

UNIVERSITÉ DE PARIS-SUD

## Rapport de stage

effectué au  
Groupe des Problèmes Inverses  
Laboratoire des Signaux et Systèmes

pour l'obtention du  
Diplôme des Etudes Approfondies

Intitulé

### **UNE APPROCHE MULTIRÉSOLUTION POUR LA RECONSTRUCTION 3D DE LA SURFACE D'UN OBJET COMPACT EN TOMOGRAPHIE**

présenté par  
Wala TOUHAMI

encadré par  
Ali MOHAMMAD-DJAFARI

Promotion 2001/2002

# Remerciements

Je tiens à remercier Monsieur Ali MOHAMMAD-DJAFARI pour l'intérêt qu'il a manifesté tout au long de la réalisation de ce travail et pour l'aide précieuse qu'il n'a cessé de prodiguer. Je le remercie sincèrement pour la pertinence de ses remarques et de ses conseils.

J'exprime toute ma gratitude à l'équipe du Groupe Problèmes Inverses pour l'accueil chaleureux qu'elle m'a réservé et plus particulièrement Monsieur Charles SOUSSEN pour son aide et ses conseils.

Enfin, je dois mes remerciements à Madame Odette LEROUX pour son aide et son soutien moral, ainsi que mes amis pour leur cordialité et leur sympathie.

# Table des matières

<b>Table de figures</b>	<b>4</b>
<b>Listes des tableaux</b>	<b>5</b>
<b>Introduction</b>	<b>8</b>
<b>1 Présentation du problème</b>	<b>9</b>
1.1 Problème de la tomographie . . . . .	9
1.2 Méthodes classiques de reconstruction . . . . .	10
1.2.1 Méthodes analytiques . . . . .	10
1.2.2 Les méthodes algébriques . . . . .	10
1.3 Contexte du problème . . . . .	11
1.4 L’approche bayésienne . . . . .	14
1.5 Sujet précis de mon stage . . . . .	15
1.6 Conclusion . . . . .	15
<b>2 Approche polyédrique en monorésolution</b>	<b>16</b>
2.1 Introduction . . . . .	16
2.2 Présentation du modèle polygonale et polyédrique . . . . .	16
2.3 Méthodes d’estimation de paramètres inconnus . . . . .	17
2.3.1 Méthodes des moments . . . . .	17
2.3.2 Méthode d’estimation directe . . . . .	18
2.4 Techniques d’optimisation . . . . .	19
2.5 Différentes étapes de calcul du critère et de son gradient . . . . .	20
2.5.1 Calcul des projections . . . . .	20
2.5.2 Mise à jour des projections . . . . .	21
2.6 Calcul du gradient du critère . . . . .	22
2.7 Conclusion . . . . .	23
<b>3 Approche multirésolution</b>	<b>24</b>
3.1 Introduction . . . . .	24
3.2 La multirésolution en 2D . . . . .	24
3.3 Approche multirésolution en 3D . . . . .	26
3.3.1 Différentes méthodes de triangulation . . . . .	27
3.3.2 Méthode proposée . . . . .	28
3.4 Conclusion . . . . .	35

---

<b>4 Simulations</b>	<b>36</b>
4.1 Introduction . . . . .	36
4.2 Nature des données . . . . .	36
4.3 Mesure de la qualité de reconstruction . . . . .	37
4.4 Approche monorésolution en 3D . . . . .	38
4.5 Approche multirésolution en 3D . . . . .	39
4.6 Conclusion . . . . .	40
<b>Conclusion</b>	<b>43</b>
<b>Bibliographie</b>	<b>45</b>

# Table des figures

1.1	Mode de projection orthogonale en 3D . . . . .	9
1.2	paramétrisation de l'ellipsoïde . . . . .	13
2.1	Intersection rayon-facettes . . . . .	21
2.2	Extraction du mini polyèdre . . . . .	22
2.3	Extraction du mini polyèdre contenant $V_1$ . . . . .	22
3.1	Insertion des sommets aux milieux des arêtes . . . . .	25
3.2	Reconstruction en monorésolution . . . . .	25
3.3	Solution initiale et les reconstruction en multirésolution pour 4 niveaux de résolutions	26
3.4	Principe de la multirésolution . . . . .	27
3.5	Triangulation de Delaunay . . . . .	28
3.6	Insertion de sommets : stratégie1 . . . . .	30
3.7	Résultats d'insertion de sommets : stratégie1 . . . . .	31
3.8	Insertion de sommets : stratégie2 . . . . .	32
3.9	Résultats d'insertion de sommets : stratégie2 . . . . .	33
3.10	Problème d'auto-intersections . . . . .	34
3.11	Fusion des sommets . . . . .	34
3.12	Inversion et création de sommets . . . . .	35
4.1	Données synthétiques 3D. . . . .	37
4.2	Initialisation par une ellipsoïde. . . . .	38
4.3	Reconstruction en monorésolution. . . . .	39
4.4	Initialisation de la reconstruction . . . . .	40
4.5	Résultats de l'approche multirésolution . . . . .	41
4.6	Résultats de l'approche multirésolution . . . . .	42
4.7	Distance entre les objets . . . . .	42
4.8	Distance entre les projections . . . . .	42

# Liste des tableaux

3.1	Méthode de détermination de lieu d'ajout des sommets . . . . .	29
3.2	Sauvegarde de la liste des sommets et des triangles . . . . .	30
3.3	Algorithme d'ajout des sommet : stratégie1 . . . . .	31
3.4	Algorithme d'ajout des sommets : stratégie2 . . . . .	32
3.5	Algorithme de la simplification de la triangulation . . . . .	33
4.1	Données 3D. Description des vecteurs directeurs des projections. . . . .	36

# Introduction

Quand une grandeur physique d'intérêt n'est pas directement accessible à la mesure, il est courant de procéder à l'observation d'autres quantités qui lui sont reliées par des lois physiques. la notion de problème inverse correspond à l'idée d'inverser ces lois physiques pour accéder indirectement à la quantité d'intérêt.

Les méthodes de résolution des problèmes inverses ont suscité un intérêt croissant depuis leur genèse. la problématique d'inversion concerne des secteurs variés tels que les sciences de la terre, l'imagerie médicale, l'industrie électromagnétique...

En particulier, nous nous intéressons dans cette étude à un domaine des problèmes inverses : la reconstruction tomographique. La tomographie a pour objectif de reconstruire d'une façon non destructive une cartographie d'un paramètre caractéristique d'un objet. Son principe repose sur l'analyse de l'interaction entre un rayonnement que l'on propage dans l'objet. Dans certains nombre de configurations, on ne dispose que d'un nombre très faible de projections.

La reconstruction tomographique d'un objet, à partir d'un nombre faible de projections, conduit à un problème inverse mal posé : les conditions d'existence, d'unicité et de stabilité ne sont pas vérifiées. Une reconstruction de type inverse généralisée, à partir de projections bruitées, conduit à une solution très dégradée. Pour remédier à cela, il est nécessaire de régulariser par introduction d'informations *a priori*.

Différentes méthodes existent pour la résolution de ce problème de reconstruction, à savoir la discétisation en voxels ou la modélisation par une forme déformable. Nous optons pour cette étude à la deuxième méthode qui présente l'avantage de réduire le nombre de paramètres à estimer et l'inconvénient de rendre le critère à optimiser non linéaire.

Nous modélisons cet objet par une forme polyédrique  $\mathcal{P}$ , par conséquent, le problème de reconstruction revient à estimer les sommets de  $\mathcal{P}$ . Cette estimation est faite par le biais d'une approche bayésienne introduisant des informations *a priori* sur  $\mathcal{P}$ .

La première idée s'était de résoudre ce problème en monorésolution, c'est à dire que le nombre de sommets de l'objet initial est fixe et ne change pas au cours du procédé. Cependant, nous avons remarqué que ce nombre doit être assez important, ce qui peut rendre le cout de calcul important. Pour cette raison, nous avons proposer l'approche multirésolution qui consiste à reconstruire successivement des polyèdres à nombre croissant de sommets et à considérer une reconstruction à faible nombre de sommets pour produire l'initialisation à la résolution immédiatement supérieure. Cette approche sera étudiée durant ce stage, nous allons essayer de décrire les modules nécessaires permettant le passage d'un niveau à un autre, ainsi que les limites qui pouvant apparaître.

Ce manuscrit est composé de quatre chapitres. Un premier chapitre introductif qui permet de positionner le problème, de décrire les méthodes de reconstruction et enfin de présenter le sujet précis du stage. Dans le deuxième chapitre, nous décrivons le modèle adapté qui est le modèle polyédrique. Nous présentons aussi une brève discription de l'approche monorésolution en 3D et les techniques d'optimisation nécessaires pour résoudre ce problème à savoir le calcul des projection

---

et leur mise à jour qui permettent le calcul du critère et de son gradient. Un troisième chapitre sera consacré à l'approche multirésolution. Nous commencerons par présenter le principe de cette approche dans le cas 2D. Ensuite nous proposons son extension dans le cas 3D. Les simulations seront présentées dans le dernier chapitre, nous présentons les résultats des modules décrits dans le troisième chapitre en les comparant avec les résultats de l'approche de monorésolution afin de montrer les performances apportées par cette approche

# Chapitre 1

## Présentation du problème

### 1.1 Problème de la tomographie

On illumine un objet 3D par un rayon X d'intensité d'entrée  $I_0$ , et on mesure son intensité  $I$  à la sortie. Si on définit l'objet par une fonction  $f(x, y, z)$  qui représente la distribution de sa constante d'atténuation  $\mu(x, y, z)$ , alors on peut écrire la relation suivante :

$$-\ln \frac{I}{I_0} = \int_{\mathcal{L}} f(x, y, z) dl$$

où  $\mathcal{L}$  est une ligne droite reliant la source au détecteur

En notant  $p = -\ln \frac{I}{I_0}$ , et en paramétrisant  $\mathcal{L}$  par sa direction  $\boldsymbol{\theta} = (\theta, \phi)$  et par la position du détecteur  $\mathbf{r}$  sur un plan  $\omega_{\boldsymbol{\theta}}^{\perp}$  perpendiculaire à  $\mathcal{L}$  ( $\mathbf{r} \in \omega_{\boldsymbol{\theta}}^{\perp} \Rightarrow \mathbf{r} = u\mathbf{u}_{\boldsymbol{\theta}} + v\mathbf{v}_{\boldsymbol{\theta}}$ ), on obtient ce qu'on appelle une projection  $p_{\boldsymbol{\theta}}(\mathbf{r})$

$$p_{\boldsymbol{\theta}}(\mathbf{r}) = \int_{\mathcal{L}} f(x, y, z) dl \quad (1.1)$$

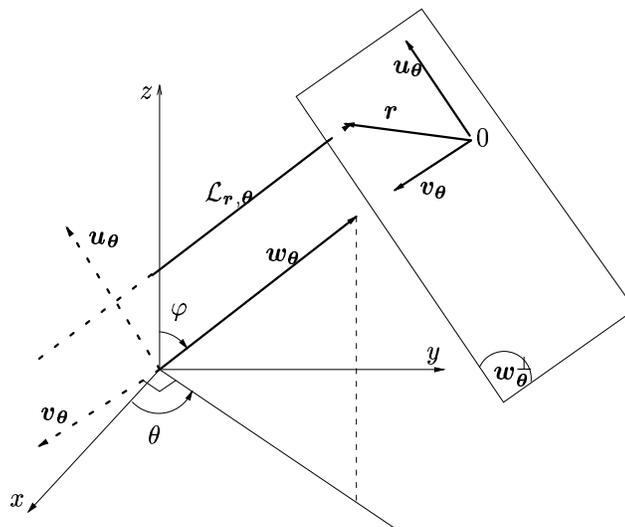


FIG. 1.1 – Mode de projection orthogonale en 3D

Une projection suivant la direction  $\boldsymbol{\omega}_\theta$  passant par  $\mathbf{r}$  s'écrit :

$$p_\theta(\mathbf{r}) = \int_{\mathbb{R}^3} f(x, y, z) \delta_2(\mathbf{r}_\theta(x, y, z) - \mathbf{r}) dx dy dz \quad (1.2)$$

L'objectif de cette étude est de partir de ces projections définies dans le cas 3D pour arriver à estimer la fonction de densité  $f(x, y, z)$  par  $\hat{f}(x, y, z)$ . Dans ce projet, on se place dans le cadre où on ne dispose que d'un nombre faible de projections donc de données.

## 1.2 Méthodes classiques de reconstruction

On distingue deux grandes approches pour la reconstruction d'un objet à partir de ces projections par rayon X

- Les méthodes analytiques basées sur la transformée inverse de (1.1) qui est connu sous le nom de la Transformée de Radon (TR).
- Les méthodes algébriques qui discrétisent directement (1.1) et cherche une solution pour l'équation algébrique qui en découle.

### 1.2.1 Méthodes analytiques

Afin de faciliter la compréhension dans ce paragraphe, nous considérons le cas 2D, la (TR) s'écrit :

$$p_\theta(r) = \int_{\mathbb{R}^2} f(x, y) \delta(r - x \cos \theta - y \sin \theta) dx dy$$

et l'expression analytique de l'inverse est :

$$f(x, y) = \frac{1}{2\pi^2} \int_0^\pi \int_0^\infty \frac{\partial p_\theta(r) / \partial r}{r - x \cos \theta - y \sin \theta} dr d\theta \quad (1.3)$$

Mais, comme les données  $p_\theta(r)$  ne sont pas connues pour toute valeur de  $r$  et  $\theta$ , une approximation des intégrales est nécessaire. Pour obtenir une solution approchée satisfaisante, un grand nombre de méthodes ont été proposées, à savoir l'inversion directe de la TR, l'interpolation dans le domaine de Fourier, la rétroprojection...

Cependant, plusieurs difficultés heurtent l'utilisation de ces méthodes. Premièrement, le contexte du notre problème, c'est à dire le nombre réduit de données. Deuxièmement, la nécessité d'interpoler la solution obtenue en coordonnées cartésiennes. Cette étape d'interpolation nécessite des informations *a priori* qui ne sont pas définies dans ces méthodes.

### 1.2.2 Les méthodes algébriques

Ces méthodes consistent à projeter les grandeurs inconnues sur une base de fonctions appropriées  $f(x, y, z) = \sum_{i=1}^n f_i b_i(x, y, z)$  Un exemple typique où la scène est discrétisée par un ensemble de pixels ou voxels. L'équation (1.1) s'écrit alors :

$$p_\theta(\mathbf{r}) = \sum_{i=1}^n f_i a_i(\mathbf{r}, \theta) \quad (1.4)$$

où  $a_i$  représente la longueur de l'intersection du  $i^{me}$  voxel de la scène avec la droite de projection  $\mathcal{L}_{\mathbf{r},\boldsymbol{\theta}}$  et  $n$  est le nombre de paramètres inconnus.

En réécrivant  $p_{\boldsymbol{\theta}}(\mathbf{r}) \forall \mathbf{r}, \boldsymbol{\theta}$  sous forme matricielle, on obtient l'équation suivante :

$$\mathbf{p} = A \mathbf{f} \quad (1.5)$$

où  $\mathbf{f} = [f_1, \dots, f_n]^T$  et  $A$  est la matrice de projection de taille  $M \times n$  avec  $M$  est le nombre de données.

Le problème se ramène alors à la résolution de l'équation :

$$\mathbf{p} = A \mathbf{f} + \boldsymbol{\epsilon} \quad (1.6)$$

où  $\boldsymbol{\epsilon}$  représente à la fois les erreurs liées à l'approximation et au bruit de mesure.

Arrivée à ce stade, on peut penser qu'une solution approchée peut être obtenue en utilisant la méthode de moindres carrés :

$$\hat{\mathbf{f}}_{MC} = \arg \min_{\mathbf{f}} \|\mathbf{p} - A \mathbf{f}\|^2 \quad (1.7)$$

Ce qui revient à résoudre :

$$A^T \mathbf{p} = A^T A \mathbf{f}$$

où  $A^T A$  est une matrice de  $n \times n$ . Or, le nombre de projections est très limitée ( $M < n$ ) et la matrice  $A^T A$  n'est pas inversible. Par conséquent, le système possède une infinité de solutions. Pour y remédier, on peut penser à l'inverse généralisée  $\mathbf{f}^\dagger = A^\dagger \mathbf{p}$ , cette solution s'obtient en effectuant une décomposition en valeur singulière de  $A$ . Mais,  $A$  est une matrice de grandes dimension et souvent très mal conditionnée. Son conditionnement dépend du choix de la base, du rapport entre le nombre de données  $M$  et le nombre de paramètres décrivant l'objet[19]. Donc, il n'est pas possible d'obtenir une solution satisfaisante du problème au sens du moindres carrés ou même en inverse généralisée sans l'introduction d'une information *a priori* sur les grandeurs inconnues, donc d'introduire une structure régularisée.

## 1.3 Contexte du problème

L'étude dans ce projet porte sur la reconstruction des objets compacts dans un milieu homogène à partir d'un nombre faible de projections. On peut rencontrer ce problème dans plusieurs techniques industrielles en contrôle non destructif (CND) des matériaux où l'on cherche à déterminer la forme précise d'un défaut à l'intérieur d'un objet métallique.

Cette contrainte nous permet de modéliser la scène par la présence de deux types de matériaux de densités différentes. La fonction de densité d'un tel objet s'écrit :

$$f(x, y, z) = \begin{cases} f_1 & \text{si } (x, y, z) \text{ appartient au défaut } \mathcal{C} \\ f_0 & \text{ailleurs} \end{cases} \quad (1.8)$$

Sans perdre la généralité, nous prenons le cas où  $f_1 = 1$  et  $f_0 = 0$ . Donc, l'expression de la projection (1.1) devient :

$$p_{\boldsymbol{\theta}}(\mathbf{r}) = \int_{\mathcal{L}_{\mathbf{r},\boldsymbol{\theta}}} f(x, y, z) dl = \int_{\mathcal{L}_{\mathbf{r},\boldsymbol{\theta}} \cap \mathcal{C}} dl \quad (1.9)$$

On distingue deux classes de méthodes pour la reconstruction d'un objet dans des telles conditions.

- Méthodes basées sur la discrétisation en voxels du volume où se trouve l'objet ([22], [6]) : ces méthodes ont été largement utilisées en imagerie médicale, en contrôle non destructif pour la reconstruction des objets à partir des radiographies. L'avantage majeure de cette méthode est la linéarité de la relation entre les données et les paramètres à estimer. Néanmoins, elle présente aussi un inconvénient important qui est la taille de nombre de paramètres à estimer, ce nombre est égale au nombre de pixels ou voxels de l'objet, et donc une charge de calcul importante. Par ailleurs, cette approche ne prend pas en compte le caractère compact de l'objet.

La fonction  $f(x, y, z)$  devient alors :

$$f(x, y, z) = \sum_{j=1}^n f_j \delta_j(x, y, z)$$

$$\text{où } \delta(x, y, z) = \begin{cases} 1 & \text{si } (x, y, z) \text{ appartient au voxel } j \\ 0 & \text{sinon} \end{cases} \quad (1.10)$$

Le problème se ramène toujours à la résolution de l'équation

$$\mathbf{p} = A \mathbf{f} + \epsilon \quad (1.11)$$

Néanmoins, cette méthode présente aussi un inconvénient important qui est la taille de nombre de paramètres à estimer  $\mathbf{f}$ , ce nombre est égale au nombre de pixels ou voxels de l'objet, et donc une charge de calcul importante. Par ailleurs, dans cette approche, il est plus difficile de prendre en compte le caractère compact et binaire de l'objet.

- Méthodes basées sur la modélisation de la surface de l'objet compact : ce sont des méthodes qui permettent l'estimation directe de la forme géométrique de l'objet qui est modélisé par un nombre de paramètres réduit d'une façon remarquable. Le seul inconvénient de cette approche est la non linéarité de la relation entre les données et ces paramètres. En effet, l'équation (1.19) devient :

$$\mathbf{p} = A(\mathbf{f}) + \epsilon \quad (1.12)$$

où  $\mathbf{f}$  représente dans ce cas le vecteur des paramètres qui décrivent la surface de l'objet compact.

Pour modéliser cet objet, il existe plusieurs formes, on cite :

1. Ellipsoïde : c'est un modèle convexe, simple définie par son centre  $\mathbf{G}$  et les trois axes d'inertie  $\boldsymbol{\xi}(\xi_1, \xi_2, \xi_3)$ .

Les paramètres à estimer pour ce modèle sont :

$$\mathbf{x} = \{\mathbf{G}, a_1, a_2, a_3, \theta, \phi\}$$

L'équation de ce modèle est la suivante :

$$\mathbf{GM}(\boldsymbol{\theta}) = [a_1 \cos \theta \sin \phi, a_2 \sin \theta \sin \phi, a_3 \cos \phi]^T \quad (1.13)$$

où  $a_i$  :les demi longueurs de l'ellipsoïde

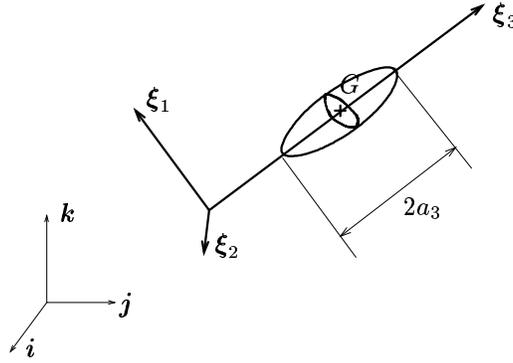


FIG. 1.2 – paramétrisation de l'ellipsoïde

2. Super-ellipsoïde : c'est un modèle qui généralise le modèle des ellipsoïdes. Il n'est pas nécessairement convexe, il dépend en plus des paramètres  $\alpha$  et  $\beta$  dont leur variation déforme la forme de l'ellipsoïde longitudinalement.

L'équation d'un tel modèle s'écrit :

$$\mathbf{GM}(\boldsymbol{\theta}) = [a_1 \cos^\beta \theta \sin^\alpha \phi, a_2 \sin^\beta \theta \sin^\alpha \phi, a_3 \cos^\alpha \phi]^T \quad (1.14)$$

3. Modèle harmonique : c'est un modèle non convexe étoilé en un point  $\mathbf{G}$ . Son équation s'écrit :

$$\mathbf{GM}(\boldsymbol{\theta}) = \rho(\boldsymbol{\theta}) \omega_{\boldsymbol{\theta}} \quad (1.15)$$

où  $\rho(\boldsymbol{\theta})$  est une fonction régulière, positive définie sur une boule unité de  $\mathbb{R}^3$  et elle se décompose en une combinaison linéaire d'harmonique sphérique [22] comme suit :

$$\rho(\boldsymbol{\theta}) = \sum_{l=0}^{\infty} \sum_{|k| \leq l} a_l^k \rho_l^k(\boldsymbol{\theta})$$

Les paramètres à estimer pour ce modèle sont  $\mathbf{x} = \{\mathbf{G}, a_l^k / |k| \leq l\}$

4. Splines : Ce sont des modèles locaux et plus adaptés que les autres pour modéliser la douceur locale d'un contour. En plus, chaque point de contrôle dans ce modèle n'influe que sur une portion du contour. Pour des splines cubiques, on considère une grille de  $(q+1)^2$  positions échantillonnées sur un intervalle  $[a, b]^2$ , toute surface spline cubique s'exprime par :

$$M(t, u) = \sum_{i=1}^q \sum_{j=1}^q P_{ij} B_i(t) B_j(u); \quad (t, u) \in \mathbb{R}^2$$

où  $P_{ij} \in \mathbb{R}^3$  : les points de contrôle

$B_i$  : fonctions de la base

Le vecteur de paramètres  $\mathbf{x}$  de ce modèle est formé par l'ensemble de coordonnées des  $n$  points de contrôle  $P_{ij}$  de la spline. Le problème le plus important pour ces modèles est le choix de l'ordre du spline. Dans notre cas, il s'avère plus judicieux de choisir les splines d'ordre 1, c'est à dire des polygones pour le cas 2D et des polyèdres pour ce cas 3D, car le calcul de leur projection se fait d'une façon directe.

## 1.4 L'approche bayésienne

On a vu dans le paragraphe (1.2.2) que la solution au sens du moindres carrés ou en inverse généralisée n'est pas satisfaisante pour la reconstruction avec un nombre limitée de données. Pour remédier à ce problème, on est amené à introduire des informations *a priori*. Ceci peut se faire par un choix judicieux des fonctions de la base de telle sorte que l'objet puisse être décrit par un faible nombre de paramètres [19]. Mais, cette approche est très spécifique car on sera obligé de construire une base pour chaque objet.

Une alternative à cette approche, une fois les fonctions de base définies d'une manière unique, est l'approche bayésienne qui permet de prendre en compte les informations *a priori* sur les paramètres inconnus  $\mathbf{f}$  et sur les données  $\mathbf{p}$  à travers des quantités probabilistes et de définir par la suite une distribution *a posteriori* sur les inconnus  $\mathbf{f}$ .

Rappelons que la relation entre des données et les inconnus est soit  $\mathbf{p} = A\mathbf{f} + \boldsymbol{\epsilon}$  dans le cas de discrétisation par voxel, soit  $\mathbf{p} = A(\mathbf{f}) + \boldsymbol{\epsilon}$  dans le cas de modélisation par des formes géométriques. On not d'une façon générale :

$$\mathbf{y} = A(\mathbf{x}) + \boldsymbol{\epsilon}$$

où  $\mathbf{y}$  : les données radiographiques.

$\mathbf{x}$  : les inconnus du problème

$\boldsymbol{\epsilon}$  : le bruit

Soit  $p(\mathbf{x}/\mathbf{y})$  : la loi *a posteriori* de  $\mathbf{x}$ . Selon la loi de Bayes :

$$p(\mathbf{x}/\mathbf{y}) = \frac{p(\mathbf{y}/\mathbf{x}) p(\mathbf{x})}{p(\mathbf{y})} \quad (1.16)$$

où  $p(\mathbf{y}/\mathbf{x})$  : la fonction de vraisemblance qui traduit l'imprécision sur les données.

$p(\mathbf{x})$  : la distribution *a priori* des paramètres inconnus qui traduit nos connaissances initiales sur la scène .

Un estimateur  $\hat{\mathbf{x}}$  au sens du maximum *a posteriori* (MAP) s'écrit :

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \{-\ln p(\mathbf{x}/\mathbf{y})\} = \arg \min_{\mathbf{x}} \{-\ln p(\mathbf{y}/\mathbf{x}) p(\mathbf{x})\} \quad (1.17)$$

Or  $p(\mathbf{y}/\mathbf{x}) \sim N(p(\mathbf{x}), \sigma_{\epsilon}^2)$ , par conséquent,

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \{\mathcal{J}(\mathbf{x})\} \quad (1.18)$$

où

$$\mathcal{J}(\mathbf{x}) = \mathcal{K}(\mathbf{x}) + \lambda \Omega(\mathbf{x}) \quad (1.19)$$

avec  $\mathcal{K}(\mathbf{x}) = \|\mathbf{y} - A(\mathbf{x})\|^2$

$\Omega(\mathbf{x})$  le terme de régularisation.

Les principales difficultés qui se posent en utilisant l'approche bayésienne est la détermination de la quantité  $p(\mathbf{x})$ , comment faut-il fixer l'information *a priori* sur l'objet et calculer par la suite la loi *a posteriori*  $p(\mathbf{x}/\mathbf{y})$ .

## 1.5 Sujet précis de mon stage

Plusieurs travaux ont été effectués dans le domaine de reconstruction d'objet en tomographie par rayon X en 2D et en 3D. Plus particulièrement, Charles Soussen, dans le Groupe des Problèmes Inverses (GPI), a travaillé sur ce sujet pendant sa thèse. Il a mis en oeuvre plusieurs programmes permettant la simulation des données, le calcul de ses projections, l'optimisation du critère et le calcul de la solution finale, et ceci en 2D et en 3D. Il a réussi à obtenir des résultats assez performants. Il a également élaboré l'approche multirésolution en 2D. Cependant, le travail reste inachevé dans le cas 3D.

Sur ce qu'a été déjà effectué, ma contribution porte sur la reconstruction 3D en multirésolution. Partant d'une modélisation polyédrique de l'objet, le problème de la reconstruction se résume à l'estimation directe des sommets  $\mathbf{V}$  de ce polyèdre à partir de ces projections  $\mathbf{p}$ . Durant cette étude, nous allons essayer d'établir les routines nécessaires permettant le passage d'un niveau de résolution à un autre, tout en essayant de relever les problèmes engendrés par cette approche.

## 1.6 Conclusion

Dans ce chapitre introductif, nous avons posé le problème de reconstruction d'objets compacts à partir d'un nombre faible de projections. Cette contrainte nous a amené à adapter une approche bayésienne et à introduire des informations *a priori* afin de rendre le problème bien posé. Nous avons présenté la nature des données et les différentes méthodes classiques pour la résolution de ce problème. Nous avons opté pour l'approche de modélisation de l'objet par une forme polyédrique. Dans le chapitre suivant, nous allons détailler cette approche en monorésolution et l'étendre par la suite pour l'approche multirésolution.

# Chapitre 2

## Approche polyédrique en monorésolution

### 2.1 Introduction

Dans ce chapitre, nous nous focalisons sur la représentation polyédrique pour le problème de reconstruction en 3D. Nous rappelons que la reconstruction se ramène à l'estimation des paramètres de la forme géométrique adoptée, qu'on note dans la suite par  $\mathcal{P}$ , à partir ses données radiographiques  $\mathbf{p}$ .

Plusieurs travaux ont été élaborés dans ce domaine. On cite, pour les méthodes d'estimation directe des sommets de  $\mathcal{P}$ , les travaux de ([15],[9], [18], [22]) pour le cas 2D et ceux de ([3], [23]) pour le 3D. On cite, notamment, les travaux de ([15],[17]) dans lesquels il propose une méthode pour estimer les paramètres inconnus à partir des moments de ses projections dans le cas 2D et par la suite, l'extension de cette méthode dans le cas 3D dans les travaux de ([24]). Dans tous les travaux qui ont été déjà cités, l'estimation es sommets de  $\mathcal{P}$  se fait en monorésolution, en d'autres terme, le nombre de sommets reste toujours le même.

Dans ce chapitre, nous allons présenter, le principe de l'estimation directe des sommets de  $\mathcal{P}$  ainsi que l'estimation à partir des moments de projections en monorésolution ainsi que la stratégie d'optimisation adoptée.

### 2.2 Présentation du modèle polygonale et polyédrique

L'idée de base est de modéliser le contour de l'objet par un polygone en 2D et par un polyèdre en 3D. Dans le premier cas, le polygone est définie seulement par les coordonnées de ses sommets  $\mathbf{V} = \{\mathbf{V}(x_i, y_i)\}$ . Par contre dans le cas 3D, un polyèdre  $\mathcal{P}$  est défini par un ensemble de n sommets et un ensemble de m facettes qui relie ces derniers entre eux. Le modèle de  $\mathcal{P}$  s'écrit alors comme suit :

$$\mathcal{P} = (\mathbf{V}, \mathbf{F})$$

$$\text{où } \mathbf{V} = [\mathbf{V}_1^T \dots \mathbf{V}_n^T]^T \text{ avec } \mathbf{V}_i = (x_i, y_i, z_i) \in \mathbb{R}^3 \text{ et } \mathbf{F} = [\mathbf{F}_1, \dots, \mathbf{F}_m]$$

Pour simplifier le problème d'estimation du polyèdre, on considère que les facettes sont triangulaires et que la relation entre les sommets voisins entre eux et entre les facettes voisines est prédéfinie [22].

Dans les deux cas, la relation entre les données et les paramètres inconnus n'est pas linéaire et l'équation à résoudre est :

$$\mathbf{p} = \mathcal{A}(\mathbf{V}) + \epsilon$$

où  $\mathcal{A}(\cdot)$  : l'opérateur de projection

Le critère à minimiser (1.19) devient alors :

$$\mathcal{J}(\mathbf{V}) = \|\mathbf{d} - \mathcal{A}(\mathbf{V})\|^2 + \lambda \Omega(\mathbf{V}) \quad (2.1)$$

## 2.3 Méthodes d'estimation de paramètres inconnus

Il existe deux méthodes de reconstruction en monorésolution, une méthode directe et une méthode utilisant les moments de projections. Cette dernière fera l'objet du paragraphe suivant, on présentera ses limites et ses avantages.

### 2.3.1 Méthodes des moments

Ces résultats sont dus essentiellement à Milanfar ([16] dans le cas 2D et à Soussen [24] dans le cas 3D. On résume dans ce qui suit ses travaux pour l'estimation des paramètres  $\mathbf{V}$  de l'objet  $\mathcal{P}$ .

Le principe de cette méthode consiste à :

- Trouver une relation entre les moments de l'objet et les moments de ses projections (les données) et estimer alors les moments de l'objet.
- Ensuite, trouver une relation entre les moments de l'objets et les sommets  $\mathbf{V}$  qui permet leur estimation.

Pour mieux illustrer cette méthode, nous allons traiter le cas 2D, plus simple, puis l'étendre dans le cas 3D. Partons de l'expression de projection :

$$p_\theta(r) = \iint_{\mathbb{R}^2} f(x, y) \delta(r - x \cos \theta - y \sin \theta) dx dy \quad (2.2)$$

Vu les conditions du problème (ref), l'expression(2.2) devient :

$$p_\theta(r) = \iint_{\mathbb{R}^2} \delta(r - x \cos \theta - y \sin \theta) dx dy$$

. Le moment d'ordre n de  $p_\theta(r)$  s'écrit :

$$h_n(\theta) = \int_{\mathbb{R}} r^n p_\theta(r) dr ; \quad n \geq 0 \quad (2.3)$$

Et le moment géométrique de  $\mathcal{P}$  est :

$$\mu_{ij} = \iint_{\mathbb{R}^2} x^i y^j f(x, y) dx dy = \iint_{\mathcal{P}} x^i y^j dx dy ; \quad i, j \geq 0 \quad (2.4)$$

On remarque qu'il y a une relation entre le moment de  $p_\theta(r)$  et le moment géométrique de  $\mathcal{P}$  et qui s'écrit :

$$h_n(\theta) = \iint_{\mathcal{P}} \left( \int_{\mathbb{R}} r^n \delta(r - x \cos \theta - y \sin \theta) dr \right) dx dy \quad (2.5)$$

$$= \iint_{\mathbb{R}} (x \cos \theta + y \sin \theta)^n dx dy \quad (2.6)$$

Donc,

$$h_n(\theta) = \cos^n \theta \sum_{j=0}^n C_n^j \sin^j \theta \mu_{n-j,j} \quad (2.7)$$

On remarque que le moment d'ordre  $n$  des projections dépend uniquement du moment géométrique d'ordre  $n$  de l'objet. Par conséquent, pour  $n$  fixé, on aura besoin au moins de  $(n + 1)$  valeurs de  $h_n(\theta)$ , donc de  $(n + 1)$  projections différentes pour estimer tous les moments  $\mu_{n-j,j}$ .

On peut étendre facilement ces équations dans le cas 3D. L'expression du moment géométrique devient alors :

$$\mu_{ijk} = \iiint_{\mathcal{P}} x^i y^j z^k dx dy dz ; \quad i, j, k \geq 0$$

et le moment de projection, en se référant à(1.1), devient :

$$h_{lq}(\theta) = \iint_{\mathbb{R}^2} u^l v^q p_{\theta}(\mathbf{r}) d\mathbf{r} ; \quad \text{où } \mathbf{r} \in \omega_{\theta}^{\perp}, \quad \mathbf{r} = u \mathbf{u} + v \mathbf{v}$$

De la même manière qu'au cas 2D, il est possible de calculer les moments géométriques de  $\mathcal{P}$  aux ordres  $0 \dots n$  à partir d'au moins  $(n + 1)$  projections ([24]), ce qui permet la résolution de la première étape de la méthode.

Pour la 2<sup>ème</sup> étape, pour l'évaluation de  $n$  sommets  $V_1 \dots V_n$  nécessite au moins  $(2n - 2)$  moments estimés, donc de  $(2n - 2)$  projections. Comme, le nombre de projections pour notre problème est limité, on ne peut pas avoir des solutions satisfaisantes.

Cependant, cette méthode est facile et rapide point de vue implémentation et elle fournit des résultats performants pour un nombre réduit de sommets. Etant donnés ces avantages, on peut en profiter pour construire un polyèdre convexe à partir des moments de projections jusqu'à l'ordre 2 et qui sera considéré comme initialisation de notre algorithme de reconstruction de  $\mathcal{P}$ .

### 2.3.2 Méthode d'estimation directe

Nous avons vu dans le paragraphe précédent qu'il est difficile d'estimer d'une façon exacte la forme de l'objet  $\mathcal{P}$  à partir des moments de ces projections. Cependant, ([18]) ont proposé des méthodes qui permettent un calcul approximatif de la solution, en se basant sur l'approche bayésienne qui permet la régularisation du critère.

On a défini le critère comme :

$$\mathcal{J}(\mathbf{V}) = \mathcal{K}(\mathbf{V}) + \lambda \Omega(\mathbf{V})$$

En l'absence de l'information *a priori* particulière sur la forme de l'objet, Soussen([22]) a choisi de favoriser sa douceur locale.  $\Omega(\mathbf{V})$  sera une énergie à caractère markovien qui pénalise les fortes courbures de l'objet  $\mathcal{P}$ . Des modèles de régularisation ont été développés abondamment par ([4] et [3]).

Le terme de régularisation s'écrit :

$$\Omega(\mathbf{V}) = \sum_{i=1}^N \phi(k_i(\mathbf{V})) \quad (2.8)$$

où  $K_i(\mathbf{V})$  : Les courbures locales approchées en les sommets  $V_i$  et qui ne dépend que de  $V_j$  voisins de  $V_i$ .

$\phi(\cdot)$  : une fonction choisie paire et croissante.

Il y a plusieurs façons pour définir les courbures locales, on peut prendre la définition de l'angle solide  $A_i(\mathbf{V})$  ([22]) qui permet d'écrire l'expression suivante de la courbure :

$$K_i(\mathbf{V}) = K(A_i(\mathbf{V})) = 1 + \cos A_i(\mathbf{V})/2 \quad (2.9)$$

où  $A_i(\mathbf{V})$  :angle solide qui est égale à l'aire de la surface obtenue en projetant tous les sommets voisins de  $V_i$  sur la sphère unité centré en  $V_i$ .

Le terme de régularisation s'écrit alors :

$$\Omega(\mathbf{V}) = \sum_{i=1}^N \phi(1 + \cos A_i(\mathbf{V})/2) \quad (2.10)$$

Il y a plusieurs formes de  $\phi$ , pour ce cas, on choisit un potentiel convexe , simple  $\phi(t) = t^2$ . Et le critère final à optimiser devient :

$$\mathcal{J}(\mathbf{V}) = \|d - \mathcal{A}(\mathbf{V})\|^2 + \sum_{i=1}^N (1 + \cos A_i(\mathbf{V})/2)^2 \quad (2.11)$$

## 2.4 Techniques d'optimisation

La reconstruction de l'objet  $\mathcal{P}$  se ramène à l'optimisation du critère(2.11)  $\mathcal{J}(\mathbf{V})$ , un critère non convexe. D'une façon générale, il existe deux grandes classes de méthodes d'optimisation : des méthodes stockastiques et des méthodes itératives déterministes qu'on retient pour cette étude. Il y a deux stratégies d'optimisation pour les méthodes itératives.

- **Technique qui modéfie l'ensemble des paramètres à chaque itération :**

Cet algorithme optimise  $\mathcal{J}$  directement sur tous les sommets simultanément et il s'écrit :

$$\mathbf{V}^{(k+1)} = \mathbf{V}^{(k)} - h_k \nabla \mathcal{J}(\mathbf{V}^{(k)}), \quad \text{où } h_k = \arg \min_{h \in \mathbb{R}_+} \mathcal{J}(\mathbf{V}^{(k)} - h \nabla \mathcal{J}(\mathbf{V}^{(k)}))$$

$h_k$  représente le pas de descente de l'algrithme. Cette technique fournit des résultats satisfaisants[22], mais elle nécessite un grand nombre d'itérations.

- **Technique qui modéfie un sommet à la fois :**

Cette algorithme consiste à réaliser séquentiellement la minimisation du critère  $\mathcal{J}(\mathbf{V})$  par rapport à chaque sommet  $V_i$  en considérant les autres fixés. Donc, les sommets sont mis à jour de la manière suivante :

$$\forall i \in \{1, \dots, n\}, \quad V_i^{(k+1)} = V_i^{(k)} - h_i^{k+1} \nabla \mathbf{g}_i(V_i^k; V_j, i \neq j)$$

où  $\mathbf{g}_i(V_i^k; V_j, i \neq j)$  représente la dérivée partielle  $\partial \mathcal{J}(Vb)/\partial V_i$  et le pas de descente  $h_i^{k+1}$  est obtenue par :

$$h_i^{k+1} = \arg \min_{h \in \mathbb{R}_+} \mathcal{J}(V_i^{(k)} - h \mathbf{g}_i(V_i^{(k)}; V_j, i \neq j))$$

Le problème de ces algorithmes est l'initialisation car il faut que la solution initiale appartient à la région attractive du minimum globale sinon on risque de rester bloquer dans un minimum local.

Ces deux techniques ont été adoptées par Soussen [22] et il s'avère qu'elles sont assez performantes. Pour notre approche multirésolution, nous allons utiliser pour le moment la première

technique. Un problème qui peut être engendré par l'étape d'optimisation et auquel il faut faire attention est à l'auto-intersection ou le caractère acceptable de la forme reconstruite. Un polyèdre est dit acceptable s'il appartient à l'ensemble  $\mathbb{A}_{nm}$ , où  $n$  est le nombre de sommets et  $m$  est le nombre de facettes, et il est défini comme suit :

$$\mathbb{A}_{nm} = \{ \mathcal{P} = \{ \mathbf{V}, \mathbf{F} \} / \forall (i, j) \in \{1 \dots m\}, i \neq j, i \approx j, \mathbf{F}_i \cap \mathbf{F}_j = \emptyset \} \quad (2.12)$$

Pour essayer de conserver le caractère acceptable du polyèdre, Soussen [22] a proposé les solutions suivantes :

- Pour la stratégie de descente par blocs, on suppose que l'algorithme est à l'itération  $k$ . On définit la quantité  $\mathbf{p} = \mathcal{A}(\mathbf{V}^{k-1})$ , cette quantité sera mis à jour pour chaque sommet déplacé. Une fois l'itération  $k$  terminée, les  $n$  sommets  $V_1^{(k)} \dots V_n^{(k)}$  ont été ré-estimés. Donc, on calcule  $\mathcal{A}(\mathbf{V}_k)$  et puis on le compare avec  $\mathbf{p}$ . Si  $\|\mathcal{A}(\mathbf{V}_k) - \mathbf{p}\| > \epsilon$ , le polyèdre n'appartient pas à  $\mathbb{A}_{nm}$  défini par l'équation 2.12. Par conséquent, on rejette les modifications des sommets faites à l'itération  $k$ .
- Pour la stratégie du gradient, cette procédure n'est pas envisageable car les sommets sont modifiés globalement. Mais une possibilité de contrôler le caractère acceptable consiste à comparer les dérivées partielles  $\partial \mathcal{J}(\mathbf{V}^{(k)}) / \partial V_i$  à leurs approximations par différences finies.

Outre les difficultés inhérentes aux calculs des projections et de leur gradient, le problème de la gestion du caractère acceptable de  $\mathcal{P}$  au sens de la définition(2.12) dans le cas 3D est beaucoup plus difficile vu que l'on déforme des contours 3D. Une méthode de détecter les auto-intersections est l'utilisation d'une voxellisation comme le font Battle et al.[3]. Ils répertorient les facettes qui intersectent chaque voxel, et détectent une auto-intersection si deux facettes non adjacentes intersectent le même voxel. Cette stratégie est efficace, mais elle s'avère coûteuse et nécessite une voxellisation fine et de nombreuses recherches d'intersection voxel-facettes.

Nous avons présentés dans ce paragraphe les deux stratégies d'optimisation qu'on peut utiliser pour la reconstruction de l'objet  $\mathcal{P}$ , ainsi que le problème engendré par ces derniers à savoir la gestion du caractère acceptable de  $\mathcal{P}$ . Dans ce qui suit, nous allons développer les éléments nécessaires à la mise en oeuvre de ces algorithmes, en particulier les différentes étapes de calcul du critère et de son gradient.

## 2.5 Différentes étapes de calcul du critère et de son gradient

Dans l'expression de  $\mathcal{J}(\mathbf{V})$ (2.11) et de son gradient, il faut calculer le terme  $\mathcal{A}(\mathbf{V})$ , c'est à dire calculer les projections de  $\mathbf{V}$  et le terme  $\partial \mathcal{A}(\mathbf{V}) / \partial v_i$ , ce qui revient à mettre à jour les projections. C'est ces deux thèmes qu'on va traiter dans ce paragraphe.

### 2.5.1 Calcul des projections

Soit le polyèdre  $\mathcal{P}$  et  $\mathcal{L} = \mathcal{L}_{\mathbf{r}, \boldsymbol{\omega}}$  le rayon de projection passant par la source S, orienté par  $\boldsymbol{\omega} = \boldsymbol{\omega}_{\theta}$  et arrivant en le pixel détecteur  $\mathbf{r}$ . En se référant à [1], le calcul de  $p_{\theta}(\mathbf{r})$  dépend des intersections entrantes et sortantes de  $\mathcal{L}$  et les facettes  $\mathbf{F}$  de  $\mathcal{P}$ . En omettant le cas singulier où le rayon appartient au support de la facette, notons  $I_j$  le point d'intersection  $I_j = \mathcal{L} \cap \mathbf{F}_j$ , le nombre de ces intersections est pair ou impair selon que le polyèdre soit convexe ou non. Soit  $l_j$  la distance entre  $I_j$  et la source S2.1

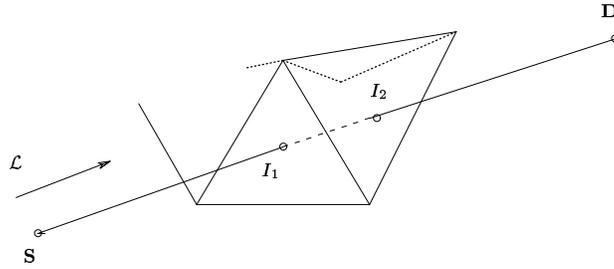


FIG. 2.1 – Intersection rayon-facettes

Soit,

$$I_j(l_j) = S + l_j \omega \quad (2.13)$$

La quantité  $p_{\theta}(\mathbf{r})$  s'écrit alors :

$$p_{\theta}(\mathbf{r}) = \sum_{j=1}^k d(I_{2j-1}, I_{2j}) = \sum_{j=1}^k (l_{2j} - l_{2j-1}) \quad (2.14)$$

les positions  $I_{2j}$  correspondent aux points d'entrée du rayon, et les positions  $I_{2j-1}$  aux points de sortie.

### 2.5.2 Mise à jour des projections

La mise à jour des projections est nécessaire lors du calcul du gradient de  $\mathcal{J}$ , chaque fois qu'on déplace un sommet il faut recalculer la projection de l'ensemble du polyèdre. Supposons qu'on est à l'itération  $k$  et qu'il faut recalculer  $\mathbf{p} = \mathcal{A}(\mathbf{V}^{(k)})$  suite à la modification du sommet  $V_1$ . Le principe de cette stratégie de mise à jour consiste en premier lieu à l'extraction du mini polyèdre  $\mathcal{P}_1$  contenant le sommet  $V_1$  et ses voisins et en second lieu, à la décomposition de  $\mathcal{P}_1$  en  $n_1$  tétraèdres où  $n_1$  est le nombre de voisins du sommets  $V_1$ .

Ainsi, on peut écrire que le polyèdre  $\mathcal{P}$  :

$$\mathcal{P} = \mathcal{P}_1^{\perp} + \epsilon_1 \mathcal{P}_1 \quad (2.15)$$

où  $\mathcal{P}_1$  : le mini-polyèdre extrait contenant le sommet  $V_1$ , ses voisins et un sommet fictif  $W$  rajouté pour le renfermer.

$\mathcal{P}_1^{\perp}$  : un polyèdre contenant  $W$  et tous les sommets de  $\mathcal{P}$  sauf  $V_1$ .

$\epsilon_1 = \pm 1$  indique si  $V_1$  est intérieur ou extérieur à  $\mathcal{P}$ .

Nous déduisons de (2.5.2) que :

$$p_{\mathbf{V}}(\mathbf{r}) = p_{\mathcal{P}_1^{\perp}}(\mathbf{r}) + \epsilon_1 \sum_{i=1}^{n_1} p_{\mathcal{T}_i}(\mathbf{r}) \quad (2.16)$$

Dans cette équation, le premier terme est indépendant de  $V_1$ . Donc, la modification de ce sommet entraîne la mise à jour seulement des tétraèdres  $\mathcal{T}_i$ . A l'itération  $k$ , la projection du polyèdre  $\mathcal{P}$  s'écrit :

$$\mathcal{A}(\mathbf{V}^{(k)}) = \mathcal{A}(\mathbf{V}^{(k-1)}) - \epsilon_i^{(k-1)} \mathcal{A}(\mathcal{T}_i^{(k-1)}) + \epsilon_i^{(k)} \mathcal{A}(\mathcal{T}_i^{(k)}) \quad (2.17)$$

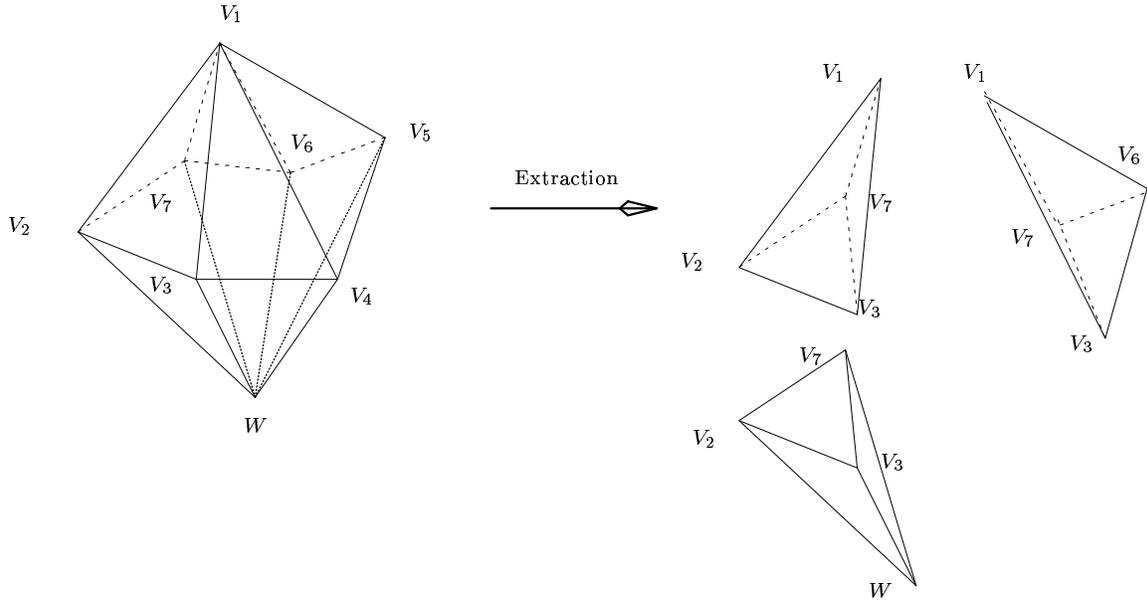


FIG. 2.2 – Extraction du mini polyèdre contenant  $V_1$ ,  $W$  est le sommet fictif et décomposition en tétraèdres

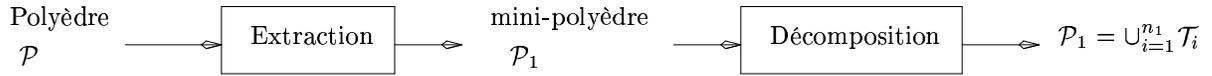


FIG. 2.3 – Extraction du mini polyèdre contenant  $V_1$

Pour la mise à jour des projections, il suffit de remettre en question que le mini-polyèdre contenant le sommet modifié et ses voisins, ce qui réduit nettement la charge de calcul et rend la procédure de mise à jour facile à implémenter.

## 2.6 Calcul du gradient du critère

Le calcul du gradient du critère  $\nabla \mathcal{J}$  dépend directement du calcul des mise à jour des projections des sommets de  $mP$ . On sait que  $\nabla \mathcal{J}(\mathbf{V}) = [\partial \mathcal{J} / \partial V_1, \dots, \partial \mathcal{J} / \partial V_n]^T$ , par conséquent pour calculer le gradient revient à calculer les dérivées partielles du critère par rapport à  $V_i$ . Rappelons que le critère à optimiser est :

$$\mathcal{J}(\mathbf{V}) = \mathcal{K}(\mathbf{V}) + \lambda \Omega(\mathbf{V})$$

Donc,

$$\partial \mathcal{J}(\mathbf{V}) / \partial V_i = \partial \mathcal{K}(\mathbf{V}) / \partial V_i + \lambda \partial \Omega(\mathbf{V}) / \partial V_i$$

### 1. Calcul de $\partial \mathcal{K}(\mathbf{V}) / \partial V_i$ :

On rappelle que :

$$\mathcal{K}(\mathbf{V}) = \|\mathbf{d} - \mathcal{A}(\mathbf{V})\|^2 = \sum_{k=1}^M \sum_{i,j=1}^{r_x, r_y} (d_{ij}^k - \mathcal{A}_{ij}^k(\mathbf{V}))^2 \quad (2.18)$$

où  $\mathbf{d}_k = \{d_{ij}^k\}$  est la  $k^{ime}$  image radiographique de dimensions  $r_x, r_y$ .

$\mathcal{A}_{ij}^k(\mathbf{V})$  est la projection des sommets  $\mathbf{V}$  suivant la  $k^{ime}$  projection.

$M$  est le nombre de projections.

La dérivée partielle de  $\mathcal{K}(\mathbf{V})$  par rapport au sommet  $V_1$ , par exemple, s'écrit alors :

$$\partial\mathcal{K}(\mathbf{V}) = 2 \sum_{k=1}^M \sum_{i,j=1}^{r_x, r_y} \partial\mathcal{A}_{ij}^k(\mathbf{V})/\partial V_1 (d_{ij}^k - \mathcal{A}_{ij}^k(\mathbf{V})) \quad (2.19)$$

Si on considère une seule projection et une seule position  $(i, j)$ , en se référant à l'équation(2.17) alors

$$\partial\mathcal{A}_{ij}^k(\mathbf{V})/\partial V_1 = \epsilon_1 \partial\mathcal{A}_{ij}^k(\mathcal{P}_1)/\partial V_1 = \epsilon_1 \partial\mathcal{A}_{ij}^k(\mathcal{T}_r)/\partial V_1 \quad (2.20)$$

Donc, l'expression(2.19) devient alors :

$$\partial\mathcal{K}(\mathbf{V}) = 2 \epsilon_1 \sum_{k=1}^M \sum_{r=1}^{n_1} \sum_{i,j=1}^{r_x, r_y} \partial\mathcal{A}_{ij}^k(\mathcal{T}_r)/\partial V_1 (d_{ij}^k - \mathcal{A}_{ij}^k(\mathbf{V})) \quad (2.21)$$

pour résumer, le calcul de la dérivée partielle de  $\mathcal{K}(\mathbf{V})$  revient à remettre à jour seulement la projection du mini-polyèdre contenant le sommets modifiés et ses voisins.

## 2. Calcul $\partial\Omega(\mathbf{V}) / \partial V_i$ :

Le calcul des dérivées partielles de  $\Omega(\mathbf{V})$  est relativement simple car il s'exprime de façon analytique. On rapelle que l'expression du terme de régularisation est :

$$\Omega(\mathbf{V}) = \sum_{i=1}^N \left(1 + \frac{\cos A_i(\mathbf{V})}{2}\right)^2$$

Et dont la dérivée partielle s'écrit :

$$\partial\Omega(\mathbf{V})/\partial V_i = 2 \sum_{j=1}^N \partial \cos(A_j(\mathbf{V})/2)/\partial V_i (1 + \cos(A_j(\mathbf{V})/2)) \quad (2.22)$$

$$= \sum_{j=1}^N \partial A_j(\mathbf{V})/\partial V_i \sin(A_j(\mathbf{V})/2) (1 + \cos(A_j(\mathbf{V})/2)) \quad (2.23)$$

Ainsi, le calcul de cette drivée partielle se résume au calcul de la dérivée partielle de l'angle solide  $A_j(\mathbf{V})$ .

## 2.7 Conclusion

On a présenté dans ce chapitre le modèle polyédrique, les techniques d'optimisation ainsi que la méthodes de calcul des projections et leur mise à jour qui est nécessaire pour le calcul du gradient du critère, et ceci était dans le cas de monorésolution. Cette approche a pour principe de fixer le nombre de sommets de l'objet  $\mathcal{P}$ . On constate que les performances des résultats dépendent nettement du nombre de sommets qu'on fixe au départ (cf chapitre result). Pour surmonter cette difficulté, nous proposons l'approche multirésoltuion qui sera bien détaillée dans le chapitre suivant.

# Chapitre 3

## Approche multirésolution

### 3.1 Introduction

On a vu que la difficulté de la reconstruction en 3D en monorésolution, hormis celle des auto-intersections, est le choix du nombre des sommets du polyèdre. Ce nombre dépend de la complexité de la forme à reconstruire. On propose donc comme amélioration l'approche multirésolution qui prend en considération ce problème.

Cette approche consiste à augmenter progressivement le nombre des sommets du polyèdre. A chaque niveau de résolution, nous utilisons l'estimation obtenue dans le niveau précédent comme initialisation. Soit  $\mathcal{P}^{(0)} = (\mathbf{V}^{(0)})$  l'initialisation du premier niveau. A chaque niveau de résolution  $r$ , le polyèdre  $\mathcal{P}^{(r)} = (\mathbf{V}^{(r)})$  est évalué à partir de la dernière estimation  $\mathcal{P}^{(r-1)} = (\mathbf{V}^{(r-1)})$  en optimisant le critère :

$$\mathcal{J}^{(r)}(\mathbf{V}) = \|\mathbf{p} - \mathcal{A}(\mathbf{V}^{(r)})\|^2 + \lambda^{(r)} \Omega(\mathbf{V}^{(r)}) \quad (3.1)$$

Le passage d'un niveau de résolution à un autre implique le réglage d'un certain nombre de paramètres à savoir le nombre de sommets à ajouter, leur emplacement et le choix de l'hyperparamètre  $\lambda^{(r)}$  pour chaque niveau. Pour cette étude, nous allons nous intéresser plus aux problèmes de l'ajout des sommets qui vont être détaillés par la suite. Pour plus de clarté, on commencera par présenter le principe et les résultats de cette approche dans le cas 2D. On passera par la suite à son extension dans le cas 3D.

### 3.2 La multirésolution en 2D

Pour le cas 2D, le principe de la multirésolution consiste à commencer par  $N$  sommets comme initialisation et qui est en générale faible. Ensuite à chaque niveau de résolution  $r$ , on ajoute un sommet au milieu de chaque arête  $(V_j, V_{j+1})$  du polygone  $\mathcal{P}$  (figure (3.1)). Donc, toutes les arêtes seront divisées en deux et le polygone manipulé au niveau  $r$  aura approximativement  $2^{r-1} \times N$  sommets.

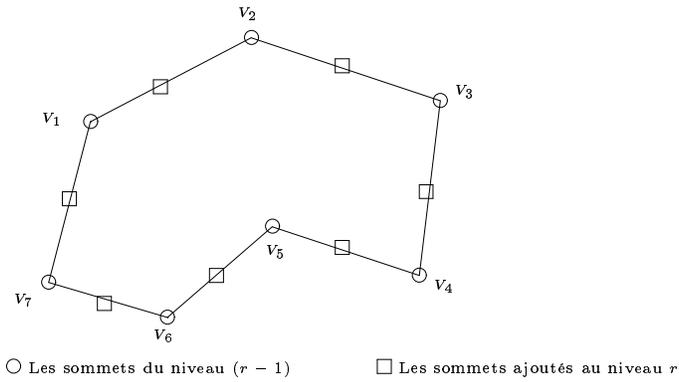


FIG. 3.1 – Insertion des sommets aux milieux des arêtes

Pour illustrer l'approche multirésolution dans le cas 2D, on présente, en premier lieu, par la figure (3.2) la reconstruction du polyèdre  $\mathcal{P}$  en monorésolution. Cette reconstruction est initialisée par une ellipse de 40 sommets [23].

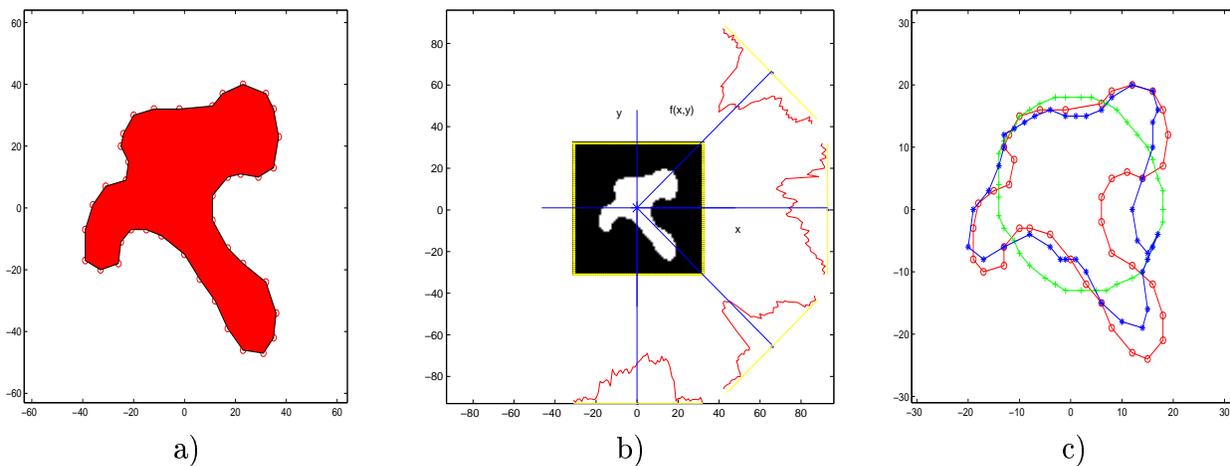


FIG. 3.2 – Reconstruction en monorésolution : a) Objet original, b) les données simulées à RSB= 20 dB, c) trois polygones : l'ellipse d'initialisation de l'algorithme, discrétisée avec 40 sommets. Le polygone en gras est celui reconstruit après 50 itérations. Et le polygone originale est représenté aussi pour la comparaison

Les résultats de la méthode de multirésolution sont illustrés par la figure (3.3). Pour cette exemple, [23] ont choisi 4 niveaux de résolution. Le nombre de sommets initiales est  $N = 5$  et on arrive à la fin avec  $N = 40$  sommets. Pour chaque niveau 10 itérations sont exécutées.

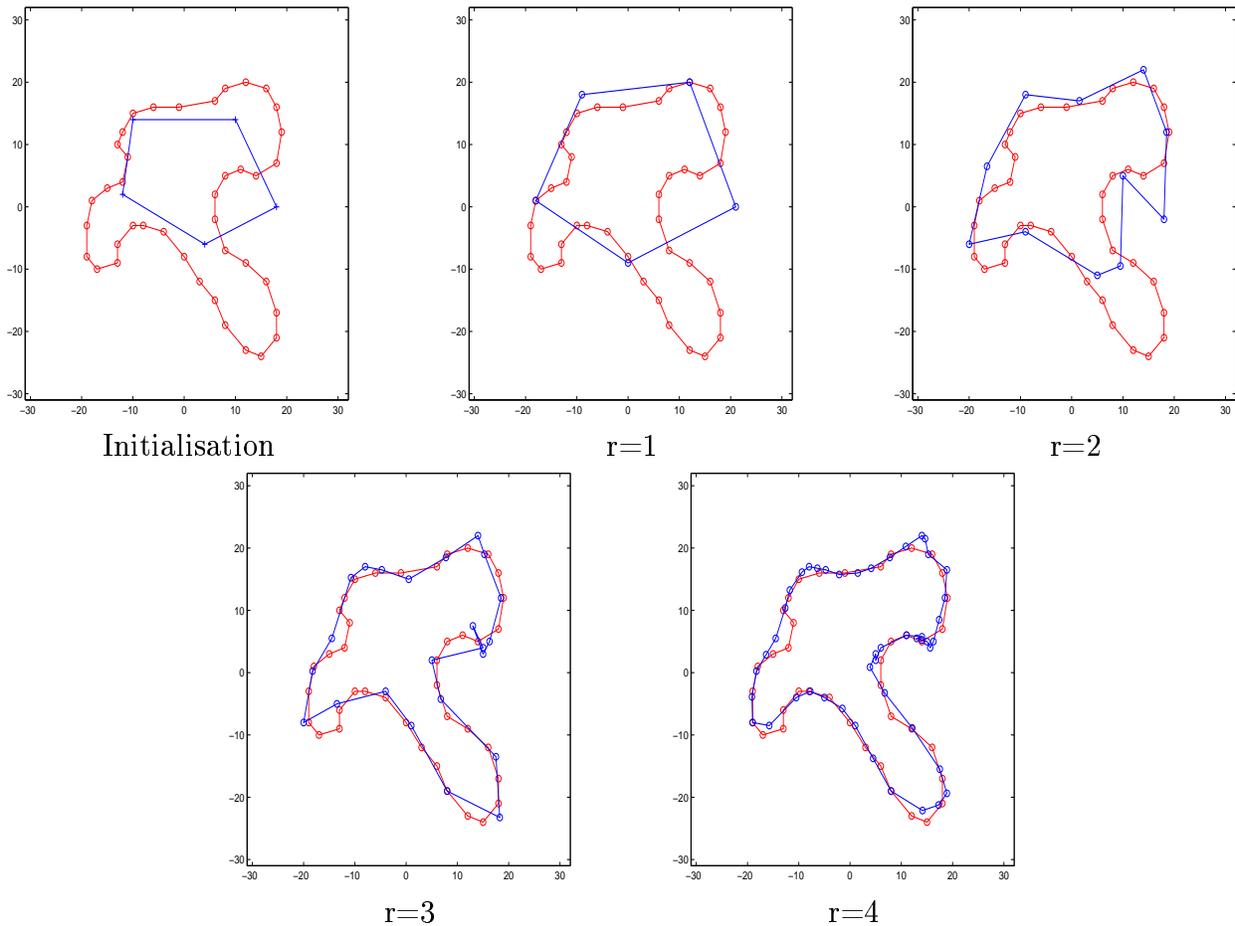


FIG. 3.3 – Solution initiale et les reconstruction en multirésolution pour 4 niveaux de résolutions

On remarque bien à travers ces deux exemples (3.2, 3.3), l'amélioration qu'apporte l'approche multirésolution où le choix du nombre initiale des sommets n'est pas primordial sur la qualité de la solution finale, alors qu'il l'est pour l'approche monorésolution dont le résultat n'est pas aussi bien à cause du blocage de la solution dans un minimum local. parlons maintenant de la charge de calcul, pour l'approche monorésolution, on a effectué 50 itérations et pour chacune d'elles  $N = 40$  sommets sont traités, Contrairement à l'approche multirésolution où on a effectué 4 niveaux de résolution avec 10 itérations chacun, mais le nombre de sommets traités à chaque niveau est nettement inférieur à  $N$ , et ce n'est qu'au dernier niveau que les  $N$  sommets seront traités. Cette comparaison nous permet de dire que la charge de calcul de l'approche multirésolution est bien inférieure à celle de l'approche monorésolution.

### 3.3 Approche multirésolution en 3D

Dans le paragraphe précédent, nous avons étudié l'approche multirésolution dans le cas 2D, nous avons présenté son principe et quelques résultats montrant les performances de cette approche. Dans celui ci, nous allons présenter l'extension de cette approche dans le cas 3D.

le principe de la multirésolution est semblable à celui ds le cas 2D. En effet, on part de  $\mathcal{P}^{(0)} = (\mathcal{V}^{(0)}, \mathcal{F}^{(0)})$  comme données initiales. A chaque niveau de résoluton  $r$ , le nombre de sommets augmente selon un certain algorithme (qu'on va détailler par la suite), et il faut estimer

$\mathcal{P}^{(r)} = (\mathcal{V}^{(r)}, \mathcal{F}^{(r)})$  en considérant les données du niveau précédent  $\mathcal{P}^{(r-1)} = (\mathcal{V}^{(r-1)}, \mathcal{F}^{(r-1)})$  comme initialisation ??.

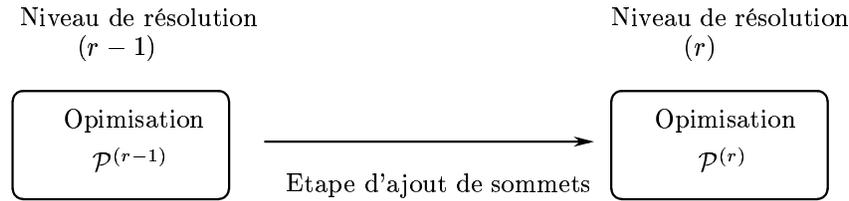


FIG. 3.4 – Principe de la multirésolution

A chaque niveau de résolution  $r$ , on utilise les mêmes algorithmes de l'approche monorésolution, déjà décrits dans le chapitre précédent à savoir les techniques d'optimisation, le calcul du critère et son gradient. Donc le critère à optimiser reste le même et il s'écrit en se référant à (2.11) :

$$\mathcal{J}^{(r)}(\mathcal{V}^{(r)}) = \mathcal{K}(\mathcal{V}^{(r)}) + \lambda^{(r)} \Omega(\mathcal{V}^{(r)}) \quad (3.2)$$

La difficulté majeure de cette approche se résume essentiellement au module de l'ajout des sommets. Cette étape est un peu difficile car il faut respecter la contrainte d'avoir toujours une forme non dégénérée. Offenser cette contrainte a pour conséquence de permettre les auto-intersections (définis dans (2.4)) qui empêchent l'étape d'optimisation d'arriver à terme de la solution finale. Les deux grandes questions auxquelles on va essayer de répondre durant ce paragraphe sont où et combien faut-il ajouter les sommets en gardant une forme cohérente de l'objet  $\mathcal{P}$  ?

### 3.3.1 Différentes méthodes de triangulation

Pour répondre à la question où faut-il ajouter des sommets, nous avons fait des recherches dans la littérature. Alors nous avons trouvé quelques méthodes qui permettent d'avoir une triangulation régulière.

#### Méthode de Delaunay [2]

Une méthode qui peut être intéressante pour traiter notre problème. Nous présentons ici des brèves définitions de la triangulation de Delaunay :

- Etant donné un ensemble de points, la triangulation de Delaunay est un ensemble de lignes connectant chaque point à ses voisins naturels.
- Pour un ensemble de sommets  $A$ , la triangulation de Delaunay [14] est une triangulation de telle sorte que pour chaque cercle circonscrit à un triangle formé par trois sommets de  $A$ , alors aucun autre sommet de  $A$  n'est à l'intérieur de ce cercle (3.5).

Plusieurs travaux ont été réalisés en se basant sur la triangulation de Delaunay : [13][5][14]. Pour notre problème, nous avons pensé s'inspirer de cette méthode aussi de la manière suivante : à chaque niveau de résolution, on ajoute des sommets à l'objet, et partant de ce nouvel ensemble de sommets, on refait la triangulation de la surface de polyèdre en utilisant le principe de la

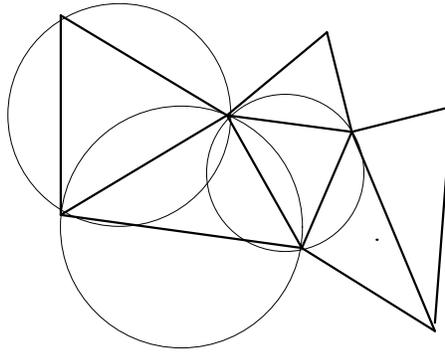


FIG. 3.5 – Triangulation de Delaunay

triangulation de Delaunay. Cependant, dans les travaux déjà cités, la triangulation de Delaunay a été utilisée pour décomposer une surface plane en un ensemble de triangles de telle sorte qu'on ne trouve pas de chevauchement entre les arêtes des triangles.

Dans[2], ils ont utilisés la triangulation de Delaunay dans le cas 3D, leur but est de construire des polyèdres convexes, ce qui n'est pas notre cas où les formes qu'on est amené à reconstruire sont généralement complexes.

### Méthode Lachaud

Lachaud dans [12] a proposé un modèle qui adapte sa topologie à la géométrie de ses sommets à chaque étape sans l'intervention d'un utilisateur et sans aucune grille incluse dans l'image. Cette méthode est basée sur des contraintes géométriques appliquées sur les sommets du modèle.

Avant d'étudier ces contraintes, nous allons définir quelques notations nécessaires à leur compréhension par la suite :

- Soient  $\mathcal{V}$  et  $\mathcal{F}$  respectivement l'ensemble de sommets et de facettes triangulaires de l'objet  $\mathcal{P}$ .
  - Pour un sommet  $v \in \mathcal{V}$ , on note  $\nu(v)$  l'ensemble des voisins de  $v$  et  $\mathbf{v}$  les coordonnées de  $v$
- Les contraintes définies dans[12] pour la triangulation sont :

$$\forall (u, v) \in \mathcal{V} \times \mathcal{V} / v \in \nu(u), \quad \delta < \|\mathbf{u} - \mathbf{v}\| \quad (3.3)$$

$$\forall (u, v) \in \mathcal{V} \times \mathcal{V} / v \in \nu(u), \quad \|\mathbf{u} - \mathbf{v}\| < 2.5 \delta \quad (3.4)$$

$$\forall (u, v) \in \mathcal{V} \times \mathcal{V} / v \notin \nu(u), \quad \frac{2.5}{\sqrt{3}} \delta \leq \|\mathbf{u} - \mathbf{v}\| \quad (3.5)$$

Les contraintes (3.3) et (3.4) expriment les bornes inférieure et supérieure de la longueur de l'arête. Elles obligent la surface triangulaire à rester relativement régulière. Et si la contrainte (3.5) n'est pas vérifiée pour un couple de voisins ça signifie qu'il y a une collision entre deux parties distinctes de la surface du polyèdre. Cette contrainte peut être utilisée pour vérifier l'existence des auto-intersections.

### 3.3.2 Méthode proposée

Nous n'allons pas nous intéresser actuellement à la méthode de Delaunay vu les difficultés qu'elle présente dans le cas 3D. Par contre nous allons adopter une autre stratégie inspirée des

travaux de Lachaud (??) pour essayer de garder au mieux une forme compact et homogène de  $\mathcal{P}$ . Cette stratégie se divise en trois parties qui vont être détaillées dans ce qui suit.

- Combien de sommets ajouter ?
- Où les ajouter ?
- Simplifications pour avoir un objet homogène.

### Combien de sommets ajouter ?

Pour un premier temps, nous avons pensé à diviser les triangles de grandes surfaces, c'est à dire, à ajouter des sommets dans les triangles dont la surface est supérieure à un certain seuil  $S_m$ . Supposons qu'on est au niveau de résolution  $r$ , l'algorithme consiste, comme est montré au tableau (3.1), à calculer les aires des triangles de  $\mathcal{F}^{(r)}$  et choisir par la suite les triangles dont l'aire est supérieure à  $S_m$  qui n'est autre que la moyenne des surfaces.

```

/* Sélection des triangles*/
Pour  $i = 1, \dots, m$ , /* m : nombre de triangles*/
    Calcul des aires des triangles.
Calcul de la moyenne des aires =  $S_m$ .
Sélectionner les triangles : Sélection= triangles
dont l'aire  $> \alpha S_m$ .
Fin Pour.

```

TAB. 3.1 – Méthode de détermination de lieu d'ajout des sommets

Cette étape nous permettra de choisir approximativement le nombre de sommets à ajouter selon le choix de  $\alpha$  défini dans (3.1), on utilisera ce paramètre pour ajuster la solution (cf ??).

### Où faut-il les ajouter ?

Il faut mentionner ici que durant toutes les étapes de la reconstruction que ce soit en mono ou en multirésolution, les données traitées  $\mathcal{V}$  et  $\mathcal{F}$  sont toujours sauvegardées sous la forme indiquée dans le tableau (3.2).

```

/* Sauvegarde de  $\mathcal{V}$  et  $\mathcal{F}$ */
  n :nombre de sommets.
   $x_1$   $y_1$   $z_1$ .
  . . .
  . . .
   $x_n$   $y_n$   $z_n$ .

  m :nombre de triangles.
   $V_i$   $V_j$   $V_k$ .
  . . .
  . . .
  . . .

```

TAB. 3.2 – Sauvegarde de la liste des sommets et des triangles

Par conséquent, chaque ajout de sommet implique une mise à jour de la liste des sommets et de la liste des triangles. Ce qui constitue parmi d'autres une contrainte pour l'emplacement des sommets. Pour placer les nouveaux sommets, nous avons pensé à deux stratégies :

1. **La première stratégie** consiste à placer les sommets au centre de gravité des triangles sélectionnés selon l'algorithme du (3.3). Ainsi, le triangle en question sera divisé en trois autres, ce processus est décrit par la figure(3.6).

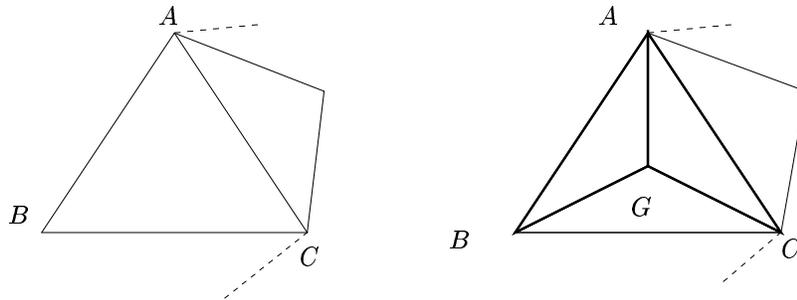


FIG. 3.6 – Insertion de sommet. Triangle a) est divisé en trois triangles b) qui se partagent un sommet commun, le centre de gravité G

Un sommet ajouté implique l'ajout d'un élément dans la liste des sommets et le remplacement d'un triangle par trois autres dans la liste des triangles comme est décrit dans (3.3). Le résultat de cette méthode est illustré par la figure (3.7).

```

/*Ajout de sommet*/
Pour  $i = 1, \dots, q$ , /*  $q$  : nombre de triangles sélectionnés*/
  /*Soit le triangle  $(ABC)$ */
  Calcul des coordonnées du centre de gravité  $G$ .
  Ajouter les coordonnées de  $G$  à la liste des sommets  $\mathcal{V}$ .
  Composer les nouveaux triangles :  $(ABG)$ ,  $(AGC)$  et  $(BCG)$ .
  Ajouter ces triangles à la liste des triangles  $\mathcal{F}$ .
Fin Pour.

```

TAB. 3.3 – Méthode d'ajout des sommets et de mise à jour des listes de sommets et de triangles

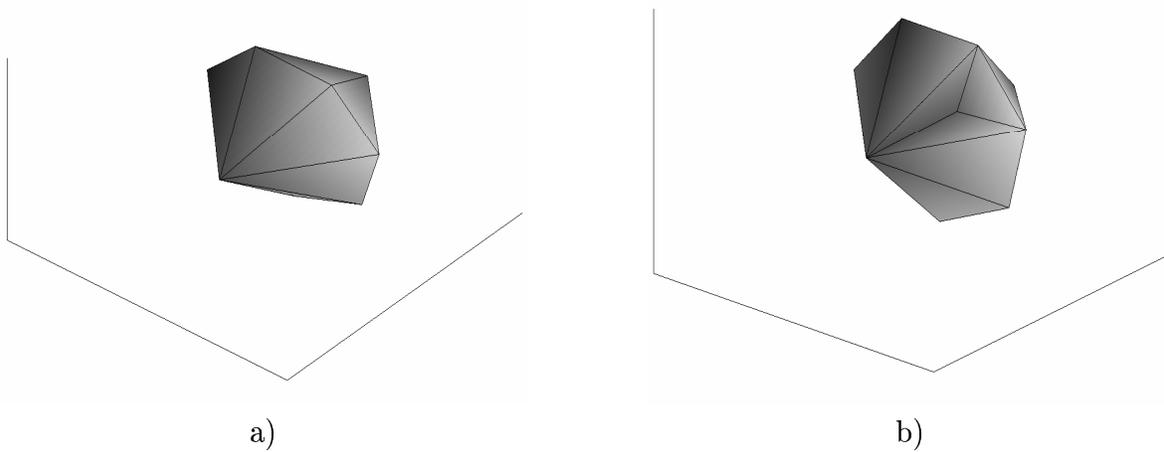


FIG. 3.7 – Insertion de sommets. b) Insertion de sommets au polyèdre a)

Mais, l'utilisation de cette stratégie s'avère inefficace. Ceci est dû à l'approximité du sommet ajouté aux autres sommets du polyèdre qui a pour conséquence de générer les auto-intersections.

2. **La deuxième stratégie** consiste à ajouter un sommet au milieu de la plus grande arête des triangles sélectionnés. Par conséquent, il y aura quatre triangles ajoutés à la liste comme est montré à la figure (3.8), et selon l'algorithme décrit dans le tableau (3.4)

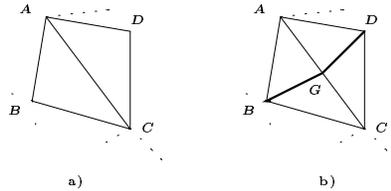


FIG. 3.8 – b) Insertion de sommets au polyèdre a)

```

/*Ajout de sommet*/
Pour  $i = 1, \dots, q$ , /*  $q$  : nombre de triangles sélectionnés*/
  /*Soit le triangle  $(ABC)$ */
  Recherche de l'arête la plus grande du triangle.
  Calcul des coordonnées de son centre  $G$ .
  Ajouter les coordonnées de  $G$  à la liste des sommets  $\mathcal{V}$ .
  Composer les 4 nouveaux triangles :
   $(ABG)$ ,  $(BCG)$ ,  $(AGD)$  et  $(GCD)$ .
  Ajouter ces triangles à la liste des triangles  $\mathcal{F}$ .
Fin Pour.

```

TAB. 3.4 – Méthode d'ajout des sommets et de mise à jour des listes de sommets et de triangles

C'est cette stratégie qui est adapté dans cette étude, elle permet d'éviter que les sommets soient trop proche. De plus, dans l'algorithme, chaque triangle est divisé une seule fois empêchant ainsi le problème de l'approximité des sommets.

### Simplification de la triangulation

Cette étape suit celle de l'ajout de sommets, elle permet de vérifier la topologie de la forme, c'est à dire vérifier si les sommets sont répartis d'une façon uniforme sur la surface du polyèdre. La méthodologie de cette étape est inspiré de la méthode de Lachaud décrite dans le paragraphe (3.3.1). Nous allons essayer d'appliquer les deux premières contraintes définies dans (3.3 et 3.4). Mais, nous allons laisser l'application de la troisième contrainte (3.5) pour un travail ultérieur vu la complexité de sa vérification. Rappelons maintenant les deux contraintes que nous allons appliquer :

Contrainte 1 :

$$\forall (u, v) \in \mathcal{V} \times \mathcal{V} / v \in \nu(u), \delta < \|\mathbf{u} - \mathbf{v}\|$$

Contrainte 2 :

$$\forall (u, v) \in \mathcal{V} \times \mathcal{V} / v \in \nu(u), \|\mathbf{u} - \mathbf{v}\| < 2.5 \delta$$

Dans le cas de violation des ces deux contraintes, un ensemble de transformations classiques de création, inversion et suppression sont appliquées [12]. L'algorithme de cette étape est décrit

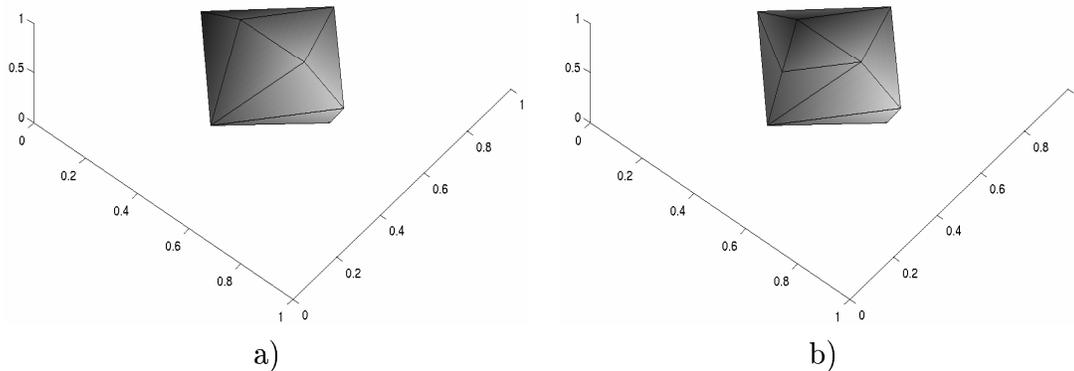


FIG. 3.9 – Insertion de sommets. b) Insertion de sommets au polyèdre a)

dans le tableau (3.5), pour chaque contrainte non satisfaite, une transformation est appliquée. Les algorithmes de ces transformations, fusion, inversion ou création, seront détaillées par la suite.

```

/* Parcours des sommets */
Pour  $i = 1, \dots, n$ , /*  $n$  : nombre de sommets du polyèdre */
/* Parcours des voisins du sommet  $v_i$  */
  Pour  $j = 1, \dots, nb$ , /*  $nb$  : nombre de voisins du sommet  $i$  */
    Calcul des distances  $dist = v_i u$ , /*  $u$  voisin de  $v_i$  */.
    .Si  $dist \leq \delta$  /* Contrainte1 non satisfaite */
      fusionner les sommets  $v_i$  et  $u$  .
    .Si  $2.5 \delta \leq dist$  /* Contrainte2 non satisfaite */
      Inverser ou créer des sommets .
    .Sinon ne rien faire
  Fin Pour j.
Fin Pour i.
Mise à jour de la liste des sommets.
Mise à jour de la liste des triangles.

```

TAB. 3.5 – Algorithme de la simplification de la triangulation

- **Fusion des sommets** : Cette contrainte nous permet de vérifier si deux sommets voisins sont très proches ou non. le non respect de celle ci peut causer un problème lors de l'optimisation en générant des auto-intersections comme est montré à la figure(3.10).

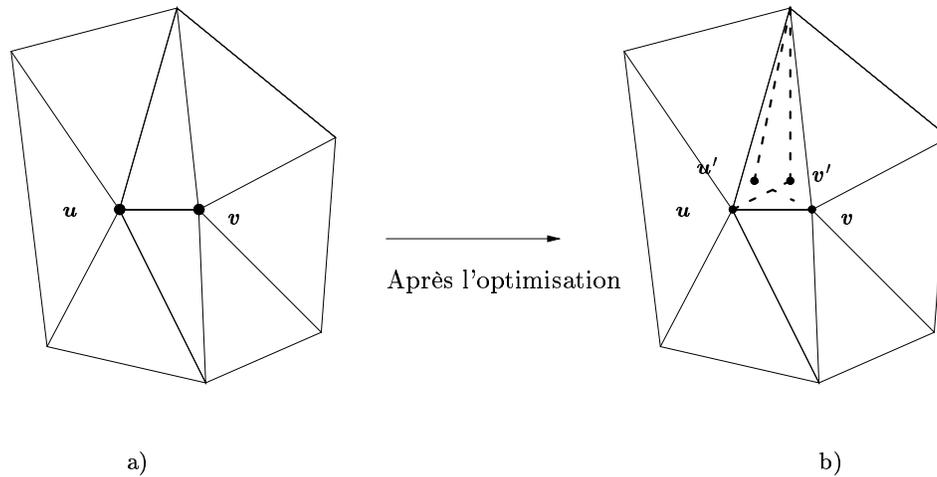


FIG. 3.10 – Problème d'auto-intersection : a) polyèdre avant l'optimisation. b) Déplacement des sommets lors de l'étape de l'optimisation : intersection des arêtes

Pour remédier à ce problème, il faut procéder par une transformation qui consiste à fusionner ces deux voisins qui sont proches en un seul, ce qui a pour conséquence la suppression de quelques triangles. Ce procédé est bien expliqué à la figure (3.11).

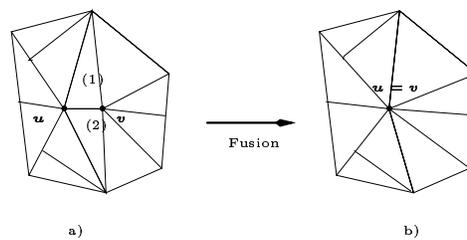


FIG. 3.11 – Fusion des sommets : a) Sommets  $u$  et  $v$  sont très proches. b) Fusion de ces deux sommets et suppression des triangles (1) et (2)

- **Inversion ou création de sommets :** Cette contrainte permet d'éviter d'avoir des triangles trop étirés. Le principe de cette transformation est résumé à la figure (3.12). On procède soit par une inversion des triangles soit par une création d'un nouveau sommet.

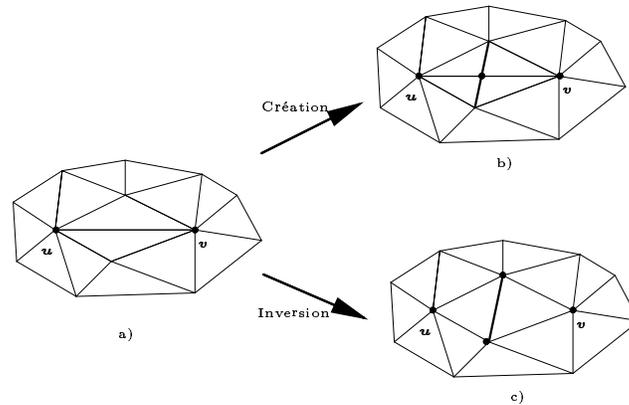


FIG. 3.12 – Inversion et création de sommets : a) Sommets  $u$  et  $v$  sont très éloignés et changement de la topologie de la surface. b) Création d'un sommet supplémentaire. c) Inversion de l'arête  $uv$

Toutes les transformations décrites ci-dessus ont été développées et appliquées sur la triangulation de l'objet  $\mathcal{P}$ . Les résultats de ces algorithmes sont illustrés dans le chapitre (4.6).

## 3.4 Conclusion

Nous avons présenté dans ce chapitre l'approche multirésolution dans le cas 2D et ainsi que dans le cas 3D en montrant la difficulté de l'extension. Nous avons étudié également les problèmes qui se posent à l'achèvement de l'étape d'optimisation. Les algorithmes décrits ont été testés et les résultats montrant les améliorations apportées par l'approche multirésolution, à savoir la qualité de la solution et la charge de calcul, sont montrés dans le dernier chapitre.

# Chapitre 4

## Simulations

### 4.1 Introduction

Ce chapitre est consacré à l'illustration des performances de l'approche multirésolution présentée au chapitre (3.4). Dans un but de comparaison, nous commençons par présenter quelques résultats de l'approche monorésolution. Ensuite, nous présentons ceux de l'approche multirésolution. Avant de procéder à ces simulations, nous décrivons brièvement le jeu de données en 3D sur lequel nous évaluons les résultats.

### 4.2 Nature des données

Les données que nous allons traiter dans ce projet sont simulées sous certaines conditions. Nous considérons des objets inclus dans un cube  $(0, 1)^3$  tels que représentés à la figure (4.1). Neuf projections parallèles et bruitées sont simulées sur le même plan  $z = 1.5$ , suivant les angles décrits, de coordonnées sphériques, dans le tableau (4.1). Ces projections sont calculés selon le procédé décrit au chapitre (1.6).

$i$	1	2	3	4	5	6	7	8	9
$\theta_i$	$\pi/4$	$3\pi/4$	$-\pi/4$	$5\pi/4$	$-\pi/2$	$\pi/2$	0	$\pi$	0
$\varphi_i$	$\varphi$	$\varphi$	$\varphi$	$\varphi$	$\pi/4$	$\pi/4$	$\pi/4$	$\pi/4$	0

TAB. 4.1 – Données 3D. Valeurs des coordonnées sphériques des vecteurs dirigeant les 9 projections ( $\varphi = \arccos(\sqrt{3}/3)$ ). Le dernier est dirigé suivant  $z$ .

Chaque projection est une image de taille  $64 \times 64$  et tous les angles  $\phi_i$  sont telsque  $|\phi_i| < \pi/4$ . Notons aussi que les objets que nous traitons sont des objets non convexes et présentent des irrégularités spatiales. Par conséquent, la modélisation de ces objets nécessite un grand nombre de sommets.

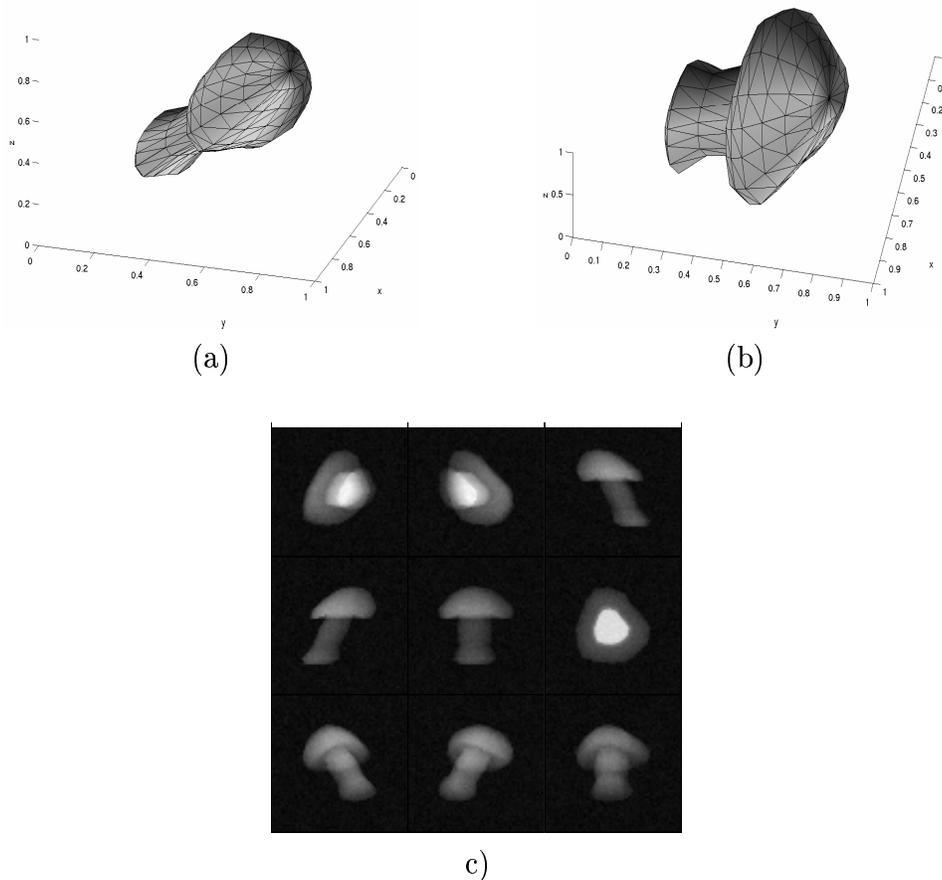


FIG. 4.1 – Données synthétiques 3D. (a) Représentation de l'objet. (b) Représentation de l'objet en vue de dessus. (c) Représentation des neuf projections à un  $RSB = 20dB$ .

### 4.3 Mesure de la qualité de reconstruction

Pour mesurer la qualité de la solution reconstruite, nous nous sommes basés sur deux paramètres :

- La charge de calcul (en temps CPU).
- La distance entre l'objet réel et l'objet reconstruit, ainsi que celle entre leur projections. Calculer cette distance entre les objets revient à discrétiser le cube contenant l'objet en voxels, ensuite comparer ces deux cubes voxel par voxel en calculant la distance selon la règle suivante :

$$D1 = \sum_{i,j,k=1}^{K1} |a_{ijk}^0 - a_{ijk}| / \sum_{i,j,k=1}^{K1} |a_{ijk}^0| \quad (4.1)$$

où  $a_{ijk}^0$  le voxel de coordonnées (i,j,k) du cube contenant l'objet réel.

$a_{ijk}$  le voxel de coordonnées (i,j,k) du cube contenant l'objet reconstruit.

Le calcul de la distance entre les projections est plus facile, il suffit de faire la différence

entre les pixels des images de projections. Donc cette distance s'écrit :

$$D2 = \sum_{i,j=1}^{K1} \sum_{k=1}^m |p_{ijk}^0 - p_{ijk}| / \sum_{i,j=1}^{K1} \sum_{k=1}^m |p_{ijk}^0| \quad (4.2)$$

où  $m$  est le nombre de projections.

$p_{ijk}^0$  le pixel de coordonnées  $(i,j)$  de la  $k^{ime}$  projection de l'objet réel.

$p_{ijk}$  le pixel de coordonnées  $(i,j)$  de la  $k^{ime}$  projection de l'objet reconstruit.

## 4.4 Approche monorésolution en 3D

Nous illustrons ici les résultats de la reconstruction de l'objet de la figure (4.1) en utilisant l'approche monorésolution. Pour cette approche, les résultats dépendent nettement de l'initialisation [22].

On initialise le problème de reconstruction par une ellipse de 59 sommets et 114 triangles (4.2). Les données utilisées sont les neuf projections suivant les directions décrites au tableau (4.1) et avec un  $RSB = 20dB$ . Les résultats sont présentés à la figure 4.3.

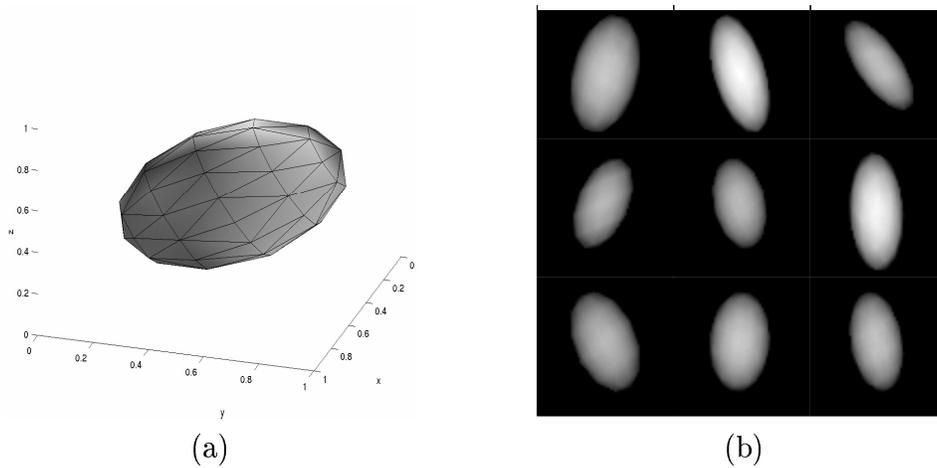


FIG. 4.2 – Initialisation par une ellipsoïde. a) Objet d'initialisation avec 59 sommets et 114 triangles. b) Ses projections.

Nous remarquons que à partir d'un certain nombre d'itérations, le critère  $\mathcal{J}$  ne diminue plus, c'est à dire que la solution est saturée et elle ne peut plus être améliorée pour ce cas là.

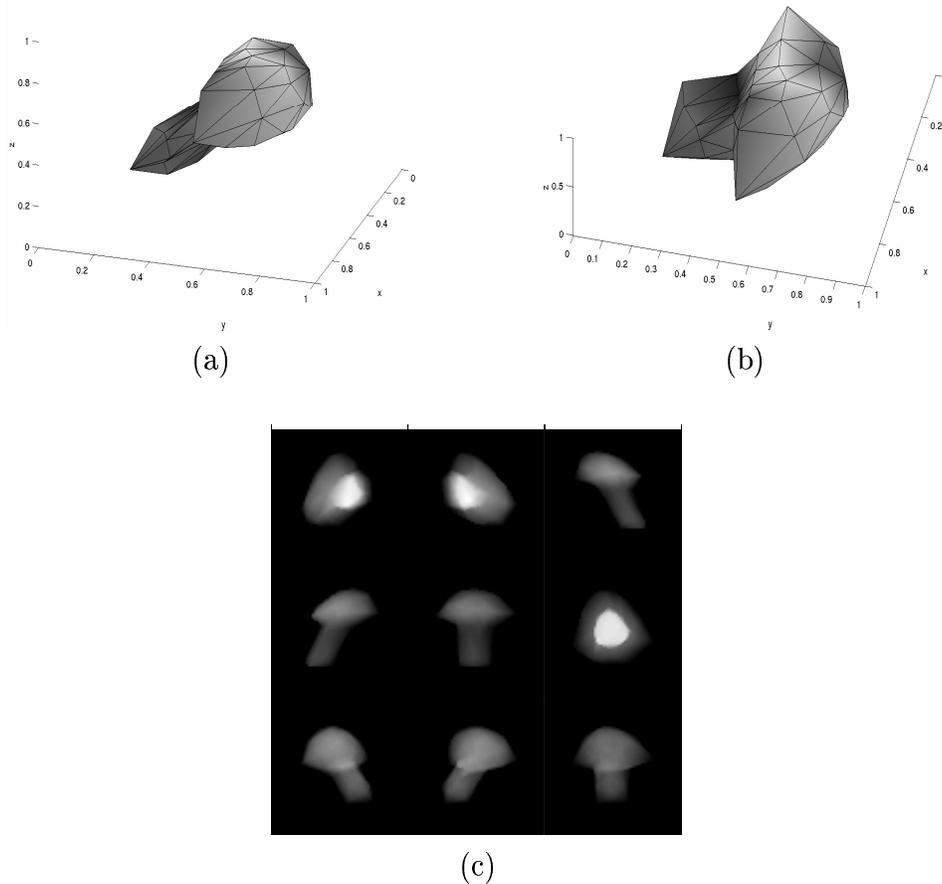


FIG. 4.3 – Reconstruction en monorésolution en 100 itérations, CPU=750,  $D1=0.3$  et  $D2=0.1$ . a) Objet reconstruit. b) Objet vu de dessus. c) Ses projections

## 4.5 Approche multirésolution en 3D

Pour ces simulations, nous utilisons les données simulées et décrites au paragraphe (4.2), c'est à dire neuf projections de l'objet suivant les directions du la tableau (4.1). Pour ce problème, nous initialisons par une ellipsoïde dont les paramètres sont déterminés par la méthode des moments, cette initialisation et ses projections sont présentées à la figure (4.4), les sommets dans ce cas sont réduits et ils sont de l'ordre de 29 sommets et 54 triangles.

L'objet que nous traitons est illustré par la figure (4.1), cet objet contient 226 sommets et 448 triangles. Pour la reconstruction en approche multirésolution, nous avons adopté la technique d'optimisation à savoir la modification globale de l'ensemble des sommets (2.4).

Nous présentons l'évolution de reconstruction de l'objet  $\mathcal{P}$  à travers les niveaux de résolution (4.5), ainsi que leurs projections (??). Pour cet exemple, le résultat est obtenu pour 4 niveaux de résolution avec 20 itérations pour chaque niveau et en initialisant le problème par l'ellipsoïde contenant 29 sommets et 54 triangles (4.4).

Pour contrôler la qualité de cette reconstruction, on se réfère aux deux paramètres définis au paragraphe (4.3) :

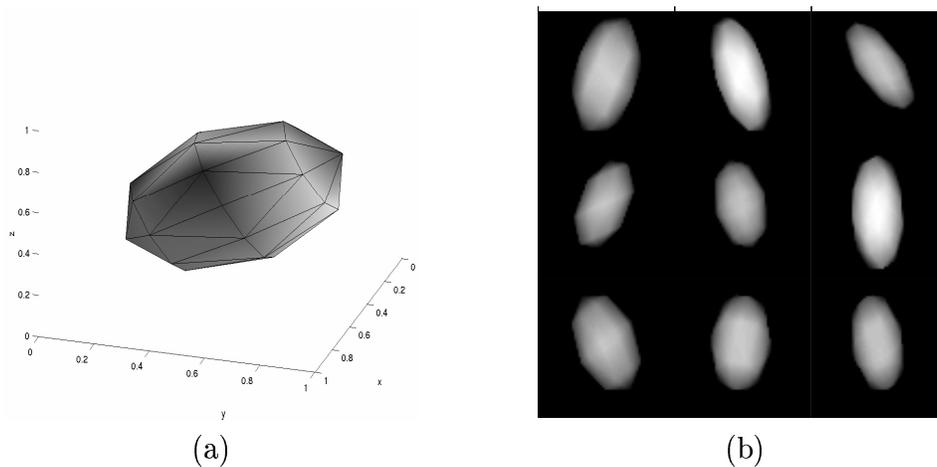


FIG. 4.4 – Initialisation de la reconstruction. a) Ellipsoïde de 29 sommets et de 54 triangles. b) projections de cette ellipsoïde.

- Le cout de calcul de cette opération est calculé en temps CPU, pour les résultats de la figure (4.5),  $temps_{CPU} = 419$ .
- Pour le calcul des distances, on représente le résultat sous forme d'un graphe la distance en fonction des niveaux de résolution (4.7) et (4.8). On remarque bien que la distance soit entre les objets soit entre les projections est une fonction décroissante au fur et à mesure que le niveau de résolution augmente. Au bout de cette simulation, nous arrivons à un pourcentage à peu près de  $D1 = 0.34$  pour les objets et de  $D2 = 0.34$  pour les projections

La solution retrouvée dépend du nombre de niveaux de résolution ainsi que du choix de l'hyperparamètre pour chaque niveau. Nous pensons que cette approche de multirésolution permet de réduire la charge de calcul. Dans le sens où on initialise le problème avec un nombre faible de sommets ce qui permet de gagner du temps dans les premiers niveaux de résolution. Ces objectifs ne sont pas tout à fait atteints, il reste encore des améliorations à introduire pour affiner le résultat.

## 4.6 Conclusion

Ce chapitre a permis d'illustrer sur un jeu de données simulées les résultats de l'approche multirésolution dans le cas 3D. Nous avons présenté en premier lieu, les résultats de l'approche monorésolution dans le but de pouvoir instorer une comparaison avec ceux de l'approche multirésolution permettant ainsi de mettre en relief les avantages qu'apportent cette approche point de vue charge de calcul et qualité de la solution.

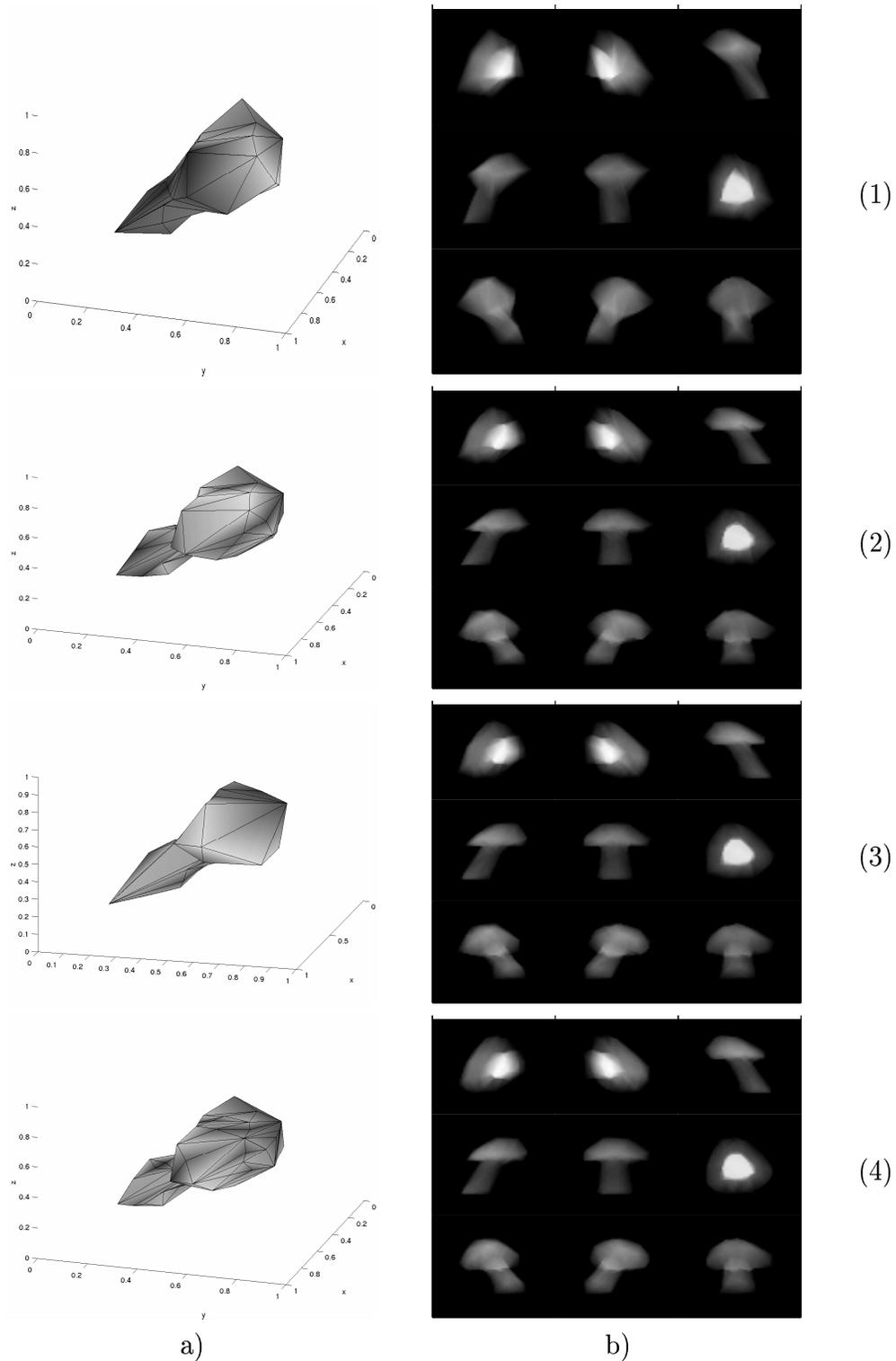


FIG. 4.5 – Résultats de l'approche multirésolution : a) l'évolution des objets à travers les niveaux de résolution. b) leurs projections. (1) Résultat de la 1<sup>ier</sup> niveau de résolution. (2) 2<sup>me</sup> niveau avec 45 sommets et 86 triangles. (3) 3<sup>me</sup> niveau avec 71 sommets et 138 triangles. (4) 4<sup>me</sup> niveau avec 91 sommets et 178 triangles

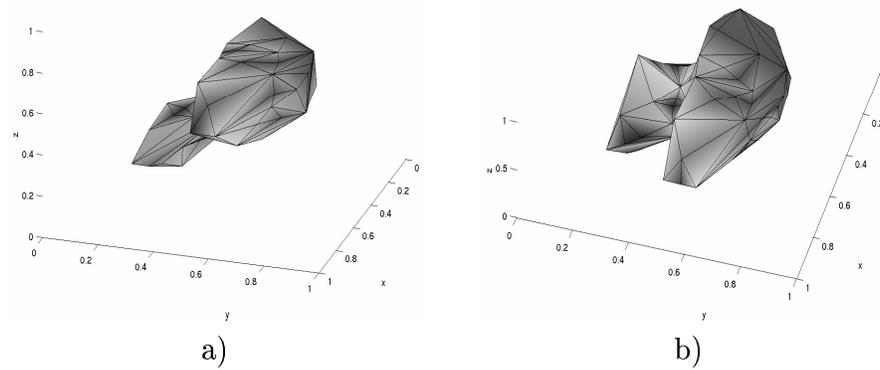


FIG. 4.6 – Résultat final après 4 niveaux de résolution, CPU=419,  $D1=0.14$  et  $D2=0.2$ . a) Objet reconstruit. b) Objet vue de dessus.

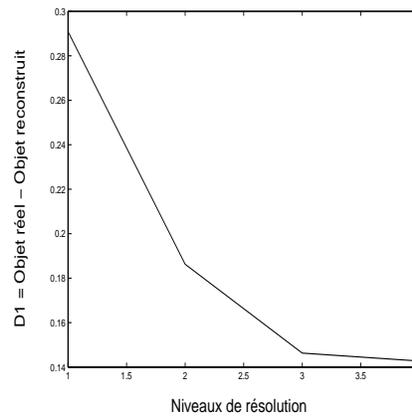


FIG. 4.7 – Distance entre les objets  $D1$  (4.1)

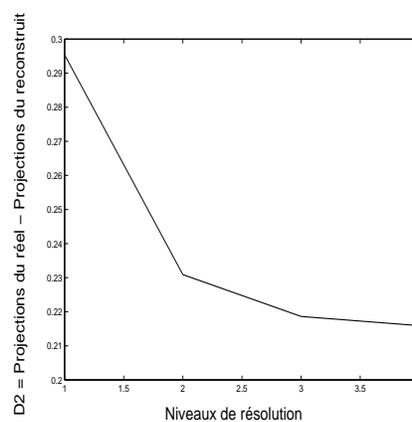


FIG. 4.8 – Distance entre les projections  $D2$  (4.2)

# Conclusion

Le but de ce travail a été d'appliquer les techniques qui ont été développées pour résoudre le problème de reconstruction tomographique en monorésolution [?], pour arriver à concevoir une extension en multirésolution. Dans un premier temps, nous avons présenté les différentes étapes de la reconstruction en monorésolution à savoir la modélisation polyédrique de l'objet  $\mathcal{P}$ , les techniques d'optimisation adoptées : minimisation du critère  $\mathcal{J}(\mathbf{V})$  où  $\mathbf{V}$  sont les sommets de  $\mathcal{P}$ . Nous avons décrit également les outils nécessaires pour le calcul de  $\mathcal{J}(\mathbf{V})$  et son gradient  $\nabla\mathcal{J}(\mathbf{V})$  : le calcul direct des projections de  $\mathcal{P}$  ainsi que leur mise à jour.

Notre contribution porte sur l'amélioration de l'approche monorésolution en introduisant la multirésolution qui consiste à reconstruire successivement les polyèdres à nombre croissant de sommets, ainsi que les modules nécessaires au passage d'un niveau à un autre. La mise en oeuvre de cette approche s'avère un peu difficile. En effet, rendre la topologie du polyèdre variante à chaque niveau de résolution rend l'étape d'optimisation plus compliquée à gérer. Nous avons réalisé durant ce stage une première tentative pour améliorer la solution. Cependant, il reste encore des améliorations à introduire pour avoir un meilleur résultat de la reconstruction.

Les perspectives que nous pouvons dégager à ce sujet sont :

- Introduire un nouveau modèle d'initialisation qui soit plus proche de l'enveloppe convexe de l'objet à reconstruire.
- Utiliser le principe de la triangulation de Delaunay pour refaire la triangulation à chaque niveau de résolution pour éviter le problème des auto-intersections.
- Estimer les hyperparamètres  $\lambda$  à chaque niveau de résolution.

# Bibliographie

- [1] J. Arvo. *Graphics Gems II*. Academic Press, 1991.
- [2] C.B. Barber, D.P. Dobkin, and H. Huhdanpaa. The quikhull algorithm for convex hull. *ACM Transactions on Mathematical Software*, 22(4) :469–483, december 1996.
- [3] X.L. Battle, K.M. Hanson, and G.S. Cunningham. Tomographic reconstruction using 3d deformable models. *Phy. Med. Biol.*, 43 :483–490, 1998.
- [4] A.J. Bulpitt and N.D. Efford. An efficient 3d deformable model with a self-optimising mesh. *Image and Vision Computing*, 14 :573–580, 1996.
- [5] O. Dellivers, S. Meiser, and M. Teillaud. Fully dynamic delaunay triangulation in logarithmic expected time per operation. *Compt. geom. Theory. Appl*, 2(2) :55–80, 1992.
- [6] J.M. Dinten. *Tomographie à partir d'un nombre limité de projections : régularisation par des champs markoviens*. PhD thesis, Université de Paris XI, Orsay, 1990.
- [7] C.R. Dyer. Volumetric scene reconstruction from multiple views. In *Foundations of Image Understanding*, chapter 16, pages 469–489. Boston, 2001.
- [8] J.A. Fessler and A. Macovski. Object based 3-d reconstruction of arterial trees from magnetic resonance angiograms. *IEEE. Trans. Medical Imaging*, 10(1) :25–39, march 1991.
- [9] K.M. Hanson and G.S. Cunningham. Exploring the reliability of bayesian reconstructions. In *Medical Imaging : Image processing, Proc. SPIE*, pages 416–423, 1995.
- [10] K.M. Hanson, G.S. Cunningham, G.R. Jennings, and D.R. Wolf. Tomographic reconstruction based on flexible geometric models. In *Proc. IEEE Int. Conf. on Image Processing*, pages 145–147, 1994.
- [11] K.M. Hanson, G.S. Cunningham, and R.J. McKee. Uncertainty assessment for reconstruction based on deformable geometry. *Int. J. Imaging Syst. Technol.*, 8 :506–512, 1997.
- [12] J.O. Lachaud and A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine 3d surface extraction. *Medical Imaging Analysis*, 3(2) :187–207, 1999.
- [13] T. Lambert. An optimal algorithm for realizing a delaunay triangulation. *Informations Processing Letters*, 62(5) :245–250, 1997.
- [14] J. Liebeherr and M. Nahas. Application-layer multicast with delaunay triangulations. Technical report, University of Virginia, department of Computer Science, 2001.
- [15] P. Milanfar. Reconstructing binary polygonal objects from projections : a statistical view. *Graphical Models and Image processing*, 56(5) :371–391, september 1994.
- [16] P. Milanfar, W. C. Karl, and A.S. Willsky. A moment-based variational approach to tomographic reconstruction. *IEEE. Trans. Image Processing*, 5(3) :459–470, february 1996.
- [17] P. Milanfar, G. C. Verghese, W.C. Karl, and A.S. Willsky. Reconstructing polygons from moments with connections to array processing. *IEEE. Trans. Signal Processing*, 43 :432–443, february 1995.

- [18] A. Mohammad-Djafari. Binary polygonal shape image reconstruction from a small number of projections. *Elektrik*, 5(1) :127–138, 1997.
- [19] A. Mohammad-Djafari and J.M. Dinten. *Approche bayésienne pour les problèmes inverses*, chapter Reconstruction tomographique pour un faible nombre de projections. Lavoisier, 2001.
- [20] D.J. Rossi and A.S. Willsky. reconstruction from projections based on detection and estimation of objects. *IEEE. Trans. Acoustics, Speech and Signal Processing*, 32(4) :886–906, august 1984.
- [21] K. Sauer, J. Sachs, and K. Klifa. Bayesian estimation of 3-d objects from few radiographs. *IEEE. Trans. Nuclear Science*, 41(5) :1780–1790, october 1994.
- [22] C. Soussen. *Reconstruction 3D d'un objet compact en tomographie*. PhD thesis, Université PARIS XI, 2000.
- [23] C. Soussen and A. Mohammad-djafari. Multiresolution approach to the estimation of the shape of a 3d compact object from its radiographic data. In *Proc. SPIE, Mathematical Modeling, Bayesian Estimation, and Inverse Problems*, pages 150–160, july 1999.
- [24] C. Soussen, A. Mohammad-Djafari, and K. Sauer. Recovery of moments of a compact homogeneous object from tomographic projection data. *soumis à Graphical Models and Image processing*, 1999.

## Résumé :

Nous considérons un problème rencontré dans les applications du contrôle non destructif qui est la reconstruction tomographique en 3D d'un objet compact à partir d'un nombre faible de projections. Nous modélisons la surface de cet objet par un polyèdre  $\mathcal{P}$  dont les sommets seront estimés à partir des données. Cette estimation est faite dans le cadre d'une approche bayésienne : estimation au sens du maximum *a posteriori* avec une modélisation *a priori* markovienne des positions des sommets du polyèdre. Pour réduire le cout de l'optimisation du critère, une approche multirésolution est proposée dans laquelle le nombre de sommets de  $\mathcal{P}$  augmente progressivement à travers les niveaux de résolution. L'algorithme est initialisé par un polyèdre convexe avec un nombre faible de sommets. Pour chaque niveau, le critère est optimisé en utilisant la solution du niveau précédent comme initialisation.

Mots clés : Reconstruction tomographique en 3D, forme polyédrique, approche multirésolution.

## Abstract :

We consider the tomographic reconstruction of 3D compact object from a few number of projections, a problem encountered in non destructive testing applications. We model the shape of the object by a deformable polyhedron whose vertices are estimated directly from data. This estimation is done through a bayesian approach : maximum *a posteriori* (MAP) estimation with a markovian *a priori* model for the position of the vertices of the polyhedron. To reduce the computational cost, a multiresolution approach is proposed, in which the number of vertices increases as the resolution becomes finer and finer. The problem of reconstruction is initialized by a convex polyhedral with a few number of vertices. In each level of resolution, the criteria is optimized using the solution of the previous level as initialisation.

Keywords : Tomographic reconstruction, deformable polyhedron, multiresolution approach.