Imagerie Médicale

Etudiant parcours M2 – GICAO :

Henry Lefèvre

Aymeric Seguret

Enseignant : Laurent Desbat

15/02/2016

A terme, le but du projet sera (si possible) de proposer une méthode de reconstruction d'une ROI (region of interest) à partir de données fanbeam incomplètes.

Partie 1 : Rétroprojection filtrée en géométrie parallèle

Dans la première partie du projet, on se place dans le cadre de la géométrie parallèle. Le but est de réussir à écrire la rétroprojection filtrée comme la rétroprojection des dérivations des projections filtrées avec Hilbert. Au final on montrera que cette méthode équivaut à filtrer les données avec un filtre ramp.

Cadre du travail:

Nous travaillons sur des données en 2 dimensions qui ont été acquises en géométrie parallèle. Le fichier '*data127x80*' nous fournit le sinogramme pour cette partie du projet. La source se déplace selon une trajectoire circulaire autour du système étudié sur une base de 127 échantillonnages entre [0 π]. Sur chaque projection, on dispose de 80 détecteurs positionnés de manière rectiligne et enregistrant le rayonnement reçu depuis la source. Chaque mesure est identifiée par un angle de projection φ , et une translation notée s.

Pour rester dans un cadre général, nous noterons :

nprojections : le nombre de projections acquises

ndetecteurs : le nombre de détecteurs présents sur chaque projection

L'image du sinogramme en entrée est donnée figure 1

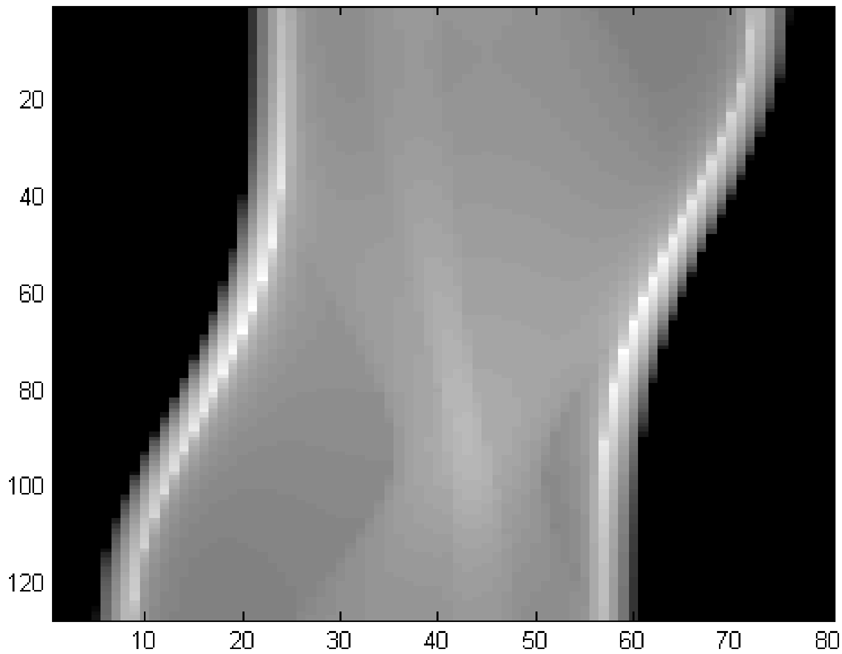

Figure 1: sinogramme lu

1. Application du filtre de Hilbert

La rétroprojection filtrée s'effectue en 2 étapes, la première consiste à filtrer les données avec un filtre, ici celui de Hilbert. Cela veut dire que chaque projection du sinogramme sera filtrée. Pour filtrer les données, on va utiliser la propriété des convolutions de fonctions dans l'espace de Fourier pour réaliser cette opération. Cette propriété dit explicitement :

$$F(f * g) = F(f) \times F(g)$$

Ainsi on se place dans l'espace de Fourier pour réaliser le filtrage des données. Dans Matlab, le filtre de Hilbert nous est donné par la fonction '*hilbertfilter*'. Ce filtre a une taille de 2*ndetecteurs - 1, nous avons utilisé la technique du zero-padding afin d'effectuer le produit de convolution.

Représentation schématique du zero-pading :

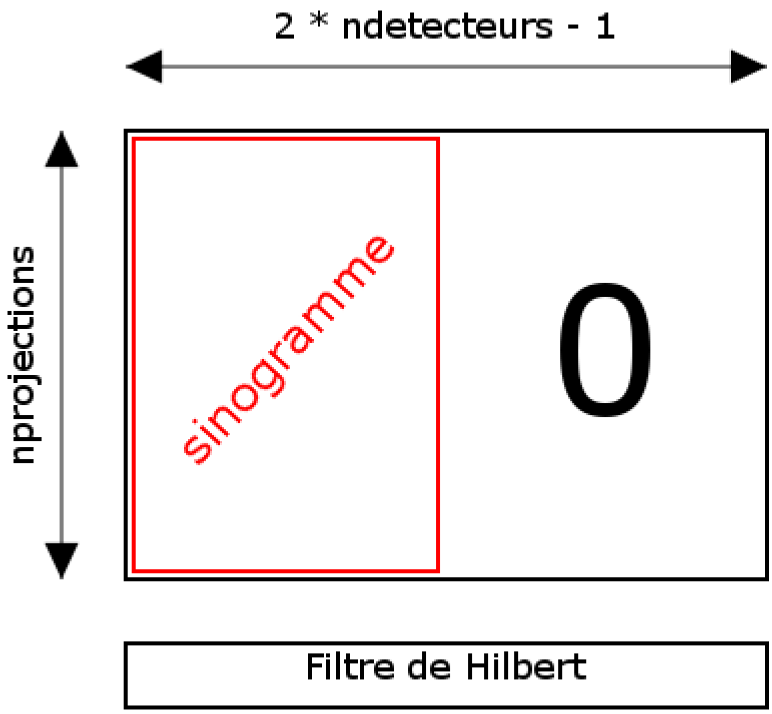

Figure 2 : zero-pading

Les projections ainsi que le filtre sont transformés dans l'espace de Fourier afin de réaliser leur produit. Le sinogramme filtré est ensuite tronqué sur les détecteurs sur [1 ndetecteurs].

2. Dérivation des projections

La seconde étape consiste à effectuer une dérivation selon le paramètre s. Le schéma de dérivation classique est appliqué sur chaque ligne du sinogramme.

$$\frac{\partial \rho(\varphi,s)}{\partial s} = \frac{\rho(\varphi,s+h) - \rho(\varphi,s-h)}{2h}$$

3. Algorithme de rétroprojection

La rétroprojection s'effectue donc sur des données dont les projections ont subi un filtrage de Hilbert suivi d'une dérivation selon le paramètre associé aux détecteurs.

Remarque : l'algorithme de rétroprojection utilisé est le même que celui du TP 1.

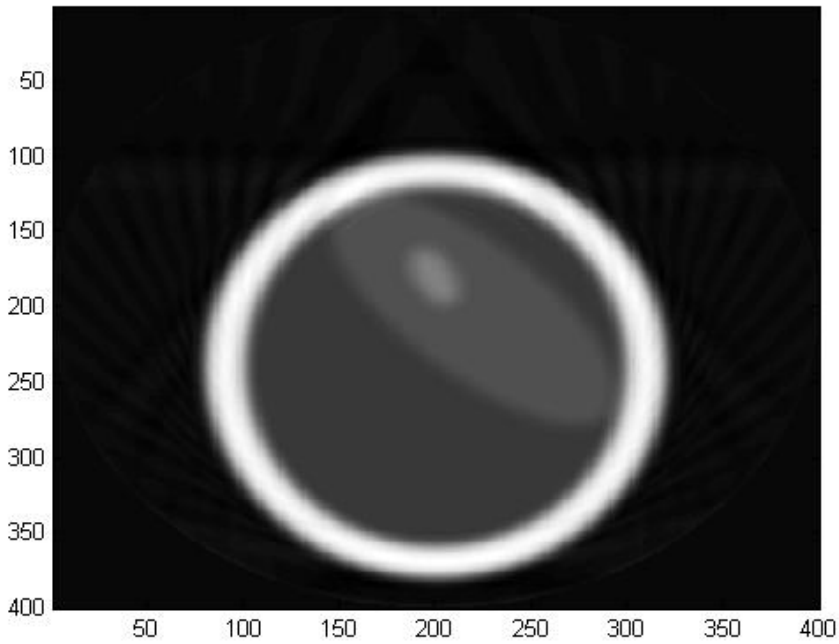

Figure 3 : Algorithme de rétroprojection parallèle avec les données du TP 1

Partie 2 : Reconstruction de ROI dans un scanner en géométrie Fanbeam

Dans la seconde partie du projet, nous tenterons d'analyser le problème et de donner une solution d'après ce que l'on a compris. Cette fois-ci, nous sommes avec une géométrie fanbeam. C'est à dire que les détecteurs sont identifiés grâce à un angle t (paramétrant la trajectoire de la source, t appartient à [0 2π]) et un angle alpha représenté par la droite '*source-détecteur*' avec la droite '*source-origine*'. L'angle alpha lui est compris entre [-dfa dfa] avec dfa, le demi grand angle d'ouverture.

$$dfa = \arcsin(\frac{1}{R})$$

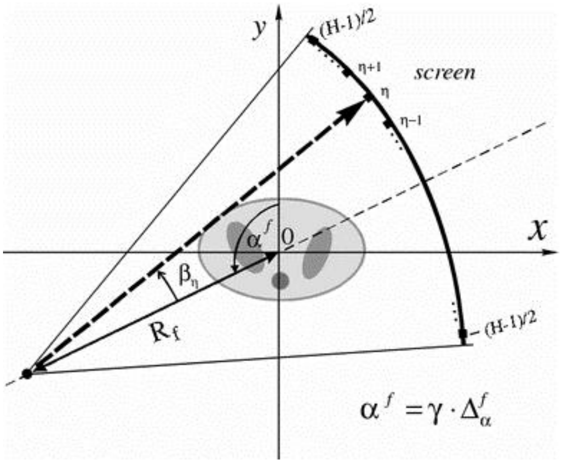

Figure 4 : géométrie fan beam

L'enjeu de cette partie est d'arriver à reconstruire une géométrie parallèle à partir de données fanbeam complètes. Pour se faire, la première étape est de filtrer les données fanbeam avec le filtre de Hilbert (même méthode que la partie 1). Cette fois-ci ce filtre nous est donné par la fonction '*fbhilbertfilter*'.

Il faut s'appuyer sur le théorème, qui dit qu'une droite de projection de Hilbert en fanbeam correspond à une droite de projection de Hilbert en géométrie parallèle. Géométriquement, cela se vérifie (voir figure 5).

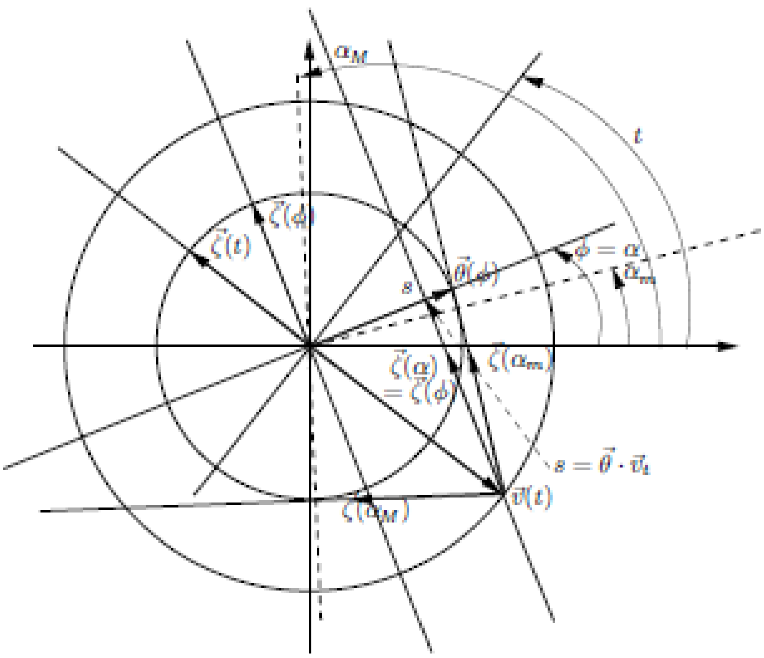

Figure 5 : changement de paramètre

La droite paramétrée par la source v(t) ayant pour vecteur directeur ξ(α) peux être exprimée en géométrie parallèle. En effet, le vecteur ξ(α) est orthogonale à $\Theta(\varphi)$ ce qui correspond à une droite de géométrie parallèle.

$$p(\varphi, s) = g(t, \alpha) \quad \text{avec} \quad s = v(t).\Theta(\varphi) \text{ et } \varphi = \alpha$$

Avec ce changement de paramètre, on serait donc capable de définir un sinogramme parallèle filtré avec Hilbert à partir d'une géométrie fanbeam. A partir de là il nous est alors possible d'utiliser ce qui a été fait dans la première partie (i.e. l'algorithme de rétroprojection) afin de recréer l'image finale. Ce procédé ne marche que si les données sont complètes. Pour le cas de données incomplètes, nous n'avons pas de solution pour reconstruire la géométrie.