



HAL
open science

Reconstruction tomographique en contraste de phase : dans le cadre du projet XPulse

Emile Barjou

► **To cite this version:**

Emile Barjou. Reconstruction tomographique en contraste de phase : dans le cadre du projet XPulse. Autre [cs.OH]. Université de Bordeaux, 2023. Français. NNT : 2023BORD0161 . tel-04223122

HAL Id: tel-04223122

<https://theses.hal.science/tel-04223122v1>

Submitted on 29 Sep 2023

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 PRÉSENTÉE
POUR OBTENIR LE GRADE DE
DOCTEUR
DE L'UNIVERSITÉ DE BORDEAUX

ECOLE DOCTORALE MATHÉMATIQUE ET INFORMATIQUE

Spécialité : Informatique

Par **Emile BARJOU**

Reconstruction tomographique en contraste de phase
dans le cadre du projet XPulse

Sous la direction de : **Pascal DESBARATS**

Soutenue le 22 Juin 2023

Membres du jury :

| | | | |
|-----------------------------|---------------------|----------------------------|--------------|
| M. Julien BERT | Maitre de conf. HDR | Univ. Bretagne Occidentale | Rapporteur |
| M. Serge MIGUET | Professeur | Univ. Lumière Lyon 2 | Rapporteur |
| M. Guillaume BLIN | Professeur | Univ. Bordeaux | Président |
| M. Julio CESAR DA SILVA | Docteur | Univ. Grenoble Alpes | Examineur |
| M. Pascal DESBARATS | Professeur | Univ. Bordeaux | Directeur |
| Mme. Francesca MASTROPIETRO | Docteure | ALPhANOV | Examinatrice |
| M. Aboubakr BAKKALI | Ingénieur R&D | ALPhANOV | Invité |
| M. Benoit RECUR | Docteur | Academia Sinica | Invité |

Reconstruction tomographique en contraste de phase

Résumé : Cette thèse s'inscrit dans le projet XPulse, visant à développer un système de mammographie innovant utilisant l'imagerie de phase et la tomographie pour extraire le maximum d'information tout en réduisant l'exposition du patient aux rayons X. Ce projet résulte d'une collaboration entre différents acteurs et cette thèse est réalisée avec ALPhANOV, coordinateur du projet et le LaBRI, le laboratoire où j'effectue mon doctorat. Dans ce projet, j'ai travaillé sur le traitement des données acquises avec les différentes installations d'imagerie. Cela implique le prétraitement des données brutes, la reconstruction tomographique et l'extraction de la phase.

Pour la reconstruction tomographique, j'ai évalué les différentes implémentations à notre disposition puis ai développé un logiciel sur-mesure permettant de répondre aux spécificités de nos installations. La contrainte principale a été la gestion de la mémoire, les jeux de données usuels s'approchant du téraoctet.

Pour la récupération de phase, deux méthodes ont été utilisées au cours du projet : la propagation libre et l'imagerie par masque de Hartmann. Mon travail sur ce volet a été d'étudier puis d'implémenter les méthodes existantes de traitement de telles images, puis de les intégrer au mieux dans la chaîne de traitement.

Mots-clés : Reconstruction tomographique, GPGPU, gestion mémoire, contraste de phase, propagation libre, masque de Hartmann

Phase contrast tomographic reconstruction

Abstract: This thesis is part of the XPulse project, which aims to develop an innovative mammography system using phase imaging and tomography to extract the maximum information while reducing the patient's exposure to X-rays. This project is the result of a collaboration between different actors and this thesis is realized with ALPhANOV, coordinator of the project and LaBRI, the laboratory where I am doing my PhD. In this project, I worked on the processing of data acquired with the different imaging facilities. This involves preprocessing of the raw data, tomographic reconstruction and phase extraction.

For the tomographic reconstruction, I evaluated the different implementations available to us and then developed a custom software to meet the specificities of our facilities. The main constraint was memory management, as the usual data sets are close to terabytes.

For phase recovery, two methods were used during the project: free propagation and Hartmann mask imaging. My work on this part of the project was to study and implement the existing methods of processing such images, and then to integrate them as well as possible in the processing chain.

Keywords: Tomographic reconstruction, GPGPU, memory management, phase contrast, free propagation, Hartmann mask

Unité de recherche

Université de Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, 33400 Talence, France

Centre technologique

ALPhANOV, Institut d'Optique d'Aquitaine, Rue François Mitterrand, 33400 Talence, France

Remerciements

Je remercie tout d'abord mon maître de thèse et ancien professeur, M. Desbarats, directeur adjoint du LaBRI et professeur de l'Université de Bordeaux, qui m'a permis de réaliser cette thèse et m'a accompagné durant celle-ci, et qui a su se rendre disponible malgré ses obligations, ainsi que Dr. Recur qui grâce à son expertise a pu m'aider à prendre en main le sujet de la tomographie.

Je remercie le centre technologique ALPhANOV, ainsi que son directeur M. Appert-Collin, pour m'avoir accueilli et financé durant ma thèse dans d'excellentes conditions.

Je remercie également toute l'équipe XPulse : Aboubakr, Benjamin, Francesca, Octave, Panyi, Damien, Pauline, ainsi que ceux qui sont travaillé dans cette équipe au cours du projet. Ce fut un plaisir de travailler avec cette équipe tout au long de ma thèse.

Merci aux membres du jury que je n'ai pas encore cités : M. Bert, maître de conférence de l'Université Bretagne Occidentale, M. Miguet, professeur de l'Université Lumière Lyon 2, M. Blin, professeur de l'Université de Bordeaux et M. Da Silva, docteur de l'Université Grenoble Alpes, d'avoir accepté d'évaluer mon travail, ainsi que pour leurs relectures et corrections de ce manuscrit.

Merci également à toute l'équipe pédagogique du LaBRI, avec laquelle ce fut un plaisir d'étudier pendant huit ans à l'Université de Bordeaux, et qui m'a donné envie de continuer mes études au-delà du Master et de réaliser cette thèse.

Merci à mes collègues de bureau doctorants du LaBRI : Christophe, Adrien et Marie. Nous avons pu réaliser notre thèse de concert sur des sujets variés en nous aidant mutuellement sur nos domaines de compétences respectifs.

Merci enfin à mes parents pour m'avoir soutenu inconditionnellement durant toute la durée de mes études.

Table des matières

| | |
|--|-----------|
| Introduction | 11 |
| I Théorie | 13 |
| 1 Imagerie par rayons X | 15 |
| 1.1 Principe de l'imagerie par rayons X | 16 |
| 1.1.1 Onde électromagnétique | 16 |
| 1.1.2 Mesure de l'absorption | 17 |
| 1.1.3 Dangerosité des rayons X en imagerie médicale | 21 |
| 1.2 Principe de la reconstruction tomographique | 22 |
| 1.2.1 Principe de l'acquisition tomographie | 22 |
| 1.2.2 Inversion de la tomographie | 23 |
| 1.2.3 Principe de la reconstruction itérative | 25 |
| 1.3 Conclusion | 28 |
| 2 Imagerie de phase par rayons X | 31 |
| 2.1 Principe de l'imagerie de phase | 31 |
| 2.1.1 La phase d'une onde électromagnétique | 31 |
| 2.1.2 L'imagerie par contraste de phase | 32 |
| 2.2 Méthode d'imagerie de phase | 33 |
| 2.2.1 Interférométrie cristalline | 33 |
| 2.2.2 Imagerie par analyseur | 34 |
| 2.2.3 Imagerie par illumination de bord et par masque de Hartmann | 34 |
| 2.2.4 Imagerie par <i>speckle</i> | 35 |
| 2.2.5 Imagerie par propagation libre | 35 |
| 2.3 État de l'art des méthodes de phase en propagation libre | 38 |
| 2.3.1 Les différentes méthodes d'extractions directes de la phase en PBI | 38 |
| 2.3.2 Méthodes Itératives de récupération de la phase | 41 |
| 2.3.3 La méthode d'extraction de phase retenue | 46 |
| 2.4 État de l'art des méthodes de phase Hartmann | 47 |
| 2.4.1 Masque de Hartmann | 47 |
| 2.4.2 Algorithmes d'intégration de la phase | 48 |
| 2.5 Conclusion | 49 |

| | | |
|------------|--|-----------|
| II | Travaux/Méthodes | 51 |
| 3 | Le projet XPulse | 53 |
| 3.1 | Setup et Objectif | 53 |
| 3.1.1 | Le projet XPulse | 53 |
| 3.1.2 | Les configurations d'imagerie du projet | 54 |
| 3.1.3 | Système d'imagerie par masque de Hartmann | 56 |
| 3.2 | État de l'art des logiciels de tomographie | 57 |
| 3.3 | Chaîne de traitement | 59 |
| 3.3.1 | Chaîne de traitement | 59 |
| 3.3.2 | Méthode de traitement | 60 |
| 3.4 | Conclusion | 63 |
| 4 | Nouvelles méthodes d'extraction de phase | 65 |
| 4.1 | Automatisation du calcul du ratio δ/β | 65 |
| 4.1.1 | Méthode itérative basée sur la formule de Paganin | 65 |
| 4.1.2 | Condition d'arrêt | 66 |
| 4.2 | Extraction de la phase par masque de Hartmann en tomographie | 68 |
| 4.2.1 | Problématique | 68 |
| 4.2.2 | Extraction des sous-faisceaux par décalage à la référence | 68 |
| 4.2.3 | Suivi des sous-faisceaux | 69 |
| 4.2.4 | Intégration à la reconstruction tomographique | 70 |
| 4.3 | Conclusion | 72 |
| III | Implémentation | 73 |
| 5 | Problématiques d'implémentation | 75 |
| 5.1 | Projection | 76 |
| 5.1.1 | Opérateurs de projection | 76 |
| 5.1.2 | Projection entre une source et un détecteur | 77 |
| 5.1.3 | Construction de la géométrie | 80 |
| 5.1.4 | Construction de la matrice de projection | 83 |
| 5.1.5 | Projection d'un point par une matrice de projection | 84 |
| 5.1.6 | Multi-détecteur | 85 |
| 5.2 | Gestion de la mémoire | 86 |
| 5.2.1 | Problématique gestion mémoire | 86 |
| 5.2.2 | Implémentation naïve | 87 |
| 5.2.3 | Minimiser l'utilisation mémoire | 87 |
| 5.2.4 | Minimiser le transfert de données | 89 |
| 5.2.5 | Méthode implémentée | 92 |
| 5.3 | Échantillonnage de la projection | 95 |
| 5.3.1 | Les techniques d'interpolation | 95 |
| 5.3.2 | L'interpolation par Monte-Carlo | 97 |
| 5.4 | Conclusion | 97 |
| 6 | Validation | 99 |
| 6.1 | Métriques | 99 |
| 6.2 | Ratio Paganin | 100 |

| | | |
|-------|---|-----|
| 6.3 | Qualité Southwell 3D | 102 |
| 6.3.1 | Méthodologie | 102 |
| 6.3.2 | Résultats | 103 |
| 6.4 | Comparaison de la qualité de reconstruction | 107 |
| 6.4.1 | Fantôme numérique | 107 |
| 6.4.2 | Mise en place de la géométrie | 107 |
| 6.4.3 | Qualité de l'interpolation | 109 |
| 6.4.4 | Comparaison des résultats | 110 |
| 6.5 | Validation de la correction d'alignement | 113 |
| 6.5.1 | Axe de rotation | 113 |
| 6.5.2 | Position de la source | 113 |
| 6.5.3 | Rotation du détecteur | 114 |
| 6.5.4 | Détecteur segmenté | 114 |
| 6.6 | Comparaison des temps de reconstruction | 116 |
| 6.6.1 | Performance globale | 116 |
| 6.6.2 | Utilisation IO | 117 |
| 6.6.3 | Impact de la projection par voxel | 118 |
| 6.6.4 | Performance avec différents GPUs | 119 |
| 6.7 | Données réelles | 120 |
| 6.7.1 | Escargot | 120 |
| 6.7.2 | Noix | 120 |
| 6.7.3 | Fantôme de sein | 120 |
| 6.8 | Conclusion | 128 |

7 Conclusion

Introduction

Le projet XPulse a pour but de concevoir un appareil d'imagerie médicale destiné à la détection précoce du cancer du sein. Pour ce faire, une source de rayons X générés par laser et un détecteur spécialisé ont été développés.

C'est un projet collaboratif regroupant un partenaire clinique (l'institut Bergonié), un partenaire académique (le CELIA), et deux partenaires industriels (Amplitude systems et Imagine Optic). ALPhANOV est le coordinateur de ce projet, qui est financé par la région Nouvelle-Aquitaine et le FEDER.

Les systèmes de mammographie actuels imposent de compresser le sein pour effectuer une image par rayons X. Le diagnostic est relativement compliqué avec cette technique, car les microcalcifications sont difficiles à classifier et nécessitent souvent une biopsie (prélèvement de tissus) pour une analyse. Le prélèvement est d'autant plus compliqué que la compression du sein déforme la géométrie mesurée, en plus d'être inconfortable pour la patiente.

Le but du système d'imagerie XPulse est de réaliser une reconstruction tomographique en trois dimensions afin d'obtenir un volume permettant une localisation précise des structures, et en utilisant des techniques d'imagerie avancées permettant de faciliter le diagnostic. Les valeurs mesurées dépendront de la modalité d'imagerie utilisée : absorption, contraste de phase en propagation libre (CPPL) et contraste de phase par masque de Hartmann. Dans le cadre de la mammographie, l'imagerie par contraste de phase offre une meilleure qualité d'image par rapport à l'imagerie classique par absorption (meilleur contraste entre les tissus) [1].

Les modalités d'imagerie par contraste de phase nécessitent des prétraitements complexes afin d'obtenir les valeurs recherchées (c'est-à-dire directement liée au retard de phase causé par l'objet imagé). Pour le CPPL comme pour avec un masque de Hartmann, le système d'imagerie est configuré dans des cas limites afin d'obtenir la meilleure résolution possible, ce qui complexifie encore davantage les traitements à effectuer.

Cette thèse s'inscrit dans ce projet afin de concevoir et de développer les méthodes algorithmiques de traitement des données, reconstruction tomographique et extraction des valeurs à partir des données capturées nécessaires à sa réalisation.

Les travaux se divisent principalement en deux axes : la reconstruction tomographique et l'imagerie de phase, avec pour but d'intégrer les deux pour obtenir de meilleurs résultats et d'accélérer le traitement.

Les différentes modalités d'imagerie utilisées viennent avec des contraintes différentes pour la reconstruction et l'extraction de la phase. Lors de l'acquisition en absorption ou

en CPPL, la quantité de données générée par le système d'imagerie est très importante, ce qui met à l'épreuve le logiciel de traitement, mais surtout le logiciel de reconstruction tomographique, car la taille du volume est bien supérieure à celle de la mémoire vive disponible sur les ordinateurs classiques. En plus de devoir contrôler l'utilisation de la mémoire, la taille des données a également un impact important sur le temps de reconstruction, qui doit rester dans un intervalle raisonnable au vu de son application médicale.

Ma thèse se partage en trois parties principales. Nous allons commencer par les points théoriques nécessaires pour comprendre la physique de l'imagerie par rayons X, de la tomographie, et où se placent mes travaux dans l'état de l'art des techniques d'imagerie en contraste de phase.

Dans une seconde partie, nous verrons plus en détails le projet XPulse sur lequel ma thèse se porte, ainsi que les problématiques qui y sont liées et les méthodes que j'ai mises en place pour y répondre.

Pour finir, je présenterai quelques points d'intérêt sur l'implémentation de mon logiciel de reconstruction tomographique dédié aux installations du projet, et enfin nous évaluerons les résultats obtenus par mes implémentations, pour la reconstruction et l'extraction de phase. Pour cela je me comparerai à des implémentations existantes quand possible.

Première partie

Théorie

Chapitre 1

Imagerie par rayons X

L'imagerie par rayons X est une technique d'imagerie découverte en 1895 par le physicien Wilhelm Röntgen. Contrairement à la photographie qui capture la lumière visible et nous informe principalement sur la surface de l'objet imagé, les rayons X peuvent traverser des objets même volumineux, permettant de réaliser une image contenant des informations sur sa composition interne. Cette technique s'est rapidement généralisée à un grand nombre de champs d'application, autant médicale qu'industrielle, grâce à ses avantages et sa simplicité de mise en œuvre.

Plus récemment, les détecteurs de rayons X numériques, couplés aux avancées en termes de puissance de calcul, ont permis l'émergence de l'acquisition tomographique calculée par ordinateur, consistant en de multiples acquisitions 2D autour de l'objet dans le but de reconstruire un volume 3D. Bien que plus complexe, autant lors de l'acquisition que lors du traitement des données, cette technique offre des informations beaucoup plus précises que l'imagerie 2D. Elle s'est développée dans de nombreux domaines d'applications, mais ne s'est réellement démocratisée que depuis les années 2000 grâce à l'explosion des performances de calcul des ordinateurs.

Cette utilisation grandissante s'est néanmoins heurtée à un problème en imagerie médicale, car si l'imagerie par rayons X permet d'obtenir des résultats très utiles, la question de la dose de rayonnement absorbée inhérent aux rayons X est une contrainte importante et encore étudiée, d'autant plus pour une acquisition tomographique où de nombreuses images sont nécessaires.

Dans ce chapitre, je présenterai les principes physiques de l'imagerie par rayons X, l'interaction de ceux-ci avec la matière, ainsi que les problèmes rencontrés lors de l'utilisation de cette technique d'imagerie dans un contexte clinique.

Je parlerai ensuite du principe de l'acquisition tomographique par rayons X et de l'implémentation de la reconstruction volumétrique.

1.1 Principe de l'imagerie par rayons X

1.1.1 Onde électromagnétique

L'imagerie par rayons X est une technique très répandue, notamment dans le milieu médical, qui permet un contrôle non invasif. Le principe de cette technique d'imagerie est d'illuminer un objet avec un rayonnement électromagnétique, puis d'observer la modification du rayonnement induite par l'objet à l'aide d'un détecteur.

L'énergie d'un photon (quantum d'énergie du rayonnement électromagnétique) dépend de sa longueur d'onde λ , de la constante de Planck $h = 6.63 \cdot 10^{-34}$ Js et de la célérité de la lumière $c = 3 \cdot 10^8$ m/s (équation (1.1)). Les rayons X ont une longueur d'onde comprise entre une dizaine de nanomètres et quelques picomètres (figure 1.1), ce qui correspond à une énergie comprise entre 100eV et 100keV. À titre de comparaison, la lumière visible est comprise entre 380 et 750 nanomètres, ce qui équivaut à une énergie comprise entre 1.7 eV et 3.3 eV.

$$E = \frac{hc}{\lambda} \quad (1.1)$$

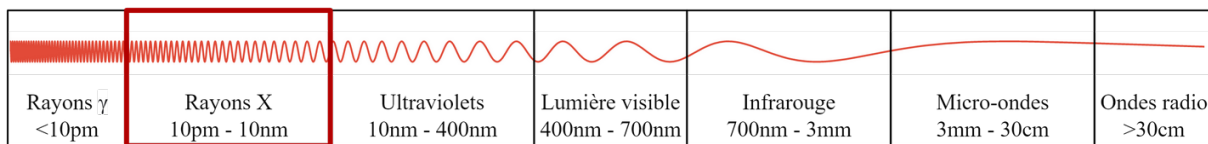


FIGURE 1.1 – Classification du spectre électromagnétique selon la longueur d'onde.

L'équation (1.2) représente une onde se propageant dans le vide à la position \vec{r} et au temps de propagation t selon son amplitude E_0 , la fréquence angulaire $\omega = 2\pi\frac{c}{\lambda}$, le nombre d'onde \vec{k} (1.4), sa direction de polarisation \vec{e} et sa phase à l'origine Φ .

$$\vec{E}(\vec{r}, t) = E_0 \vec{e} \cdot \cos(\vec{k} \cdot \vec{r} - \omega t + \Phi) \quad (1.2)$$

En plus de sa longueur d'onde, une onde électromagnétique peut être décrite par son amplitude et sa phase. Quand l'onde se propage dans un milieu, celui-ci peut altérer son amplitude et décaler sa phase. En règle générale, une onde traversant un objet verra son amplitude réduite (on parle d'absorption), et sa phase retardée ou avancée selon le type de rayonnement et les milieux traversés (figure 1.2).

Ce changement dépend du matériau ainsi que de l'épaisseur de l'objet traversé. On peut modéliser cet effet à l'aide de l'indice de réfraction complexe n du matériau (équation (1.3)), comprenant une partie imaginaire β dont dépend l'absorption et une partie réelle δ dont dépend le déphasage. Cet indice est relié à la formule du front d'onde par le nombre d'onde k tel que présenté dans l'équation (1.4).

$$n = 1 - \delta + i\beta \quad (1.3)$$

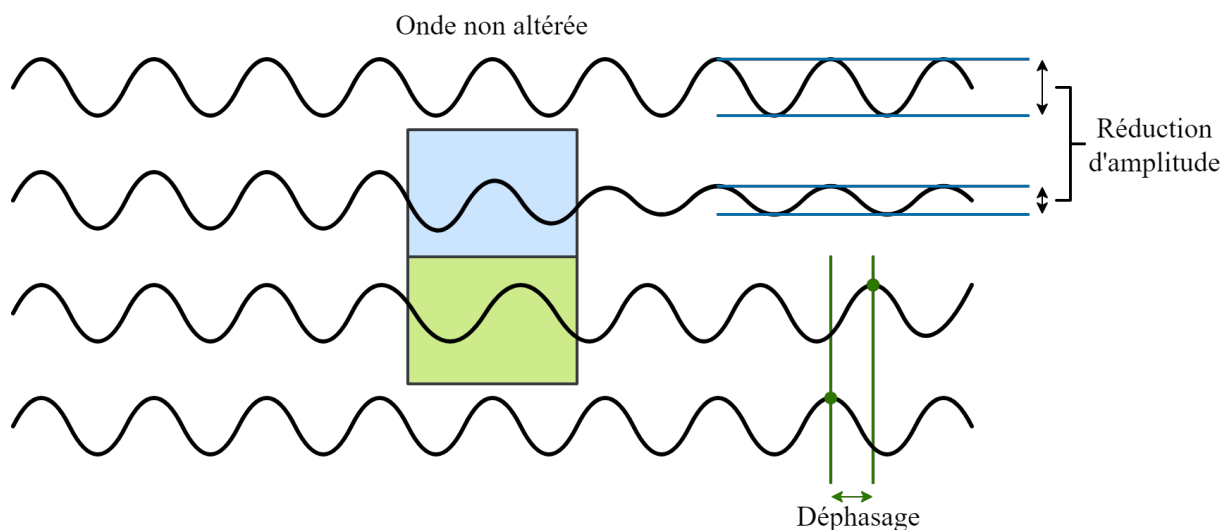


FIGURE 1.2 – Schéma de l'altération d'une onde électromagnétique passant à travers un objet. L'objet bleu réduit l'amplitude, tandis que l'objet vert retarde la phase.

$$k = \frac{2\pi}{\lambda} \quad (1.4)$$

Cette valeur est une propriété intrinsèque du matériau (décrit comme un milieu diélectrique, soit en l'absence de charge électrique), mais qui varie avec l'énergie de l'onde qui le traverse. Différentes énergies sont donc plus ou moins adaptées selon l'application étudiée.

Le coefficient d'atténuation linéaire μ d'un matériau peut être calculé à partir de l'indice β selon l'équation (1.5), pour un objet homogène. Ce coefficient nous renseigne sur l'absorption d'une onde en fonction de la distance traversée, et selon l'énergie utilisée, pour un matériau donné.

$$\mu = 4\pi\beta/\lambda \quad (1.5)$$

Le déphasage est calculé comme étant le décalage entre la valeur de la phase à l'entrée et à la sortie de l'objet. Elle peut se mesurer à l'aide de la formule suivante (1.6), avec λ la longueur d'onde du rayonnement et t l'épaisseur de l'objet traversée (pour le cas d'un objet homogène) :

$$\phi = -\frac{2\pi}{\lambda} \delta t \quad (1.6)$$

1.1.2 Mesure de l'absorption

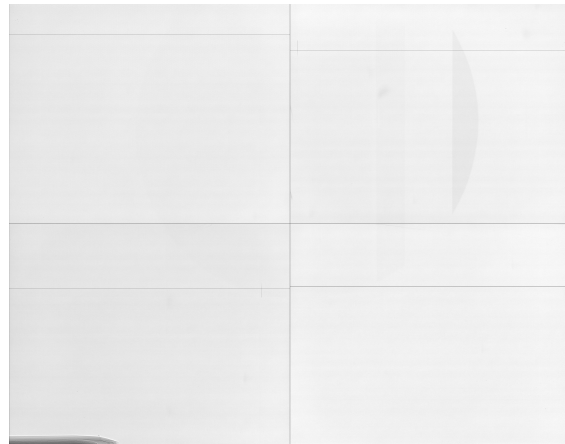
En imagerie par rayon X en régime d'absorption, on mesure la variation de l'amplitude du rayonnement qui arrive sur un détecteur. Le rayonnement émis par une source est atténué en passant à travers l'objet. On obtient alors une image en mesurant l'intensité du rayonnement pour chaque pixel du détecteur, avec un fond lumineux et un objet plus sombre (figures 1.3a, 1.3c).

$$I(x, y)$$



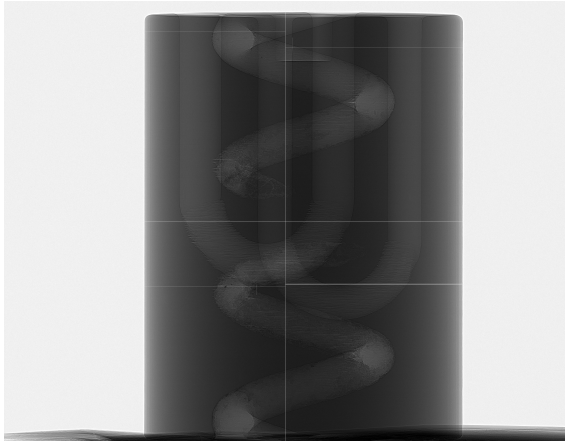
(a) Exemple d'image obtenue avec les valeurs brutes d'intensité mesurée par le détecteur

$$I_0(x, y)$$



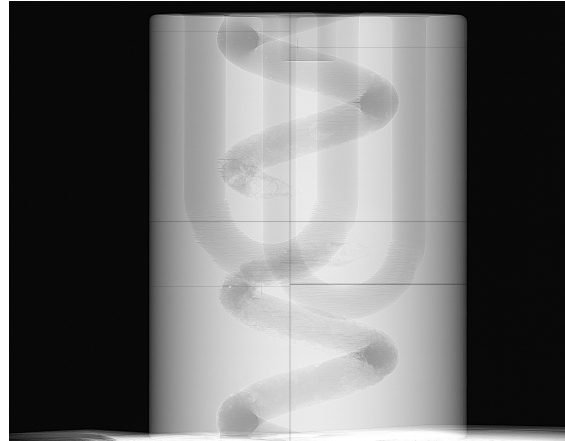
(b) Exemple de *flatfield*, soit une acquisition sans objet. Cela permet de mesurer les éventuelles variations d'intensité de la source et de sensibilité du détecteur.

$$\frac{I(x, y)}{I_0(x, y)}$$



(c) Exemple d'image normalisée par un *flatfield*. On remarque que les variations de sensibilité du détecteur sont largement atténué par cette opération.

$$-\log\left(\frac{I(x, y)}{I_0(x, y)}\right)$$



(d) Exemple d'image d'absorption après normalisation et application de l'opposé du logarithme. C'est cette image qui sera utilisée pour effectuer une reconstruction tomographique.

FIGURE 1.3 – Illustration de l'état d'une projection à différentes étapes du processus de normalisation. L'objet imagé est un cylindre imprimé en 3D percé de trois canaux (une spirale et deux 'U').

L'équation (1.7) exprime l'intensité du signal mesuré, à l'aide du coefficient d'atténuation linéaire μ qui dépend lui-même de l'indice β .

$$\begin{aligned} I(x, y) &= I_0(x, y)e^{\int_{z=0}^t (-\mu(x, y, z))} \\ &= I_0(x, y)e^{-\mu t} \quad (\text{pour un matériau homogène}) \end{aligned} \quad (1.7)$$

Une fois l'intensité du rayonnement mesurée, on définit la valeur d'absorption A dû à l'objet comme :

$$A(x, y) = -\log\left(\frac{I(x, y)}{I_0(x, y)}\right) \quad (1.8)$$

Dans cette formule, la valeur I_0 représente l'intensité initiale du rayonnement. Celle-ci peut être approximée expérimentalement par une image prise sans l'objet, avant ou après l'acquisition. On obtient une image de référence appelée *flatfield* qui remplace alors le I_0 . En effet, on peut considérer que le rayonnement avant l'objet est identique à un rayonnement dans les mêmes conditions traversant uniquement de l'air, si on travaille avec une énergie suffisante pour que l'absorption de l'air soit négligeable. Un exemple de *flatfield* et d'image normalisée est présenté en figures 1.3b, 1.3c et 1.3d, à partir d'une image brute en figure 1.3a.

On peut remarquer que cette méthode permet donc d'imager une valeur dépendante uniquement de l'indice β , pour une longueur d'onde donnée. L'indice δ se retrouve dans le déphasage du rayonnement, comme décrit par l'équation (1.6), mais l'imagerie par absorption n'est pas capable de le mesurer.

On peut également noter que l'équation (1.7) ne considère qu'un seul coefficient μ , et donc qu'une seule énergie (rayonnement monochromatique). Si ce type de rayonnement existe, il nécessite de filtrer le rayonnement, ce qui n'est généralement pas le cas pour les sources médicales ou de laboratoire, qui ne génèrent pas assez de photons pour pouvoir n'en garder que la proportion ayant une énergie donnée. Les sources plus classiques (comme les tubes à rayons X) émettent un spectre d'émission plus large, donc la mesure finale comprendra les contributions de toutes les énergies présentes.

Pour représenter une source polychromatique, on peut considérer l'intensité résultant comme l'intégrale des intensités sur l'intervalle d'énergies délivrées par la source.

$$I = \int_{\lambda} I_0(\lambda)e^{-\mu_{\lambda}t} d\lambda \quad (1.9)$$

Cela implique que différentes énergies dans un même rayonnement seront plus ou moins absorbées par l'objet imagé. Les gammes d'énergies totalement absorbées par l'objet lorsque celui-ci est suffisamment épais ne sont donc pas visibles sur toute l'image, ce qui se traduit par une diminution non linéaire de l'intensité mesurée et cause des artefacts lors de la reconstruction tomographique. C'est le phénomène de durcissement de faisceau (ou *beam hardening*), dont nous parlerons plus en détail dans le chapitre 3.

Cela implique également que l'énergie du rayonnement (inversement proportionnelle à sa fréquence) doit être adaptée à l'échantillon pour pouvoir obtenir un bon contraste sur l'image acquise. Si l'énergie est trop faible par rapport à la densité et à l'épaisseur de

l'objet, le rayonnement est entièrement absorbé par l'objet, et au contraire, s'il est trop énergétique, il traverse l'objet sans interagir avec : l'objet est alors transparent pour ce rayonnement et ne l'absorbe pas, comme l'air l'est pour les fréquences du rayonnement visible.

Les figures 1.4 et 1.5 montrent l'évolution des indices δ et β pour l'eau pure, qui illustre bien cet effet : à faible énergie, β est très élevé et la majorité du rayonnement est absorbé. À l'inverse pour les hautes énergies, β est presque nul, ce qui correspond à l'absence d'absorption. Le contraste d'absorption risque alors de ne pas être suffisant pour distinguer l'objet de l'air.

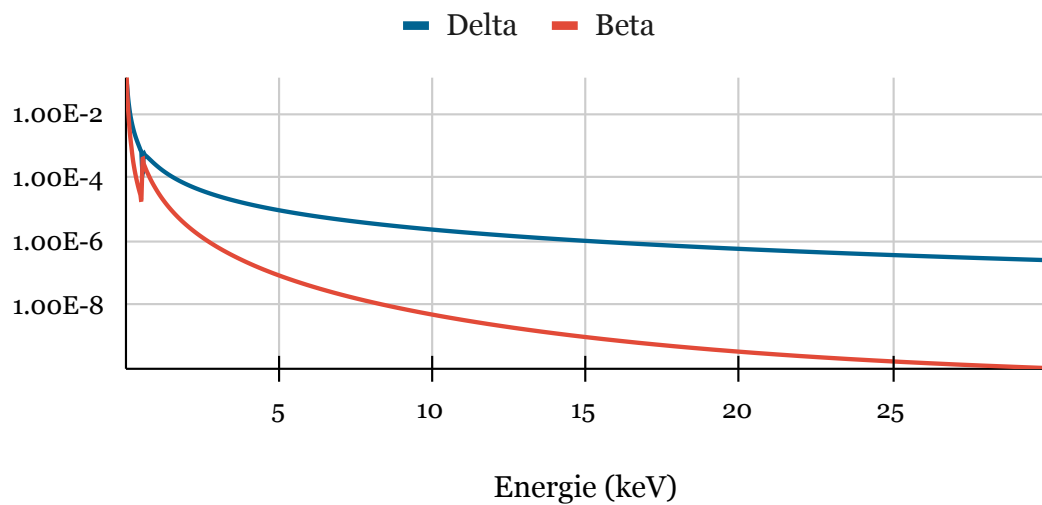


FIGURE 1.4 – Évolution des indices δ et β de l'eau par rapport à l'énergie des rayons X.

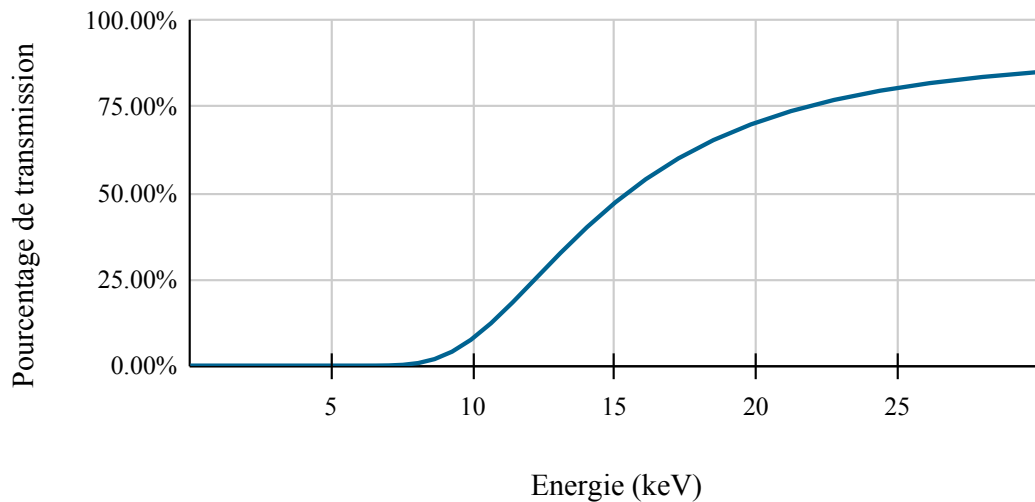


FIGURE 1.5 – Évolution du pourcentage de transmission de 1cm d'eau par rapport à l'énergie des rayons X.

1.1.3 Dangerosité des rayons X en imagerie médicale

Les rayons X sont un rayonnement ionisant, ce qui implique que les interactions physiques avec un milieu donnent lieu à des réactions chimiques allant jusqu'à la destruction de molécules. Dans des tissus vivants, cela peut causer des dommages cellulaires et causer des lésions. Si une faible quantité de ces lésions n'ont pas d'impact mesurable sur la santé, une exposition prolongée ou un rayonnement très énergétique peut causer des dommages importants en tuant un grand nombre de cellules, ainsi qu'une augmentation mesurable du risque de développement d'un cancer.

En imagerie médicale, l'énergie nécessaire pour obtenir un contraste correct est généralement de quelques dizaines de keV, l'énergie pouvant varier selon la zone du corps qui doit être imagée [2].

Un point très important à considérer dans ce cas-ci est la dangerosité de l'exposition aux rayons X, qui est directement liée à l'énergie et à la durée d'exposition.

Le rayonnement sera davantage absorbé par le corps si l'énergie est faible, mais à l'inverse si l'énergie est haute, le peu d'interactions qui auront lieu seront bien plus énergétiques, et donc potentiellement dangereuses. Généralement, la dose déposée en néanmoins plus faible à haute énergie qu'à basse énergie.

Cet effet est néanmoins délicat à mesurer, car la dangerosité pour un corps traversé par un rayonnement est complexe et dépendante des tissus qui sont touchés, la relation n'est donc pas directement linéaire avec l'absorption.

Pour mesurer la dangerosité d'une exposition à des rayons X, l'énergie du rayonnement émis et le temps d'exposition ne suffisent donc pas, il faut prendre davantage de paramètres en compte. L'émission d'une source à rayon X est mesurée en becquerels, qui représentent la quantité totale de rayonnement émis par celle-ci. Lorsque les rayons passent à travers un objet, celui-ci absorbe une partie de l'énergie, ce qui se mesure en Gray.

Pour ensuite estimer la dangerosité réelle pour un organisme, la mesure en Gray est pondérée par deux facteurs, l'un dépendant du type de rayonnement (W^R) et l'autre des propriétés des tissus imagés (W^T), représentés par des sieverts. Le tableau 1.6 représente les valeurs W^T pour quelques organes. Plus le W^T d'un organe est haute, plus celui-ci est sensible aux radiations. On peut remarquer que le sein, dont l'imagerie est le but du projet, fait partie des organes particulièrement sensibles, avec un W^T élevé.

| Organe | W^T | Organe | W^T |
|----------------|-------|------------------|-------|
| moelle osseuse | 0,12 | foie | 0,04 |
| côlon | 0,12 | œsophage | 0,04 |
| poumon | 0,12 | cerveau | 0,01 |
| estomac | 0,12 | peau | 0,01 |
| sein | 0,12 | surface des os | 0,01 |
| thyroïde | 0,04 | glande salivaire | 0,01 |
| vessie | 0,04 | | |

FIGURE 1.6 – Tableau des coefficients W^T pour quelques organes, issus des recommandations de la CIPR 103 [3]

1.2 Principe de la reconstruction tomographique

1.2.1 Principe de l'acquisition tomographie

La tomographie est une technique permettant d'obtenir des informations sur la structure interne d'un objet donné à partir d'acquisitions réalisées à l'extérieur de celui-ci. Le principe de base est de capturer suffisamment d'information depuis l'extérieur de l'objet pour pouvoir retrouver une information dans une dimension supérieure sur l'intérieur de l'objet. Par exemple, passer d'images 2D à une cartographie 3D. Comme l'acquisition consiste à capturer une information d'une dimension inférieure à l'objet imagé, on appelle une image capturée ainsi une projection. De nombreuses techniques et applications à la tomographie existent, mesurant différentes caractéristiques à l'aide d'ondes variées (onde électromagnétique, onde acoustique, onde sismique, etc.). Nous nous intéresserons ici uniquement à la tomographie par rayons X [4], très répandue notamment dans le cadre de l'imagerie médicale.

La tomographie par rayons X a été inventée en 1972 par G. N. Hounsfield, et se base sur l'imagerie d'absorption. Elle permet d'obtenir un volume 3D où chaque voxel (c.-à-d. le plus petit élément constituant le volume, dans le cas courant d'un détecteur pixélisé) représente la densité en ce point de l'objet. Un ensemble de radiographies (l'image obtenue par un système d'imagerie par rayon X) est pris à différentes positions soit de l'objet, soit de la source de rayons X couplée au détecteur. Il existe de nombreuses configurations de mouvement relatif entre le système d'imagerie et l'objet. La plus commune est la tomographie circulaire, où l'axe de rotation est fixe et orthogonal à la direction source-détecteur (figure 1.7). Lorsque la hauteur de l'objet imagé ne rentre pas dans le champ de capture, un mouvement supplémentaire le long de l'axe de rotation peut être ajouté. Si le mouvement est uniforme entre chaque angle d'acquisition, on parle alors de tomographie hélicoïdale, mais un mouvement plus complexe peut être réalisé selon des contraintes de l'échantillon et du système d'acquisition. Il est par exemple possible de combiner plusieurs axes de rotation pour réaliser une tomographie sphérique [5].

Une fois une série d'images capturée, la reconstruction du volume imagé consiste à inverser numériquement le processus d'acquisition, afin de retrouver la densité en chaque

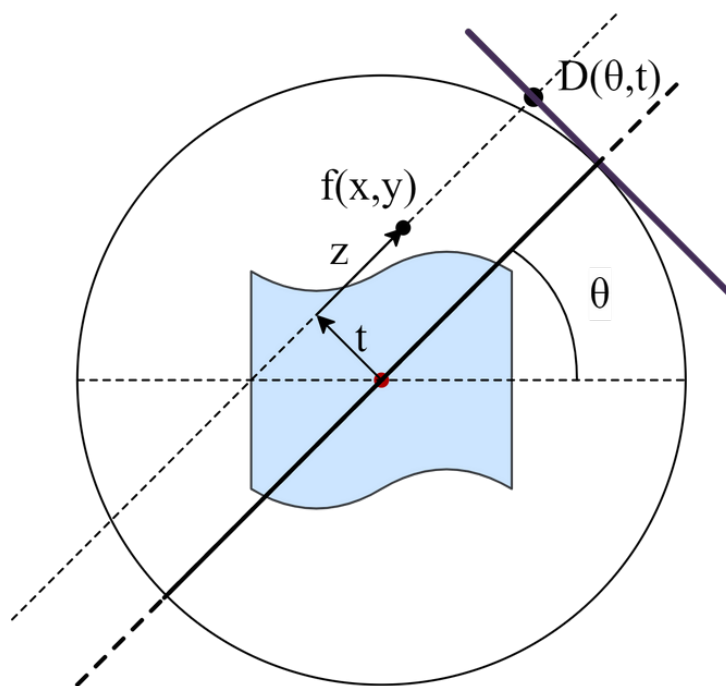


FIGURE 1.7 – Illustration de la transformée de Radon. Chaque point t sur le détecteur D à l'angle θ représente la somme de la grandeur mesurée de l'objet f en chaque point le long d'une ligne dépendante de la géométrie d'acquisition.

point de l'objet qui a été intégré par l'acquisition.

1.2.2 Inversion de la tomographie

Cette méthode se base sur le théorème de la projection de Radon [6]. Ce théorème indique qu'il est possible de reconstruire un objet de dimension N avec une infinité de projections distinctes de celui-ci à une dimension $N-1$. Dans le cas de la tomographie médicale, cela se traduit généralement par la reconstruction d'un volume 3D à partir de projections 2D, ou d'une coupe 2D à partir d'acquisitions 1D. Une représentation discrète du théorème de Radon est détaillée dans [7]

La transformée de Radon se décrit de la manière suivante en 2D :

$$D_{\theta}(t) = \int_{-\infty}^{\infty} f(-z \sin \theta + t \cos \theta, z \cos \theta + t \sin \theta) dz \quad (1.10)$$

Chaque point d'acquisition $D_{\theta}(t)$ est décrit par l'angle d'acquisition θ et la distance t du point de mesure par rapport à l'axe d'acquisition, $f(x, y)$ étant la valeur mesurée dans l'objet à la coordonnée (x, y) . La valeur enregistrée à ce point est alors égale à la somme de tous les points sur la droite de projection passant par $D_{\theta}(t)$. Ces points peuvent être caractérisés par la somme de deux vecteurs, $z \begin{pmatrix} -\sin \theta \\ \cos \theta \end{pmatrix}$ qui décrit la position sur la droite et $t \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix}$ qui décrit la position de la droite.

Cette formule représente le processus d'acquisition, il faut donc l'inverser pour retrouver l'objet imagé. On peut utiliser la formule de l'intégrale pour obtenir la transformée de Radon inverse (1.11), qui définit la valeur d'un point de l'objet à partir de l'intégralité des projections autour de celui-ci.

$$f(x, y) = \frac{1}{2\pi} \int_0^{2\pi} D_\theta(x \cos \theta + y \sin \theta) d\theta \quad (1.11)$$

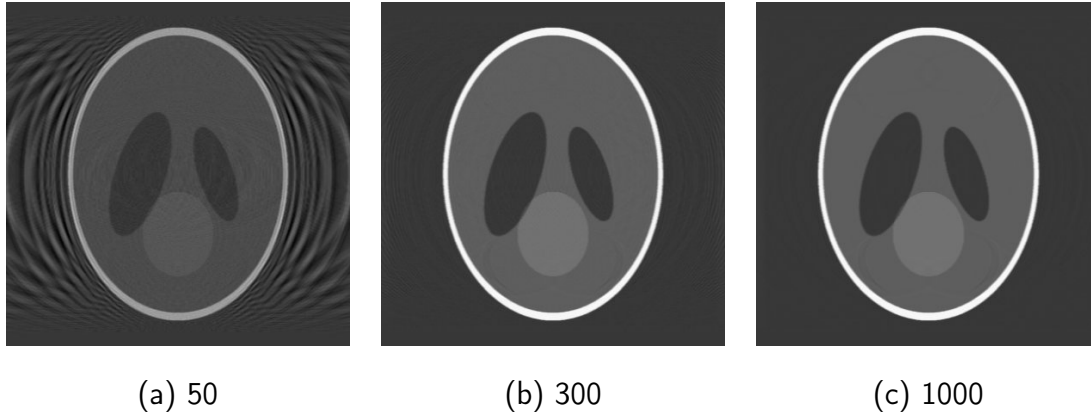


FIGURE 1.8 – Exemple de reconstruction tomographique par rétroprojection filtrée pour 50, 300 et 1000 angles d’acquisition. Le fantôme présenté est celui de Shepp-Logan [8].

Néanmoins, cette inversion directe pose un problème pour une acquisition réelle : pour qu’elle fonctionne correctement, il faudrait une infinité d’angles d’acquisition. On peut voir en figure 1.8 différentes reconstructions avec un nombre d’angles de projection croissants. Le nombre d’angles nécessaire pour obtenir une reconstruction sans artefact et le nombre à partir duquel la précision de la reconstruction n’est plus améliorée dépend de plusieurs facteurs tels que la définition des radiographies, celle du volume à reconstruire, ainsi que les types de structures présentes et la taille de celles-ci sur le détecteur [9], [10].

L’inversion de la transformée de Radon peut également être obtenue à l’aide du théorème de la projection de la tranche centrale [11] et de la transformée de Fourier. Ce théorème permet de relier la transformée de Fourier de la projection à celle de la reconstruction. La transformée de Fourier d’une projection $D\theta(t)$ se formule de la manière suivante :

$$F_t[D_\theta(t)] = \int_{-\infty}^{\infty} D_\theta(t) e^{-2\pi i vt} dt \quad (1.12)$$

En remplaçant $D\theta(t)$ par l’équation (1.10), on obtient alors cette formule :

$$F_t[D_\theta(t)] = \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(-z \sin \theta + t \cos \theta, z \cos \theta + t \sin \theta) dz \right) e^{-2\pi i vt} dt \quad (1.13)$$

Qui peut être alors considéré comme la transformée de Fourier de la reconstruction pour l’angle θ . En changeant l’espace de coordonnées polaires à cartésien, on obtient :

$$F_t[D_\theta(t)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i v(x \cos \theta + y \sin \theta)} dx dy \quad (1.14)$$

Ce qui nous donne la formule suivante, montrant que la tranche centrale de la transformée de Fourier d’un tomogrammes est égale à la Transformée de Fourier de la projection à un angle donné :

$$F_t[D_\theta(t)] = F_{x,y}[f(x, y)](x \cos \theta + y \sin(\theta)) \quad (1.15)$$

Cette méthode de calcul de la reconstruction tomographique est très utilisée du fait de sa rapidité d'exécution : la transformée de Fourier est une opération utilisée dans un très grand nombre de domaines et qui profite d'une très grande optimisation des implémentations existante. Néanmoins, comme cette formule n'est exacte que pour un nombre infini d'angles d'acquisition, ainsi le processus de reconstruction fait à partir de quelques centaines de projections comme c'est généralement le cas en situation réelle cause l'apparition d'artefacts dus au sous-échantillonnage. Il faut également adapter la formule pour prendre en compte les géométries non parallèles [12].

Une amélioration de cette technique, la rétropropagation filtrée (*Filtered Back-Projection, FBP*) permet de réduire ces effets en appliquant un filtre aux projections avant la rétropropagation. Le filtre utilisé varie selon la méthode exacte, le cas d'utilisation, etc., mais prend généralement la forme d'un filtre de dé-floutage. L'algorithme de la FDK (pour Feldkamp-Davis-Kress) est une variation de la FBP permettant de prendre en compte des géométries coniques [12].

Lorsque la quantité et la qualité des données ne sont pas idéales (peu d'angles d'acquisition, image très bruitée, alignement et géométrie complexe), ces méthodes laissent généralement leur place aux méthodes itératives, plus coûteuses, mais plus robustes.

1.2.3 Principe de la reconstruction itérative

La reconstruction tomographique itérative a été conçue pour obtenir une meilleure reconstruction qu'une reconstruction directe pour un même jeu de données. En maximisant l'utilisation de l'information présente sur les projections acquises, on peut donc soit augmenter la précision du volume en conservant les mêmes conditions d'acquisition, soit conserver une précision similaire tout en réduisant le temps d'exposition lors de l'acquisition.

L'idée principale de la reconstruction itérative est qu'après une rétroprojection, la projection du volume obtenue à un des angles d'acquisition nous donne une image différente de celle acquise, nous informant que la reconstruction n'est pas idéale. On peut donc calculer l'erreur entre le volume et les projections et tenter de corriger les valeurs du volume pour qu'il corresponde mieux aux images acquises.

Cet algorithme demande donc l'implémentation d'un opérateur de projection (*forward projection*) en plus de la rétroprojection. Ces deux opérateurs ont une complexité de calcul égale, leurs temps de calcul sont donc similaires en dehors des différences d'implémentations. Cette méthode de reconstruction est compatible avec n'importe quelle géométrie d'acquisition, tant que le projecteur utilisé est capable de représenter la géométrie visée.

Cet opérateur peut être représenté par la formule suivante :

$$P(V, \alpha)(i) = \sum_x V(x)W^\alpha(i, x) \quad (1.16)$$

Avec $V(x)$ la valeur du volume V pour l'élément x , α l'angle de projection, $P(V, \alpha)$ la projection du volume V à l'angle α , i l'indice du pixel de la projection, et $W^\alpha(i, x)$ le poids de l'élément x projeté sur i à l'angle α

La matrice W^α est une matrice essentiellement remplie de zéros, qui peut être précalculée ou bien recalculée à chaque projection selon les contraintes de mémoire.

Le processus de la reconstruction tomographique itérative se déroule de la façon suivante :

1. Une première estimation du volume est calculée à l'aide d'une rétroprojection (ou autre).
2. Ce volume estimé est projeté aux mêmes angles que les projections réelles.
3. Les projections calculées sont comparées aux projections acquises pour calculer une erreur en chaque pixel.
4. Une rétroprojection visant à corriger les valeurs des voxels est appliquée de manière à réduire l'erreur calculée en 3.
5. On recommence à partir de l'étape 2 jusqu'à ce que l'erreur obtenue à l'étape 3 soit inférieure à un seuil déterminé.

Le premier algorithme de reconstruction tomographique itératif à avoir été développé est l'*Algebraic reconstruction technique*, ART (équation (1.17) [13], [14]).

$$V^{k+1}(x) = V^k(x) - \lambda^k \sum^i \left(\frac{P(V, \alpha)(i) - I^\alpha(i)}{\sum^x W_{i,x}^\alpha} \right) \quad (1.17)$$

Avec $V^k(x)$ une valeur du volume à l'itération k , α un des angles d'acquisition et λ un paramètre de relaxation pouvant être égal ou inférieur à 1.

Cette technique met à jour le volume pour chaque angle d'acquisition et chaque pixel séquentiellement : pour chaque angle, le volume est projeté, comparé, et l'erreur est rétroprojetée.

Le *Simultaneous Iterative reconstruction technique*, SIRT (équation (1.18)), fonctionne sur le même principe, mais cette fois-ci tous les pixels d'une projection sont utilisés pour mettre à jour le volume. Les projections sont donc prises séquentiellement et le volume entier est mis à jour pour chacune d'entre elles.

$$V^{k+1}(x) = V^k(x) - \lambda^k \sum^i \left(\frac{P(V, \alpha)(i) - I^\alpha(i)}{\sum^x W_{i,x}^\alpha} \right) \quad (1.18)$$

Une amélioration de cette technique, le *Simultaneous Algebraic Reconstruction Technique*, SART (équation (1.19) [15], [16]), propose non plus de traiter chaque angle séparément, mais de calculer l'erreur moyenne de tous les angles pour chaque voxel. Cette technique est généralement considérée comme supérieure à la version non simultanée, offrant une convergence plus rapide et une meilleure résistance au bruit.

$$V^{k+1}(x) = V^k(x) - \lambda^k \sum^\alpha \sum^i \left(\frac{P(V, \alpha)(i) - I^\alpha(i)}{\sum^x W_{i,x}^\alpha} \right) \quad (1.19)$$

Ces méthodes conservent la même opération pour le calcul d'erreur, mais d'autres variantes existent. Par exemple, le *Multiplicative Algebraic Reconstruction Technique*, MART (équation (1.20) [17]), calcule l'erreur et met à jour le volume de manière multiplicative et non plus additive. Si la convergence est généralement plus rapide et que cet algorithme est

moins sensible au bruit [18], [19], il apporte néanmoins la contrainte de pouvoir traiter des nombres positifs exclusivement.

$$V^{k+1}(x) = V^k(x) + \lambda \sum^i \left(\frac{I^\alpha(i)}{P(V, \alpha)(i)} \right) \quad (1.20)$$

Similairement au SART, le SMART est une amélioration du MART qui propose de moyenniser l'erreur à chaque angle (équation (1.21)). Il est possible d'utiliser une moyenne géométrique, offrant des caractéristiques de convergence différente (équation (1.22)).

$$V^{k+1}(x) = V^k(x) * \sum^\alpha \left(\sum^i \left(\frac{I^\alpha(i)}{P(V, \alpha)(i)} \right)^{\lambda^k} \right) * \frac{1}{\sum^{i,x} W_{i,x}^\alpha} \quad (1.21)$$

$$V^{k+1}(x) = V^k(x) * \prod^\alpha \left(\sum^i \left(\frac{I^\alpha(i)}{P(V, \alpha)(i)} \right)^{\lambda^k} \right)^{\frac{1}{\sum^{i,x} W_{i,x}^\alpha}} \quad (1.22)$$

Compte tenu de l'imperfection des images acquises, due au bruit de mesure et aux imprécisions de l'alignement, des méthodes statistiques intégrant une connaissance *a priori* sur la modélisation de l'image permettent dans certains cas d'améliorer le résultat des méthodes algébriques, qui supposent des images parfaites.

Par exemple, la méthode MLEM (Maximum-Likelihood Expectation-Maximization) [20], basé sur le maximum de vraisemblance, permet de prendre en compte lors du calcul de l'erreur la distribution de Poisson du bruit sur les projections. Il introduit p la probabilité de la mesure par le détecteur pour calculer l'erreur. Cette méthode est principalement pensée pour la tomographie d'émission et nécessite de renseigner à l'algorithme les statistiques de distribution du système d'imagerie.

Une amélioration du MLEM est le OSEM (Ordered-Subset Expectation-Maximization), qui consiste à considérer des sous-itérations comprenant une partie seulement des projections [21]. L'ordre et le nombre de projections par sous-itération doivent être choisis de manière à optimiser la convergence et dépendent du jeu de données utilisé. Cette méthode de découpage des itérations peut également être étendue au SIRT et au SMART très simplement. Si le nombre de sous-itérations est bien choisi, ce découpage permet d'accélérer la convergence de la reconstruction.

De nombreuses autres méthodes de reconstructions algébriques et statistiques existent (Total variation [22], Optimisation convexe [23], ...). Les méthodes statistiques imposent néanmoins de connaître la statistique du bruit présent sur les projections utilisées. Si c'est tout à fait possible avec des images acquises avec un détecteur connu, cela devient plus difficile lorsque des étapes de prétraitement complexes sont appliquées sur les images brutes. Dans notre cas, les étapes de traitements et d'extraction de la phase [24], [25] changent radicalement la nature de l'image reconstruite et les caractéristiques du bruit d'acquisition, car le signal est reconstruit. C'est notamment avec l'imagerie par masque de Hartmann (discuté dans le chapitre suivant), car les valeurs utilisées sont la position de faisceaux sur le détecteur et non l'intensité mesurée par celui-ci.

Toutes ces méthodes fonctionnent en règle générale sur un schéma commun, les variations se font au niveau de l'ordre des opérations, du calcul de l'erreur et de la méthode

de mise à jour du volume. Le tableau 1.9 montre les différentes combinaisons de ces paramètres pour quelques algorithmes.

| | Ordre d'opération | Calcul de l'erreur | Mise à jour |
|-------|--------------------|--------------------|----------------|
| ART | Rayon-par-Rayon | Additive | Additive |
| SIRT | Angle-par-angle | Additive | Additive |
| SART | Données complètes | Additive | Additive |
| MART | Angle-par-angle | Multiplicative | Multiplicative |
| SMART | Données complètes | Multiplicative | Multiplicative |
| MLEM | Données complètes | Statistique | Multiplicative |
| OSEM | Données partielles | Statistique | Multiplicative |
| TV | Données complètes | Statistique | Multiplicative |

FIGURE 1.9 – Tableau des différences entre quelques méthodes de reconstruction tomographiques.

Il est tout à fait possible de combiner certaines méthodes, un cas notable étant l'utilisation des sous-itérations du OSEM qui peut être utilisé avec le SART ou le SMART et qui délivre les mêmes avantages.

Compte tenu des contraintes liées à l'application du projet XPulse (imagerie du sein, un organe particulièrement radiosensible), l'algorithme de reconstruction doit être capable de reconstruire au mieux avec aussi peu de données que possible (peu d'angles et temps d'exposition réduit). Les algorithmes non itératifs offrent des performances limitées dans ce cas-ci, aussi je me référerai pour la suite presque exclusivement aux méthodes itératives lorsque je mentionnerai la reconstruction tomographique. Dans le cadre du projet XPulse et de ma thèse, les méthodes utilisées sont le SART et le SMART.

1.3 Conclusion

L'imagerie par rayons X permet d'obtenir des informations sur la composition interne d'un objet de manière non invasive et est de ce fait une technique d'imagerie privilégiée dans de nombreux domaines, dont l'imagerie médicale dans le cadre de ce projet. Couplé à la reconstruction tomographique, il est même possible de reproduire une carte 3D d'une caractéristique de cet objet, généralement sa densité, pour pouvoir visualiser de manière précise les informations locales de celui-ci.

L'étape de reconstruction tomographique par ordinateur est cruciale pour obtenir le résultat souhaité. Le projet XPulse dans lequel s'inscrit ma thèse repose sur cette technique pour réaliser une mammographie 3D et fait donc face aux contraintes d'exposition aux rayons X inhérentes à l'imagerie X, exacerbées par l'acquisition tomographique.

Le but final du projet étant la réalisation d'un volume de valeurs liées à la phase acquis à l'aide d'une géométrie complexe, nous devons être capables de maîtriser le programme

de reconstruction pour pouvoir y intégrer des algorithmes de traitements de phase et de prendre en compte les particularités géométriques de nos installations.

L'implémentation de ce programme de reconstruction fait l'objet du chapitre suivant.

Chapitre 2

Imagerie de phase par rayons X

Dans la première partie de cette thèse, nous avons expliqué le principe de l'imagerie par rayons X classique par absorption, et comment il est possible de réaliser une tomographie pour obtenir une carte 3D de la densité d'un échantillon. On considère ici la densité comme étant la densité électronique, qui est proportionnelle à la masse volumique du matériau.

Cette valeur d'absorption nous donne une information sur la densité de l'objet, mais elle ne permet pas à elle seule de déterminer la composition chimique des matériaux étudiés. En effet, différents matériaux peuvent avoir un niveau d'absorption similaire pour l'énergie de rayons X utilisée.

Une méthode pour obtenir une source d'information supplémentaire est d'imager le retard de phase du rayonnement, en plus de son absorption. Cette valeur étant décorrélée de la densité de l'objet, elle offre une information totalement nouvelle et peut permettre de discerner deux matériaux d'absorption proche. Il est également possible dans certains cas, à l'aide de l'absorption et du décalage de phase, de caractériser précisément la composition chimique des matériaux à la suite d'une reconstruction tomographique.

Dans le cadre du projet XPulse, l'imagerie du sein présente un défi du fait de la densité proche de différents tissus mous, qui offrent un contraste en absorption faible. L'imagerie de phase a été choisie pour sa haute sensibilité lors de l'imagerie des tissus organiques, permettant notamment de différencier plus aisément les tissus sains des tissus cancéreux tout en permettant une exposition aux rayons X moindre. Il est alors également possible d'obtenir la composition chimique des microcalcifications, ce qui permet de mieux différencier celles présentant un risque (malignes) des autres (bénignes) [26]-[28].

Dans ce chapitre, nous parlerons des caractéristiques et des différentes méthodes de l'imagerie de phase.

2.1 Principe de l'imagerie de phase

2.1.1 La phase d'une onde électromagnétique

Comme présenté dans le chapitre 1, un rayonnement électromagnétique est caractérisé par sa longueur d'onde, son amplitude et sa phase.

La phase est une valeur plus difficile à mesurer que l'amplitude. C'est une propriété cyclique de l'onde, qui peut être décrite par un décalage dans ce cycle, appelé le déphasage. On ne mesure donc pas une valeur absolue, mais plutôt un décalage par rapport à une référence.

À partir de l'indice de réfraction, le décalage de phase induit par un matériau peut se calculer à l'aide du coefficient δ uniquement, pour un rayonnement de longueur d'onde λ donnée. C'est donc ce coefficient δ qui est mesuré plus ou moins directement avec les techniques d'imagerie de phase. Celui-ci est en réalité la partie réelle de l'indice de réfraction, ce qui implique que le décalage de phase est proportionnel à la réfraction induite par le changement de milieu. Cette relation ouvre la voie à une grande partie des méthodes d'imagerie de phase, se reposant sur la mesure de l'angle de réfraction pour retrouver le décalage de phase.

Ce coefficient étant largement supérieur à β pour les tissus mous, il offre une meilleure sensibilité et les rend donc plus facilement différenciables en les imageant de cette manière. Cette technique d'imagerie se développe donc rapidement dans le milieu médical et le domaine de recherche est très actif.

2.1.2 L'imagerie par contraste de phase

La contraste de phase est une technique d'imagerie innovante très étudiée dans le domaine médical pour ses avantages dans l'étude des tissus mous du corps humain. Cette technique a principalement été développée sur synchrotron du fait de ses caractéristiques propices au contraste de phase (rayons parallèles, haute brillance et potentiellement monochromatique). Plus récemment les sources et détecteurs de nouvelle génération commencent à faciliter l'accès à ce domaine, en permettant d'utiliser le contraste de phase en laboratoire.

Le contraste de phase donne accès au déphasage sans mesurer directement cette valeur. En faisant interagir le front d'onde initial avec l'objet, le changement de phase est enregistré comme un changement d'amplitude, pouvant alors être mesuré par un détecteur classique. Comme les capteurs ne sont sensibles qu'à l'amplitude, le contraste de phase consiste donc à transformer le déphasage en modulation d'amplitude selon différentes techniques pour pouvoir être ensuite capable de reconstruire la phase du front d'onde à l'aide d'algorithmes spécifiques.

C'est néanmoins une technique délicate à mettre en place, imposant certaines contraintes sur la source et le système d'imagerie. Pour pouvoir capturer ce déphasage, la plupart des techniques nécessitent que la source de rayons X soit cohérente, c'est-à-dire que les rayons émis par la source ont tous une même phase initiale. Un front d'onde cohérent permet alors de faire interagir le front d'onde avec des optiques, ou avec lui-même, de manière à le rendre mesurable.

C'est la raison principale qui fait que ce type d'imagerie était jusqu'à récemment principalement réservé aux études sur synchrotron, qui délivre un faisceau de rayons X cohérents et une haute brillance. Ce n'est pas le cas des sources normalement utilisées dans le domaine médical, les tubes à rayons X. La recherche en imagerie de phase par rayons X a donc longtemps été faite majoritairement sur des sources de type synchrotron, ce qui rend son étude et son utilisation dans le domaine médical ou en laboratoire bien

plus difficile.

Plus récemment, le développement et la commercialisation de sources à microfocus ont permis le développement de techniques d'imagerie de phase en laboratoire. Ces sources offrent une cohérence (par un filtrage) et une brillance suffisantes pour le contraste de phase, permettant d'accroître grandement la faisabilité de l'imagerie de phase à but clinique [29], et d'élargir les possibilités de recherche dans le domaine.

Pour obtenir cette quasi-cohérence avec une source d'encombrement similaire à celles utilisées classiquement dans le domaine médical, au lieu d'utiliser un accélérateur donnant des rayons parallèles comme le synchrotron, la solution est d'avoir une source ponctuelle, ou tout du moins la plus petite possible (un spot d'émission de quelques dizaines de micromètres) permettant de se rapprocher des caractéristiques d'une source cohérente.

Les principales différences entre le synchrotron et les sources de laboratoires sont la géométrie du faisceau (parallèle pour le synchrotron, conique pour les sources de laboratoire), et le spectre : le rayonnement du synchrotron est polychromatique, mais le faisceau est tellement puissant qu'il est possible de n'utiliser qu'une seule longueur d'onde à l'aide d'un cristal monochromateur, tout en gardant suffisamment de flux (photons par seconde). Les sources de laboratoires n'offrent généralement pas cette possibilité et le rayonnement utilisé est donc polychromatique.

Il y a deux grandes familles de techniques de contraste de phase. Il est par exemple possible de détecter ce déphasage à l'aide des interférences qui se forment lors de l'interaction entre les rayons déphasés et non déphasés : ce sont les techniques d'interférométrie. La deuxième famille repose sur le lien entre déphasage et réfraction, qui permet de retrouver une quantité de déphasage des rayons à partir de la réfraction induite par l'objet.

Dans la partie suivante, nous verrons les différentes techniques existantes pour mesurer le décalage de phase en imagerie par rayons X.

2.2 Méthode d'imagerie de phase

2.2.1 Interférométrie cristalline

Cette méthode est la première méthode d'imagerie de phase mise en œuvre, introduite en 1965 par Bonse et Hart [30].

Elle consiste à séparer le faisceau trois fois à l'aide de cristaux réfractant l'onde incidente, comme présenté en figure 2.1. Le premier cristal permet de diviser le faisceau en deux, en préservant le front d'onde. Le second fait reconverger les faisceaux précédemment divisés pour qu'ils entrent finalement en interférence sur le troisième disque. En plaçant l'échantillon entre le deuxième et le troisième disque, soit uniquement dans l'un des deux faisceaux, les interférences formées sur le troisième disque sont alors directement dues au changement de phase du faisceau qui a traversé l'échantillon. Cette technique nécessite une source parallèle, et comme la majorité du flux de photons est perdu, elle nécessite soit un flux initial très élevé, soit un temps d'exposition très long. Elle est donc utilisée aujourd'hui principalement sur synchrotron.

Le très faible rendement de cette technique (une grande partie des rayons X émis ne seront pas enregistrés par le détecteur), associé à sa complexité de mise en œuvre, la rend

difficilement applicable à l'imagerie médicale.

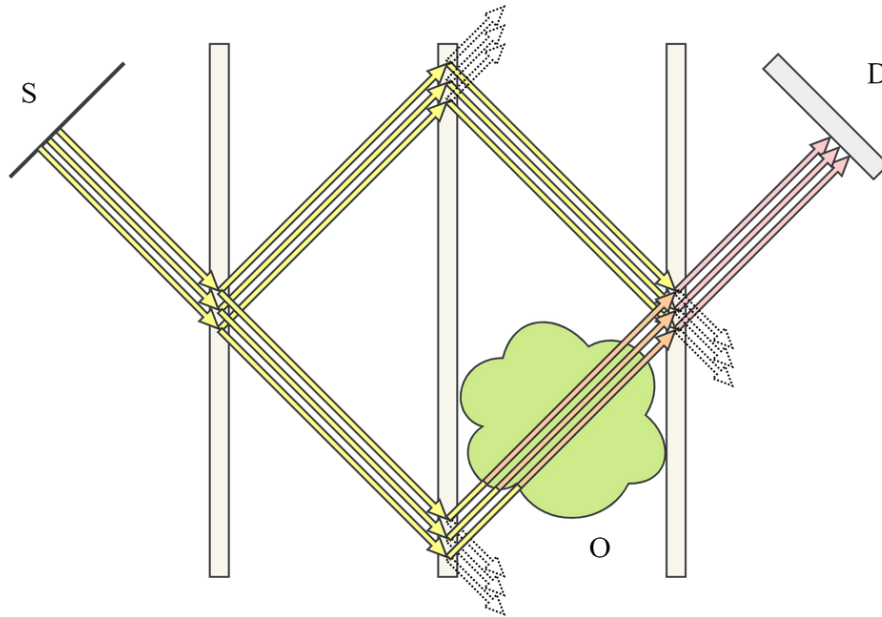


FIGURE 2.1 – Schéma de fonctionnement d'un système d'imagerie par interférométrie. Le rayonnement issu de la source (S) passe à travers une succession de cristaux de manière que seule une partie traverse l'objet (O). Le détecteur (D) mesure alors le rayonnement résultant de l'interaction entre le front d'onde perturbé par l'objet et le front d'onde non perturbé.

2.2.2 Imagerie par analyseur

L'imagerie par analyseur a été développée dans les années 1990 [31]. Cette technique consiste à mesurer la réfraction des rayons causée par l'objet imagé à l'aide d'un cristal analyseur. L'analyseur ne réfléchit les rayons incidents que sur une très faible variation d'angle d'incidence. Pour capturer une image complète, il faut donc placer le détecteur de manière à capturer les rayons réfléchis à différents angles de manière répétée (Figure 2.2). Avec suffisamment d'angles de capture, on peut alors réaliser une cartographie complète de la déflexion des rayons.

Cette technique nécessite un faisceau parallèle et monochromatique, et une exposition aux rayons X très élevée du fait qu'une grande partie des rayons passant par l'objet imagé ne seront pas réfléchis vers le détecteur et qu'une image nécessite plusieurs acquisitions. L'alignement du système est également très sensible.

2.2.3 Imagerie par illumination de bord et par masque de Hartmann

Plusieurs techniques d'imagerie de phase utilisent le même principe de formation du flux incident par un masque pour former une grille de sous-faisceaux. Le principe de base est que chacun de ces sous-faisceaux peut être réfracté par l'objet. En mesurant le déplacement sur le détecteur de chaque sous-faisceau entre une image sans objet et une image avec, il est possible de recréer une carte de déflexion dont la résolution dépend de la

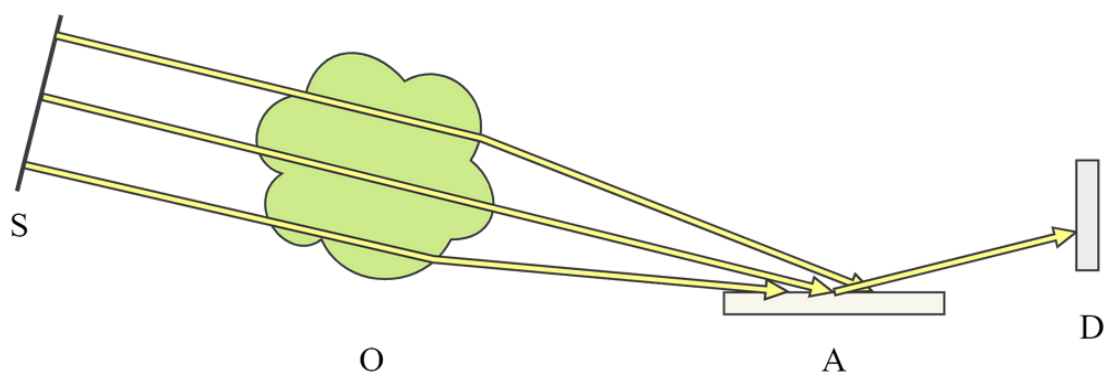


FIGURE 2.2 – Schéma de fonctionnement d'un système d'imagerie par analyseur. Le miroir (A) est incliné à différents angles pour prendre une série de mesures. Le détecteur (D) enregistre le flux réfléchi par le miroir.

grille ou des grilles utilisées. Cette technique a l'avantage de pouvoir être réalisée avec une source microfocus et ne nécessite pas de faisceau parallèle, monochromatique ou cohérent.

L'imagerie par masque de Hartmann est relativement simple. Elle est réalisable avec une seule grille, placée avant ou après l'objet à imager. La définition de l'image est dictée par la densité des trous de la grille, mais également la précision avec laquelle les sous-faisceaux sont échantillonnés (quantité de photon, taille et sensibilité des pixels). Cela peut nécessiter un alignement précis si la taille et l'espacement des spots sont d'une taille proche de celle d'un pixel du détecteur (figure 2.3).

L'imagerie par illumination de bord consiste à ajouter une grille au niveau du détecteur, qui n'expose que la partie centrale des pixels. Avec des sous-faisceaux de la taille des trous de la grille, il est alors possible de relier la quantité de déflexion qu'a subi un faisceau à l'inverse de l'intensité mesurée sur le pixel : un faisceau centré illuminera toute la partie exposée du pixel, alors que s'il est défléchi une partie de son énergie sera absorbée par la grille (figure 2.4).

2.2.4 Imagerie par *speckle*

Cette technique d'imagerie consiste à introduire une membrane avec un motif connu entre la source et le détecteur pour en étudier les modifications dues à l'échantillon, en suivant la modification du motif [32] (figure 2.5). Le motif de la membrane introduit des changements d'amplitude à haute fréquence dans l'image mesurée. Avec une image de référence, on peut repérer des points reconnaissables dans la structure sur les deux images qui seront légèrement décalés si les rayons ont été déviés par l'objet. On peut également prendre plusieurs acquisitions chacune avec l'objet, mais avec des distances différentes, pour obtenir un effet similaire.

Cette méthode est relativement facile à mettre en place et ne nécessite pas de source parallèle et monochromatique.

2.2.5 Imagerie par propagation libre

La propagation libre (PBI, *propagation-based imaging* ou CPPL, *contraste de phase par propagation libre*) est la technique la plus simple à mettre en place pour obtenir une

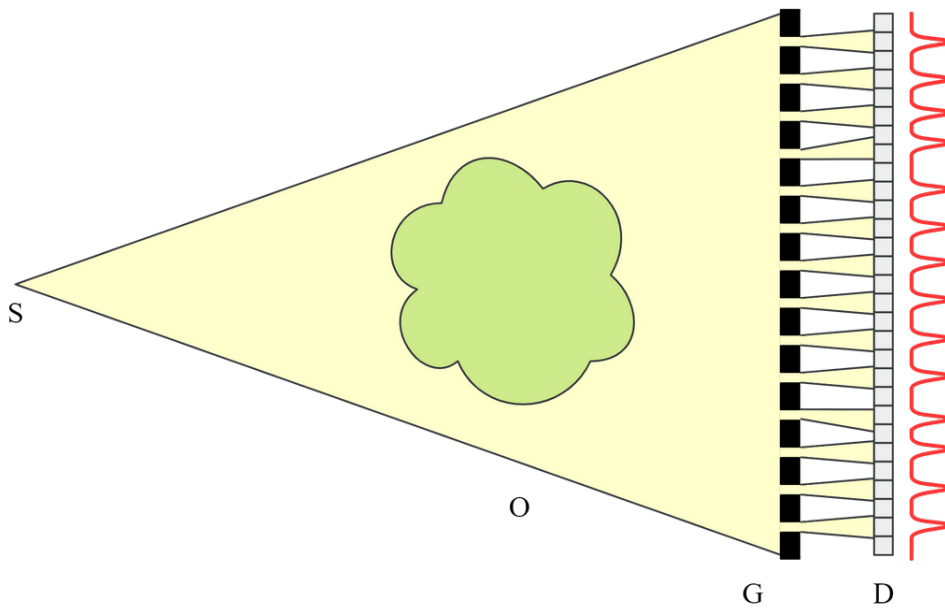


FIGURE 2.3 – Schéma de fonctionnement d'un système d'imagerie par masque de Hartmann. Une grille (G) forme le front d'onde en une grille de sous-faisceaux qui peuvent être résolus individuellement par le détecteur. L'objet cause une déflexion des faisceaux.

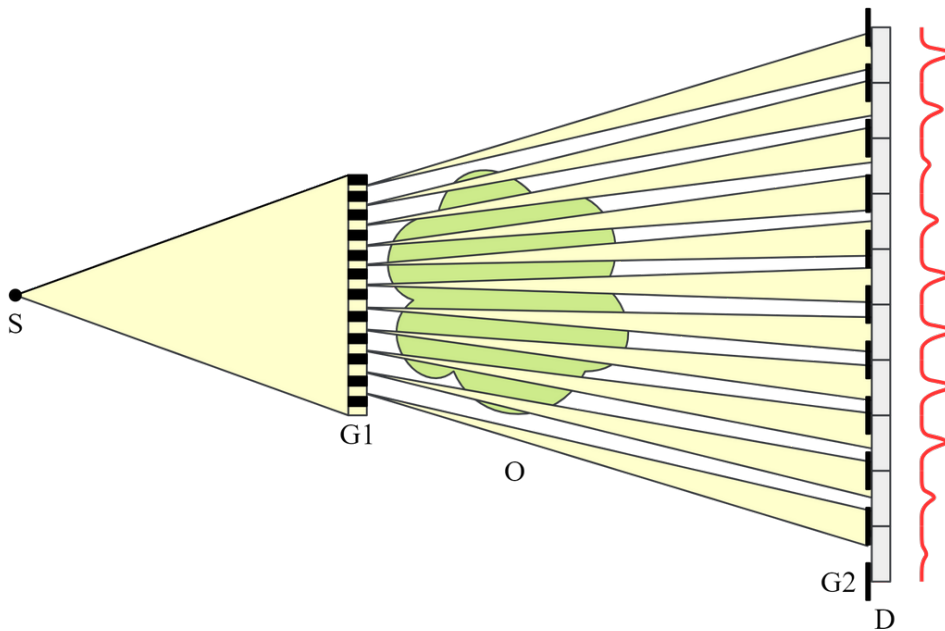


FIGURE 2.4 – Schéma d'un système d'imagerie par illumination de bords. Les faisceaux formés par la grille G1 se projettent sur des trous de la grille G2. L'objet cause la réfraction des faisceaux, et ceux-ci sont déviés des trous dans la grille G2. Le pixel correspondant enregistrera donc une valeur plus faible.

information sur la phase. Comme les rayons sont légèrement déviés localement lors de leur passage dans l'échantillon, en se plaçant à une distance suffisante de celui-ci (dépendant des conditions expérimentales), des interférences entre les rayons déviés se forment, donnant un effet de rehaussement des contours sur l'image capturée (figure 2.6).

On obtient une image mêlant l'amplitude et le laplacien de la phase. Elle peut être

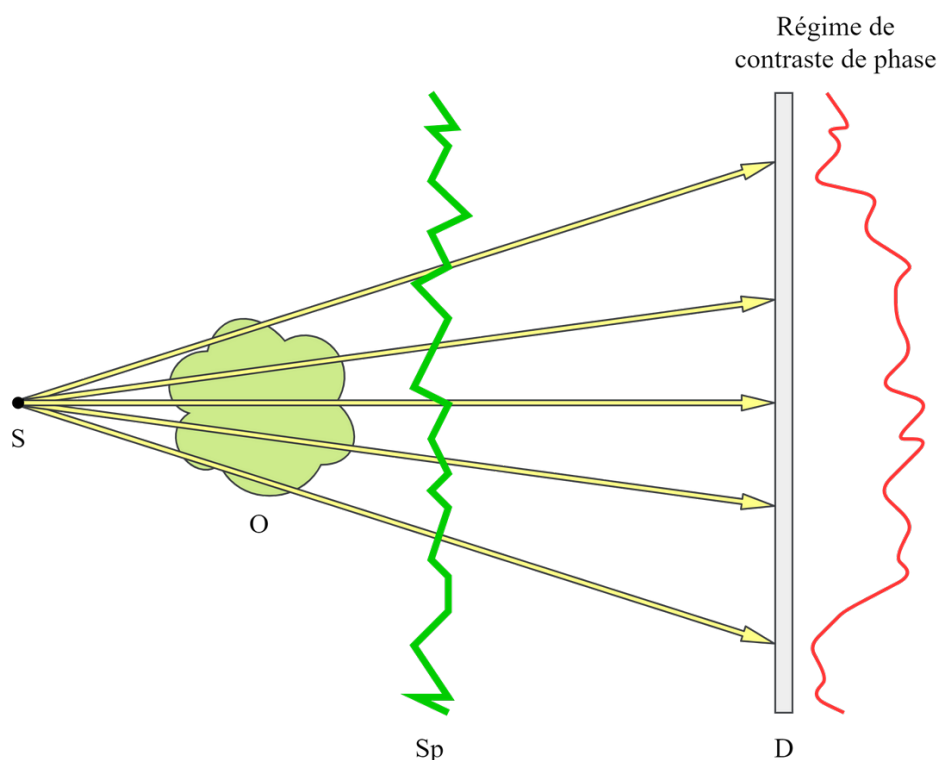


FIGURE 2.5 – Schéma d'un système d'imagerie par *speckle*. Le filtre Sp encode un motif dans le front d'onde qui est analysé afin de retrouver la déformation qu'il a subie. Le filtre peut être placé avant ou après l'objet.

utilisée telle quelle pour un diagnostic, aidé par le rehaussement de contours, mais on peut aussi extraire l'amplitude et la phase pour une reconstruction tomographique.

Cette méthode a l'avantage de ne nécessiter aucune optique supplémentaire et d'être plus simple à mettre en place que celles précédemment présentées [33], [34]. La distance de propagation peut néanmoins poser problème, car pour obtenir un effet de rehaussement de contours visible, plus la longueur d'onde est petite ou plus les pixels du détecteur sont grands, plus la distance de propagation doit être grande. Dans les gammes d'énergie utilisée en imagerie médicale, la distance doit déjà être de plusieurs mètres avec une taille de pixel de l'ordre de $100\ \mu\text{m}$. Le rayonnement doit être relativement cohérent et monochromatique.

Plusieurs configurations d'imagerie existent, dépendant de la distance de propagation (du plus proche au plus lointain : régime de Fresnel, holographie, régime de diffraction).

De nombreux algorithmes existent pour traiter les données acquises en PBI et en extraire le décalage de phase, tel que la formule de Paganin que je détaille dans le chapitre suivant.

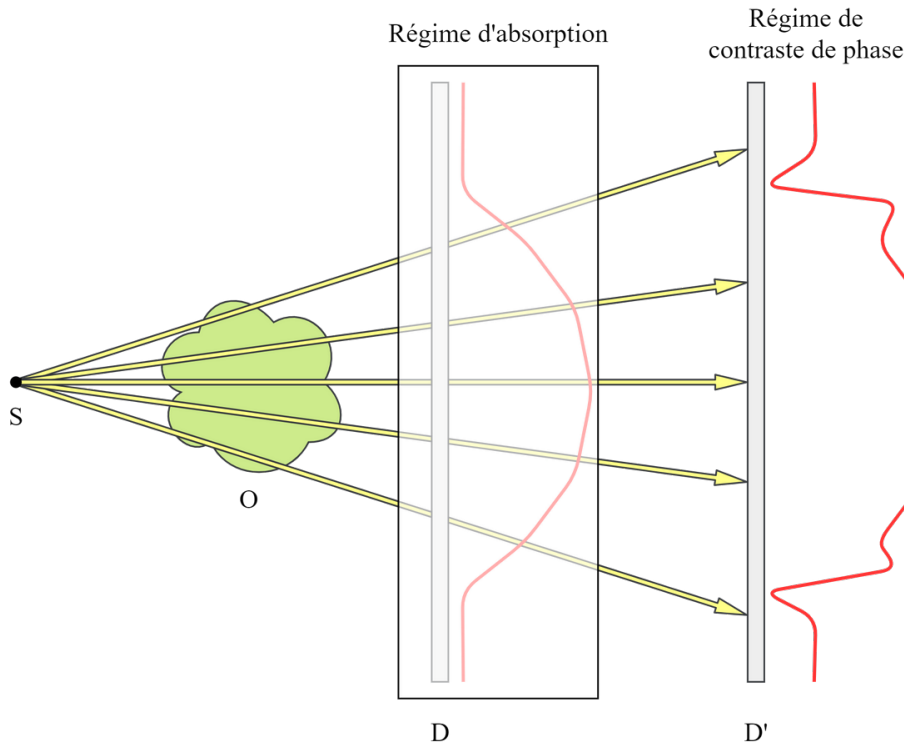


FIGURE 2.6 – Schéma d'un système d'imagerie par contraste de phase. Le détecteur est placé suffisamment loin de l'objet pour que la réfraction de celui-ci cause des interférences sur l'image capturée, qui seront ensuite analysées.

2.3 État de l'art des méthodes de phase en propagation libre

2.3.1 Les différentes méthodes d'extractions directes de la phase en PBI

Les méthodes d'extraction de phase pour les images acquises en propagation libre reposent sur l'inversion de la propagation d'un front d'onde pour retrouver la phase à la sortie de l'objet à partir de l'intensité propagée à une distance supérieure. Dans le cas général, il n'est pas possible d'inverser cette formule avec une seule image capturée, mais avec deux images prises à deux distances de propagation différentes, des formules permettent alors de retrouver une carte de phase unique.

Dans le cadre du projet XPulse, et plus généralement pour l'imagerie médicale où l'exposition aux rayons X doit être gardée la plus courte possible. Capturer deux images à chaque angle de la tomographie n'est donc pas souhaitable, cette méthode est donc écartée et nous nous concentrons sur les méthodes de récupération de phase qui visent à se passer de la seconde acquisition. Dans ce cas-ci, il est nécessaire de faire des approximations pour pouvoir retrouver la valeur de la phase, ou tout du moins s'en rapprocher. De nombreuses méthodes pour résoudre ce problème existent pour différents cas de figure.

$$\frac{2\pi}{\lambda} \frac{\partial}{\partial z} I(x, y, z) = -\nabla_{x,y} \cdot [I(x, y, z) \nabla_{x,y} \Phi(x, y, z)] \quad (2.1)$$

La majorité des méthodes reposent sur l'équation de transfert d'intensité (2.1), qui définit l'évolution selon l'axe de propagation (z) du champ d'intensité I en fonction du gradient du plan orthogonal de l'intensité et du gradient second de la phase Φ [35].

Pour dresser l'état de l'art, nous nous sommes appuyés sur l'article [36] qui décrit sept méthodes de récupération de phase en propagation libre avec une seule distance d'acquisition. Toutes ces méthodes reposent sur un filtrage dans le domaine de Fourier (● dans les formules), avec une normalisation de l'image en entrée (●) et pour certaines une normalisation dans le domaine spatial (●). Dans tous les cas, ces méthodes supposent que l'approximation de Fresnel est vraie, donc que la géométrie est parallèle ou quasi-parallèle (dite paraxiale) [37]. En géométrie conique, c'est le cas lorsque la distance source-objet est largement supérieure à la taille de l'objet imagé, ce qui est souvent vrai.

Ces méthodes ont un certain nombre de paramètres en commun : u et v les fréquences du domaine de Fourier, avec $w = \sqrt{u^2 + v^2}$, d la distance de propagation, λ la longueur d'onde de rayonnement (qui est considéré monochromatique) et \mathcal{F} la transformée de Fourier. Les opérations sur des matrices (comme $\frac{I}{I_0}$) sont faites élément par élément.

$$N_F = \frac{a^2}{\lambda z} \quad (2.2)$$

- **Méthodes de Bronnikov** [38] : Cette méthode fait la supposition que l'objet imagé a une absorption nulle et que le nombre de Fresnel (équation 2.2, avec λ la longueur d'onde, z la distance de propagation, et a la taille du plus petit détail à imager) est très grand. La condition d'une absorption nulle est très forte et rend son utilisation restreinte à des cas très particuliers. (équation (2.3))

$$\phi = \mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[\frac{I}{I_0} - 1 \right]}{2\pi\lambda d |w|^2} \right] \quad (2.3)$$

- **Méthode de Bronnikov modifiée** [39] : Une forme plus récente de la méthode de Bronnikov qui introduit un paramètre α permettant de contre-balancer une absorption faible et de sortir du cas où elle doit être parfaitement nulle. Excepté ce point, les autres conditions d'application de la méthode de Bronnikov restent vraies. (équation (2.4))

$$\phi = \mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[\frac{I}{I_0} - 1 \right]}{2\pi\lambda d |w|^2 + \alpha} \right] \quad (2.4)$$

- **Dualité phase-atténuation** : Cette méthode ne requiert pas une absorption nulle, mais que l'absorption soit proportionnelle au déphasage. Cela implique généralement qu'un seul matériau soit imagé. Il faut également que le signal de phase varie lentement et les autres conditions d'application de Bronnikov restent vraies. Elle ajoute deux constantes au calcul, r_e le rayon d'un électron et σ_{KN} , la section efficace des photons diffusés à partir d'un seul électron libre. (équation

(2.5))

$$\phi = \frac{\lambda r_e}{\sigma_{KN}} \ln \left(\mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[\frac{I}{I_0} \right]}{2\pi \frac{r_e \lambda^2 d}{\sigma_{KN}} |w|^2 + 1} \right] \right) \quad (2.5)$$

- **Méthode de Paganin [24]** : Cette méthode fait également l'approximation d'une absorption et phase proportionnelles (donc un matériau unique), mais ne nécessite pas que celle-ci varie lentement. Pour cette formule, il faut connaître les indices δ et β de l'indice de réfraction du matériau imagé. Les conditions $N_F \gg 1$ et l'approximation de Fresnel sont toujours requises. (équation (2.6))

$$\phi = -\frac{1}{\mu} \ln \left(\mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[\frac{I}{I_0} \right]}{4\pi^2 d \frac{\delta}{\mu} |w|^2 + 1} \right] \right) \quad (2.6)$$

- **Méthode de Paganin multi-matériaux [40]** : Une extension de la formule de Paganin qui permet de retrouver la phase lorsque deux matériaux sont imagés. Pour parvenir à cela, la formule nécessite les indices δ et β des deux matériaux, et introduit A l'épaisseur totale traversée de l'objet imagé par les rayons, qui doit varier lentement. (équation (2.7))

$$\phi = -\frac{1}{\mu_1 - \mu_2} \ln \left(\mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[\frac{I e^{\mu_2 A(r)}}{I_0} \right]}{4\pi^2 d \frac{\delta_1 - \delta_2}{\mu_1 - \mu_2} |w|^2 + 1} \right] \right) \quad (2.7)$$

- **Méthode de Fourier (Born) [41]** : Cette méthode impose l'approximation de Fresnel, une absorption soit faible, soit proportionnelle à la phase, et l'approximation de Born ($|\Psi| \ll 1$, amplitude et phase très faible). (équations (2.8) et (2.9))

$$\text{(Pour } \mu \approx 0) \quad \phi = \mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[0.5 \left(\frac{I}{I_0} - 1 \right) \right]}{\sin(\pi \lambda d |w|^2)} \right] \quad (2.8)$$

$$\text{(Pour } \mu \propto 0) \quad \phi = \mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[0.5 \left(\frac{I}{I_0} - 1 \right) \right]}{\mu \cos(\pi \lambda d |w|^2) \sin(\pi \lambda d |w|^2)} \right] \quad (2.9)$$

- **Méthode de Fourier (Rytov) [41]** : Cette méthode est presque identique à la méthode précédente à la différence que l'approximation de Rytov est utilisée à la place de celle de Born ($|\nabla \Psi| \ll |\nabla^2 \Psi|$, gradient du front d'onde largement inférieur à son laplacien). L'unique différence dans la formule est le changement dans la normalisation de l'image en entrée ($0.5 \left(\frac{I}{I_0} - 1 \right)$ et $0.5 \ln \left(\frac{I}{I_0} \right)$). (équations

(2.10) et (2.11))

$$\text{(Pour } \mu \approx 0) \quad \phi = \mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[0.5 \ln \left(\frac{I}{I_0} \right) \right]}{\sin(\pi \lambda d |w|^2)} \right] \quad (2.10)$$

$$\text{(Pour } \mu \propto 0) \quad \phi = \mathcal{F}^{-1} \left[\frac{\mathcal{F} \left[0.5 \ln \left(\frac{I}{I_0} \right) \right]}{\mu \cos(\pi \lambda d |w|^2) \sin(\pi \lambda d |w|^2)} \right] \quad (2.11)$$

Ces méthodes forment la catégorie des méthodes de récupération de phase directes. Elles permettent d'obtenir une image de phase très rapidement si les conditions de validité de ces formules sont remplies.

Il existe également un algorithme de récupération de phase itératif permettant, au prix de temps de calcul, de se passer de certaines des conditions des méthodes directes. C'est ce que je vais présenter dans la partie suivante.

L'article [36] présente un tableau récapitulatif des différentes méthodes reproduit en table 2.1.

2.3.2 Méthodes Itératives de récupération de la phase

La récupération de phase par méthode itérative est un algorithme connu et utilisé dans le cas de la diffraction cohérente par rayons X [42]. Dans cette configuration, il est possible de retrouver le front d'onde complexe (amplitude et phase) sans information sur les matériaux imagés et sans approximation d'un comportement de celui-ci.

La méthode que j'ai étudiée pour tenter d'utiliser cet algorithme avec notre installation est basée sur la réduction d'erreur de Fienup (ER) [42], [43], et nous avons également effectué des tests avec l'*Hybrid Input-Output* (HIO). Elle consiste à retrouver le front d'onde complexe au niveau de l'objet en connaissant l'amplitude mesurée au niveau du détecteur. Pour cela, elle fait appel à un propagateur de front d'onde [44], [45] permettant de simuler la propagation d'un front d'onde le long de l'axe optique, pour se placer au choix au niveau de l'objet ou au niveau du détecteur. En appliquant des contraintes au niveau de l'objet et du détecteur, on peut faire converger le front d'onde pour obtenir le résultat souhaité (figure 2.7) Le calcul est bien plus coûteux que les autres méthodes, mais nous permettrait de nous passer de l'approximation d'un seul matériau [43].

Plusieurs propagateurs de front d'onde existent, prenant la forme d'une convolution (généralement faite par transformée de Fourier), comme la propagation du spectre angulaire ou la propagation de Fresnel. Différents propagateurs seront plus ou moins adaptés à la nature de la propagation (distance, taille de pixel, ...).

Nous avons opté pour l'utilisation du propagateur du spectre angulaire, comme utilisé dans l'article [43], mais des tests réalisés avec le propagateur de Fresnel n'ont pas modifié de manière significative nos résultats. Le propagateur du spectre angulaire est formulé de manière très simple en notation d'opérateur avec ψ^i le front d'onde au niveau de l'objet, ψ_d^i au niveau de détecteur, P et P^{-1} les propagateurs avant et arrière (figure 2.8).

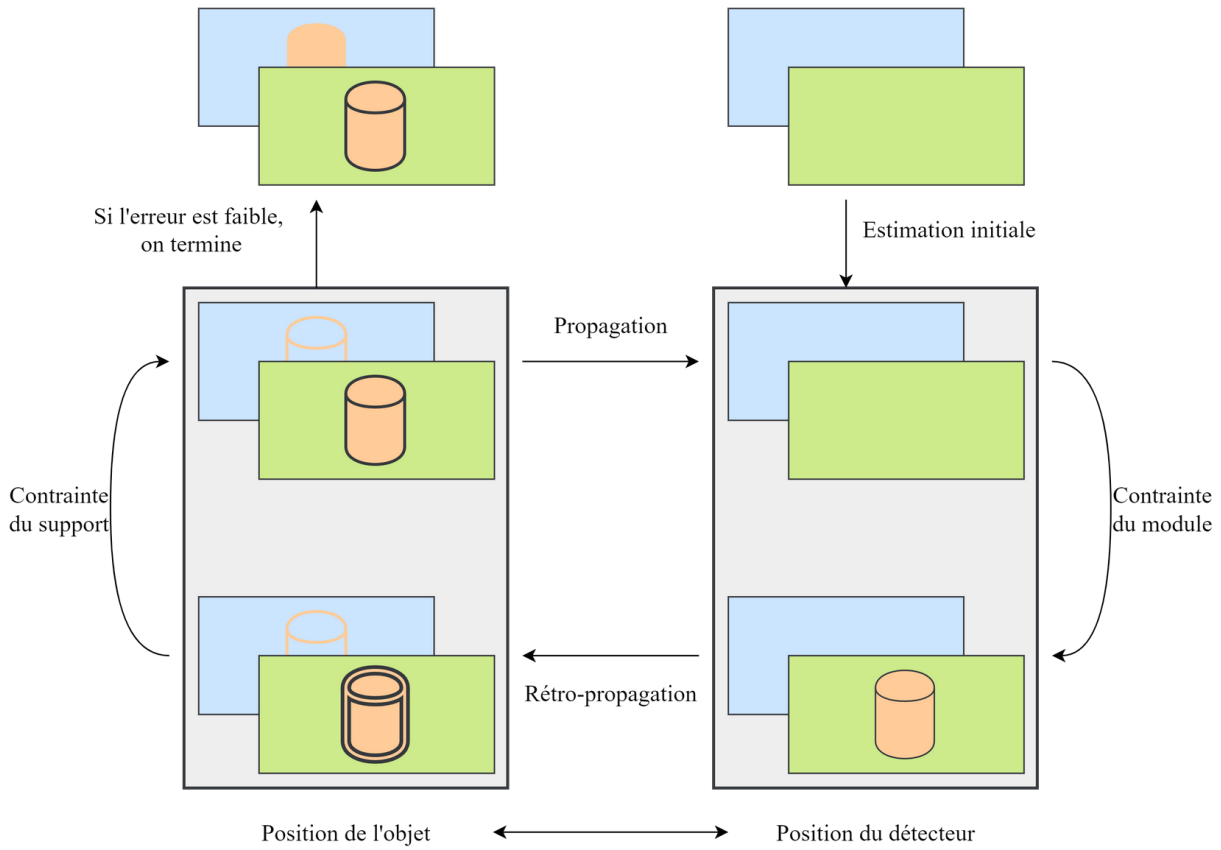


FIGURE 2.7 – Schéma de fonctionnement de la récupération itérative de phase. Le front d’onde est propagé entre le détecteur et l’objet et des contraintes sont appliquées à chaque position selon la connaissance que l’on a sur l’objet imagé.

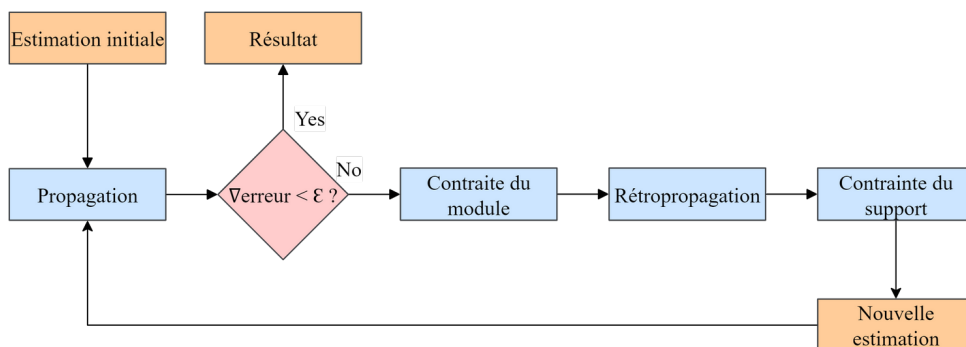


FIGURE 2.8 – Diagramme de fonctionnement de la récupération itérative de phase

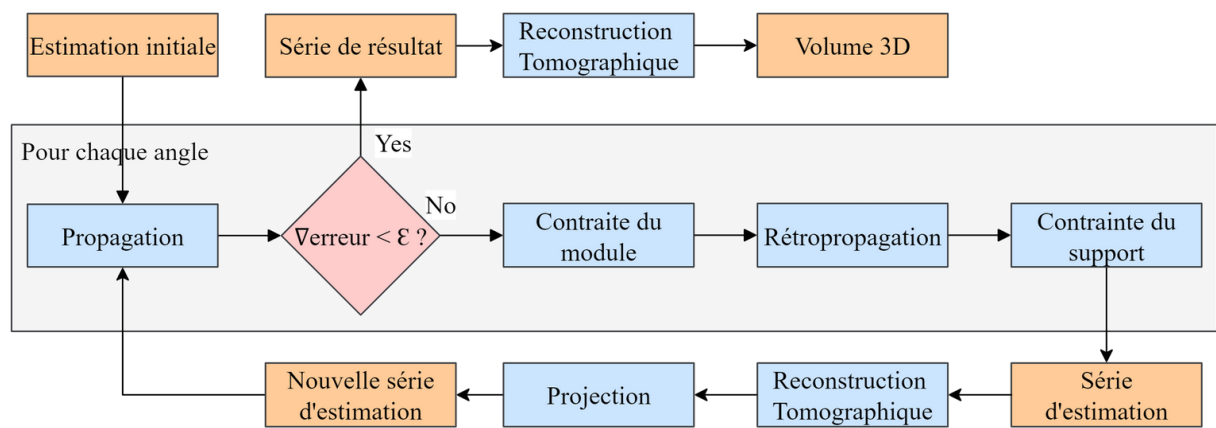


FIGURE 2.9 – Diagramme de fonctionnement de la récupération itérative de phase 3D.

| Méthode | $\mu = 0$ | $\mu \approx 0$ | $\mu \propto \delta$ | 2 matériaux | Fresnel | $N_F \gg 1$ | Signal varie lentement | Born | Rytov |
|-------------------------------|-----------|-----------------|----------------------|-------------|---------|-------------|------------------------|------|-------|
| Bronnikov | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| Bronnikov modifié | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| Dualité phase-atténuation | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ |
| Formule de Paganin | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| Paganin multi-matériaux | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✓ | ✓ | ✗ |
| Méthode de Fourier, Born (a) | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ |
| Méthode de Fourier, Born (b) | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ |
| Méthode de Fourier, Rytov (a) | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ |
| Méthode de Fourier, Rytov (b) | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ |

TABLE 2.1 – $\mu = 0$ signifie une absorption nulle, $\mu \approx 0$ signifie une absorption négligeable, $\mu \propto \delta$ signifie un rapport $\frac{\delta}{b}$ constant, \mathcal{Z} *matériau* signifie que la méthode peut récupérer la phase de deux matériaux, *Fresnel* signifie que l'approximation de Fresnel est utilisée, N_F signifie que le nombre de Fresnel est largement supérieur à 1 (équation (2.2)), avec a un détail de l'image et z la distance de propagation), *Born* et *Rytov* implique que l'approximation correspondante est nécessaire. Tableau issu de l'article [36].

La contrainte du module π_m décrit par l'équation (2.13) qui revient à fixer la valeur de l'amplitude du front d'onde après propagation au niveau du détecteur, puis d'effectuer une rétropropagation pour revenir au niveau de l'objet.

Ensuite, la contrainte du support π_s (2.15) (pouvant prendre différentes formes [42]), force la position de l'objet sur le front d'onde.

$$\psi^{i+1} = \pi_s (\pi_m (\psi^i)) \quad (2.12)$$

$$\pi_m (\psi) = P^{-1} [\pi_m^d (P [\psi])] \quad (2.13)$$

$$\pi_m^d (\psi) = \frac{\psi}{|\psi|} \sqrt{I} \quad (2.14)$$

$$\pi_s (\psi) = \begin{cases} \psi, & \text{Si support.} \\ 0 & \text{Sinon.} \end{cases} \quad (2.15)$$

On commence par une estimation de ψ^0 , donc au niveau de l'objet, qui peut être uniforme ou aléatoire si l'on n'a pas d'estimation à proposer. On le propage ensuite vers le détecteur, où l'on connaît l'amplitude qu'il doit avoir, et que l'on force : c'est la contrainte du module (équation (2.13)). On est maintenant au niveau du détecteur, avec la bonne amplitude, mais avec une phase erronée.

On rétropropage ce front d'onde dans l'autre sens, vers l'objet, où l'on ne connaît que l'amplitude hors de l'objet, qui doit être à 0 : c'est la contrainte du support. On peut ajouter à cette étape d'autres contraintes sur le signal que l'on doit observer si on est en mesure de les connaître [46].

En recommençant ces étapes, on fait converger peu à peu la phase à chaque propagation, jusqu'à ce que le front d'onde au niveau de l'objet projeté sur le détecteur donne l'amplitude mesurée, auquel cas l'algorithme a fini de converger.

Pour évaluer l'erreur et détecter la convergence, on calcule l'erreur quadratique moyenne entre l'amplitude calculée au niveau du détecteur et celle mesurée par la formule :

$$\chi^2 = \frac{\sum_{n=0}^{N-1} ((\psi_n)^2 - I_n)^2}{\sum_{n=0}^{N-1} I_n} \quad (2.16)$$

Pour l'opérateur de propagation, plusieurs formules existent et fonctionnent dans différentes contraintes. J'ai utilisé la formule de propagation de la méthode du spectre angulaire [44], [47], [48], qui est définie par l'équation (2.17), avec z la distance de propagation, k le nombre d'onde et f_x, f_y les fréquences dans le domaine de Fourier.

$$F^{-1} e^{(iz\sqrt{k^2 - f_x^2 - f_y^2})} F \quad (2.17)$$

Cela revient à une multiplication dans le domaine de Fourier. Elle est réversible (si la distance est négative, la formule s'inverse), et les conditions d'applications sont valides pour nos géométries d'acquisition ($F_n \simeq 1$ et géométrie paraxiale).

Si les conditions théoriques d'application de cet algorithme sont respectées dans notre cas, son utilisation se montre problématique. La contrainte du support est très puissante dans le cas de la diffraction où de l'holographie, mais dans notre cas les interférences ne sont définies que sur quelques pixels et contraindre le front d'onde de cette manière n'a que peu d'impact. De plus, l'opérateur de propagation ayant un effet très faible et local sur notre image, seuls les contours des objets parviennent à converger vers un résultat.

Une variation de cette méthode permet d'améliorer la compatibilité avec notre installation, en intégrant la récupération de phase dans la reconstruction tomographique [49] (figure 2.9). Si le résultat obtenu est meilleur, il ne nous permet toujours pas d'obtenir des valeurs exploitables.

Nous avons fini par conclure que dans notre configuration, cet algorithme ne nous permet pas de récupérer un signal exploitable.

2.3.3 La méthode d'extraction de phase retenue

Après avoir étudié les différentes méthodes et implémenté la plupart, il est clair que les méthodes directes sont bien plus robustes que les méthodes itératives dans nos conditions.

Nous avons donc opté pour celles-ci, et avons dû les départager pour choisir celle qui convient le mieux au projet. Tout d'abord, l'hypothèse d'une absorption nulle ou très faible n'est pas justifiée dans la plupart de nos acquisitions, on a donc pu rapidement éliminer les méthodes correspondantes. Il reste donc les méthodes de Paganin et celles de Fourier.

D'après [50], la méthode de Fourier a tendance à introduire un effet de *cupping* (le signal au bord de l'objet est plus haut que celui dans l'objet), qui pourrait causer des artefacts importants sur la reconstruction.

Un autre argument en faveur de la méthode de Paganin est son utilisation très répandue pour l'imagerie de phase, notamment dans le contexte clinique. C'est un algorithme qui a donc largement été éprouvé dans un environnement proche du nôtre.

Pour la version multi-matériaux de la formule de Paganin, obtenir la carte d'épaisseur traversée nécessite de réaliser une reconstruction complète ainsi qu'une segmentation avant le traitement de la phase, ce qui allonge fortement le temps de traitement. Pour une complexité en calcul similaire, il est possible de réaliser la reconstruction en traitant la phase selon les deux ratios, puis de segmenter les deux reconstructions [51].

Il est également possible d'appliquer la formule de Paganin après la reconstruction tomographique [52] ou durant celle-ci [53].

Nous avons donc décidé d'utiliser la formule de récupération de phase de Paganin pour traiter les images acquises en propagation libre. Pour nous assurer de la validité de notre implémentation, nous nous sommes appuyés sur une implémentation existante pour obtenir nos premiers résultats, ANKAPhase [54].

2.4 État de l'art des méthodes de phase Hartmann

2.4.1 Masque de Hartmann

La seconde méthode d'imagerie de phase utilisée et développée durant le projet XPulse est l'imagerie par masque de Hartmann. Cette technique consiste à mesurer la réfraction des rayons, dont l'importance dépend de la valeur de δ , lui-même directement relié au retard de phase.

Un masque de Hartmann est une plaque percée de trous placée entre l'objet et le détecteur servant à définir dans le faisceau incident une matrice de sous-faisceaux (figure 2.10) qui seront individuellement suivis sur le détecteur pour pouvoir déterminer la réfraction induite par l'objet pour chacun d'eux. La plaque doit être opaque pour l'énergie

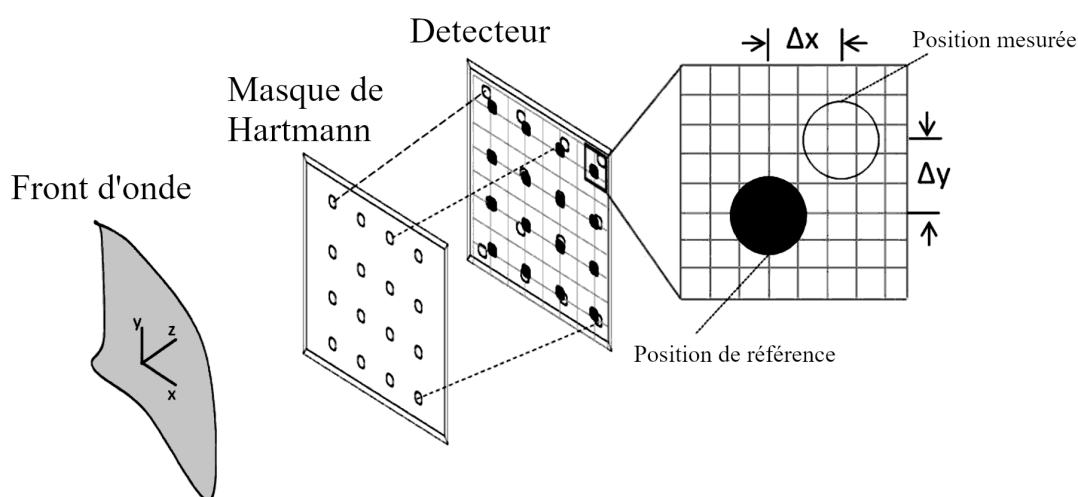


FIGURE 2.10 – Schéma du système d'imagerie de phase Hartmann à rayons X, montrant un front d'onde incident se déplaçant le long de l'axe z , illuminant un réseau d'ouvertures et les taches générées sur le détecteur, et le déplacement d'une seule tache par rapport à l'emplacement de référence sur le détecteur. Modifié à partir de [55].

des rayons X utilisée afin d'obtenir un contraste suffisant, et la taille et l'espacement des trous doit correspondre au grandissement et à la taille des pixels du détecteur. La résolution finale de l'image dépendra du pas entre deux faisceaux, étant donné que chaque faisceau représente une valeur dans l'image (plus précisément une carte de déflexion). Chaque faisceau doit pouvoir être résolu sur le détecteur, imposant un pas minimum, tandis que la taille des spots doit être suffisamment grande pour pouvoir approximer la position avec une précision plus grande que la taille du pixel. En effet, si le faisceau est plus petit qu'un pixel au niveau du détecteur, il sera alors très compliqué, voire impossible, de savoir exactement où il se situe, sur ce pixel. À l'inverse, plus le faisceau couvre de pixels, plus il sera possible de calculer la tache gaussienne le représentant et donc d'obtenir une position précise. On a donc un compromis à faire entre une taille de faisceau suffisamment grande pour pouvoir le résoudre correctement, et suffisamment petite pour conserver une définition suffisante.

Un tel capteur capture donc un réseau de faisceaux de rayons X déviés et nécessite un traitement pour extraire les valeurs recherchées. Tout d'abord, la grille de faisceaux doit

être résolue pour extraire la déviation locale de chaque spot. Il en résulte une image qui peut être considérée comme la dérivée du front d'onde. La carte de déflexion résultante ne peut pas être utilisée directement pour une reconstruction tomographique sans traitement spécifique pour reconstruire le front d'onde. L'installation étant destinée à un usage clinique, l'exposition aux rayons X doit être maintenue aussi faible que possible, nous devons donc être capables de traiter des images avec une quantité élevée de bruit.

Les algorithmes itératifs existants utilisés pour intégrer la carte de déviation [25] conduisent à des artefacts de basse fréquence lorsqu'ils sont confrontés à des images bruitées, ce qui a un impact négatif sur le résultat de la tomographie, car l'information incohérente de chaque projection se propage dans le tomogramme.

2.4.2 Algorithmes d'intégration de la phase

La carte de déviation doit être intégrée pour obtenir les valeurs du front d'onde liées à la phase qui seront utilisées dans la reconstruction tomographique itérative. En raison du bruit de la capture, les méthodes directes telles que l'intégration numérique ou l'intégration dans le domaine de Fourier ne donnent pas le résultat escompté (artefacts à basse et haute fréquence, fond variable). Pour éviter ce problème, des algorithmes dédiés aux capteurs Hartmann ou équivalents ont été développés. L'algorithme le plus courant pour la reconstruction d'un front d'onde arbitraire est décrit dans [25] comme un algorithme itératif.

Les méthodes de Gauss-Seidel (GS) et de Jacobi sont basées sur les mêmes algorithmes, mais la convolution est effectuée *in-place* (GS) ou non (Jacobi) [25]. La méthode *Successive-Over-Relaxation* (SOR) est une extension de cet algorithme qui introduit un paramètre de relaxation utilisé pour contrôler le taux de convergence. La méthode SOR converge généralement plus rapidement que les autres lorsqu'on lui donne un paramètre de relaxation optimal. Cela implique néanmoins de trouver ce paramètre optimal. La méthode pour le calculer donnée dans l'article n'étant pas adapté au cas d'une image réelle bruitée, le résultat a tendance à diverger après un certain nombre d'itérations.

La méthode GS demande plus d'itérations que SOR, mais ne nécessite pas de paramètre supplémentaire, ce qui la rend plus facile à appliquer et plus robuste si le paramètre de relaxation n'est pas correct.

Bien que ces algorithmes fournissent des résultats pratiques, le bruit d'acquisition dans les images acquises dégrade la mesure des faisceaux et donc la carte de déviation extraite. Ces défauts causent des artefacts à basse fréquence à la suite de l'intégration, en particulier sur les bords de l'image.

C'est un problème pour les algorithmes de tomographie puisque les projections de front d'onde traitées ne sont plus homogènes entre elles, ce qui génère des artefacts plus importants sur le tomogramme reconstruit.

Certaines améliorations pour ces algorithmes ont été proposées [56]-[58], mais les problèmes liés à la convergence séparée des projections ne sont pas impactés.

Pour obtenir un jeu de projections cohérentes, il est alors nécessaire de trouver une méthode permettant de lier l'information présente dans chacune des projections pour éviter qu'elles ne convergent individuellement vers un résultat différent. C'est sur ce point

que j'ai travaillé pour permettre d'obtenir une meilleure reconstruction tomographique avec ce système, et que j'approfondis dans le chapitre 4.

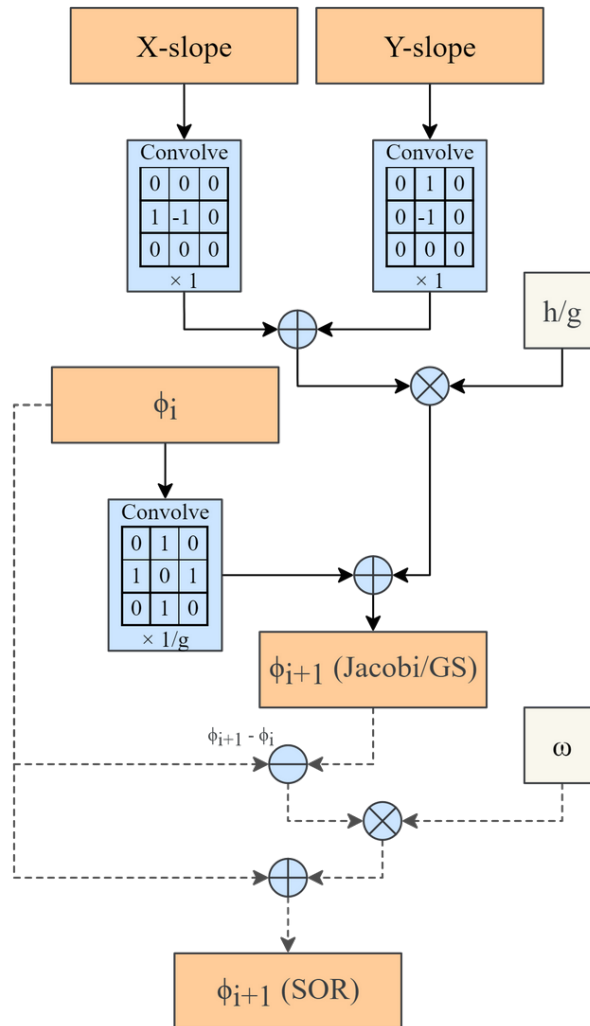


FIGURE 2.11 – Diagramme de fonctionnement de l'intégration itérative de la phase de Jacobi, GS et SOR tel que décrit par Southwell.

2.5 Conclusion

Dans ce chapitre, nous avons vu que l'imagerie par rayons X peut se faire en mesurant d'autres grandeurs que l'absorption par l'objet du rayonnement.

Dans notre cas, nous mesurons le décalage de phase induit par l'objet, qui est directement lié à la réfraction dans celui-ci. Nous avons vu les méthodes majeures d'imagerie de phase, et étudié plus en détail les méthodes qui ont été retenues pour le projet XPulse, la propagation libre et l'imagerie par masque de Hartmann.

L'imagerie de phase permettra, dans le contexte de la mammographie et de ce projet, de mieux détecter et classer les calcifications qui se forment dans le sein et qui peuvent être bénignes ou malignes.

Deuxième partie
Travaux/Méthodes

Chapitre 3

Le projet XPulse

Cette thèse se fait dans le cadre du projet XPulse [59], et son déroulement s'est fait conjointement avec les contraintes et les avancées du projet.

Le projet XPulse est un projet collaboratif financé par la région Nouvelle-Aquitaine et le F.E.D.E.R. (Fond Européen de Développement Régional). Les acteurs participant au projet sont l'hôpital Bergonié en tant que partenaire clinique, le laboratoire CELIA en tant que partenaire universitaire, les entreprises Amplitude Systems qui fournit les lasers et Imagine Optique qui fournit le détecteur de Hartmann, et enfin ALPhANOV jouant le rôle de coordinateur et qui a financé ma thèse.

ALPhANOV est une association loi 1901 créée en 2007 par le Conseil Régional d'Aquitaine, le CEA (Commissariat à l'énergie atomique et aux énergies alternatives), le CNRS (Centre national de la recherche scientifique), l'Université de Bordeaux et Alpha RLH.

Dans ce chapitre, nous allons détailler les contraintes d'imagerie du projet et la place de mon travail de thèse dans celui-ci.

3.1 Setup et Objectif

3.1.1 Le projet XPulse

Le but de ce projet est la conception d'un système d'imagerie par rayon X destiné à la détection précoce du cancer du sein. L'objectif est d'extraire le maximum d'informations des images acquises avec ce système afin d'améliorer le diagnostic, tout en réduisant la dose déposée et en améliorant le confort de la patiente. Il est financé initialement sur une durée de 5 ans s'étant terminé en décembre 2022, et un financement pour la seconde partie du projet débute en 2023.

La principale innovation du projet est le développement d'une source de rayons X générés par laser qui permettra d'obtenir la brillance et la cohérence nécessaire pour de l'imagerie par contraste de phase. Les rayons X sont générés par l'interaction laser-plasma, obtenu en focalisant un laser pulsé puissant sur une bande métallique. Cette bande est constamment renouvelée par un système de bobines permettant au laser de toujours tirer sur une zone non fondue de celle-ci. L'intérêt pour le contraste de phase dans ce projet

est que les tissus mous, notamment en mammographie, ont une variation de densité très faible, mais peuvent induire un décalage de phase important [60].

L'équipe XPulse au sein d'ALPhANOV est composée de quatre membres permanents : Aboubakr Bakkali, le chef de projet, Benjamin Barbreil, Octave Delorme et Francesca Mastropietro et de deux doctorants, Panyi Song travaillant sur la conversion laser-rayons X et moi-même. Deux membres de l'équipe mécanique, Charly Reignault et Damien Bonaudo travaillent également sur le projet de manière régulière.

Un prototype de la machine d'acquisition finale a été installé dans des locaux au CHU (Centre Hospitalier Universitaire) Pellegrin au sein de l'Institut de Bio Imagerie (IBIO)

3.1.2 Les configurations d'imagerie du projet

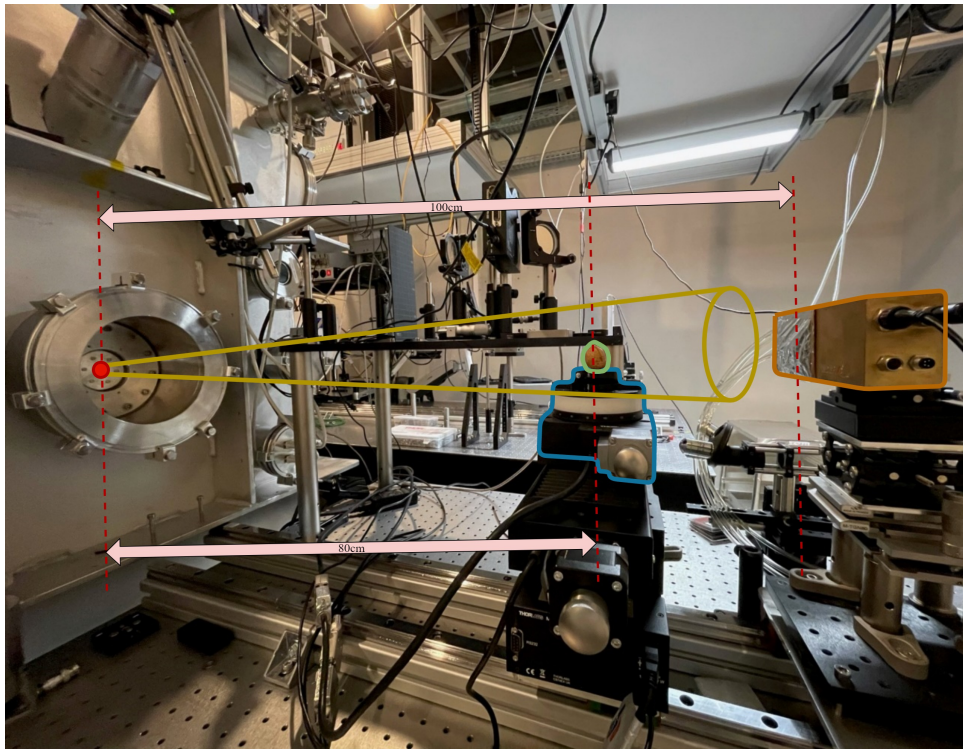


FIGURE 3.1 – Photographie de l'installation de tomographie par source laser, avec la source (point rouge), le prote échantillon (bleu), l'échantillon (vert) et le détecteur (orange).

Durant la durée du projet, trois détecteurs ont été utilisés :

- Advacam, de définition 1280x256 avec une taille de pixel de 45 μm .
- Teledyne DALSA de définition 5888x4608 avec une taille de pixel de 50 μm . C'est ce détecteur qui a été principalement utilisé pour réaliser les acquisitions pendant le développement du projet.
- DirectConversion de définition 5120x256 avec une taille de pixel de 100 μm . Ce détecteur a la particularité d'être composé de 5 modules placés horizontalement en arc de cercle. Du fait de la faible hauteur, il sera principalement utilisé en tomographie hélicoïdale. Ce détecteur correspond le mieux au format final du projet.

Trois sources ont été utilisées durant le projet :

- Une source Excillum MetalJet, avec une anode liquide faite principalement de gallium et d'indium et produisant un spectre polychromatique avec un pic d'émission à 9.2 keV. C'est avec cette source que la plupart des tests et des acquisitions ont été faits.
- Une source Hamamatsu, avec un tube à rayon X à microfocus avec une cible en tungstène.
- La source XPulse, développée pour le projet et générant des rayons X grâce à l'interaction laser-métal. La cible métallique prend la forme d'une bande souple qui défile dans une cassette, les caractéristiques d'émission dépendent donc du métal choisi. Les tests ont été réalisés à l'aide d'une bande en molybdène.



FIGURE 3.2 – Photographie de l'installation de tomographie installée à l'IBIO. Le détecteur, la source à rayons X et le plateau tournant sont situés sous le lit.

Trois installations différentes ont également été mises en places avec trois sources de rayons X différentes :

- L'installation de test de reconstruction, avec la source Excillum et les détecteurs Advacam ou Teledyne. Cette installation permet le plus grand grandissement, avec une distance source-objet d'environ 200 mm et une distance source-détecteur pouvant aller jusqu'à 3200 mm. La source et le détecteur sont fixes et l'objet est placé sur une platine rotative.
- L'installation avec source par laser, utilisée généralement avec le détecteur Advacam du fait de sa meilleure sensibilité, car la puissance du laser durant le

développement du projet est relativement faible par rapport à sa version définitive. Comme pour la précédente, c'est l'objet qui tourne dans cette configuration. (Figure 3.1)

- L'installation de L'IBIO, qui représente la géométrie visée pour l'installation finale. Cette fois, c'est la source (Hamamatsu) et le détecteur (DirectConversion) qui tournent et bougent verticalement. Ici les distances sont fixes, avec une distance source-objet de 450 mm et source-détecteur de 900 mm. La courbure du détecteur est telle que la source est confondue avec son centre de courbure. (Figure 3.2)

Les installations sont utilisées aussi pour l'imagerie d'absorption, l'imagerie par propagation libre, et pour l'imagerie par masque de Hartmann, dont la configuration est expliquée sans la section suivante. Mon travail dans ce contexte a été de mettre en place et d'optimiser la chaîne de traitement des images, pour réaliser le prétraitement des images brutes, l'extraction de la phase et la reconstruction tomographique, pour les différentes géométries et méthodes d'imagerie.

3.1.3 Système d'imagerie par masque de Hartmann

La configuration d'imagerie par masque de Hartmann que nous avons développée est composée d'une source de rayons X à microfocus, d'un détecteur planaire et d'un capteur Hartmann développé par Imagine Optic [61] (figure 3.3). Cette configuration permet d'obtenir une grille de spots sur l'image capturée, dont la position sur le détecteur est décalée en fonction de la déviation causée par l'objet imagé.

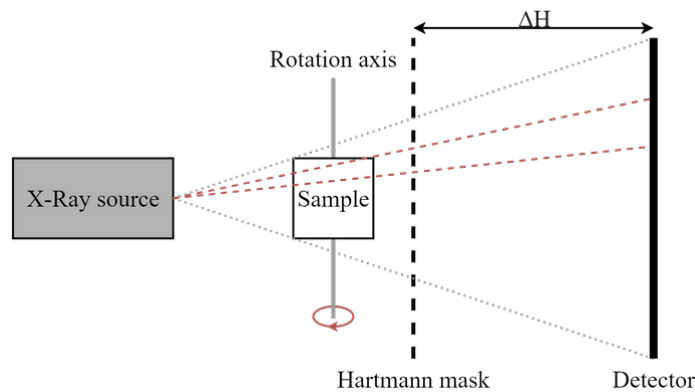


FIGURE 3.3 – Schéma d'un système d'imagerie par masque de Hartmann

Une acquisition de *flatfield* (sans objet) doit être réalisée pour calibrer les positions des spots avant l'acquisition tomographique. La tomographie est réalisée en plaçant l'objet sur un support mobile. La configuration est illustrée dans la figure 3.3.

Plusieurs étapes de traitement sont nécessaires pour obtenir un tomogramme 3D des informations liées à la phase à partir de la grille de faisceau. Les points doivent d'abord être transformés en une carte de déflexion représentant la distance entre les points d'une image et une référence. Cette carte de déflexion doit ensuite être traitée pour reconstruire le front d'onde approprié, qui est ensuite utilisé pour la reconstruction tomographique itérative.

3.2 État de l'art des logiciels de tomographie

De nombreuses méthodes et implémentations d'algorithmes de reconstruction tomographique sont en développement actif aujourd'hui, autant dans le domaine de la recherche que de l'industrie.

Les principales méthodes de reconstruction utilisées pendant longtemps étaient basées sur la FBP dû à sa simplicité et à son coût réduit en puissance de calcul. En effet, si les algorithmes de reconstruction tomographique itératifs sont développés depuis longtemps, avec notamment l'ART en 1980, les capacités de calcul limitées des ordinateurs ont fait que les implémentations pour ordinateur classique n'ont réellement commencé à se développer que vers les années 2010.

La reconstruction tomographique étant une opération très coûteuse autant en termes de temps de calcul que d'espace mémoire, son exécution sur des données réelles avec une définition exploitable était et reste aujourd'hui complexe (voir table 3.4). Les implémentations sur CPU notamment sont très lentes, demandant des heures voire des jours de calculs. Avec l'augmentation des capacités de calcul des GPU moderne et l'essor du GPGPU, l'accélération du calcul de la reconstruction tomographique est devenu un impératif tant la différence en temps de calcul par rapport aux versions CPU est grande [62], [63]. Il faut néanmoins prendre en compte le fait que le calcul sur GPU pose des problématiques de gestion de la mémoire du fait de sa taille réduite, ce qui fait que des implémentations utilisant uniquement le CPU sont encore développées aujourd'hui [64].

| Angles | Définition | Taille des proj. | Taille du vol. | Total |
|--------|------------|------------------|----------------|-----------|
| 500 | 512x512 | 0.48 Go | 0.5 Go | 0.98 Go |
| 500 | 1024x1024 | 1.95 Go | 4 Go | 5.95 Go |
| 500 | 2048x2048 | 7.81 Go | 32 Go | 39.81 Go |
| 500 | 4096x4096 | 31.25 Go | 256 Go | 287.25 Go |
| 500 | 5888x4608 | 50.54 Go | 595.13 Go | 645.67 Go |

FIGURE 3.4 – Exemple de taille de données pour des acquisitions planaires avec un détecteur de définition donnée. La définition finale, 5888x4608, est celle du détecteur Teledyne Dalsa couramment utilisé durant le projet.

Cet intérêt pour l'utilisation des GPU pour la tomographie étant relativement récent, peu d'implémentations disponibles offrent cette possibilité et des limitations sont encore présentes dans l'état de l'art.

Trois implémentations ont été considérées pour représenter l'état de l'art de la reconstruction tomographique sur GPU : OpenRTK, Astra Toolbox et TIGRE.

OpenRTK (2014) [65] est une implémentation de la reconstruction tomographique faisant partie de la bibliothèque Insight Toolkit (ITK) et développée par le consortium RTK. Elle est écrite en C++/CUDA avec une interface Python. Elle supporte la reconstruction par FDK ainsi que par plusieurs algorithmes de reconstruction itérative (ADMM, Gradient conjugué, SART, OSEM, et autres).

En la compilant soi-même, il est possible d'activer l'accélération par GPU pour ces algorithmes. Cette implémentation n'est néanmoins pas axée sur une optimisation poussée et souffre de la comparaison avec les autres, que ce soit en temps de calcul ou en utilisation mémoire. Sa version GPU ne prend en charge que les jeux de données pouvant rentrer entièrement en mémoire vidéo, ce qui limite grandement les cas d'utilisation.

Astra Toolbox (2016) [66] est une bibliothèque pouvant être utilisée depuis Python ou MATLAB, et est spécialement conçue pour la rapidité d'exécution. Elle est développée par Vision Lab de l'Université d'Anvers, Belgique. Sa version accélérée par GPU supporte la FBP, FDK (Feldkamp-Davis-Kress), et le SIRT et CGLS pour les algorithmes itératifs. Elle fait appel à une optimisation importante pour ses opérateurs de projections qui la rendent très performante. Cela a néanmoins un impact négatif sur l'espace mémoire nécessaire, problème exacerbé par le fait que la reconstruction de données plus grande que la mémoire vidéo est restreinte aux géométries parallèles ou similaires [67], et incompatible en CBCT (*Cone-Beam Computed Tomography*, géométrie conique).

TIGRE (2016) [68] est également une bibliothèque proposant une interface Python et MATLAB et développée par le CERN. Elle utilise de la même manière CUDA pour l'accès au GPU, et supporte le SART, SIRT, FDK, MLEM et autres comme algorithmes de reconstruction itératifs. Son principal avantage est qu'il est capable d'utiliser la RAM du système pour décharger la VRAM dans toutes les géométries [69]. Il ne supporte néanmoins pas véritablement les jeux de données plus grands que la RAM, car il se repose alors exclusivement sur le mécanisme de *swap* du système, ce qui provoque un effondrement des performances. Cette implémentation est plus rapide que RTK, mais moins que Astra Toolbox (environ 2.4 fois plus lente d'après eux-mêmes [68]).

Il existe également un grand nombre d'implémentations propriétaires, généralement associées à un appareil de tomographie avec lequel elles sont commercialisées. Compte tenu de leur nature, ces implémentations n'ont pas pu être testées et comparées avec les autres.

Les implémentations qui ont été testées montrent des limitations quant à la taille maximale des données que l'on peut reconstruire. Pour commencer, en passant par leurs interfaces Python et MATLAB, les trois bibliothèques requièrent de charger *a priori* les données en mémoire. Cela limite grandement les possibilités de reconstruction, car les données doivent pouvoir rentrer entièrement en RAM avant même de commencer le processus de reconstruction. Cette limitation est d'autant plus importante que pour le CBCT, seul TIGRE est capable de répartir la charge mémoire entre la VRAM et la RAM. Les GPU actuels les plus compétants à cet égard n'ont qu'au maximum quelques dizaines de Go de mémoire vidéo. Un volume cubique de taille 2048^3 stocké en nombres flottants 32 bits représente déjà une taille de 32 Go, il est donc clair qu'une implémentation ne pouvant pas répartir la charge mémoire et gérer intelligemment le chargement des données montrera très vite ses limites.

Si TIGRE est de ce fait capable de reconstruire des volumes plus grands que les deux autres implémentations, il montre tout de même une limitation importante dans sa gestion de la mémoire. En effet, il stocke toutes les données en RAM sans considération pour la taille de celle-ci comparée à la taille du jeu de données, ce qui force alors le système

d'opération à gérer lui-même les échanges entre la RAM et le disque si les données sont trop grandes. Si la taille des données est légèrement plus grande que la RAM libre, cela ralentit fortement le processus, mais au-delà (plusieurs fois la taille de la RAM), le temps de calcul devient exponentiellement long. Il est alors courant que le système d'exploitation lui-même rencontre des instabilités et que la reconstruction ne parvienne alors pas à se terminer correctement.

Du fait de la taille des données que nous aurons à traiter, il nous est alors nécessaire de concevoir une implémentation de reconstruction itérative pouvant gérer les transferts de grande quantité de données.

Du fait des particularités de notre système d'acquisition et du fait que toutes n'ont pas pu être évaluées avant de les rencontrer sur des acquisitions, nous avons fait le choix de développer nos propres fonctions de projection au lieu d'utiliser celles des bibliothèques existantes.

3.3 Chaîne de traitement

3.3.1 Chaîne de traitement

Avant de pouvoir reconstruire le volume à l'aide des projections, il faut d'abord s'assurer que celles-ci soient valides et correctement traitées. En effet, une seule projection problématique peut empêcher l'algorithme de reconstruction de converger correctement et certains problèmes sur les projections peuvent mener à des artefacts sur le volume reconstruit [70].

Certaines étapes de la chaîne de traitement doivent être exécutées avant d'autres, et certaines doivent être adaptées selon l'ordre d'exécution. Pour pouvoir reproduire nos résultats au mieux, nous avons mis en place une chaîne de traitement pour traiter toutes nos données. La chaîne de traitement globale est illustrée en figure 3.5.

Chacune des méthodes d'imagerie (absorption, propagation libre et masque de Hartmann) possède des étapes de prétraitement propres. Voici, en détail, les étapes du prétraitement et dans quel ordre nous les appliquons :

- Absorption
 - (1) Retrait des valeurs aberrantes
 - (2) Correction propre au détecteur
 - (3) Normalisation
 - (4) Rotation et recoupage
- Propagation libre
 - (1) Retrait des valeurs aberrantes
 - (2) Correction propre au détecteur
 - (3) Normalisation
 - (4) Rotation et recoupage
 - (5) Récupération de phase par Paganin
- Hartmann
 - (1) Extraction de la carte de déflexion
 - (2) Correction propre au détecteur
 - (3) Intégration de la phase

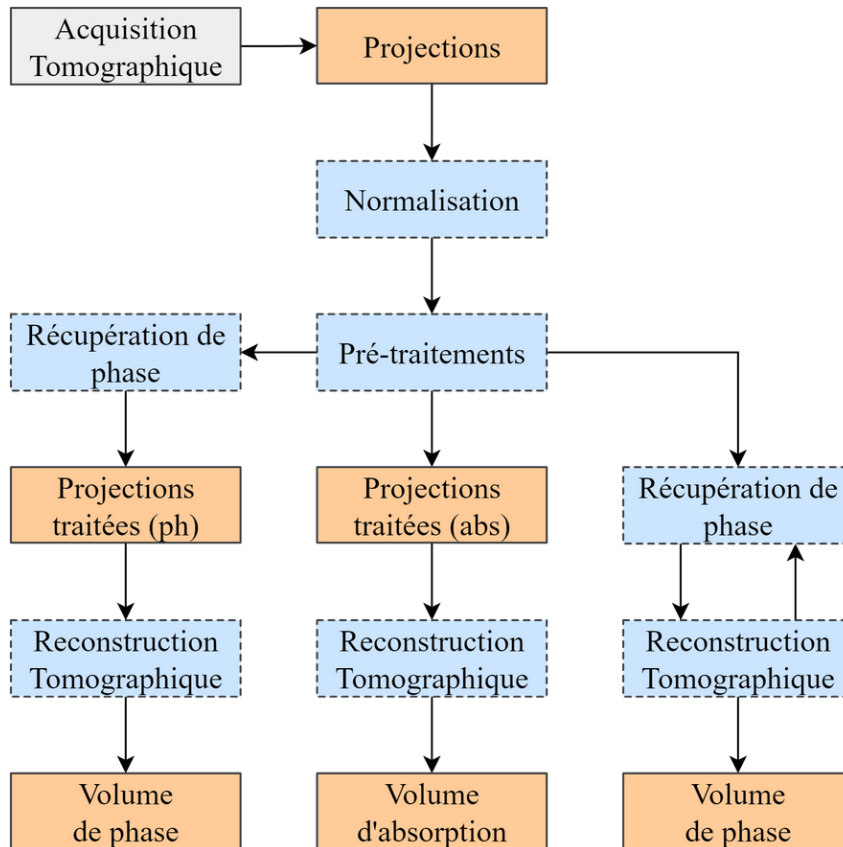


FIGURE 3.5 – Diagramme des chaînes de traitement couramment utilisées lors du traitement des données acquises avec nos installations

(4) Rotation et recoupage

Pour le cas de l'imagerie par masque de Hartmann, la nature des images les rend difficiles à prétraiter sans risquer de modifier les valeurs du résultat. L'étape de normalisation n'est pas nécessaire et est remplacée par l'extraction de la carte de déflexion. Les valeurs au bord des modules des détecteurs ou situées sur des pixels anisotropes ne sont pas exploitées, et ces zones ne pouvant pas être résolues sont soit corrigée à partir des valeurs voisines, soit ignorées lors de la reconstruction.

3.3.2 Méthode de traitement

Nous allons voir dans cette section quelques méthodes de prétraitement utilisées en routine avec le système d'imagerie XPulse.

Flatfields et normalisation Le calcul de l'absorption à partir de l'intensité mesurée nécessite de connaître l'intensité arrivant sur l'objet. Une capture sans l'objet, un *flatfield* est donc effectué pour obtenir ces valeurs. Cette image ne comporte alors que les inhomogénéités de la source et du capteur et permet donc de les atténuer fortement.

Il suffit ensuite, pour obtenir l'absorption A , d'effectuer le calcul suivant avec I l'image enregistrée et I_0 le *flatfield* :

$$A = -\log\left(\frac{I}{I_0}\right) \quad (3.1)$$

Cette méthode présente néanmoins un problème, l'image avec et sans l'objet est capturée indépendamment, les effets temporels sont donc différents entre les deux captures. Cela induit donc que le bruit du capteur n'est pas corrigé, de même que les variations de l'intensité de la source et de la sensibilité du détecteur.

Pour atténuer cet effet, des *flatfields* peuvent être capturés à plusieurs moments durant la tomographie, ce qui réduira les effets temporels de faible fréquence (fréquence inférieure à celle de la prise de *flatfields*) mais n'aura aucune incidence sur les variations rapides.

Lorsque la fréquence de prise de *flatfield* est inférieure à celle de la prise d'image, ils peuvent être interpolés pour simuler le fait que la différence entre deux *flatfield* s'est formée de manière régulière entre temps. Cette approximation n'est pas toujours valide, mais fonctionne bien par exemple pour limiter les effets de l'opacification d'éléments optiques avec la source par laser expérimentale.

Cette étape est propice aux erreurs de valeurs, étant donné qu'elle comporte une division et un logarithme. Il faut donc pouvoir traiter les valeurs problématiques (pixel mort, erreur de capture, objet trop absorbant) avant ou après cette étape pour éviter qu'une valeur non finie ne se propage durant la reconstruction.

Correction des valeurs aberrantes Pour garantir la validité de toutes les valeurs présentes dans les projections, il faut pouvoir détecter et corriger les valeurs non finies ou incohérentes par rapport aux autres.

Une première méthode est de couper les valeurs à un certain centile, ce qui a pour effet d'enlever les valeurs aux bords de l'histogramme. Une autre méthode, donnant des résultats équivalents, est de calculer l'écart-type de l'image, et de couper les valeurs s'éloignant trop de celle-ci. Ces deux techniques suppriment la majorité des valeurs aberrantes, mais peuvent réduire la dynamique de l'image si les seuils sont trop contraignants, et les valeurs incohérentes par rapport à leur voisinage uniquement ne seront pas corrigées. Pour éviter cela, on peut traiter les pixels par rapport aux valeurs et à l'écart-type de son voisinage.

Un filtre médian peut également corriger le problème, en enlevant effectivement toutes les valeurs aberrantes sur un voisinage de taille donnée, mais cause alors une perte de détail, ce qu'il est particulièrement important d'éviter en contraste de phase, où les franges d'interférences peuvent être localisées sur un seul pixel.

Il est également possible de comparer les différentes projections entre elles, pour détecter les pixels ne variant pas et qui sont généralement des pixels morts. Cela permet d'ajouter une contrainte temporelle au traitement des projections. Cette vérification revient à traiter et comparer les lignes du sinogramme de l'acquisition, car une même ligne prise sur chaque image d'une acquisition forme un sinogramme.

Toutes ces méthodes sont valides, mais seront plus ou moins efficaces selon le jeu de données et le type de valeur problématique, ainsi que leur distribution.

Une fois les valeurs incorrectes détectées, le plus simple est de les remplacer par une valeur calculée à partir des voisins, par moyenne ou médiane. Si des zones plus grandes sont manquantes, il faut alors utiliser des méthodes plus complexes, notamment une interpolation à partir des voisins corrects les plus proches.

Si des valeurs incorrectes restent présentes sur les projections, deux types d'artefacts peuvent se présenter : si le problème est local à une projection, un trait risque de se former sur le volume. Si le problème est présent au même endroit sur toutes les projections, on obtient alors un artefact circulaire, qui forme un cercle centré autour de l'axe de rotation.

Corrections de la taille de pixels Deux des capteurs (L'Advacam et le DirectConversion) utilisés dans le projet présentent une surface de détection non continue (composés de modules disjoints) et non homogène. Ce genre de capteur nécessite des traitements dédiés pour éviter que les interstices ne faussent la reconstruction tomographique.

Ces détecteurs sont composés de modules dont les pixels des bords n'ont pas la même taille que les autres, et ces modules sont séparés par des interstices physiques. La différence de taille entre les pixels est déterminée statistiquement à l'aide d'une série de *flatfield* normalisée par les pixels centraux des modules, puis moyennée. L'image résultante est ensuite segmentée pour faire ressortir les différentes tailles de pixel.

La valeur des pixels est ensuite normalisée par leur taille et les interstices sont comblés en interpolant les valeurs adjacentes

Traitement ring artefact pré/post Les artefacts circulaires sont des artefacts de reconstruction facile à reconnaître, formant des cercles concentriques autour de l'axe de rotation dans le volume. Ils sont causés par des pixels avec des valeurs non cohérentes, dont l'erreur est propagée circulairement pendant la reconstruction.

La méthode la plus simple et efficace pour empêcher cette erreur et de prétraiter les données comme vu précédemment, mais il se peut que le prétraitement ne corrige pas tous les problèmes. Il est possible de réduire l'intensité des artefacts circulaires en post-reconstruction en traitant les couches horizontales du volume [71]-[74].

De nombreuses méthodes de traitement existent, la plupart se basant sur une méthode simplifiant le traitement : en passant les images en coordonnées polaires autour du centre de rotation, les artefacts circulaires deviennent alors des lignes droites. Il est ensuite plus facile de détecter ces lignes de différente manière, puis de les effacer autant que possible sans compromettre la qualité de l'image.

Estimation de l'axe de rotation Pour pouvoir reconstruire correctement un volume à partir de ses projections, les données géométriques de l'acquisition sont obligatoires. Elles comprennent notamment la position de l'axe de rotation et du détecteur par rapport à la source. Il est possible de calculer la position de l'axe de rotation sur l'installation expérimentale, mais également de la déduire à partir des projections capturées.

Dans le programme de reconstruction, la position de l'axe de rotation est définie au niveau du détecteur, il faut donc trouver la position de la projection de l'axe de rotation sur le détecteur.

Les différentes méthodes existantes reposent sur le fait que lorsque l'on capture des projections en tournant régulièrement autour de l'échantillon à un multiple de 180 degrés, chaque point de celui-ci sur chaque projection se retrouve sur une autre projection en symétrique par rapport à l'axe de rotation [75]. On doit donc retrouver le même poids de chaque côté de l'axe de rotation en comptabilisant toutes les projections.

La méthode que j'ai implémentée consiste à segmenter la somme des projections entre le fond (0) et l'objet (1), puis à calculer le centre de masse horizontal de l'image segmentée.

3.4 Conclusion

Dans ce chapitre, nous avons vu les objectifs et contraintes du projet XPulse dans lequel ma thèse s'inscrit. Mes travaux ont évolué de concert avec l'avancement du projet, avec l'utilisation de nouveaux matériels et de nouvelles techniques d'imagerie durant les trois ans de ma thèse, qui se plaçait sur les trois dernières années sur les cinq ans de financement initial du projet.

Mes objectifs dans ce projet sont donc de mettre en place les solutions techniques nécessaires pour traiter les données brutes issues des différents systèmes d'acquisition et d'arriver jusqu'à un tomogramme. J'ai été amené à travailler avec les membres de mon équipe pour que je puisse m'appuyer sur leurs travaux et inversement durant la durée du projet.

Les deux principaux sujets de ma thèse auront donc été la récupération de phase et la reconstruction tomographique.

Chapitre 4

Nouvelles méthodes d'extraction de phase

Ce chapitre présente les algorithmes que j'ai mis en place et implémentés pour répondre aux besoins du projet XPulse sur le sujet de la récupération de phase.

Pour la récupération de phase en propagation libre, j'ai principalement travaillé sur l'automatisation de la récupération du ratio $\frac{\delta}{\beta}$. Un article récent montre qu'il est possible de réaliser cela à l'aide d'une couche de la reconstruction tomographique [76], mais le faible contraste de notre signal de phase nous empêche de le conserver de manière fiable après reconstruction. J'ai donc travaillé sur l'automatisation de cette technique à l'aide des projections uniquement.

Pour la récupération de phase par masque de Hartmann, la configuration très dense de notre grille de sous-faisceau implique les rend difficilement séparables et donc réduit la qualité de la mesure de leurs positions, ce qui met à mal les algorithmes d'intégration normalement utilisés avec cette technique.

Je vais donc présenter mes travaux pour répondre à ces problématiques, d'abord pour la propagation libre et ensuite pour le masque de Hartmann.

4.1 Automatisation du calcul du ratio δ/β

4.1.1 Méthode itérative basée sur la formule de Paganin

La contrainte principale à l'utilisation de l'algorithme de récupération de phase de Paganin (voir 2.3.1) est la nécessité de connaître *a priori* les caractéristiques du matériau imagé, afin de pouvoir effectuer la récupération de phase pour le matériau ciblé et son milieu. Un ratio optimal dépend du matériau, de son milieu, et de l'énergie utilisée.

Ainsi, même si le contraste de phase permet en théorie de séparer deux matériaux différents, cette contrainte limite les cas d'applications.

On peut néanmoins remarquer que l'information du matériau est bien présente dans l'image capturée : les interférences formées lors de la propagation à la frontière entre deux matériaux prennent une forme dépendante des caractéristiques de ceux-ci : plus le

ratio $\frac{\delta}{\beta}$ de deux matériaux est différent, plus l'interférence entre ceux-ci sera importante. À l'inverse, une image dont la récupération de phase est faite avec un ratio correct ne contiendra plus de pic d'interférence pour le matériau résolu (figure 4.1).

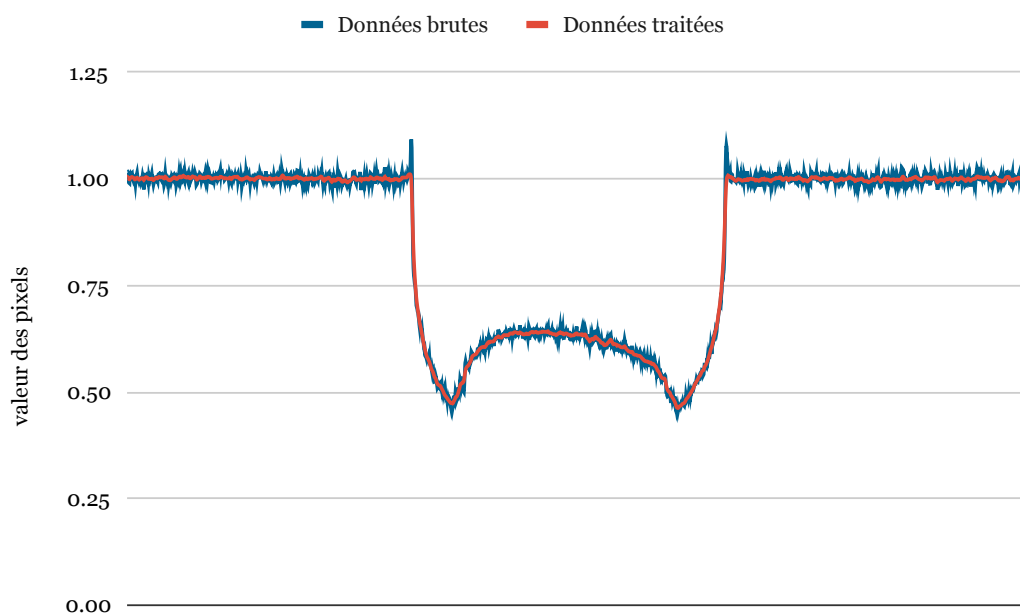


FIGURE 4.1 – Exemple du profil avant (en bleu) et après (en rouge) application de la formule de Paganin à une image acquise en contraste de phase. On remarque que les pics d'interférence sont bien effacés par ce filtre.

On peut donc tenter, à partir des pics de contraste de phase, d'approximer le ratio $\frac{\delta}{\beta}$ optimal pour une interface donnée. Une manière simple de trouver ce ratio est d'établir une condition permettant de dire si le ratio sur l'image actuelle est trop faible ou pas. Avec cette condition, il est alors facile de rechercher de manière itérative le ratio correct pour une interface donnée.

Cette méthode fonctionne très bien lorsque les pics de contrastes de phase sont suffisamment définis pour être conservés durant la reconstruction tomographique, car si elle est appliquée sur la couche d'un volume alors l'épaisseur de l'image traitée est constante, connue et faible (égale à la taille d'un voxel).

Dans notre cas, les pics ne sont pas assez bien définis (généralement un à trois pixels, avec une différence de valeur de 5 à 10% entre la valeur haute du pic et le fond de l'image) pour que la reconstruction tomographique puisse les conserver de manière fiable. Nous avons donc travaillé à automatiser cette méthode lorsqu'elle est appliquée sur une projection.

4.1.2 Condition d'arrêt

Le filtre de Paganin peut être simplifié par la formule suivante, avec $r = \frac{\delta}{\beta}$ le ratio à trouver pour une interface entre deux matériaux, I_d l'intensité mesurée au niveau du détecteur, et F le filtre fréquentiel de Paganin, équivalent à un filtre passe-bas.

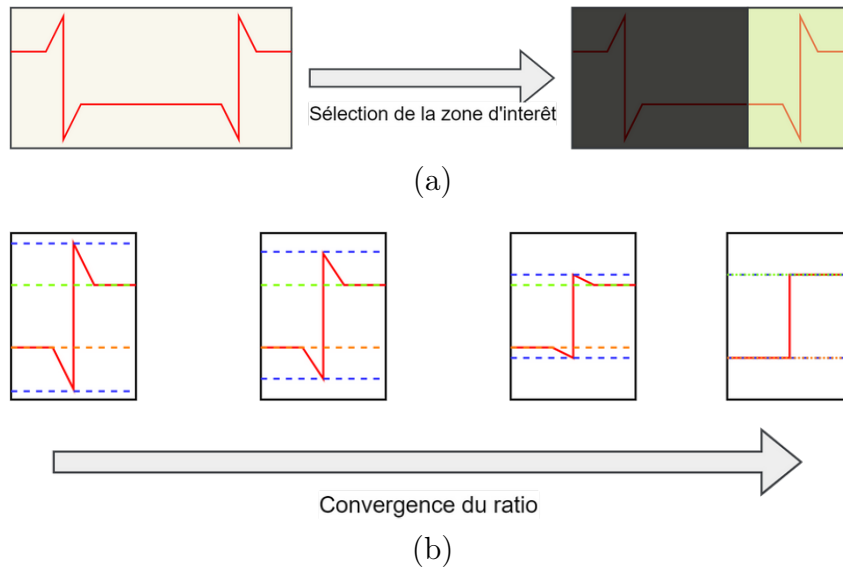


FIGURE 4.2 – Représentation des deux étapes de l’algorithme de convergence du ratio par la formule de Paganin. (a) On sélectionne une zone d’intérêt présentant une interférence de contraste de phase. (b) On fait converger le ratio pour trouver une valeur cohérente vis-à-vis du pic d’interférence.

$$\phi_o = \frac{r}{2} \ln(F(I_d, r)) \quad (4.1)$$

La formule de Paganin nous donne la phase du front d’onde au niveau de l’objet, celle-ci ne comprend donc pas d’interférences dues à la propagation si elle est récupérée correctement, c’est-à-dire avec le ratio correct. Néanmoins, un ratio trop élevé va bien permettre de récupérer une phase sans interférences, mais on perdra également les hautes fréquences de l’image. On pose donc le ratio idéal pour une interface donnée comme étant le ratio le plus faible tel que les pics d’interférence ne soient plus distinguables du bruit.

On peut noter que pour trouver la valeur du ratio à partir des pics, l’étape de normalisation après l’application du filtrage n’est pas nécessaire, on pose donc $\Phi = F(I_d, r)$ qui est l’état de l’image sur laquelle on cherchera à minimiser le pic.

En récupérant les valeurs autour de l’interface, on peut calculer plusieurs métriques qui nous serviront à surveiller l’évolution du pic. On peut commencer par diviser les valeurs appartenant au fond de l’image et celles appartenant à l’objet de manière simple, en considérant que chacune de ces régions est relativement uniforme autour de l’interface. Le pic d’interférence se trouve alors entre ces deux régions et peut être affiné en ne prenant que les valeurs supérieures au niveau du fond. On peut donc calculer à partir de ces régions la valeur moyenne du pic, celle du fond et le niveau de bruit du fond comme étant l’écart-type de celui-ci.

Pour un quelconque ratio r donné, on peut donc établir si le ratio est trop faible à l’aide de la condition $Moyenne(\Phi \in P) > Moyenne(\Phi \in B) + EcartType(\Phi \in B)$ avec P la région du pic et B la région du fond.

On peut alors établir un algorithme itératif pour optimiser le ratio à l’aide de cette condition, par exemple par dichotomie, en prévoyant une borne maximale pour le ratio

au-delà de ce qu'il est possible de trouver (figure 4.2).

Cet algorithme permet de réaliser une estimation rapide du ratio pour une interface donnée, mais peut être instable si la zone sélectionnée est trop petite pour que les statistiques soient fiables. On note également que cette méthode est dépendante du niveau d'absorption de l'objet, et plus cette absorption est forte, plus le pic est difficile à isoler de l'objet, réduisant la précision de l'estimation.

4.2 Extraction de la phase par masque de Hartmann en tomographie

4.2.1 Problématique

La principale limitation de l'algorithme de Southwell et des algorithmes similaires est que le bruit présent sur la carte de déflexion aura forcément un impact, plus ou moins important, sur l'image reconstruite. Or, comme l'algorithme effectue une intégration, le bruit de hautes fréquences présent dans les déflexions se transforme alors en bruit de basses et moyennes fréquences, qui forme des variations d'intensité sur l'image visuellement proche d'un bruit de Perlin.

Même s'il est possible d'améliorer le processus itératif en lui-même [56]-[58], un problème subsiste : chaque carte de déflexion est bruitée et présente des aberrations qui lui sont propres. Le processus itératif aura donc tendance à faire diverger les cartes de phase entre elles, ce qui pose problème lors de la reconstruction tomographique, car chaque angle de projection apportera une information différente.

Pour obtenir un résultat qui soit cohérent, il faut pouvoir forcer toutes les projections à converger vers un unique résultat, au lieu de les traiter séparément comme les algorithmes existants le font.

De plus, afin d'obtenir des données correctes pour l'algorithme de récupération de phase, il faut également pouvoir extraire précisément la position des sous-faisceaux formés par la grille de Hartmann.

Bien que j'ai travaillé sur les méthodes d'extraction des sous-faisceaux, les algorithmes que je présente dans cette section ne sont pas utilisés actuellement dans le projet ou pour les exemples utilisés dans les chapitres suivants. L'entreprise *Imagine Optic* réalise elle-même ce traitement et nous fournit directement les cartes de déflexion.

4.2.2 Extraction des sous-faisceaux par décalage à la référence

Dans les capteurs Hartmann, les pentes locales du front d'onde sont extraites à partir du déplacement des sous-faisceaux individuels. Une approche classique pour obtenir une carte de déflexion est de comparer les centroïdes de l'image acquise et le *flatfield* [77]. Cela signifie que chaque sous-faisceau représente un point de donnée, et donc un pixel dans l'image finale. Pour obtenir une meilleure définition, la seule solution est d'avoir plus de sous-faisceaux, qui sont donc moins résolus et plus proches les uns des autres.

Il y a donc un équilibre entre la précision des points de mesure et le nombre de ceux-ci. Je définis L comme la taille d'un spot (en pixel) et P le pas entre deux sous-faisceaux, en

considérant une grille de N par N sous-faisceaux.

Je vais présenter dans cette section une méthode simple pour obtenir la carte de déflexion, adaptée à notre cas d'utilisation (N élevé, L et P petit.)

Dans notre cas, nous devons également tenir compte du fait que la grille n'est pas parfaite : elle ne couvre pas parfaitement le détecteur, de sorte que les bords n'apportent pas d'information. De plus, certaines parties de la grille peuvent avoir un trou manquant ou légèrement mal placé.

Pour atténuer ces problèmes, je commence par résoudre l'ensemble de la grille et non sur chaque point, en estimant d'abord la position globale de la grille pour obtenir une estimation robuste. Cette estimation est ensuite affinée à l'aide de la valeur des pixels locaux.

Pour trouver le décalage global, nous simulons des grilles avec un décalage dans l'intervalle $[0;N-1]$ dans les deux directions sur une matrice N par N avec des bords miroirs. Les endroits où un sous-faisceau doit se trouver sont représentés par des 1, et le reste de la matrice est à zéro. En multipliant cette matrice à l'image capturée, la somme de l'image résultante nous donne un score de correspondance entre la grille réelle et celle simulée, avec un score de zéro si s ne correspond pas du tout et un score égal à la somme de l'image brute si la correspondance est parfaite. Avec ce décalage global trouvé, je peux alors extraire N-par-N patches de L par L pixels comprenant chacun un faisceau unique.

Nous obtenons finalement la position exacte de la tache en calculant le centroïde de chacun des patches. Comme notre L est petit, j'utilise le centre de masse pour obtenir la position exacte du sous-faisceau [78], que l'on peut calculer comme une moyenne des pondérée :

$$x = \frac{\sum_{i=0}^L \left(\sum_{j=0}^L (P(i, j)) * i \right)}{\sum P} \quad (4.2)$$

$$y = \frac{\sum_{j=0}^L \left(\sum_{i=0}^L (P(i, j)) * j \right)}{\sum P} \quad (4.3)$$

Une fois que nous avons la position exacte du spot sur le *flatfield*, nous pouvons extraire les différentes valeurs :

- L'absorption est la somme des pixels (L par L) autour du centre du spot.
- Les pentes sont les composantes des axes X et Y du déplacement entre les points correspondants sur le *flatfield* et l'image.

Dans le cas où les points sont mal échantillonnés, les pentes extraites deviennent fortement bruitées, ce qui affecte la reconstruction tomographique. De plus, notre installation impose l'utilisation d'un *flatfield* à chaque angle de capture à cause de mouvements mécaniques qui modifient légèrement l'alignement.

4.2.3 Suivi des sous-faisceaux

Pour tenter d'améliorer l'extraction de la carte de déflexion, notamment en se séparant du besoin d'obtenir un *flatfield* à chaque angle, j'ai étudié la possibilité de traiter les sous-faisceaux individuellement tout au long de l'acquisition, au lieu de réaliser le traitement image par image.

On peut en effet partir du principe que le changement de déflexion entre deux angles de capture reste très faible tant que le pas angulaire est faible, ce qui est généralement le cas (toutes les acquisitions faites dans le cadre du projet ont un pas inférieur 1°). On peut donc partir du principe qu'un spot ne sautera jamais d'une distance supérieure au pas de la grille, il est donc possible de suivre le déplacement de chaque sous-faisceau pour retracer son trajet durant l'acquisition. Les algorithmes classiques de suivi de *features* ne sont néanmoins pas adaptés pour ce genre de problèmes, car on veut pouvoir suivre une grande quantité de détails très petits et presque identiques et proches les uns des autres. Ces algorithmes ont donc tendance à sauter d'un sous-faisceau à l'autre régulièrement. J'ai donc développé un algorithme spécifique à ce cas :

- La première étape est de détecter sur la première image chaque sous-faisceau valide (correctement résolu et entier). Pour cela, deux passes sont effectuées : la première consiste à appliquer une convolution de l'image avec un noyau approximant la forme d'un spot. Pour améliorer la détection, la forme du spot est représentée par des nombres positifs et le contour par des nombres négatifs, pénalisant les positions où des valeurs sont présentes là où le spot devrait se trouver. Cela donne une carte de score de correspondance, et en extrayant chaque point dont le score est supérieur à un seuil fixé, on extrait alors une première approximation des sous-faisceaux en nombre entier. La seconde passe consiste à faire correspondre chaque spot approximatif à une gaussienne pour en déterminer la position exacte. Cette première étape résulte donc en une liste de positions où se trouvent chaque sous-faisceau de la première image.
- Ensuite, pour chaque image, on cherche à l'aide d'une approximation de gaussienne le spot le plus proche de chaque position sur l'image précédente. On obtient donc à la fin de cet algorithme la position de chaque sous-faisceau sur chaque image.
- Pour finir, en étudiant le chemin suivi par les sous-faisceaux, on peut extraire le déplacement relatif entre les angles de capture pour déterminer comment chaque spot a été défléchi par l'objet.

Cette méthode n'a pas été utilisée pour traiter les jeux de données présentées dans cette thèse, car un des collaborateurs du projet, Imagine Optics, a réalisé ces tâches lui-même. Néanmoins, quelques tests réalisés avec cette méthode ont indiqué une très bonne précision des déflexions extraite sans avoir besoin de *flatfield*.

4.2.4 Intégration à la reconstruction tomographique

Pour pouvoir effectivement intégrer l'information de toutes les projections dans une seule boucle itérative, la méthode retenue est de fusionner les algorithmes d'intégration de la phase et de reconstruction tomographique pour que ce dernier contraigne la convergence du premier. Comme la méthode de Southwell [25] améliore l'estimation précédente à chaque itération, on peut, à chaque itération de cette méthode, effectuer une étape de reconstruction tomographique suivie d'une reprojection, de manière à ce que toutes les projections représentent un même objet.

Après avoir travaillé sur différentes méthodes pour mettre en œuvre ce principe, la méthode que j'ai implémentée est la suivante :

1. Une première estimation de la phase intégrée est calculée à l'aide d'un des algorithmes de Southwell, avec peu d'itérations.

2. Une ou plusieurs itérations de reconstruction tomographique sont effectuées avec l'estimation de la phase utilisée comme référence.
3. Le volume est projeté pour obtenir les estimations de la phase en chaque angle
4. Ces estimations sont de nouveau traitées par quelques itérations de l'intégration de Southwell.
5. On reprend à l'étape 2 avec les nouvelles estimations.

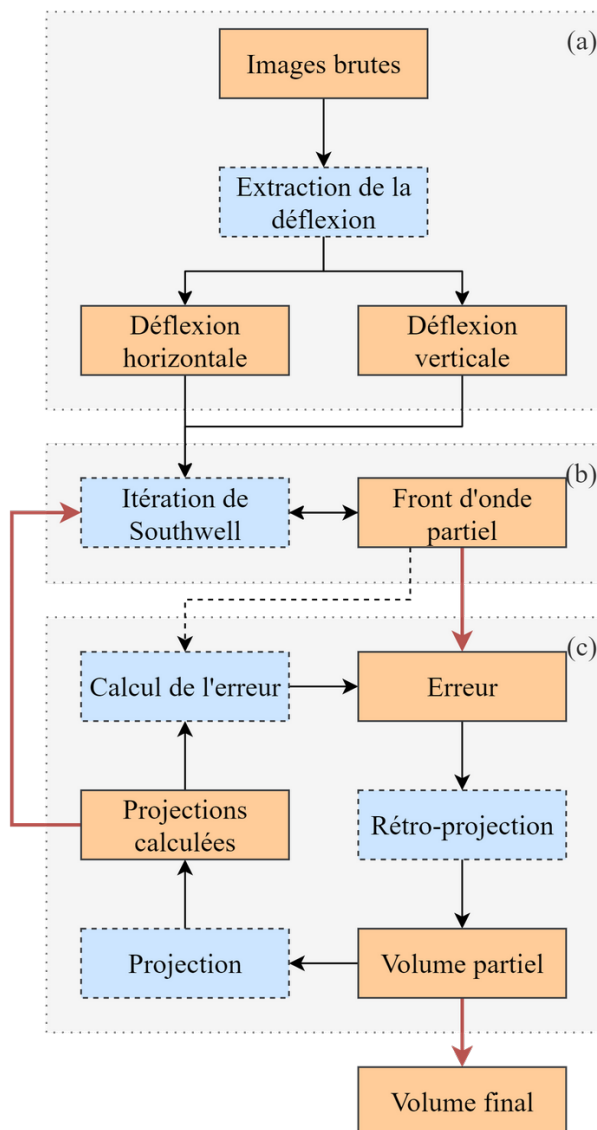


FIGURE 4.3 – Diagramme de fonctionnement de la récupération itérative de phase par masque de Hartmann en 3D.

Cet algorithme est présenté avec la figure 4.3. En intégrant la récupération de la phase directement dans l'algorithme de reconstruction tomographique, cela permet de fusionner l'information de chaque projection de manière correcte et de la redistribuer à chaque angle. De cette manière, on force tous les angles à converger vers un seul résultat représenté par le volume de phase, ce qui résout le problème d'inhomogénéité entre les projections.

Cette méthode requiert un paramètre supplémentaire, la fréquence des itérations de reconstruction et d'intégration l'une par rapport à l'autre, ainsi que le nombre successif de celles-ci. Nous verrons dans les tests réalisés l'impact que cela peut avoir.

4.3 Conclusion

Dans ce chapitre, nous avons présenté les deux algorithmes de récupération de phase qui sont utilisés dans le projet XPulse. La récupération automatique du ratio avec Paganin reste expérimentale, mais nous permet d'obtenir une estimation de la phase suffisamment correcte pour reconstruire un volume et travailler dessus.

Pour l'imagerie de phase par masque de Hartmann, l'algorithme de récupération de phase et reconstruction tomographique simultanées permet de cibler un problème important avec la méthode classique (l'inhomogénéité des projections) et de le régler de manière robuste en faisant converger tous les fronts d'onde vers un résultat unique.

Cette méthode a été publiée aux conférences *International Conference on Tomography of Materials and Structures 2022* (ICTMS) et *SPIE Medical Imaging 2023* sous la forme de deux posters.

Troisième partie
Implémentation

Chapitre 5

Problématiques d'implémentation

La reconstruction tomographique par rayons X assistée par ordinateur est une méthode répandue d'imagerie non invasive, qui nécessite des algorithmes spécialisés.

Le projet XPulse étant un projet d'imagerie médicale, le nombre d'images acquises doit être réduit autant que possible, ce qui rend obligatoire l'utilisation d'algorithmes de reconstruction itératifs, capables de maximiser la qualité de reconstruction pour une acquisition donnée.

De nombreuses implémentations existent, qui seront comparées dans ce chapitre avec une étude de l'état-de-l'art des logiciels de reconstruction tomographique itérative.

Du fait des spécifications de l'installation physique (CBCT en haute définition), les implémentations considérées comme l'état-de-l'art rencontrent des problèmes dans notre cas. Nous avons donc décidé de mettre au point notre propre implémentation prenant en compte toutes les particularités du projet : taille des données, géométrie complexe.

- Concernant la taille des données, les volumes reconstruits atteignent les 600 Go seuls (en nombre à virgule flottante 32 bits), ce qui demande une gestion de la mémoire et du stockage des données très particulière dont ne sont pas capables les implémentations testées.
- Nous devons également gérer des détecteurs segmentés non planaires en plus des différents problèmes d'alignement que nous avons rencontrés au fil du développement du système d'acquisition.
- Enfin, le système final étant destiné à une application clinique, la reconstruction doit se faire en un temps raisonnable sur des ordinateurs tout public. Il n'est pas concevable que le calcul prenne une journée entière si des dizaines d'exams peuvent être faits pendant ce temps. De même, il serait très contraignant de demander aux utilisateurs de ce système de posséder un cluster d'ordinateurs pour utiliser le programme de reconstruction. Aussi, le temps d'exécution sera une métrique critique du logiciel et devra pouvoir se comparer aux meilleures implémentations de l'état-de-l'art sur ce point.

Nous présenterons les choix d'implémentation effectués en sections 2 et 3 et nos réponses aux problèmes rencontrés avec l'état-de-l'art en section 4, 5 et 6.

Pour développer le code de la reconstruction tomographique, nous avons choisi

d'utiliser OpenCL 3.0 et non CUDA [79] comme le font les autres implémentations étudiées. En effet, si CUDA offre un accès très complet et relativement haut niveau au GPU, il est limité aux GPU NVidia exclusivement. OpenCL a l'avantage de fonctionner sur les GPU de tous les constructeurs (NVidia, AMD et Intel), et sa dernière version. OpenGL (Avec les *Compute Shaders* en version core 4.3 et plus) et Vulkan ont également été considérés, mais les limitations liées au GPGPU du premier et la relative nouveauté du second nous ont fait nous tourner vers OpenCL. Une version de l'algorithme a tout de même été développée avec Vulkan en vu d'un éventuel changement d'API, car les constructeurs semblent désormais se tourner vers cette technologie plutôt que OpenCL pour le futur.

5.1 Projection

5.1.1 Opérateurs de projection

Les opérateurs de projection et de rétroprojection sont une composante commune à tout algorithme de reconstruction. Ils déterminent la relation entre les voxels et les pixels pour calculer respectivement la projection du volume actuel ou pour rétro-propager l'erreur calculée. Il existe plusieurs méthodes pour réaliser ces opérateurs, répartis en deux familles : celles itérant sur les pixels et celles sur les voxels (figure 5.1). Ces méthodes sont très similaires en termes de complexité de calcul, mais présentent des particularités qui peuvent encourager l'utilisation de l'une ou l'autre selon le cas d'utilisation.

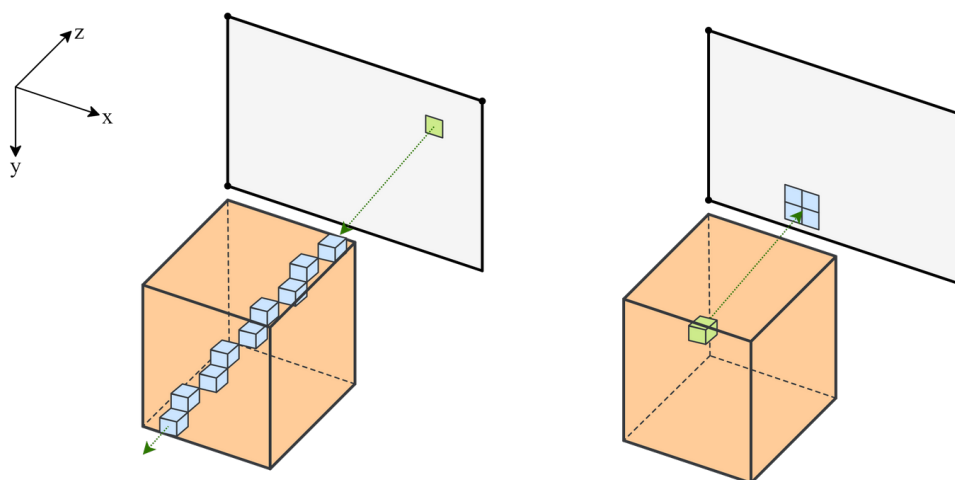


FIGURE 5.1 – Différence entre la projection par pixel (à gauche) et par voxel (à droite).

Dans un environnement fortement parallèle, comme sur GPU, la meilleure pratique consiste à itérer sur les éléments en cours d'écriture afin d'éviter les problèmes de concurrence (si plusieurs unités de calcul écrivent au même endroit au même moment, le résultat est généralement non déterminé). Ainsi le meilleur choix serait alors l'itération sur les pixels pour la projection et l'itération sur les voxels pour la rétroprojection. Il y a néanmoins une contrainte sur la parallélisation des méthodes qui itèrent à travers les pixels et traversent le volume : chaque pixel représentera une charge de calcul variable en fonction du nombre de voxels traversés. Cela ralentit fortement le calcul étant donné que pour une unité de calcul SIMD, le temps d'exécution total sera égal au temps maximum de calcul de chaque donnée. Dans le pire cas (avec un angle facteur de 45 degrés), les pixels centraux traverseront environ $\sqrt{2}h$ voxels (avec h le nombre de colonnes de pixels du détecteur), tandis que les

pixels des bords peuvent ne traverser qu'une fraction de cette valeur, potentiellement zéros selon la géométrie.

Nous optons donc pour une approche basée sur l'itération par voxels pour la projection et la rétroprojection. De cette manière, la charge de chaque noyau de calcul est prédéterminée et cela permet d'éliminer la quasi-totalité des branchements, ce qui est souhaitable pour un calcul sur GPU. L'inconvénient de ce choix est que l'itération à travers les voxels lors de la projection entraînera une potentielle concurrence dans les opérations d'écriture, nécessitant donc une opération d'addition atomique qui pourrait ralentir le calcul. Nous verrons par la suite que l'utilisation d'une opération d'addition atomique dans notre cas n'a qu'un impact marginal sur les performances pour un GPU moderne.

Dans cette configuration, la relation entre un voxel et un pixel se résume donc à la projection d'un point sur un plan.

On peut noter qu'en imagerie par contraste de phase, on considère que les rayons sont réfracté par l'objet, et la projection ne se fait donc plus par une ligne droite. Néanmoins, cette réfraction est très faible pour les cas sur lesquels nous travaillons (Déplacement sur le détecteur inférieur à un pixel), nous pouvons donc l'ignorer pour la reconstruction tomographique.

5.1.2 Projection entre une source et un détecteur

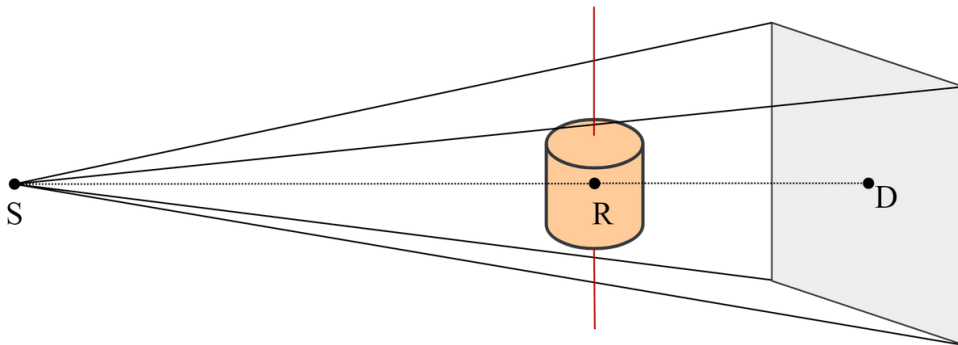


FIGURE 5.2 – Schéma d'une géométrie d'acquisition conique simple, avec une source ponctuelle (S), un détecteur (D), et un objet placé sur l'axe de rotation (R).

Une géométrie standard de CBCT est illustrée en figure 5.2, avec une source, un détecteur et un objet entre les deux. Le but de la projection est de représenter de manière précise la position des différentes parties et d'intégrer les valeurs entre la source et le détecteur pour chaque pixel de ce dernier, comme présenté dans l'équation suivante :

$$D(x, y) = \int_0^z \ell(\vec{r}_l) dl \quad (5.1)$$

Avec z la distance source-détecteur, $D(x, y)$ le pixel sur le détecteur à la position (x, y) , \vec{r}_l la position le long de la ligne directrice $S - D(x, y)$ à la distance l , et ℓ la valeur du volume en un point.

Projection par rayon La projection par rayon prend généralement la forme d'un suivi de chemin entre le pixel du détecteur traité et la source, en passant par la zone

de reconstruction voxel par voxel. Il existe de nombreux algorithmes de tracé de lignes, notamment l'algorithme de Bresenham [80], mais il peut aussi prendre la forme d'un lancer de rayon à l'aide d'une structure accélératrice. Dans les deux cas, on retrouve la liste des voxels traversés par les rayons partant de la source et atteignant le pixel ciblé. On peut ensuite calculer un poids pour chaque voxel dépendant de la distance entre le centre du voxel et le rayon.

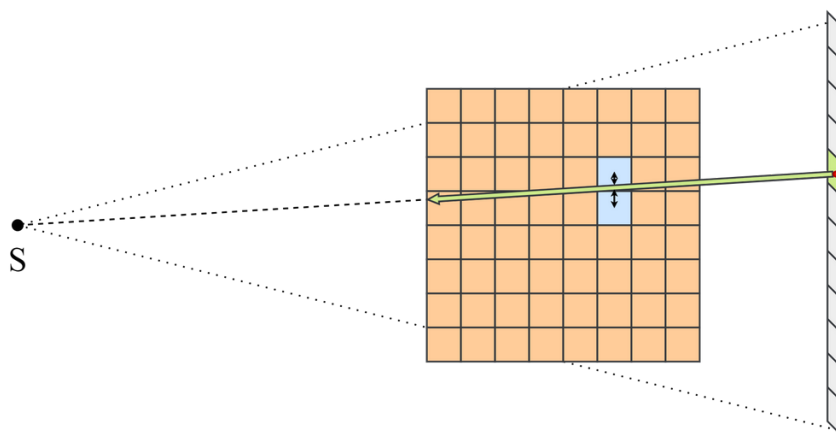


FIGURE 5.3 – Schéma de la projection d'un pixel du détecteur (en vert) sur les voxels présents sur le chemin entre lui et la source.

Projection par distance La projection par distance [81], [82] est une amélioration de la méthode par rayon offrant une meilleure estimation des poids permettant de minimiser les problèmes d'artefacts de discrétisation. En projetant les bords des pixels et des voxels sur un axe, il est possible de calculer précisément la contribution de chacun sur celui-ci. La complexité est légèrement supérieure à l'approche par rayon.

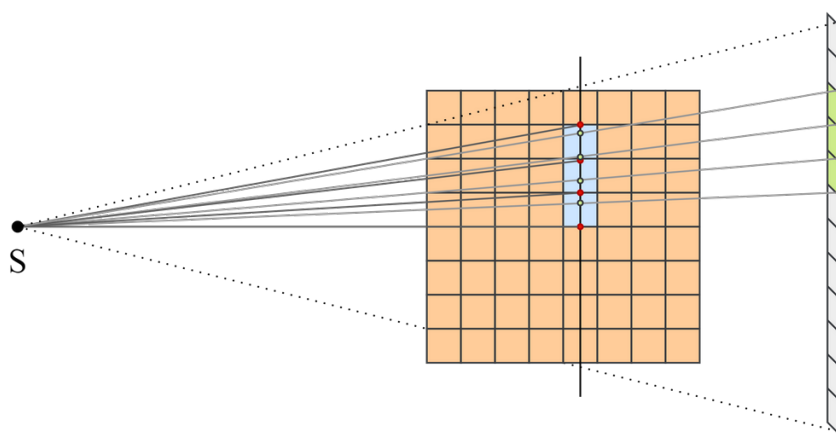


FIGURE 5.4 – Schéma de la projection par distance de pixels du détecteur. Les positions de la projection sur la ligne verticale sont utilisées pour calculer le poids du pixel sur le voxel traversé.

Projection par pixel (Lancé de rayon) La projection par pixel consiste à itérer sur les voxels et à calculer pour chacun la position de celui-ci une fois projeté sur le détecteur. Pour cela il suffit de calculer l'intersection entre un rayon partant de la source et allant

en direction du voxel actuel et le plan du détecteur. Pour cela il est par exemple possible d'utiliser l'algorithme de Moller-Trumbore [83] ou autre [84]. Les voxels traversés peuvent ensuite être pondérés par différentes méthodes de calcul de poids pour éviter les problèmes d'échantillonnage.

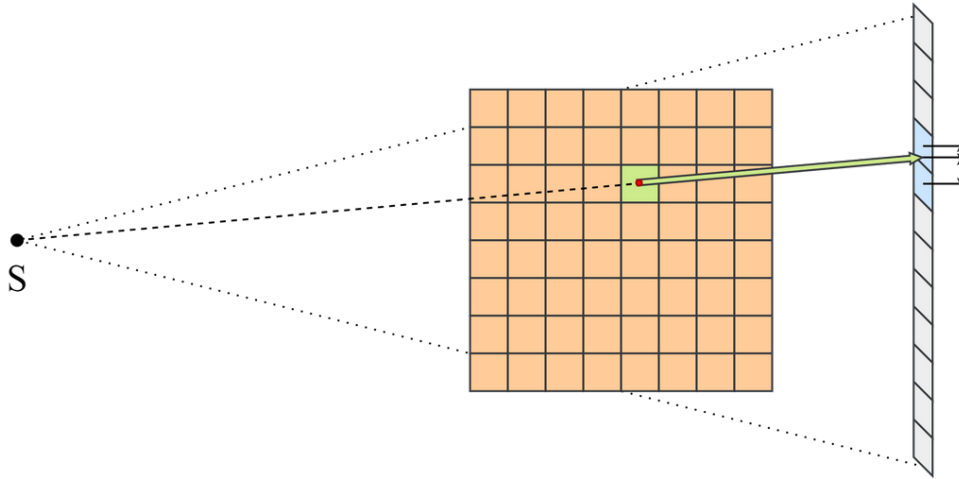


FIGURE 5.5 – Schéma de la projection d'un voxel (en vert) sur les pixels d'un détecteur (en bleu).

Projection par pixel (matrice de projection) Il est également possible de précalculer la géométrie de projection faite par le lancer de rayon dans une matrice de projection, ce qui permet d'en réduire grandement la complexité durant l'exécution.

Pour ce faire, il faut représenter la géométrie de manière légèrement différente. La zone de reconstruction est de taille finie par construction, il est donc possible de déterminer une distance n telle qu'aucune information n'est présente entre la source et le plan orthogonal à l'axe de projection à une distance n de celle-ci :

$$\int_0^n \ell(\vec{r}_l) dl = 0 \quad (5.2)$$

On peut ensuite séparer l'intégrale en deux parties. La première étant nulle, la seconde partie suffit donc à représenter la projection. Cette formule représente un frustum (Volume géométrique représentant le champ de vision d'une caméra, illustré en figure 5.6).

$$P(x, y) = \int_0^N \ell(\vec{r}_l) dl + \int_N^D \ell(\vec{r}) dl \quad (5.3)$$

$$P(x, y) = \int_N^D \ell(\vec{r}_l) dl = C(x', y') \quad (5.4)$$

La projection d'un point dans un frustum peut être effectuée à l'aide d'une matrice de projection de manière analogue à ce qui est fait dans les applications de 3D en temps réel [85]. Une matrice de projection 4 par 4 est donc précalculée pour chaque angle de projection avant la reconstruction. Dans cette configuration la matrice de projection représente une géométrie d'acquisition conique, mais il est également possible de représenter

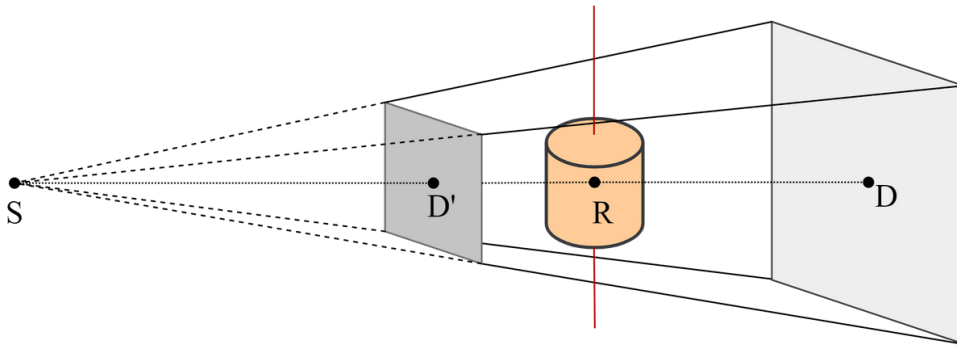


FIGURE 5.6 – Représentation de la géométrie d'acquisition à l'aide d'un frustum, avec D le plan lointain et D' le plan proche.

une géométrie parallèle à l'aide d'une matrice de projection orthogonale. La matrice de géométrie conique est plus complexe à obtenir, mais une fois calculée, la complexité du calcul de la projection est la même, quel que soit le contenu de la matrice. Une telle matrice peut également représenter un frustum oblique [86] et ainsi émuler toute géométrie définie par une source ponctuelle et un détecteur planaire.

La construction de la géométrie et de cette matrice est détaillée dans les sections suivantes.

5.1.3 Construction de la géométrie

Avant de pouvoir calculer la matrice de projection, il faut d'abord pouvoir formuler la géométrie du système d'acquisition utilisé. Pour cela, les valeurs suivantes doivent être connues : la distance source-objet (SO), la distance source-détecteur (SD), le décalage horizontal de la projection de l'axe de rotation par rapport au milieu détecteur (RX), le décalage par rapport au centre du détecteur de la projection orthogonale de la source (SX, SY), et pour une acquisition hélicoïdale, le pas vertical entre deux prises d'image.

Les points nécessaires à la construction de la matrice de projection sont la position relative de la source (S), le centre de la zone à reconstruire (O) et trois coins du détecteur, en considérant que ceux-ci forment un parallélogramme (D_o, D_u, D_v).

Pour les calculer, une méthode simple consiste à tout d'abord considérer la source comme le point d'origine (0,0,0), avec \vec{x} et \vec{y} les axes du plan du détecteur horizontal et vertical, et \vec{z} la direction de projection partant de la source et allant vers le détecteur.

Avec cette configuration, on peut donc placer la source (S), et calculer la position orthogonale de celle-ci sur le détecteur (P_s) :

$$S = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (5.5)$$

$$P_s = S + \begin{bmatrix} 0 \\ 0 \\ SD \end{bmatrix} \quad (5.6)$$

Le centre du détecteur (D) est décalé de SX, SY (décalage de la projection de la source par rapport au centre du détecteur) par construction :

$$D = P_s - \begin{bmatrix} SX \\ SY \\ 0 \end{bmatrix} \quad (5.7)$$

On peut ensuite déduire la position de l'axe de rotation avec le décalage en P_s et le centre du détecteur, et entre le centre du détecteur et la projection de l'axe de rotation (Pr) :

$$P_r = D + \begin{bmatrix} RX \\ 0 \\ 0 \end{bmatrix} \quad (5.8)$$

Ces deux points nous permettent d'obtenir le vecteur unitaire représentant la direction source-axe de rotation \vec{SR} :

$$\vec{SR} = \frac{P_r - S}{\|P_r - S\|} \quad (5.9)$$

Et on peut calculer la position centrale du volume à reconstruire, à une distance SO de la source dans la direction SR :

$$O = S + \vec{SR} \times SO \quad (5.10)$$

Nous avons donc désormais la position relative de la source, de l'objet et du détecteur. Pour simplifier par la suite la rotation tomographique, on peut passer dans un repère centré sur l'axe de rotation pour que celle-ci se fasse autour de l'origine. Ensuite, on peut simplement appliquer une rotation à S et D (R étant l'origine, il n'y a pas besoin de le traiter).

$$R^\theta = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \quad (5.11)$$

$$S^\theta = S * R^\theta \quad (5.12)$$

$$P_s^\theta = P_s * R^\theta \quad (5.13)$$

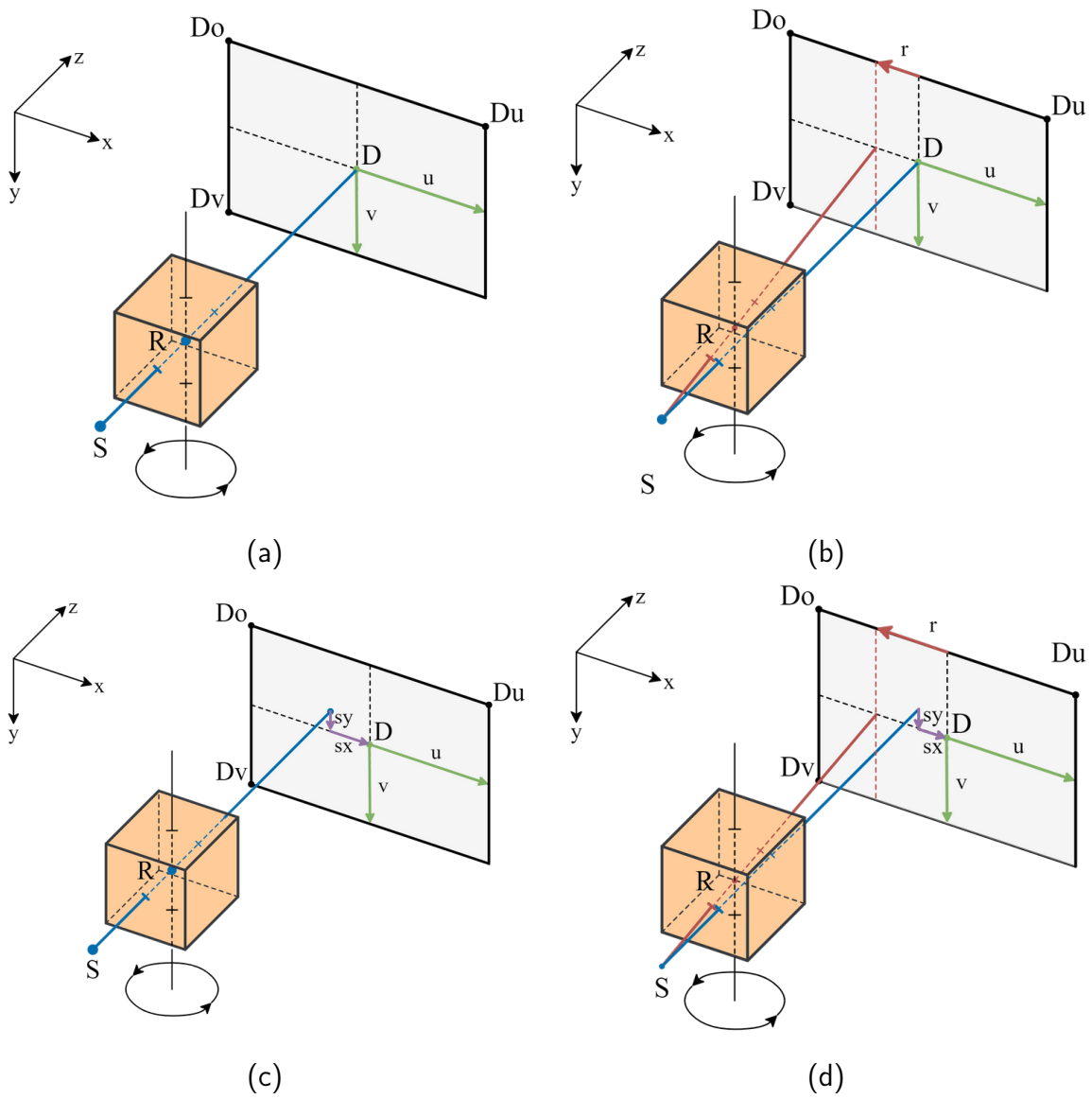


TABLE 5.1 – Schéma de la représentation de notre géométrie. (a) Système aligné (b) Correction de la position de l'axe de rotation (c) Correction de la position de la source (d) Correction de la source et de l'axe de rotation.

Finalement, on peut déterminer la position des coins du détecteur en trouvant le repère de coordonnées 2D local à celui-ci, sur un plan orthogonal à $S - S_d$:

$$\vec{j} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (5.14)$$

$$\vec{i} = j \times \langle P_s - S \rangle \quad (5.15)$$

Avec $\langle \vec{a} \rangle = \frac{\vec{a}}{\|\vec{a}\|}$. Le centre du détecteur est alors :

$$D = P_s - sx \times \vec{i} - sy \times \vec{j} \quad (5.16)$$

Et ses coins :

$$D_o = D - \vec{i} \times 0.5D_w - \vec{j} \times 0.5D_h \quad (5.17)$$

$$D_u = D_o + \vec{i} \times 0.5D_w \quad (5.18)$$

$$D_v = D_o + \vec{j} \times 0.5D_h \quad (5.19)$$

Avec D_h et D_w la hauteur et la largeur du détecteur. Ces trois points peuvent ensuite être tournés autour de la projection de la source pour corriger d'éventuels angles. La figure 5.1 illustre la construction de la géométrie à partir des paramètres présentés ici.

5.1.4 Construction de la matrice de projection

Avec la position de la source (S), la projection de la source sur le détecteur (P_s), la position de l'axe de rotation (R), et trois coins du détecteur (D_o, D_u, D_v), on peut construire la matrice de projection.

S, D_o, D_u , et D_v représentent exactement le frustum décrivant notre géométrie d'acquisition à chaque pas angulaire du scanner. Nous pouvons construire la matrice de projection comme indiqué dans [87]. Dans notre cas, et en considérant un système matriciel *row-major* (ligne par ligne), elle peut être décrite de la façon suivante :

$$M = P * D * T \quad (5.20)$$

Avec :

$$P = \left\{ \begin{array}{cccc} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{array} \right\} \quad (5.21)$$

- $\vec{v} = \langle D_v - D_o \rangle$, $\vec{u} = \langle D_u - D_o \rangle$ le vecteur unitaire de l'espace du détecteur,
- $\vec{n} = \langle \vec{u} \times \vec{v} \rangle$ la normale au détecteur
- SD_o, SD_u, SD_v les vecteurs de S vers respectivement D_o, D_u, D_v .
- $d = -(\vec{n} \cdot SD_o)$, qui devrait être égale à la distance source-détecteur
- n et f sont les distances les plus proches et les plus éloignées des limites du volume par rapport à la source.
- r, l, t, b sont respectivement les distances des bords du détecteur par rapport à la projection de la source : droite, gauche, haut et bas : $r = (\vec{u} \cdot S\vec{D}_v) \frac{n}{d}$, $l = (\vec{u} \cdot S\vec{D}_o) \frac{n}{d}$, $b = (\vec{v} \cdot S\vec{D}_o) \frac{n}{d}$, $t = (\vec{v} \cdot S\vec{D}_u) \frac{n}{d}$

Les matrices M et T gèrent respectivement la direction et la position du détecteur et de la source.

$$D = \begin{pmatrix} \text{---} & \vec{u} & \text{---} & 0 \\ \text{---} & \vec{v} & \text{---} & 0 \\ \text{---} & \vec{n} & \text{---} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.22)$$

$$T = \begin{pmatrix} 1 & 0 & 0 & | \\ 0 & 1 & 0 & -S \\ 0 & 0 & 1 & | \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.23)$$

Il est enfin possible d'ajouter une matrice de fenêtrage pour éviter d'avoir à dénormaliser les coordonnées à chaque projection.

$$V = \begin{pmatrix} \frac{r'-l'}{2} & 0 & 0 & \frac{r'+l'}{2} \\ 0 & \frac{t'-b'}{2} & 0 & \frac{t'+b'}{2} \\ 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.24)$$

Avec l', r' et t', b' coordonnées minimales et maximales, horizontales et verticales. En multipliant la matrice M par V , le résultat obtenu par la projection ne sera plus normalisé, mais représentera la coordonnée en pixel sur le détecteur directement.

Cette matrice doit être calculée pour chaque angle de projection, nous noterons M^θ la matrice de projection pour l'angle θ .

5.1.5 Projection d'un point par une matrice de projection

Pour calculer les coordonnées 2D normalisées sur le détecteur à l'angle θ d'un point 3D, il suffit d'effectuer la multiplication suivante :

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_{world} \times M^\theta = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}_{camera} \quad (5.25)$$

$$\begin{bmatrix} x \\ y \end{bmatrix}_{camera} \times \frac{1}{w_{camera}} = \begin{bmatrix} x \\ y \end{bmatrix}_{\acute{e}cran} \quad (5.26)$$

Avec \times étant la multiplication matrice-matrice ou vecteur-matrice. La multiplication d'un vecteur 4 par une matrice 4x4 est une opération hautement optimisée sur les GPU modernes, rendant ce calcul très efficace [88].

Maintenant, avec la matrice calculée précédemment, nous pouvons calculer les coordonnées de la projection d'un voxel à l'angle du détecteur θ :

$$\vec{proj} = \vec{voxel} * M^\theta \quad (5.27)$$

Puisque nous n'avons pas besoin de la composante z de la matrice, nous pouvons seulement calculer les lignes nécessaires avec un produit scalaire.

$$\vec{xy} = \frac{\vec{voxel}_{xy} \cdot M_{xy}^\theta}{\vec{voxel}_z \cdot M_z^\theta} \quad (5.28)$$

Si la matrice de projection n'a pas été fenêtrée, il faut ensuite dé-normaliser les coordonnées :

$$\vec{ij} = \begin{cases} (x * 0.5 + 0.5) * w \\ (y * 0.5 + 0.5) * h \end{cases} \quad (5.29)$$

5.1.6 Multi-détecteur

Le détecteur du dernier prototype du système d'acquisition XPulse présente une particularité importante pour la reconstruction tomographique : il est composé de cinq modules segmentés en huit sous-modules chacun (figure 5.7). Les modules sont placés horizontalement en arc de cercle de manière à obtenir une zone de détection de 5120 par 128 pixels dont la courbure correspond à la conicité de la source. Les images obtenues présentent les modules côte à côte de manière analogue à leur placement physique.

Pour correctement reconstruire de telles acquisitions, le système de création des matrices de projection a été étendu pour prendre en compte des détecteurs segmentés.

Lors de la création de la matrice pour une image donnée, les coins du détecteur sont redéfinis pour chaque module et la matrice de projection correspondante est construite en conservant la position de la source existante.

Chaque matrice de projection est donc désormais liée à une matrice de projection et un *viewport* correspondant à sa zone sur le détecteur.

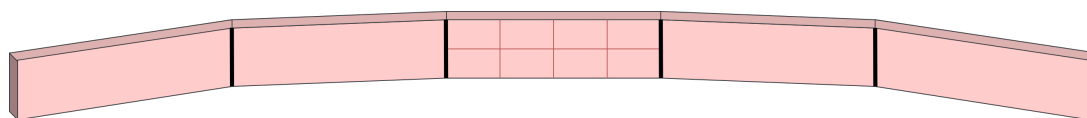


FIGURE 5.7 – Schéma de l'arrangement des modules du détecteur Direct Conversion courbe, composé de 5 modules arrangés en arc de cercle.

Cette méthode est relativement simple à mettre en place, mais pose tout de même un souci : le temps de calcul est multiplié par le nombre de modules par détecteur.

Pour pallier ce problème, une solution simple peut être mise en place : en approximant la projection sur un détecteur planaire, il est possible de faire correspondre les coordonnées sur ce détecteur virtuel à un des cinq modules. Ce faisant, la projection ne demande que deux calculs de coordonnées, quel que soit le nombre de modules.

Dans les faits, comme le calcul de la projection des voxels ne représente qu'une partie du processus de reconstruction, passer de un à cinq modules multiplie le temps total de reconstruction par environ 1.5. La méthode du détecteur planaire virtuel n'est donc pas nécessaire si le nombre de modules est relativement faible.

5.2 Gestion de la mémoire

5.2.1 Problématique gestion mémoire

La première méthode de reconstruction tomographique que j'ai implémentée était réalisée à l'aide d'opérateur de matrice uniquement, à l'aide de la bibliothèque Array-Fire [89]. Si les performances d'une telle implémentation se sont montrées correctes, cette méthode a un inconvénient majeur : la quantité de mémoire nécessaire est très élevée, demandant plusieurs fois la taille du volume en mémoire pour pouvoir fonctionner. J'ai donc rapidement opté pour une implémentation avec une technologie plus adéquate pour notre problématique.

La reconstruction tomographique 3D impose de travailler avec une matrice 3D pleine. La mémoire nécessaire augmente donc fortement avec la définition du détecteur utilisé et cela impose donc une gestion précise de la mémoire utilisée et des transferts de données.

Comme l'implémentation est faite sur GPU, un niveau supplémentaire d'espace mémoire doit être géré : les données sont initialement stockées sur le disque dur de l'ordinateur, et elles devront ensuite être transférées à la RAM, puis à la VRAM, avant de pouvoir être dispatchées de manière plus ou moins transparente aux différents caches accessibles des cœurs d'exécution du GPU (figure 5.8).

Le but de mon implémentation est de pouvoir gérer des volumes de données très importants (au moins dix fois la taille de la VRAM et de la RAM), tout en préservant au maximum les performances du temps d'exécution de la reconstruction. Pour cela, il aura fallu itérer à plusieurs reprises sur différentes méthodes permettant cette reconstruction pour évaluer les performances de chacune. Je présenterai ici les deux méthodes principalement utilisées, puis la méthode finalement choisie pour le projet ainsi que les optimisations qui y ont été ensuite apportées.

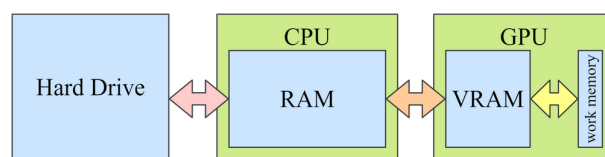


FIGURE 5.8 – Représentation des différents espaces mémoire utilisables pour un programme s’exécutant sur un GPU. Les différents caches ne sont représentés.

5.2.2 Implémentation naïve

Pour établir une référence pour nos optimisations, nous avons développé un algorithme de reconstruction naïf qui stocke toutes les données dans la VRAM, ne nécessitant donc aucun transfert de données pendant le calcul. Cela nous permettra de mesurer et de comparer la différence de temps de traitement, d’espace mémoire requis et de charge GPU entre une version idéale de la reconstruction et celle que nous avons développée et qui réduit la consommation de mémoire GPU.

Sur GPU, pour un opérateur de projection donné, l’algorithme de reconstruction le plus rapide serait celui qui obtient une utilisation du GPU de 100 % pendant toute la durée de la reconstruction. Cela signifie que toutes les données dont le projecteur a besoin doivent être disponibles au moment où il termine sa tâche de calcul précédente.

Pour notre implémentation naïve, les données sont lues sur le disque dur au début du processus, puis directement transférées au GPU qui les utilise pour calculer le volume avec toutes les itérations nécessaires. Pour finir, le volume est renvoyé dans la RAM puis écrit sur le disque dur. Cet algorithme est représenté par un schéma dans la figure 5.9. Le profilage du GPU 5.10 illustre bien ce comportement : le calcul sur le GPU est ininterrompu pendant toute sa durée. Pour le jeu de données que nous utilisons, cela représente 97.6% d’utilisation du GPU. En utilisant cette mesure de performance comme référence, le but de notre implémentation sera de minimiser l’utilisation de la mémoire GPU tout en maximisant l’utilisation du GPU pour rester au plus proche de ce chiffre.

Nous pouvons également noter que même en tenant compte du coût de l’opération atomique, la projection avant est plus rapide que la rétroprojection, même si elles ont la même complexité théorique ($O(n * v)$, avec n le nombre d’angles et v le nombre de voxels). Cela s’explique par le faible nombre de collisions (c’est-à-dire d’opérations d’addition simultanées au même emplacement mémoire) qui devraient se produire, combiné avec des opérations légèrement moins coûteuses (principalement des additions contre des multiplications et divisions).

5.2.3 Minimiser l’utilisation mémoire

La manière la plus directe de reconstruire un volume qui ne tient pas dans la VRAM est de diviser le volume en sous-volumes (SV) plus petits pour que chaque appel à un opérateur de projection se fasse plusieurs fois sur des données réduites (figure 5.11).

Cela permet de traiter les données presque indépendamment de la taille de la VRAM, mais nécessite de nombreux transferts de données entre le GPU et la RAM, et également entre la RAM et le disque dur, si le volume est trop grand pour rentrer dans la RAM. L’optimisation des transferts de données pour la reconstruction tomographique a été

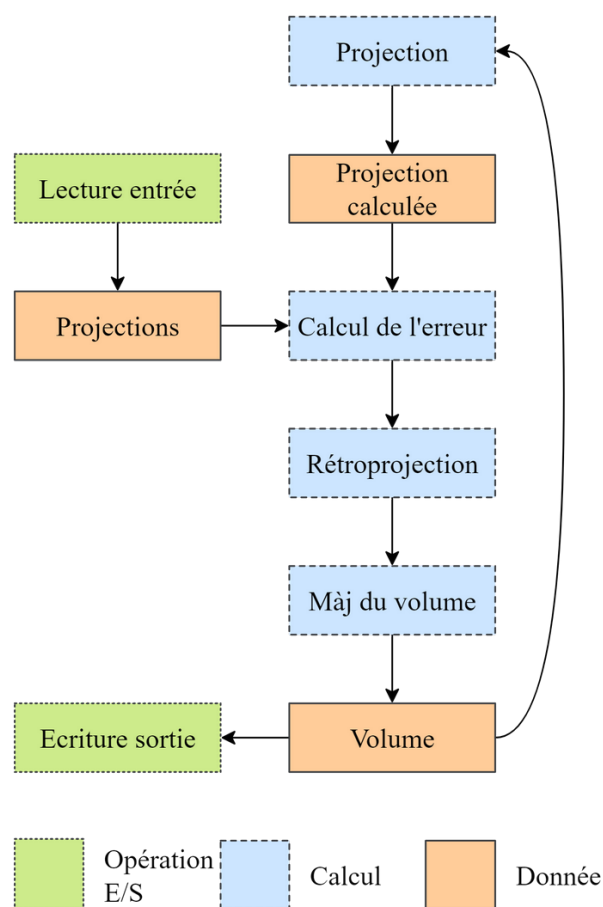


FIGURE 5.9 – Schéma de fonctionnement d'un algorithme de reconstruction tomographique simple.

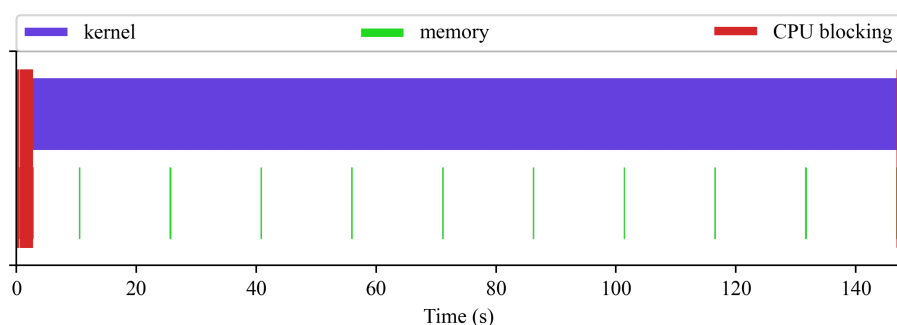


FIGURE 5.10 – Exemple de profilage relevé durant l'exécution d'une reconstruction tomographique simple. Comme toutes les données sont stockées sur la mémoire GPU, aucun temps de transfert ne vient interrompre celui-ci une fois l'exécution lancée.

discutée dans [90]. TIGRE utilise cette méthode, en stockant toutes les données requises dans la RAM et en transférant les données dans les deux sens lorsque cela est nécessaire. Les méthodes de segmentation pour une telle méthode sont discutées dans [91], mais cet article est principalement destiné aux systèmes distribués, et n'est donc pas optimal dans notre contexte (une seule machine avec un ou plusieurs GPU et un pool de mémoire partagée, par opposition aux configurations multi-machines avec des mémoires séparées et des transferts de données coûteux entre eux). Nous utilisons simplement des blocs de voxels

consécutifs dans la mémoire pour notre méthode, améliorant ainsi le taux de transfert de données.

Cette méthode fonctionne relativement bien lorsque toutes les données sont stockées dans la RAM, mais elle s'adapte mal à la taille de l'ensemble de données, car la vitesse de transfert du disque dur devient rapidement un goulot d'étranglement majeur. Dans la figure 5.12, nous pouvons voir que beaucoup de temps est passé à attendre que les données soient transférées au GPU, mettant en pause la reconstruction.

Elle peut néanmoins être grandement améliorée en tirant parti des capacités des API GPU modernes pour transférer des données entre la VRAM et la RAM tout en exécutant du code sur le GPU. Les transferts entre le disque dur et la RAM peuvent également être effectués de manière préventive pour améliorer encore l'utilisation du GPU.

Nous avons implémenté cette méthode en utilisant quatre threads CPU et trois files d'attente de commandes GPU. Le nombre de threads peut être ajusté en fonction des performances relatives du GPU et du disque dur. Cela nous permet de réduire les temps morts que le GPU passe à attendre des transferts de données, ce qui est visible sur la figure 5.13 comparé à la figure 5.12.

Chaque thread (1) lit un sous-volume du disque dur vers la RAM, (2) l'envoie à la VRAM, (3) demande l'exécution du noyau, et enfin (4) transfère le sous-volume mis à jour vers la RAM, puis (5) vers le disque dur. Les étapes (1) et (5) sont effectuées en parallèle sans synchronisation. Chacune des autres étapes, (2), (3) et (4), est envoyée à une file d'attente différente du GPU dédiée aux appels d'écriture, de lecture ou de calcul, de sorte qu'un seul thread peut envoyer une commande à une file d'attente donnée en même temps.

5.2.4 Minimiser le transfert de données

Nous avons vu que la division de chaque opérateur de projection individuellement est très efficace pour réduire l'empreinte mémoire de l'algorithme, mais qu'elle nécessite un transfert de données important entre les différents pools de mémoire, ce qui peut avoir un impact important sur le temps de calcul résultant. Pour éviter ce goulot d'étranglement de la bande passante, une autre méthode est de diviser l'ensemble de la reconstruction, et non chaque opérateur, en petits ensembles de données autosuffisants qui peuvent tenir individuellement dans la mémoire du GPU. De cette façon, l'algorithme n'a pas besoin de lire les données précédemment traitées sur le disque.

L'idée générale est de segmenter le volume et les projections en sous-ensembles comme on le ferait pour une configuration à faisceaux parallèles. En géométrie parallèle, chaque couche du volume correspond à un nombre fixe de lignes sur la projection, et vice-versa, ce qui simplifie grandement le problème : chaque couche et ses lignes de projection sont indépendantes et peuvent être reconstruites sans interagir avec les autres couches. Chaque sous-volume est alors une série de couches consécutives (division sur le vecteur \vec{y} dans notre configuration). Ceci ne peut pas être appliqué directement à la géométrie conique, car la relation entre les couches et les lignes sur la projection est plus complexe. La figure 5.14 illustre les problèmes rencontrés lors de la division du volume dans cette géométrie.

Pour tout sous-volume, les lignes de projection nécessaires à sa reconstruction peuvent être déterminées facilement. Dans notre cas, nous utilisons les mêmes matrices de projection que pour le noyau du projecteur et projetons les huit coins de la boîte englobante du

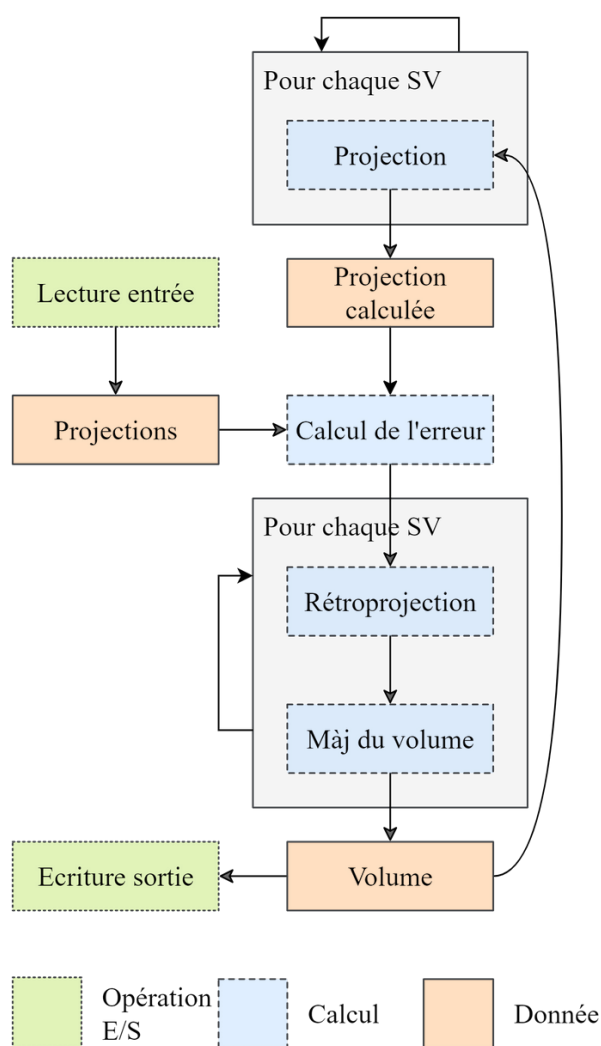


FIGURE 5.11 – Schéma de fonctionnement d'un algorithme de reconstruction tomographique avec les opérateurs de projection divisés en sous-volume.

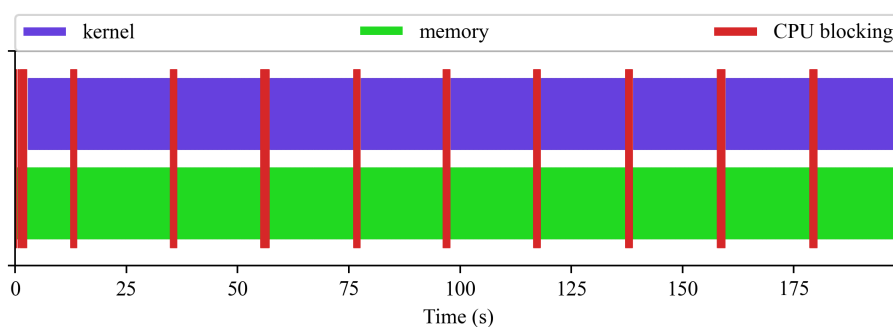


FIGURE 5.12 – Exemple de profilage relevé durant l'exécution d'une reconstruction tomographique avec les opérateurs de projection divisés en sous-volume. L'exécution de la reconstruction est mise en pause pour permettre le transfert de données de manière régulière.

sous-volume. La plage de coordonnées \vec{y} sur le détecteur indique les lignes que nous devons considérer (Figure 5.2 (a)).

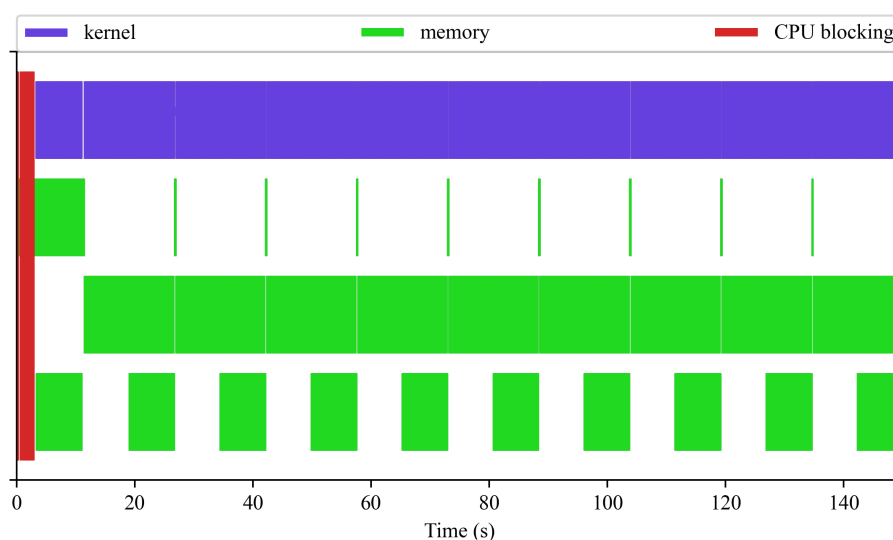


FIGURE 5.13 – Exemple de profilage relevé durant l'exécution d'une reconstruction tomographique avec les opérateurs de projection divisés en sous-volume et les transferts mémoires réalisée de manière asynchrone. L'exécution de la reconstruction n'est pas mise en pause durant les transferts, ce qui permet de les réaliser pendant ce temps.

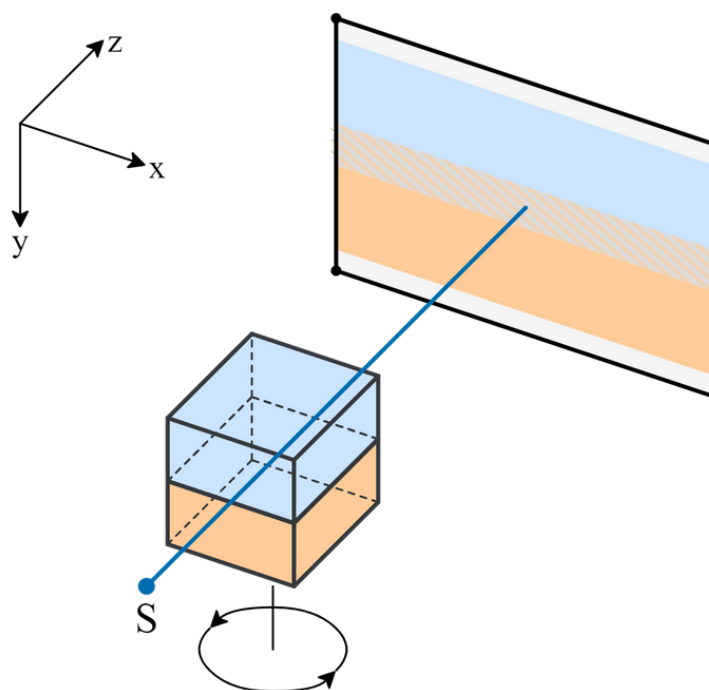


FIGURE 5.14 – Schéma représentant la zone de recouvrement sur les projections de deux sous-volumes horizontaux.

Mais maintenant, ces lignes portent également des informations qui n'appartiennent pas au sous-volume sélectionné, et les ignorer provoquerait des artefacts dans les couches supérieures et inférieures des sous-volumes. Pour répartir correctement les informations là où elles doivent être, nous devons considérer un sous-volume plus grand, dont les limites peuvent être déterminées à partir de l'étendue des lignes de la projection. (Figure 5.2 (b)).

Une fois que nous avons déterminé ce sous-volume limite et la plage de lignes sur les

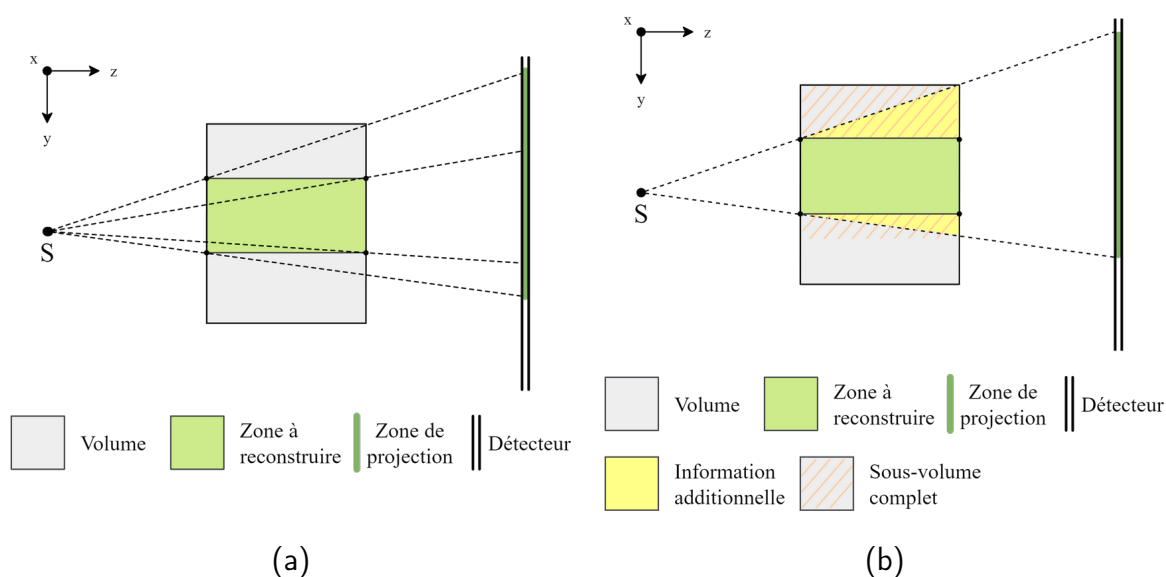


TABLE 5.2 – Schéma du découpage en sous/jeux de données de la reconstruction tomographique. (a) À partir d'une zone définie du volume de reconstruction, on détermine la zone sur le détecteur où elle se projette. (b) À partir de la zone sur le détecteur, on détermine la taille du sous-volume total dans lequel l'information des projections est présente.

projections, nous pouvons exécuter la reconstruction et ne sauvegarder que le sous-volume interne. La figure 5.16 illustre le déroulement général de l'algorithme.

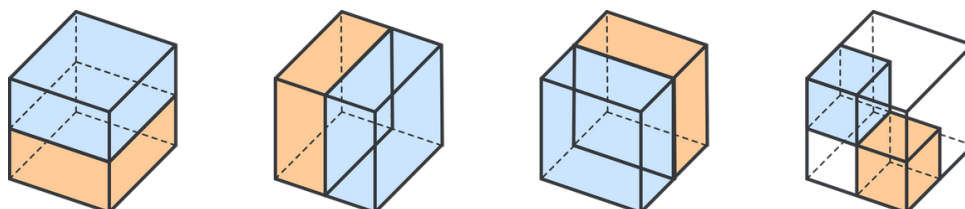


FIGURE 5.15 – Représentation des quatre méthodes principales de découpage d'un volume.

Cette méthode (que nous appellerons méthode 2) nécessite certes plus de VRAM pour pouvoir fonctionner par rapport à la méthode 1, mais elle est généralement plus rapide. Ceci est dû aux couches supplémentaires qui doivent être reconstruites pour chaque sous-volume et qui prennent la forme d'un *padding* (espace tampon) en dessous et au-dessus.

Cette méthode permet d'optimiser les transferts de données et résulte en une occupation GPU très élevée comme le montre la figure 5.17, mais montre néanmoins des limites quand le volume est très grand. En effet, non seulement le temps de calcul augmente avec le nombre de sous-divisions, mais une sous-division a une taille minimale dépendante de la conicité de la géométrie, qui augmente à la fois la taille mémoire nécessaire et le temps de calcul supplémentaire dû aux espaces tampons.

5.2.5 Méthode implémentée

Pour obtenir un algorithme qui soit robuste tout en mitigant la réduction de performance due aux très grands jeux de données, nous sommes partis de la base de l'algorithme

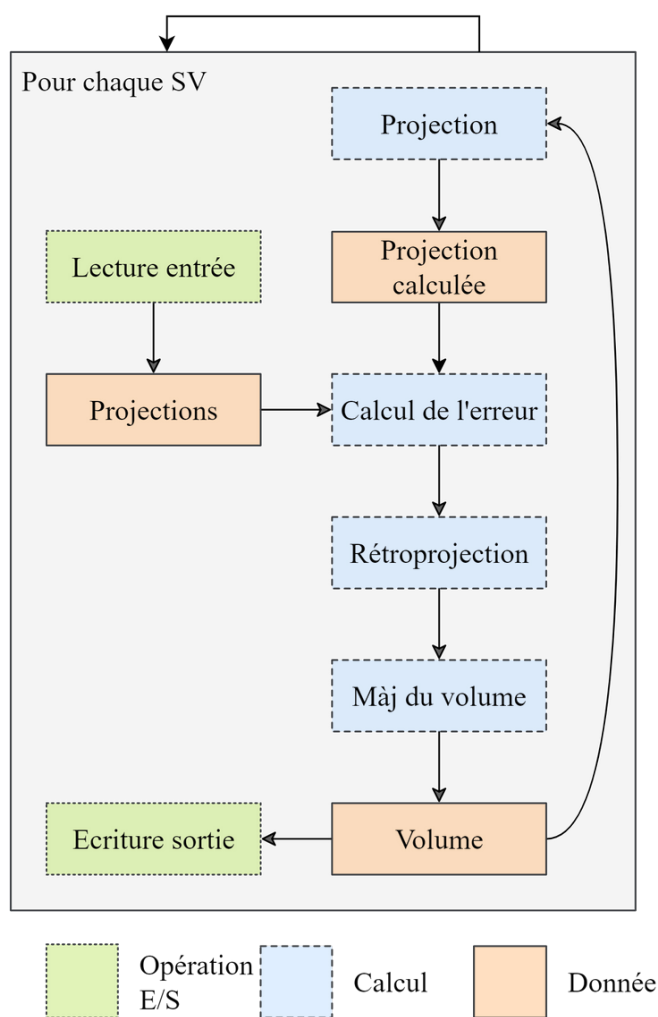


FIGURE 5.16 – Schéma de fonctionnement d'un algorithme de reconstruction tomographique séparé en sous-jeux de données

minimisant l'espace mémoire nécessaire et avons poussé l'optimisation un peu plus loin pour limiter au maximum l'utilisation du disque dur.

Au chargement des données, le programme va calculer la taille mémoire GPU nécessaire et la quantité de projections qu'il peut stocker à la fois. Le nombre de sous-itérations est choisi à partir de cette valeur. Ensuite, un nombre défini de fils d'exécutions va effectuer les opérations de projections couche par couche en travaillant chacun avec un espace mémoire GPU dédié.

À ce stade, on est toujours limité par la taille de la RAM et/ou la vitesse de lecture du disque, car stocker les couches du volume sur celui-ci va toujours ralentir le calcul. Pour pallier ce problème, la puissance de calcul disponible et non utilisé du CPU est mise à profit de la manière suivante : chaque fil d'exécution qui gère le GPU est doublé d'un autre qui va gérer la disponibilité des couches du volume. Ces couches sont compressées et décompressées à la volée en utilisant un algorithme de compression rapide. Cela permet à la fois de réduire l'occupation de la RAM, mais également dans le cas où le disque dur est utilisé, de réduire la quantité de données à écrire.

La quantité de compression peut être adaptée selon les besoins de la reconstruction : les

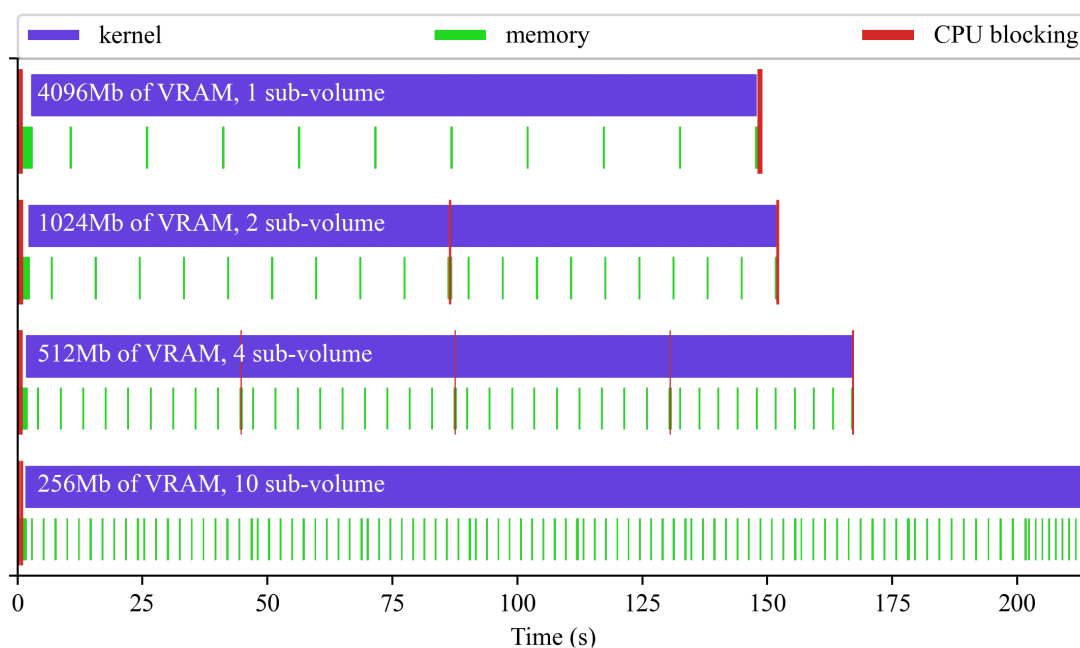


FIGURE 5.17 – Exemple de profilage relevé durant l'exécution d'une reconstruction tomographique découpée en sous-jeux de données pour différentes quantités de mémoire vidéo disponible. Le temps de reconstruction augmente avec moins de mémoire, car plus de voxels doivent être traités du fait des espaces tampons nécessaires à la division du volume.

premières itérations de reconstruction peuvent être compressées avec perte sans qu'aucun impact significatif soit visible sur la reconstruction finale, tandis que les dernières itérations peuvent profiter d'une compression sans perte pour éviter d'introduire des artefacts. De même, le volume final peut être stocké sans compression, avec compression sans perte, ou avec perte, selon le cas d'usage.

Nous avons étudié les différentes méthodes de compressions disponibles qui puissent être adaptées à notre problématique. Pour commencer, les méthodes de compression de nombres flottant sans perte imposent un très faible taux de compression. Nous optons donc pour stocker les couches en nombre entier stocké sur 16 bits (65536 valeurs), ce qui est courant même en imagerie médicale et suffisante vis-à-vis de la précision des détecteurs disponibles, généralement de 10 à 12 bits (1024 à 4096 valeurs). Le format PNG permet de stocker des entiers 16bits compressés (sans perte uniquement), mais les implémentations disponibles ne sont pas optimisées pour la vitesse de compression et décompression et demandent donc trop de temps de calcul. Le JPEG2000 [92] est un format utilisé dans l'imagerie médicale pour sa capacité à compresser sans perte des images 16bits avec un fort de taux de compression, mais encore une fois sa compression avec les bibliothèques disponibles est trop lente pour notre application. Nous nous sommes donc tournés vers le PGF [93] qui utilise un algorithme équivalent au JPEG2000, mais est pensée pour la vitesse de compression et décompression. Il supporte la compression avec et sans perte pour les entiers 16bits. Ce format supporte également la décompression de version sous-échantillonnée des images, ce qui pourra nous être utile si nous ajoutons le support de la reconstruction multi-échelle.

La bibliothèque est relativement facile à utiliser et les taux de compression sont très bons, similairement au JPEG2000. Comme les couches d'un volume sont relativement

uniformes et que le bruit présent sur les projections est lissé par la reconstruction, on peut obtenir un taux de compression de 5 à 10 en compression sans perte, et plus que 10 en acceptant de légères pertes. Un volume complet de 600 Go est donc réduit à moins de 100 Go, ce qui permet d'en stocker une partie en RAM et le reste sur le disque sans que la vitesse de lecture de celui-ci ne limite le calcul.

Il sera également intéressant d'étudier le comportement d'étudier le comportement du format HTJ2K (High-Throughput JPEG2000), récemment rentré dans le standard du JPEG2000 et qui promet une réduction du temps de compression et décompression d'un ordre de grandeur. Encore peu de bibliothèque supportent cette extension, mais le support de ce format est déjà plus répandu que le PGF.

5.3 Échantillonnage de la projection

La projection par matrice donne des coordonnées réelles qui doivent être interprétées pour obtenir des coordonnées de pixel. Avec l'algorithme de projection détaillé précédemment (5.1.5), nous obtenons une coordonnée 2D en virgule flottante (FP) pour chaque voxel sur chaque projection.

La technique utilisée pour calculer précisément la projection sur un détecteur pixelisé aura un impact important sur les performances finales de l'algorithme, tant en termes de précision que de temps d'exécution.

5.3.1 Les techniques d'interpolation

Les techniques de haute précision telles que le modèle intégral de volume (VIM) donneront une solution exacte au prix d'un temps de calcul fortement augmenté. Son principe est de projeter les sommets des voxels sur le détecteur et de calculer un poids pour chaque pixel sur lequel il se projette. Cela nécessite beaucoup de calculs et d'accès à la mémoire pour chaque voxel, car il y aura beaucoup d'accès liés à un poids négligeable. Cette méthode est la plus précise, mais aussi la plus coûteuse.

On peut noter que cette relation pixel-voxel peut être précalculée dans une matrice qui stockera le poids pour chaque paire de voxels et de pixels. Cela peut être très efficace pour les petits volumes, mais la taille de cette matrice, même lorsqu'elle est stockée dans une matrice éparsée (chaque voxel projettera sur presque toutes les projections sur plusieurs pixels, donc cette matrice est largement plus grande que le volume lui-même). Ceci n'est pas faisable dans la majorité des cas, nous devons donc trouver un moyen plus rapide de calculer une projection précise.

Pour réduire le temps nécessaire à une projection, nous pouvons bien sûr essayer d'optimiser le nombre d'opérations nécessaires, mais il est également très important de prendre en compte le coût d'un accès mémoire sur un GPU. Par rapport à la mémoire vive (DDR) qui est optimisée pour la latence, la VRAM que l'on trouve sur les GPU actuels (GDDR) est davantage axée sur une bande passante élevée, au détriment du temps d'accès [94], [95]. Un accès à la mémoire peut prendre plus d'une centaine de cycles GPU [96], donc l'optimisation de l'accès à la mémoire a autant sinon plus d'impact que l'optimisation du nombre d'opérations dans de nombreux cas.

Compte tenu de la vitesse à laquelle les coordonnées flottantes peuvent être calculées

(multiplication d'une matrice 4x4 par un vecteur 4, qui est une opération très efficace sur les GPU modernes [88]), et comme le compilateur ne sera pas en mesure de prédire où se feront la plupart des accès mémoire, cette étape représentera une grande partie du temps passé dans chaque exécution du noyau.

Les deux premières façons les plus évidentes d'obtenir rapidement les coordonnées d'un pixel sont soit d'arrondir la valeur, soit d'interpoler les valeurs sur les quatre voisins en leur attribuant un poids.

Dans les deux cas, cela n'est valable que si le volume est construit de telle sorte que la taille du voxel soit égale ou très proche de la taille d'un pixel divisée par le grossissement. Avec la géométrie spécifiée dans la section précédente, nous pouvons dire que par construction, le côté du pixel d'un détecteur est le même que le côté d'un voxel au niveau de l'axe de rotation en tenant compte du grossissement. Ainsi, si nous ignorons le grossissement à l'intérieur de l'objet, un voxel sera toujours projeté sur au plus quatre pixels. Cette approximation est correcte lorsque la taille de l'objet est très petite par rapport à SO et SD.

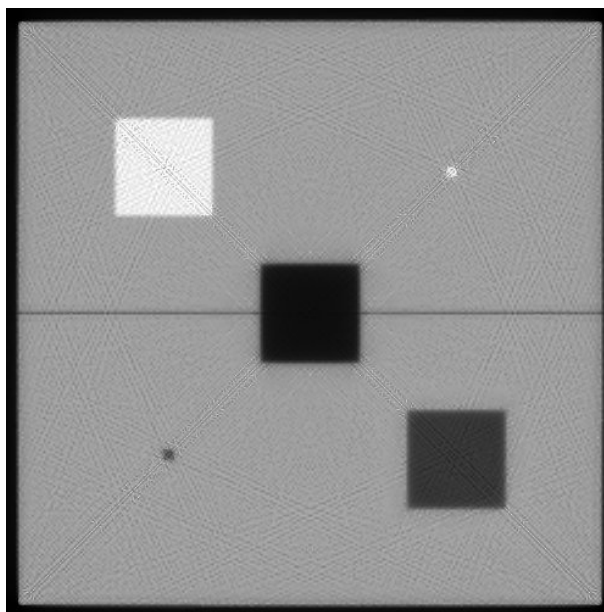


FIGURE 5.18 – Illustration d'artefacts de Moiré sur une couche de volume reconstruit.

Pour la méthode d'arrondi, une seule opération et un seul accès mémoire sont nécessaires, cependant elle provoque également des artefacts de Moiré (figure 5.18) qui deviennent de plus en plus prononcés avec la convergence de la reconstruction.

Pour l'interpolation linéaire, le coût en complexité numérique est augmenté d'une quantité relativement faible, mais cela multiplie par quatre le nombre d'accès mémoire. En contrepartie, Les artefacts de Moiré sont considérablement réduits, les seuls artefacts restants étant causés par l'approximation de la taille du voxel et de son angle par rapport au détecteur.

La projection pilotée par la distance [97], qui consiste à attribuer un poids basé sur le temps de parcours du rayon pixel-voxel à l'intérieur du voxel, donnera un résultat similaire à l'interpolation, mais ne fonctionnera pas avec une projection pilotée par le pixel.

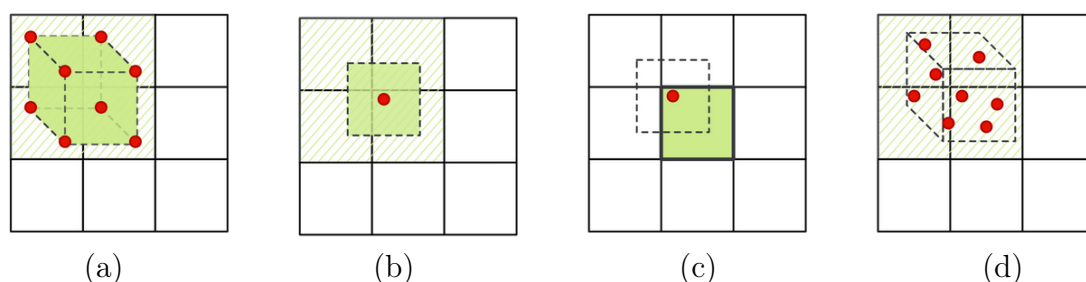


FIGURE 5.19 – Exemple de quatre méthodes d'interpolation de la projection d'un voxel sur une grille régulière. (a) Interpolation exacte du cube (b) Interpolation linéaire (c) Interpolation au plus proche voisin (d) Interpolation par Monte-Carlo.

5.3.2 L'interpolation par Monte-Carlo

Compte tenu de la nature itérative de l'algorithme, nous pouvons utiliser chaque étape de la reconstruction comme un échantillonnage des projections avec la méthode de Monte-Carlo (MC) [98], [99]. En projetant non pas le voxel, ou un point déterminé dans celui-ci, mais un point aléatoire à l'intérieur de chaque voxel pour chaque itération/sous-itération, le coût est identique à la projection par arrondi (une seule opération de calcul et de lecture/écriture pour chaque voxel).

Étant donné que les algorithmes de reconstruction algébriques réalisent un calcul similaire à une moyenne, en choisissant un point d'échantillonnage différent à chaque itération, la reconstruction va moyenniser tous les échantillons aléatoires et converger vers le résultat attendu, sans artefact de Moiré.

Cela fonctionnera particulièrement bien avec S(M)ART ou OS-(M)ART puisque plusieurs projections sont considérées en même temps et que la valeur de mise à jour sera la moyenne de plusieurs échantillons.

L'avantage principal de cette méthode est qu'elle prend en compte tous les paramètres géométriques : perspective, orientation et temps de parcours à l'intérieur du voxel, tout en ayant un coût négligeable.

5.4 Conclusion

Dans ce chapitre, nous avons vu les choix techniques et les points importants rencontrés durant l'implémentation de mon logiciel de reconstruction tomographique.

La principale particularité de celui-ci par rapport aux implémentations existantes est qu'il est conçu pour minimiser l'impact sur le temps de reconstruction que des jeux de données très lourds peuvent causer.

Pour cela la séquence de reconstruction minimise la quantité de mémoire vive et de mémoire vidéo dont elle a besoin à tout moment, et le chargement des données est optimisé et réalisé parallèlement au calcul de la reconstruction sur le GPU afin de minimiser le temps où celui-ci n'a pas de données à disposition pour continuer le processus.

Nous avons également accéléré la méthode d'interpolation en utilisant les propriétés de la reconstruction tomographique.

Nous allons voir dans le chapitre suivant comment cette implémentation se comporte vis-à-vis des implémentations existantes.

Les travaux de ce chapitre ont été présentés à la conférence *Recherche en Imagerie et Technologies de la Santé (RITS) 2022* sous la forme d'un poster, également accepté pour la conférence *Rayons X et Matière (RXM) 2021*, mais je n'ai malheureusement pas pu assister à celle-ci.

Chapitre 6

Validation

Dans ce chapitre, je vais présenter les résultats obtenus avec les différentes méthodes et implémentations développées durant ma thèse.

Dans un premier temps, je vais évaluer la récupération automatique du ratio avec la formule de Paganin à l'aide de données réelles et simulées. Ensuite, je vais évaluer ma méthode de reconstruction du front d'onde pour masque de Hartmann et la comparer à la méthode existante.

Dans un second temps, je vais évaluer l'implémentation de la reconstruction tomographique, avec une analyse quantitative en le comparant aux résultats de TIGRE [68], avec et sans correction d'alignement. J'évaluerai ensuite les performances en temps et en mémoire de mon implémentation en la comparant à TIGRE et Astra Toolbox [66], puis je finirai par une présentation de résultats obtenus avec des acquisitions réelles tentant de s'approcher de la situation ciblée par le projet XPulse.

6.1 Métriques

Dans ce chapitre, je vais évaluer la qualité des résultats obtenus avec nos méthodes et les comparer avec des logiciels et des algorithmes existants. Trois algorithmes sont étudiés : La reconstruction tomographique, l'extraction de ratio automatique par Paganin et l'intégration de la phase par masque de Hartmann à la reconstruction tomographique. Les cas d'utilisations étant très différents, chacune de ces méthodes sera testée de manière adaptée.

Pour la reconstruction tomographique, je vais comparer à deux logiciels qui feront office d'état-de-l'art : TIGRE développé par le CERN et Astra Toolbox par l'Université d'Antwerp. Pour mesurer la qualité de la reconstruction de manière fiable, je ferai les tests avec une simulation. Afin de ne pas fausser le résultat, ces simulations seront faites à l'aide de la projection de TIGRE, la définition de la géométrie sera donc faite selon sa configuration et sera ensuite traduite pour Astra Toolbox et XPRecon. Même s'il est possible d'obtenir une mesure de convergence des algorithmes de reconstruction durant la convergence de ceux-ci à l'aide de la différence entre le volume projeté et les projections mesurées, cela ne nous informe qu'indirectement sur la véritable différence entre le volume reconstruit et le volume réel. En simulant un fantôme numérique, on peut à la place

mesurer la qualité de la convergence de la reconstruction à partir de la différence entre le volume du fantôme simulé et celui reconstruit par chaque algorithme, qui est une mesure plus intéressante, car directement effectuée sur le résultat obtenu. Nous utiliserons pour cela la moyenne de l'erreur quadratique, MSE .

$$MSE = \text{moyenne} [(Reference - Reconstruit)^2] \quad (6.1)$$

Après avoir vérifié la convergence, les différents points d'alignements seront testés pour vérifier que la représentation de notre géométrie me permet bien de les corriger, en comparant le résultat désaligné et corrigé avec un résultat sans désalignement.

Pour l'algorithme de récupération du ratio $\frac{\delta}{\beta}$ automatique pour la formule de Paganin, on utilise d'abord des simulations pour comparer le ratio simulé à celui retrouvé, puis on applique l'algorithme à des images réelles pour des objets dont on connaît les propriétés. Le but est de retrouver le ratio théorique, ou si ce n'est pas possible, de pouvoir différencier deux matériaux à partir du ratio calculé.

Pour l'algorithme d'intégration de la phase par masque de Hartmann à la reconstruction tomographique, nous comparerons les résultats obtenus avec des données réelles lorsque reconstruits par la chaîne de traitement standard, et par mon algorithme. Nous évaluerons la qualité des résultats avec quatre métriques : le ratio du signal sur bruit (SNR), le ratio du signal crête sur bruit ($PSNR$), le ratio du contraste sur bruit (CNR), et enfin le contraste (CTR).

$$SNR = \frac{\text{Moyenne}(\text{Signal})}{\sqrt{\text{Variance}(\text{Bruit})}} \quad (6.2)$$

$$PSNR = \frac{\text{Max}(\text{Signal}) - \text{Min}(\text{Signal})}{\sqrt{\text{Variance}(\text{Bruit})}} \quad (6.3)$$

$$CNR = \frac{\text{Max}(\text{Signal}) - \text{Min}(\text{Signal})}{\sqrt{\text{Variance}(\text{Bruit})}} \quad (6.4)$$

$$CTR = \text{Max}(\text{Signal}) / \text{Min}(\text{Signal}) \quad (6.5)$$

Pour établir les parties de la reconstruction représentant le bruit et le signal, nous effectuerons une segmentation manuelle pour isoler les différents objets dans le volume.

6.2 Ratio Paganin

L'imagerie de phase par propagation libre permet, grâce à la formule de Paganin, de retrouver la phase d'un front d'onde à partir de l'amplitude mesurée et de la valeur du ratio $\frac{\delta}{\beta}$. C'est pour tenter de se passer de cette dernière condition que j'ai tenté de développer une méthode d'extraction du ratio à partir des projections.

L'intérêt principal de cette technique est de pouvoir trouver le ratio $\frac{\delta}{\beta}$ d'un matériau à partir des données géométriques et de l'énergie utilisée, ou tout du moins de pouvoir

discriminer deux matériaux différents, sans connaissance *a priori* sur l'objet imagé. Étant donné que notre système d'acquisition n'est pas mono-énergie, nous ne pouvons pas prétendre à trouver le ratio exact sur des données réelles, mais au mieux obtenir une estimation, sinon des valeurs incorrectes, mais cohérentes entre les matériaux.

Pour étudier cette méthode, nous l'avons donc appliquée à un jeu de données comprenant cinq tiges, avec un de silice de diamètre 2 mm, deux de quartz de diamètre 1 mm et 2 mm, ainsi que deux autres d'oxyde d'aluminium, également de diamètre 1 et 2 mm. Je me suis concentré sur les tubes de Quartz et d'oxyde d'aluminium, en laissant de côté la tige de silice (figure 6.3).

Nous avons donc trois matériaux différents et deux épaisseurs de tube différentes. Nous avons isolé un certain nombre de projections montrant plusieurs tubes (figure 6.1), et avons calculé le ratio pour chaque interface 5 fois afin d'obtenir la valeur et la stabilité de la mesure. Les résultats sont affichés sur la figure 6.2.

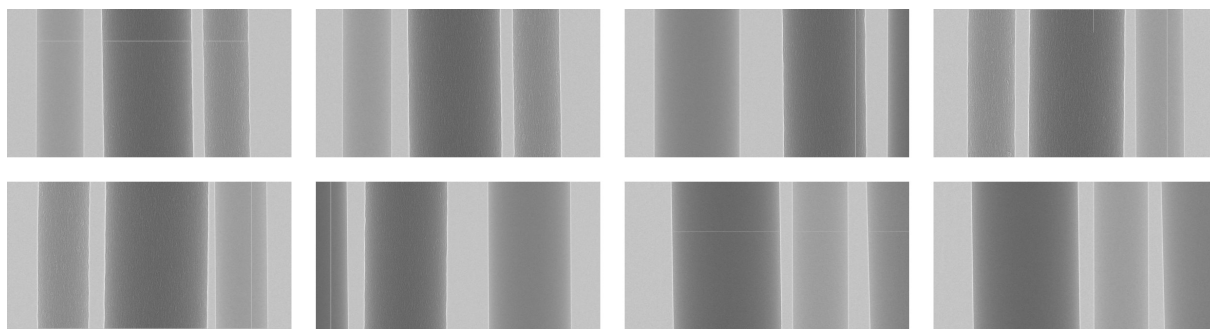


FIGURE 6.1 – Jeu de projections utilisé pour tenter de retrouver le ratio automatiquement.

Les ratios réels des deux matériaux sont 509 pour l'oxyde d'aluminium et 450 pour le quartz. On ne parvient pas à les retrouver, mais on obtient des valeurs du même ordre de grandeur.

Il est par contre difficile de séparer les deux matériaux à partir des ratios trouvés, ce que l'on aurait voulu pouvoir faire. On peut tout de même voir deux groupes qui semblent se former, l'un en dessous de 750 et l'autre au-dessus.

Cette méthode d'extraction du ratio va demander encore du travail pour parvenir au résultat souhaité, mais les tests réalisés jusqu'à présent semblent prometteurs. Je continuerai à travailler sur cette méthode durant la suite du projet XPulse.

Pour conclure sur cette technique, si l'on obtient des résultats prometteurs pour les simulations de faibles épaisseurs, il faut encore travailler cette technique pour pouvoir mieux isoler les conditions d'applications valides, ainsi que tenter d'élargir ces conditions pour tenter de l'appliquer à des objets réels plus fortement absorbants.

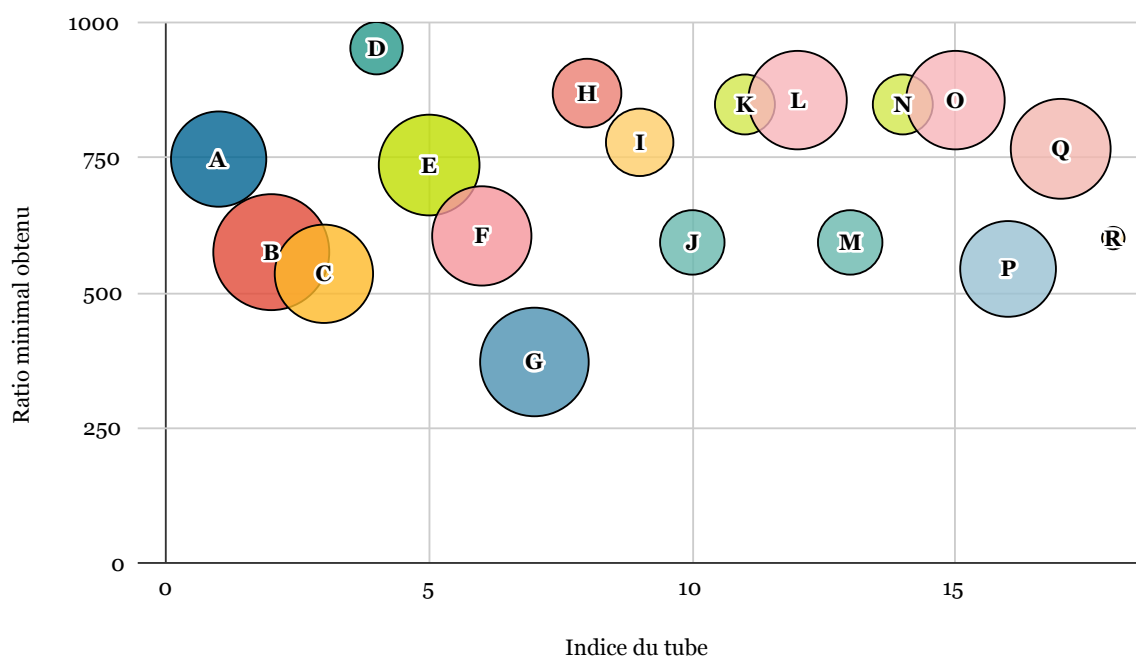


FIGURE 6.2 – Résultat obtenu avec de multiples mesures du ratio sur les images de la figure 6.1. L’incertitude est calculée à partir de multiples mesures sur un même tube (minimum/maximum, 25ème/75ème centile).

6.3 Qualité Southwell 3D

6.3.1 Méthodologie

Pour évaluer la convergence et le résultat final de notre implémentation de GS 3D (Algorithme d’intégration de phase de Gauss-Seidel 3D, 4.2.4), nous avons utilisé une acquisition réelle avec un fantôme physique réalisé sur l’installation de test à ALPhANOV. En effet, nous n’avons pas la possibilité pour le moment de réaliser une simulation d’acquisition avec un masque de Hartmann.

Ce fantôme est illustré en figure 6.3 et se comporte de cinq tubes de quartz, d’alumine et de silicium cristallin, dans un support en bois dont trois tiges sont visibles sur les bords de la zone de reconstruction.

Pour les tests, j’ai commencé par établir un nombre d’itérations de reconstruction tomographique nécessaires pour que la reconstruction ait convergé correctement, que j’ai fixé à 100 itérations ici. Nous avons ensuite fait des intégrations de phase avec GS 2D en faisant varier le nombre d’itérations de GS et en utilisant à chaque fois le résultat pour faire une reconstruction tomographique de 100 itérations de OS-SMART. J’ai ensuite utilisé le GS 3D en utilisant les mêmes nombres d’itérations de GS total, pour 3 pas différents (toutes les 1, 2 et 3 itérations de reconstruction tomographique).

Pour évaluer la qualité des reconstructions, des mesures de SNR, PSNR, CNR et de contraste sont réalisées sur les couches de la reconstruction.

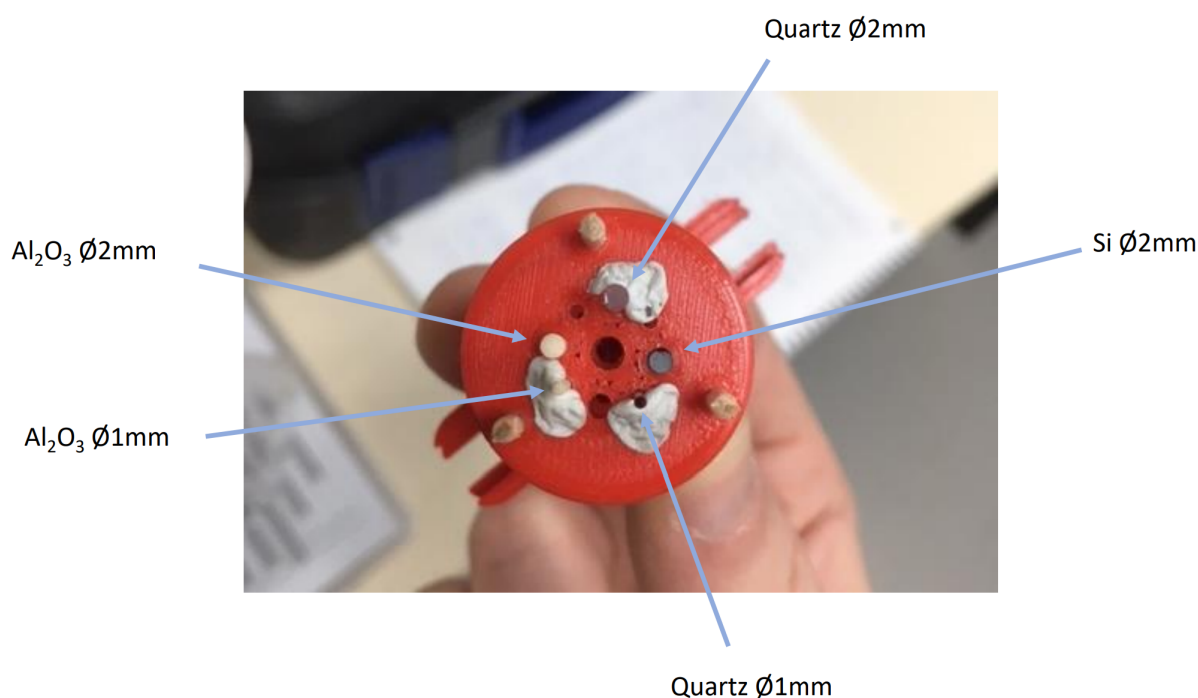


FIGURE 6.3 – Photographie du fantôme utilisé pour les tests d'extraction de phase.

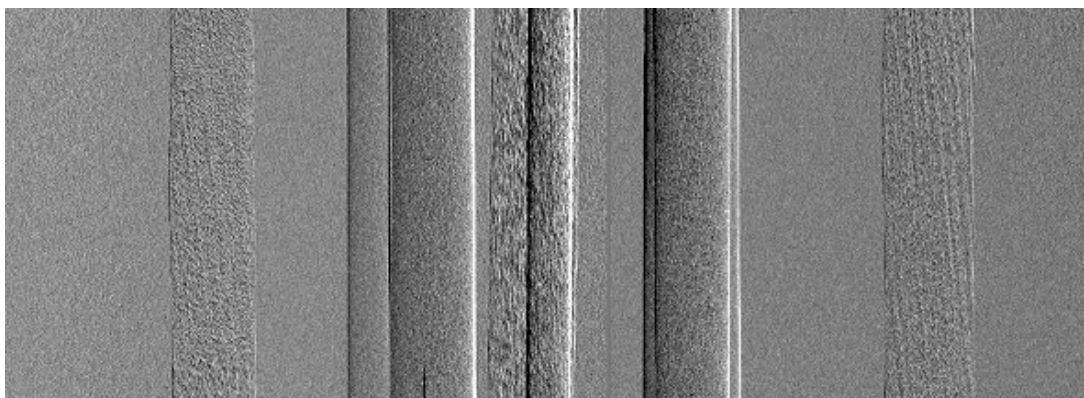


FIGURE 6.4 – Exemple de carte de déflexion horizontale du fantôme composé de cinq tiges.

6.3.2 Résultats

Le jeu de donnée, illustré par une projection en figure 6.4 est reconstruit avec les différentes méthodes, et les projections et couches intermédiaires sont affichées en table 6.1 et figure 6.2. Les courbes 6.5a, 6.5b, 6.5c et 6.5d présentent les mesures de qualité calculées à partir de ces reconstructions. On peut voir que pour chaque valeur, notre algorithme permet d'obtenir un meilleur score, en plus d'offrir une convergence stable. Les résultats obtenus avec le GS 2D mettent en évidence le problème de cohérence des projections par une convergence erratique. On peut tout de même noter qu'en dessous de 1000 itérations de GS, la version 2D converge plus rapidement dû au fait que dans sa version 3D, la reconstruction tomographique ralentit la convergence du GS en mélangeant l'information de chaque angle.

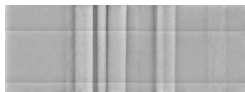
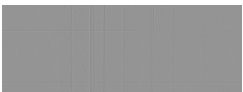
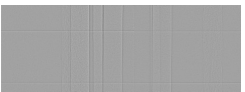
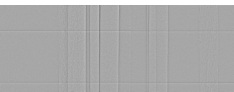
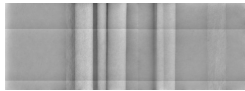



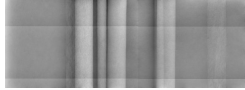

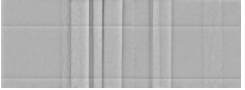
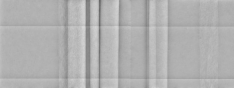
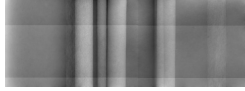
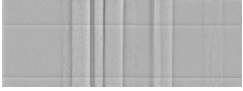
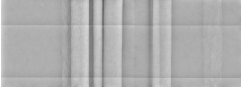
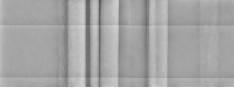
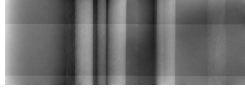
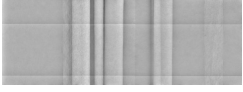
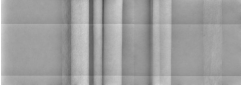
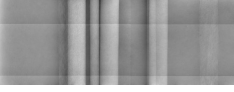
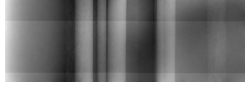
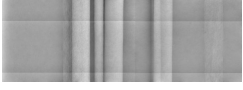
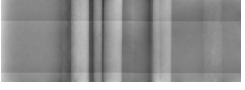
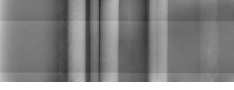

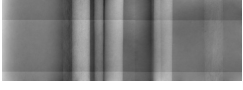
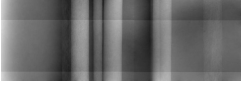
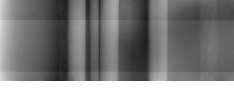
| Nb. iter. | GS 2D | GS 3D (Pas de 1) | GS 3D (Pas de 2) | GS 3D (Pas de 3) |
|-----------|---|---|--|---|
| 100 |  |  |  |  |
| 200 |  |  |  |  |
| 400 |  |  |  |  |
| 800 |  |  |  |  |
| 1600 |  |  |  |  |
| 3200 |  |  |  |  |
| 10000 |  |  |  |  |

TABLE 6.1 – Exemples de projections extraites du processus itératif de la reconstruction tomographique et de phase simultanée selon l’algorithme utilisé et le nombre d’itérations de GS. Le pas indiqué pour chaque colonne représente le nombre d’itérations de reconstruction tomographique consécutives entre les applications d’intégration de la phase.

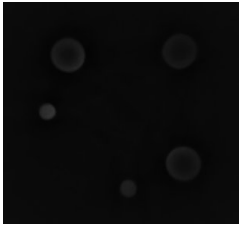






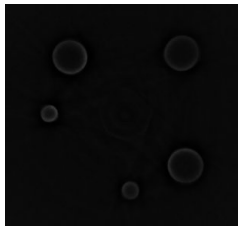


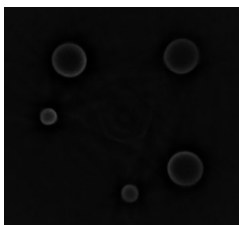
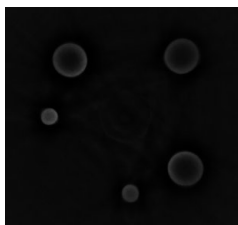
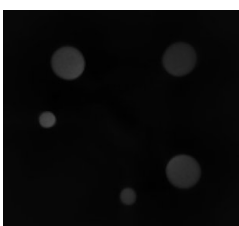
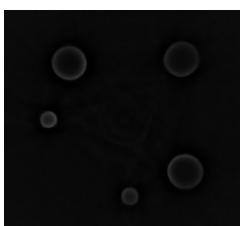
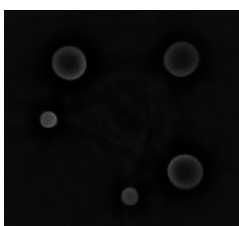
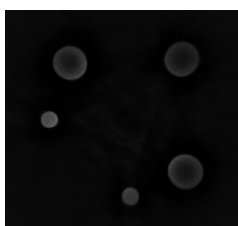

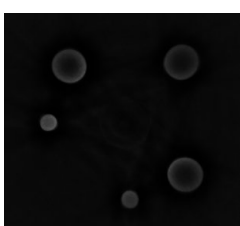
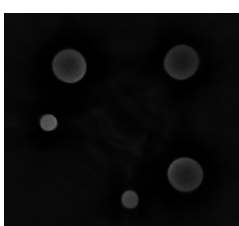

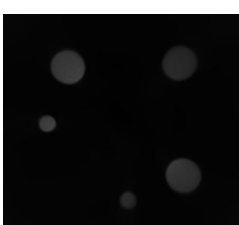
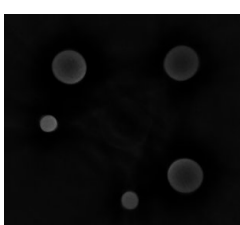


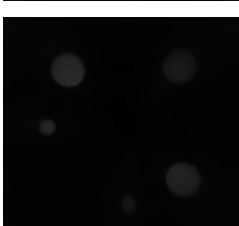
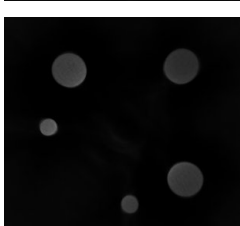

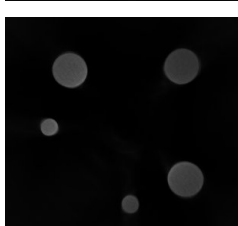
| Nb. iter. | GS 2D | GS 3D (Pas de 1) | GS 3D (Pas de 2) | GS 3D (Pas de 3) |
|-----------|---|---|--|---|
| 100 |  |  |  |  |
| 200 |  |  |  |  |
| 400 |  |  |  |  |
| 800 |  |  |  |  |
| 1600 |  |  |  |  |
| 3200 |  |  |  |  |
| 10000 |  |  |  |  |

TABLE 6.2 – Exemples de couches du volume extrait du processus itératif de la reconstruction tomographique et de phase simultanée selon l’algorithme utilisé et le nombre d’itérations de GS.

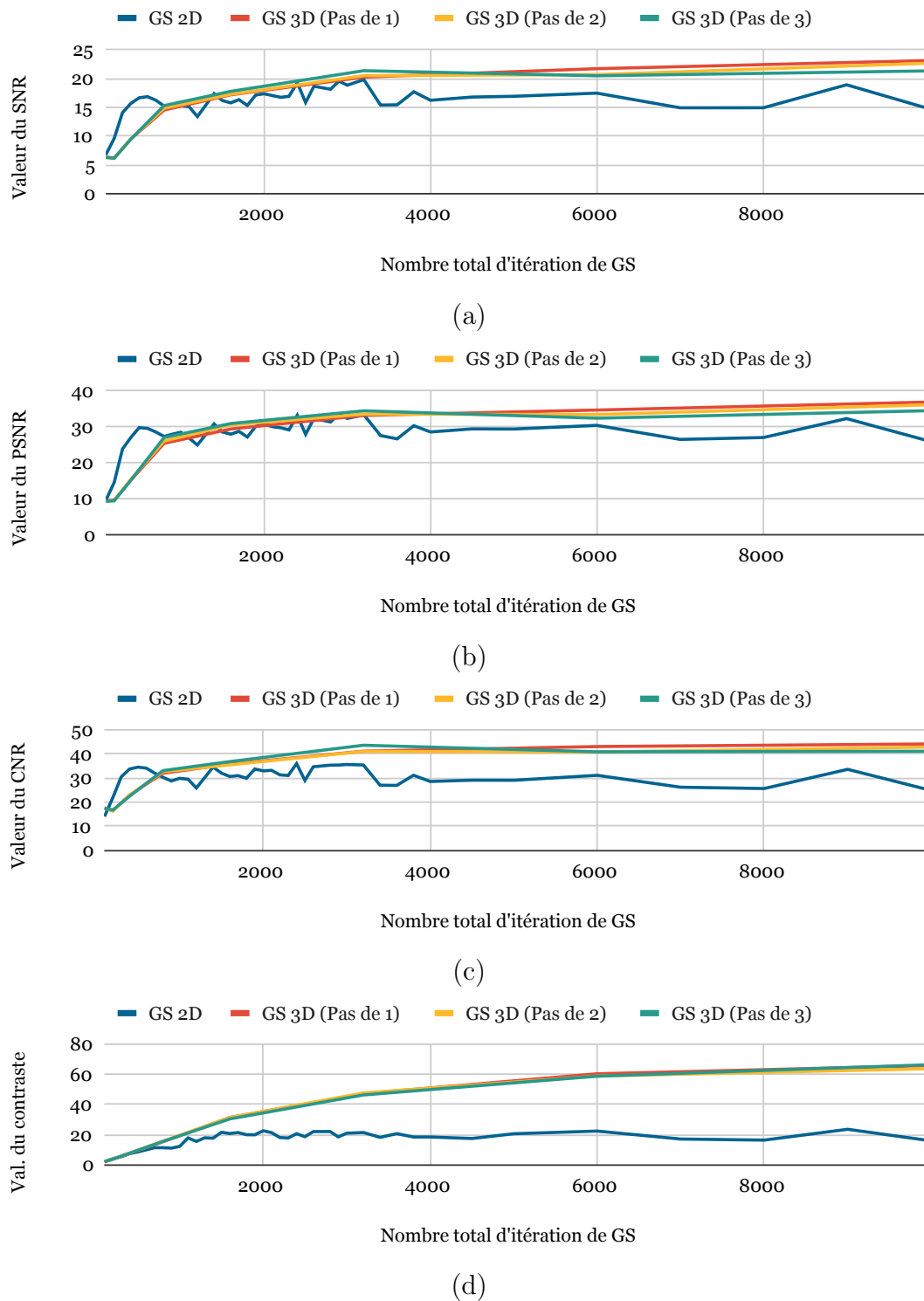


FIGURE 6.5 – Évolution des différentes métriques de mesure de qualité de la reconstruction tomographique et de phase simultanée en fonction du nombre d'itérations de GS. Pour chaque métrique, une valeur plus élevée indique une meilleure qualité de reconstruction.

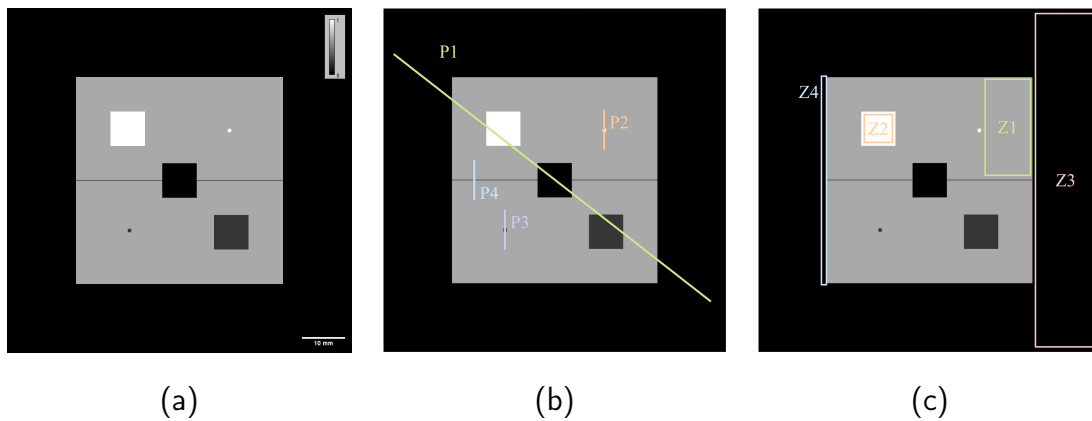


TABLE 6.3 – (a) Coupe du fantôme numérique. (b) Zones utilisées pour réaliser les profils (c) Zones utilisées pour calculer les différentes métriques.

6.4 Comparaison de la qualité de reconstruction

6.4.1 Fantôme numérique

Pour évaluer la qualité de reconstruction de notre algorithme, j’ai construit un fantôme numérique simple dont les projections ont été simulées par TIGRE et reconstruites avec Astra ToolBox, TIGRE et XPrecon. Ce fantôme prend la forme d’un objet cubique simple traversé de structures plus ou moins épaisses et de densité différente. Cet objet possède des arêtes franches et des détails de quelques voxels de large, des structures qui sont généralement difficiles à reconstruire en tomographie conique.

Le fantôme utilisé est représenté en figure 6.3a. La figure 6.3b indique où les profils utilisés dans cette section seront mesurés.

6.4.2 Mise en place de la géométrie

Chacun des logiciels testés demande une formalisation différente de la géométrie. Pour construire la géométrie, tous partent des distances entre source, objet et détecteur, puis ajoutent des points de réglage. Dans TIGRE, il est possible de régler le décalage du volume reconstruit par rapport à l’axe de rotation, le décalage du détecteur par rapport à la projection de la source et l’angle du détecteur sur les trois axes. Il est possible de donner un décalage différent pour chaque angle. Pour Astra Toolbox, la géométrie de base ne permet pas d’ajuster l’alignement, à la place, il est possible de construire sa propre géométrie en plaçant des différents éléments manuellement. Un point particulier à noter est que toutes les distances sont requises avec la taille du voxel comme unité. Pour XPrecon, j’utilise pour construire la géométrie des valeurs qui nous sont facilement accessible pour nos installations : décalage horizontal de l’axe de rotation sur le détecteur en pixels, décalage de la position de la source sur le détecteur en pixels et angle des détecteurs autour de la projection de la source. Cette méthode est plus adaptée à notre installation simplement par le fait qu’elle a été développée pour, il est probable qu’elle ne soit pas pratique pour d’autres cas.

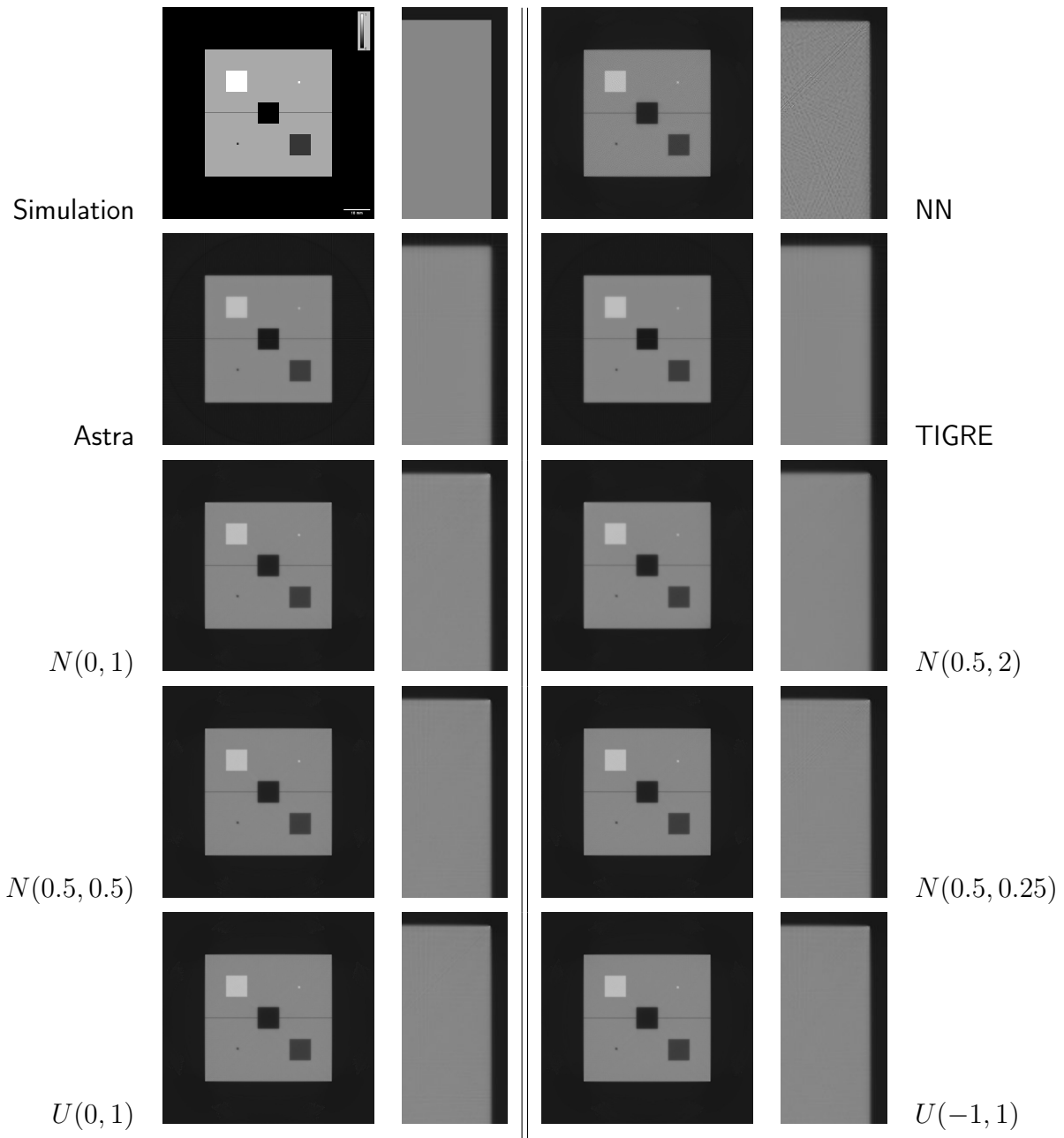


TABLE 6.4 – Tableau du résultat de la reconstruction pour différentes méthodes d'interpolations. NN = Plus proche voisin, $N(a,b)$ = distribution normale de moyenne a et d'écart-type b , $U(a,b)$ = distribution uniforme entre a et b .

| Méthode | StdDev (R1) | StdDev (R2) | StdDev (R3) | Moy. (R4) |
|----------------|-------------|-------------|-------------|-----------|
| Simu. | 0.000 | 0.000 | 0.000 | 0.01 |
| NN | 0.030 | 0.007 | 0.064 | 0.111 |
| Astra | 0.006 | 0.007 | 0.007 | 0.13 |
| TIGRE | 0.007 | 0.005 | 0.008 | 0.136 |
| $N(0, 1)$ | 0.005 | 0.006 | 0.006 | 0.129 |
| $N(0.5, 2)$ | 0.008 | 0.005 | 0.007 | 0.139 |
| $N(0.5, 0.5)$ | 0.006 | 0.006 | 0.005 | 0.094 |
| $N(0.5, 0.25)$ | 0.008 | 0.007 | 0.016 | 0.088 |
| $U(0, 1)$ | 0.007 | 0.007 | 0.007 | 0.087 |
| $U(-1, 1)$ | 0.006 | 0.006 | 0.009 | 0.122 |

TABLE 6.5 – Tableau de valeurs de différentes métriques calculées à partir des résultats présenté en figure 6.4.

6.4.3 Qualité de l’interpolation

Pour s’assurer que notre méthode d’interpolation n’introduit pas d’artefacts, nous pouvons faire varier les paramètres de celle-ci et comparer les résultats entre eux et avec les algorithmes de reconstruction existants. Trois types de reconstruction ont été réalisés : sans interpolation, avec interpolation par Monte-Carlo de probabilité uniforme, et avec interpolation par Monte-Carlo de probabilité normale. Pour les deux dernières, nous avons fait varier les valeurs minimale et maximale pour l’uniforme, et la valeur moyenne et la variance pour la normale.

Ces résultats sont affichés en figure 6.4, et les données mesurées dans la table 6.5.

On peut remarquer l’amélioration très prononcée entre la reconstruction sans interpolation et toutes les autres, montrant que notre méthode améliore nettement le résultat. Plus en détail, on voit que lorsque la zone représentée par un voxel est trop grande on perd en précision, et si elle est trop petite on perd en homogénéité, ce qui est le comportement attendu.

On notera l’interpolation uniforme $U(min, max)$, l’interpolation normale $N(\mu, \sigma)$, et NN (*nearest-neighbor*) pour aucune interpolation. Les valeurs numériques sont donnée en fonction de la taille d’un voxel. Les zones utilisées pour réaliser les mesures sont indiquées en figure 6.3c. Les zones 1 à 3 serviront à mesurer l’homogénéité de la reconstruction, tandis que la zone 4 servira à mesurer la précision de la reconstruction, la valeur moyenne de la zone augmentant avec la quantité de flou.

Parmi les différentes valeurs, les plus stables semblent être $U(0, 1)$ et $N(\mu = 0.5, \sigma = 0.5)$. Nous avons opté pour le $U(0, 1)$, et les résultats présentés et les mesures utiliseront cette méthode.

6.4.4 Comparaison des résultats

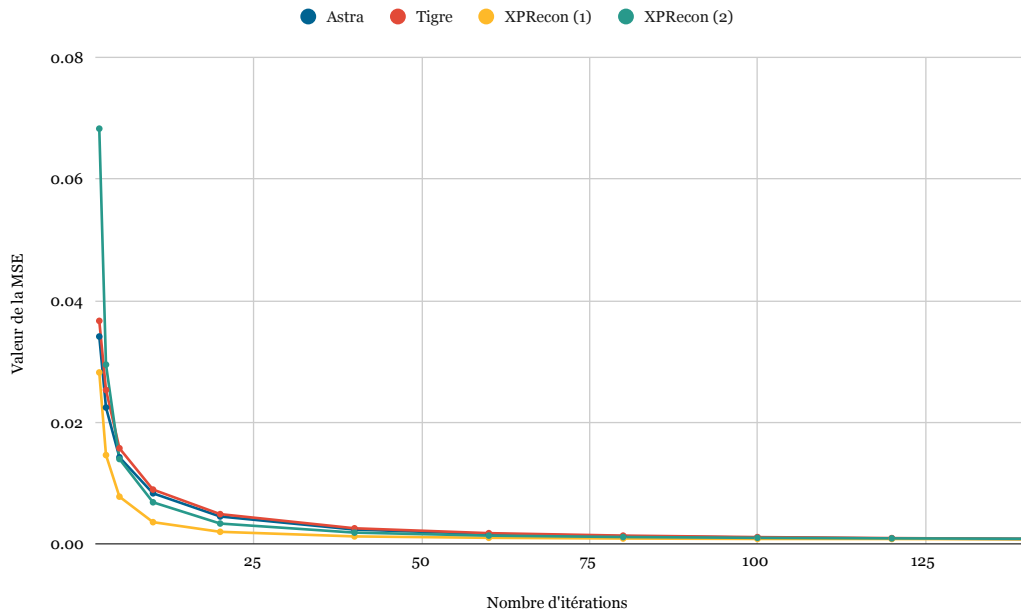


FIGURE 6.6 – Graphique de la valeur de la MSE de la reconstruction par rapport au nombre d'itérations pour Astra Toolbox, TIGRE et XPREcon (avec en (1) un poids des itérations constant et en (2) un facteur de 0.99, valeur par défaut pour les deux autres logiciels).

Pour assurer que notre algorithme reconstruit correctement, nous l'avons donc comparé à TIGRE et Astra Toolbox en mesurant la qualité des reconstructions pour différents nombres d'itérations.

On doit tout de même noter que les algorithmes disponibles ne sont pas identiques : XPREcon utilise le MART, Astra Toolbox le SIRT et TIGRE le SART. Les différences de vitesse de convergence sont donc principalement dûs à cette différence [18], [19].

Nous comparons donc principalement la qualité de la reconstruction une fois que l'algorithme a fini de converger, ainsi que la forme globale de la courbe de convergence. La figure 6.6 présente les résultats mesurés. Chaque algorithme a été exécuté deux fois pour s'assurer que les performances sont stables, ce qui a été le cas. Étant donné que les trois algorithmes utilisent une méthode de mise à jour différente, le réglage de la constante de vitesse de convergence est également difficile à équilibrer entre chacun. Pour être cohérent entre les algorithmes, les poids par défaut de $1 * 0.99^i$ (avec i le numéro de l'itération courante) ont été laissés, et une seconde exécution de XPREcon a été faite avec un poids un peu plus fort (plus adapté au MART), pour obtenir une estimation haute et basse.

On remarque qu'avec un poids de 1, les trois algorithmes convergent de manière presque identique, le MART de XPREcon lui permettant d'être légèrement plus rapide après 5 itérations, mais commence avec une erreur plus élevée avant ce point. Avec un poids plus haut (2.0 avec un facteur de 0.9 par itérations), il converge bien plus vite, mais il serait également possible de choisir un poids plus fort pour les autres logiciels. On peut

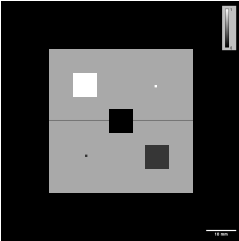
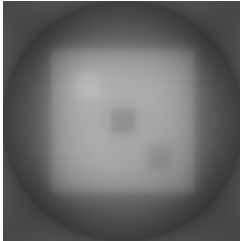
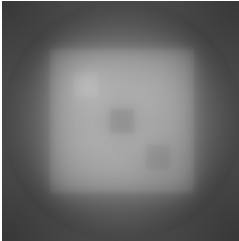
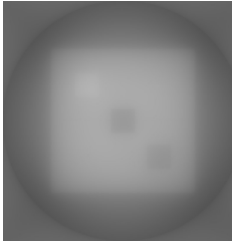
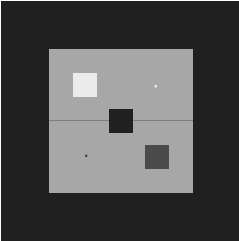
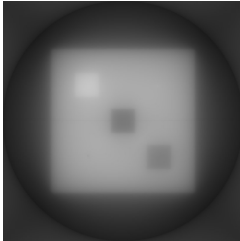
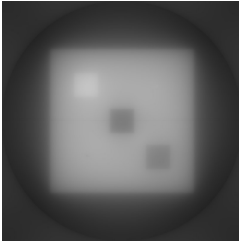
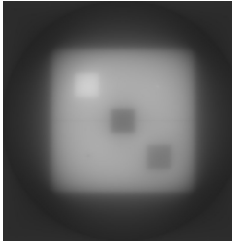
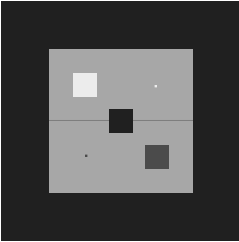
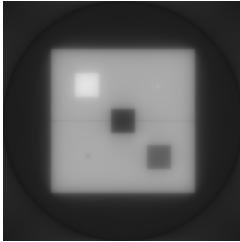
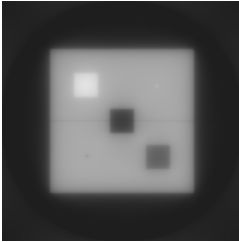
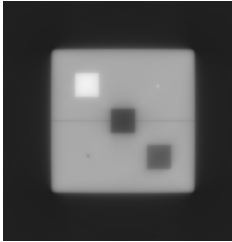
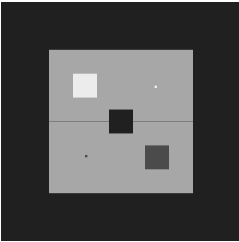
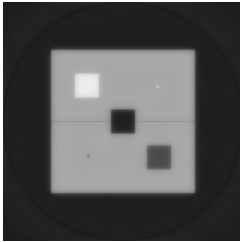
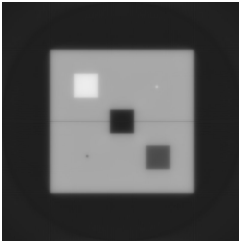
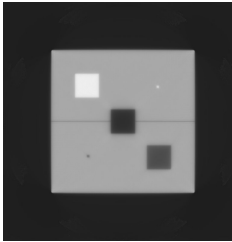
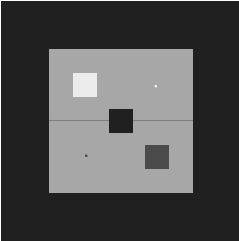
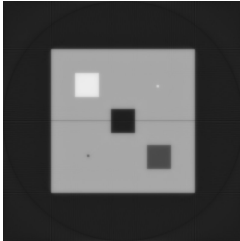
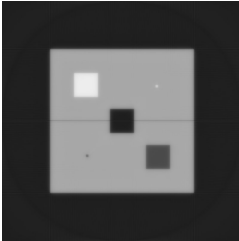
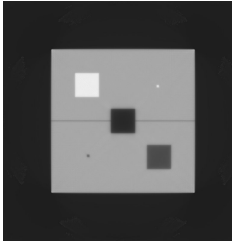
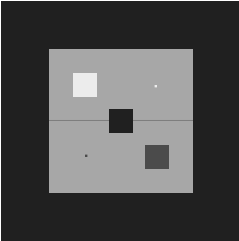
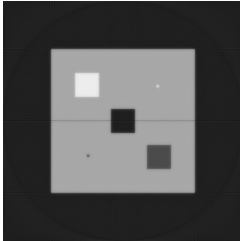
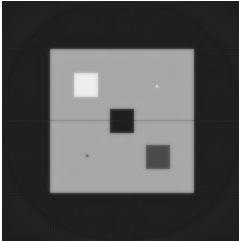
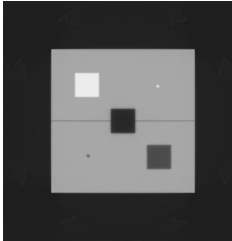
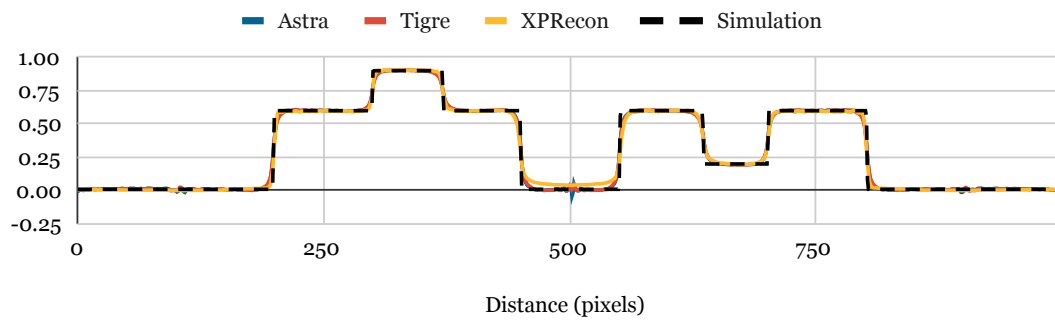
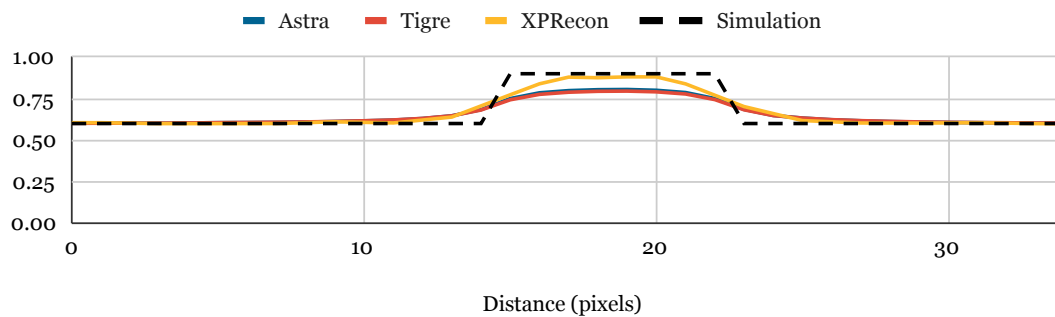
| Nb. iter. | Simulation | Astra | TIGRE | XPRcon |
|-----------|---|---|--|---|
| 2 |  |  |  |  |
| 5 |  |  |  |  |
| 20 |  |  |  |  |
| 60 |  |  |  |  |
| 100 |  |  |  |  |
| 140 |  |  |  |  |

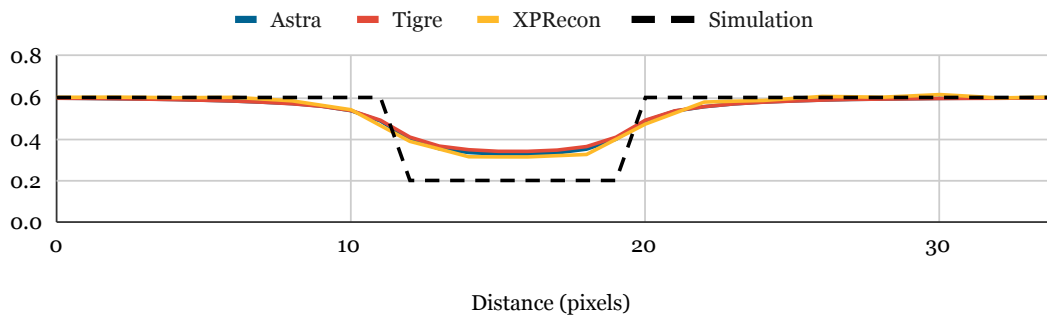
TABLE 6.6 – Tableau représentant l'état d'une couche de la reconstruction selon le nombre d'itérations pour les trois logiciels de reconstruction, comparée à la simulation numérique.



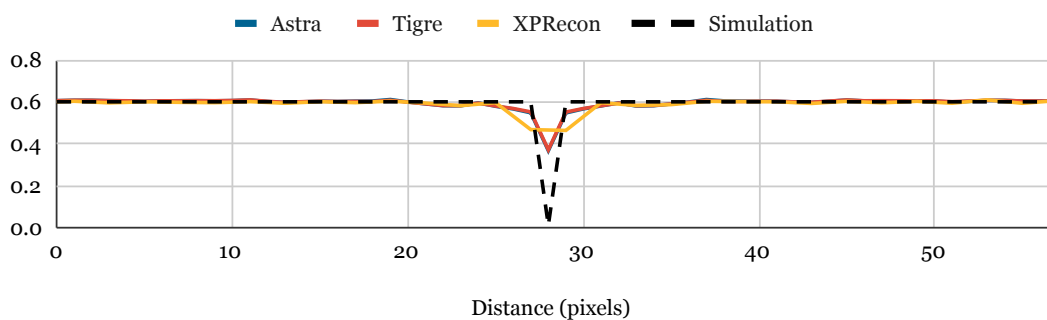
(a)



(b)



(c)



(d)

FIGURE 6.7 – Profils d’une couche de la reconstruction faits avec les 3 logiciels étudiés plus la simulation. Les zones utilisées pour le profil sont représenté sur la table 6.3b : (a) P1, (b) P2, (c) P3, (d) P4

conclure de ce graphique que les trois implémentations convergent de manière similaire et tendent à obtenir un résultat avec une MSE identique.

Le tableau 6.6 montre l'état d'une couche de la reconstruction à différents nombres d'itérations pour chaque logiciel, avec la couche équivalente de la simulation. C'est la version avec un poids non optimisé de XPREcon qui est affichée dans ce tableau.

On peut ensuite comparer les profils de quelques zones de la reconstruction (tel que délimitées à la section précédente).

Les profils 6.7a, 6.7b et 6.7c montrent que les trois logiciels reconstruisent correctement l'objet. On peut noter quelques particularités. Premièrement le bruit de reconstruction semble être légèrement plus présent sur les valeurs hautes que sur les valeurs basses pour XPREcon, alors que l'inverse est vrai pour TIGRE et Astra Toolbox. Sur le trou central du profil 6.7a, XPREcon reconstruit une valeur trop élevée alors que TIGRE et Astra Toolbox introduisent un artefact à cet endroit.

Sur les profils 6.7b et 6.7c, XPREcon semble mieux reconstruire ces détails d'une taille de 8 voxels, surtout visible pour le détail avec une valeur haute.

Sur le profil 6.7d, on voit que les trois logiciels peinent à reconstruire convenablement ce détail de la taille d'un voxel, mais XPREcon semble tout de même introduire plus de flou que les deux autres.

6.5 Validation de la correction d'alignement

Pour m'assurer que les paramètres de réglage de la géométrie du logiciel permettent bien de corriger les erreurs d'alignement requises, j'ai effectué une série de simulations et comparé le résultat de la reconstruction avec XPREcon et TIGRE, avec et sans correction de l'alignement. Le fantôme numérique utilisé est le même que jusqu'à présent (figure 6.3).

Comme nous travaillons sur des installations expérimentales, celles-ci sont fréquemment démontées et remontées, il est donc plus facile de corriger l'alignement numériquement que de faire des acquisitions d'alignement très régulièrement.

6.5.1 Axe de rotation

Le recalage de la projection de l'axe de rotation est l'erreur d'alignement la plus courante, mais aussi la plus simple à mesurer et à identifier. En effet, lors d'une acquisition tomographique à 360 degrés, il est possible de facilement calculer sa position à partir des projections : leur somme résulte en une image avec une symétrie verticale et la position de l'axe de symétrie nous donne directement la position de l'axe. Une reconstruction dont la position de l'axe de rotation est incorrecte comportera une zone de flou autour de chaque élément dont la taille augmente avec l'erreur d'alignement.

6.5.2 Position de la source

La position de la source est une erreur d'alignement propre à la géométrie conique (ou *fan-beam*). En règle générale, son effet est difficile à discerner si la distance source-détecteur est très grande par rapport au décalage de la source.

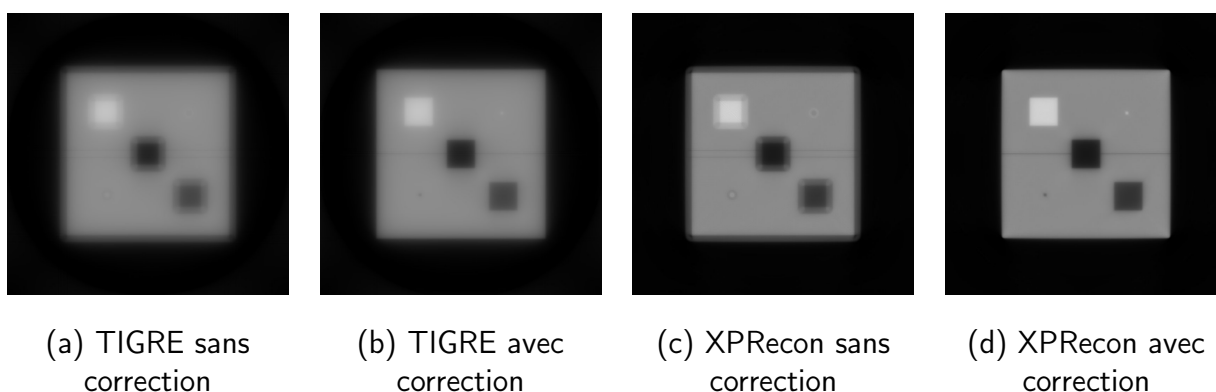


FIGURE 6.8 – Couche de la reconstruction pour TIGRE et XPREcon avec et sans correction de l'axe de rotation

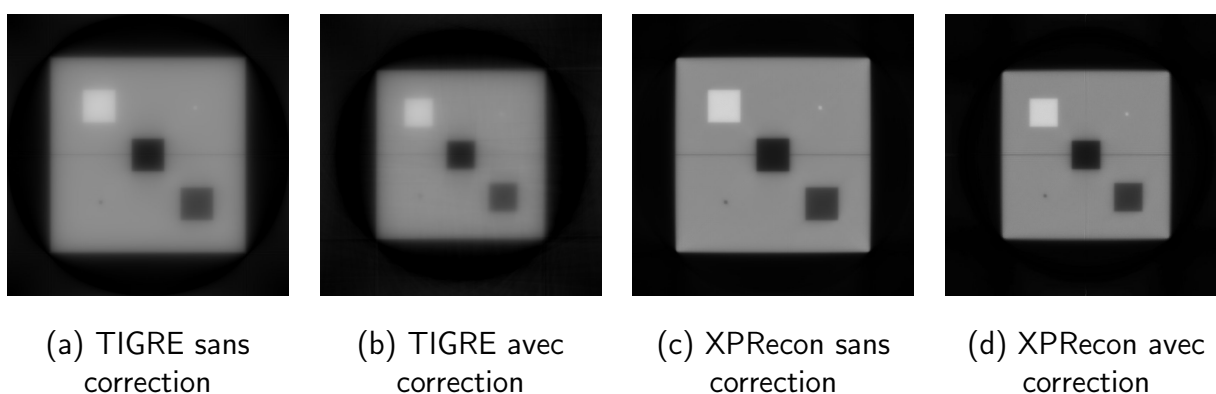


FIGURE 6.9 – Couche de la reconstruction pour TIGRE et XPREcon avec et sans correction de l'axe de la position de la source

6.5.3 Rotation du détecteur

Les rotations du détecteur peuvent aussi engendrer des problèmes lors de la reconstruction. Si la rotation est faite autour des axes x et y du détecteur, alors on peut à la place corriger la position de la source, car la déformation est identique. Le cas de la rotation autour de l'axe de projection est lui plus grave, car cela revient à avoir un décalage de l'axe de rotation différent le long de la hauteur du volume reconstruit. Si ce décalage n'est pas corrigé, un artefact similaire à un décalage de l'axe de rotation, mais évolutif le long de l'axe y sera visible.

6.5.4 Détecteur segmenté

Le détecteur segmenté de prototype du système XPulse est également une source de problèmes d'alignements, car chaque module du détecteur doit être correctement positionné et orienté lors de la création de la géométrie. Pour visualiser l'effet causé par un tel détecteur et la correction appliquée dans le programme de reconstruction, nous avons utilisé une acquisition réelle de quatre tiges de bois se projetant chacune sur chaque module du détecteur durant la rotation [6.12](#).

La figure [6.13a](#) montre une couche de la reconstruction avec la géométrie correcte, mais sans prendre en compte la courbure du détecteur. On remarque que les tiges ne sont

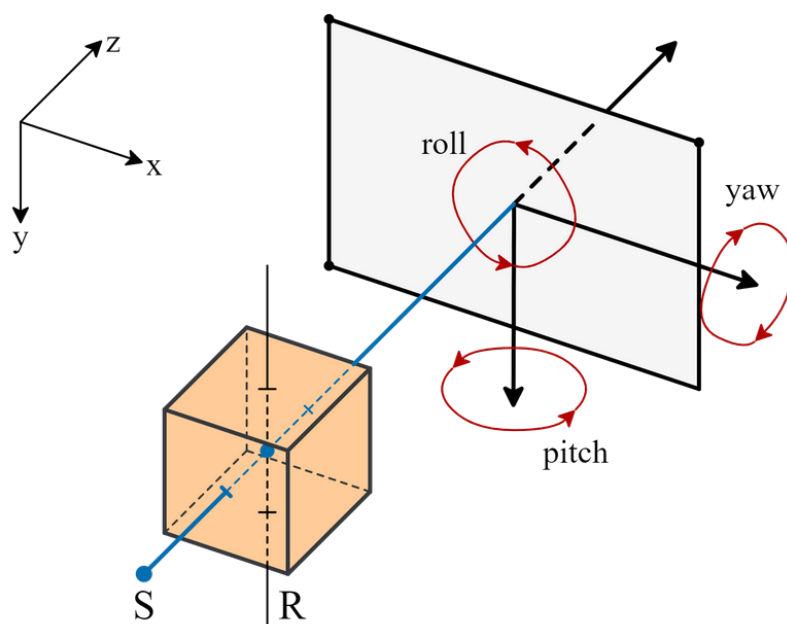


FIGURE 6.10 – Schéma des trois rotations possibles d'un détecteur : le *roll* autour de l'axe de projection, le *yaw* autour de l'axe droite-gauche du détecteur et le *pitch* autour de son axe haut-bas.

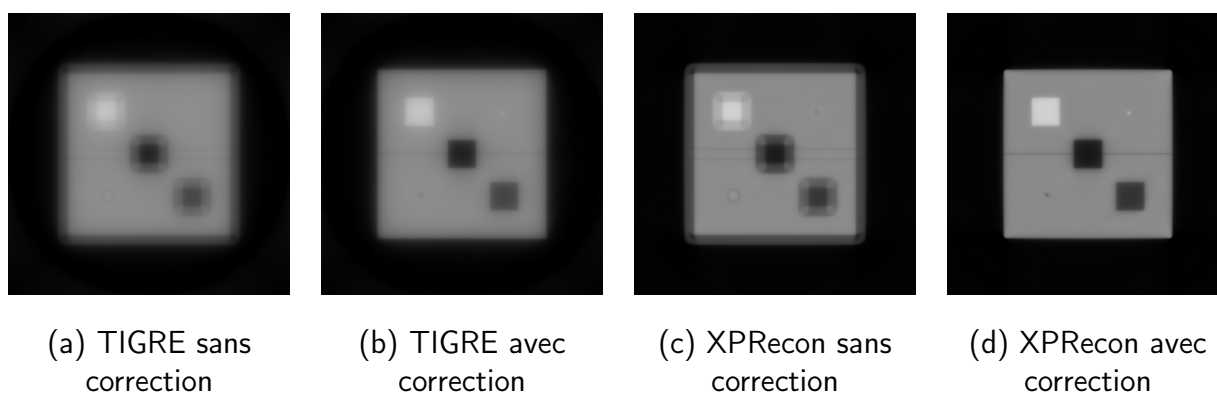


FIGURE 6.11 – Couche de la reconstruction pour TIGRE et XPREcon avec et sans correction de la rotation du détecteur.

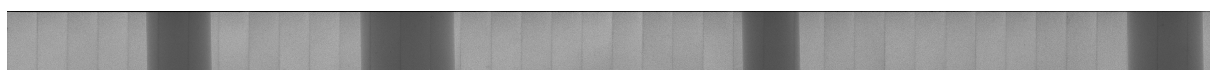


FIGURE 6.12 – Exemple d'une projection du jeu de données utilisé pour le test de la correction du détecteur non planaire.

pas circulaires comme elles devraient l'être. En figure 6.13b le détecteur est correctement représenté et tous les autres paramètres sont identiques. Cette fois les tiges sont bien circulaires, la couche ne montre plus d'artefact géométrique.

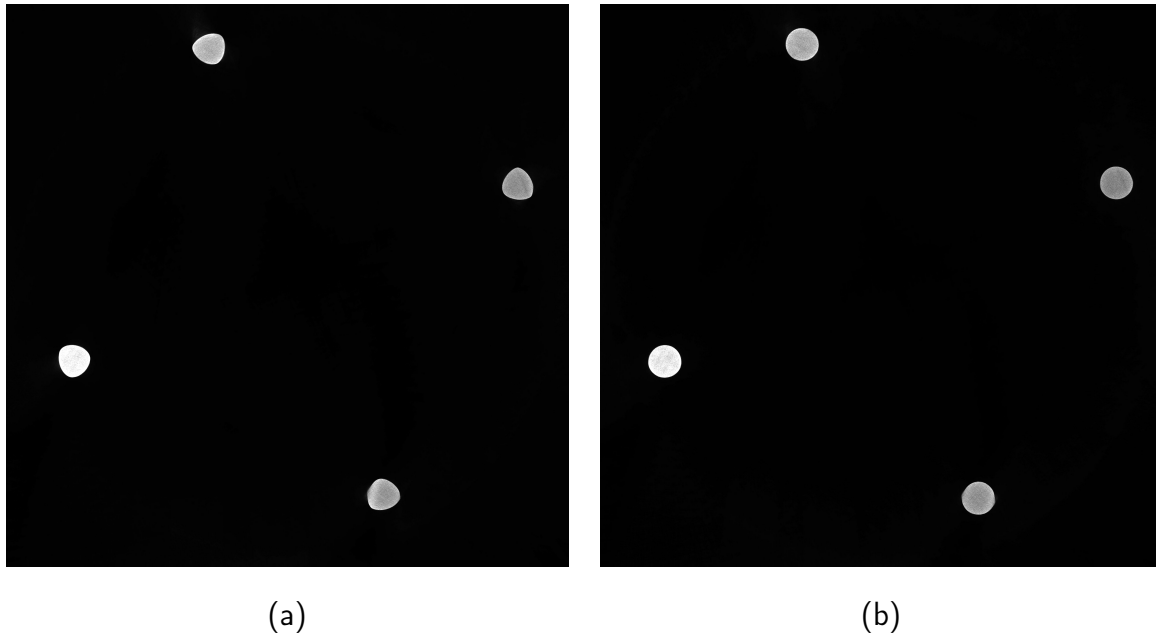


FIGURE 6.13 – Couche de la reconstruction pour TIGRE et XPrecon avec (b) et sans (a) correction de la rotation du détecteur.

6.6 Comparaison des temps de reconstruction

6.6.1 Performance globale

Pour mesurer les performances de notre implémentation par rapport à celle de TIGRE et d’Astra Toolbox, nous avons mesuré le temps de reconstruction pour chacun, et ce pour différentes tailles de volume reconstruit. On peut, de cette manière, mesurer l’évolution du temps de reconstruction et estimer la taille maximale du volume reconstruit pour chaque implémentation. Pour effectuer les relevés, nous avons utilisé une machine équipée d’un Intel Core i9 de 12^{ème} génération (6 cœurs P et 8 cœurs E) dont les cœurs E à économie d’énergie ont été désactivés, avec 32 Go de mémoire vive. Le GPU utilisé est une NVidia GeForce RTX3070ti avec 8 Go de mémoire vidéo. Comparé à un GPU de *workstation*, ce GPU est basé sur la même architecture que la Nvidia A100 (Ampere) et délivre des performances similaires en FP32 (6144 coeurs à 1700Mhz pour la RTX3070ti contre 6912 coeurs à 1410Mhz pour la A100, avec une puissance en TFLOPS annoncé à 21.75 pour la première et 19.5 pour la seconde). Utiliser une carte graphique de gamme supérieure (Quadro ou Tesla) permettrait d’accéder à une très large augmentation de la puissance de calcul en double précision (de l’ordre de 32x), mais je ne réalise aucun calcul avec une telle précision pour la reconstruction tomographique. Les temps mesurés sont la durée de la reconstruction pour deux itérations de reconstruction tomographique, en excluant le temps pour la lecture initiale et l’écriture finale des données (Astra Toolbox et TIGRE ne gèrent pas cette partie en interne). La géométrie est celle utilisée de manière standard avec notre installation.

Les résultats obtenus sont compilés dans le graphique 6.14. En termes de temps d’exécution, les trois implémentations obtiennent des résultats comparables, notre méthode parvenant à être tout de même systématiquement plus rapide. On peut noter que cette méthode de test n’est pas en la faveur d’Astra Toolbox en terme de temps de calcul, car il

demande un coût de précalcul important pour toute reconstruction, mais exécuté ensuite ses itérations de manière très efficace : il sera donc toujours plus rapide que TIGRE ou XPREcon à partir d'un nombre d'itérations suffisant.

On peut tout d'abord voir les limites de Astra Toolbox, qui peine à reconstruire plus de 300 millions de voxels. Au-delà de cette valeur, il n'est pas possible de lancer la reconstruction, car il ne parvient pas à allouer suffisamment de mémoire vidéo pour fonctionner.

TIGRE quant à lui parvient à reconstruire des données plus volumineuses, jusqu'à environ 2,5 milliards de voxels, grâce à son utilisation de la RAM en plus de la mémoire vidéo. À partir de ce point, son implémentation se heurte à la limite de la taille de la RAM, et comme il se repose sur le mécanisme de *swap* du système, celui-ci ne répond plus correctement et l'exécution ne parvient pas à se finir la plupart du temps.

XPREcon parvient de son côté à reconstruire des volumes bien plus gros, ici testé avec 38,4 milliards de voxels. Le facteur limitant ici n'a pas été ici la mémoire vidéo, mais l'utilisation que fait OpenCL (ou son implémentation par NVidia) de la RAM, car les espaces d'échanges alloués lors de la copie de données vers le GPU ne sont pas libérés immédiatement, ce qui finit par causer une erreur d'allocation. Résoudre ce souci nous permettra de reconstruire des volumes plus grands encore.

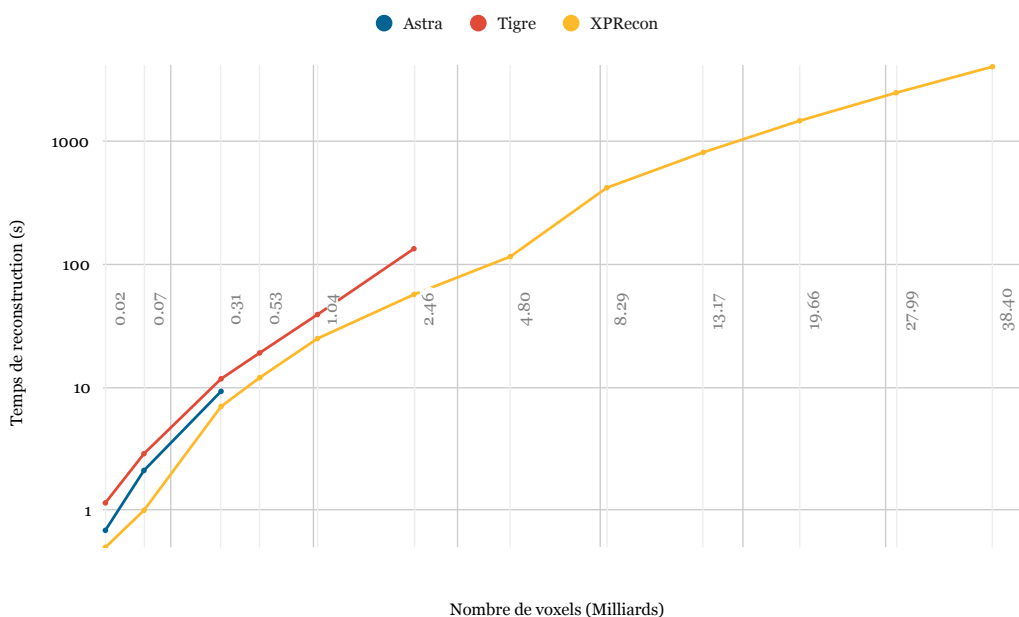


FIGURE 6.14 – Graphique du temps de reconstruction pour trois implémentations de reconstruction tomographique en fonction du nombre de voxels à reconstruire pour un nombre d'itérations défini.

6.6.2 Utilisation IO

XPREcon utilisant le disque pour stocker ses données durant la reconstruction, nous avons également mesuré la quantité de données écrites et lues sur le disque. XPREcon est montré ici avec et sans compression (sans perte si utilisée), et en ne stockant aucune

partie du volume en RAM. Comme ces valeurs varient avec le nombre d'itérations pour XPrecon, nous affichons ici la quantité totale pour 10 itérations.

La quantité d'opérations sur le disque dur pour les différents algorithmes peut être vue dans la figure 6.15. Nous pouvons constater que, TIGRE, Astra Toolbox et notre implémentation naïve sont identiques à cet égard. Pour XPrecon, on voit l'intérêt de la compression, qui permet de conserver une quantité raisonnable d'accès au disque et sans laquelle le temps de copie deviendrait un facteur limitant de la reconstruction. Il faut tout de même noter que le niveau de compression n'est pas constant, différents jeux de données permettront des gains plus ou moins importants.

Ces tests ont été réalisés avec un volume de 410 Mo et un jeu de projection de 310 Mo. Les quantités de lecture et écriture évoluent de manière linéaire avec la taille des données.

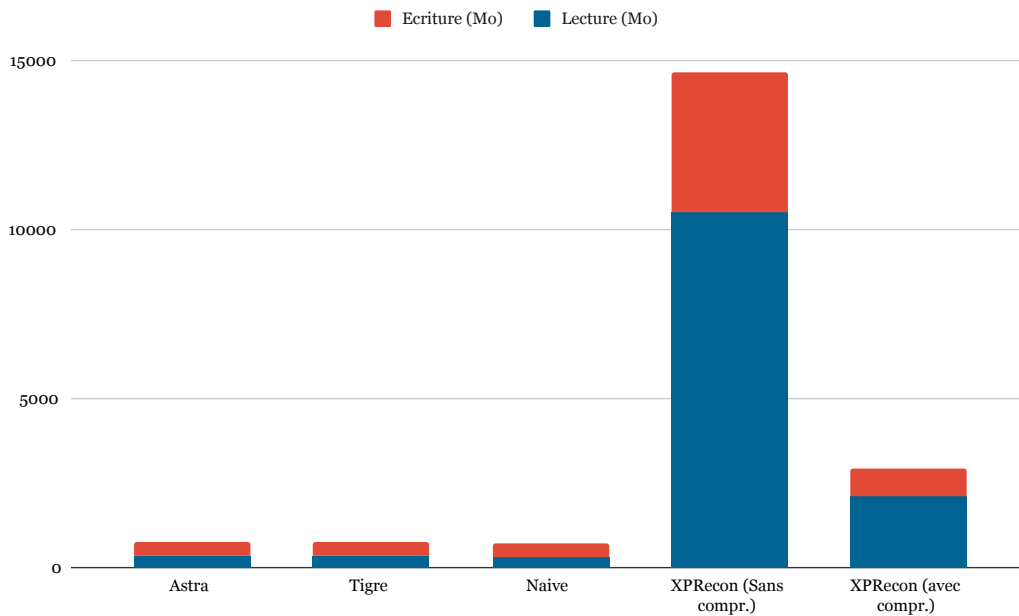


FIGURE 6.15 – Quantité de données écrite et lue sur le stockage durant l'exécution de différentes implémentations de reconstruction tomographique.

6.6.3 Impact de la projection par voxel

Le choix d'utiliser une projection par matrice impose l'utilisation d'une addition atomique lors de l'écriture sur une projection. Pour évaluer l'impact de ce choix sur les performances de l'implémentation, nous avons mis en place un environnement de test permettant de forcer les probabilités de collision lors de l'écriture sur une projection à une valeur souhaitée. Uniquement l'opérateur de projection avant est testé, étant donné que ce problème ne se pose pas pour l'opération inverse, qui écrit uniquement sur le volume.

La figure 6.16 montre les performances du projecteur avant avec une probabilité de collision croissante, comparées à une version avec aucune collision et l'exécution normale de l'algorithme. On voit qu'au-delà de $1/256$, le temps d'exécution est indiscernable de celui de la version sans addition atomique, ainsi que de celui de l'algorithme normal.

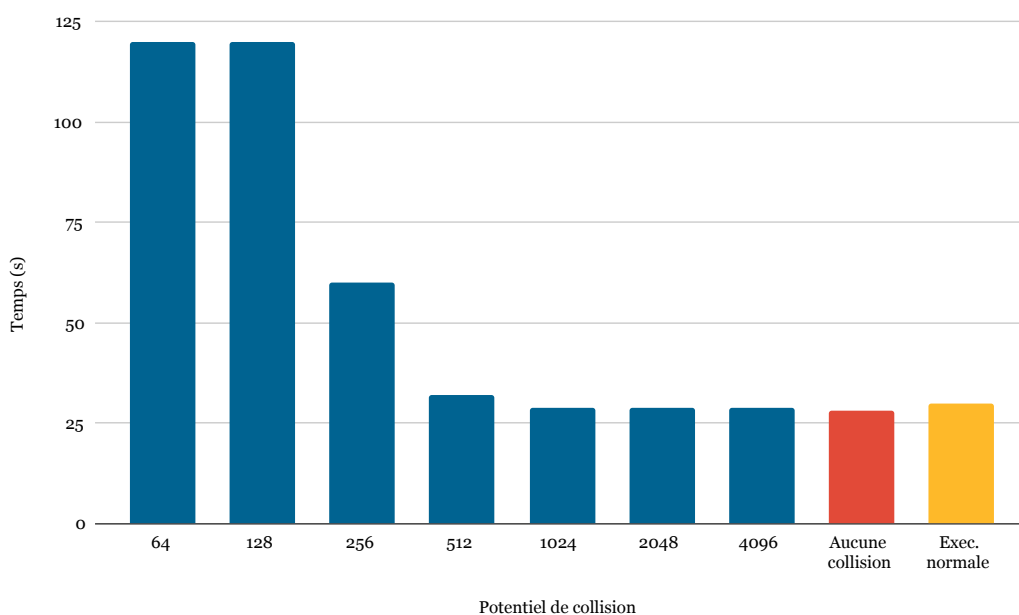


FIGURE 6.16 – Temps d’exécution de la projection de XPrecon avec un potentiel défini de collision sur l’addition atomique. Le potentiel de collision représente le nombre de pixels sur chaque image, 64 signifie donc que chaque voxel a une chance sur 64 de se projeter sur le même pixel qu’un autre.

6.6.4 Performance avec différents GPUs

Un avantage dû à l’utilisation d’OpenCL à la place de CUDA, utilisé dans la majorité des implémentations de reconstruction tomographique, est que la nôtre peut fonctionner sur des GPU de constructeurs autres que NVidia. Nous pouvons nous assurer que l’implémentation présente des performances correctes sur un GPU AMD en comparant au résultat obtenu avec un GPU NVidia.

Sur le graphique 6.17, nous comparons le temps d’exécution entre une NVidia GeForce RTX3090 et une AMD Radeon RX580. La seule modification des paramètres est le passage de la largeur du front de 32 à 64, cette valeur étant une propriété intrinsèque de l’architecture du GPU. Les résultats obtenus sont cohérents avec les estimations de performance des deux GPU se trouvant en ligne (score 3DMark, PassMark, ...), à savoir un facteur deux en puissance de calcul.

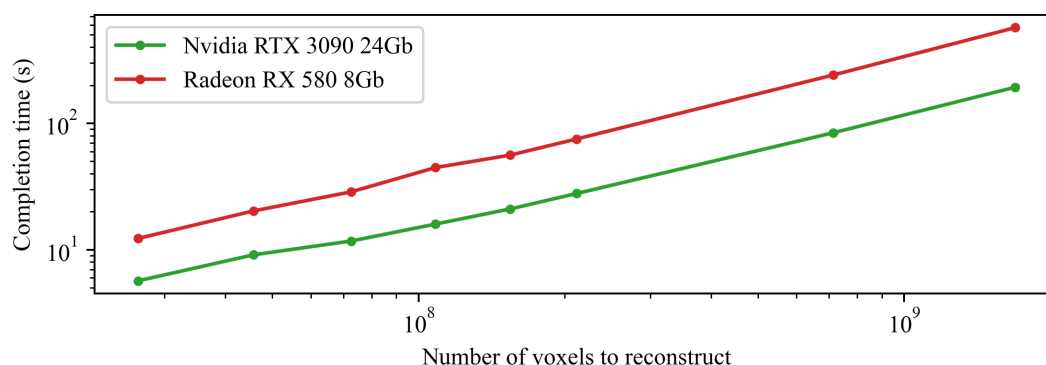


FIGURE 6.17 – Temps de reconstruction de XPREcon en fonction du nombre de voxels à reconstruire sur une carte NVidia et une AMD.

6.7 Données réelles

Dans cette section, je vais présenter une série de trois jeux de données représentant les cas d'utilisation du système d'imagerie XPulse durant ma thèse.

6.7.1 Escargot

Ce premier jeu de données est celui d'une coquille d'escargot acquise avec le système d'imagerie avec l'installation de test à ALPhANOV. C'est un jeu de données lourd acquis en propagation libre, mais reconstruit en absorption. Les projections de la table 6.7 sont de très bonne qualité avec un niveau de bruit faible et un bon niveau d'absorption, et 600 angles de capture sont utilisés pour la tomographie. Quatre couches du volume sont présentées en table 6.8, et un rendu du volume fait avec ParaView est présenté en table 6.9. Du fait du volume de données, le rendu est effectué avec une définition réduite.

6.7.2 Noix

Le second jeu de données est l'acquisition d'une noix prise à l'aide de la source X expérimentale par laser. La figure 6.10 illustre quelques projections de cette acquisition. Du fait de la faible puissance du laser utilisé dans le test, les projections acquises ont un niveau de bruit élevé. Ce niveau de bruit est toujours visible sur la reconstruction (figure 6.11), mais celle-ci est tout de même suffisante pour effectuer une segmentation simple, réalisée avec Amira Software de ThermoFisher en figure 6.12.

6.7.3 Fantôme de sein

Ce troisième jeu de données est une tomographie hélicoïdale d'un fantôme de sein acquis avec l'installation de l'IBIO. Le fantôme est imprimé en 3D en plusieurs parties, avec des inserts d'hydroxyapatite et d'oxalate de calcium. La géométrie hélicoïdale avec le détecteur segmenté non planaire permet de tester la géométrie finale du projet XPulse. 2877 projections sur 14.3 tours ont été acquises (figure 6.13), avec un déplacement vertical régulier de 130 mm. Un rendu avec ParaView est présenté en table 6.15 à partir du volume reconstruit (table 6.14), et une segmentation réalisée avec Amira est illustrée en figure 6.18.

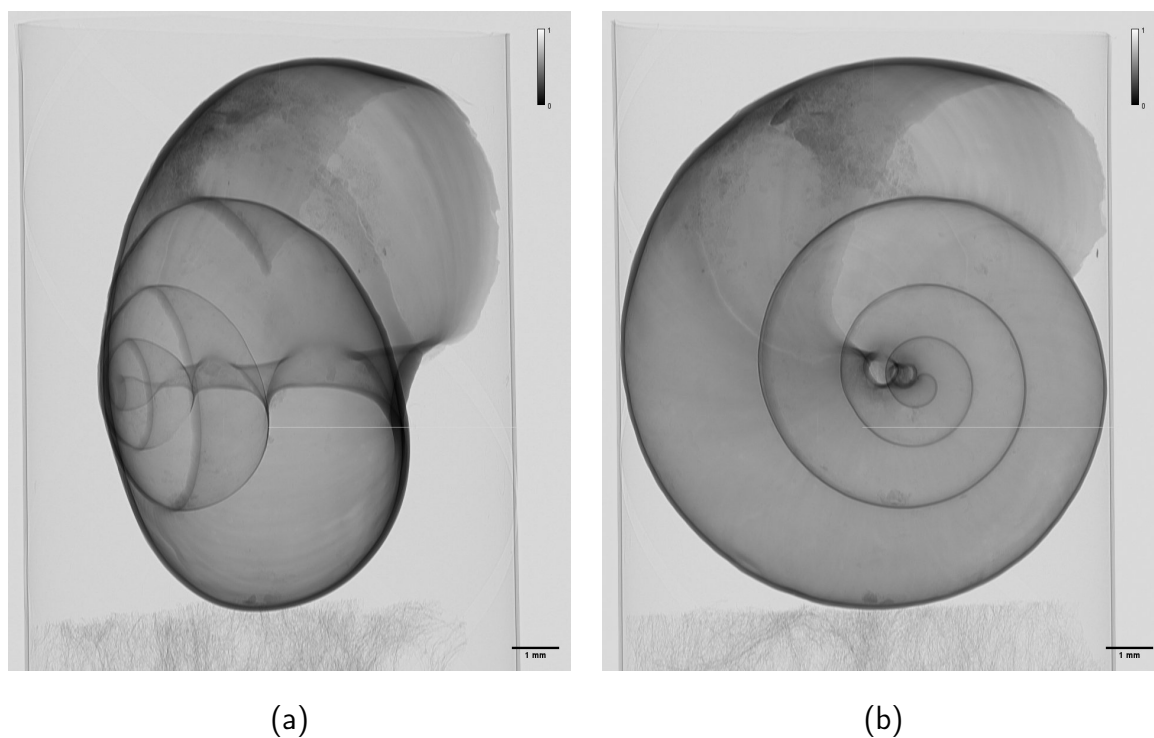


TABLE 6.7 – Exemple de deux projections de la coquille d’escargot



FIGURE 6.18 – Segmentation de trois matériaux du fantôme de sein. En rouge le support en bois, en bleu le plastique imprimé en 3D, et en magenta les inserts d’hydroxyapatite ($\text{Ca}_5(\text{PO}_4)_3(\text{OH})$) et d’oxalate de calcium (CaC_2O_4).

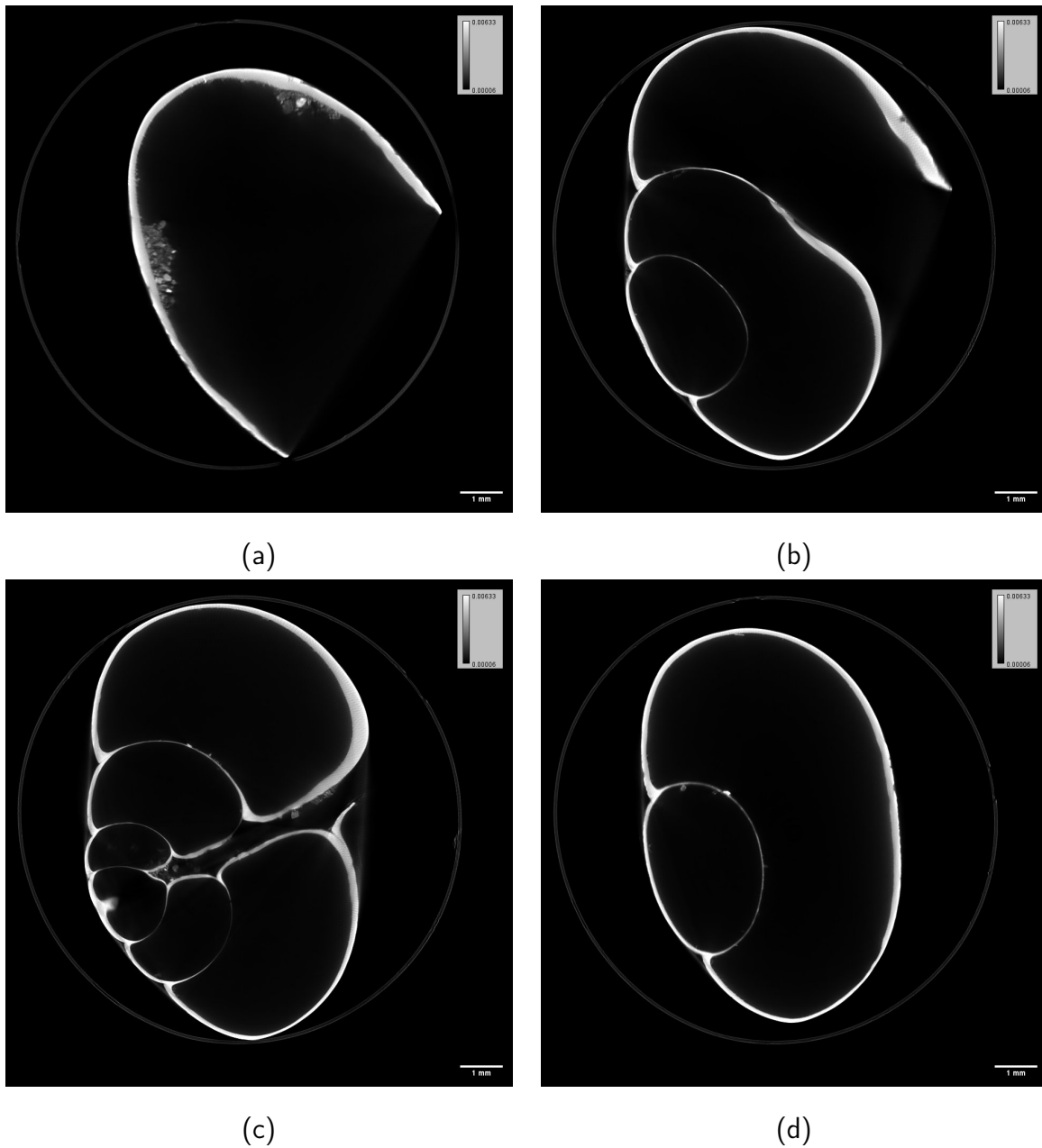


TABLE 6.8 – Exemple de quatre couches extraites du volume reconstruit de la coquille d’escargot.

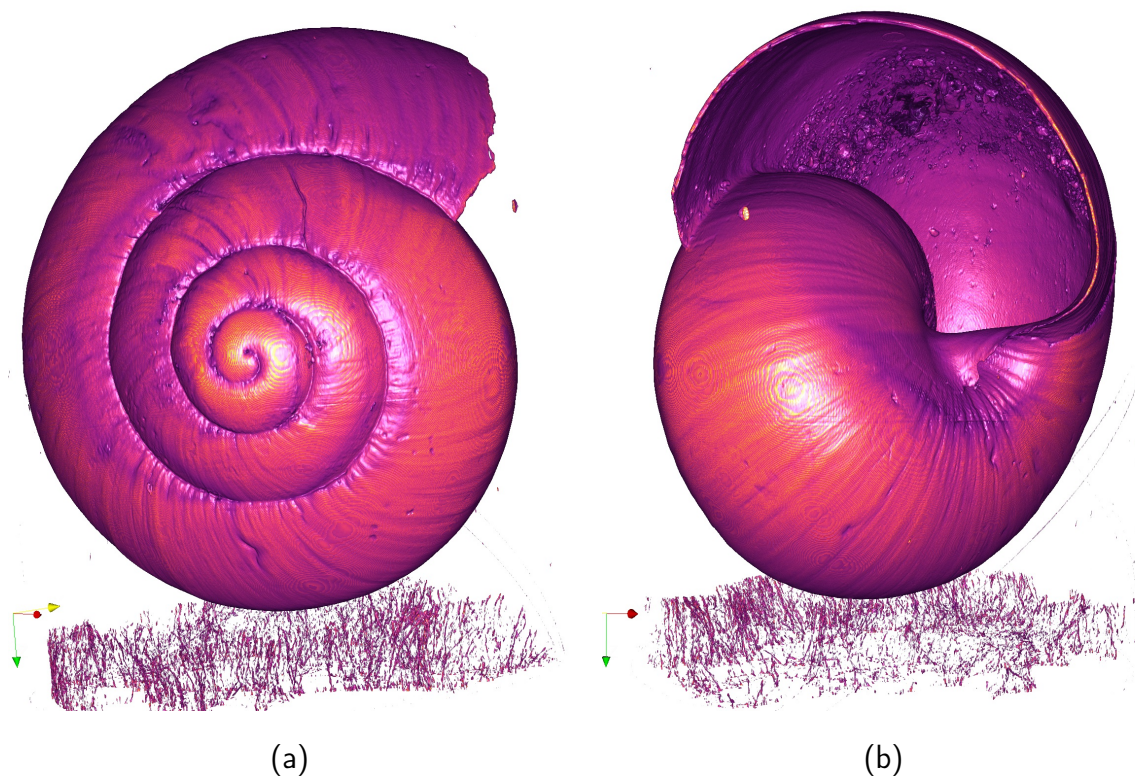


TABLE 6.9 – Rendu du volume reconstruit, réalisé avec ParaView.

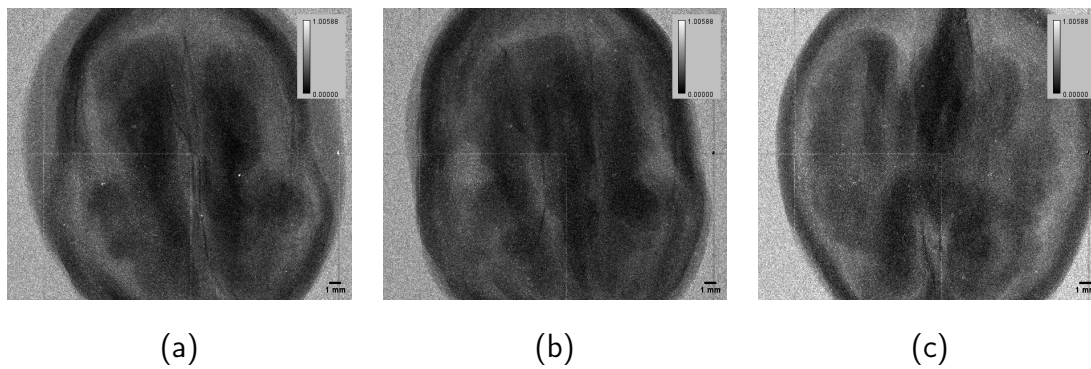


TABLE 6.10 – Exemple de projection de l'acquisition de la noix

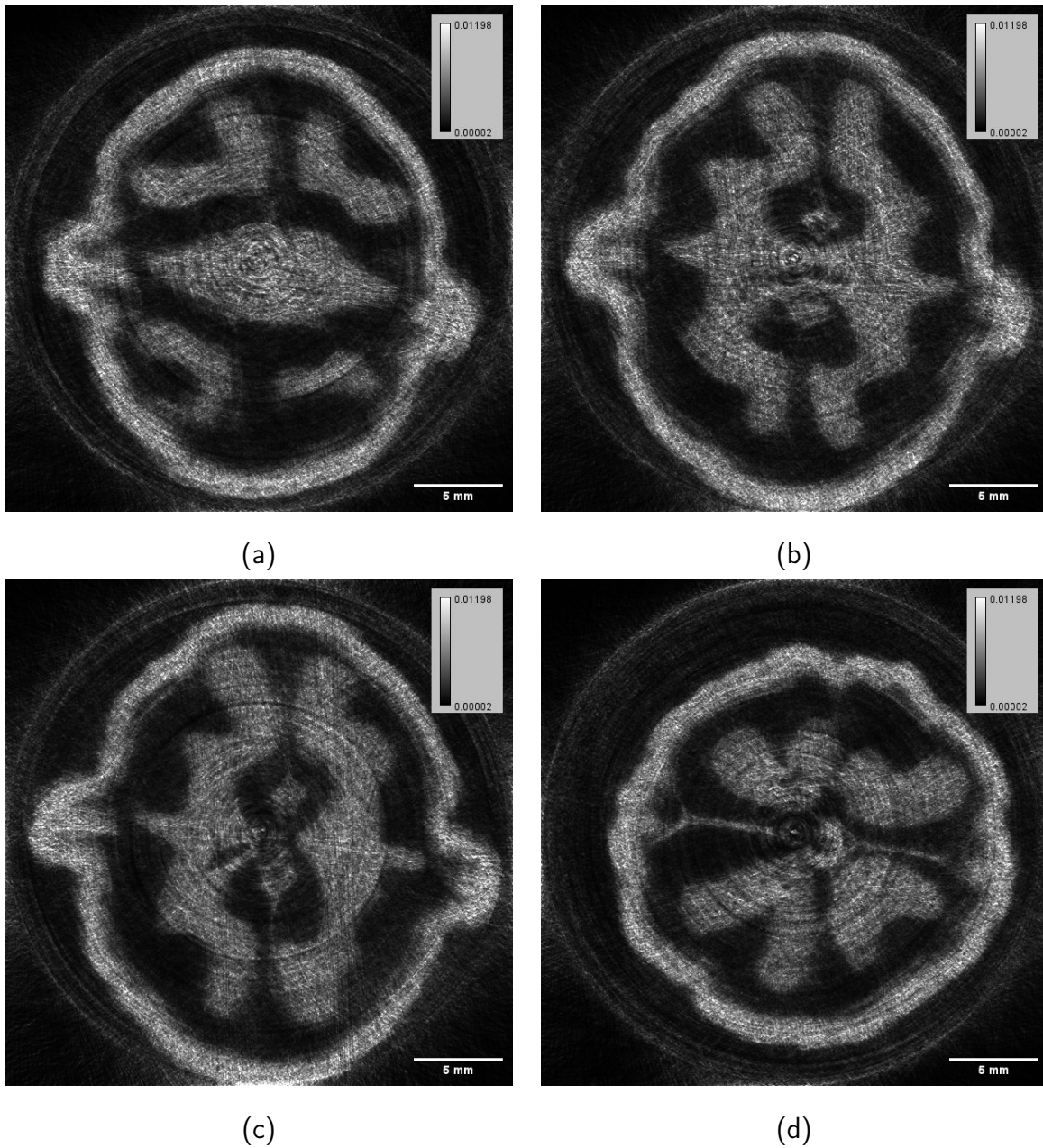


TABLE 6.11 – Exemple de couches de la reconstruction tomographique de la noix

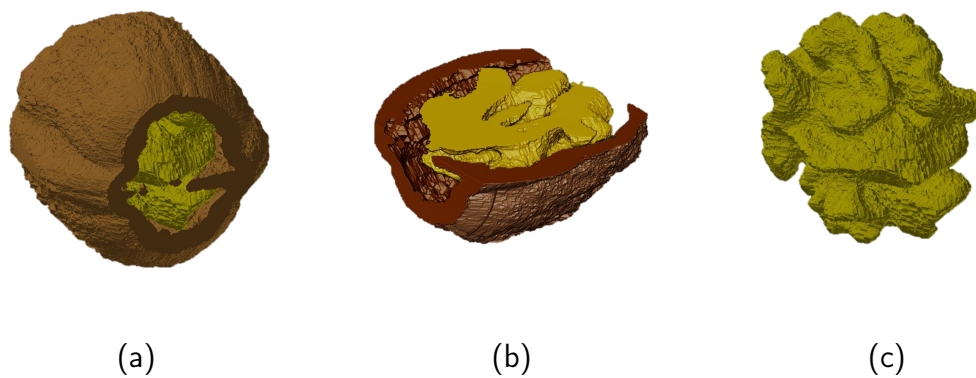


TABLE 6.12 – Segmentation de la coquille des cerneaux à partir de la reconstruction de la noix.

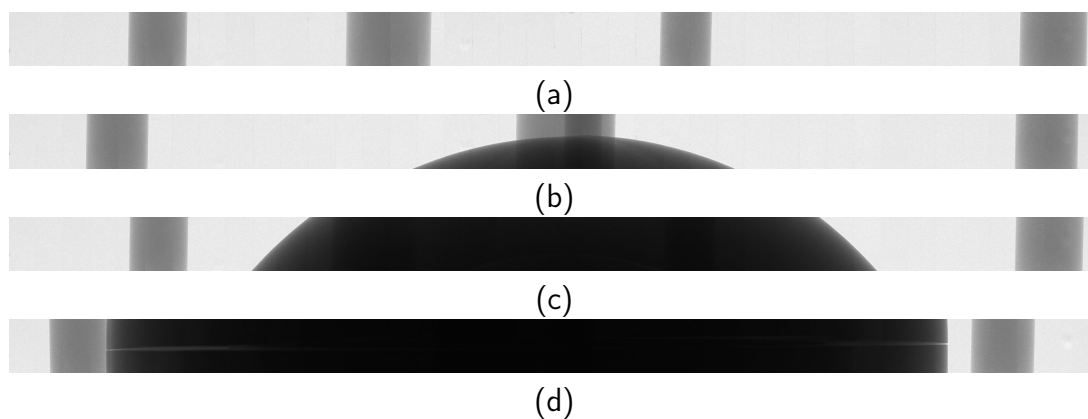


TABLE 6.13 – Exemple de quatre projections normalisée du fantôme de sein.

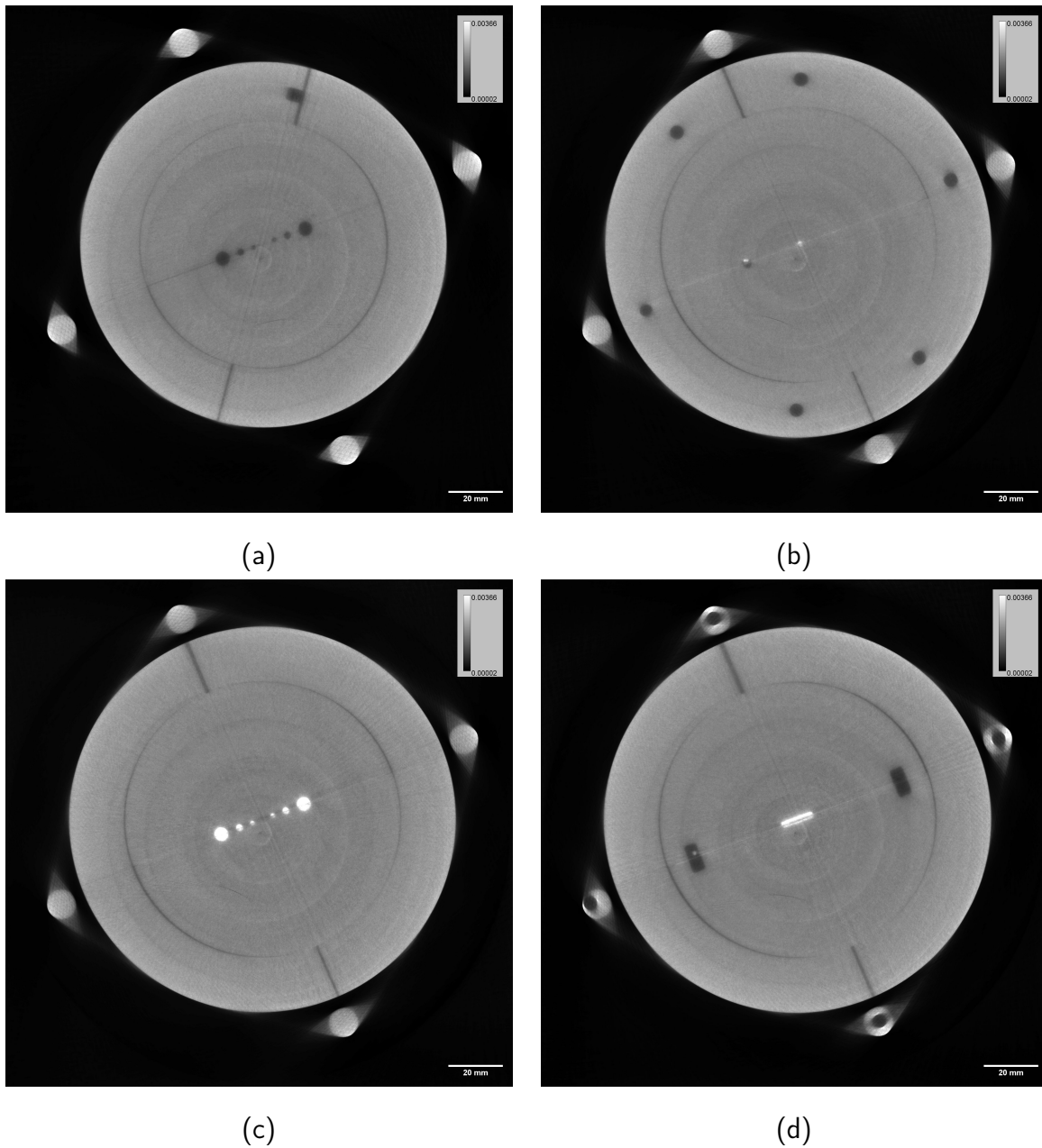


TABLE 6.14 – Exemple de quatre couches de la reconstruction tomographique du fantôme de sein

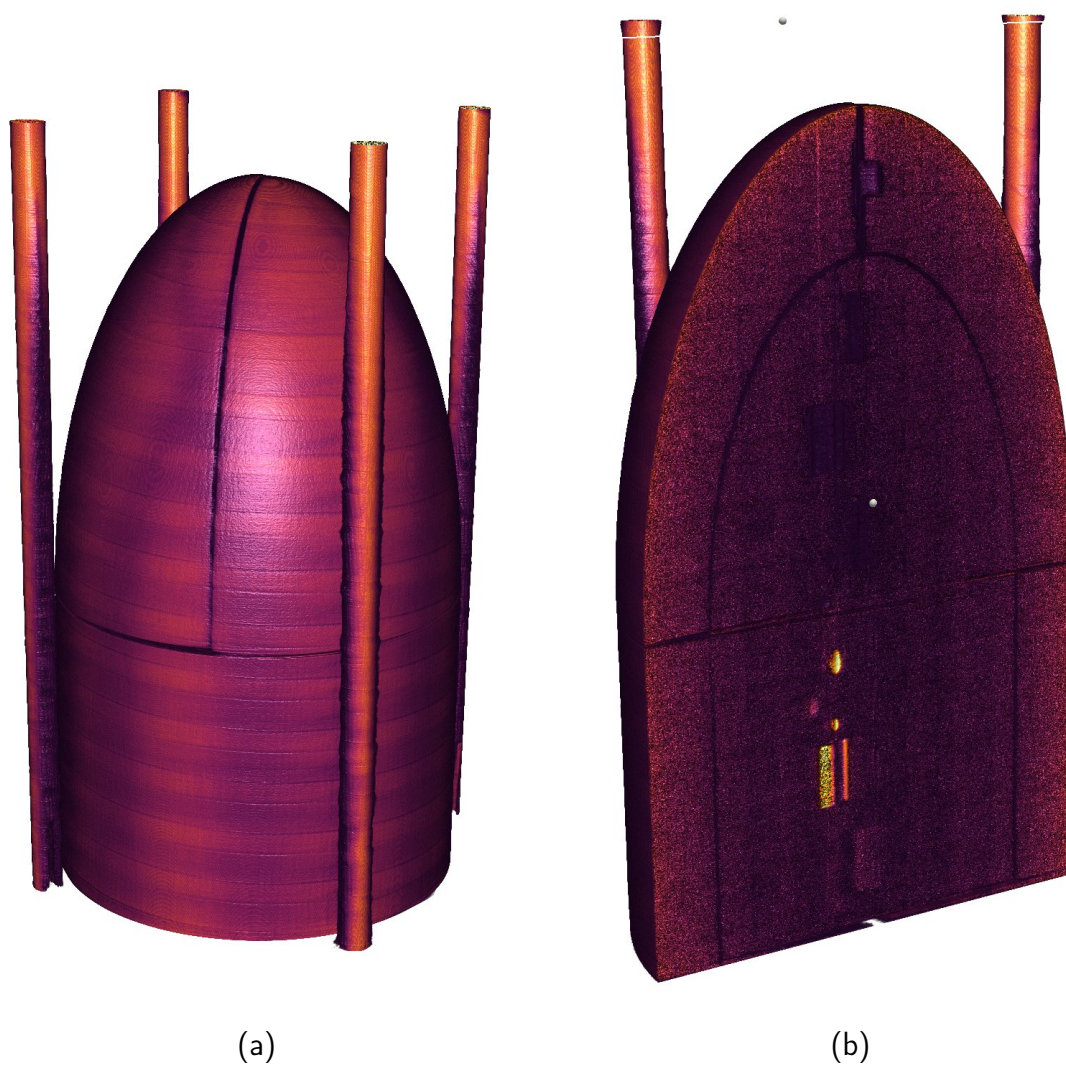


TABLE 6.15 – Images de rendu 3D du volume reconstruit du fantôme de sein

6.8 Conclusion

Dans ce chapitre, nous avons évalué les performances de l'algorithme de récupération de phase et de reconstruction tomographique implémentées.

L'automatisation de l'extraction du ratio de Paganin automatique est complexe à réaliser sur les projections, et bien qu'une formule analytique semble montrer la faisabilité de cette technique, il reste encore du travail pour parvenir au résultat souhaité. Les tests préliminaires sont tout de même prometteurs et ce sujet sera davantage travaillé par la suite.

Pour l'imagerie de phase par masque de Hartmann, notre algorithme d'intégration de phase itérative combiné à la reconstruction tomographique itérative fournit des résultats supérieurs à ceux de l'algorithme existant lorsqu'il est utilisé avec un ensemble de données tomographiques en utilisant les informations de toutes les projections dans le processus itératif.

Combiné à notre extraction de cartes de déviation basée sur les pixels, nous pouvons reconstruire un tomogramme lié à la phase avec une haute définition en utilisant un détecteur standard en quelques étapes de traitement et sans sacrifier le temps de calcul.

Cette méthode ajoute cependant de nouveaux paramètres au traitement, le nombre et la fréquence des itérations du front d'onde et de la tomographie. Pour l'instant, nous trouvons ces paramètres de manière expérimentale et une analyse plus approfondie des résultats doit être effectuée pour déterminer leurs valeurs optimales.

Pour la reconstruction tomographique, les optimisations apportées nous permettent effectivement de reconstruire un volume d'une taille supérieure à ce dont TIGRE et Astra Toolbox sont capables, tout en conservant une qualité de reconstruction similaire. Le temps de reconstruction est également similaire à TIGRE, Astra Toolbox conserve un net avantage sur ce point.

Chapitre 7

Conclusion

Mon travail de thèse s'est articulé autour des avancées et des besoins du projet XPulse, dans le but de pouvoir traiter au mieux les images acquises. Pour cela j'ai travaillé en étroite collaboration avec les autres membres du projet qui utilisent mes implémentations ou leurs résultats, afin de prendre en compte leurs retours et d'anticiper les éventuels problèmes qui peuvent être rencontrés.

Dans ce contexte, j'ai donc pu travailler sur les algorithmes de traitement de l'image, particulièrement ceux liés aux radiographies et à l'imagerie de phase, sur l'implémentation et l'optimisation d'un logiciel de reconstruction tomographique, tout en prenant en compte les attentes des personnes travaillant avant et après moi sur la chaîne complète de l'acquisition et de l'utilisation des données.

Pour pouvoir répondre aux problématiques liées au développement du prototype d'imagerie de phase XPulse, j'ai eu à prendre en compte à la fois la physique des rayonnements électromagnétiques et les contraintes spécifiques liées à l'imagerie médicale. Cela m'a permis de mieux comprendre les phénomènes mesurés et le fonctionnement des méthodes de traitement existantes, afin de pouvoir travailler sur des implémentations de ces méthodes et parvenir à traiter correctement les données avec celles-ci.

À la suite de cette thèse, je continuerai de travailler sur plusieurs de mes travaux de thèse dans le cadre de la suite du projet, car j'ai été recruté par ALPhANOV pour travailler notamment sur la suite du projet XPulse, celle-ci ayant été confirmée peu de temps avant la fin de ma thèse.

Pour la partie liée à la reconstruction tomographique, mon travail s'est orienté selon deux axes principaux : la compréhension des algorithmes de reconstruction tomographique, des différentes méthodes de représentation de la géométrie et les spécificités des méthodes de reconstruction itératives d'une part, et l'implémentation haute performance de la reconstruction tomographique d'autre part. Ce dernier point m'a demandé de me former sur le calcul hautement parallélisé et l'utilisation du GPU pour le calcul général, ainsi que sur la gestion des données à une échelle que je n'avais pas encore rencontrée.

J'ai ainsi effectué des recherches pour trouver les méthodes et algorithmes pertinents

pour les différentes problématiques du projet, puis je les ai testés avec des scripts Python et Matlab pour m'assurer de leur intérêt dans notre cas. J'ai enfin implémenté ces algorithmes de manière plus réfléchie à destination d'utilisateurs non-informaticiens, avec pour but de les rendre utilisables facilement, rapidement, et de manière robuste.

Actuellement, mon implémentation est utilisée en routine pour toutes les acquisitions tomographiques réalisée dans le cadre de XPulse. Cette implémentation est indispensable pour l'installation du prototype à l'IBIO car à notre connaissance elle est la seule qui puisse permettre de gérer toutes les spécificités de l'installation (gros volume de données, détecteur segmenté et non planaire).

Pour la suite du projet, je vais être amené à intervenir plus amont dans le processus d'acquisition afin de développer, avec d'autres collaborateurs du projet, une suite logicielle permettant de réaliser l'acquisition de manière robuste et de traiter le jeu de donnée de manière automatisée (stockage, prétraitement, reconstruction, récupération de phase). Le but sera de délivrer un support logiciel permettant de rendre plus facile et rapide l'utilisation de l'installation pour nous-même, mais également pouvant être fourni aux utilisateurs futurs du système d'imagerie du sein XPulse. En m'impliquant plus tôt dans le processus d'acquisition, je pourrai également ajouter des étapes automatiques, par exemple pour une construction automatique ou assistée de la géométrie à l'aide d'une étape de calibration. Je serais également plus apte à prendre en compte des éventuelles erreurs rencontrées avant le traitement des données.

L'algorithme de reconstruction tomographique pourra être amélioré sur certains points, notamment sa capacité à détecter automatiquement les anomalies dans le jeu de données ou dans la géométrie, puis de les corriger ou d'avertir l'utilisateur.

Pour la récupération de phase en propagation libre, il a été difficile de trouver une méthode plus adaptée que la formule de Paganin, qui m'a été indiquée dès le début de ma thèse comme étant la méthode de référence en imagerie médicale.

J'ai travaillé longtemps sur les méthodes itératives, mais je n'ai jamais pu obtenir le résultat que nous souhaitions. Nous en sommes arrivés à la conclusion que cette méthode n'était pas applicable telle quelle à notre cas, et qu'il serait probablement nécessaire d'y apporter des modifications importantes pour parvenir à un résultat exploitable.

L'algorithme de récupération automatique du ratio sur les projections nous a permis d'obtenir des résultats préliminaires intéressants, mais il manque encore de robustesse et il faudra encore travailler dessus pour pouvoir l'utiliser de manière stable.

Il s'agirait tout d'abord, de développer ou trouver une implémentation existante capable de simuler plus précisément le front d'onde pour un objet simple. À l'aide de cette simulation, je pourrais vérifier étape par étape ma dérivation et tenter de corriger mon calcul de la prise en compte de l'absorption pour le calcul du ratio.

Une fois cette problématique de l'absorption mieux maîtrisée, l'algorithme itératif de calcul du ratio pourra être amélioré avec une condition d'arrêt plus précise et robuste, par exemple avec une détection automatique des zones d'interférence. En effet, la précision du résultat final est directement liée à celle de la condition d'arrêt.

Il est prévu que ce travail soit mené dans le cadre de l'un des axes du projet XPulse visant à réaliser des radiographies de biopsies en contraste de phase. Le but recherché est la discrimination morphologique et chimique des calcifications présentes dans une biopsie afin de reconnaître les calcifications potentiellement malignes, toujours dans le cadre d'un dépistage plus efficace du cancer du sein.

Enfin, mon algorithme de reconstruction simultanée du front d'onde et tomographique pour l'imagerie par masque de Hartmann est utilisé par l'équipe pour traiter toutes les données de ce type, car il offre un résultat supérieur aux méthodes utilisées précédemment tout en regroupant les deux étapes principales du traitement en un seul algorithme.

Mon algorithme semble robuste et délivre de bons résultats, mais il faudra encore le tester sur davantage de jeux de données. Un des points critiques est la localisation des sous-faisceaux, en prenant en compte les mouvements du système d'acquisition. L'algorithme basé sur le suivi des points me paraît être prometteur, car il permet de détecter plusieurs sources d'erreur et de les corriger, tout en se passant de flatfields.

J'ai donc de nombreux axes sur lesquels travailler à la suite de ma thèse sur ce projet, à la fois sur la continuation des travaux que j'y ai engagés, et sur de nouvelles thématiques. Le système d'acquisition continuant d'évoluer au fil des développements, j'aurai également à prendre en compte les nouvelles contraintes qui émergeront nécessairement à mesure que l'on se rapprochera d'une utilisation en clinique de notre technologie, ce qui est l'objectif visé.

Enfin, il est possible que le volet du projet lié à l'acquisition multi-énergie, qui a été mise de côté pour XPulse 1, soit réétudiée pour la suite du projet. Il serait alors nécessaire de reprendre toutes les étapes de l'acquisition, avec notamment un travail important autour du traitement des données.

A l'issue de ce travail de thèse, j'ai également pu identifier un certain nombre de sujets de recherche pouvant ouvrir de nouvelles possibilités quant à l'imagerie de phase.

En ce qui concerne la reconstruction tomographique des images de phase, il me semblerait pertinent d'étudier davantage les propriétés des données obtenues après traitement afin de déterminer si une méthode de reconstruction tomographique statistique pourrait être développée ou adaptée à nos conditions particulières. L'utilisation des méthodes statistique permet généralement de converger plus rapidement, et sont plus performantes dans le cas de données très bruitées.

Une autre piste serait d'utiliser les avancées en matière d'apprentissage par réseaux de neurones pour améliorer certains points de traitement.

Par exemple pour la propagation libre, la méthode de Paganin pourrait éventuellement être remplacée par un réseau ne nécessitant pas de connaissance a priori sur l'objet imagé. Un réseau de neurones pourrait alors approximer directement le ratio d'une interface donnée en lui fournissant des informations sur les conditions de l'acquisition.

L'apprentissage a également le potentiel de nous faire progresser sur l'aspect de la récupération de phase itérative.

Dans tous ces cas, le développement d'un modèle de simulation de l'ensemble de la chaîne d'acquisition de notre installation nous permettrait de générer les données nécessaires à l'apprentissage de tels réseaux.

Bibliographie

- [1] M. RAWLIK, A. PEREIRA, S. SPINDLER et al., « Refraction beats attenuation in breast CT », *Optica*, 2023. DOI : [10.48550/arXiv.2301.00455](https://doi.org/10.48550/arXiv.2301.00455).
- [2] C. L. WINGATE, W. GROSS et G. FAILLA, « Experimental Determination of Absorbed Dose from X-Rays Near the Interface of Soft Tissue and Other Material », *Radiology*, t. 79, p. 984-1000, 1962. DOI : [10.1148/79.6.984](https://doi.org/10.1148/79.6.984).
- [3] N. JEAN-CLAUDE, B. JEAN, L. DOMINIQUE et al., « ICRP Publication 103. The 2007 Recommendations of the International Commission on Radiological Protection », *Institut de Radioprotection et de Surete Nucleaire*, 2007. DOI : [10.1016/j.icrp.2007.10.003](https://doi.org/10.1016/j.icrp.2007.10.003).
- [4] H. TODA, *X-Ray CT*. Springer Singapore, 2021. DOI : [10.1007/978-981-16-0590-1](https://doi.org/10.1007/978-981-16-0590-1). adresse : <https://books.google.fr/books?id=v4kiEAAAQBAJ>.
- [5] M. REHANI, R. GUPTA, S. BARTLING et al., « ICRP Publication 129 : Radiological Protection in Cone Beam Computed Tomography (CBCT) », *Annals of the ICRP*, t. 44, p. 7-127, 2015. DOI : [10.1177/0146645315575485](https://doi.org/10.1177/0146645315575485).
- [6] J. RADON, « On the determination of functions from their integral values along certain manifolds », *IEEE Transactions on Medical Imaging*, t. 5, p. 170-176, 1986. DOI : [10.1109/tmi.1986.4307775](https://doi.org/10.1109/tmi.1986.4307775).
- [7] N. NORMAND et J.-P. GUÉDON, « La transformée Mojette : une représentation redondante pour l'image », *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, t. 326, p. 123-126, 1998. DOI : [10.1016/s0764-4442\(97\)82724-3](https://doi.org/10.1016/s0764-4442(97)82724-3).
- [8] L. A. SHEPP et B. F. LOGAN, « The Fourier reconstruction of a head section », *IEEE Transactions on Nuclear Science*, t. 21, n° 3, p. 21-43, 1974. DOI : [10.1109/tns.1974.6499235](https://doi.org/10.1109/tns.1974.6499235).
- [9] H. VILLARRAGA-GÓMEZ et S. T. SMITH, « Effect of the number of projections on dimensional measurements with X-ray computed tomography », *Precision Engineering*, t. 66, p. 445-456, 2020. DOI : [10.1016/j.precisioneng.2020.08.006](https://doi.org/10.1016/j.precisioneng.2020.08.006). adresse : <https://www.sciencedirect.com/science/article/pii/S0141635920302944>.
- [10] P. STEFANOV, *The Radon transform with finitely many angles*, 2022. DOI : [10.48550/ARXIV.2208.05936](https://doi.org/10.48550/ARXIV.2208.05936). adresse : <https://arxiv.org/abs/2208.05936>.
- [11] S.-R. ZHAO, D. JIANG, K. YANG et K. YANG, « Generalized fourier slice theorem for cone-beam image reconstruction », *Journal of X-Ray Science and Technology*, t. 23, p. 157-188, 2015. DOI : [10.3233/xst-150479](https://doi.org/10.3233/xst-150479).

-
- [12] L. A. FELDKAMP, L. C. DAVIS et J. W. KRESS, « Practical cone-beam algorithm », *Journal of the Optical Society of America A*, t. 1, p. 612, 1984. DOI : [10.1364/JOSAA.1.000612](https://doi.org/10.1364/JOSAA.1.000612).
- [13] R. GORDON, R. BENDER et G. T. HERMAN, « Algebraic Reconstruction Techniques (ART) for three-dimensional electron microscopy and X-ray photography », *Journal of Theoretical Biology*, t. 29, p. 471-481, 1970. DOI : [10.1016/0022-5193\(70\)90109-8](https://doi.org/10.1016/0022-5193(70)90109-8).
- [14] R. GORDON, « A tutorial on art (algebraic reconstruction techniques) », *IEEE Transactions on Nuclear Science*, t. 21, p. 78-93, 3 1974. DOI : [10.1109/tns.1974.6499238](https://doi.org/10.1109/tns.1974.6499238).
- [15] A. H. ANDERSEN et A. C. KAK, « Simultaneous Algebraic Reconstruction Technique (SART) : A Superior Implementation of the Art Algorithm », *Ultrasonic Imaging*, t. 6, p. 81-94, 1984. DOI : [10.1177/016173468400600107](https://doi.org/10.1177/016173468400600107).
- [16] K. MUELLER et R. YAGEL, « Rapid 3-D cone-beam reconstruction with the simultaneous algebraic reconstruction technique (SART) using 2-D texture mapping hardware », *IEEE Transactions on Medical Imaging*, t. 19, p. 1227-1237, 12 2000. DOI : [10.1109/42.897815](https://doi.org/10.1109/42.897815). adresse : <http://www.cs.sunysb.edu/~mueller/papers/ieeeTexMapCT.pdf>.
- [17] P. DEBASISH MISHRA K. Muralidhar, « A ROBUST MART ALGORITHM FOR TOMOGRAPHIC APPLICATIONS », *Numerical Heat Transfer, Part B : Fundamentals*, t. 35, p. 485-506, 4 1999. DOI : [10.1080/104077999275857](https://doi.org/10.1080/104077999275857).
- [18] E. F. OLIVEIRA, S. B. MELO, C. C. DANTAS et al., « Comparison among tomographic reconstruction algorithms with a limited data », 2011, ISBN : 9788599141045.
- [19] P. SUBBARAO, P. MUNSHI et K. MURALIDHAR, « Performance of iterative tomographic algorithms applied to non-destructive evaluation with limited data », *NDT and E International*, t. 30, p. 359-370, 1997. DOI : [10.1016/s0963-8695\(97\)00005-4](https://doi.org/10.1016/s0963-8695(97)00005-4).
- [20] L. A. SHEPP et Y. VARDI, « Maximum Likelihood Reconstruction for Emission Tomography », *IEEE Transactions on Medical Imaging*, t. 1, p. 113-122, 2 1982. DOI : [10.1109/tmi.1982.4307558](https://doi.org/10.1109/tmi.1982.4307558).
- [21] H. HUDSON et R. LARKIN, « Accelerated image reconstruction using ordered subsets of projection data », *IEEE Transactions on Medical Imaging*, t. 13, p. 601-609, 4 1994. DOI : [10.1109/42.363108](https://doi.org/10.1109/42.363108).
- [22] A. KOSTENKO, K. J. BATENBURG, A. KING, S. E. OFFERMAN et L. J. van VLIET, « Total variation minimization approach in in-line x-ray phase-contrast tomography », *Optics Express*, t. 21, n° 10, p. 12185, 2013. DOI : [10.1364/OE.21.012185](https://doi.org/10.1364/OE.21.012185). adresse : <https://opg.optica.org/oe/abstract.cfm?URI=oe-21-10-12185>.
- [23] P. OSKOUI-FARD et H. STARK, « Tomographic image reconstruction using the theory of convex projections », *IEEE Transactions on Medical Imaging*, t. 7, p. 45-58, 1 1988. DOI : [10.1109/42.3928](https://doi.org/10.1109/42.3928).
- [24] D. PAGANIN, S. C. MAYO, T. E. GUREYEV, P. R. MILLER et S. W. WILKINS, « Simultaneous phase and amplitude extraction from a single defocused image of a homogeneous object », *Journal of Microscopy*, t. 206, p. 33-40, 2002, ISSN : 0022-2720.

- [25] W. SOUTHWELL, « Wave-front estimation from wave-front slope measurements », *Journal of the Optical Society of America*, t. 70, p. 998, 8 1980. DOI : [10.1364/josa.70.000998](https://doi.org/10.1364/josa.70.000998).
- [26] O. de LA ROCHEFOUCAULD, G. BEGANI PROVINCIALI, A. CEDOLA et al., « Element differentiation with a Hartmann- based X-ray phase imaging system », *Nondestructive Testing and Evaluation*, t. 37, n° 5, p. 707-720, 2022. DOI : [10.1080/10589759.2022.2095383](https://doi.org/10.1080/10589759.2022.2095383). eprint : <https://doi.org/10.1080/10589759.2022.2095383>. adresse : <https://doi.org/10.1080/10589759.2022.2095383>.
- [27] A. BRAVIN, P. COAN et P. SUORTTI, *X-ray phase-contrast imaging : from pre-clinical applications towards clinics*, 2013. DOI : [10.1088/0031-9155/58/1/r1](https://doi.org/10.1088/0031-9155/58/1/r1).
- [28] M. KRENKEL, M. TÖPPERWIEN, M. BARTELS, P. LINGOR, D. SCHILD et T. SALDITT, « X-ray phase contrast tomography from whole organ down to single cells », in *SPIE Proceedings*, t. 9212, SPIE, 2014. DOI : [10.1117/12.2060390](https://doi.org/10.1117/12.2060390). adresse : <https://bib-pubdb1.desy.de/record/192598/files/PUBDB-2014-04180.pdf>.
- [29] TOHORU, I. YUJI, Y. AKIO et al., « Perspective for Medical Applications of Phase-Contrast X-Ray Imaging », C. A. MASAMI et UYAMA, éd., Springer Japan, 1998, p. 54-62, ISBN : 978-4-431-68485-5.
- [30] U. BONSE et M. HART, « Principles and design of Laue-case X-Ray interferometers », *Zeitschrift fr Physik*, t. 188, p. 154-164, 1965. DOI : [10.1007/bf01339402](https://doi.org/10.1007/bf01339402).
- [31] B. E. INGAL VN., « X-ray plane-wave topography observation of the phase contrast from a non-crystalline object », *J Phys D Appl Phys*, 1995. DOI : [10.1088/0022-3727/28/11/012](https://doi.org/10.1088/0022-3727/28/11/012).
- [32] M.-C. ZDORA, « State of the Art of X-ray Speckle-Based Phase-Contrast and Dark-Field Imaging », *Journal of Imaging*, t. 4, p. 60, 2018. DOI : [10.3390/jimaging4050060](https://doi.org/10.3390/jimaging4050060). adresse : <https://www.mdpi.com/2313-433X/4/5/60/pdf?version=1525349853>.
- [33] K. A. NUGENT, D. PAGANIN et T. E. GUREYEV, « A Phase Odyssey », *Physics Today*, t. 54, p. 27-32, 2001. DOI : [10.1046/j.1365-2818.2002.01010.x](https://doi.org/10.1046/j.1365-2818.2002.01010.x). adresse : <https://physicstoday.scitation.org/doi/pdf/10.1063/1.1404846>.
- [34] T. ZHOU, U. LUNDSTRÖM, T. THÜRING et al., « Comparison of two x-ray phase-contrast imaging methods with a microfocus source », *Optics Express*, t. 21, p. 30 183, 2013. DOI : [10.1364/OE.21.030183](https://doi.org/10.1364/OE.21.030183).
- [35] C. ZUO, J. LI, J. SUN et al., « Transport of intensity equation : a tutorial », *Optics and Lasers in Engineering*, t. 135, p. 106 187, 2020. DOI : <https://doi.org/10.1016/j.optlaseng.2020.106187>. adresse : <https://www.sciencedirect.com/science/article/pii/S0143816619320858>.
- [36] A. BURVALL, U. LUNDSTRÖM, P. A. C. TAKMAN, D. H. LARSSON et H. M. HERTZ, « Phase retrieval in X-ray phase-contrast imaging suitable for tomography », *Optics Express*, t. 19, n° 11, p. 10 359, 2011. DOI : [10.1364/OE.19.010359](https://doi.org/10.1364/OE.19.010359). adresse : <https://opg.optica.org/oe/abstract.cfm?URI=oe-19-11-10359>.
- [37] K. A. NUGENT, « Coherent methods in the X-ray sciences », *Advances in Physics*, t. 59, p. 1-99, 2010, Approximation coaxiale. DOI : [10.1080/00018730903270926](https://doi.org/10.1080/00018730903270926). adresse : <https://arxiv.org/pdf/0908.3064>.

- [38] A. V. BRONNIKOV, « Theory of quantitative phase-contrast computed tomography », *Journal of the Optical Society of America A*, t. 19, p. 472, 2002. DOI : [10.1364/josaa.19.000472](https://doi.org/10.1364/josaa.19.000472).
- [39] M. BOONE, Y. DE WITTE, M. DIERICK, J. VAN DEN BULCKE, J. VLASSENBROECK et L. VAN HOOREBEKE, « Practical use of the modified Bronnikov algorithm in micro-CT », *Nuclear Instruments and Methods in Physics Research Section B : Beam Interactions with Materials and Atoms*, t. 267, p. 1182-1186, 2009. DOI : [10.1016/j.nimb.2009.01.129](https://doi.org/10.1016/j.nimb.2009.01.129). adresse : <https://biblio.ugent.be/publication/671119/file/817565.pdf>.
- [40] M. BELTRAN, D. PAGANIN, K. UESUGI et M. KITCHEN, « 2D and 3D X-ray phase retrieval of multi-material objects using a single defocus distance », *Optics Express*, t. 18, p. 6423, 2010. DOI : [10.1364/oe.18.006423](https://doi.org/10.1364/oe.18.006423).
- [41] T. E. GUREYEV, T. J. DAVIS, A. POGANY, S. C. MAYO et S. W. WILKINS, « Optical phase retrieval by use of first Born- and Rytov-type approximations », *Applied Optics*, t. 43, p. 2418, 2004. DOI : [10.1364/ao.43.002418](https://doi.org/10.1364/ao.43.002418).
- [42] J. R. FIENUP, « Phase retrieval algorithms : a comparison », *Applied Optics*, t. 21, p. 2758, 1982. DOI : [10.1364/AO.21.002758](https://doi.org/10.1364/AO.21.002758).
- [43] A. J. CARROLL, G. A. v. RIESSEN, E. BALAUR, I. P. DOLBANYA, G. N. TRAN et A. G. PEELE, « An iterative method for robust in-line phase contrast imaging », *New Journal of Physics*, t. 18, p. 043003, 2016. DOI : [10.1088/1367-2630/18/4/043003](https://doi.org/10.1088/1367-2630/18/4/043003). adresse : <https://iopscience.iop.org/article/10.1088/1367-2630/18/4/043003/pdf>.
- [44] S. MEHRABKHANI, L. WEFELNBERG et T. SCHNEIDER, « Fourier-based solving approach for the transport-of-intensity equation with reduced restrictions », *Optics Express*, t. 26, p. 11458, 2018. DOI : [10.1364/oe.26.011458](https://doi.org/10.1364/oe.26.011458).
- [45] E. R. SHANBLATT, Y. SUNG, R. GUPTA et al., « Forward model for propagation-based x-ray phase contrast imaging in parallel- and cone-beam geometry », *Optics Express*, t. 27, p. 4504, 2019. DOI : [10.1364/oe.27.004504](https://doi.org/10.1364/oe.27.004504).
- [46] A. PEIN, S. LOOCK, G. PLONKA et T. SALDITT, « Using sparsity information for iterative phase retrieval in x-ray propagation imaging », *Optics Express*, t. 24, p. 8332, 2016. DOI : [10.1364/oe.24.008332](https://doi.org/10.1364/oe.24.008332).
- [47] D. M. PAGANIN, *X-ray interactions with matter*. Oxford University Press, 2006, p. 64-135. DOI : [10.1093/acprof:oso/9780198567288.003.0002](https://doi.org/10.1093/acprof:oso/9780198567288.003.0002).
- [48] U. VYAS et D. CHRISTENSEN, « Ultrasound beam simulations in inhomogeneous tissue geometries using the hybrid angular spectrum method », *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, t. 59, p. 1093-1100, 2012. DOI : [10.1109/tuffc.2012.2300](https://doi.org/10.1109/tuffc.2012.2300).
- [49] A. RUHLANDT, M. KRENKEL, M. BARTELS et T. SALDITT, « Three-dimensional phase retrieval in propagation-based phase-contrast imaging », *Physical Review A*, t. 89, p. 033847, 2014. DOI : [10.1103/physreva.89.033847](https://doi.org/10.1103/physreva.89.033847). adresse : <https://bib-pubdb1.desy.de/record/192619/files/PUBDB-2014-04201.pdf>.

- [50] S. MOHAMMADI, F. BRUN, C. DULLIN, D. DREOSI et G. TROMBA, « A comparison of free software implementations of phase retrieval algorithms for propagation-based X-ray microtomographic imaging », in *2013 8th International Symposium on Image and Signal Processing and Analysis (ISPA)*, IEEE, 2013, p. 622-626. DOI : [10.1109/ispa.2013.6703814](https://doi.org/10.1109/ispa.2013.6703814).
- [51] I. HÄGGMARK, W. VÅGBERG, H. M. HERTZ et A. BURVALL, « Comparison of quantitative multi-material phase-retrieval algorithms in propagation-based phase-contrast X-ray tomography », *Optics Express*, t. 25, n° 26, p. 33 543, 2017. DOI : [10.1364/OE.25.033543](https://doi.org/10.1364/OE.25.033543). adresse : <https://opg.optica.org/oe/abstract.cfm?URI=oe-25-26-33543>.
- [52] F. BRUN, L. BROMBAL, V. DI TRAPANI et al., « Post-reconstruction 3D single-distance phase retrieval for multi-stage phase-contrast tomography with photon-counting detectors », *Journal of Synchrotron Radiation*, t. 26, p. 510-516, 2019. DOI : [10.1107/s1600577519000237](https://doi.org/10.1107/s1600577519000237). adresse : <https://journals.iucr.org/s/issues/2019/02/00/mo5194/mo5194.pdf>.
- [53] L. HEHN, K. MORGAN, P. BIDOLA et al., « Nonlinear statistical iterative reconstruction for propagation-based phase-contrast tomography », *APL Bioengineering*, t. 2, 2018. DOI : [10.1063/1.4990387](https://doi.org/10.1063/1.4990387). adresse : <https://aip.scitation.org/doi/pdf/10.1063/1.4990387>.
- [54] T. WEITKAMP, D. HAAS, D. WEGRZYNEK et A. RACK, « *ANKAphase* : software for single-distance phase retrieval from inline X-ray phase-contrast radiographs », *Journal of Synchrotron Radiation*, t. 18, p. 617-629, 2011. DOI : [10.1107/s0909049511002895](https://doi.org/10.1107/s0909049511002895).
- [55] O. de LA ROCHEFOUCAULD, G. BEGANI PROVINCIALI, A. CEDOLA et al., « Element differentiation with a Hartmann- based X-ray phase imaging system », *Nondestructive Testing and Evaluation*, t. 37, p. 707-720, 2022. DOI : [10.1080/10589759.2022.2095383](https://doi.org/10.1080/10589759.2022.2095383).
- [56] S.-W. BAHK, « Highly accurate wavefront reconstruction algorithms over broad spatial-frequency bandwidth », *Optics Express*, t. 19, n° 20, p. 18 997, 2011. DOI : [10.1364/OE.19.018997](https://doi.org/10.1364/OE.19.018997). adresse : <http://opg.optica.org/oe/abstract.cfm?URI=oe-19-20-18997>.
- [57] G. LI, Y. LI, K. LIU, X. MA et H. WANG, « Improving wavefront reconstruction accuracy by using integration equations with higher-order truncation errors in the Southwell geometry », *Journal of the Optical Society of America A*, t. 30, p. 1448, 2013. DOI : [10.1364/josaa.30.001448](https://doi.org/10.1364/josaa.30.001448).
- [58] J. WOLFSON-POU et E. CHOW, « Distributed southwell », in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, B. MOHR et P. RAGHAVAN, éd., sér. SC '17, Denver, Colorado : ACM, 2017, p. 1-13. DOI : [10.1145/3126908.3126966](https://doi.org/10.1145/3126908.3126966). adresse : <https://doi.org/10.1145/3126908.3126966>.
- [59] O. de LA ROCHEFOUCAULD, B. BARBREL, M. PIPONNIER et al., « XPulse : Towards a Laser-driven X-ray imaging system », in *OSA High-brightness Sources and Light-driven Interactions Congress 2020 (EUVXRAY, HILAS, MICS)*, Optica Publishing Group, 2020, ETu1A.3. DOI : [10.1364/euvxray.2020.etu1a.3](https://doi.org/10.1364/euvxray.2020.etu1a.3). adresse : <http://www.osapublishing.org/abstract.cfm?URI=EUVXRAY-2020-ETu1A.3>.

- [60] S. TAVAKOLI TABA, B. D. ARHATARI, Y. I. NESTERETS et al., « Propagation-Based Phase-Contrast CT of the Breast Demonstrates Higher Quality Than Conventional Absorption-Based CT Even at Lower Radiation Dose », *Academic Radiology*, t. 28, e20-e26, 2021. DOI : [10.1016/j.acra.2020.01.009](https://doi.org/10.1016/j.acra.2020.01.009).
- [61] G. BEGANI PROVINCIALI, M. PIPONNIER, L. OUDJEDI et al., « High-Sensitivity X-ray Phase Imaging System Based on a Hartmann Wavefront Sensor », *Condensed Matter*, t. 7, p. 3, 2021. DOI : [10.3390/condmat7010003](https://doi.org/10.3390/condmat7010003). adresse : <https://www.mdpi.com/2410-3896/7/1/3/pdf?version=1640588233>.
- [62] K. MUELLER, F. XU et N. NEOPHYTOU, « Why do commodity graphics hardware boards (GPUs) work so well for acceleration of computed tomography? », in *SPIE Proceedings*, C. A. BOUMAN, E. L. MILLER et I. POLLAK, éd., International Society for Optics et Photonics, t. 6498, SPIE, 2007, 64980N. DOI : [10.1117/12.716797](https://doi.org/10.1117/12.716797). adresse : <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/6498/64980N/Why-do-commodity-graphics-hardware-boards-GPUs-work-so-well/10.1117/12.716797.pdf>.
- [63] B. JANG, D. KAELI, S. DO et H. PIEN, « Multi GPU implementation of iterative tomographic reconstruction algorithms », in *2009 IEEE International Symposium on Biomedical Imaging : From Nano to Macro*, IEEE, 2009, p. 185-188. DOI : [10.1109/isbi.2009.5193014](https://doi.org/10.1109/isbi.2009.5193014). adresse : <http://www.ece.neu.edu/groups/nucar/publications/bjang-isbi09.pdf>.
- [64] T. BICER, D. GÜRSOY, V. D. ANDRADE et al., « Trace : a high-throughput tomographic reconstruction engine for large-scale datasets », *Advanced Structural and Chemical Imaging*, t. 3, 2017. DOI : [10.1186/s40679-017-0040-7](https://doi.org/10.1186/s40679-017-0040-7). adresse : <https://ascimaging.springeropen.com/track/pdf/10.1186/s40679-017-0040-7>.
- [65] S. RIT, M. VILA OLIVA, S. BROUSMICHE, R. LABARBE, D. SARRUT et G. C. SHARP, « The Reconstruction Toolkit (RTK), an open-source cone-beam CT reconstruction toolkit based on the Insight Toolkit (ITK) », t. 489, IOP Publishing, 2014, p. 012079. DOI : [10.1088/1742-6596/489/1/012079](https://doi.org/10.1088/1742-6596/489/1/012079). adresse : <http://iopscience.iop.org/article/10.1088/1742-6596/489/1/012079/pdf>.
- [66] W. van AARLE, W. J. PALENSTIJN, J. DE BEENHOUWER et al., « The ASTRA Toolbox : A platform for advanced algorithm development in electron tomography », *Ultramicroscopy*, t. 157, p. 35-47, 2015. DOI : [10.1016/j.ultramic.2015.05.002](https://doi.org/10.1016/j.ultramic.2015.05.002). adresse : <https://ir.cwi.nl/pub/23858/23858D.pdf>.
- [67] W. J. PALENSTIJN, J. BÉDORF, J. SIJBERS et K. J. BATENBURG, « A distributed ASTRA toolbox », *Advanced Structural and Chemical Imaging*, t. 2, 1 2016. DOI : [10.1186/s40679-016-0032-z](https://doi.org/10.1186/s40679-016-0032-z). adresse : <https://ascimaging.springeropen.com/track/pdf/10.1186/s40679-016-0032-z>.
- [68] A. BIGURI, M. DOSANJH, S. HANCOCK et M. SOLEIMANI, « TIGRE : a MATLAB-GPU toolbox for CBCT image reconstruction », *Biomedical Physics and Engineering Express*, t. 2, n° 5, p. 055010, 2016. DOI : [10.1088/2057-1976/2/5/055010](https://doi.org/10.1088/2057-1976/2/5/055010). adresse : <https://iopscience.iop.org/article/10.1088/2057-1976/2/5/055010/pdf>.
- [69] A. BIGURI, R. LINDROOS, R. BRYLL et al., « Arbitrarily large tomography with iterative algorithms on multiple GPUs using the TIGRE toolbox », *Journal of Parallel and Distributed Computing*, t. 146, p. 52-63, 2020. DOI : [10.1016/j.jpdc.2020.07.004](https://doi.org/10.1016/j.jpdc.2020.07.004). adresse : <https://eprints.soton.ac.uk/434722/1/main.pdf>.

- [70] X. TANG, E. A. KRUPINSKI, H. XIE et A. E. STILLMAN, « On the data acquisition, image reconstruction, cone beam artifacts, and their suppression in axial MDCT and CBCT - A review », *Medical Physics*, t. 45, e761-e782, 9 2018. DOI : [10.1002/mp.13095](https://doi.org/10.1002/mp.13095).
- [71] J. SIJBERS et A. POSTNOV, « Reduction of ring artefacts in high resolution micro-CT reconstructions », *Physics in Medicine and Biology*, t. 49, N247-N253, 2004. DOI : [10.1088/0031-9155/49/14/n06](https://doi.org/10.1088/0031-9155/49/14/n06).
- [72] E. M. ABU ANAS, J. G. KIM, S. Y. LEE et M. K. HASAN, « Comparison of ring artifact removal methods using flat panel detector based CT images », *BioMedical Engineering OnLine*, t. 10, p. 72-72, 2011. DOI : [10.1186/1475-925x-10-72](https://doi.org/10.1186/1475-925x-10-72). adresse : <https://biomedical-engineering-online.biomedcentral.com/counter/pdf/10.1186/1475-925X-10-72>.
- [73] D. PRELL, Y. KYRIAKOU et W. A. KALENDER, « Comparison of ring artifact correction methods for flat-detector CT », *Physics in Medicine and Biology*, t. 54, p. 3881-3895, 2009. DOI : [10.1088/0031-9155/54/12/018](https://doi.org/10.1088/0031-9155/54/12/018).
- [74] F. BRUN, G. KOUROUSIAS, D. DREOSSI, L. MANCINI et G. TROMBA, « A comparative evaluation of ring artifacts reduction filters for X-ray computed microtomography images », in *2011 18th IEEE International Conference on Image Processing*, B. MACQ et P. SCHELKENS, éd., IEEE, 2011, p. 405-408. DOI : [10.1109/icip.2011.6116535](https://doi.org/10.1109/icip.2011.6116535).
- [75] C. GEORGIY et A. VLADISLAV, « Calculation of the rotation axis projection in X-ray uCT », 2016, Stage de master ? Compare des algos de calcul d'axe de rotation. adresse : http://students.jinr.ru/en/jinr2016-participants?task=application.downloadreportandreport_id=47.
- [76] S. J. ALLOO, D. M. PAGANIN, K. S. MORGAN et al., « Tomographic phase and attenuation extraction for a sample composed of unknown materials using x-ray propagation-based phase-contrast imaging », *Optics Letters*, t. 47, n° 8, p. 1945, 2022. DOI : [10.1364/ol.445802](https://doi.org/10.1364/ol.445802). adresse : <http://arxiv.org/pdf/2110.06284>.
- [77] C. CANOVAS et E. N. RIBAK, « Comparison of Hartmann analysis methods », *Applied Optics*, t. 46, n° 10, p. 1830, 2007. DOI : [10.1364/ao.46.001830](https://doi.org/10.1364/ao.46.001830). adresse : <http://opg.optica.org/ao/abstract.cfm?URI=ao-46-10-1830>.
- [78] D. BAILEY, « Sub-pixel estimation of local extrema », *Proceedings of Image and Vision Computing New Zealand*, p. 414-419, 2003.
- [79] J. NICKOLLS, I. BUCK, M. GARLAND et K. SKADRON, « Scalable Parallel Programming with CUDA », *Queue*, t. 6, p. 40-53, 2008. DOI : [10.1145/1365490.1365500](https://doi.org/10.1145/1365490.1365500).
- [80] J. E. BRESENHAM, « Algorithm for computer control of a digital plotter », *IBM Systems Journal*, t. 4, p. 25-30, 1965. DOI : [10.1147/sj.41.0025](https://doi.org/10.1147/sj.41.0025).
- [81] Y. LIU et J. C. YE, « Comparative study of projection/back-projection schemes in cryo-EM tomography », in *Image Reconstruction from Incomplete Data IV*, P. J. BONES, M. A. FIDDY et R. P. MILLANE, éd., sér. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, t. 6316, SPIE, 2006, 631606, p. 631 606. DOI : [10.1117/12.682287](https://doi.org/10.1117/12.682287).
- [82] R. LIU, L. FU, B. DE MAN et H. YU, « GPU-Based Branchless Distance-Driven Projection and Backprojection », *IEEE Transactions on Computational Imaging*, t. 3, p. 617-632, 4 2017. DOI : [10.1109/tci.2017.2675705](https://doi.org/10.1109/tci.2017.2675705).

- [83] T. MÖLLER et B. TRUMBORE, « Fast, minimum storage ray/triangle intersection », in *ACM SIGGRAPH 2005 Courses on - SIGGRAPH '05*, J. FUJII, éd., ACM Press, 2005, p. 7. DOI : [10.1145/1198555.1198746](https://doi.org/10.1145/1198555.1198746).
- [84] D. BALDWIN, « Fast Ray-Triangle Intersections by Coordinate Transformation », *Journal of Computer Graphics Techniques (JCGT)*, t. 5, 3 2016, ISSN : 2331-7418.
- [85] D. SHREINER, G. SELLERS, J. M. KESSENICH et al., *OpenGL Programming Guide : The Official Guide to Learning OpenGL, Version 4.3*, 8th. Addison-Wesley Professional, 2013, ISBN : 0321773039.
- [86] E. LENGYEL, « Oblique View Frustum Depth Projection and Clipping », *Journal of game development*, t. 1, p. 1-16, 2 2005, ISSN : 1543-9399. adresse : <http://www.terathon.com/code/oblique.html>.
- [87] R. KOOIMA, « Generalized Perspective Projection », *J. Sch. Electron. Eng. Comput. Sci.*, 2009.
- [88] M. RANDIMA PHARR et FERNANDO, *GPU Gems 2 : GPU Program Optimization*. Addison-Wesley Professional, 2005. DOI : [10.5555/1062395](https://doi.org/10.5555/1062395).
- [89] P. YALAMANCHILI, U. ARSHAD, Z. MOHAMMED et al., *ArrayFire - A high performance software library for parallel computing with an easy-to-use API*, Atlanta, 2015. adresse : <https://github.com/arrayfire/arrayfire>.
- [90] M. HIDAYETOĞLU, T. BIÇER, S. G. de GONZALO et al., « MemXCT », in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, M. TAUFER, P. BALAJI et A. J. PEÑA, éd., sér. SC '19, Denver, Colorado : ACM, 2019, 85 :1-85 :56. DOI : [10.1145/3295500.3356220](https://doi.org/10.1145/3295500.3356220). adresse : <https://dl.acm.org/doi/pdf/10.1145/3295500.3356220>.
- [91] J. BUURLAGE, R. BISSELING et K. BATENBURG, « A geometric partitioning method for distributed tomographic reconstruction », *Parallel Computing*, t. 81, p. 104-121, 2019. DOI : [10.1016/j.parco.2018.12.007](https://doi.org/10.1016/j.parco.2018.12.007). adresse : https://dspace.library.uu.nl/bitstream/1874/377882/1/1_s2.0_S0167819118301856_main.pdf.
- [92] A. SKODRAS, C. CHRISTOPOULOS et T. EBRAHIMI, « The JPEG 2000 still image compression standard », *IEEE Signal Processing Magazine*, t. 18, n° 5, p. 36-58, 2001. DOI : [10.1109/79.952804](https://doi.org/10.1109/79.952804). adresse : <https://infoscience.epfl.ch/record/111813/files/223%20The%20JPEG%202000%20still%20image%20compression%20standard.pdf>.
- [93] C. STAMM, « PGF - A New Progressive File Format for Lossy and Lossless Image Compression », in *WSCG*, 2002, p. 421-428.
- [94] X. MEI, K. ZHAO, C. LIU et X. CHU, « Benchmarking the Memory Hierarchy of Modern GPUs », in *Advanced Information Systems Engineering*, C.-H. HSU, X. SHI et V. SALAPURA, éd., t. 8707, Springer Berlin Heidelberg, 2014, p. 144-156. DOI : [10.1007/978-3-662-44917-2_13](https://doi.org/10.1007/978-3-662-44917-2_13). adresse : https://link.springer.com/content/pdf/10.1007%2F978-3-662-44917-2_13.pdf.
- [95] S. RYOO, C. I. RODRIGUES, S. S. BAGHSORKHI, S. S. STONE, D. B. KIRK et W.-m. W. HWU, « Optimization principles and application performance evaluation of a multithreaded GPU using CUDA », *ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, S. CHATTERJEE et M. L. SCOTT, éd., p. 73-82, June 2014 2008. DOI : [10.1145/1345206.1345220](https://doi.org/10.1145/1345206.1345220).

- [96] G. J. W. V. D. BRAAK, *Improving GPU performance : reducing memory conflicts and latency*. 2015. DOI : [10.13140/RG.2.1.4147.0803](https://doi.org/10.13140/RG.2.1.4147.0803).
- [97] B. D. MAN et S. BASU, « Distance-driven projection and backprojection in three dimensions », *Physics in Medicine and Biology*, t. 49, p. 2463-2475, 11 2004. DOI : [10.1088/0031-9155/49/11/024](https://doi.org/10.1088/0031-9155/49/11/024).
- [98] R. L. HARRISON, C. GRANJA et C. LEROY, « Introduction to Monte Carlo Simulation », *AIP Conference Proceedings*, t. 1204, p. 17-21, 2010. DOI : [10.1063/1.3295638](https://doi.org/10.1063/1.3295638). adresse : <https://europepmc.org/articles/pmc2924739?pdf=render>.
- [99] N. METROPOLIS et S. ULAM, « The Monte Carlo Method », *Journal of the American Statistical Association*, t. 44, p. 335-341, 1949. DOI : [10.1080/01621459.1949.10483310](https://doi.org/10.1080/01621459.1949.10483310).