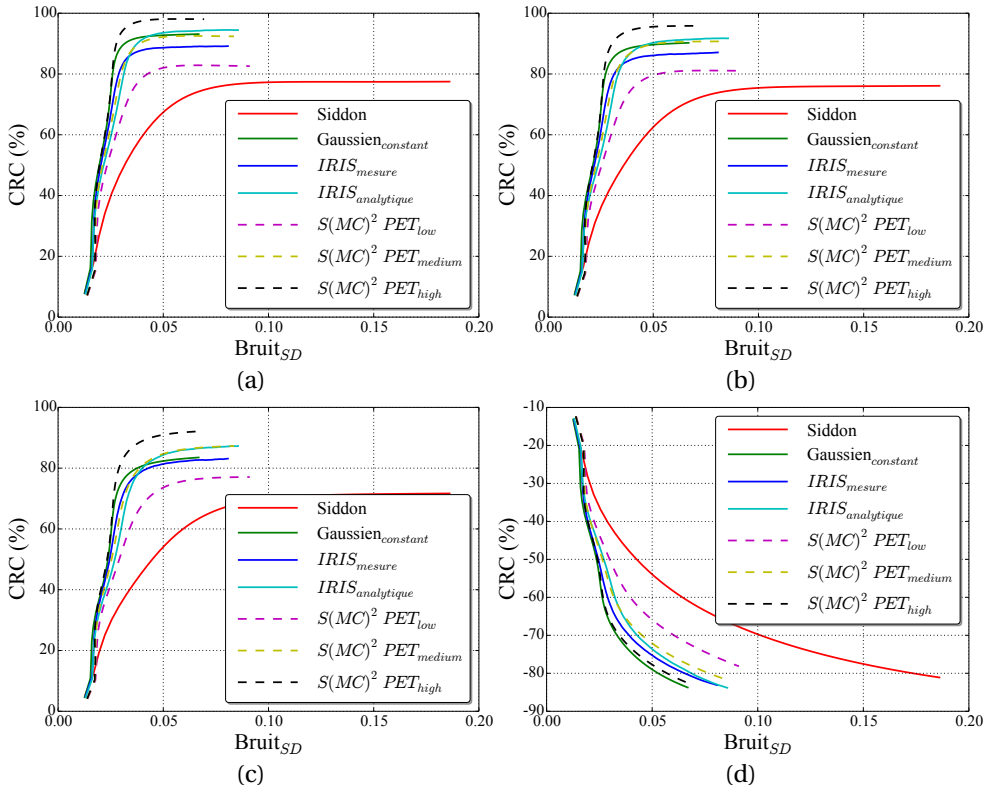
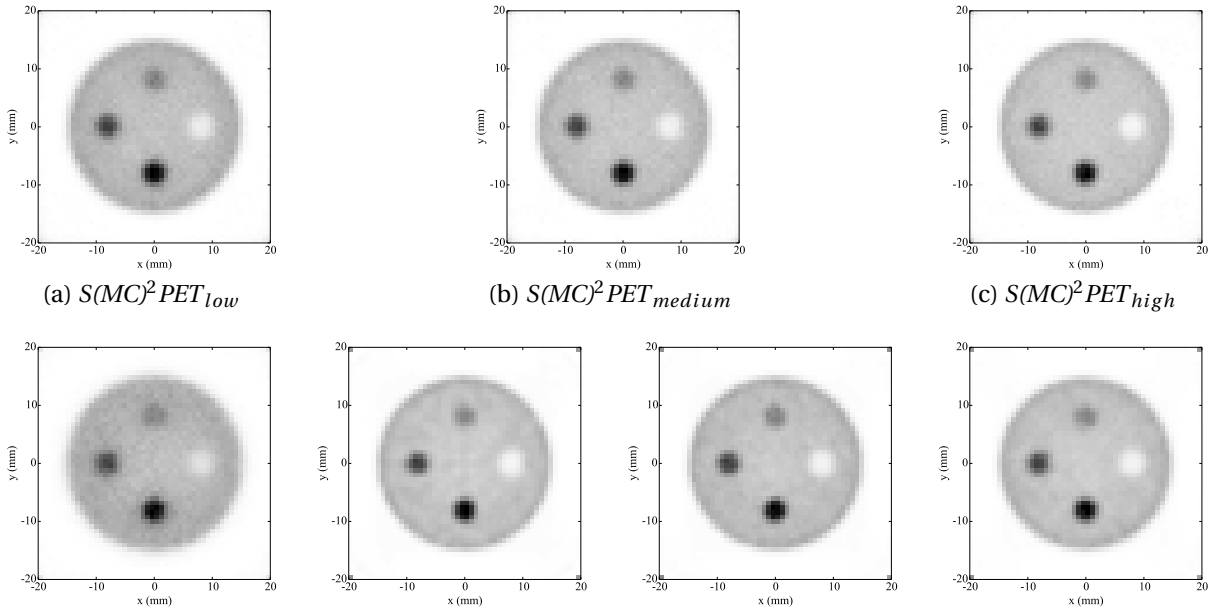


FIGURE 4.9 – CRC en fonction du $Bruit_{SD}$ mesurés dans les reconstructions du fantôme Jaszczak. (a), (b) et (c) donnent ces courbes pour les CRC mesurés dans les cylindre avec des activités respectives quatre, trois et deux fois plus grandes que dans le fond. (d) donne les CRC mesurés dans le cylindre sans activité.

La figure 4.10 montre des coupes des images reconstruites du fantôme Jaszczak, données avec des nombres d'itérations permettant d'atteindre le même niveau de $Bruit_{\mathrm{SD}}$, égale à 0,05. Visuellement, on peut voir que le projecteur de Siddon fournit une image moins contrastée, tandis qu'on ne distingue pas de grandes différences pour les autres méthodes. Avec le projecteur $\mathrm{Gaussien}_{constant}$, on peut constater un léger artefact en forme de croix au centre de l'image.



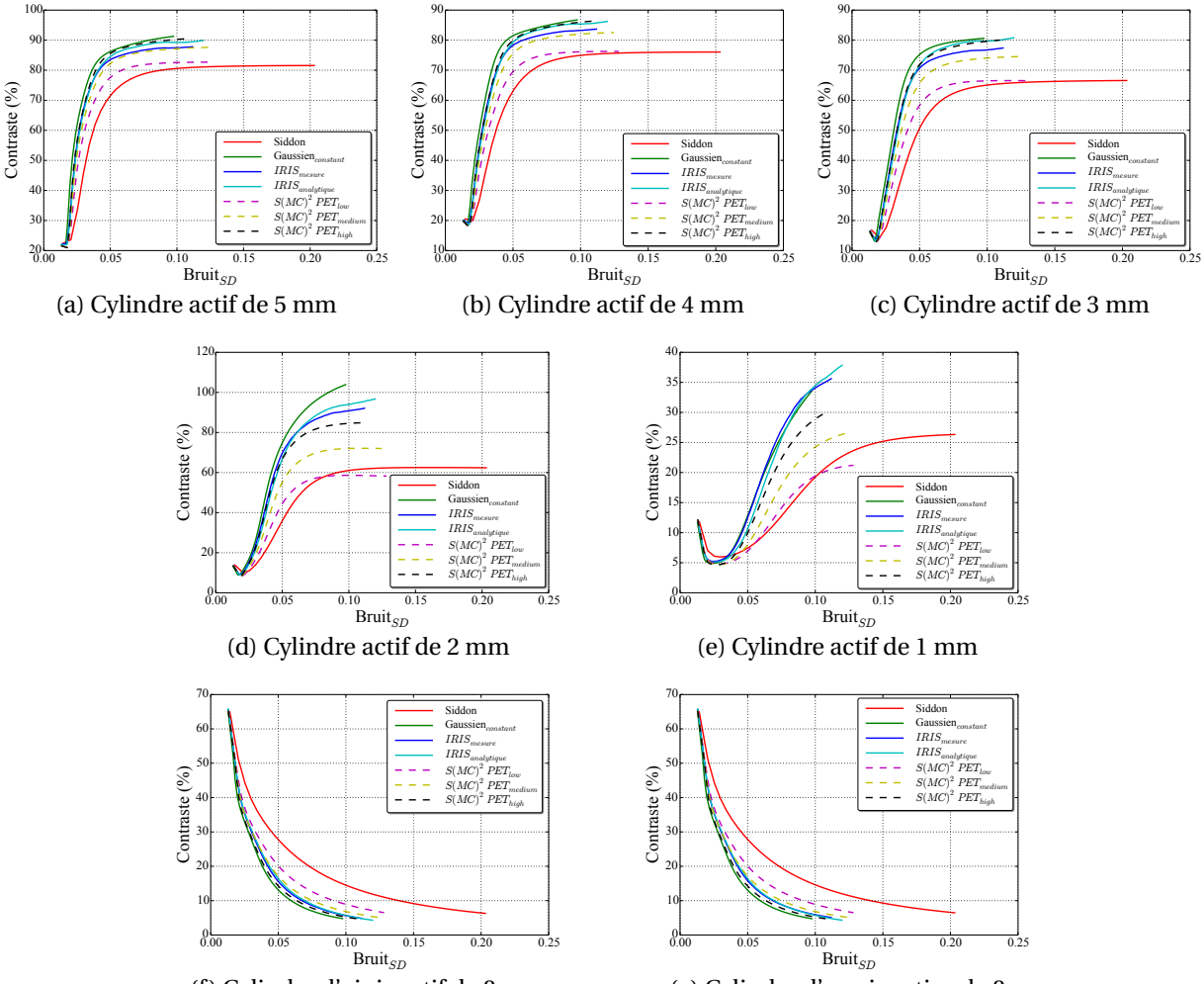
(a) $S(MC)^2PET_{low}$ (b) $S(MC)^2PET_{medium}$ (c) $S(MC)^2PET_{high}$

(d) Siddon (e) $\mathrm{Gaussien}_{constant}$ (f) $IRIS_{mesure}$ (g) $IRIS_{analytique}$

FIGURE 4.10 – Coupes transverses des reconstructions du fantôme Jaszczak pour un niveau de $Bruit_{\mathrm{SD}} = 0,05$. La première ligne montre les reconstructions obtenues avec la méthode $S(MC)^2PET$ tandis que la seconde ligne montre les reconstructions obtenues avec les projecteurs.

Les courbes du contraste en fonction du niveau de $Bruit_{\mathrm{SD}}$ dans le fond, mesuré dans les reconstructions du fantôme NEMA, sont présentées dans la figure 4.11. Avec ce fantôme, le projecteur Siddon fourni toujours les moins bons contrastes, suivi de près par la matrice $S(MC)^2PET_{low}$. Avec les cylindres de 3 mm de diamètre et plus, toutes les autres méthodes fournissent des résultats très similaires. Pour les deux cylindres les plus petits, les projecteurs $\mathrm{Gaussien}_{constant}$, $IRIS_{mesure}$ et $IRIS_{analytique}$ donnent les meilleurs contrastes.

Si on observe les images reconstruites du fantôme NEMA dans la figure 4.12, on retrouve les observations faites sur les courbes du contraste. La matrice $S(MC)^2PET_{low}$ et le projecteur Siddon donnent des images qui sont visuellement très similaires. Le cylindre de 1 mm de diamètre semble être le plus visible sur les images reconstruites avec les projecteurs $\mathrm{Gaussien}_{constant}$ et *IRIS*. Sur l'image 4.12e reconstruite avec le projecteur $\mathrm{Gaussien}_{constant}$, on voit des artefacts de bord au niveau du cylindre de 5 mm de diamètre, similaire à des artefacts de Gibbs, que l'on voit généralement apparaître sur les reconstructions itératives intégrant une modélisation de la *PSF*, comme cela a été constaté dans plusieurs études [Qi *et al.*, 1998, Panin *et al.*, 2006, Ortuno *et al.*, 2006, Bai et Esser, 2010].



(a) Cylindre actif de 5 mm (b) Cylindre actif de 4 mm (c) Cylindre actif de 3 mm

(d) Cylindre actif de 2 mm (e) Cylindre actif de 1 mm

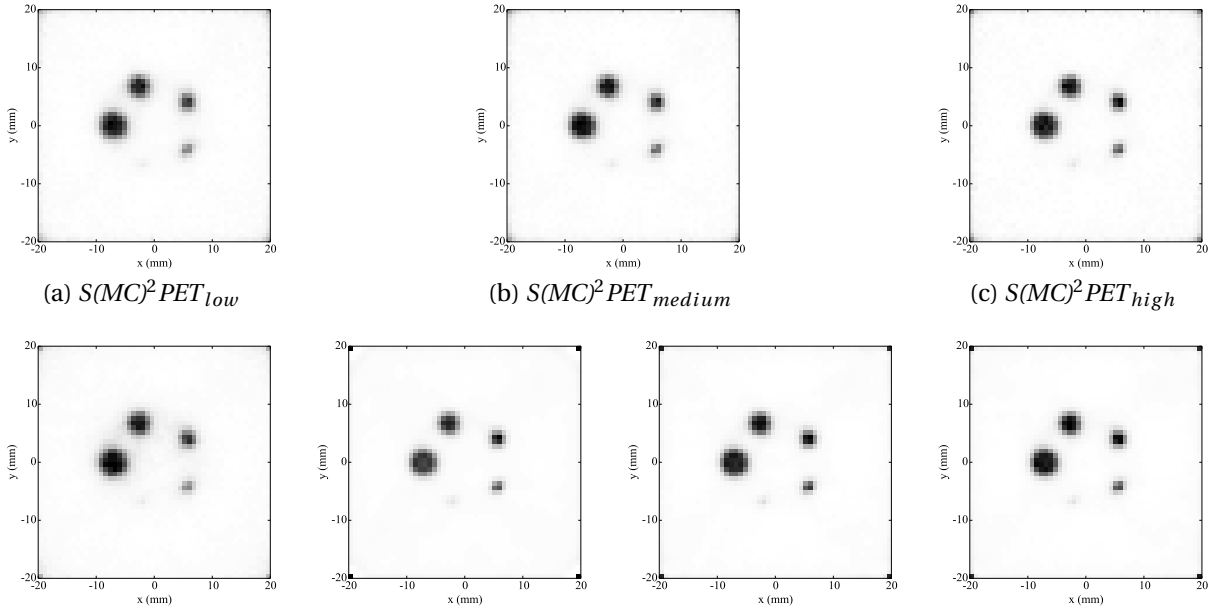
(f) Cylindre d'air inactif de 8 mm (g) Cylindre d'eau inactive de 8 mm

FIGURE 4.11 – Contraste en fonction du B$ruit_{\mathrm{SD}}$, mesurés dans les reconstructions du fantôme NEMA.

4.3.7.2 Résolution

La figure 4.13 donne l'évolution de la largeur des distributions Gaussiennes ajustées sur les images reconstruites des fantômes Derenzo 4.13a et NEMA 4.13b. Avec les deux fantômes, le projecteur de Siddon converge le plus vite, ce qui lui permet de surpasser toutes les autres méthodes pendant les 30 premières itérations. Avec le fantôme Derenzo, il est finalement rattrapé par tous les projecteurs et la matrice $S(MC)^2 PET_{low}$ après les 30 itérations. Avec le fantôme NEMA, il est rattrapé seulement par les deux projecteurs *IRIS*. Les deux projecteurs *IRIS* fournissent des résultats quasiment identiques et surpassent les autres méthodes dès les premières itérations, sauf le projecteur de Siddon, comme nous venons de le noter. Les projecteurs *IRIS* convergent vers une *FWHM* de 1,2 mm tandis que le projecteur $S(MC)^2 PET_{high}$ converge plutôt vers une *FWHM* de 1,5 mm.

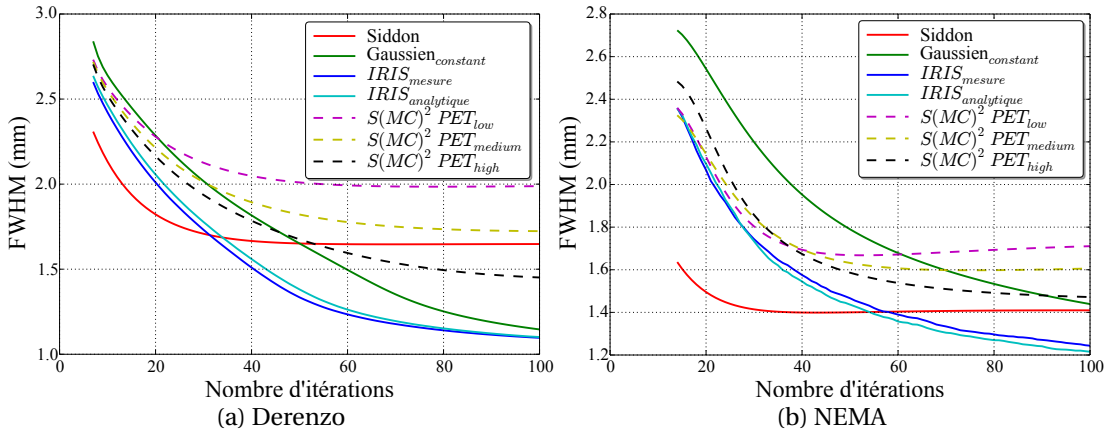
La figure 4.14 montre une coupe transverse du fantôme Derenzo reconstruit, à la centième itération. Les images données par les projeteurs $S(MC)^2 PET$ avec les deux plus faibles statistiques ont les cylindres de 1,6 mm qui sont les moins séparables visuellement. À l'inverse, ces cylindres sont les mieux séparés avec les projecteurs *IRIS*. Cependant, avec le projecteur Gaussien$_{constant}$, les valeurs



(a) $S(MC)^2\,PET_{low}$ (b) $S(MC)^2\,PET_{medium}$ (c) $S(MC)^2\,PET_{high}$

(d) Siddon (e) $\text{Gaussien}_{constant}$ (f) $IRIS_{mesure}$ (g) $IRIS_{analytique}$

FIGURE 4.12 – Coupes transverses du fantôme NEMA au niveau des petits cylindres chauds pour l'image du haut et au niveau des cylindres froids placés dans un fond actif pour l'image du bas. Ces coupes ont été extraites des volumes reconstruits avec un nombre d'itérations permettant d'atteindre le même niveau de $Bruit_{\text{SD}}$ = 0.08.



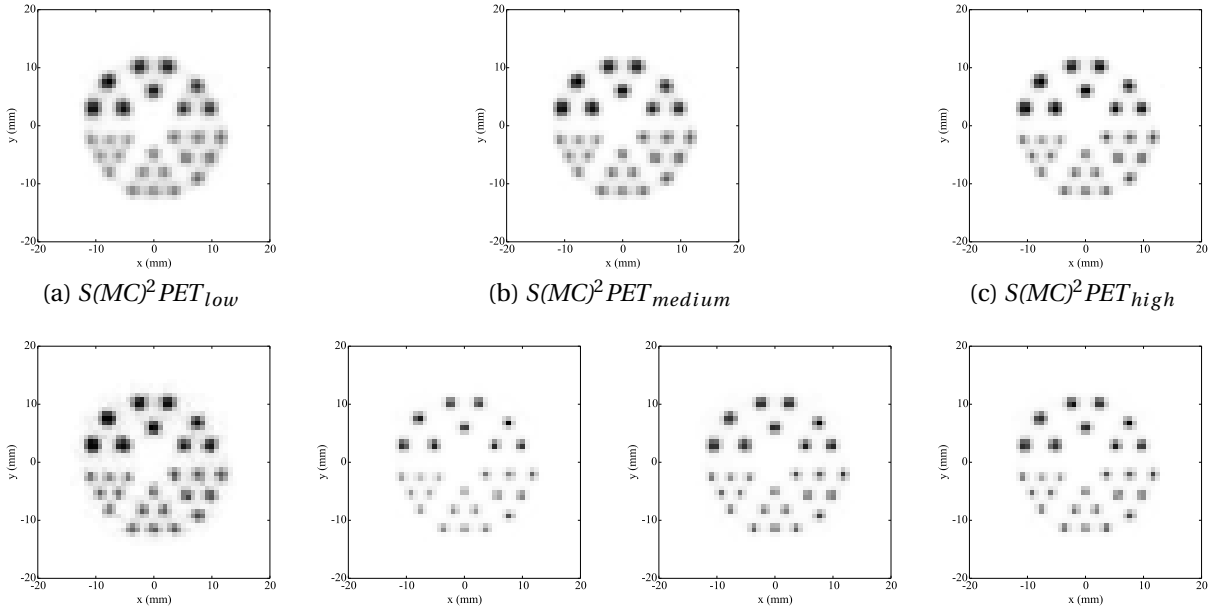
(a) Derenzo (b) NEMA

FIGURE 4.13 – Mesures de la résolution sur les fantômes Derenzo (a) et NEMA (b).

maximales des *voxels* dans les cylindres de petits diamètres sont particulièrement faibles par rapport à celles dans les cylindres de plus grands diamètres, malgré qu'ils soient bien séparables.

4.3.7.3 Temps de reconstruction

Les temps de reconstruction, donnés pour une itération, associés aux méthodes évaluées dans ce chapitre sont présentés dans le tableau 4.1. Les projecteurs ayant été implémentés sur *CPU* et sur *GPU*, les deux temps sont indiqués. Sur *CPU*, le projecteur de Siddon reste le plus rapide. Ensuite, on trouve les projecteurs *IRIS* et la matrice $S(MC)^2\,PET_{low}$ qui s'exécutent en environ 11 minutes, un temps quatre fois plus long qu'avec le projecteur de Siddon. La matrice $S(MC)^2\,PET_{medium}$ s'exécute



(a) $S(MC)^2PET_{low}$ (b) $S(MC)^2PET_{medium}$ (c) $S(MC)^2PET_{high}$

(d) Siddon (e) $\text{Gaussien}_{constant}$ (f) $IRIS_{mesure}$ (g) $IRIS_{analytique}$

FIGURE 4.14 – Coupes transverses des images reconstruites, du fantôme Derenzo, à la centième itération.

lui en deux fois plus de temps, 20 minutes, et les projecteurs $\text{Gaussien}_{constant}$ et $S(MC)^2PET_{high}$, les plus lents, s'exécutent en environ une demi-heure. L'implémentation *GPU* permet d'accélérer 200 fois le projecteur de Siddon, 130 fois le projecteur $\text{Gaussien}_{constant}$, 150 fois le projecteur $IRIS_{mesure}$ et 170 fois le projecteur $IRIS_{analytique}$.

TABLEAU 4.1 – Temps de reconstruction du fantôme NEMA, par itération.

Modèle	*CPU*	*GPU*
Siddon	2,6 minutes	0,76 secondes
$\text{Gaussien}_{constant}$	30,4 minutes	14 secondes
$IRIS_{mesure}$	10,3 minutes	4,1 secondes
$IRIS_{analytique}$	11,0 minutes	3,8 secondes
$S(MC)^2PET_{low}$	11 minutes	
$S(MC)^2PET_{medium}$	20 minutes	
$S(MC)^2PET_{high}$	35 minutes	

Méthode $S(MC)^2PET$: CPU Intel Xeon E5645 à 2.4GHz.
Projecteurs : CPU Intel Xeon E5-2680 à 2.7GHz et GPU NVIDIA GTX 980 Ti à 1GHz

Avec un seul cœur de *CPU*, aucune méthode ne permet d'obtenir des temps de reconstruction suffisamment courts pour être utilisable dans un contexte clinique. En effet, il est nécessaire d'effectuer plusieurs itérations avant d'obtenir des images exploitables. La méthode la plus rapide nécessite 2,6 minutes par itération. Une reconstruction complète exigerait plusieurs dizaines de minutes. En utilisant un *GPU* les temps de reconstruction sont drastiquement réduits. Les projecteurs de Siddon

et *IRIS* permettraient d'obtenir une reconstruction en seulement quelques minutes, tandis que le projecteur Gaussien$_{constant}$ aurait besoin d'un temps de l'ordre de la dizaine de minutes, ce qui reste envisageable pour une application clinique.

4.4 Discussion et conclusion

Dans ce chapitre nous avons comparé des projecteurs calculant à la volée les coefficients de la matrice système avec la méthode *S(MC)*^{2}*PET*, basée sur une matrice stockée, préestimée par une SMC complète.

TABLEAU 4.2 – Quantités de mémoire et temps de simulation nécessaires aux différents modèles.

Méthode	Taille	Temps de simulation
Siddon	0	
Gaussien$_{constant}$	8 octets	
$IRIS_{mesure}$	6,87 Megaoctets	27 heures
$IRIS_{analytique}$	1008 octets	27 heures
$S(MC)^2PET_{low}$	2,5 Gigaoctets	24 heures
$S(MC)^2PET_{medium}$	4,6 Gigaoctets	34 heures
$S(MC)^2PET_{high}$	8,0 Gigaoctets	60 heures

Simulations effectuées sur un cluster de calcul de 336 cœurs composé de Intel Xeon E5645 cadencés à 2,4 GHz.

La qualité statistique de la matrice utilisée avec la méthode *S(MC)*^{2}*PET* a un impact important sur la qualité des images reconstruites. La matrice $S(MC)^2PET_{low}$, avec la variance la plus élevée, donne des valeurs de contrastes et de résolutions jusqu'à 20 % inférieures que celles obtenues avec la matrice $S(MC)^2PET_{high}$, avec la variance la plus faible. Ce gain se fait cependant au prix d'un temps de reconstruction plus important. En effet, la matrice $S(MC)^2PET_{low}$ possède bien plus de coefficients nuls que $S(MC)^2PET_{high}$, ce qui implique de traiter plus de *LOR* pour chaque *voxel*, et donc un temps plus long pour effectuer les projections et les rétroprojections. La réduction du nombre de coefficients nuls implique aussi une réduction de l'aspect creux de la matrice système, qui entraîne à son tour une augmentation de l'espace mémoire nécessaire à son stockage. Le tableau 4.2 indique la taille mémoire de chacun des modèles des méthodes comparées dans cette étude. On peut évidemment constater que les projecteurs nécessitent bien moins d'espace mémoire. Le projecteur de Siddon n'a aucun paramètre, il nécessite donc un espace mémoire nul. Le projecteur Gaussien$_{constant}$, qui lui se base sur une modélisation des coupes des *CDRF* par une distribution Gaussienne 2D qui ne varie pas dans le champ de vue, est entièrement défini par seulement deux paramètres. Ces deux paramètres sont déterminés par des mesures expérimentales, effectuées par [Goertzen *et al.*, 2012]. Le projecteur $IRIS_{mesure}$ utilise des *IDRF*, estimées par une SMC de 27 heures, stockées dans 6×6 volumes de $50 \times 50 \times 20$ *voxels*, ce qui nécessite au total 6,87 Mo de stockage. Le projecteur $IRIS_{analytique}$ modélise chaque *IDRF* avec 7 paramètres, ceux-ci étant dérivés des estimations des $IDRF_{3D}$ utilisées par

le projecteur *IRIS* précédent. L'espace mémoire nécessaire est donc celui occupé par 7 paramètres (codés sur 4 octets) pour chacune des 36 *IDRF* modélisées, soit 1008 octets. Ces faibles quantités de mémoire permettent d'envisager des implémentations *GPU*, où la mémoire est généralement limitée. Par exemple, le *GPU* GTX 980 Ti que nous avons utilisé dans cette étude est un modèle haut de gamme qui contient 6 Go de mémoire, ce qui est supérieur à ce que l'on trouve sur la majeure partie des *GPU* disponibles actuellement. Cette taille ne serait pas suffisante pour héberger la matrice $S(MC)^2PET_{high}$, qui nécessite 8 Go.

Avec la méthode $S(MC)^2PET$, pour les trois variances comparées, si tous les éléments de la matrice système étaient estimés, la taille mémoire de la matrice ne devrait pas augmenter lorsque l'on augmente le nombre d'éléments pour l'estimer. Cependant, on ne constate pas cela avec les trois matrices $S(MC)^2PET$. La matrice $S(MC)^2PET_{medium}$ a une taille qui est $1,84$ fois plus importante celle de $S(MC)^2PET_{low}$, alors qu'elle est construite avec un jeu de données deux fois plus petit que cette première. Ceci nous indique que la plupart des coïncidences supplémentaires échantillonnent des coefficients qui n'étaient pas déjà présents dans la matrice $S(MC)^2PET_{low}$. De même, entre la matrice $S(MC)^2PET_{high}$ et $S(MC)^2PET_{medium}$, l'augmentation de taille est de $1,74$, ce qui montre qu'un grand nombre de coefficients ont encore été ajoutés à la matrice système. On pourrait donc envisager d'augmenter encore le nombre de coïncidences utilisé pour construire la matrice système, ce qui permettrait d'améliorer la qualité des images reconstruites. Cependant, cela impliquerait un temps de SMC bien plus long (la convergence des méthodes de Monte-Carlo étant en $\frac{1}{\sqrt{N}}$, où N correspond au nombre d'événements simulés) ainsi qu'un espace de stockage encore plus important, alors qu'ils sont déjà respectivement de, 60 heures sur un *cluster* de 336 cœurs et $8,0$ Go, avec la matrice $S(MC)^2PET_{high}$.

Les projecteurs *IRIS* surpassent, tant en matière de qualité d'image que de résolution, tous les autres projecteurs et fournissent des résultats équivalents à la matrice $S(MC)^2PET_{high}$, avec une charge mémoire négligeable, et un temps d'estimation par SMC plus faible. L'implémentation *CPU* des projecteurs *IRIS* s'exécute à la même vitesse que la reconstruction avec la matrice $S(MC)^2PET_{low}$ et plus rapidement qu'avec les matrices $S(MC)^2PET_{medium}$ et $S(MC)^2PET_{high}$. Les temps obtenus avec les implémentations *CPU* sont de l'ordre de la dizaine de minutes par itération, ce qui est incompatible avec une application en routine clinique. Cependant, les implémentations *GPU* des projecteurs *IRIS* permettent de les accélérer d'un facteur supérieur à 100 fois, ce qui rend envisageable une exploitation de ces méthodes dans un contexte clinique. Finalement, les projecteurs *IRIS* par rapport aux matrices système stockées $S(MC)^2PET$, fournissent des reconstructions de qualité égale voire supérieure, tout en s'exécutant plus rapidement et en étant plus flexibles aux changements de taille des *voxels* et du champ de vue.

Chapitre 5

Correction du parcours du positon par simulation sur *GPU*

5.1 Introduction **130**
5.2 Estimation du parcours du positon **131**
5.3 Correction du parcours du positon **132**
5.3.1 Réduction du parcours du positon 132
5.3.2 Correction pré et postreconstruction 133
5.3.3 Correction dans le processus de reconstruction 134
5.4 Proposition d'une nouvelle méthode de correction du parcours du positon **135**
5.4.1 Simulation du parcours du positon 135
5.4.2 Estimation des distributions utiles à la simulation 137
5.4.2.1 Énergies des positons 137
5.4.2.2 Paramètres de navigation des positons 138
5.4.3 Intégration au processus de reconstruction 139
5.5 Étude d'évaluation **140**
5.5.1 Données simulées 140
5.5.2 Correction du parcours du positon 143
5.5.3 Implémentations 143
5.5.4 Reconstruction 143
5.6 Résultats **144**
5.6.1 Simulation des positons 144
5.6.2 Reconstructions 148
5.6.2.1 Contraste et bruit 149
5.6.2.2 Résolution 152
5.6.2.3 Temps de reconstruction 152
5.7 Discussion et conclusion **153**

5.1 Introduction

Aujourd'hui, on trouve des méthodes de correction satisfaisantes pour quasiment tous les effets intervenant pendant l'acquisition des données en TEP de, comme l'atténuation, la diffusion ou les coïncidences fortuites. Cependant, les méthodes de correction du parcours du positon restent généralement sommaires, en supposant par exemple le milieu objet comme étant homogène, ou la distribution des annihilations autour d'un point comme étant isotrope. À l'heure actuelle, peu de méthodes de correction du parcours du positon dans des milieux hétérogènes ont été proposées et aucune d'entre elles ne permet une modélisation vraiment précise des hétérogénéités des matériaux dans le corps du patient. Nous avons vu dans la sous-section 1.2.3.1 qu'avec certains traceurs, le parcours du positon peut dégrader les images reconstruites. En effet, la reconstruction fournit la distribution des annihilations alors que celle que l'on cherche à estimer est la distribution des émissions de positons, qui correspond à la répartition du traceur dans le patient. Ces distributions peuvent être très différentes en fonction du spectre d'énergie cinétique du traceur radioactif utilisé et de la distribution des matériaux dans le champ de vu du scanner (le corps du patient).

Tous les isotopes utilisés en TEP ont une demi-vie brève, afin de ne pas rester dans l'organisme pendant une période trop longue après l'examen. En conséquence, il n'est pas possible de stocker ces traceurs pendant plusieurs jours, ni de les transporter sur de trop longues distances. Les isotopes les plus communs en TEP, comme le ^{18}F, sont produits à l'aide d'accélérateurs de particules se trouvant à proximité du scanner TEP, pour éviter leur dégradation pendant le transport jusqu'au patient. Les isotopes produits par séparation chimique d'un radionucléide père, dont la demi-vie est suffisamment longue pour permettre son stockage et son transport, pourraient permettre de se passer de disposer d'un accélérateur de particules très coûteux directement dans l'hôpital. On trouve principalement deux radionucléides produits de cette manière, le ^{68}Ga et le ^{82}Rb, générés respectivement à partir de ^{68}Ge et de ^{82}Sr. Ceux-ci peuvent avoir de nombreuses applications en fonction de la molécule marquée. Par exemple, le ^{68}Ga est utilisé pour diagnostiquer le cancer de la prostate [Afshar-Oromieh *et al.*, 2012], mais aussi pour mesurer la perfusion myocardique [Green *et al.*, 1993]. Le ^{82}Rb est lui aussi utiliser pour cette dernière application [Bateman *et al.*, 2006]. Cependant, ces isotopes émettent des positons dont l'énergie cinétique moyenne est beaucoup plus importante que celle de ceux émis par le ^{18}F, ce qui implique un parcours des positons accru qui peut entraîner une importante dégradation des images reconstruites. Avec de tels traceurs, la capacité des reconstructions à fournir une bonne qualité d'image et une quantification précise, repose en grande partie sur la correction du parcours du positon.

Dans ce chapitre, nous présenterons les problèmes liés à la simulation du parcours des positons, nécessaire pour le modéliser précisément. Ensuite, nous introduirons différentes méthodes de correction du parcours du positon. Nous proposerons ensuite une nouvelle méthode de modélisation du parcours du positon et nous présenterons comment nous l'avons insérée dans le processus de reconstruction pour corriger cet effet. Pour finir, nous évaluerons cette approche avec deux isotopes différents, le ^{18}F, émettant des positons avec une énergie cinétique faible, et le ^{82}Rb qui émet des positons avec une énergie cinétique beaucoup plus élevée.

5.2 Estimation du parcours du positon

La correction précise du parcours du positon nécessite dans un premier temps d'estimer ses effets avec le milieu objet et avec l'isotope considérés. Dans ce contexte, plusieurs approches ont été développées, certaines reposant sur des modèles analytiques dont les paramètres sont dérivés de SMC ou de mesures expérimentales, ou d'autres sur une utilisation directe des données issues de SMC.

Plusieurs tentatives de mesures empiriques du parcours du positon ont été réalisées dans [Phelps *et al.*, 1975, Cho *et al.*, 1975, Hoffmann *et al.*, 1976], mais ces mesures sont de faible précision parce que les résolutions des dispositifs de mesure étaient du même ordre de grandeur que le parcours du positon mesuré. Dans les travaux de [Derenzo, 1986], la distribution des annihilations est mesurée avec une source placée dans de la mousse de polyuréthane de faible densité, où le parcours du positon est particulièrement étendu, puis dans de l'aluminium où les autres effets prédominent sur le parcours du positon. La première distribution est ensuite déconvoluée de la seconde afin de conserver uniquement la *PSF* associée au parcours du positon. Un modèle analytique est ensuite adapté à cette mesure par un changement d'échelle, pour modéliser le parcours des positons dans l'eau, en utilisant comme facteur d'échelle le rapport des densités de l'eau et du polyuréthane. Un problème de cette approche, noté dans [Levin et Hoffman, 1999], est que la distribution des distances parcourues par les positons ne dépend pas uniquement de la densité du matériau, elle est aussi liée par des relations complexes aux numéros atomiques des éléments du matériau.

Étant donné les difficultés à estimer le parcours du positon de manière expérimentale, de nombreuses études se sont portées sur des estimations basées sur des SMC. Dans ce contexte, il existe plusieurs plates-formes de SMC permettant de simuler les interactions des positons avec la matière. Le logiciel Champion, développé par [Champion et Le Loirec, 2006, Champion et Le Loirec, 2007], est dédié spécifiquement à ce type de simulations. Les plates-formes PeneloPET [España *et al.*, 2009] et *GATE* [Jan *et al.*, 2011], basées respectivement sur PENELOPE [Salvat *et al.*, 2006] et Geant4 [Agostinelli *et al.*, 2003, Allison *et al.*, 2006], sont dédiées à la simulation de dispositifs de médecine nucléaire et de radiothérapie, et sont donc capables de simuler le parcours du positon. Les codes EGS4 [Nelson *et al.*, 1985] et leur dérivé EGSnrc [Kawrakow, 2000] sont, comme Geant4, des plates-formes de simulation des interactions des particules avec la matière, utilisées dans plusieurs études pour estimer le parcours du positon.

La simulation du trajet d'un positon dans la matière pose de nombreuses problématiques. Premièrement, le nombre de types d'interactions possibles est relativement important. La figure 5.1 montre les sections efficaces de différentes interactions possibles des électrons et des positons avec la matière (similaires à l'exception de l'annihilation). Pour les positons, on peut noter comme interactions physiques, la diffusion Bhabha et l'ionisation, qui résultent d'interactions avec les électrons et l'effet *bremsstrahlung* qui résulte de l'interaction du positon avec le champ électrostatique à proximité du noyau d'un atome. On peut aussi noter que, les positons dont l'énergie est supérieure à 264 keV se déplacent plus rapidement que la lumière dans l'eau (en considérant que le corps est un volume d'eau) et perdent donc de l'énergie par radiation Cherenkov. Cependant, cet effet est négligeable relativement aux autres. Le dernier type d'interaction physique à prendre en compte est l'annihilation positon-électron, qui conduit à l'émission de deux photons ayant chacun une énergie de 511

keV.

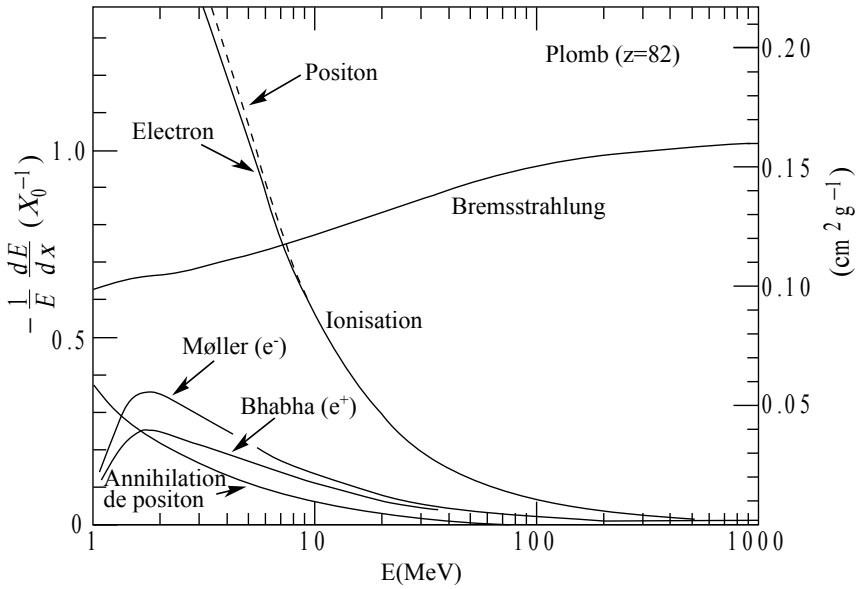


FIGURE 5.1 – Pertes d'énergie fractionnelle des électrons et positons par unité de longueur dans le plomb, associées aux différents effets physiques, en fonction de l'énergie. [Nakamura *et al.*, 2010]

Contrairement aux photons d'annihilations, qui subissent tout au plus quelques dizaines d'interactions avec la matière avant d'être absorbés, les positons sont continuellement diffusés, d'atome en atome jusqu'à annihilation. Afin de déterminer précisément leurs trajectoires, il faudrait les échantillonner avec un pas spatial très faible, d'une taille inférieure à la taille d'un atome. Un tel échantillonnage n'est évidemment pas envisageable, pour des raisons de temps de calcul. Dans Geant4 l'ionisation est approximée par un processus continu, intégrant toutes les interactions le long d'un pas d'échantillonnage qui varie en fonction de l'énergie, de telle sorte que la section efficace ne varie pas sensiblement le long de ce pas.

5.3 Correction du parcours du positon

Dans l'état de l'art de la correction du parcours du positon, on distingue quatre types d'approches. La première consiste à essayer de supprimer ou réduire l'origine du problème, c'est-à-dire la distance parcourue par les positons. Le second type repose sur une correction préreconstruction des projections mesurées. La troisième approche consiste à corriger les images reconstruites *a posteriori*. Le dernier type de correction du parcours du positon repose sur une modélisation de cet effet dans le processus de reconstruction itératif.

5.3.1 Réduction du parcours du positon

Les positons étant des particules chargées, ils sont soumis à la force de Lorentz lorsqu'ils sont placés dans un champ magnétique. Cette force s'applique perpendiculairement au vecteur champ magnétique et au vecteur vitesse du positon, et son intensité est proportionnelle au module du produit

vectoriel de ces deux vecteurs. Cette force est donc maximale lorsque le positon se déplace perpendiculairement au vecteur champ magnétique et est nulle lorsqu'il se déplace parallèlement à celui-ci. Cette force va contraindre le positon à effectuer un virage constant, qui va lui donner une trajectoire en forme de spirale, limitant la distance parcourue entre son émission et son annihilation. Les travaux de [Hammer *et al.*, 1994, Wirrwar *et al.*, 1997] ont étudié cette approche. On peut noter plusieurs limites à cette méthode. D'une part, pour être efficace, elle nécessite des champs magnétiques très intenses (plusieurs teslas) ce qui implique de fortes contraintes sur les matériaux employés dans le scanner (aucun objet ferromagnétique) et un coût important de l'aimant supraconducteur nécessaire pour générer un tel champ magnétique. De plus, la force de Lorentz ne s'applique que dans le plan perpendiculaire au vecteur champ magnétique, donc la distance parcourue par les positons est réduite uniquement dans ce plan, mais n'est pas modifiée dans la direction parallèle au vecteur champ magnétique. Avec cette méthode, on réduit donc le parcours du positon sur seulement deux des trois axes de l'espace.

Actuellement, seuls des prototypes non commerciaux ont exploité cette approche pour réduire le parcours du positon. Toutefois, ce principe est actif de manière intrinsèque avec les scanners combinés TEP/IRM, en raison du champ magnétique généré par l'IRM.

5.3.2 Correction pré et postreconstruction

Une autre approche de correction du parcours du positon repose sur une correction des projections avant de les utiliser pour reconstruire les images de répartition du traceur. Cette méthode nécessite d'abord d'évaluer l'impact du parcours du positon sur les projections en mode sinogramme. [Derenzo, 1986, Haber *et al.*, 1990] ont mesuré la *PSF* associée au parcours du positon dans les projections pour différents isotopes, avec un milieu de propagation homogène composé d'eau. Ces mesures ont servi à déterminer les paramètres d'un modèle analytique décrivant la *PSF* du parcours du positon dans les projections. La correction de ces projections repose ensuite sur une déconvolution. Cette méthode suppose que la *PSF* associée au parcours du positon est uniforme sur les projections, donc dans le champ de vue, ce qui n'est vrai qu'à la condition que le milieu objet soit homogène. Si une *PSF* variable était modélisée dans le sinogramme, elle serait toujours fixe pour un *bin* du sinogramme, ce qui supposerait que la *PSF* du parcours du positon ne varie pas le long de la droite de l'espace associée à ce *bin*, dans le champ de vue. En TEP, les projections sont généralement entachées d'un important bruit statistique et appliquer une déconvolution à ce type de données aurait tendance à amplifier encore le niveau de bruit, impliquant une réduction de la qualité des images reconstruites.

Une reconstruction sans correction du parcours du positon fournit la distribution des annihilations, qui est égale à la distribution des émissions de positons floutée (convoluée) par une *PSF* spécifique en chaque *voxel* de champ de vue. On peut alors imaginer appliquer une correction postreconstruction pour déconvoluer les images reconstruites et ainsi obtenir la distribution des émissions des positons. Les travaux de [Johnson *et al.*, 2011] ont utilisé cette approche pour corriger les effets du parcours du positon associés au ^{82}Rb pour l'imagerie cardiaque. Cependant, cette méthode est peu exploitée parce qu'elle entraîne une augmentation du niveau de bruit qui dégrade les images reconstruites.

5.3.3 Correction dans le processus de reconstruction

La modélisation du parcours du positon dans la *SRM* utilisée dans les reconstructions itératives, permet de corriger cet effet tout en contrôlant l'augmentation du niveau de bruit.

La méthode la plus directe pour obtenir une telle *SRM* consiste à construire cette matrice avec une SMC complète qui intègre des modèles du milieu objet et de l'isotope utilisé. Dans les travaux de [Matthieu, 2014], la réponse du système est estimée par une SMC intégrant une modélisation du détecteur, de l'objet imagé, d'une source, composée du même isotope émetteur de positons que celui utilisé dans l'acquisition des projections, et des interactions des positons et des photons avec l'objet et le détecteur. L'inconvénient de cette approche est qu'elle nécessite de construire une nouvelle *SRM* pour chaque objet imagé, ce qui nécessite des temps de calcul de l'ordre de la dizaine de jours avec un *cluster* de calcul d'une centaine de cœurs.

Une approche plus courante, vue dans la sous-section 1.5.1, est de décomposer la *SRM* en un produit de matrices, chacune modélisant une partie de la réponse du système, l'une d'elles étant dédiée au parcours du positon. La matrice modélisant le parcours du positon n'est généralement utilisée que dans le calcul de la projection et pas dans celui de la rétroprojection, c'est à dire un projecteur non symétrique (*unmatched*), comme nous l'avions présenté dans l'équation de la reconstruction avec toutes les corrections dans le paragraphe 1.5.9. En effet, [Cal-González *et al.*, 2011] ont montré que le fait de modéliser le parcours du positon dans la rétroprojection, ralentit la convergence de la reconstruction, mais il n'est pas exclu que cela implique une qualité d'image inférieure. La modélisation du parcours du positon seulement dans l'étape de projection réduire la charge de travail à chaque itération et d'accélérer la convergence (donc réduire le nombre d'itérations), ce qui permet d'espérer conserver des temps de reconstruction compatible avec les temps cliniques.

La prise en compte du parcours du positon dans l'opération de projection repose sur la convolution de l'image reconstruite à l'itération courante, par des noyaux de convolution différents en chaque *voxel* du champ de vue. Chacun de ces noyaux donne la distribution des annihilations des positons émis dans le *voxel* considéré, qui dépend de la répartition de matériaux autour du *voxel* et de l'énergie des positons émis par le traceur considéré. Cependant, estimer et stocker un noyau pour chaque *voxel* du champ de vue nécessiterait des ressources en calcul ainsi qu'un espace de stockage importants. Pour pallier ce problème, certaines méthodes font l'approximation que ce noyau de convolution ne varie pas et est isotrope, comme présenté dans les travaux de [Bai *et al.*, 2005, Ruangma *et al.*, 2006, Rahmim *et al.*, 2008b, Cal-González *et al.*, 2009]. Ce noyau est généralement estimé pour modéliser le parcours du positon dans l'eau. Cette méthode n'est efficace qu'à condition que l'objet imagé ne soit pas trop hétérogène et que les tumeurs ne sont pas à proximité d'interfaces entre des matériaux de densités différentes. En pratique, ces conditions sont rarement satisfaites. De plus, cette méthode ne permet pas une modélisation précise du parcours du positon dans les tissus dont la densité est très éloignée de celle de l'eau, comme les os et les poumons. D'autres approches, comme celle de [Cal-González *et al.*, 2011], estiment des noyaux de convolutions pour chacun des matériaux présents dans l'objet, sans modéliser les hétérogénéités de matériaux dans les voisinages des *voxel*. La modélisation du parcours du positon est obtenue en convoluant le volume en utilisant, en chaque *voxel*, le noyau associé au matériau présent dans celui-ci. Cette méthode est efficace dans les zones

où le matériau est homogène, mais cause d'importants artefacts au voisinage des transitions entre matériaux de densités très différentes. Enfin, un dernier type de modélisations du parcours du positon est celui où un noyau de convolution spécifique au voisinage de chaque *voxel*, est estimé. Précédemment, nous avons noté que l'estimation d'un noyau spécifique pour chaque *voxel* n'est pas envisageable pour des raisons de temps de calcul et de consommation mémoire trop importante. L'approche de [Alessio et MacDonald, 2008] consiste à estimer les noyaux, dans un milieu homogène, pour tous les matériaux présents dans l'objet imagé. Les noyaux de convolution de chaque *voxel* sont ensuite construits pendant la reconstruction, en interpolant les noyaux préestimés. La méthode proposée par [Bai *et al.*, 2003] construit un noyau de convolution par une croissance itérative, en partant du *voxel* central puis en s'éloignant de proche en proche, estimant à chaque pas le nombre de positons s'annihilant et survivant, en fonction du matériau.

5.4 Proposition d'une nouvelle méthode de correction du parcours du positon

Précédemment, nous avons noté que la correction du parcours du positon passe par son estimation dans le processus de reconstruction, dans l'opération de projection. Celle-ci est calculée par une étape de convolution qui permet de passer d'une carte de la distribution des émissions de positon à une carte de la distribution des annihilations. Dans ce contexte, plusieurs modèles ont été proposés, mais aucun d'eux ne permet de modéliser précisément les hétérogénéités des matériaux dans l'objet imagé.

L'estimation précise de la distribution des annihilations à partir de la distribution des émissions de positons à l'aide d'une convolution demande d'appliquer un noyau de convolution spécifique à chaque *voxel*, ce qui pose des problèmes de temps de calcul et de stockage. Pour répondre à ce problème, nous proposons une nouvelle méthode de modélisation du parcours du positon, permettant d'estimer à la volée, pendant le processus de reconstruction, la distribution des annihilations à partir d'une carte des émissions de positon (l'image reconstruite à l'itération courante). Cette méthode se base sur une simulation SMC simplifiée et accélérée sur *GPU*, qui peut être exécutée pendant la reconstruction sans impacter trop lourdement les temps de reconstruction. Le principe général de notre approche est de fournir à notre SMC, avant d'effectuer l'opération de projection, l'image reconstruite à l'itération courante. Celle-ci est utilisée comme distribution des émissions de positons. Après avoir simulé un certain nombre de positons, la distribution de leurs annihilations fournit l'image qui est ensuite utilisée par l'opération de projection. Cela permet de calculer la distribution des annihilations sans construire explicitement de noyaux de convolution. Cette méthode, nommée *MCC-PR* pour *Monte Carlo Convolution for Positron Range correction*, est présentée dans cette section.

5.4.1 Simulation du parcours du positon

La simulation avec la méthode *MCC-PR* repose sur l'utilisation d'histogrammes précalculés décrivant l'état initial du positon en fonction du traceur et l'évolution de son état en fonction de son

énergie. Nous détaillons ici comment sont générés les positons, et comment s'effectue leur navigation dans un fantôme voxélisé.

La création des positons nécessite d'initialiser leur état, c'est-à-dire leur position, direction de propagation et énergie cinétique. Pour initialiser la position, nous utilisons la même procédure que celle décrite dans la sous-section 3.3.3.1, basée sur l'utilisation de la fonction de répartition. L'émission des positons est isotrope, on définit donc une direction aléatoire répartie uniformément sur la sphère unité. L'énergie cinétique initiale est générée aléatoirement en utilisant l'histogramme de la distribution de l'énergie cinétique des positons émis par l'isotope considéré, toujours avec la même méthode estimant la fonction de répartition avec la somme cumulée de l'histogramme. Cette méthode repose sur le calcul de l'histogramme cumulé des énergies, que l'on normalise en 0 et 1. Ensuite, la création d'une énergie aléatoire se fait en générant d'abord un nombre uniformément distribué dans l'intervalle $[0,1[$ et en trouver dans l'histogramme cumulé, la première classe d'énergie dont la valeur est supérieure à ce nombre. Afin d'éviter de générer des valeurs d'énergies discrétisées (au centre de chaque classe de l'histogramme), on génère une énergie uniformément distribuée dans l'intervalle d'énergie associé à la classe de l'histogramme.

Le principe de la navigation des positons avec la méthode *MCC-PR* repose sur un déplacement pas à pas, avec une taille de pas qui dépend du matériau et de l'énergie du positon (modèle de simulation utilisé dans *Geant4*), et sur l'utilisation d'un ensemble de tables définissant pour chaque matériau et intervalle d'énergie cinétique du positon, l'histogramme de la variation d'énergie cinétique et celui de la variation d'angle de propagation ainsi que la probabilité d'annihilation du positon, à la différence de *GATE* qui calcule à chaque pas ces différentes distributions en fonction de l'énergie du positon et du matériau.

Un résumé de la simulation utilisée avec la méthode *MCC-PR* est présenté dans le diagramme de la figure 5.2. Un pas de la navigation d'un positon d'énergie cinétique E, de position $\vec{X}$ et de direction $\vec{D}$ se décompose de la manière suivante. Premièrement, en fonction de son énergie et du matériau du *voxel* dans lequel il se trouve, on lit dans les tables précalculées la probabilité P_{ann} qu'il s'annihile. Un nombre aléatoire u uniformément distribué entre 0 et 1 est généré, si $u < P_{ann}$, on considère que le positon s'annihile et la simulation s'arrête là, sinon elle continue. Si la simulation continue, la taille du pas τ, que va effectuer le positon est déterminée en lisant dans les tables précalculées, qui donnent la valeur de ce pas en fonction de l'énergie du positon et du matériau dans lequel il se propage. Le positon est ensuite déplacé à sa nouvelle position $\vec{X} = \vec{X} + \tau\vec{D}$, sauf si, le long de ce pas, le positon traverse une frontière entre deux matériaux, auquel cas, on calcule la fraction f du pas permettant d'atteindre cette frontière, et on déplace le positon sur cette dernière, avec l'équation $\vec{X} = \vec{X} + f\tau\vec{D}$. Deuxièmement, en fonction du matériau le long du pas et de l'énergie du positon, les histogrammes précalculés de l'angle de diffusion et de la perte d'énergie sont lus. La perte d'énergie δE et l'angle de diffusion ϕ sont générés aléatoirement avec leurs histogrammes respectifs. Ces valeurs sont multipliées par f, la fraction du pas calculé précédemment, ou sont conservées telles quelles s'il n'y a aucun changement de matériau le long du pas. L'énergie est mise à jour de la façon suivante $E = E - \delta E$, et la direction $\vec{D}$ est pivotée d'un angle ϕ. Un second angle θ, nécessaire pour définir complètement la rotation, est tiré aléatoirement dans la distribution répartie uniformément sur l'intervalle $[0, 2\pi[$. Cette procédure est répétée jusqu'à ce que le positon s'annihile.

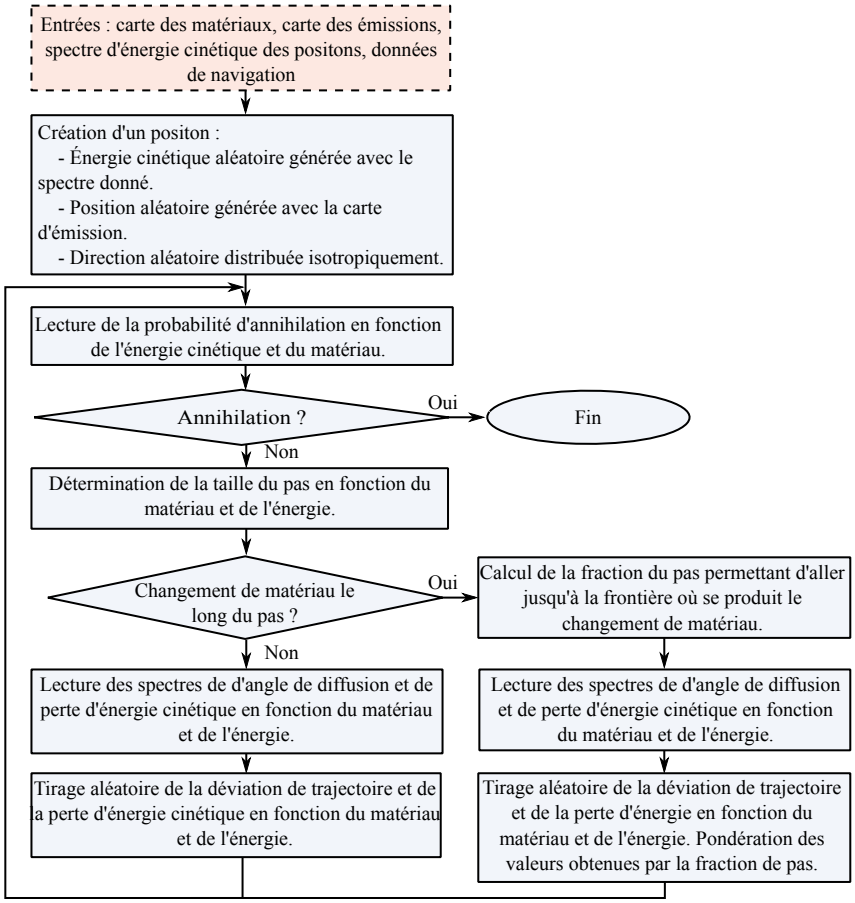


FIGURE 5.2 – Diagramme du processus de simulation du positons avec la méthode *MCC-PR*.

5.4.2 Estimation des distributions utiles à la simulation

5.4.2.1 Énergies des positons

Avec la méthode *MCC-PR*, nous avons vu que pour générer les positons, il est nécessaire de disposer d'un histogramme de l'énergie cinétique des positons émis par l'isotope considéré. Pour mesurer cet histogramme, nous utilisons le *phase space actor* de *GATE*. Cet outil permet d'exporter dans un fichier l'ensemble des particules et de leurs caractéristiques, qui entrent dans un objet donné. Nous avons donc placé dans le vide, une source ponctuelle de l'isotope considéré, autour de laquelle nous avons mis une sphère creuse d'un diamètre quelconque. Le *phase space actor* enregistre toutes les particules qui entrent dans cette sphère creuse. Le fichier de sortie est traité afin de ne conserver que l'énergie cinétique des positons, dont on construit l'histogramme, avec 100 classes d'énergie, réparties linéairement entre 0 eV et l'énergie maximale associée à l'isotope.

Le résultat de cette mesure est affiché dans la figure 5.3, pour quelques isotopes couramment employés en TEP.

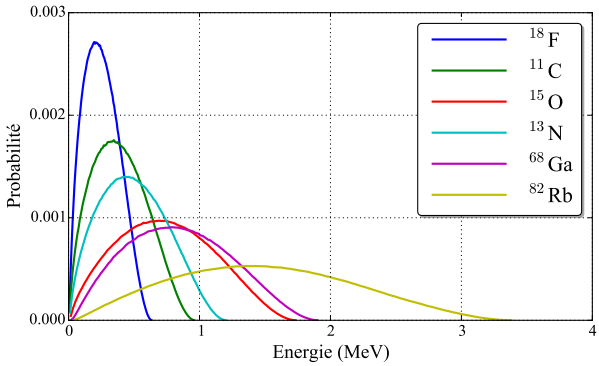


FIGURE 5.3 – Spectres d'énergie cinétique des positons émis par des traceurs communs, mesurés avec *GATE*.

5.4.2.2 Paramètres de navigation des positons

Notre méthode de navigation d'un positon nécessite pour chaque matériau présent dans le fantôme, quatre tables. Parmi celles-ci, une donne la taille du pas en fonction de l'énergie du positon, une autre donne la probabilité d'annihilation en fonction de l'énergie du positon, une autre encore fournit les histogrammes de la distribution des angles de diffusion, pour différentes classes d'énergies, et pour finir une dernière donne les histogrammes de la distribution de l'énergie perdue le long du pas, pour différentes classes d'énergies.

Dans *GATE* il n'existe pas d'*actor* permettant d'exporter l'ensemble des caractéristiques des particules à chaque pas effectué, ce dont nous avons besoin pour estimer les quatre distributions nécessaires à la navigation des positons avec la méthode *MCC-PR*. Il est cependant possible d'ajouter des *actors* personnels, ce que nous avons fait. Cet *actor* que nous avons créé, exporte dans un fichier texte pour un positon donné et à chaque pas de sa navigation : sa position et son énergie au début du pas, la taille du pas, l'énergie perdue le long du pas et la différence d'angle entre sa direction de propagation au début et la fin du pas.

Nous avons exécuté des simulations *GATE* avec cet *actor*, pour un catalogue de matériaux communs. Le principe de cette simulation est de placer une source ponctuelle de positons, émis avec une énergie cinétique initiale de 3,5 MeV, placée dans une sphère composée du matériau considéré d'une taille suffisamment importante pour que les positons ne s'en échappent pas. La valeur initiale a été fixée à 3,5 MeV parce qu'elle est supérieure à la valeur maximale de l'énergie cinétique des positons émis par le ^{82}Rb, qui est l'isotope émettant les positons les plus énergétiques, parmi les isotopes considérés. Nous avons choisi d'utiliser une énergie cinétique initiale unique parce qu'au fil des interactions les positons perdent de l'énergie, ce qui permet d'observer des positons avec toutes les énergies possibles inférieures à l'énergie initiale.

La sortie de l'*actor* est traitée pour construire les différentes tables. Nous avons choisi d'utiliser des classes dont les tailles varient de manière logarithmique pour l'énergie, la variation d'énergie et l'angle de diffusion. Tous les histogrammes ont été construits avec 100 classes, les valeurs minimales et maximales ont été fixées respectivement à 500 eV et 3,5 MeV pour l'énergie, 100 eV et 500 keV pour la variation d'énergie, 0,06° et 180° pour l'angle de diffusion.

Des exemples de résultats de ces mesures, effectuées pour trois matériaux, l'eau, le tissu pulmo-

naire et l'os, sont présentés dans la figure 5.4.

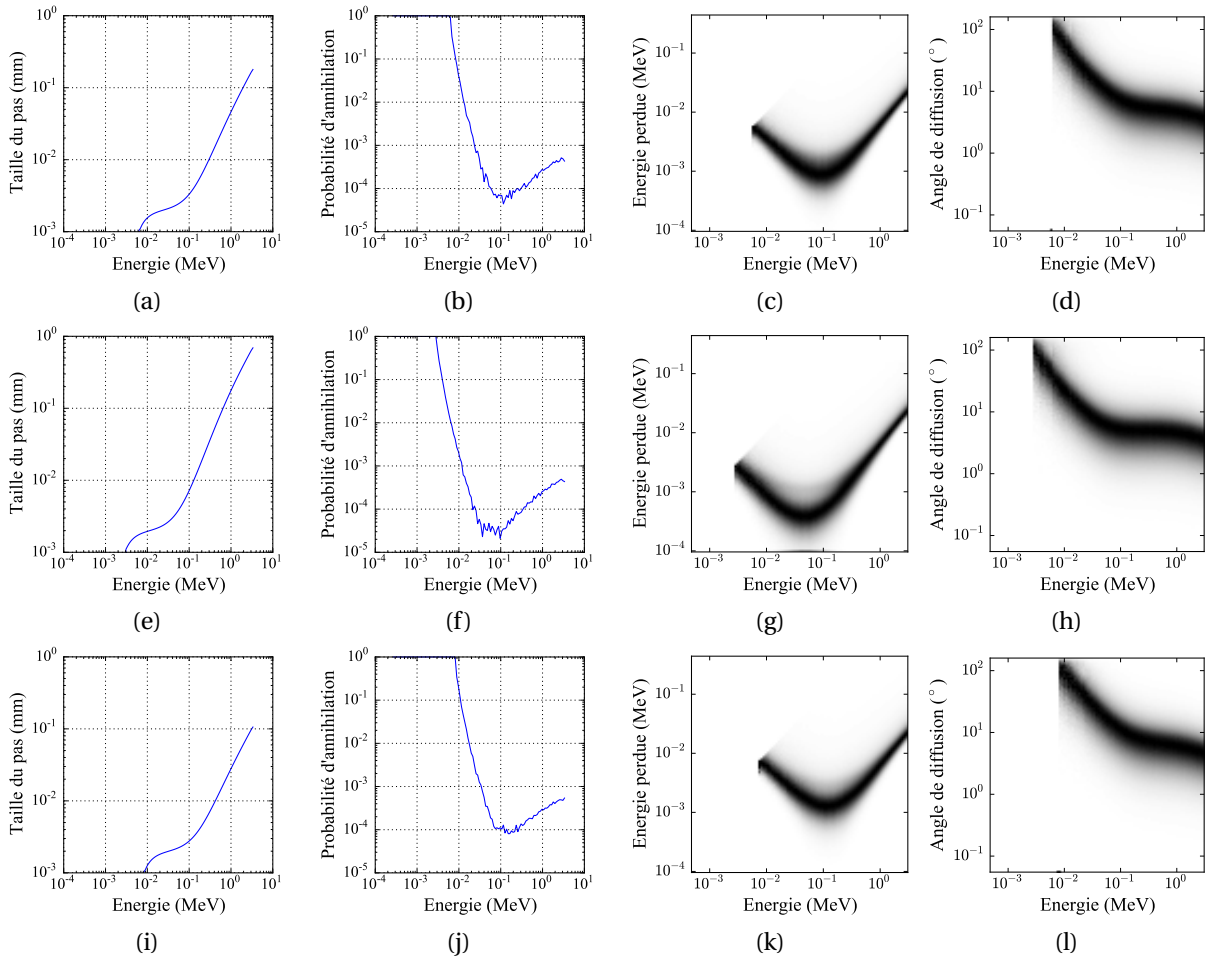


FIGURE 5.4 – Histogrammes des variations des paramètres des positons le long d'un pas en fonction de son énergie, pour trois matériaux différents, l'eau en (a), (b), (c) et (d), dans du tissu pulmonaire en (e), (f), (g) et (h), et dans de l'os en (i), (j), (k) et (l). La première colonne montre la taille du pas, la deuxième colonne la probabilité d'annihilation, la troisième colonne montre sous la forme d'une image les histogrammes de l'énergie perdue le long du pas pour les différentes énergies des positons, et la dernière colonne montre le même type d'histogramme que la colonne précédente, mais pour l'angle de diffusion.

5.4.3 Intégration au processus de reconstruction

Cette simulation est intégrée au processus de reconstruction avant la projection, comme nous l'avons présenté dans la sous-section 5.3.3. C'est-à-dire que le volume reconstruit à l'itération courante est fourni à la simulation *MCC-PR* comme une source voxélisée. La distribution des annihilations des positons simulés donne un nouveau volume, qui est ensuite utilisé pour calculer la projection. Nous avons choisi de simuler autant de positons qu'il y a de coïncidences dans le jeu de données fourni à la reconstruction.

5.5 Étude d'évaluation

5.5.1 Données simulées

L'ensemble des SMC de cette étude ont été exécutées sur la plate-forme *GATE*. Les processus physiques modélisés pour les photons sont l'effet photoélectrique et l'effet Compton, avec le modèle standard et l'effet Rayleigh, avec le modèle Penelope. Les effets physiques modélisés pour les électrons et les positons sont l'ionisation (inclut l'effet Bhabha et la diffusion Möller), l'effet Bremsstrahlung et l'annihilation, tous modélisés avec le modèle standard. L'ionisation et l'effet Bremsstrahlung sont modélisés par variation continue de l'état du positon/électron. *GATE* définit automatiquement un pas d'échantillonnage, mais il est nécessaire de donner une limite basse à ce pas afin de limiter le temps de simulation. Nous avons fixé à 1 micromètre la valeur minimale pour la taille du pas. Il est aussi nécessaire de fixer une limite basse à la variation de la taille de deux pas consécutifs afin que l'énergie varie peu le long du pas, et donc les sections efficaces aussi. Nous avons fixé ce rapport de variation maximale à $0,01$.

Un premier fantôme présenté dans la figure 5.5 a été utilisé pour comparer les résultats d'une simulation *MCC-PR* avec une simulation *GATE*. Ce fantôme cubique de $11 \times 11 \times 11$ mm^3 se compose, suivant l'axe z, des cinq couches suivantes, une d'eau de 2 mm, une couche de tissu pulmonaire de 2 mm, une d'eau de 3 mm, une d'os de 2 mm et une d'eau de 2 mm. Une source ponctuelle de ^{82}Rb est placée au centre de ce fantôme. Les positons sortants par les plans x=-5,5 mm et x=5,5 mm sont enregistrés dans un espace de phase. 50 millions de positons ont été simulés et les histogrammes des positions z et direction suivant ce même axe, ont été construit pour les positons de l'espace de phase de chacune des deux simulations.

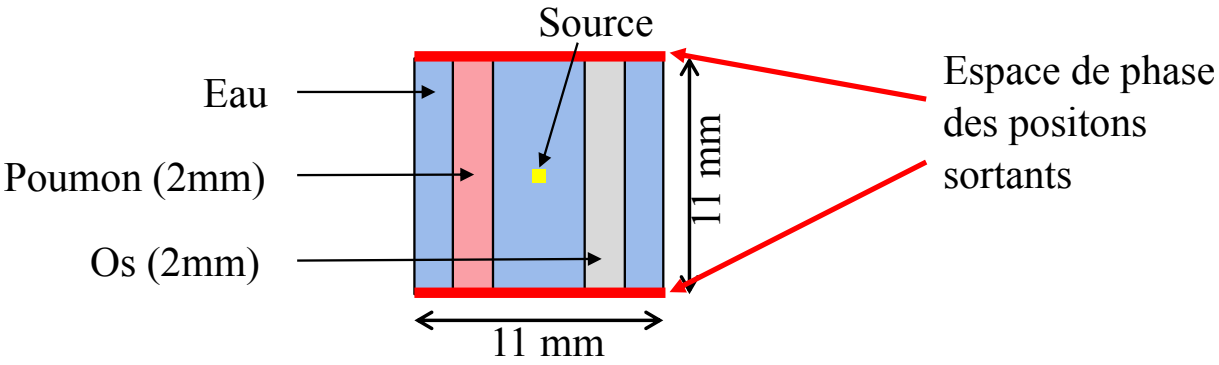


FIGURE 5.5 – Fantôme pour l'évaluation de l'espace de phase. Ce fantôme cubique de $11 \times 11 \times 11$ mm^3 est constitué de trois matériaux, de l'eau, de l'os et de tissu pulmonaire répartie en 5 couches le long de l'axe z.

Les jeux de données *list-mode* ont été obtenus en simulant les différents fantômes dans le scanner préclinique INVEON de siemens, validé par [Anizan, 2010, Matthieu, 2014].

Pour cette étude, nous avons développé trois fantômes pour évaluer d'une part la qualité de la méthode *MCC-PR* par rapport à une SMC réalisée avec *GATE* et d'autre part, la qualité des images reconstruites, corrigées ou non de l'effet du parcours du positon.

Le fantôme F_{prof}, présenté dans la figure 5.6, a été utilisé pour faire une évaluation visuelle, de la qualité de la simulation de la méthode *MCC-PR* et des images reconstruites. Il est composé de trois

sphères concentriques, de 46 mm, 74 mm et 140 mm de diamètre, constituées respectivement de tissu pulmonaire, d'os et d'eau. Une source cylindrique de 10 mm de diamètre et 34 mm de long est placée le long de l'axe du scanner de telle sorte qu'elle chevauche les trois matériaux. Trois sources sphériques de 10 mm de diamètre sont placées le long de l'axe du scanner, chacune dans un des trois matériaux qui constituent le fantôme.

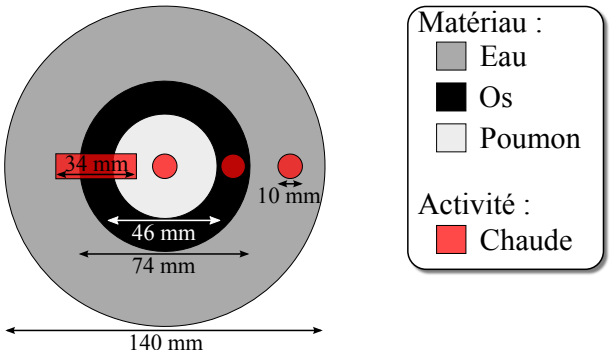


FIGURE 5.6 – Fantôme F_{prof} pour une étude visuelle de la simulation et de la correction du parcours du positon. Il est constitué de trois sphères de matériaux différents, imbriquées les unes dans les autres, une sphère de tissu pulmonaire au centre, une sphère d'os ensuite et une sphère d'eau autour. Quatre sources de même densité d'activité sont placées le long de l'axe du scanner. Une source cylindrique placée de telle sorte qu'elle traverse les trois matériaux qui composent le fantôme. Les trois autres sources, des sphères, sont chacune placée dans un des trois matériaux.

Le fantôme F_{cont}, présenté dans la figure 5.7, est dédié à l'étude du contraste et du bruit dans les images reconstruites. Il est composé de quatre cylindres concentriques de 50 mm de long et de 48 mm, 103 mm, 108 mm et 140 mm de diamètre, composés respectivement de tissu pulmonaire, d'eau, d'os et encore d'eau. Le second cylindre d'eau est actif, tandis que le reste est inactif. Six sphères, remplies d'une activité quatre fois supérieure à celle du cylindre d'eau, sont placées dans le cylindre de tissu pulmonaire et dans le cylindre d'eau active, par groupes de trois sphères de 20 mm , 10 mm et 5 mm de diamètre.

À chaque itération, le CRC est calculé dans chacune des six sphères, relativement à l'activité reconstruite dans le large cylindre d'eau active, mesurée dans trois régions d'intérêts cylindriques de 20 mm de diamètre et 30 mm de long. Le $Bruit_{SD}$ a aussi été calculé dans le large cylindre d'eau active, avec l'équation 4.3.

Pour finir, nous avons aussi utilisé ce fantôme pour évaluer l'impact sur le temps de reconstruction des différentes méthodes de correction du parcours du positon.

Enfin, un dernier fantôme F_{res} a été utilisé pour étudier la résolution dans les images reconstruites. Ce fantôme, présenté dans la figure 5.8, est composé d'un cylindre d'eau, aligné sur l'axe du scanner, de 110 mm de diamètre et 110 mm de long, dans lequel sont placés deux plus petits cylindres constitués d'os et de tissu pulmonaire, ayant un diamètre de 18 mm et une longueur de 90 mm. Dans le plan, perpendiculaire à l'axe du scanner et passant par son centre, trois sources ponctuelles, de même activité, sont placées chacune dans un des trois matériaux composant le fantôme. La résolution dans les images reconstruites est estimée en ajustant une distribution Gaussienne 3D sur chaque source ponctuelle dans les images reconstruites. Le résultat donné est la moyenne des *FWHM* de la distribution Gaussienne 3D suivant les trois axes de l'espace.

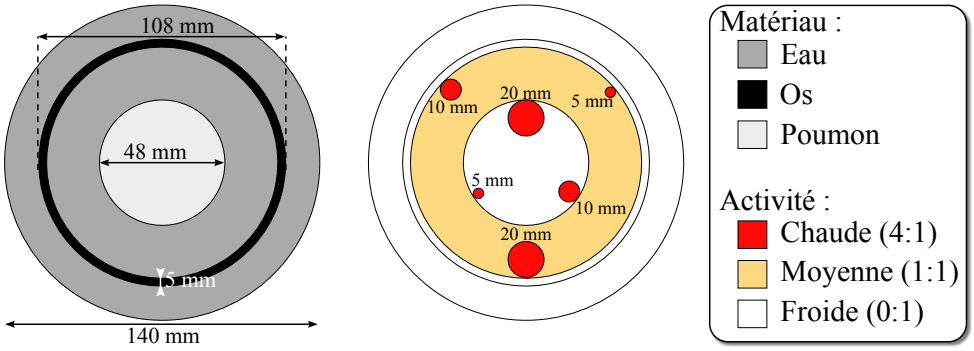


FIGURE 5.7 – Fantôme F_{cont} pour l'étude du CRC en fonction du $Bruit_{SD}$ des images reconstruites. Ce fantôme contient deux groupes de trois sphères actives de 5 mm , 10 mm et 20 mm de diamètre, le premier groupe étant placé dans un fond d'eau moyennement active et le second groupe, dans de l'air inactif.

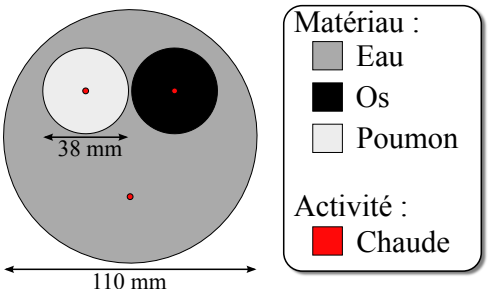


FIGURE 5.8 – Le fantôme F_{res} est dédié à l'évaluation de la résolution dans les images reconstruites. Il est constitué d'un large cylindre d'eau dans lequel sont placés deux cylindres de plus petit diamètre, constitués d'os et de tissu pulmonaire. Trois sources ponctuelles sont placées dans la coupe centrale du scanner, chaque source étant placée dans un des trois matériaux composant le fantôme.

Afin de valider la simulation de la méthode *MCC-PR*, ces trois fantômes ont été simulés sur *GATE* et avec notre simulation. Les fantômes F_{res}, F_{prof} et F_{cont} ont été simulés avec respectivement 2, 100 et 200 millions de positons avec *GATE* et 10 fois plus avec la méthode *MCC-PR*, afin de réduire le niveau de bruit statistique dans les résultats des simulations et parce que la simulation est beaucoup moins coûteuse en temps de calcul qu'avec *GATE*.

Ensuite, afin d'évaluer la méthode *MCC-PR* dans un contexte de reconstruction, ces fantômes ont été simulés une seconde fois sur la plate-forme *GATE*, avec le modèle du scanner INVEON, afin de générer des jeux de données *list-mode*. Trois types d'émissions ont été utilisés. Le mode *back-to-back* qui, comme nous l'avons vu précédemment, émet directement les photons d'annihilation sans simuler de positon. Ce type d'émission fournit des jeux de données qui ne sont pas impactés par le parcours du positon. Les deux autres types d'émissions ont été simulés avec des isotopes émetteurs de positons, le ^{18}F et le ^{82}Rb. Le premier est l'isotope le plus commun en TEP et est associé un parcours du positon faible. Le second est quant à lui moins commun, mais c'est l'isotope émetteur de positons utilisé en routine clinique qui émet les positons les plus énergétiques. Il est donc associé à un parcours du positon important. Les fantômes F_{res}, F_{prof} et F_{cont} ont été simulés, avec ces trois types d'émissions de positons, afin d'obtenir des jeux de données *list-mode* de respectivement 1, 10 et 40 millions de coïncidences vraies.

5.5.2 Correction du parcours du positon

Toutes les méthodes de correction du parcours du positon comparées dans cette étude sont incluses dans le processus de reconstruction avec la méthode évoquée précédemment dans la section 5.3.3.

La première méthode de correction modélise le parcours du positon par un noyau de convolution isotrope et invariant. Ce noyau a été calculé à l'aide d'une SMC, sur *GATE*, dans laquelle une source ponctuelle de l'isotope considéré, a été placée dans un volume d'eau. Le noyau a été construit en comptant en chacun de ses *voxel* le nombre d'annihilations s'y étant produit. Dans la suite, on nommera cette méthode de correction "convolution invariante".

La deuxième méthode de correction du parcours du positon est similaire à la précédente à la différence que le noyau de convolution varie en fonction du matériau présent dans chaque *voxel*. L'ensemble de nos fantômes est composé de trois matériaux, l'eau, l'os et le tissu pulmonaire. Nous avons donc construit les trois noyaux de convolution, associés à ces matériaux, avec la même méthode que celle présentée dans le paragraphe précédent. Par la suite, on notera cette méthode de correction "convolution variante".

La dernière méthode de correction du parcours du positon est la méthode *MCC-PR*, présentée dans la section 5.4. Le nombre total de positons simulés à chaque itération a été fixé égal au nombre de coïncidences dans le jeu de données.

5.5.3 Implémentations

Les trois méthodes de correction du parcours du positon ont été implémentées sur *GPU*. Les méthodes par convolution invariante et variante sont implémentées de la même manière, c'est-à-dire que chaque *voxel* du champ de vue est traité par un *thread*. Les noyaux de convolution sont stockés dans la mémoire globale du *GPU*. Chaque *thread* sélectionne le bon noyau de convolution, toujours celui associé à l'eau pour la convolution invariante et celui du matériau du *voxel* traité avec la convolution variante. Toutes les valeurs des *voxels* du volume d'entrée couvertes par le noyau sont multipliées par les valeurs du noyau, puis sommées, et le résultat est enregistré dans le *voxel* du volume de sortie.

Avec la méthode *MCC-PR*, la simulation est implémentée de telle sorte que chaque positon simulé soit traité par un *thread* spécifique. L'ensemble des tableaux nécessaires à la simulation des positons sont stockés dans la mémoire globale du *GPU*. Pour chaque positon simulé, sa position finale est imprévisible, donc l'indice du *voxel* du volume de sortie aussi. Par conséquent, afin d'éviter les collisions mémoire (deux *threads* dont les positons s'annihilent dans le même *voxel* au même moment), les *voxels* sont incrémentés avec une opération atomique.

5.5.4 Reconstruction

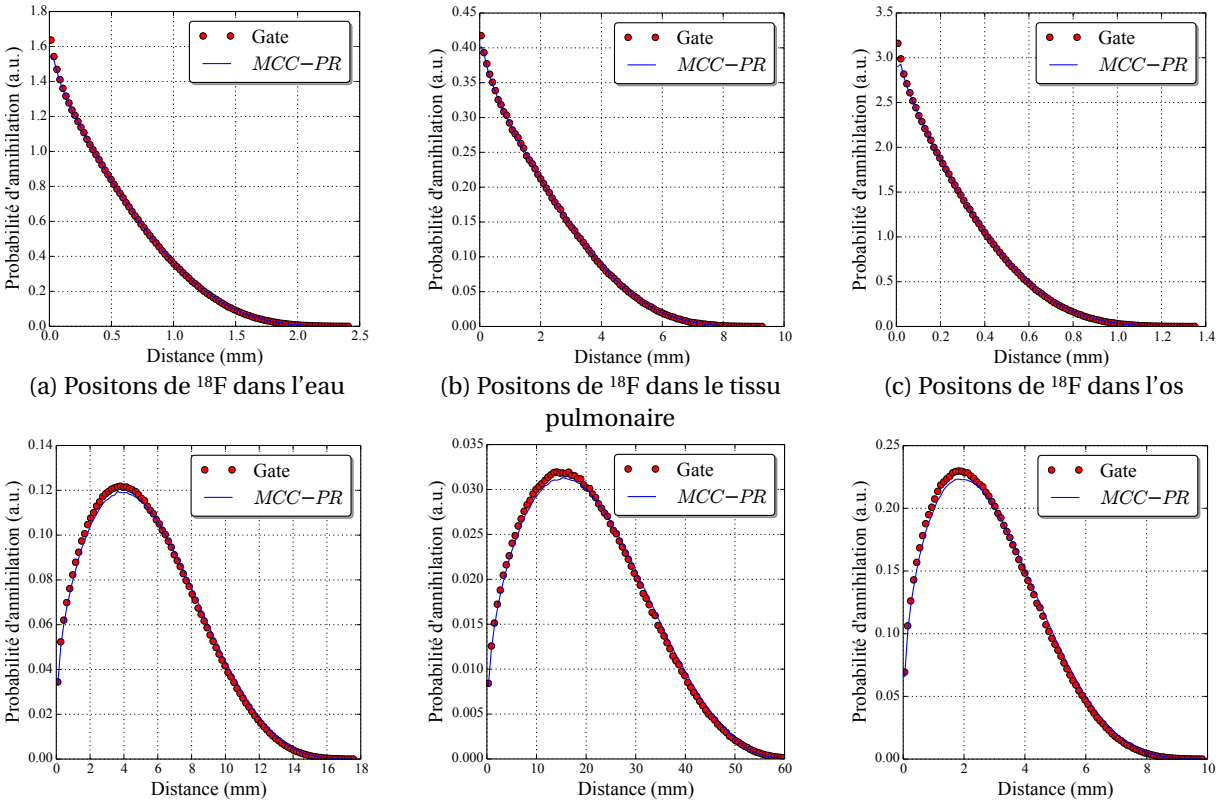
L'ensemble des reconstructions ont été traitées avec l'algorithme *LM-EM*, présenté dans le paragraphe 1.4.2.2, avec des *voxels* de 1^3 mm^3, le projecteur $IRIS_{analytique}$, présenté dans la sous-section 3.3.3.2, et la correction d'atténuation. Toutes les reconstructions ont été exécutées sur un *GPU* NVIDIA GTX 980 Ti cadencé à 1GHz.

Les données acquises avec le mode *back-to-back* n'étant pas impactées par le parcours du positon, aucune correction de cet effet ne leurs a été appliquée. Les données acquises avec le ^{18}F et le ^{82}Rb comme émetteurs de positons, ont été reconstruites sans correction du parcours du positon, ainsi qu'avec les trois méthodes de correction de cet effet.

Les facteurs de mérite sont mesurés pour les 100 premières itérations et les images, et les profils sont extraits des images reconstruites avec 30 itérations.

5.6 Résultats

5.6.1 Simulation des positons



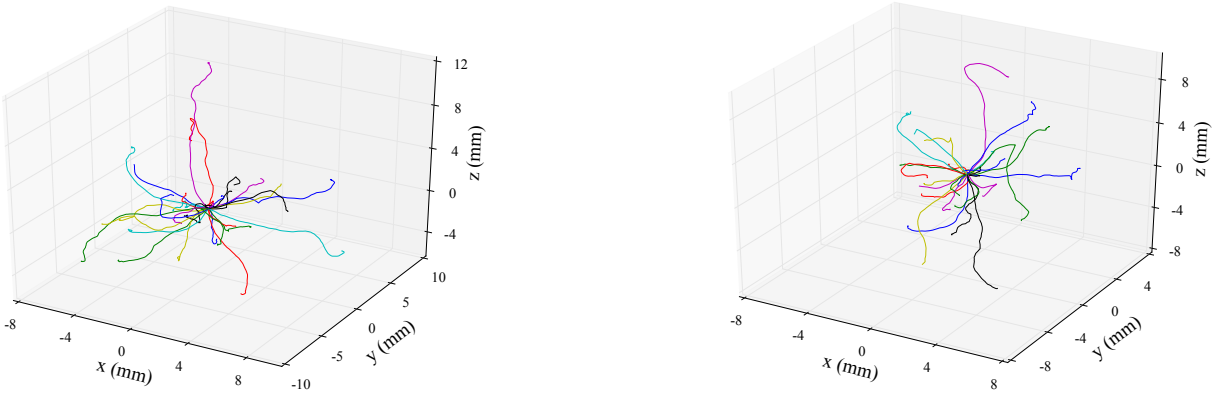
(a) Positons de ^{18}F dans l'eau (b) Positons de ^{18}F dans le tissu pulmonaire (c) Positons de ^{18}F dans l'os

(d) Positons de ^{82}Rb dans l'eau (e) Positons de ^{82}Rb dans le tissu pulmonaire (f) Positons de ^{82}Rb dans l'os

FIGURE 5.9 – Histogrammes de la distance parcourue par les positons jusqu'à annihilation, simulés avec *GATE* et *MCC-PR*. Les positons sont émis avec du ^{18}F dans (a), (b) et (c) et avec du ^{82}Rb dans (d), (e) et (f). Les positons des figures (a) et (d) sont émis dans l'eau, (b) et (e) dans du tissu de poumon et (c) et (f) dans de l'os.

La figure 5.9 présente les histogrammes de la distance entre la position d'émission et celle d'annihilation, pour 1 million de positons émis par du ^{18}F et du ^{82}Rb, simulés dans trois matériaux, l'eau le tissu pulmonaire et l'os, avec *GATE* et la méthode *MCC-PR*. Dans tous les cas, on constate que la courbe du *MCC-PR* se superpose dans sa majeure partie à celle obtenue avec *GATE*. Avec le ^{18}F on peut noter un écart principalement pour la distance la plus faible, où la courbe de *GATE* est toujours légèrement au-dessus de celle de la simulation *MCC-PR*, avec des écarts de 4 %, 2,9 % et 7,8 %, dans

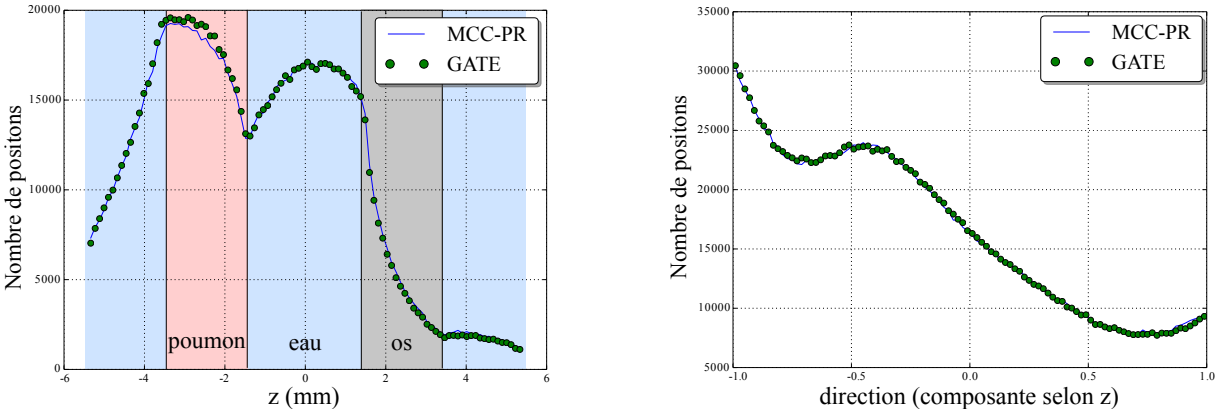
l'eau, le tissu pulmonaire et l'os respectivement. Avec le ^{82}Rb, la courbe de *GATE* passe toujours au-dessus de celle de *MCC-PR* au niveau du maximum de la distribution, où les écarts sont de 1,5 %, 1,5 % et 3 %, dans l'eau, le tissu pulmonaire et l'os respectivement. Cette tendance s'inverse pour les plus grandes distances. Ces écarts sont la conséquence de la simplification de la simulation.



(a) *GATE* (b) *MCC-PR*

FIGURE 5.10 – Trajectoires de 20 positons, émis par du ^{82}Rb dans l'eau, simulés avec *GATE* en (a) et avec *MCC-PR* en (b).

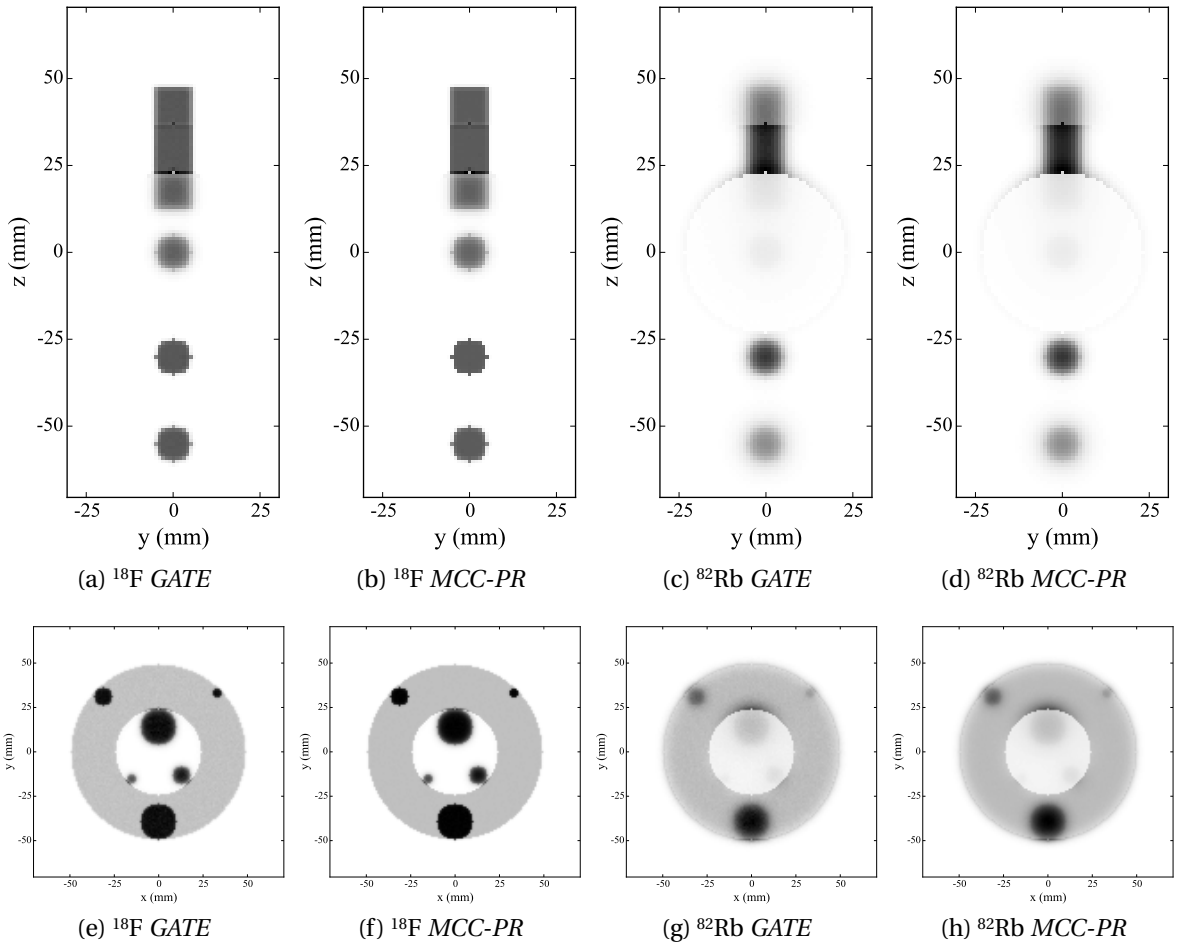
La figure 5.10 montre les trajectoires de 20 positons émis par du ^{82}Rb dans l'eau, simulés par *GATE* et par *MCC-PR*. On peut observer que les trajectoires ont le même type d'allure, avec de grandes courbes lisses au début, puis une trajectoire erratique sur la fin. Si on se réfère aux tables de la figure 5.4, les positons s'éloignent plus de leur point d'émission pendant leurs premiers pas, qui sont les plus grands et ceux où l'angle de diffusion est le plus faible, ce qui est en accord avec ce qu'on observe ici.



(a) (b)

FIGURE 5.11 – Histogramme des positions et directions suivant l'axe z des positons des espaces de phase obtenus avec *GATE* et *SMC-PR*.

La figure 5.11 présente les histogrammes des positions et directions suivant l'axe z des positons des espaces de phase obtenus avec les simulations *GATE* et *SMC-PR* du fantôme présenté dans la figure 5.5. L'histogramme de position présente peu d'écart entre les deux simulations. On peut seulement noter qu'au niveau du poumon il y a plus de positons avec la simulation *GATE*. L'histogramme des directions ne présent quant à lui aucun écart notable.



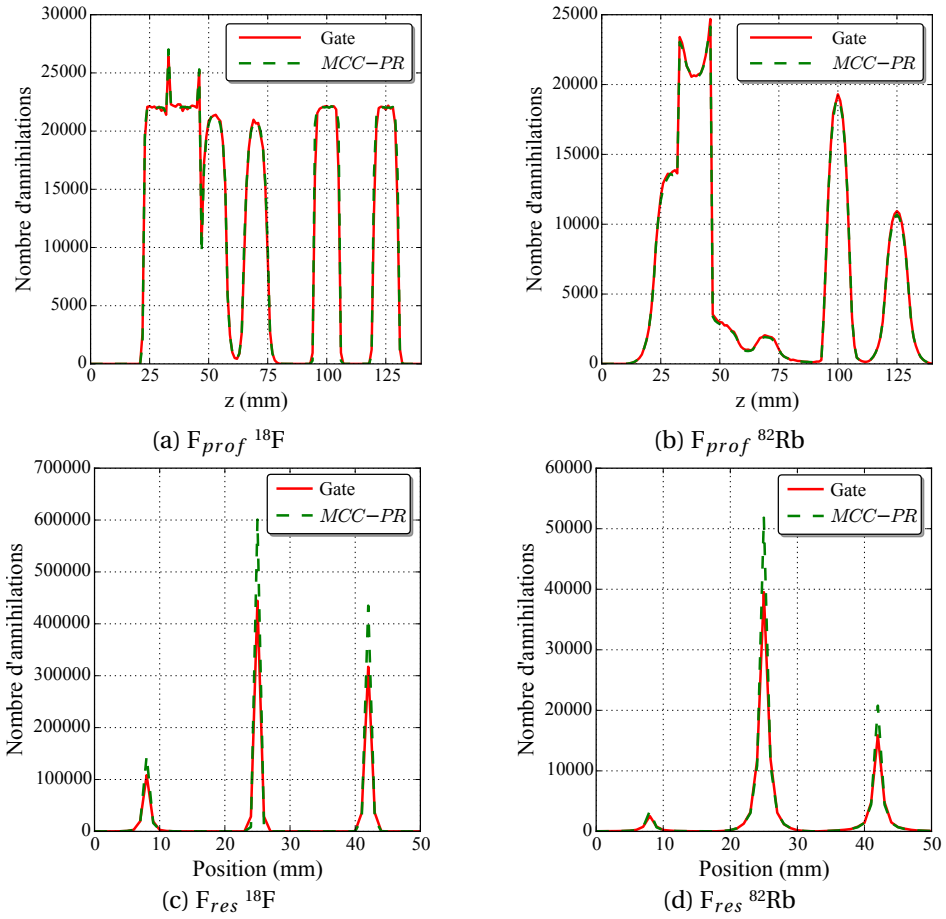
(a) ^{18}F *GATE* (b) ^{18}F *MCC-PR* (c) ^{82}Rb *GATE* (d) ^{82}Rb *MCC-PR*

(e) ^{18}F *GATE* (f) ^{18}F *MCC-PR* (g) ^{82}Rb *GATE* (h) ^{82}Rb *MCC-PR*

FIGURE 5.12 – Distributions des annihilations des positons émis par du ^{18}F et du ^{82}Rb dans les fantômes F_{prof} et F_{cont}, simulés avec *GATE* et *MCC-PR*.

La figure 5.12 montre les distributions des annihilations des positons émis par du ^{18}F et du ^{82}Rb dans les fantômes F_{prof}, F_{cont}, obtenues avec les simulations *GATE* et *MCC-PR*. On peut voir qu'avec le ^{18}F, le parcours des positons modifie peu la distribution des émissions. Il y a principalement une légère perte de contraste des sources placées dans le tissu pulmonaire, ainsi qu'une surintensité au niveau des interfaces avec celui-ci. Les deux simulations donnent des images qui sont visuellement très proches. On peut cependant observer que la sphère placée dans le tissu pulmonaire a des contours mieux définis avec la simulation *MCC-PR*, et donc qu'une partie des positons qui aurait dû la quitter ne l'ont pas fait. Avec l'isotope ^{82}Rb, la différence entre la distribution des annihilations et celle des émissions, est beaucoup plus importante. Les sources dans l'eau perdent beaucoup de contraste tandis que celles dans le tissu pulmonaire disparaissent quasiment.

Des profils tracés dans les distributions des annihilations des fantômes F_{prof} et F_{res} sont présentés dans la figure 5.13. Avec le fantôme F_{res}, on peut noter des différences importantes au centre des sources ponctuelles. Avec la méthode *MCC-PR*, les profils dépassent d'environ 30 % ceux obtenus avec *GATE*. Cependant, avec une distribution plus réaliste, comme celle du fantôme F_{prof}, les deux courbes se superposent parfaitement, ce qui montre que les approximations de la méthode *MCC-PR* sont valides dans un tel contexte d'utilisation.



(a) F_{prof} ^{18}F (b) F_{prof} ^{82}Rb

(c) F_{res} ^{18}F (d) F_{res} ^{82}Rb

FIGURE 5.13 – Profils dans les distributions des annihilations des positons émis par du ^{18}F et du ^{82}Rb dans les fantôme F_{prof} et F_{res}, simulés avec *GATE* et *MCC-PR*. Dans le fantôme F_{cont}, le profil suit l'axe du scanner, tandis que celui du fantôme F_{res} passe par les trois sources ponctuelles dans le plan perpendiculaire à l'axe du scanner, passant d'abord par la source placée dans le tissu pulmonaire, puis par celle placée dans l'os pour finir par celle qui est dans l'eau.

Le tableau 5.1 référence les temps de simulation avec *GATE* et *MCC-PR*, pour les trois fantômes et les deux isotopes étudiés. Ce tableau présente aussi les rapports des temps de simulation de *GATE* sur les temps de simulation de *MCC-PR*. Les simulations *GATE* ont été exécutées sur un cœur de *CPU* Intel Xeon E5-2680 cadencé à 2.7GHz, tandis que les simulations *MCC-PR* ont été exécutées sur un *GPU* NVIDIA GTX 980 Ti à 1GHz. Les simulations *GATE* nécessitent plusieurs dizaines de minutes pour simuler un million de positons, alors qu'il faut moins d'une seconde avec la méthode *MCC-PR*. En moyenne, le facteur d'accélération avec la méthode *MCC-PR*, est de l'ordre de 7000, permettant d'effectuer cette simulation pendant la reconstruction sans trop impacter les temps de reconstruction.

En résumé, les simulations avec la méthode *MCC-PR* fournissent des résultats très proches de ceux obtenus avec les mêmes simulations exécutées sur la plate-forme *GATE*, suffisants pour une correction du parcours du positon dans la reconstruction TEP.

TABLEAU 5.1 – Temps de simulation du parcours d'un million de positons émis par du ^{18}F et du ^{82}Rb dans les trois fantômes étudiés, simulés avec *GATE* et avec la méthode *MCC-PR*.

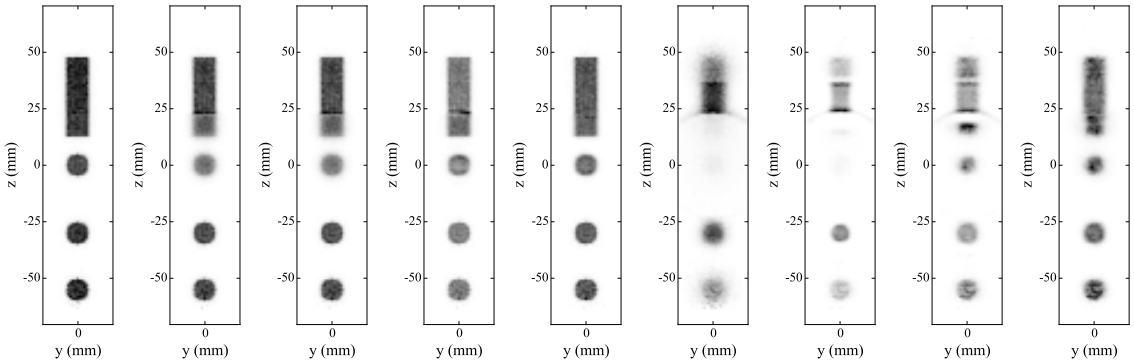
Émetteur	Fantôme	*GATE*	*MCC-PR*	Rapport (*GATE*/*MCC-PR*)
^{18}F	F_{prof}	44 minutes	0.36 secondes	7201
	F_{cont}	41 minutes	0.32 secondes	7654
	F_{res}	38 minutes	0.4 secondes	5715
^{82}Rb	F_{prof}	66 minutes	0.75 secondes	5250
	F_{cont}	93 minutes	0.67 secondes	8367
	F_{res}	63 minutes	0.79 secondes	4777

CPU : Intel Xeon E5-2680 à 2.7GHz
GPU : NVIDIA GTX 980 Ti à 1GHz

5.6.2 Reconstructions

Les coupes des reconstructions du fantôme F_{prof} à l'itération 30, avec les trois jeux de données et les trois méthodes de corrections du parcours du positon, sont présentées dans la figure 5.14. La reconstruction des données *back-to-back*, qui ne sont pas impactées par le parcours du positon, montre bien les différentes sphères et la barre avec les mêmes intensités. On peut noter que les reconstructions non corrigées fournissent des images très proches de la distribution des annihilations, présentées dans la figure 5.12. Sans correction, on reconstruit bien la distribution des annihilations. La méthode de correction basée sur une convolution invariante fournit une image quasiment identique à celle non corrigée pour les données ^{18}F. Avec les données ^{82}Rb, la sphère et la portion de la barre qui sont dans l'eau ont des contours mieux définis, mais ce qui se trouve dans l'os semble toujours surestimé, et ce qui se trouve dans le tissu pulmonaire, sous-estimé. On peut aussi noter de forts artefacts au niveau des transitions entre les matériaux. La méthode de convolution variante semble quant à elle restaurer les intensités correctement, mais présentes aussi de forts artefacts aux transitions entre les matériaux, surtout sur la transition poumons-os. Avec la méthode *MCC-PR*, l'image reconstruite des données ^{18}F est quasiment identique à celle des données *back-to-back*. L'image obtenue avec les données ^{82}Rb est moins bien définie, mais on retrouve les bonnes valeurs de contraste dans les différentes sphères, sans artefact dans la barre aux transitions entre matériaux.

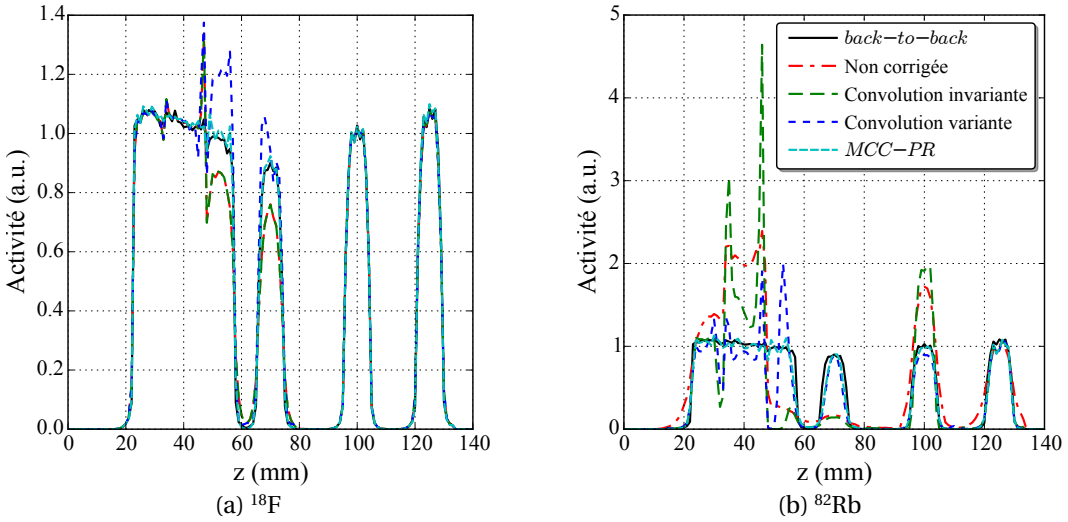
Pour voir plus précisément les variations de contraste et les artefacts sur les transitions entre matériaux, la figure 5.15 montre les profils tracés le long de l'axe du scanner dans les images reconstruites du fantôme F_{prof} à l'itération 30. Dans la partie de gauche, le profil passe le long de la barre. On peut noter, dans cette partie du profil, d'importants écarts entre les corrections par convolution invariante et variante. Ces artefacts sont causés par la non-isotropie de la distribution des annihilations au voisinage des transitions entre matériaux, qui ne sont pas prises en compte dans les deux méthodes de correction par convolution avec des noyaux précalculés. La méthode de convolution variante semble surestimer l'activité des sources situées dans le tissu pulmonaire avec le ^{18}F. La correction par convolution invariante améliore la définition des bords des sphères, relativement au profil de la reconstruction non corrigée, mais elle donne des surestimations dans l'os et des sous-estimations dans le tissu pulmonaire, encore plus importantes que sans correction. Pour les deux jeux de données, les profils



(a) *back-to-back* (b) ^{18}F non corrigée (c) ^{18}F convolution invariante (d) ^{18}F convolution variante (e) ^{18}F *MCC-PR* (f) ^{82}Rb non corrigée (g) ^{82}Rb convolution invariante (h) ^{82}Rb convolution variante (i) ^{82}Rb *MCC-PR*

FIGURE 5.14 – Coupes frontales des reconstructions du fantôme F_{prof} à l'itération 30.

de la méthode *MCC-PR* se superposent presque complètement aux profils des données *back-to-back*. Sur le profil des données ^{82}Rb, on peut quand même noter que les transitions sont moins franches sur la partie du profil qui passe sur la sphère centrale, dans du tissu pulmonaire.

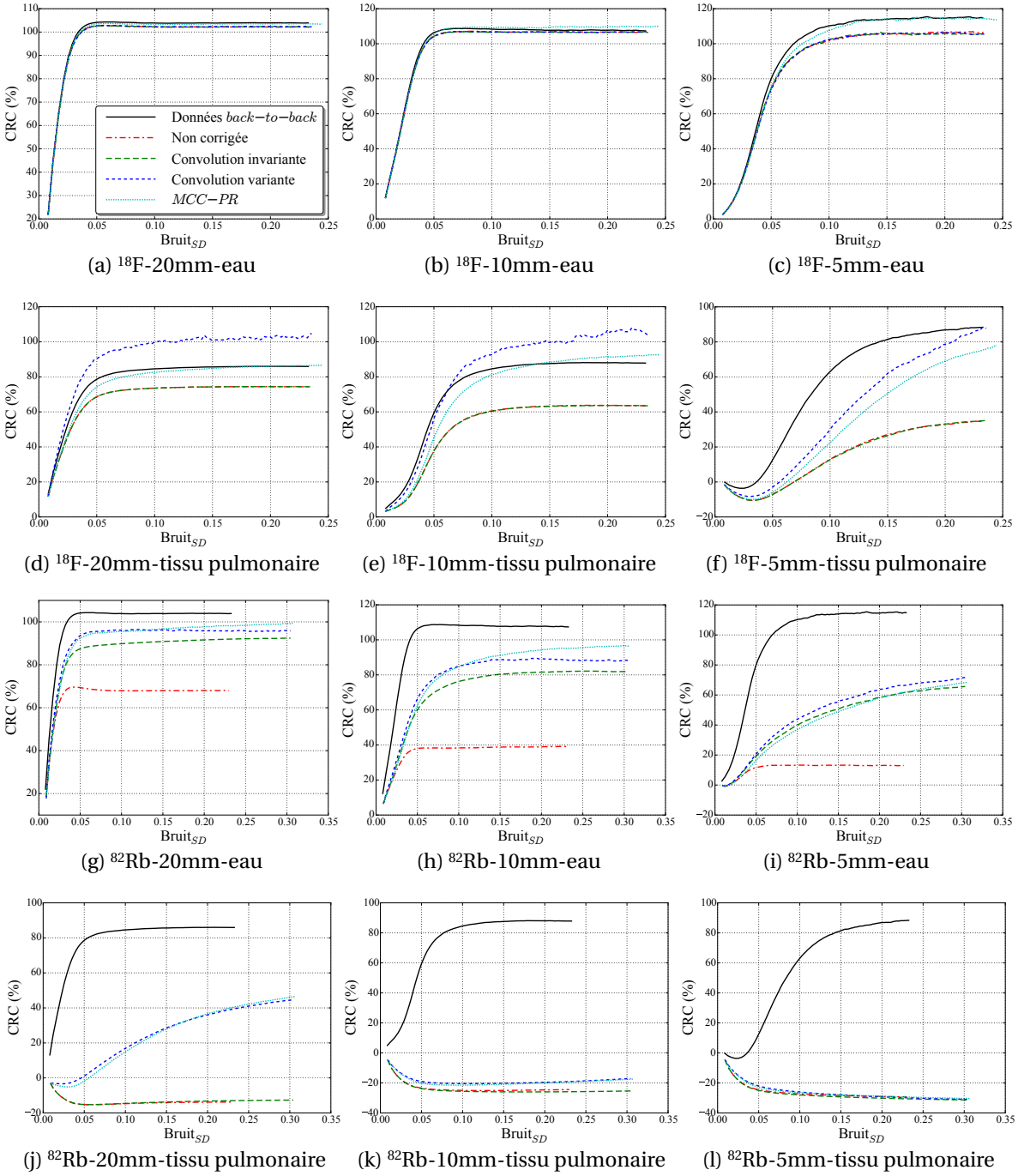


(a) ^{18}F (b) ^{82}Rb

FIGURE 5.15 – Profil dans les reconstructions du fantôme F_{prof}, reconstruit à partir des données $back-to-back$ avec la ligne noire pleine dans les deux figures, avec les données ^{18}F dans la figure (a) et les données ^{82}Rb dans la figure (b).

5.6.2.1 Contraste et bruit

Les graphiques de la figure 5.16 montrent les CRC mesurés dans les sphères chaudes de 20, 10 et 5 mm placées dans l'eau et dans l'air, en fonction du $Bruit_{SD}$ dans le fond. Avec les données ^{18}F et les sphères placées dans l'eau, toutes les méthodes donnent des courbes proches de celles données par les reconstructions des données sans parcours du positon. Dans le tissu pulmonaire, sans correction et avec correction par convolution invariante, les courbes se superposent parfaitement. On ne peut donc pas espérer améliorer le contraste avec la méthode par convolution invariante. Dans le tissu



(a) ^{18}F-20mm-eau (b) ^{18}F-10mm-eau (c) ^{18}F-5mm-eau

(d) ^{18}F-20mm-tissu pulmonaire (e) ^{18}F-10mm-tissu pulmonaire (f) ^{18}F-5mm-tissu pulmonaire

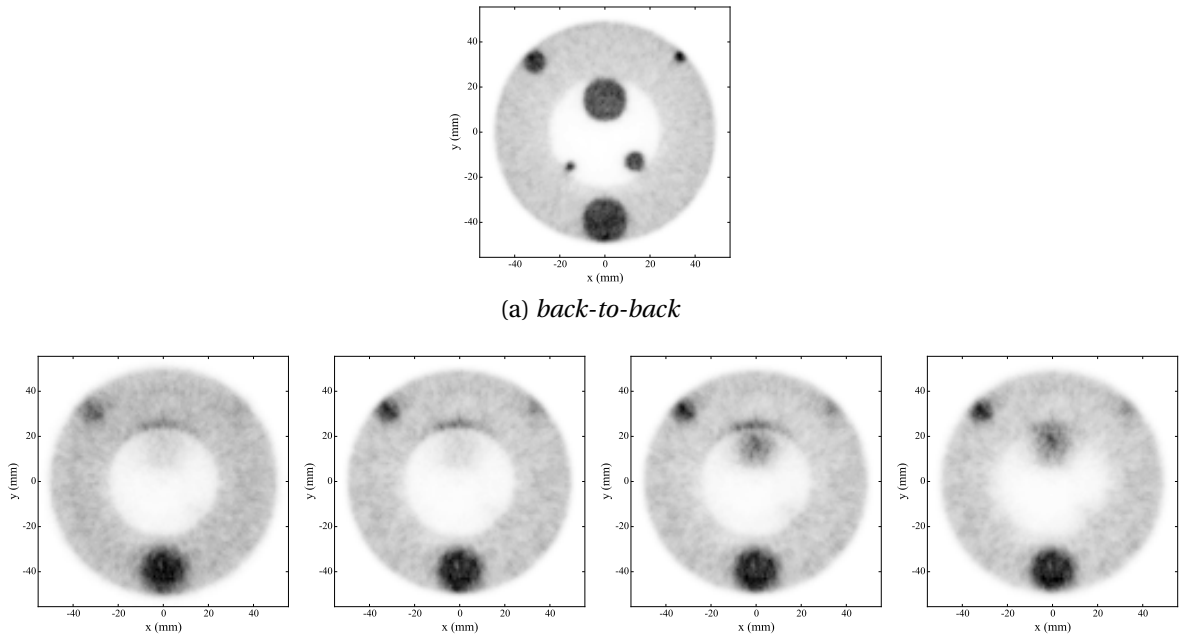
(g) ^{82}Rb-20mm-eau (h) ^{82}Rb-10mm-eau (i) ^{82}Rb-5mm-eau

(j) ^{82}Rb-20mm-tissu pulmonaire (k) ^{82}Rb-10mm-tissu pulmonaire (l) ^{82}Rb-5mm-tissu pulmonaire

FIGURE 5.16 – Courbes du CRC en fonction du $Bruit_{\mathrm{SD}}$ mesurés dans les reconstructions des trois jeux de données du fantôme F_{cont}.

pulmonaire, la méthode de convolution variante donne les meilleurs contrastes, et suivi de près par la méthode *MCC-PR*. Avec les données ^{82}Rb, les trois méthodes de correction du parcours du positon donnent des courbes très proches. Elles ne permettent pas d'atteindre les valeurs de contrastes obtenues avec les données *back-to-back* mais les améliorent significativement. Il n'est pas étonnant de retrouver les mêmes courbes avec les trois méthodes. En effet, les sphères sont placées dans des zones homogènes constituées d'eau. Les méthodes de convolution utilisent donc le même noyau dans ces

régions. La méthode *MCC-PR* estime une distribution des annihilations qui doit être très proche de celle qui a été précalculé pour les deux autres méthodes de correction. Les méthodes de convolution variante et *MCC-PR* permettent de retrouver une partie du contraste dans la sphère de 20mm placée dans le tissu pulmonaire. En revanche, pour les sphères plus petites de 10 et 5mm, aucune méthode de correction n'obtient un meilleur contraste que ceux obtenus sans correction. Sur le fantôme F_{prof}, la sphère de 10mm placée était restaurée avec les méthodes de convolution variante et *MCC-PR*, ce qui n'est pas le cas ici. Cela peut être expliqué par la présence d'activité autour des sphères dans le fantôme F_{cont} alors que les sources du fantôme F_{prof} sont placées dans un fond totalement inactif.



(a) *back-to-back*

(b) Non corrigée (c) Convolution invariante (d) Convolution variante (e) *MCC-PR*

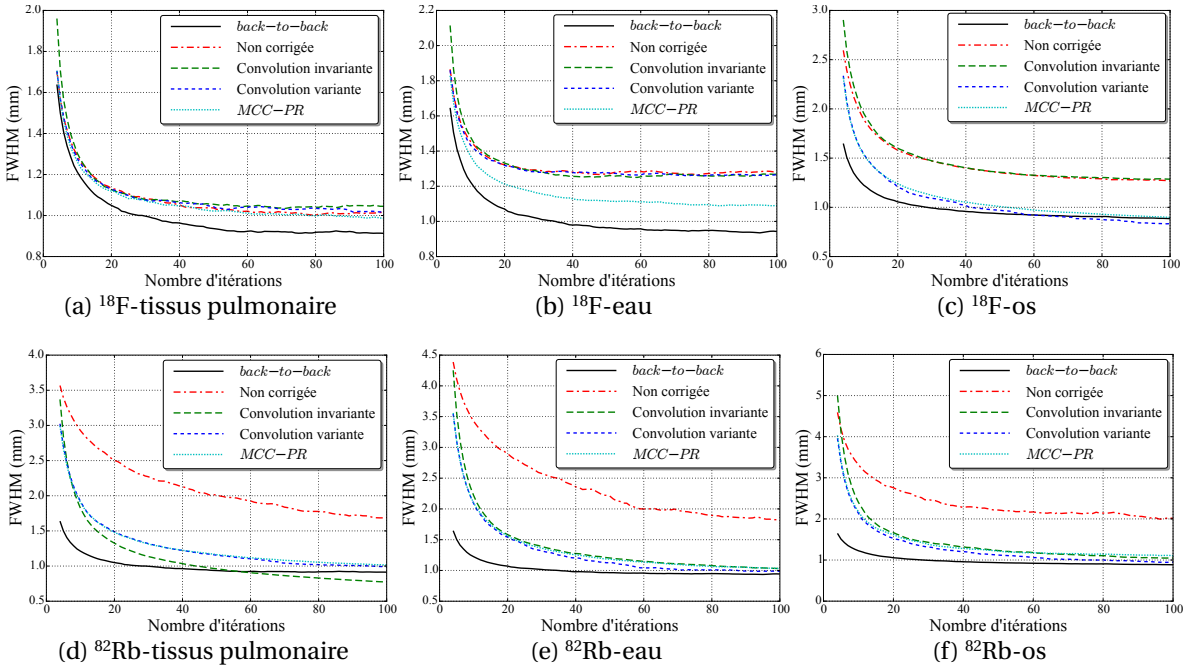
FIGURE 5.17 – Coupes transverses des reconstructions des données ^{82}Rb du fantôme F_{cont}.

La figure 5.17 présente des coupes des reconstructions des données ^{82}Rb du fantôme F_{cont}. Comme nous l'avons noté avec le fantôme F_{prof}, les méthodes de correction par convolution invariante ou variante, créent des artefacts au niveau des transitions entre matériaux, bien visibles ici au-dessus de la sphère de 20 mm placée dans le tissu pulmonaire. Cet artefact n'est pas présent avec la méthode *MCC-PR*.

Dans les régions homogènes, loin de toute transition entre différents matériaux, la méthode *MCC-PR* ne peut pas surpasser la méthode de convolution variante parce que cette dernière dispose d'une estimation de la distribution des annihilations qui est précise, parce qu'elle est estimée directement par SMC et est peu bruitée. Cependant, à proximité des interfaces, l'utilisation d'un noyau de convolution variant isotrope ne suffit pas, ce qui cause des erreurs comme nous avons pu le constater précédemment, ce qui n'est pas le cas avec notre approche. En pratique, le corps d'un patient ne présente pas autant de zones homogènes qu'un fantôme et aussi moins de transitions aussi brutales de la densité, comme nous l'avons utilisé dans cette étude.

5.6.2.2 Résolution

Les mesures de résolutions sur le fantôme F_{res} sont présentées dans la figure 5.18. Dans le cas du ^{18}F, toutes les méthodes donnent des résultats équivalents dans le tissu pulmonaire en convergeant vers une *FWHM* de 1 mm, alors que dans l'eau la méthode *MCC-PR* surpasse légèrement les autres. Dans l'os, les méthodes *MCC-PR* et la convolution variante donnent les mêmes *FWHM* et surpassent la reconstruction non corrigée et la correction par convolution invariante. Avec l'isotope ^{82}Rb, toutes les corrections apportent le même gain de résolution relativement à la reconstruction non corrigée. Il faut cependant relativiser ces résultats avec les profils mesurés dans les images reconstruites avec 100 itérations, présentés dans la figure 5.19. En effet, bien que la méthode de convolution invariante donne des résultats équivalents aux autres méthodes sur les courbes d'estimation de la résolution, si on observe la crête de gauche sur les deux profils, on peut constater qu'elle a des amplitudes beaucoup plus faibles que celles obtenues avec les autres corrections. Sur les profils, la méthode *MCC-PR* est celle qui fournit les crêtes dont les amplitudes sont les plus proches de celles de la reconstruction avec les données *back-to-back*, suivi de la convolution variante, puis de la reconstruction non corrigée et enfin de la convolution invariante.



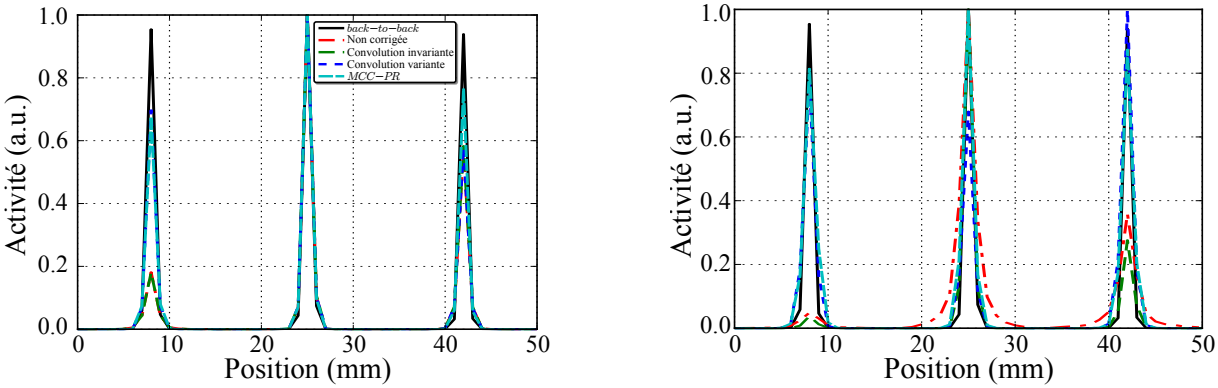
(a) ^{18}F-tissus pulmonaire (b) ^{18}F-eau (c) ^{18}F-os

(d) ^{82}Rb-tissus pulmonaire (e) ^{82}Rb-eau (f) ^{82}Rb-os

FIGURE 5.18 – Mesures de la résolution sur le fantôme F_{res}.

5.6.2.3 Temps de reconstruction

Les reconstructions du fantôme F_{cont}, sans correction du parcours du positon, s'exécutaient en 17,6 secondes par itération avec les données ^{18}F et en 16,1 secondes par itération avec les données ^{82}Rb. Les temps de calcul par itération, associés à chacune des méthodes de correction du parcours du positon, pour les deux jeux de données, sont présentés dans le tableau 5.2. La méthode de correction par convolution invariante s'exécute en un temps négligeable relativement au temps d'exécution



(a) ^{18}F (b) ^{82}Rb

FIGURE 5.19 – Profils passant par les sources ponctuelles du fantôme F_{res} dans les reconstructions des données *back-to-back*, ^{18}F et ^{82}Rb.

d'une itération sans correction. Cela s'explique par la faible étendue du noyau de convolution associé au parcours de positons du ^{18}F et du ^{82}Rb dans l'eau. Avec la correction par convolution variante, la correction augmente le temps de calcul par itération de 2 % avec les données ^{18}F et de 26 % avec les données ^{82}Rb. Le coût en calcul est plus faible pour la correction du parcours des positons du ^{18}F pour les mêmes raisons que précédemment. Avec le ^{82}Rb, l'opération la plus coûteuse est la convolution des *voxels* de tissu pulmonaire, car le noyau de convolution est beaucoup plus étendu, ce qui implique de lire les valeurs d'un grand nombre de *voxels* pour chaque *voxel* de tissu pulmonaire. La correction avec la méthode *MCC-PR* implique une augmentation du temps de reconstruction de 83 % et de 181 % avec les données ^{18}F et ^{82}Rb respectivement. La correction du parcours des positons émis par du ^{82}Rb est plus coûteuse, en temps de calcul, parce que ces positons ont, en moyenne, une énergie cinétique plus importante, impliquant d'une part qu'il faut simuler plus de pas jusqu'à annihilation, d'autre part qu'ils traversent plus de *voxels* pour lesquelles il faut à chaque fois lire les matériaux qui les constituent.

TABLEAU 5.2 – Temps supplémentaires, en seconde, nécessaires à chaque itération pour corriger le parcours du positon. Les temps sont donnés pour la reconstruction du fantôme F_{cont}.

Méthode de correction	^{18}F	^{82}Rb
Convolution invariante	0,02 s	0,06 s
Convolution variante	0,42 s	4,18 s
MCC-PR	14,6 s	29,2 s

5.7 Discussion et conclusion

Dans ce chapitre, nous avons proposé une méthode de correction du parcours du positon, fondée sur une SMC exécutée à la volée pendant le processus de reconstruction. Elle permet de modéliser dans l'opération de projection, les variations spatiales et l'anisotropie de la *PSF* associée au parcours des positons. Nous avons montré qu'elle permet d'améliorer la qualité des images reconstruites sans

introduire les artefacts que l'on obtient avec les méthodes basées sur des convolutions par des noyaux isotropes. Elle implique une augmentation du temps de reconstruction, qui peut aller jusqu'à un facteur trois, mais qui reste suffisamment raisonnable pour envisager de l'appliquer en routine clinique. De plus, l'utilisation de plusieurs *GPU* permettrait d'accélérer facilement cette méthode, étant donné la forte capacité des méthodes de Monte-Carlo à être parallélisées.

Dans l'évaluation de la méthode *MCC-PR*, nous avons fixé la valeur du nombre de positons à simuler comme étant égale au nombre de coïncidences dans le jeu de données *list-mode*. Ce choix étant totalement arbitraire, une perspective de travaux futurs pourrait être d'évaluer l'impact du nombre de positons simulés sur la qualité des images reconstruites.

Durant la navigation d'un positon avec la méthode *MCC-PR*, un nombre important de pas très petits sont calculés juste avant l'annihilation. Ces pas étant très courts et l'angle de diffusion important, ils participent peu à l'éloignement d'un positon de son point d'émission. Une autre perspective d'extension de l'étude réalisée dans ce chapitre, pourrait être d'évaluer l'impact d'un changement de la valeur limite de la taille du pas sur la qualité de la simulation, la qualité de la correction des reconstructions et sur les temps de simulation, et donc les temps de reconstruction. Finalement, une dernière perspective serait d'évaluer cette méthode avec des données anthropomorphiques simulées dans un premier temps, puis des données cliniques réelles dans un second temps.

Dans des travaux futurs, nous évaluerons plus précisément l'impact des simplifications introduites dans la simulation de positons avec la méthode *MCC-PR* ainsi que celui de l'utilisation d'un projecteur *unmatched* sur la qualité d'image et le temps de reconstruction.

Conclusion et perspectives

La TEP est une modalité d'imagerie médicale essentielle pour le diagnostic et pronostic dans plusieurs branches de la médecine. Elle souffre cependant de limites importantes, comme une faible résolution spatiale et un niveau de bruit élevé, qui sont les conséquences de la nature stochastique du processus d'émission des positons et des nombreux effets intervenant durant le processus d'acquisition des données. La correction complète de tous ces effets est nécessaire pour obtenir une reconstruction qui soit qualitative et quantitative. Cependant, l'ensemble de ces corrections nécessite une puissance de calcul qui n'est atteignable qu'avec un large *cluster*, ce qui est incompatible avec les applications en clinique. Depuis quelques années, l'utilisation des *GPU* est devenue la solution privilégiée pour les problèmes de calcul intensif dans le domaine scientifique. Ils permettent en effet de disposer d'une puissance de calcul très importante dans une seule machine et pour un coût bien inférieur à celui d'une solution équivalente basée sur des *CPU*. Dans ce contexte, l'objectif de cette thèse a été de proposer des approches permettant d'améliorer la qualité des images reconstruites en exploitant la puissance de calcul des *GPU*, pour conserver des temps de reconstruction compatibles avec les applications cliniques. Il existe des méthodes de correction efficaces pour la plupart des effets intervenant pendant l'acquisition des données TEP, comme les temps-morts, les coïncidences fortuites avec la méthode de la fenêtre retardée, l'atténuation avec l'utilisation de données TDM ou IRM, la diffusion avec les méthodes *SSS* et *MSS*, et la normalisation. Les effets physiques et géométriques associés au détecteur sont généralement modélisés dans une matrice système préestimée. Cependant, cette matrice possède des dimensions très importantes ce qui pose des problèmes de stockage, contraint à utiliser des méthodes de compression et pose des problèmes de flexibilité qui empêchent de modifier certains paramètres de la reconstruction comme la taille des *voxels* ou la position et la taille du champ de vue, ou encore d'intégrer certaines informations comme la *POI* par exemple. Pour remédier à ces problèmes, des projecteurs calculant la réponse du détecteur à la volée pendant la reconstruction peuvent être utilisés. Cependant, aujourd'hui, il n'existe pas de projecteur intégrant une modélisation précise de tous les effets associés au détecteur. Un autre effet souvent négligé est celui associé au parcours des positons, principalement parce qu'avec les isotopes les plus couramment employés en routine clinique, la perte de résolution qui lui est associée est inférieure à la résolution intrinsèque du système. Cependant, des isotopes pour lesquels le parcours du positon est beaucoup plus important, sont aussi utilisés et pourraient voir leur usage s'étendre pour des raisons de coût de production. Actuellement, il n'existe pas de méthodes permettant de modéliser précisément les effets associés aux parcours des positons de tels isotopes, en tenant compte des hétérogénéités des matériaux. Dans ce manuscrit, nous nous sommes donc intéressés à modéliser la réponse du détecteur dans un projecteur, à la correction du parcours du positon et à accélérer la reconstruction en exploitant la puissance de calcul des *GPU*.

L'utilisation d'une implémentation mono-*GPU* n'est généralement pas suffisante pour permettre l'exécution d'une reconstruction totalement corrigée dans des temps compatibles avec les applica-

tions cliniques. Toutefois, il est possible d'intégrer dans une seule machine plusieurs *GPU*. Afin d'exploiter la puissance de calcul de tels systèmes pour accélérer les reconstructions TEP, nous avons proposé une méthode permettant de paralléliser la reconstruction en TEP sur une plate-forme muli-*GPU* en découpant le volume de reconstruction en morceaux, chacun de ces sous-volumes étant ainsi traité par un *GPU*. Deux variantes de cette méthode ont été développées, une première qui permet de répartir de manière équilibrée la quantité de données hébergée sur la mémoire de chaque *GPU* et une seconde qui répartit la charge de travail équitablement sur les *GPU*. Nous avons montré que cette seconde approche permet d'exploiter plus efficacement la puissance de calcul des *GPU* que la méthode standard qui fait état de l'art et qui repose sur un découpage des données *list-mode*. Par conséquent, la méthode proposée fournit une reconstruction plus rapide et avec des communications plus faibles entre les *GPU*. Par rapport à une reconstruction *LM-OSEM* sur un seul *GPU*, une reconstruction utilisant 24 *GPU* a été accélérée d'un facteur 20 avec notre méthode, tandis que la méthode standard n'a atteint qu'un facteur de 12.

Aujourd'hui, il n'existe pas de projecteur qui permet de modéliser précisément les effets physiques et géométriques associés au détecteur d'un scanner en TEP. Nous avons proposé deux nouveaux projecteurs, nommés $IRIS_{mesure}$ et $IRIS_{analytique}$, qui reposent sur une approche multiligne pour estimer précisément la réponse du système. Ils utilisent de modèles de la *IDRF* pour générer des lignes aléatoires qui une fois accumulées donnent la représentation physique d'une *LOR*. Le projecteur $IRIS_{mesure}$ utilise une estimation échantillonnée des *IDRF*, tandis que le projecteur $IRIS_{analytique}$ utilise un modèle analytique de ces *IDRF*. Nous avons montré que ces deux projecteurs permettent d'obtenir des images reconstruites de meilleure qualité, avec des contrastes jusqu'à 20 % supérieurs et des résolutions au moins équivalentes à ceux obtenus avec d'autres projecteurs de l'état de l'art, comme ceux qui modélisent la réponse du système avec des fonctions Gaussiennes. Le projecteur $IRIS_{analytique}$ surpasse légèrement le projecteur $IRIS_{mesure}$ parce qu'il utilise un modèle analytique non-bruité des *IDRF*, ce qui n'est pas le cas de ce second projecteur où les *IDRF* sont estimées par SMC. Ces projecteurs s'exécutent aussi de 2 à 9 fois plus rapidement que ceux basés sur un modèle Gaussien.

La modélisation de la réponse du système avec une matrice préestimée et stockée permet d'exploiter des méthodes précises mais coûteuses en temps de calcul, comme les SMC. L'approche $S(MC)^2PET$ proposées par [Matthieu, 2014] utilise une SMC complète du détecteur du scanner pour construire une matrice système, ensuite stockée avec des compressions exploitant les symétries du détecteur et l'aspect creux de la matrice. Nous avons évalué les projecteurs *IRIS* face à cette approche, avec trois matrices système différentes, chacune ayant une qualité statistique donnée. Nous avons montré que les projecteurs *IRIS* surpassent la méthode $S(MC)^2PET$, tant en matière de qualité d'image que de temps de reconstruction, avec un contraste jusqu'à 15 % supérieur, une résolution 20 % inférieure et un temps d'exécution jusqu'à 3,5 fois plus court.

Nous avons proposé une nouvelle méthode de correction du parcours du positon, basée une SMC simplifiée et accélérée sur *GPU*, exécutée pendant la reconstruction. Celle-ci permet de prendre en compte l'ensemble des hétérogénéités des matériaux dans le corps d'un patient. Nous avons montré que cette approche apporte une amélioration du contraste et de la résolution, sans introduire d'artefacts, typiques des méthodes de correction standards qui ne prennent pas en compte les variations

locales des matériaux. De plus, ses temps de calcul sont compatibles avec la routine clinique.

Plusieurs évolutions des projecteurs *IRIS* sont possibles. Ceux-ci s'appuient sur l'accumulation d'une multitude de lignes aléatoires pour modéliser la réponse du système. Il est toutefois possible d'effectuer cette estimation en accumulant des lignes pondérées et tracées à des endroits déterminés, comme proposé dans les travaux de [Moehrs *et al.*, 2008]. L'utilisation d'une telle approche avec le projecteur *IRIS* pourra être étudiée, pour évaluer ses bénéfices sur la qualité d'image et sur les temps de reconstruction. Ensuite, la correction de la non-colinéarité des photons d'annihilation est un effet qui est totalement négligé. Son impact sur la résolution croit avec le diamètre du détecteur et pour cette raison il a un effet mineur avec les scanners précliniques et cérébraux. Dans le cas de scanner corps entier sa *PSF* peut atteindre 2 mm de *FWHM*. Le projecteur *IRIS* pourrait modéliser cet effet en remplaçant les droites accumulées par la figure géométrique décrivant les positions possibles d'annihilation, connaissant les points des détections (générées aléatoirement avec les modèles de *IDRF*) et l'angle de non-colinéarité (généré aléatoirement avec un modèle de la distribution de l'angle de non-colinéarité). Les modèles des *IDRF* se basent sur des données issues de SMC exécutées sur *GATE*, c'est-à-dire sans modélisation du parcours des photons optiques dans les cristaux jusqu'aux TPM. Une modélisation de cet effet semble nécessaire pour pouvoir envisager d'exploiter les projecteurs *IRIS* sur des données cliniques, ce que nous envisageons de développer dans de futurs travaux.

La modélisation de la réponse du détecteur dans la reconstruction permet d'améliorer le rapport contraste sur bruit des images, mais peut aussi introduire des artefacts de Gibbs dans ces images. Il est d'ailleurs possible d'obtenir des rapports contraste sur bruit très élevés en augmentant artificiellement la taille de la *PSF* du détecteur. Toutefois, cela entraîne aussi une augmentation des artefacts de Gibbs. L'utilisation du rapport contraste sur bruit pour évaluer la qualité des images reconstruites semble être insuffisante. Il serait intéressant d'évaluer les projecteurs *IRIS* avec une figure de mérite tenant aussi compte de ces artefacts de Gibbs.

Dans notre étude de la correction du parcours du positon, nous nous sommes basés sur une modélisation de cet effet seulement dans l'étape de projection et donc sur l'utilisation d'un projecteur dit *unmatched*. Ce choix a été fait dans l'objectif de conserver des temps de reconstruction raisonnables pour une application clinique, grâce à une charge de travail moins importante à chaque itération et en évitant de ralentir trop la convergence de la reconstruction. Toutefois, ce choix implique des conséquences sur la qualité des images reconstruites, ce que nous évaluerons dans de futurs travaux.

Dans cette thèse, la diffusion dans le patient, le temps-mort et les coïncidences fortuites ont toujours été exclus de nos données. Pour obtenir une reconstruction entièrement corrigée, il sera nécessaire d'intégrer des corrections de ces effets ainsi que nos méthodes dans une reconstruction, et d'évaluer si toutes ces corrections n'interfèrent pas entre elles et si elles permettent d'améliorer les images reconstruites. Nos travaux ont entièrement reposé sur l'utilisation de données simulées, ce qui nous a permis de contrôler exactement les volumes d'activités et de matériaux, ainsi que les effets incluent dans les données fournies aux reconstructions. Une perspective intéressante serait d'utiliser des données réelles dans nos reconstructions. Une ouverture de nos travaux pourrait être d'adapter nos méthodes à une reconstruction corrigeant les mouvements du patient (respiratoires, cardiaques…), et d'utiliser des techniques de super-résolution, comme celle proposée par [Wallach, 2011], pour améliorer la qualité des images reconstruites.

La solution de calcul parallèle que nous avons exploitée, repose sur l'utilisation des *GPU* de la marque Nvidia et l'*API* de programmation *CUDA*. Toutefois, ce domaine change rapidement et le choix que nous avons fait devra être remis en question au fil de ses évolutions. Par exemple, on a vu apparaître des cartes de calcul comme le Xeon Phi de Intel, dédiées au calcul massivement parallèle, mais avec une architecture qui se rapproche de celle des *CPU*. De leur côté les *CPU* possèdent de plus en plus de cœurs, ce qui conduira peut-être ces deux types d'architectures à un jour fusionner.

Pour conclure, nous avons proposé des projecteurs qui permettent de calculer à la volée la réponse du détecteur, aussi précisément qu'une matrice stockée et sans les limites liées à son stockage et à sa rigidité. Une méthode de parallélisation de la reconstruction sur une plate-forme multi-*GPU*, a été proposée, permettant d'exploiter plus efficacement la puissance de calcul disponible, tout en réduisant la quantité d'informations communiquées entre les *GPU*, par rapport à l'approche classique. Enfin, une nouvelle méthode de correction du parcours du positon a été introduite, permettant d'améliorer les images reconstruites sans introduire d'artefacts typiques de ce type de correction. Finalement, ce travail est un pas de plus vers une reconstruction TEP entièrement corrigée, s'exécutant dans des temps compatibles avec les contraintes cliniques, pour permettre un diagnostic et un suivi thérapeutique plus précis dans de nombreuses branches de la médecine.

Annexe A

Définitions

A.1 Glossaire

^{18}F-FDG Fluorodésoxyglucose

accélérateur linéaire Accélérateur de particule linéaire permettant la production de certains isotopes émetteurs de particules β^+ utilisés en TEP

API Interface de programmation ou *application programming interface* en anglais

ART *Algebraic reconstruction technique*

bin Élément d'un sinogramme

Bq Becquerel

CDRF Fonction de réponse en coïncidence du détecteur ou *coincidence detector response function* en anglais

cluster Ferme de calcul ou grappe d'ordinateurs

cœur Unité de calcul des *CPU* et *GPU*

coïncidence Couple de *singles* ayant été détectés dans un laps de temps défini

coïncidence diffusée Coïncidences issues d'une unique annihilation et dont les photons γ n'ont été diffusés

coïncidence fortuite Coïncidences à deux photons issues de plusieurs annihilations

coïncidence prompte Coïncidences issues d'une unique annihilation. Inclut les coïncidences coïncidences vraies et coïncidences diffusées.

coïncidence vraie Coïncidences issues d'une unique annihilation et dont les photons γ n'ont pas été diffusés

CPU Microprocesseur ou *central processing unit* en anglais

CUDA *Compute unified device architecture*

cyclotron Accélérateur de particule circulaire. Permet la production de certains isotopes émetteurs de particules β^+ utilisés en TEP

DOI Profondeur d'interaction ou *depth-of-interaction* en anglais

eV Électron-volt

FBP Rétroprojection filtrée ou *filtered back projection* en anglais

FORE *Fourier rebinning*

FOSA *Fourier simple averaging*

FWHM Largeur à mi-hauteur ou *full width at half maximum* en anglais

GATE *Geant4 application for tomographic emission*

GPGPU Calcul générique sur processeurs graphiques ou *general-purpose computing on graphics processing units* en anglais

GPU processeur graphique en anglais *graphics processing unit*)

hit Intéraction d'un photon d'annihilation avec un cristal scintillateur

Hyper-Threading Consiste à créer deux processeurs logiques sur une seule puce, chacun étant doté de ses propres registres de données et de contrôle

IDRF Fonction de réponse intrinsèque du détecteur ou *intrinsic detector response function* en anglais

IRM Imagerie par résonance magnétique

LM-EM *List-mode expectation maximization*

LM-OSEM *List-mode ordered subset expectation maximization*

LOR Ligne de réponse ou *line of response* en anglais

MFR Méthode de la fenêtre retardée

MIMD « Des instructions multiples pour des données multiples » ou *multiple instructions multiple data* en anglais

ML-EM *Maximum likelihood expectation maximization*

MPI *Message passing interface*

MSRB *Multi-slice rebinning*

MSS *Multiple scatter simulation*

multi plate-forme Désigne un programme pouvant fonctionner sur différents systèmes d'exploitation et différents ordinateurs

nœud Élément d'un *cluster*

OpenCL *Open computing language*

OSEM *Ordered subset expectation maximization*

pixel Élément d'une image matricielle

POI Position d'interaction ou *position-of-interaction* en anglais

polarisation Elle est dû au déplacement du barycentre des charges négatives (électrons) par rapport à celui des charges positives (noyaux)

PSF Fonction d'étalement du point ou *point spread function* en anglais

pulse Barycentre des *hits* d'un photon d'annihilation dans un cristal ou un groupe de cristaux scintillateur

S(MC)^2 PET *System Matrix Computation by Monte Carlo simulations in PET*

section efficace C'est une grandeur reliée à la probabilité d'interaction d'une particule pour une réaction physique donnée

***shader* unifié** C'est une unité de calcul dédiée à l'exécution de programme décrivant l'absorption, la diffusion, la réflexion et la diffraction de la lumière ainsi que la gestion des ombrages et de la texture en image de synthèse. Elle peut cependant exécuter des calculs généralistes. C'est l'ancêtre des cœurs *GPU*

SIMD « Une instruction unique pour des données multiples » ou *single instruction multiple data* en anglais

single C'est une *pulse* dont l'énergie se trouve dans un intervalle défini

SiPM Photomultiplicateur en silicium

SISD « Une instruction unique pour une donnée unique » ou *single instruction single data* en anglais

SM Processeur de flux ou *streaming multiprocessor* en anglais

SMC Simulation Monte-Carlo

SRM Matrice de réponse du système ou *system response matrix* en anglais

SSRB *Single-slice rebinning*

SSS *Single scatter simulation*

TDM Tomodensitométrie

TEMP Tomographie d'émission monophotonique

TEP Tomographie par émission de positons

TEP/IRM Système TEP et IRM combiné dans une seule machine

TEP/TDM Système TEP et TDM combiné dans une seule machine

thread C'est un ensemble d'instructions à exécuter

TOF Temps de vol ou *time-of-flight* en anglais

TPM Tube photomultiplicateur

voxel *Pixel* volumétrique

A.2 Liste des symboles

δ Fonction delta de Dirac

$\mathscr{F}$ Transformée de fourrier

Annexe B

Communications

— Autret, A., Bert, J., Strauss, O., et Visvikis, D. (2012). Projector with realistic detector scatter modelling for pet list-mode reconstruction. Dans *IEEE Nuclear Science Symposium Conference Record*.

— Autret, A., Bert, J., Strauss, O., and Visvikis, D. (2013). 3D pet list-mode reconstruction including all information provided by the detector. Dans *Numerical Modeling and Simulation of Inverse Problems in Medical Imaging*.

— Autret, A., Bert, J., Strauss, O., and Visvikis, D. (2013). Accurate fully 3d list-mode pet reconstruction on multi-gpus. Dans *Recherche en Imagerie et Technologies pour la Santé*.

— Autret, A., Bert, J., Strauss, O., et Visvikis, D. (2013). Amélioration qualitative et quantitative de reconstruction tep basée sur architecture graphique. Dans *7èmes Journées du Cancéropôle Grand Ouest*.

— Autret, A., Bert, J., Strauss, O., et Visvikis, D. (2013). Fully 3d pet list-mode reconstruction including an accurate detector modeling on gpu architecture. Dans *International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology et Nuclear Medicine*.

— Autret, A., Bert, J., Strauss, O., et Visvikis, D. (2013). Incorporation of time-of-flight information in pet list-mode reconstruction using a projector with accurate detector psf modeling. Dans *IEEE Nuclear Science Symposium et Medical Imaging Conference Record*.

— Autret, A., Bert, J., Strauss, O., et Visvikis, D. (2013). Qualitative an quantitative improvement in positron emission tomographic reconstruction using graphics processor architecture. Dans *Journée Futur et Ruptures*.

— Autret, A., Moreau, M., Carlier, T., Strauss, O., Bert, J., et Visvikis, D. (2015). Detector modeling in PET list-mode reconstruction : comparison between pre-calculated and on-the-fly computed system matrices. Dans *International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology et Nuclear Medicine*.

— Autret, A., Moreau, M., Carlier, T., Strauss, O., Bert, J., et Visvikis, D. (2015). Detector modeling in PET list-mode reconstruction : comparison between pre-calculated and on-the-fly computed system matrices. Dans *Society of Nuclear Medicine and Molecular Imaging Annual Meeting*.

— Autret, A., Moreau, M., Carlier, T., Strauss, O., Bert, J., et Visvikis, D. (2015). Detector Modeling in PET List-Mode Reconstruction : Comparison Between Pre-calculated and On-the-Fly Computed System Matrices. Dans *IEEE Nuclear Science Symposium et Medical Imaging Conference Record*.

— Bahi, Z., Bert, J., Autret, A., et Visvikis, D. (2012). High performance multi-gpu acceleration for fully 3d list-mode pet reconstruction. Dans *IEEE Nuclear Science Symposium Conference Record*, pages 3390–3393.

— Benhalouche, S., Bert, J., Autret, A., Visvikis, D., Pradier, O., et Boussion, N. (2013). Imaging et radiation therapy : Gate monte carlo simulation of a megavolt cone beam ct. Dans *IEEE Nuclear Science Symposium et Medical Imaging Conference Record*.

— Benhalouche, S., Bert, J., Autret, A., Visvikis, D., Pradier, O., et Boussion, N. (2013). Imaging et radiation therapy : Gate monte carlo simulation of a mv-cbct flat panel with specific application in head et neck cancer. Dans *The future of radiation oncology*.

— Mountris, K., Autret, A., Papadimitroulas, P., Loudos, G., Visvikis, D., et Nikiforidis, G. (2014). Optimization of image-based dosimetry in y90 radioembolization : a monte carlo approach using the gate simulation toolkit. Dans *8th European Conference on Medical Physics*.

Annexe C

Détails sur l'implémentation *GPU* avec *CUDA*

L'implémentation *GPU* avec l'*API* CUDA repose sur des mécanismes et un vocabulaire spécifique développés autour de l'architecture des *GPU* de marque NVIDIA. Premièrement, on distingue trois types de fonctions :

1. *hosts* qui sont des fonctions classiques exécutées par le *CPU*.
2. *globals* qui sont les fonctions exécutées en parallèle par le *GPU*. On appelle ce type de fonction un *kernel*.
3. *devices* ce sont des fonctions qui ne peuvent être appelées qu'à partir d'une fonction s'exécutant sur le *GPU*, c'est-à-dire dans une fonction *global* ou dans une autre fonction *device*.

Les fonctions *globals* définissent ce qu'on appelle les *kernels*. Ce sont elles qui s'exécutent en parallèle sur le *GPU*, avec un nombre de *threads* spécifié par le développeur. Avec CUDA, les *threads* sont regroupés dans ce qu'on appelle des blocs, dont la taille maximale est limitée, en fonction de l'architecture du *GPU*. Pour spécifier le nombre de *threads* à exécuter, il faut donc d'abord donner la taille des blocs puis déduire le nombre de blocs nécessaires pour traiter toutes les données pour définir ce qu'on appelle la grille de blocs. Par exemple, si on veut exécuter 1000 *threads* on peut utiliser des blocs de 64 *threads* et une grille de 16 blocs, mais il existe beaucoup d'autres possibilités. Pour définir ces nombres, la pratique la plus courante consiste à fixer le nombre de *threads* par bloc et ensuite de fixer la taille de la grille. Avec les dernières générations de *GPU*, le taille des blocs peut être fixée de manière arbitraire entre 1 et 1024. Cependant, il est préférable de suivre certaines règles pour fixer ce nombre afin d'exploiter efficacement la puissance du *GPU* en maximisant son occupation, c'est-à-dire, maximiser le nombre de *threads* s'exécutant simultanément.

C.1 Définition de la taille des blocs

La première contrainte à prendre en considération pour définir la taille des blocs est de choisir un nombre multiple de 32. En effet, lorsqu'un bloc est traité par un *SM*, celui-ci est découpé en sous-groupes, les *warps*. Les *warps* sont des groupes d'exactement 32 *threads*. Si un bloc avec une taille qui n'est pas multiple de 32 est transmis au *SM* celui-ci va lui ajouter des *threads* qui ne vont rien exécuter, mais vont se voir allouer les mêmes ressources que les autres *threads*. Par exemple, si le développeur crée des blocs de 100 *threads*, le *SM* va allouer des registres et des unités de calcul pour 128 *threads* dont 28 seront totalement inactifs, impliquant une occupation maximale de 88%.

Les *SM* peuvent héberger un nombre limité de *threads*, plus élevé que le nombre de *threads* qu'il peut exécuter à un instant donné, ce qui permet de masquer, dans une certaine mesure, les latences nécessaires aux accès mémoire. En effet, le *SM* met en attente les *threads* faisant un appel mémoire pendant le temps de la latence mémoire associé et en profite pour exécuter d'autres *threads*. Pour maximiser l'occupation, il faut donc héberger autant de *threads* que possible afin que le *SM* ait toujours des *threads* à exécuter pendant les temps de latence mémoire. Les *SM* ne peuvent héberger que des nombres limités de blocs, variables en fonction de la génération de l'architecture considérée.

Nous avons vu dans la section précédente qu'un *GPU* se découpe en *SM* qui possède chacun une quantité limite de registres (voir le tableau 2.1), ce qui conduit à une autre limite du nombre de *threads* que peut héberger un *SM*. Le nombre de registres consommés par un *kernel* est déterminé par le compilateur. Il est donc nécessaire de compiler le *kernel* pour vérifier sa consommation et modifier la taille des blocs en conséquence.

En résumé, il y trois principes à suivre pour définir le nombre de *threads* par bloc. Le premier est de choisir un nombre multiple de 32. Le second principe est d'essayer d'héberger autant de *threads* que possible dans le *SM* en considérant les nombres limites de blocs et de *threads* qu'il peut héberger. Le troisième principe est de vérifier le nombre de registres consommés par le *kernel*. Si les tailles de bloc données par les deux premiers principes ne permettent pas d'héberger le nombre maximal de *threads* que peut héberger le *SM*, la taille du bloc est optimisée pour pouvoir héberger autant de *threads* que possible par *SM* en tenant compte du premier principes et de la limite du nombre de registres.

Prenons un exemple. Nous avons un *kernel* consommant 40 registres de 32 bits que l'on souhaite exécuter sur un *GPU* de génération Maxwell. Avec cette architecture, les nombres maximaux de blocs et de *threads* pouvant être hébergés par un *SM* sont respectivement de 32 blocs et 2048 *threads*. En considérant les deux premiers principes, on trouve qu'il possible d'héberger 2048 *threads* avec des blocs dont la taille est de 64, 128, 256, 512 ou 1024. Avec la génération de *GPU* considérée, chaque *SM* possède 65536 registres de 32 bits. Le troisième principe nous dit donc que le nombre maximal de *threads* de ce *kernel* que peut héberger un *SM* est 1638, ce qui est inférieur au nombre maximal de 2048. Il faut donc trouver une taille de bloc permettant d'herber un nombre de *threads* multiple de 32 et aussi proche que possible de la limite des 1638 *threads* sans la dépasser. Cet optimum est atteint avec des blocs de 96 ou 544 *threads*, qui permettent d'héberger 1632 *threads*.

C.2 Exemple d'implémentation : la convolution 1D

Nous allons voir dans cette section comment implémenter une fonction simple adaptée à une exécution massivement parallèle, la convolution de deux fonctions. La convolution discrète d'une fonction f par un noyau k prend la forme suivante :

$$g(i) = (f * k)(i) = \sum_{j \in \text{supp} k} f(i-j)k(j) \tag{C.1}$$

où supp$k$ est le support du noyau de convolution. L'implémentation en langage C de cette opération peut prendre la forme suivante :

```
void convolution_C(float* f, int taille_f, float* k, int taille_k, float* g)
{
    // Boucle sur les éléments du vecteur.
    for( int i=0; i<taille_f; ++i)
    {
        // Initialisation de la valeur de sortie.
        g[i] = 0;

        // Boucle sur les éléments du noyau.
        for(int j = 0; j<taille_k; ++j)
            if(i-j>=0 and i-j<taille_f)    g[i] += f[i-j]*k_shared[j];
    }
}
```

Ici, chaque élément de la sortie est traité séquentiellement avec la première boucle *for*. Comme chacun de ces traitements peut être fait indépendamment des autres, on peut imaginer une parallélisation où chaque *thread* traiterait un élément de la sortie. On peut alors écrire un *kernel* qui encapsule simplement ce qu'on trouve dans la boucle parcourant les éléments i. Ce *kernel* pourrait être écrit de la manière suivante :

```
__global__ void kernel(float* f, float* g, int taille_f, float* k,int taille_k)
{
    // Calcul de l'indice du thread.
    int i = blockIdx.x * blockDim.x + threadIdx.x;

    // Interruption des threads dont les indices dépassent la taille du vecteur
        à convoluer.
    if(i>=taille_f) return;

    // Initialisation de la valeur de sortie.
    g[i] = 0;

    // Boucle sur tous les éléments du noyau.
    for(int j = 0; j<taille_k; ++j)
        if(i-j>=0 and i-j<taille_f)    g[i] += f[i-j]*k[j];
}
```

Au début du *kernel*, une étape commune est de calculer l'indice du *thread* à partir de l'indice du bloc, de la taille des blocs et de l'indice du *thread* dans le bloc. Ensuite, cet indice correspond à l'élément de la sortie qui va être traité. Généralement, pour ce type de convolution, le support du noyau est assez limité. On peut donc envisager une implémentation *GPU* exploitant la mémoire partagée pour stocker le noyau de convolution. Ce *kernel* pourrait être implémenté de la manière suivante :

```
__global__ void kernel(float* f, float* g, int taille_f, float* k,int taille_k)
{
    // Calcul de l'indice du thread.
    int i = blockIdx.x * blockDim.x + threadIdx.x;

    // Chargement du noyau de convolution en mémoire partagée.
    extern __shared__ float k_shared[];
    if(threadIdx.x<taille_k)    k_shared[threadIdx.x] = k[threadIdx.x];

    // Synchronisation des threads pour garantir que le noyau a été entièrement
        chargé en mémoire partagée.
    __syncthreads();
```

```
    // Interruption des threads dont les indices dépassent la taille du vecteur
        à convoluer.
    if(i>=taille_f) return;

    // Initialisation de la valeur de sortie.
    g[i] = 0;

    // Boucle sur tous les éléments du noyau.
    for(int j = 0; j<taille_k; ++j)
        if(i-j>=0 and i-j<taille_f)    g[i] += f[i-j]*k_shared[j];
}
```

Au début du *kernel* les éléments du noyau sont recopiés dans la mémoire partagée. Ensuite, tous *threads* du bloc sont synchronisés pour être sûr que, dans la suite des traitements, tous les éléments du noyau sont bien chargés. Chaque *thread* charge en mémoire partagée un élément du noyau, dont l'indice est égal à celui du *thread* dans le bloc. Cette implémentation limite la taille du *kernel*, elle ne doit pas dépasser la taille des blocs.

Ces deux *kernels*, compilés pour l'architecture de *GPU* Maxwell, consomme 13 registres de 32 bits pour la version n'utilisant pas la mémoire partagée et 12 registres pour la version l'utilisant. Cette architecture permet d'héberger au maximum 2048 *threads* par *SM* et possède 64k registres de 32 bits par *SM*. Ces deux *kernels* consomment peut de registres, ce qui permet d'héberger le nombre maximal de *threads* (la limite pour satisfaire cette condition étant de 32 registres). Sachant que ce type de *SM* peut héberger au maximum 32 blocs, on peut donc utiliser une des tailles de bloc suivantes : 64, 128, 256, 512 et 1024. Nous avons fixé cette valeur à 1024, le maximum, pour permettre de stocker des noyaux de convolution les plus grands possible avec le *kernel* utilisant la mémoire partagée.

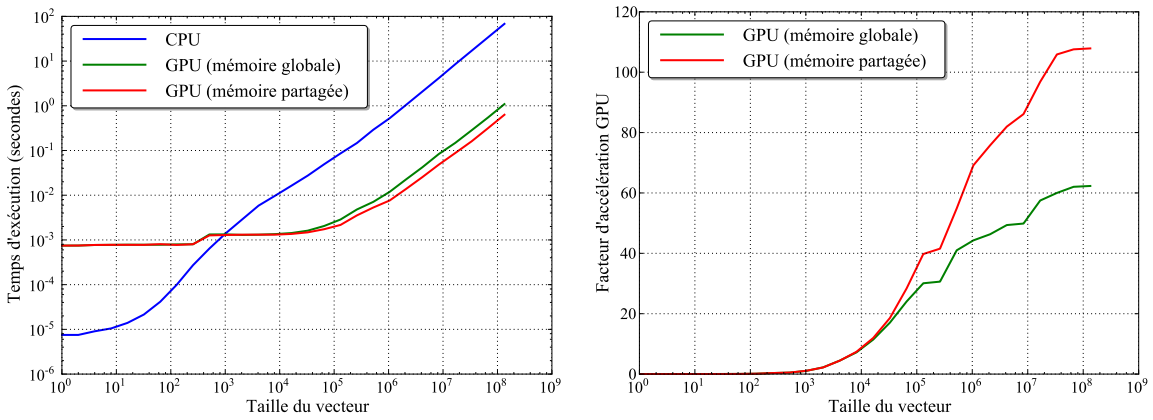
Avant d'appeler les *kernels*, il est nécessaire d'allouer de l'espace dans la mémoire globale du *GPU* pour les données à traiter et les résultats puis de recopier les données à traiter, de la mémoire du *CPU* vers la mémoire globale du *GPU*. Nous avons implémenté la fonction *host* appelant les *kernels* de la manière suivante :

```
void convolution_CUDA(float* f, int taille_f, float* k, int taille_k, float* g)
{
    int taille_bloc = 1024;
    // Si le noyau est plus grand que le bloc : interruption.
    if(taille_k > taille_bloc) return;

    // Allocation de l'espace necessaire en mémoire globale.
    float *_f, *_g, *_k;
    cudaMalloc((void**)&_f, sizeof(float)*taille_f);
    cudaMalloc((void**)&_g, sizeof(float)*taille_f);
    cudaMalloc((void**)&_k, sizeof(float)*taille_k);

    // Copie du vecteur et du noyau en mémoire globale.
    cudaMemcpy(_f, f, sizeof(float)*taille_f, cudaMemcpyHostToDevice);
    cudaMemcpy(_k, k, sizeof(float)*taille_k, cudaMemcpyHostToDevice);

    // Calcul du nombre de blocs necessaire pour couvrir l'ensemble du vecteur.
    int nbre_blocs = (taille_f+taille_bloc-1)/taille_bloc;

    // Execution de la convolution.
    kernel<<<nbre_blocs, taille_bloc, taille_k*sizeof(float)>>>
        (_f, _g, taille_f, _k, taille_k);
```

```
    // Copie du résultat de la mémoire globale vers la mémoire du CPU.
    cudaMemcpy(g, _g, sizeof(float)*taille_f, cudaMemcpyDeviceToHost);

    // Désallocations de la mémoire globale.
    cudaFree(_f);    cudaFree(_g);    cudaFree(_k);
}
```

Lors de l'appel d'un *kernel*, en plus des arguments, il est nécessaire de fournir plusieurs autres options entre des triples chevrons. Dans l'ordre, ces valeurs sont le nombre de blocs de *threads* qu'il faut créer, le nombre de *threads* par bloc, le nombre d'octets de mémoire partager à allouer dynamiquement par bloc (0 par défaut) et le flux associé (optionnel).

Les temps d'exécution de la convolution sur *CPU* et *GPU* avec et sans utilisation de la mémoire partagée ont été évalué sur une plateforme composée d'un *CPU* Intel Xeon E5-2680 à 2.70GHz et d'une carte graphique NVIDIA GTX 980 Ti à 1GHz. La taille du noyau de convolution a été fixée à 200, la taille du vecteur à convoluer varie entre 2^1 et 2^{28}. Les temps d'exécution et les facteurs d'accélération sont présentés dans la figure C.1.



(a) (b)

FIGURE C.1 – Temps d'exécution d'une convolution d'un vecteur de taille variable, implémenté sur *CPU* et sur *GPU* avec et sans utilisation de la mémoire partagée dans sous-figure (a). Facteurs d'accélération des deux implémentations *GPU* dans la sous-figure (a).

On peut constater que pour toutes les implémentations, les temps d'exécution varient linéairement pour les convolutions avec des vecteurs de grande taille et sont constants pour les vecteurs de petite taille. Le temps d'exécution de l'implémentation *CPU* devient linéaire très rapidement, à partir des tailles supérieures à 100. Les temps d'exécution des implémentations *GPU* évoluent linéairement beaucoup plus tard, pour des vecteurs contenant plus d'un million d'éléments. Cette différence vient du fait qu'avec le *GPU* chaque instruction envoyée par l'hôte (allocation, exécution de *kernel*, copie ...) est exécutée après un certain temps de latence fixe. Avec l'implémentation *CPU* ce type de latences existe aussi, mais elles sont de l'ordre de la microseconde. Pour les vecteurs de petites tailles, la latence est largement supérieure au temps d'exécution réel, ce qui explique qu'on observe un temps d'exécution constant. Les deux implémentations *GPU* donnent des temps d'exécution proches avec une différence qui s'amplifie pour les grandes tailles de vecteur. Les implémentations *GPU* surpassent l'implémentation *CPU* pour des vecteurs ayant plus de 1000 éléments et

donnent des facteurs d'accélération qui tendent vers environ 60 et 110, respectivement pour l'implémentation n'utilisant pas la mémoire partagée et pour l'implémentation utilisant cette mémoire.

Bibliographie

[Adam *et al.*, 1998] ADAM, L.-E., KARP, J. et FREIFELDER, R. (1998). Scatter correction using a dual energy window technique for 3d pet with nai (tl) detectors. *Dans Nuclear Science Symposium*, volume 3, pages 2011–2018. IEEE. 41

[Afshar-Oromieh *et al.*, 2012] AFSHAR-OROMIEH, A., HABERKORN, U., EDER, M., EISENHUT, M. et ZECHMANN, C. (2012). [68ga] gallium-labelled psma ligand as superior pet tracer for the diagnosis of prostate cancer : comparison with 18f-fech. *European journal of nuclear medicine and molecular imaging*, 39(6):1085–1086. 130

[Agostinelli *et al.*, 2003] AGOSTINELLI, S., ALLISON, J., AMAKO, K. a., APOSTOLAKIS, J., ARAUJO, H., ARCE, P., ASAI, M., AXEN, D., BANERJEE, S., BARRAND, G. *et al.* (2003). Geant4—a simulation toolkit. *Nuclear instruments and methods in physics research section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303. 131

[Alessio *et al.*, 2006] ALESSIO, A., KINAHAN, P. et LEWELLEN, T. (2006). Modeling and incorporation of system response functions in 3-d whole body pet. *IEEE Transactions on Medical Imaging*, 25:828 – 837. 35, 78

[Alessio et MacDonald, 2008] ALESSIO, A. et MACDONALD, L. (2008). Spatially variant positron range modeling derived from ct for pet image reconstruction. *Dans Nuclear Science Symposium Conference Record*, pages 3637–3640. IEEE. 38, 135

[Alessio *et al.*, 2010] ALESSIO, A., STEARNS, C., TONG, S., ROSS, S., KOHLMYER, S., GANIN, A. et KINAHAN, P. (2010). Application and evaluation of a measured spatially variant system model for pet image reconstruction. *IEEE Transactions on Medical Imaging*, 29:938 – 949. 78

[Allison *et al.*, 2006] ALLISON, J., AMAKO, K., APOSTOLAKIS, J., ARAUJO, H., DUBOIS, P. A., ASAI, M., BARRAND, G., CAPRA, R., CHAUVIE, S., CHYTRACEK, R. *et al.* (2006). Geant4 developments and applications. *Nuclear Science, IEEE Transactions on*, 53(1):270–278. 131

[Anizan, 2010] ANIZAN, N. (2010). *Imagerie quantitative à l'iode-124 en tomographie par émission de positons du petit animal.* Thèse de doctorat, Nantes. 117, 140

[Badawi *et al.*, 2000] BADAWI, R., FERREIRA, N., KOHLMYER, S., DAHLBOM, M., MARSDEN, P. et LEWELLEN, T. (2000). A comparison of normalization effects on three whole-body cylindrical 3d pet systems. *Physics in medicine and biology*, 45(11):3253. 36

[Badawi *et al.*, 1999] BADAWI, R., MILLER, M., BAILEY, D. et MARSDEN, P. (1999). Randoms variance reduction in 3d pet. *Physics in medicine and biology*, 44(4):941. 40

[Bai et Esser, 2010] BAI, B. et ESSER, P. D. (2010). The effect of edge artifacts on quantification of positron emission tomography. *Dans Nuclear Science Symposium Conference Record*, pages 2263–2266. IEEE. 122

[Bai *et al.*, 2005] BAI, B., LAFOREST, R., SMITH, A. M. et LEAHY, R. M. (2005). Evaluation of map image reconstruction with positron range modeling for 3d pet. *Dans Nuclear Science Symposium Conference Record*, volume 5, pages 2686–2689. IEEE. 38, 134

[Bai *et al.*, 2003] BAI, B., RUANGMA, A., LAFOREST, R., TAI, Y.-C. et LEAHY, R. M. (2003). Positron range modeling for statistical pet image reconstruction. *Dans Nuclear Science Symposium Conference Record*, volume 4, pages 2501–2505. IEEE. 38, 135

[Bailey *et al.*, 1991] BAILEY, D., JONES, T., SPINKS, T., GILARDI, M.-C. et TOWNSEND, D. (1991). Noise equivalent count measurements in a neuro-pet scanner with retractable septa. *Medical Imaging, IEEE Transactions on*, 10(3):256–260. 26

[Bailey, 1992] BAILEY, D. L. (1992). 3d acquisition and reconstruction in positron emission tomography. *Annals of nuclear medicine*, 6(3):123–130. 26

[Balcerzyk *et al.*, 2000] BALCERZYK, M., MOSZYŃSKI, M., KAPUSTA, M., WOLSKI, D., PAWELKE, J. et MELCHER, C. (2000). Yso, lso, gso and lgso. a study of energy resolution and nonproportionality. *Nuclear Science, IEEE Transactions on*, 47(4):1319–1323. 17

[Barrett *et al.*, 1994] BARRETT, H. H., WILSON, D. W. et TSUI, B. M. (1994). Noise properties of the em algorithm. i. theory. *Physics in medicine and biology*, 39(5):833. 32

[Bateman *et al.*, 2006] BATEMAN, T. M., HELLER, G. V., MCGHIE, A. I., FRIEDMAN, J. D., CASE, J. A., BRYNGELSON, J. R., HERTENSTEIN, G. K., MOUTRAY, K. L., REID, K. et CULLOM, S. J. (2006). Diagnostic accuracy of rest/stress ecg-gated rb-82 myocardial perfusion pet : comparison with ecg-gated tc-99m sestamibi spect. *Journal of nuclear cardiology*, 13(1):24–33. 130

[Ben Bouallègue *et al.*, 2007] BEN BOUALLÈGUE, F., CROUZET, J.-F., COMTAT, C., FOURCADE, M., MOHAMMADI, B. et MARIANO-GOULART, D. (2007). Exact and approximate fourier rebinning algorithms for the solution of the data truncation problem in 3-d pet. *Medical Imaging, IEEE Transactions on*, 26(7):1001–1009. 26

[Bendriem *et al.*, 1998] BENDRIEM, B., TOWNSEND, D. W. et TOWNSEND, D. W. (1998). *The theory and practice of 3D PET*, volume 32. Springer Science & Business Media. 41

[Bert *et al.*, 2013] BERT, J., PEREZ-PONCE, H., EL BITAR, Z., BOURSIER, Y., VINTACHE, D., BONISSENT, A., MOREL, C., BRASSE, D., VISVIKIS, D. *et al.* (2013). Geant4-based monte carlo simulations on gpu for medical applications. *Physics in medicine and biology*, 58(16):5593. 79

[Bert et Visvikis, 2011] BERT, J. et VISVIKIS, D. (2011). A fast cpu/gpu ray projector for fully 3d list-mode pet reconstruction. *Dans IEEE Nuclear Science Symposium Conference Record*, pages 4126 – 4130. 35, 48, 80, 90, 95

[Boussion *et al.*, 2009] BOUSSION, N., LE REST, C. C., HATT, M. et VISVIKIS, D. (2009). Incorporation of wavelet-based denoising in iterative deconvolution for partial volume correction in whole-body pet imaging. *European journal of nuclear medicine and molecular imaging*, 36(7):1064–1075. 32

[Braem *et al.*, 2004] BRAEM, A., LLATAS, M., CHESI, E., CORREIA, J., GARIBALDI, F., JORAM, C., MATHOT, S., NAPPI, E., Ribeiro da SILVA, M., SCHOENAHL, F., GUINOT, J., WEILHAMMER, P. et ZAIDI, H. (2004). Feasibility of a novel design of high resolution parallax-free compton enhanced pet scanner dedicated to brain research. *Physics in medicine and biology*, 49:2547–2562. 35

[Bresenham, 1965] BRESENHAM, J. (1965). Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4. 79

[Browne et De Pierro, 1996] BROWNE, J. et DE PIERRO, A. R. (1996). A row-action alternative to the em algorithm for maximizing likelihood in emission tomography. *Medical Imaging, IEEE Transactions on*, 15(5):687–699. 31

[Brownell *et al.*, 1971] BROWNELL, G., BURNHAM, C., HOOP JR, B. et BOHNING, D. (1971). Quantitative dynamic studies using short-lived radioisotopes and positron detection. *Dans Dynamic Studies with Radioisotopes in Medicine. Proceedings of the Symposium on Dynamics Studies with Radioisotopes in Clinical Medicine and Research*. 22

[Cabello et Rafecas, 2012] CABELLO, J. et RAFECAS, M. (2012). Comparison of basis functions for 3d pet reconstruction using a monte carlo system matrix. *Physics in medicine and biology*, 57(7):1759. 113

[Cal-González *et al.*, 2009] CAL-GONZÁLEZ, J., HERRAIZ, J., ESPAÑA, S., DESCO, M., VAQUERO, J. J. et UDÍAS, J. M. (2009). Positron range effects in high resolution 3d pet imaging. *Dans Nuclear Science Symposium Conference Record*, pages 2788–2791. IEEE. 38, 134

[Cal-González *et al.*, 2011] CAL-GONZÁLEZ, J., HERRAIZ, J., ESPAÑA, S., VICENTE, E., HERRANZ, E., DESCO, M., VAQUERO, J. J. et UDÍAS, J. M. (2011). Study of ct-based positron range correction in high resolution 3d pet imaging. *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, 648:S172–S175. 134

[Casey et Hoffman, 1986] CASEY, M. E. et HOFFMAN, E. J. (1986). Quantitation in positron emission computed tomography : 7. a technique to reduce noise in accidental coincidence measurements and coincidence efficiency calibration. *Journal of computer assisted tomography*, 10(5):845–850. 33, 40

[Champion et Le Loirec, 2006] CHAMPION, C. et LE LOIREC, C. (2006). Positron follow-up in liquid water : I. a new monte carlo track-structure code. *Physics in medicine and biology*, 51(7):1707. 131

[Champion et Le Loirec, 2007] CHAMPION, C. et LE LOIREC, C. (2007). Positron follow-up in liquid water : Ii. spatial and energetic study for the most important radioisotopes used in pet. *Physics in medicine and biology*, 52(22):6605. 131

[Chan *et al.*, 2010] CHAN, C., FULTON, R., FENG, D. D. et MEIKLE, S. (2010). Median non-local means filtering for low snr image denoising : Application to pet with anatomical knowledge. *Dans Nuclear Science Symposium Conference Record*, pages 3613–3618. IEEE. 32

[Chen *et al.*, 1991] CHEN, C., LEE, S.-Y. et CHO, Z. (1991). Parallelization of the em algorithm for 3-d pet image reconstruction. *Medical Imaging, IEEE Transactions on*, 10(4):513–522. 58, 80

[Chen et Glick, 2007] CHEN, Y. et GLICK, S. (2007). Determination of the system matrix used in list-mode em reconstruction of pet. *Dans IEEE Nuclear Science Symposium Conference Record*, pages 3855 – 3858. 35, 87

[Cherry et Huang, 1995] CHERRY, S. R. et HUANG, S.-C. (1995). Effects of scatter on model parameter estimates in 3d pet studies of the human brain. *Nuclear Science, IEEE Transactions on*, 42(4):1174–1179. 33, 41

[Cho *et al.*, 1975] CHO, Z., CHAN, J., ERICKSSON, L., SINGH, M., GRAHAM, S., MACDONALD, N. et YANO, Y. (1975). Positron ranges obtained from biomedically important positron-emitting radionuclides. *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, 16(12):1174–1176. 131

[Colombino *et al.*, 1965] COLOMBINO, P., FISCELLA, B. et TROSSI, L. (1965). Study of positronium in water and ice from 22 to-144 c by annihilation quanta measurements. *Il Nuovo Cimento Series 10*, 38(2):707–723. 10

[Colsher, 1980] COLSHER, J. G. (1980). Fully-three-dimensional positron emission tomography. *Physics in medicine and biology*, 25(1):103. 26

[Cui *et al.*, 2013] CUI, J., PRATX, G., MENG, B. et LEVIN, C. S. (2013). Distributed mlem : An iterative tomographic image reconstruction algorithm for distributed memory architectures. *Medical Imaging, IEEE Transactions on*, 32(5):957–967. 48, 58

[Cui *et al.*, 2011a] CUI, J., PRATX, G., PREVRHAL, S. et LEVIN, C. (2011a). Fully 3d list-mode time-of-flight pet image reconstruction on gpus using cuda. *Medical Physics*, 38:6775–6786. 35, 44, 48, 57, 81

[Cui *et al.*, 2010] CUI, J., PRATX, G., PREVRHAL, S., SHAO, L. et LEVIN, C. S. (2010). Fully 3-d list-mode positron emission tomography image reconstruction on gpu using cuda. *Dans Nuclear Science Symposium Conference Record*, pages 2635–2637. IEEE. 35, 48, 56, 81

[Cui *et al.*, 2011b] CUI, J., PRATX, G., PREVRHAL, S., ZHANG, B., SHAO, L. et LEVIN, C. (2011b). Measurement-based spatially-varying point spread function for list-mode pet reconstruction on gpu. *Dans IEEE Nuclear Science Symposium Conference Record*. 35, 48, 82

[Cui *et al.*, 2011c] CUI, J., PREVRHAL, S., PRATX, G., SHAO, L. et LEVIN, C. (2011c). Fully 3-d list-mode positron emission tomography image reconstruction on a multi-gpu cluster. *Fully 3D*, pages 35–39. 48, 56, 58, 81

[Daniel et Fessler, 2000] DANIEL, F. Y. et FESSLER, J. A. (2000). Mean and variance of single photon counting with deadtime. *Physics in medicine and biology*, 45(7):2043. 32

[Daube-Witherspoon et Muehllehner, 1987] DAUBE-WITHERSPOON, M. et MUEHLLEHNER, G. (1987). Treatment of axial data in three-dimensional pet. *The Journal of Nuclear Medicine*, 28:1717–1724. 26

[De Bernardi *et al.*, 2003] DE BERNARDI, E., ZITO, F., MICHELUTTI, L., MAINARDI, L., GERUNDINI, P. et BASELLI, G. (2003). Improving pet image spatial resolution by experimental measurement of scanner blurring properties. *Dans IEEE Engineering in Medicine and Biology Society*. 35, 78

[Defrise, 1995] DEFRISE, M. (1995). A factorization method for the 3d x-ray transform. *Inverse Problems*, 11:983 – 994. 26

[Defrise *et al.*, 1997] DEFRISE, M., KINAHAN, P., TOWNSEND, D., MICHEL, C., SIBOMANA, M. et NEWPORT, D. (1997). Exact and approximate rebinning algorithms for 3-d pet data. *IEEE Transactions on Medical Imaging*, 16:145 – 158. 26

[Defrise *et al.*, 2012] DEFRISE, M., REZAEI, A. et NUYTS, J. (2012). Time-of-flight pet data determine the attenuation sinogram up to a constant. *Physics in medicine and biology*, 57(4):885. 38

[Defrise *et al.*, 1991] DEFRISE, M., TOWNSEND, D., BAILEY, D., GEISSBUHLER, A. et JONES, T. (1991). A normalization technique for 3d pet data. *Physics in medicine and biology*, 36(7):939. 36

[Defrise *et al.*, 1989] DEFRISE, M., TOWNSEND, D. et CLACK, R. (1989). Three-dimensional image reconstruction from complete projections. *Physics in medicine and biology*, 34(5):573. 26

[Dempster *et al.*, 1977] DEMPSTER, A. P., LAIRD, N. M. et RUBIN, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38. 30

[Derenzo, 1986] DERENZO, S. E. (1986). Mathematical removal of positron range blurring in high resolution tomography. *Nuclear Science, IEEE Transactions on*, 33(1):565–569. 38, 131, 133

[Eriksson *et al.*, 1994] ERIKSSON, L., WIENHARD, K. et DAHLBOM, M. (1994). A simple data loss model for positron camera systems. *Nuclear Science, IEEE Transactions on*, 41(4):1566–1570. 40, 41

[España *et al.*, 2009] ESPAÑA, S., HERRAIZ, J., VICENTE, E., VAQUERO, J. J., DESCO, M. et UDÍAS, J. M. (2009). Penelopet, a monte carlo pet simulation tool based on penelope : features and validation. *Physics in medicine and biology*, 54(6):1723. 131

[Frese *et al.*, 2003] FRESE, T., ROUZE, N., BOUMAN, C., SAUER, K. et HUTCHINS, G. (2003). Quantitative comparison of fbp, em, and bayesian reconstruction algorithms for the indypet scanner. *IEEE Transactions on Medical Imaging*, 22:258 – 276. 35, 78

[Gaens *et al.*, 2013] GAENS, M., BERT, J., PIETRZYK, U., JON SHAH, N. et VISVIKIS, D. (2013). Gpu-accelerated monte carlo based scatter correction in brain pet/mr. *Dans Nuclear Science Symposium and Medical Imaging Conference*, pages 1–3. IEEE. 43

[Gifford *et al.*, 2000] GIFFORD, H. C., KING, M. A., WELLS, R. G., HAWKINS, W. G., NARAYANAN, M. V. et PRETORIUS, P. H. (2000). Lroc analysis of detector-response compensation in spect. *Medical Imaging, IEEE Transactions on*, 19(5):463–473. 77

[Goertzen *et al.*, 2012] GOERTZEN, A., BAO, Q., BERGERON, M., BLANKEMEYER, E., BLINDER, S., CANADAS, M., CHATZIIOANNOU, A., DINELLE, K., ELHAMI, E., JANS, H., LAGE, E., LECOMTE, R., SOSSI, V., SURTI, S., TAI, Y., VAQUERO, J., VICENTE, E., WILLIAMS, D. et LAFOREST, R. (2012). Nema nu 4-2008 comparison of preclinical pet imaging systems. *Journal of Nuclear Medicine*. 118, 126

[Gonzalez *et al.*, 2011] GONZALEZ, E., CUI, J., PRATX, G., BIENIOSEK, M., OLCOTT, P. et LEVIN, C. (2011). Point spread function for pet detectors based on the probability density function of the line segment. *Dans IEEE Nuclear Science Symposium Conference Record*, pages 4386 – 4389. 85, 86

[Gordon *et al.*, 1970] GORDON, R., BENDER, R. et HERMAN, G. (1970). Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of Theoretical Biology*, 29:471–481. 29

[Green *et al.*, 1993] GREEN, M. A., MATHIAS, C. J., NEUMANN, W. L., FANWICK, P. E., JANIK, M. et DEUTSCH, E. A. (1993). Potential gallium-68 tracers for imaging the heart with pet : evaluation of four gallium complexes with functionalized tripodal tris (salicylaldimine) ligands. *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, 34(2):228–233. 130

[Groiselle et Glick, 2004] GROISELLE, C. J. et GLICK, S. J. (2004). 3d pet list-mode iterative reconstruction using time-of-flight information. *Dans Nuclear Science Symposium Conference Record, 2004 IEEE*, volume 4, pages 2633–2638. IEEE. 44

[Guérin et El Fakhri, 2011] GUÉRIN, B. et EL FAKHRI, G. (2011). Novel scatter compensation of list-mode pet data using spatial and energy dependent corrections. *Medical Imaging, IEEE Transactions on*, 30(3):759–773. 43

[Ha *et al.*, 2012] HA, S., ISPIRYAN, M., MATEJ, S. et MUELLER, K. (2012). Gpu-based spatially variant sr kernel modeling and projections in 3d direct tof pet reconstruction. *Dans IEEE Medical imaging conference*. 48, 82

[Ha *et al.*, 2013] HA, S., MATEJ, S., ISPIRYAN, M. et MUELLER, K. (2013). Gpu-accelerated forward and back-projections with spatially varying kernels for 3d direct tof pet reconstruction. *IEEE transactions on nuclear science*, 60(1):166. 82

[Haber *et al.*, 1990] HABER, S., DERENZO, S. E. et UBER, D. (1990). Application of mathematical removal of positron range blurring in positron emission tomography. *Nuclear Science, IEEE Transactions on*, 37(3):1293–1299. 38, 133

[Hammer *et al.*, 1994] HAMMER, B. E., CHRISTENSEN, N. L. et HEIL, B. G. (1994). Use of a magnetic field to increase the spatial resolution of positron emission tomography. *Medical physics*, 21(12): 1917–1920. 38, 133

[Harrison *et al.*, 1991] HARRISON, R. L., HAYNOR, D. R. et LEWELLEN, T. K. (1991). Dual energy window scatter corrections for positron emission tomography. *Dans Nuclear Science Symposium and Medical Imaging Conference, 1991., Conference Record of the 1991 IEEE*, pages 1700–1704. IEEE. 41

[Hebert et Leahy, 1989] HEBERT, T. et LEAHY, R. (1989). A generalized em algorithm for 3-d bayesian reconstruction from poisson data using gibbs priors. *Medical Imaging, IEEE Transactions on*, 8(2): 194–202. 33

[Herman et Meyer, 1993] HERMAN, G. T. et MEYER, L. B. (1993). Algebraic reconstruction techniques can be made computationally efficient [positron emission tomography application]. *Medical Imaging, IEEE Transactions on*, 12(3):600–609. 29

[Hoffman *et al.*, 1979] HOFFMAN, E. J., HUANG, S.-C. et PHELPS, M. E. (1979). Quantitation in positron emission computed tomography : 1. effect of object size. *Journal of computer assisted tomography*, 3(3):299–308. 33

[Hoffman *et al.*, 1981] HOFFMAN, E. J., HUANG, S.-C., PHELPS, M. E. et KUHL, D. E. (1981). Quantitation in positron emission computed tomography : 4. effect of accidental coincidences. *Journal of computer assisted tomography*, 5(3):391–400. 33, 39, 40

[Hoffman *et al.*, 1982] HOFFMAN, E. J., HUANG, S.-C., PLUMMER, D. et PHELPS, M. E. (1982). Quantitation in positron emission computed tomography : 6. effect of nonuniform resolution. *Journal of computer assisted tomography*, 6(5):987–999. 33

[Hoffmann *et al.*, 1976] HOFFMANN, E., PHELPS, M., MULLANI, N., HIGGINS, C. et TER-POGOSSIAN, M. (1976). Design and performance characteristics of a whole-body positron transaxial tomograph. *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, 17(6):493–502. 131

[Hofmann *et al.*, 2011] HOFMANN, M., BEZRUKOV, I., MANTLIK, F., ASCHOFF, P., STEINKE, F., BEYER, T., PICHLER, B. J. et SCHÖLKOPF, B. (2011). Mri-based attenuation correction for whole-body pet/mri : quantitative evaluation of segmentation-and atlas-based methods. *Journal of Nuclear Medicine*, 52(9):1392–1399. 37

[Huang *et al.*, 1979] HUANG, S.-C., HOFFMAN, E. J., PHELPS, M. E. et KUHL, D. E. (1979). Quantitation in positron emission computed tomography : 2. effects of inaccurate attenuation correction. *Journal of computer assisted tomography*, 3(6):804–hyhen. 33

[Hudson et Larkin, 1994] HUDSON, H. et LARKIN, R. (1994). Accelerated image reconstruction using ordered subsets of projection data. *IEEE Transactions on Medical Imaging*, 13:601 – 609. 31

[Jacobs *et al.*, 1998] JACOBS, F., SUNDERMANN, E., DE SUTTER, B., CHRISTIAENS, M. et LEMAHIEU, I. (1998). A fast algorithm to calculate the exact radiological path through a pixel or voxel space. *Journal of computing and information technology*, 6(1):89–94. 80

[Jan *et al.*, 2011] JAN, S., BENOIT, D., BECHEVA, E., CARLIER, T., CASSOL, F., DESCOURT, P., FRISSON, T., GREVILLOT, L., GUIGUES, L., MAIGNE, L., MOREL, C., PERROT, Y., REHFELD, N., SARRUT, D., SCHAART, D., STUTE, S., PIETRZYK, U., VISVIKIS, D., ZAHRA, N. et BUVAT, I. (2011). Gate v6 : a major enhancement of the gate simulation platform enabling modelling of ct and radiotherapy. *Physics in medicine and biology*, 56:881 – 901. 66, 79, 90, 131

[Jødal *et al.*, 2012] JØDAL, L., LE LOIREC, C. et CHAMPION, C. (2012). Positron range in pet imaging : an alternative approach for assessing and correcting the blurring. *Physics in medicine and biology*, 57(12):3931. 38

[Jødal *et al.*, 2014] JØDAL, L., LE LOIREC, C. et CHAMPION, C. (2014). Positron range in pet imaging : non-conventional isotopes. *Physics in medicine and biology*, 59(23):7419. 2, 7

[Johnson *et al.*, 1995] JOHNSON, C., YAN, Y., CARSON, R. E., MARTINO, R. L., DAUBE-WITHERSPOON, M. E. *et al.* (1995). A system for the 3d reconstruction of retracted-septa pet data using the em algorithm. *Nuclear Science, IEEE Transactions on*, 42(4):1223–1227. 80

[Johnson *et al.*, 2011] JOHNSON, N. P., SDRINGOLA, S. et GOULD, K. L. (2011). Partial volume correction incorporating rb-82 positron range for quantitative myocardial perfusion pet based on systolic-diastolic activity ratios and phantom measurements. *Journal of nuclear cardiology*, 18(2): 247–258. 133

[Kaczmarz, 1937] KACZMARZ, S. (1937). Angenäherte auflösung von systemen linearer gleichungen. *Bulletin International de l'Academie Polonaise des Sciences et des Lettres*, 35:355–357. 28

[Kak et Slaney, 1988] KAK, A. C. et SLANEY, M. (1988). *Principles of computerized tomographic imaging*, volume 33. Siam. 25

[Karonis *et al.*, 2013] KARONIS, N. T., DUFFIN, K. L., ORDOÑEZ, C. E., ERDELYI, B., URAM, T. D., OLSON, E. C., COUTRAKON, G. et PAPKA, M. E. (2013). Distributed and hardware accelerated computing for clinical medical imaging using proton computed tomography (pct). *Journal of Parallel and Distributed Computing*, 73(12):1605–1612. 58

[Karp *et al.*, 2008] KARP, J., SURTI, S., DAUBE-WITHERSPOON, M. et MUEHLLEHNER, G. (2008). Benefit of time-of-flight in pet : experimental and clinical results. *The Journal of Nuclear Medicine*, 49:462 – 470. 43

[Karp *et al.*, 1990] KARP, J. S., MUEHLLEHNER, G., MANKOFF, D. A., ORDONEZ, C. E., OLLINGER, J. M., DAUBE-WITHERSPOON, M. E., HAIGH, A. T. et BEERBOHM, D. J. (1990). Continuous-slice penn-pet : a positron tomograph with volume imaging capability. *J Nucl Med*, 31(5):617–627. 41

[Kawrakow, 2000] KAWRAKOW, I. (2000). Accurate condensed history monte carlo simulation of electron transport. i. egsnrc, the new egs4 version. *Medical physics*, 27(3):485–498. 131

[Keereman *et al.*, 2010] KEEREMAN, V., FIERENS, Y., BROUX, T., DE DEENE, Y., LONNEUX, M. et VANDENBERGHE, S. (2010). Mri-based attenuation correction for pet/mri using ultrashort echo time sequences. *Journal of nuclear medicine*, 51(5):812–818. 37

[Kim *et al.*, 2014] KIM, K. S., SON, Y. D., CHO, Z. H., RA, J. B. et YE, J. C. (2014). Ultra-fast hybrid cpu-gpu multiple scatter simulation for 3-d pet. *IEEE Journal of Biomedical and Health Informatics*, 18(1):148–156. 43, 45

[Kinahan *et al.*, 1998] KINAHAN, P., TOWNSEND, D., BEYER, T. et SASHIN, D. (1998). Attenuation correction for a combined 3d pet/ct scanner. *Medical physics*, 25(10):2046–2053. 37

[Kinahan et Rogers, 1989] KINAHAN, P. E. et ROGERS, J. (1989). Analytic 3d image reconstruction using all detected events. *Nuclear Science, IEEE Transactions on*, 36(1):964–968. 26

[Kinouchi *et al.*, 2012] KINOUCHI, S., YAMAYA, T., YOSHIDA, E., TASHIMA, H., KUDO, H., HANEISHI, H. et SUGA, M. (2012). Gpu-based pet image reconstruction using an accurate geometrical system model. *Nuclear Science, IEEE Transactions on*, 59(5):1977–1983. 48

[Klein et Nishina, 1929] KLEIN, O. et NISHINA, Y. (1929). Über die streuung von strahlung durch freie elektronen nach der neuen relativistischen quantendynamik von dirac. *Zeitschrift für Physik*, 52(11-12):853–868. 42

[Knoll, 2010] KNOLL, G. F. (2010). *Radiation detection and measurement*. John Wiley & Sons. 40

[Labbé *et al.*, 1999] LABBÉ, C., THIELEMANS, K., ZAIDI, H. et MOREL, C. (1999). An object-oriented library incorporating efficient projection/backprojection operators for volume reconstruction in 3d pet. *Dans Proc. of 3D99 conference*. Citeseer. 57

[Lamare *et al.*, 2007] LAMARE, F., CARBAYO, M. L., CRESSON, T., KONTAXAKIS, G., SANTOS, A., LE REST, C. C., READER, A. et VISVIKIS, D. (2007). List-mode-based reconstruction for respiratory motion correction in pet using non-rigid body transformations. *Physics in medicine and biology*, 52(17): 5187. 45

[Lamare *et al.*, 2006] LAMARE, F., TURZO, A., BIZAIS, Y., CHEZE LE REST, C. et VISVIKIS, D. (2006). Validation of a monte carlo simulation of the philips allegro/gemini pet systems using gate. *Physics in medicine and biology*, 51:943–962. 66, 95

[Lange *et al.*, 1984] LANGE, K., CARSON, R. *et al.* (1984). Em reconstruction algorithms for emission and transmission tomography. *J. Comput. Assist. Tomogr*, 8(2):306–316. 31

[Lazaro *et al.*, 2005] LAZARO, D., EL BITAR, Z., BRETON, V., HILL, D. et BUVAT, I. (2005). Fully 3d monte carlo reconstruction in spect : a feasibility study. *Physics in Medicine and Biology*, 50(16):3739. 88, 116

[Lecomte *et al.*, 1984] LECOMTE, R., SCHMITT, D. et LAMOUREUX, G. (1984). Geometry study of a high resolution pet detection system using small detectors. *IEEE Transactions on Nuclear Science*, 1:556 – 561. 78, 83, 84, 86

[Lee *et al.*, 2004] LEE, K., KINAHAN, P., FESSLER, J., MIYAOKA, R., JANES, M. et LEWELLEN, T. (2004). Pragmatic fully 3d image reconstruction for the mices mouse imaging pet scanner. *Physics in medicine and biology*, 49:4564 – 4579. 35, 78

[Lehmer, 1951] LEHMER, D. H. (1951). Mathematical methods in large-scale computing units. *Dans Proc. 2nd Symp. on Large-Scale Digital Calculating Machinery*, pages 141–146. Harvard Univ. Press Cambridge, MA. 95

[Lehnert *et al.*, 2011] LEHNERT, W., GREGOIRE, M.-C., REILHAC, A. et MEIKLE, S. R. (2011). Analytical positron range modelling in heterogeneous media for pet monte carlo simulation. *Physics in medicine and biology*, 56(11):3313. 38

[Levin et Hoffman, 1999] LEVIN, C. S. et HOFFMAN, E. J. (1999). Calculation of positron range and its effect on the fundamental limit of positron emission tomography system spatial resolution. *Physics in medicine and biology*, 44(3):781. 10, 110, 131

[Lewitt *et al.*, 1994] LEWITT, R., MUEHLLEHNER, G. et KARP, J. (1994). Three-dimensional image reconstruction for pet by multi-slice rebinning and axial image filtering. *Physics in medicine and biology*, 39:321–339. 26

[Lewitt, 1983] LEWITT, R. M. (1983). Reconstruction algorithms : transform methods. *Proceedings of the IEEE*, 71(3):390–408. 24

[Ling *et al.*, 2007] LING, T., LEWELLEN, T. et MIYAOKA, R. (2007). Depth of interaction decoding of a continuous crystal detector module. *Physics in Medicine and Biology*, 52(8):2213. 17

[Liow *et al.*, 2000] LIOW, J.-S., ANDERSON, J. R. et STROTHER, S. C. (2000). Comparing reconstruction algorithms using a multi-variate analysis. *Nuclear Science, IEEE Transactions on*, 47(3):1136–1142. 26

[Livieratos *et al.*, 2005] LIVIERATOS, L., STEGGER, L., BLOOMFIELD, P., SCHAFERS, K., BAILEY, D. et CAMICI, P. (2005). Rigid-body transformation of list-mode projection data for respiratory motion correction in cardiac pet. *Physics in Medicine and Biology*, 50(14):3313. 45

[Lodge *et al.*, 2010] LODGE, M. A., RAHMIM, A. et WAHL, R. L. (2010). Simultaneous meausrement of noise and spatial resolution in pet phantom images. *Physics in medicine and biology*, 55:1069–1081. 98

[Lougovski *et al.*, 2014] LOUGOVSKI, A., HOFHEINZ, F., MAUS, J., SCHRAMM, G., WILL, E. et van den HOFF, J. (2014). A volume of intersection approach for on-the-fly system matrix calculation in 3d pet image reconstruction. *Physics in medicine and biology*, 59(3):561. 80

[Lubberink *et al.*, 2004] LUBBERINK, M., BOELLAARD, R., van der WEERDT, A. P., VISSER, F. C. et LAMMERTSMA, A. A. (2004). Quantitative comparison of analytic and iterative reconstruction methods

in 2-and 3-dimensional dynamic cardiac 18f-fdg pet. *Journal of Nuclear Medicine*, 45(12):2008–2015. 48

[MacDonald et Dahlbom, 1998] MACDONALD, L. R. et DAHLBOM, M. (1998). Parallax correction in pet using depth of interaction information. *Nuclear Science, IEEE Transactions on*, 45(4):2232–2237. 44

[Malits *et al.*, 2012] MALITS, R., BOLOTIN, E., KOLODNY, A. et MENDELSON, A. (2012). Exploring the limits of gpgpu scheduling in control flow bound applications. *ACM Transactions on Architecture and Code Optimization (TACO)*, 8(4):29. 55

[Marsaglia, 2003] MARSAGLIA, G. (2003). Xorshift rngs. *Journal of Statistical Software*. 95

[Martinez-Moller *et al.*, 2009] MARTINEZ-MOLLER, A., SOUVATZOGLOU, M., DELSO, G., BUNDSCHUH, R. A., CHEFD'HOTEL, C., ZIEGLER, S. I., NAVAB, N., SCHWAIGER, M. et NEKOLLA, S. G. (2009). Tissue classification as a potential approach for attenuation correction in whole-body pet/mri : evaluation with pet/ct data. *Journal of nuclear medicine*, 50(4):520–526. 37

[Matej *et al.*, 1998] MATEJ, S., KARP, J. S., LEWITT, R. M. et BECHER, A. J. (1998). Performance of the fourier rebinning algorithm for pet with large acceptance angles. *Physics in medicine and biology*, 43(4):787. 26

[Matthieu, 2014] MATTHIEU, M. (2014). *Reconstruction tomographique 3D complète par modélisation Monte Carlo de la matrice système en TEP pré-clinique à l'iode 124*. Thèse de doctorat, Faculté de Médecine et des Techniques Médicales-Nantes. 88, 112, 113, 114, 115, 116, 117, 134, 140, 156

[Mayorov, 1964] MAYOROV, F. V. (1964). *Electronic Digital Integration : Computers-Digital Differential Analyzers*. Ilife Books Ltd. 79

[Mazziotta *et al.*, 1981] MAZZIOTTA, J. C., PHELPS, M. E., PLUMMER, D. et KUHL, D. E. (1981). Quantitation in positron emission computed tomography : 5. physical-anatomical effects. *Journal of Computer Assisted Tomography*, 5(5):734–743. 33

[Meikle *et al.*, 1995] MEIKLE, S. R., BAILEY, D. L., HOOPER, P. K., EBERL, S., HUTTON, B. F., JONES, W. F., FULTON, R. R. et FULHAM, M. J. (1995). Simultaneous emission and transmission measurements for attenuation correction in whole-body pet. *Journal of Nuclear Medicine*, 36(9):1680–1688. 37

[Miller *et al.*, 2014] MILLER, M., GRIESMER, J., JORDAN, D., LAURENCE, T., MUZIC, R., NARAYANAN, M., NATARAJAMANI, D., SU, K.-H. et WANG, S. (2014). Initial characterization of a prototype digital photon counting pet system. *Journal of Nuclear Medicine*, 55(supplement 1):658–658. 20

[Moehrs *et al.*, 2008] MOEHRS, S., DEFRISE, M., BELCARI, N., DEL GUERRA, A., BARTOLI, A., FABBRI, S. et ZANETTI, G. (2008). Multi-ray-based system matrix generation for 3d pet reconstruction. *Physics in medicine and biology*, 53:6925 – 6945. 87, 103, 157

[Moisan *et al.*, 1997] MOISAN, C., ROGERS, J. et DOUGLAS, J. (1997). A count rate model for pet and its application to an lso hr plus scanner. *Nuclear Science, IEEE Transactions on*, 44(3):1219–1224. 40, 41

[Moreau *et al.*, 2014] MOREAU, M., BUVAT, I., AMMOUR, L., CHOUIN, N., KRAEBER-BODERE, F., CHEREL, M. et CARLIER, T. (2014). Preliminary assessment of fully 3d monte carlo reconstruction for

preclinical pet using iodine-124. *Dans IEEE Nuclear Science Symposium and Medical Imaging Conference Record*. 38, 78

[Moses, 2003] MOSES, W. (2003). Time of flight in pet revisited. *IEEE Transactions on Nuclear Science*, 50:1325 – 1330. 43

[Moses et Derenzo, 1994] MOSES, W. et DERENZO, S. (1994). Design studies for a pet detector module using a pin photodiode to measure depth of interaction. *IEEE Transactions on Nuclear Science*, 41:1441 – 1445. 17

[Moses *et al.*, 1995] MOSES, W., DERENZO, S., MELCHER, C. et MANENTE, R. (1995). A room temperature lso/pin photodiode pet detector module that measures depth of interaction. *Nuclear Science, IEEE Transactions on*, 42(4):1085–1089. 17

[Mumcuoglu *et al.*, 1996a] MUMCUOGLU, E., LEAHY, R., CHERRY, S. et HOFFMAN, E. (1996a). Accurate geometric and physical response modelling for statistical image reconstruction in high resolution pet. *Dans IEEE Nuclear Science Symposium Conference Record*. 35, 78

[Mumcuoglu *et al.*, 1996b] MUMCUOGLU, E. Ü., LEAHY, R. M. et CHERRY, S. R. (1996b). Bayesian reconstruction of pet images : methodology and performance analysis. *Physics in medicine and Biology*, 41(9):1777. 33

[Nakamoto *et al.*, 2002] NAKAMOTO, Y., OSMAN, M., COHADE, C., MARSHALL, L. T., LINKS, J. M., KOHLMYER, S. et WAHL, R. L. (2002). Pet/ct : comparison of quantitative tracer uptake between germanium and ct transmission attenuation-corrected images. *Journal of Nuclear Medicine*, 43(9):1137–1143. 37

[Nakamura *et al.*, 2010] NAKAMURA, K., GROUP, P. D. *et al.* (2010). Review of particle physics. *Journal of Physics G : Nuclear and Particle Physics*, 37(7A):075021. 132

[Nassiri *et al.*, 2015] NASSIRI, M. A., CARRIER, J.-F. et DESPRÉS, P. (2015). Fast gpu-based computation of spatial multigrid multiframe lmem for pet. *Medical & biological engineering & computing*, pages 1–13. 48

[Nelson *et al.*, 1985] NELSON, W. R., HIRAYAMA, H. et ROGERS, D. W. (1985). Egs4 code system. Rapport technique, Stanford Linear Accelerator Center, Menlo Park, CA (USA). 131

[NEMA, 2001] NEMA (2001). *Performance measurements of positron emission tomographs*. NEMA. 66

[Nickolls et Dally, 2010] NICKOLLS, J. et DALLY, W. J. (2010). The gpu computing era. *IEEE micro*, 30(2):56–69. 49

[Nikolopoulos *et al.*, 2006] NIKOLOPOULOS, D., KANDARAKIS, I., TSANTILAS, X., VALAIS, I., CAVOURAS, D. et LOUIZI, A. (2006). Comparative study using monte carlo methods of the radiation detection efficiency of lso, luap, gso and yap scintillators for use in positron emission imaging (pet). *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, 569(2):350–354. 16

[Ollinger, 1995] OLLINGER, J. M. (1995). Detector efficiency and compton scatter in fully 3d pet. *Nuclear Science, IEEE Transactions on*, 42(4):1168–1173. 36

[Ortuno *et al.*, 2010] ORTUNO, J., KONTAXAKIS, G., RUBIO, J., GUERRA, P. et SANTOS, A. (2010). Efficient methodologies for system matrix modelling in iterative image reconstruction for rotating high-resolution pet. *Physics in medicine and biology*, 55(7):1833. 113

[Ortuno *et al.*, 2006] ORTUNO, J. E., GUERRA-GUTIERREZ, P., RUBIO, J. L., KONTAXAKIS, G. et SANTOS, A. (2006). 3d-osem iterative image reconstruction for high-resolution pet using precalculated system matrix. *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, 569(2):440–444. 122

[O'sullivan, 1985] O'SULLIVAN, J. (1985). A fast sinc function gridding algorithm for fourier inversion in computer tomography. *Medical Imaging, IEEE Transactions on*, 4(4):200–207. 25

[Palmer *et al.*, 2005] PALMER, M. R., ZHU, X. et PARKER, J. A. (2005). Modeling and simulation of positron range effects for high resolution pet imaging. *Nuclear Science, IEEE Transactions on*, 52(5):1391–1395. 38

[Panin *et al.*, 2006] PANIN, V., KEHREN, F., MICHEL, C. et CASEY, M. (2006). Fully 3-d pet reconstruction with system matrix derived from point source measurements. *IEEE Transactions on Medical Imaging*, 25:907 – 921. 78, 122

[Parra et Barrett, 1998] PARRA, L. et BARRETT, H. (1998). List-mode likelihood : Em algorithm and image quality estimation demonstrated on 2-d pet. *IEEE Transactions on Medical Imaging*, 17:228–235. 31

[Phelps *et al.*, 1975] PHELPS, M. E., HOFFMAN, E. J., HUANG, S.-C. et TER-POGOSSIAN, M. M. (1975). Effect of positron range on spatial resolution. *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, 16(7):649–652. 131

[Popescu et Lewitt, 2004] POPESCU, L. M. et LEWITT, R. M. (2004). Ray tracing through a grid of blobs. *Dans Nuclear Science Symposium Conference Record*, volume 6, pages 3983–3986. IEEE. 44

[Popescu *et al.*, 2004] POPESCU, L. M., MATEJ, S. et LEWITT, R. M. (2004). Iterative image reconstruction using geometrically ordered subsets with list-mode data. *Dans Nuclear Science Symposium Conference Record*, volume 6, pages 3536–3540. IEEE. 44

[Pratx *et al.*, 2006] PRATX, G., CHINN, G., HABTE, F., OLCOTT, P. et LEVIN, C. (2006). Fully 3-d list-mode osem accelerated by graphics processing units. *Dans IEEE Nuclear Science Symposium Conference Record*, pages 2196 – 2202. 35, 48, 56, 81

[Pratx *et al.*, 2009] PRATX, G., CHINN, G., OLCOTT, P. D. et LEVIN, C. S. (2009). Fast, accurate and shift-varying line projections for iterative reconstruction using the gpu. *Medical Imaging, IEEE Transactions on*, 28(3):435–445. 48, 81

[Pratx et Levin, 2011] PRATX, G. et LEVIN, C. (2011). Online detector response calculations for high-resolution pet image reconstruction. *Physics in medicine and biology*, 56(13):4023. 35, 83, 84

[Qi *et al.*, 1998] QI, J., LEAHY, R., CHERRY, S., CHATZIIOANNOU, A. et FARQUHAR, T. (1998). High-resolution 3d bayesian image reconstruction using the micropet small-animal scanner. *Physics in medicine and biology*, 43:1001 – 1013. 34, 35, 78, 122

[Qi et Leahy, 2006] QI, J. et LEAHY, R. M. (2006). Iterative reconstruction techniques in emission computed tomography. *Physics in medicine and biology*, 51(15):R541. 30

[Raczyński *et al.*, 2014] RACZYŃSKI, L., MOSKAL, P., KOWALSKI, P., WIŚLICKI, W., BEDNARSKI, T., BIAŁAS, P., CZERWIŃSKI, E., KOCHANOWSKI, A., KORCYL, G., KOWAL, J. *et al.* (2014). Novel method for hit-position reconstruction using voltage signals in plastic scintillators and its application to positron emission tomography. *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, 764:186–192. 17

[Radon, 1917] RADON, J. (1917). Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Akad. Wiss.*, 69:262–277. 23

[Rafecas *et al.*, 2004] RAFECAS, M., MOSLER, B., DIETZ, M., POGL, M., STAMATAKIS, A., MCELROY, D. P., ZIEGLER, S. *et al.* (2004). Use of a monte carlo-based probability matrix for 3-d iterative reconstruction of madpet-ii data. *Nuclear Science, IEEE Transactions on*, 51(5):2597–2605. 78, 113

[Rahmim *et al.*, 2005] RAHMIM, A., CHENG, J.-C., BLINDER, S., CAMBORDE, M.-L. et SOSSI, V. (2005). Statistical dynamic image reconstruction in state-of-the-art high-resolution pet. *Physics in medicine and biology*, 50:4887–4912. 45

[Rahmim *et al.*, 2004] RAHMIM, A., LENOX, M., READER, A., MICHEL, C., BURBAR, Z., RUTH, T. et SOSSI, V. (2004). Statistical list-mode image reconstruction for the high resolution research tomograph. *Physics in medicine and biology*, 49(18):4239. 20

[Rahmim *et al.*, 2008a] RAHMIM, A., TANG, J., LODGE, M., LASHKARI, S., AY, M., R., L., TSUI, B. et BENGEL, F. (2008a). Analytic system matrix resolution modeling in pet : an application to rb-82 cardiac imaging. *Physics in medicine and biology*, 53:5947 – 5965. 35, 38, 80

[Rahmim *et al.*, 2008b] RAHMIM, A., TANG, J., LODGE, M. A., LASHKARI, S., AY, M. R. et BENGEL, F. M. (2008b). Resolution modeled pet image reconstruction incorporating space-variance of positron range : Rubidium-82 cardiac pet imaging. *Dans Nuclear Science Symposium Conference Record*, pages 3643–3650. IEEE. 38, 134

[Reader *et al.*, 2002] READER, A., ALLY, S., BAKATSELOS, F., MANAVAKI, R., WALLEDGE, R., JEAVONS, A., JULYAN, P., ZHAO, S., HASTING, D. et ZWEIT, J. (2002). One-pass list-mode em algorithm for high-resolution 3-d pet image reconstruction into large arrays. *IEEE Transactions on Nuclear Science*, 49:693–699. 35

[Reader *et al.*, 1998] READER, A., ERLANDSSON, K., FLOWER, M. et OTT, R. (1998). Fast accurate iterative reconstruction for low-statistics positron volume imaging. *Physics in medicine and biology*, 43:835–846. 31

[Ruangma *et al.*, 2006] RUANGMA, A., BAI, B., LEWIS, J. S., SUN, X., WELCH, M. J., LEAHY, R. et LAFOREST, R. (2006). Three-dimensional maximum a posteriori (map) imaging with radiopharmaceuticals labeled with three cu radionuclides. *Nuclear medicine and biology*, 33(2):217–226. 38, 134

[Salvat *et al.*, 2006] SALVAT, F., FERNÁNDEZ-VAREA, J. M. et SEMPAU, J. (2006). Penelope-2006 : A code system for monte carlo simulation of electron and photon transport. *Dans Workshop Proceedings*, volume 4, page 7. 131

[Schmitt *et al.*, 1988] SCHMITT, D., KARUTA, B., CARRIER, C. et LECOMTE, R. (1988). Fast point spread function computation from aperture functions in high-resolution positron emission tomography. *IEEE Transactions on Medical Imaging*, 7:2 – 12. 35

[Schreibmann *et al.*, 2010] SCHREIBMANN, E., NYE, J. A., SCHUSTER, D. M., MARTIN, D. R., VOTAW, J. et FOX, T. (2010). Mr-based attenuation correction for hybrid pet-mr brain imaging systems using deformable image registration. *Medical physics*, 37(5):2101–2109. 37

[Schretter, 2006] SCHRETTER, C. (2006). A fast tube of response ray-tracer. *Medical Physics*, 33:4744–4748. 80

[Selivanov *et al.*, 2000] SELIVANOV, V., PICARD, Y., CADORETTE, J., RODRIGUE, S. et LECOMTE, R. (2000). Detector response models for statistical iterative image reconstruction in high resolution pet. *IEEE Transactions on Nuclear Science*, 47:1168 – 1175. 35, 78, 81

[Shao *et al.*, 1994] SHAO, L., FREIFELDER, R. et KARP, J. (1994). Triple energy window scatter correction technique in pet. *Medical Imaging, IEEE Transactions on*, 13(4):641–648. 41

[Shattuck *et al.*, 2002] SHATTUCK, D., RAPELA, J., ASMA, E., CHATZIOANNOU, A., QI, J. et LEAHY, R. (2002). Internet2-based 3d pet image reconstruction using a pc cluster. *Physics in Medicine and Biology*, 47(15):2785. 58

[Shepp et Vardi, 1982] SHEPP, L. A. et VARDI, Y. (1982). Maximum likelihood reconstruction for emission tomography. *IEEE Transactions on Medical Imaging*, 1:113 – 122. 30

[Siddon, 1985] SIDDON, R. (1985). Fast calculation of the exact radiological path length for a three-dimensional ct array. *Medical Physics*, 12:252–257. 35, 80

[Smith *et al.*, 2003] SMITH, M. F., MAJEWSKI, S., WEISENBERGER, A. G., KIEPER, D., RAYLMAN, R. R., TURKINGTON, T. G. *et al.* (2003). Analysis of factors affecting positron emission mammography (pem) image formation. *Nuclear Science, IEEE Transactions on*, 50(1):53–59. 80

[Somayajula *et al.*, 2005] SOMAYAJULA, S., ASMA, E. et LEAHY, R. M. (2005). Pet image reconstruction using anatomical information through mutual information based priors. *Dans Nuclear Science Symposium Conference Record*, volume 5, pages 2722–2726. IEEE. 33

[Sportelli *et al.*, 2011] SPORTELLI, G., ORTUNO, J. et SANTOS, A. (2011). Efficient rendering of regions of response in list-mode reconstruction for pet. *Dans IEEE Nuclear Science Symposium Conference Record*. 95

[Staelens *et al.*, 2004] STAELENS, S., D'ASSELER, Y., VANDENBERGHE, S., KOOLE, M., LEMAHIEU, I. et Van de WALLE, R. (2004). A three-dimensional theoretical model incorporating spatial detection uncertainty in continuous detector pet. *Physics in medicine and biology*, 49:2338 – 2351. 35

[Stark *et al.*, 1981] STARK, H., WOODS, J., PAUL, I. et HINGORANI, R. (1981). An investigation of computerized tomography by direct fourier inversion and optimum interpolation. *IEEE Transactions on Biomedical Engineering*, 28:495–505. 26

[Stearns *et al.*, 2003] STEARNS, C. W., MCDANIEL, D. L., KOHLMYER, S. G., ARUL, P. R., GEISER, B. P. et SHANMUGAM, V. (2003). Random coincidence estimation from single event rates on the discovery st pet/ct scanner. *Dans Nuclear Science Symposium Conference Record*, volume 5, pages 3067–3069. IEEE. 40

[Strul *et al.*, 2003] STRUL, D., SLATES, R., DAHLBOM, M., CHERRY, S. et MARSDEN, P. (2003). An improved analytical detector response function model for multilayer small-diameter pet scanners. *Physics in medicine and biology*, 48:980 – 995. 35

[Stute, 2010] STUTE, S. (2010). *Modélisation avancée en simulations Monte Carlo de tomographie par émission de positons pour l'amélioration de la reconstruction et de la quantification*. Thèse de doctorat, Sciences et Technologies de l'Information des Télécommunications et des Systèmes. 35

[Stute *et al.*, 2011] STUTE, S., BENOIT, D., MARTINEAU, A., REHFELD, N. et BUVAT, I. (2011). A method for accurate modelling of the crystal response function at a crystal sub-level applied to pet reconstruction. *Physics in medicine and biology*, 56:793 – 809. 87

[Sureau *et al.*, 2008] SUREAU, F. C., READER, A. J., COMTAT, C., LEROY, C., RIBEIRO, M.-J., BUVAT, I. et TRÉBOSSEN, R. (2008). Impact of image-space resolution modeling for studies with the high-resolution research tomograph. *Journal of Nuclear Medicine*, 49(6):1000–1008. 78

[Surti, 2015] SURTI, S. (2015). Update on time-of-flight pet imaging. *Journal of Nuclear Medicine*, 56(1):98–105. 43

[Surti *et al.*, 2006] SURTI, S., KARP, J., POPESCU, L., DAUBE-WITHERSPOON, M. et WERNER, M. (2006). Investigation of time-of-flight benefit for fully 3-d pet. *IEEE Transactions on Medical Imaging*, 25: 529 – 538. 43

[Tai *et al.*, 1997] TAI, Y.-C., CHATZIIOANNOU, A., DAHLBOM, M. et HOFFMAN, E. J. (1997). Investigation on deadtime characteristics for simultaneous emission-transmission data acquisition in pet. *Dans Nuclear Science Symposium*, volume 2, pages 1489–1493. IEEE. 40

[Tanaka et Amo, 1998] TANAKA, E. et AMO, Y. (1998). A fourier rebinning algorithm incorporating spectral transfer efficiency for 3d pet. *Physics in medicine and biology*, 43(4):739. 26

[Taschereau *et al.*, 2011] TASCHEREAU, R., RANNOU, F., et CHATZIIOANNOU, A. (2011). A modeled point spread function for a noise-free system matrix. *Dans IEEE Nuclear Science Symposium Conference Record*. 82

[Terstegge *et al.*, 1996] TERSTEGGE, A., WEBER, S., HERZOG, H., MULLER-GÄRTNER, H. et HALLING, H. (1996). High resolution and better quantification by tube of response modelling in 3d pet reconstruction. *Dans Nuclear Science Symposium, 1996. Conference Record.*, volume 3, pages 1603–1607. IEEE. 80

[Thompson, 1993] THOMPSON, C. (1993). The problem of scatter correction in positron volume imaging. *Medical Imaging, IEEE Transactions on*, 12(1):124–132. 13

[Tong *et al.*, 2010] TONG, S., ALESSIO, A. et KINAHAN, P. (2010). Noise and signal properties in psf-based fully 3d pet image reconstruction : an experimental evaluation. *Physics in medicine and biology*, 55(5):1453. 77

[Vardi *et al.*, 1985] VARDI, Y., SHEPP, L. et KAUFMAN, L. (1985). A statistical model for positron emission tomography. *American Statistical Association*, 80:8 – 20. 30

[Veklerov *et al.*, 1998] VEKLEROV, E., LLACER, J. et HOFFMAN, E. (1998). Mle reconstruction of a brain phantom using a monte carlo transition matrix and a statistical stopping rule. *IEEE Transactions on Nuclear Science*, 35:603 – 607. 35, 78

[Veklerov *et al.*, 1988] VEKLEROV, E., LLACER, J. et HOFFMAN, E. J. (1988). Mle reconstruction of a brain phantom using a monte carlo transition matrix and a statistical stopping rule. *Nuclear Science, IEEE Transactions on*, 35(1):603–607. 78, 113

[Visvikis *et al.*, 2014] VISVIKIS, D., MONNIER, F., BERT, J., HATT, M. et FAYAD, H. (2014). Pet/mr attenuation correction : where have we come from and where are we going ? *European journal of nuclear medicine and molecular imaging*, 41(6):1172–1175. 38

[Vollmar *et al.*, 2002] VOLLMAR, S., MICHEL, C., TREFFERT, J., NEWPORT, D., CASEY, M., KNÖSS, C., WIENHARD, K., LIU, X., DEFRISE, M. et HEISS, W. (2002). Heinzelcluster : accelerated reconstruction for fore and osem3d. *Physics in medicine and biology*, 47(15):2651. 58

[Vollmar *et al.*, 2000] VOLLMAR, S., WIENHARD, K., LERCHER, M., KNOSS, C., MICHEL, C., TREFFERT, J., SCHMAND, M., CASEY, M., NEWPORT, D., LUK, P. *et al.* (2000). Beehive : cluster reconstruction of 3-d pet data in a windows nt network using fore. *Dans Nuclear Science Symposium Conference Record*, volume 2, pages 15–213. IEEE. 58

[Vuduc *et al.*, 2010] VUDUC, R., CHANDRAMOWLISHWARAN, A., CHOI, J., GUNEY, M. et SHRINGARPURE, A. (2010). On the limits of gpu acceleration. *Dans Proceedings of the 2nd USENIX conference on Hot topics in parallelism*, pages 13–13. USENIX Association. 55

[Wallach, 2011] WALLACH, D. (2011). *Compensation du mouvement respiratoire en TEP/TDM à l'aide de la super-résolution*. Thèse de doctorat, Université de Bretagne Occidentale. 157

[Wang *et al.*, 2006] WANG, W., HU, Z., GUALTIERI, E., PARMA, M., WALSH, E., SEBOK, D., HSIEH, Y., TUNG, C., SONG, X., GRIESMER, J. *et al.* (2006). Systematic and distributed time-of-flight list mode pet reconstruction. *Dans Nuclear Science Symposium Conference Record*, volume 3, pages 1715–1722. IEEE. 58

[Watson, 2000] WATSON, C. (2000). New, faster, image-based scatter correction for 3-d pet. *IEEE Transactions on Nuclear Science*, 47:1587 – 1594. 42

[Watson, 2007] WATSON, C. (2007). Extension of single scatter simulation to scatter correction of time-of-flight pet. *IEEE Transactions on Nuclear Science*, 54:1679 – 1686. 42

[Watson *et al.*, 1996] WATSON, C. C., NEWPORT, D. et CASEY, M. E. (1996). A single scatter simulation technique for scatter correction in 3d pet. *Dans Three-dimensional image reconstruction in radiology and nuclear medicine*, pages 255–268. Springer. 42

[Williams *et al.*, 2003] WILLIAMS, J. J., MCDANIEL, D. L., KIM, C. L. et WEST, L. J. (2003). Detector characterization of discovery st whole-body pet scanner. *Dans Nuclear Science Symposium Conference Record*, volume 2, pages 717–721. IEEE. 39

[Wilson *et al.*, 1994] WILSON, D. W., TSUI, B. M. et BARRETT, H. H. (1994). Noise properties of the em algorithm. ii. monte carlo simulations. *Physics in Medicine and Biology*, 39(5):847. 32

[Wirrwar *et al.*, 1997] WIRRWAR, A., VOSBERG, H., HERZOG, H., HALLING, H., WEBER, S. et MULLER-GARTNER, H.-W. (1997). 4.5 tesla magnetic field reduces range of high-energy positrons-potential implications for positron emission tomography. *Nuclear Science, IEEE Transactions on*, 44(2):184–189. 38, 133

[Yamaya *et al.*, 2005] YAMAYA, T., HAGIWARA, N., OBI, T., YAMAGUCHI, M., OHYAMA, N., KITAMURA, K., HASEGAWA, T., HANEISHI, H., YOSHIDA, E., INADAMA, N. *et al.* (2005). Transaxial system models for jpet-d4 image reconstruction. *Physics in medicine and biology*, 50(22):5339. 86

[Yao *et al.*, 2012] YAO, R., RAMACHANDRA, R., MAHAJAN, N., RATHOD, V., GUNASEKAR, N., PANSE, N., MA, T., JIAN, Y., YAN, J. et CARSON, R. (2012). Assessment of a three-dimensional line-of-response probability density function system matrix for pet. *Physics in medicine and biology*, 57:6827 – 6848. 78, 88, 113

[Zaidi *et al.*, 2004] ZAIDI, H., MONTANDON, M.-L. et SLOSMAN, D. O. (2004). Attenuation compensation in cerebral 3d pet : effect of the attenuation map on absolute and relative quantitation. *European journal of nuclear medicine and molecular imaging*, 31(1):52–63. 33

[Zhao et Reader, 2002] ZHAO, H. et READER, A. (2002). Fast projection algorithm for voxel arrays with object dependent boundaries. *Dans IEEE Nuclear Science Symposium Conference Record*. 95

[Zhou et Qi, 2011] ZHOU, J. et QI, J. (2011). Fast and efficient fully 3d pet image reconstruction using sparse system matrix factorization with gpu acceleration. *Physics in medicine and biology*, 56(20): 6739. 35, 48, 80, 88

Résumé

En tomographie par émission de positon, les images souffrent d'un bruit élevé et d'une résolution faible. Leur reconstruction à l'aide d'un processus itératif nécessite d'estimer la réponse du système (scanner et patient) et leur qualité dépend directement de la précision de cette estimation. Des méthodes fidèles et rapides d'exécution existent pour estimer les composantes d'atténuation, de diffusion, les coïncidences fortuites ainsi que les temps morts. Cette thèse propose des méthodes de modélisation précises des composantes du système associées au détecteur du scanner et au parcours du positon. Une nouvelle méthode de parallélisation de la reconstruction sur plateforme multi-GPU basée sur une découpe du volume reconstruit est aussi proposée, afin d'exploiter la puissance de calcul d'une telle architecture pour accélérer la reconstruction. Le modèle de la réponse du détecteur proposé exploite une approche multiligne et intègre les effets associés à la géométrie du détecteur et à la diffusion intercristaux. Une étude d'évaluation basée sur des données obtenues par simulation Monte-Carlo (SMC) montre, par rapport à l'état de l'art, une amélioration du rapport contraste sur bruit et de la résolution des images reconstruites. Le modèle du parcours du positon proposé repose sur une SMC simplifiée, intégrée à l'opération de projection dans la reconstruction. Cette simulation implémentée sur GPU fournit des résultats proches de ceux obtenus avec la plateforme GATE pour des temps d'exécution de trois ordres de grandeur plus courts. Une étude d'évaluation montre que cette méthode permet une amélioration importante du contraste et de la résolution, sans introduire d'artefacts.

Mots-clés : Tomographie par émission, Tomographie (mathématiques), Processeurs à hautes performances, Simulation par ordinateur, Monte-Carlo, Méthode de Positons, Particules (physique nucléaire), Statistique -- Logiciels

Abstract

In positron emission tomography, reconstructed images suffer from a high noise level and a low resolution. Iterative reconstruction processes require an estimation of the system response (scanner and patient) and the quality of the images depends on the accuracy of this estimate. Accurate and fast to compute models already exists for the attenuation, scattering, random coincidences and dead times. Thus, this thesis focuses on modeling the system components associated with the detector response and the positron range. A new multi-GPU parallelization of the reconstruction based on a cutting of the volume is also proposed to speed up the reconstruction exploiting the computing power of such architectures. The proposed detector response model is based on a multi-ray approach that includes all the detector effects as the geometry and the scattering in the crystals. An evaluation study based on data obtained through Mote Carlo simulation (MCS) showed this model provides reconstructed images with a better contrast to noise ratio and resolution compared with those of the methods from the state of the art. The proposed positron range model is based on a simplified MCS, integrated into the forward projector during the reconstruction. A GPU implementation of this method allows running MCS three order of magnitude faster than the same simulation on GATE, while providing similar results. An evaluation study shows this model integrated in the reconstruction gives images with better contrast recovery and resolution while avoiding artifacts.

Keywords : Emission tomography, Tomography (mathematics), High performance processors, Computer simulation, Monte Carlo method, Positrons, Particles (nuclear physic), Statistic -- software

TELECOM Bretagne

n° d'ordre : 2015telb0363

Télécom Bretagne

Technopôle Brest-Iroise - CS 83818 - 29238 Brest Cedex 3

Tél : + 33(0) 29 00 11 11 - Fax : + 33(0) 29 00 10 00