

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE SAAD DAHLEB DE BLIDA
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE



MEMOIRE DE FIN D'ETUDES

Pour l'obtention

D'un diplôme de master en informatique.

Option : Génie des systèmes informatiques

THÈME :

*Conception et développement d'une application distribuée
pour la reconstruction des modèles 3D à partir d'un nuage de
points quelconque*

Réalisé par :

Sissani Djihad

Zouaoui Ouissam

Soutenu le : 26 juin 2016, devant :

Mme Rezzoug Nachida

Promotrice

Mr Bey Mohamed

Encadreur

Mme Tchantchane Zahida

Co-Encadreur

Mr. Bala Mohamed

Président

Mr. Farfera Sofiane

Examineur

2015/2016

MA-004-314-1

Remerciement

En terminant notre mémoire de fin d'étude nous remercions Allah le tout puissant de nous avoir donné le courage et la patience pour achever ce travail.

Nous remercions nos encadreurs,

Au centre de développement des technologies avancées :

Mr Bey Mohamed et Mme Tchantchane zahida

A l'université Notre promotrice, Mme Rezzoug Nachida

Nous tenons à remercier toute l'équipe de CDTA en particulier Mr Bendifallah hassen, Mme Bouhadja Khadidja et Mme Sahla.

Nous remercions tous nos enseignants qui nous ont fourni les outils nécessaires pour terminer ce projet,

en particulier Mme Zahra.

A la fin il nous est agréable d'adresser nos vifs remerciements à tous nos collègues d'étude, particulièrement ceux de notre promotion et tous qui nous ont aidés de près ou de loin pour élaborer cet ouvrage.

Dédicace

A la mémoire de mon Père Zouaoui Ali,

A ma très chère maman, Houria

A mon frère Adlane et ma sœur Nesrine

A ma très chère copine Mayssa

Ma copine et mon binôme Djihad

Mr Mustapha Houli

A toute la famille Zouaoui en particulier ma chère tante Fafi

A toute la famille Nafa en particulier mon très chère grand père Said et ma très chère grand mère ouardia

A Mr Mokrani Nordine

A Mr Brakni Smail

A toute la famille tafat

Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement et le respect que j'ai toujours eu pour vous.

Rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être. Ce travail est le fruit de vos sacrifices .je vous aime

Zouaoui Ouissam

Dédicace

Je dédie ce mémoire à :

Mes parents :

Ma mère Leïla, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon père Hamid, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Allah faire en sorte que ce travail porte son fruit ; Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

Mes frères Mayssara, El yamene et mon chouchou Nidhal

Mon amie d'enfance Nawal

Ma chère sœur Ouissam

Ma jolie Mayssa

Mes deux familles Sissani et Hasbellaoui, en particulier ma grande mère Fettouma

A une personne très chère pour moi Cherbal Nasreddine, Je te dédie ce travail avec tous mes vœux de bonheur, de santé et de réussite. Sans ton aide, tes conseils et tes encouragements ce travail n'aurait vu le jour.

Sissani Djihad

ملخص

نهتم في هذه المذكرة بتطوير مجال تصنيع القطع ذات الاسطح المعقدة التي يتم معالجتها على مستوى وحدة التصنيع و الروبوتيك التابعة لمركز تنمية التكنولوجيات المتطورة .
الهدف من هذا العمل هو اقتراح وتنفيذ طريقتين الاولى تهدف إلى إعادة تصنيع النماذج ثلاثية الأبعاد انطلاقاً من سحابة نقط والثانية تهدف إلى الاقتصاد في زمن إعادة التصنيع والتشكيل ولهذا الغرض.
ولتحقيق هذا قمنا بتطبيق عدة طرق تقوم أساساً على ثلاثية دو لوناى و خوارزميات التي قمنا بتحويلها حتى تنسجم في الفضاء ثلاثي الأبعاد
الحل المقترح موافق عليه والنتائج مرضية
الكلمات المفتاحية: الأسطح المعقدة، النماذج ثلاثية الأبعاد، إعادة التصنيع، ثلاثية دو لوناى نموذج ستل

Résumé

Ce travail s'intègre dans le cadre de développement d'un environnement de fabrication de surfaces gauches initiée au niveau de la division de productique et robotique du centre de développement des technologies avancées

L'objectif de ce travail consiste à proposer et à implémenter deux approches, une permet la reconstruction des modèles 3D à partir d'un nuage de points quelconque, et l'autre vise à optimiser le temps de cette reconstruction.

Pour se faire nous avons utilisé plusieurs approches basées sur la triangulation de Delaunay 3D et des algorithmes que nous avons translaté vers la 3D, qui permettent d'arriver à un modèle STL tout en optimisant le temps.

La solution proposée a été validée, et les résultats sont satisfaisantes.

Mots clés : Surfaces gauches, modèles 3D, reconstruction, triangulation Delaunay, un modèle STL,

Abstract

This work fits into the context of development of curved surfaces manufacturing environment, initiated at the level of the division of productics and robotics of the Advanced Technology Development Center. The objective of this work is to propose and implement two approaches, one allows the reconstruction of 3D models from a cloud of points, and the other is to optimize the time of reconstruction. To do so we have used several approaches based on 3D Delaunay triangulation and algorithms that we have translated to 3D. allowing us to reach a STL model while optimizing time. The proposed solution has been approved, and the results are satisfying

Key words: curved surfaces, 3D models, reconstruction, 3D Delaunay triangulation, STL model.

III.1	Introduction.....	18
III.2	Les méthodes de reconstruction.....	18
III.2.1	Techniques de reconstruction par approximation.....	18
III.2.2	Techniques de reconstruction par interpolation.....	18
III.3	Triangulation de Delaunay.....	19
III.3.1	Les dérivés de Delaunay.....	19
III.3.2	Propriétés de la triangulation de Delaunay.....	19
III.3.3	Méthodes de calculs de la triangulation de Delaunay.....	21
IV.	Format d'échange de données.....	22
IV.1	Introduction.....	22
IV.2	Modèle IGES.....	22
IV.2.1	Avantages et inconvénients.....	23
IV.3	Modèle STEP.....	23
IV.3.1	Avantages et inconvénients.....	23
IV.4	Modèle STL.....	24
IV.4.1	Avantages et inconvénients.....	25
IV.5	Conclusion :.....	26

Chapitre 02 : SOLUTIONS PROPOSEES

I.	Introduction.....	28
II.	Problématique.....	28
III.	Objectifs.....	30
IV.	Architecture logicielle pour la reconstruction des surfaces complexes.....	31
IV.1	Description de l'organigramme global de la reconstruction.....	32
IV.2	Les tâches de la reconstruction.....	32
IV.2.1	Lecture du nuage de point.....	32
IV.2.2	Subdivision/distribution du nuage de points :.....	33
IV.3	Approximation des surfaces des sous nuages :.....	35
IV.3.1	Création du tétraèdre englobant.....	37
IV.3.2	Insertion des points.....	38

Sommaire

Introduction générale	2
Chapitre 01 : ETUDE BIBLIOGRAPHIQUE	
I. Processus de conception d'objets de formes complexes.....	6
I.1 Introduction	6
I.2 La modélisation 3D des objets complexes	6
I.2.1 Modélisation Filaire	7
I.2.2 Modélisation surfacique.....	7
I.2.3 Modélisation volumique	8
I.3 Construction d'un modèle surfacique.....	8
I.4 Méthodes de modélisation.....	8
I.4.1 Méthodes basées sur l'utilisation des points	9
I.5 Avantages et inconvénients	11
II. Ingénierie inverse.....	12
II.1 Introduction	12
II.2 Définition.....	12
II.3 Raisons d'utilisation de l'ingénierie inverse	12
II.4 Domaines d'application.....	12
II.5 Processus de l'ingénierie inverse.....	13
II.5.1 Numérisation de l'objet.....	14
II.5.2 Recalage	15
II.5.3 Décimation.....	15
II.5.4 Segmentation.....	16
II.5.5 Approximation des surfaces	16
II.6 Avantages de l'ingénierie inverse	17
III. Étude des méthodes de reconstruction des objets 3D	18

IV.3.3	Triangulation de Delaunay.....	38
IV.3.4	Génération du modèle STL.....	40
V.	Conclusion :	41

Chapitre 03 : IMPLEMENTATION ET TEST -VALIDATION

I.	Introduction.....	43
II.	Implémentation	43
II.1	Langage de programmation.....	43
II.2	Environnement de programmation Builder c++.....	44
II.3	OpenGL.....	44
II.4	Présentation des fenêtres	44
II.4.1	Fenêtre principale	44
II.4.2	Fenêtre de «reconstruction 3D».....	45
1.	Implémentation des tâches proposées	45
1.1	Lecture du fichier nuage de point.....	46
1.2	Subdivision du nuage de point	47
1.2.1	Subdivision en deux sous nuages.....	47
1.2.2	Calcul dimension.....	47
1.3	Approximation des surfaces des objets complexes	48
1.3.1	Création du tétraèdre englobant.....	48
1.3.2	Insertion de point.....	50
1.3.3	Triangulation Delaunay finale.....	53
1.4	génération du modèle STL.....	54
III.	Test des résultats	55
III.1	Exemple de test.....	55
1.	Premier nuage de point.....	55
2.	Deuxième nuage de point.....	61
3.	Troisième nuage de point	62
III.2	Etude critique.....	63
IV.	Validation de l'application logicielle.....	65
IV.1.1	Triangulation en deux et trois dimensions.....	65
IV.1.2	Modes de subdivisions.....	66
IV.1.3	Modes de triangulation	66
IV.1.4	Modes d'insertion	67

Liste des Figures

Figure 1.1: Différentes manières de modélisation.....	6
Figure 1.2: facette sous forme de triangle.....	7
Figure 1.3: Différence entre une surface simple et une surface gauche	8
Figure 1.4: Méthode de modélisation de surface par interpolation.....	9
Figure 1.6 : localisation d'un point sur une paramétrique.....	10
Figure 1.7 : Relation entre les différentes représentations.....	10
Figure 1.8 : Schéma générale de la modélisation 3D.....	11
Figure 1.9 : Étapes du processus de l'ingénierie inverse.....	13
Figure 1.10 : numérisation d'un objet.....	14
Figure 1.11 : recalage des nuages de points	15
Figure 1.12 : exemple d'un objet avant et après décimation.....	16
Figure 1.13 : Segmentation d'un lapin.....	16
Figure 1.14 : Exemple d'une approximation par des triangles.....	17
Figure 1.16 : Triangulations possibles d'un quadrilatere	20
Figure 1.17 : Cercle circonscrit.....	20
Figure 1.18 : les Etapes de l'algorithme destruction et instruction	21
Figure 1.19 : structure d'un fichier STL ASCII.....	24
Figure 1.20: Représentation STL d'un objet.....	25
Figure 2.1: Processus de reconstruction des surfaces en deux dimensions	29
Figure 2.2: Processus proposé pour la reconstruction des surfaces en trois dimensions.....	30
Figure 2.3: Organigramme global de la reconstruction des surfaces en trois dimensions.....	31
Figure 2.4: Organigramme de lecture du nuage de points.....	33
Figure 2.5: Subdivision du nuage de points.....	34
Figure 2.6: Architecture de la distribution des sous nuages sur plusieurs corps	35
Figure 2.7: Organigramme de l'approximation par des tétraèdres	36
Figure 2.8: Organigramme du tétraèdre englobant	37
Figure 2.9: Organigramme de la triangulation Delaunay	39
Figure 2.10: Organigramme de la génération du modèle STL.....	40

V. Conclusion	67
Conclusion générale.....	69
Références Bibliographiques	70

<i>Figure3.1: Fenêtre principale.....</i>	<i>44</i>
<i>Figure3.2: fenetre de reconstruction 3D.....</i>	<i>45</i>
<i>Figure 3.3 : Bloc subdivision en deux.....</i>	<i>48</i>
<i>Figure 3.4: bloc du tétraèdre englobant.....</i>	<i>49</i>
<i>Figure 3.5 : bloc de modes de triangulation.....</i>	<i>50</i>
<i>Figure 3.6 : Bouton de triangulation de Delaunay.....</i>	<i>51</i>
<i>Figure 3.7 : Bloc insertion des points.....</i>	<i>51</i>
<i>Figure 3.8: Bouton de vérification et modification.....</i>	<i>52</i>
<i>Figure 3.9 : bouton de triangulation Delaunay.....</i>	<i>53</i>
<i>Figure 3.10 : bouton de génération du modèle STL.....</i>	<i>54</i>
<i>Figure 3.11 : Lecture du nuage de point.....</i>	<i>56</i>
<i>Figure 3.12 : Lecture du nuage de point.....</i>	<i>56</i>
<i>Figure 3.13 : Calcul des dimensions du brut.....</i>	<i>57</i>
<i>Figure 3.14 : Calcul des dimensions du brut.....</i>	<i>57</i>
<i>Figure 3.15 : Brut rendu.....</i>	<i>58</i>
<i>Figure 3.16: Points du tétraèdre englobant.....</i>	<i>58</i>
<i>Figure 3.17 : Sphère inscrite du tétraèdre englobant.....</i>	<i>58</i>
<i>Figure 3.18 : Triangulation de Delaunay.....</i>	<i>59</i>
<i>Figure 3.19 : triangulation Delaunay finale.....</i>	<i>59</i>
<i>Figure 3.20 : modèle STL.....</i>	<i>59</i>
<i>Figure 3.21 : Nuage de point subdivisé en deux.....</i>	<i>60</i>
<i>Figure 3.22 : Triangulation Delaunay des deux sous nuages.....</i>	<i>60</i>
<i>Figure 3.23 : le modèle STL de chaque sous nuage.....</i>	<i>61</i>
<i>Figure 3.24 : processus de génération du modèle STL à partir d'un nuage de points.....</i>	<i>62</i>
<i>Figure 3.25 : processus de génération du modèle STL à partir d'un nuage de points.....</i>	<i>62</i>
<i>Figure 3.26 : comparaison entre les différents modes de subdivisions.....</i>	<i>62</i>

Liste des tableaux

<i>Tableau 1 : Résultat de test avant et après la subdivision.....</i>	<i>63</i>
<i>Tableau 2 : Résultat de test avant et après la subdivision en deux.....</i>	<i>64</i>
<i>Tableau 3 : Résultat de test avant et après la subdivision en huit.....</i>	<i>64</i>
<i>Tableau 4 : Comparaison entre la triangulation de deux et trois dimensions</i>	<i>65</i>
<i>Tableau 5 : mode de subdivision</i>	<i>66</i>
<i>Tableau 6 : Mode de triangulation</i>	<i>66</i>
<i>Tableau 7 : Mode d'insertion.....</i>	<i>67</i>

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Contexte

Depuis des années, les progrès des technologies d'acquisition 2D ont favorisé l'essor de la modélisation géométrique d'objets à deux dimensions à partir de données issues de leur numérisation. Malheureusement cette modélisation est devenue obsolète car son espace de représentation est défini seulement par deux informations ; la longueur et la largeur, ce qui induit à une représentation insuffisante de l'objet.

Avec l'apparition de la 3D, une nouvelle vision dans l'espace nous a donné la possibilité de visualiser le modèle sous toutes les perspectives, ce qui permet une meilleure représentation et approximation de l'objet surtout quand il s'agit des objets très complexes.

La capacité de créer, de visualiser et de manipuler des représentations numériques de la forme et de l'apparence d'objets physiques jouent un rôle important dans divers domaines d'applications tels que l'aéronautique, le design industriel, . . . etc. En raison de leurs complexités géométriques, ces objets sont conçus selon deux approches :

L'une dite classique, basée sur la modélisation de l'objet selon un cahier des charges avec l'utilisation des fonctions de conception et de manipulation offertes par les logiciels de conception assistée par ordinateur - CAO -. L'inconvénient de cette approche réside dans la difficulté d'obtenir les formes désirées sans recourir à des phases de modifications et de validations successives, afin de respecter les contraintes géométriques (tangentes, courbures, etc.) et fonctionnelles imposées, ce qui induit à des temps de traitement très importants.

L'autre approche dite ingénierie inverse est utilisée dans le cas où les formes des objets sont très complexes et ne peuvent être conçues dans un logiciel de CAO, ou si le modèle de description de l'objet n'est pas disponible, elle est basée sur la modélisation de l'objet à partir d'un nuage de points, Cette approche est devenue de nos jours une pratique très courante dans le monde industriel puisqu'elle permet de réduire les temps de développement de nouveaux produits à partir de prototypes.

Problématique

La reconstruction des modèles en trois dimensions avec une bonne précision est une tâche ardue, coûteuse en temps, dépendant des possibilités de modélisation disponibles dans les systèmes CAO et exigeant une grande interactivité entre le logiciel et l'utilisateur. En outre, le modèle de l'objet qui en résulte est une approximation du nuage de points et nécessite très

INTRODUCTION GENERALE

souvent des ajustements. Pour résoudre ce problème, le nuage de points est approximé par des éléments géométriques simples tels que les triangles, les tétraèdres, . . . etc.

Les nuages de points sont très volumineux et nécessitent parfois des calculs intensifs ce qui augmente les temps de traitement et par conséquent les coûts pour la génération d'une triangulation surtout si un seul processeur est utilisé [1].

Objectifs et besoins du travail

Ce travail s'insère dans le cadre de développement d'une application logicielle distribuée graphique et interactive sous Windows automatisant la tâche de la production des surfaces de formes complexes initié par l'équipe Conception et Fabrication Assistées par Ordinateur (CFAO) de la Division Productique et Robotique (DPR) du Centre de Développement des Technologies Avancées (CDTA).

Dans ce projet, nous nous intéressons en premier lieu à la génération d'une triangulation 3D, plus précisément au (modèle STL) approximant la peau extérieure d'objets de n'importe quelles formes à partir de nuages de points quelconques, puis la proposition d'une méthodologie permettant la réduction des temps de traitement en parallélisant les calculs sur plusieurs processeurs ou plusieurs corps. Afin de remédier aux problèmes cités ci-dessus, nous avons assigné à notre étude les objectifs suivants :

- Remédier au problème de la modélisation en deux dimensions pour les objets complexes
- Recherche des approches qui peuvent être réalisables et efficaces pour une modélisation en trois dimensions.
- Génération d'une triangulation 3D (modèle STL) approximant la peau extérieure d'objets de n'importe quelles formes à partir de nuages de points quelconques.
- Proposition d'une méthodologie permettant la réduction des temps de traitement.
- Parallélisme des calculs sur plusieurs processeurs.

Structure du mémoire

Le présent mémoire est composé des chapitres suivants :

- **Chapitre 1 : Etude bibliographique sur toutes les aspects théoriques permettant :**
 - Etude du processus de conception d'objets avec des surfaces de formes complexes.
 - Étude du processus de Rétro-conception
 - Etude des méthodes de reconstruction des modèles des objets à partir d'un nuage de points quelconque.

INTRODUCTION GENERALE

- Etude du format d'échange de données « STL ».
- **Chapitre 2 : solutions proposées**
 - Proposition d'une approche pour la reconstruction des modèles des objets digitalisés qui est une suite de techniques et d'algorithmes
 - Proposition d'une approche pour l'optimisation du temps de reconstruction des objets de formes complexes.
- **Chapitre 3 : Implémentation et Test et validation sur des objets réels.**
 - Développement des algorithmes correspondant
 - Visualisation du résultat obtenu de la modélisation (modèle STL)
 - Distribution de l'application sur plusieurs processeurs (application distribuée).

CHAPITRE I
ETUDE
BIBLIOGRAPHIQUE

I. Processus de conception d'objets de formes complexes

I.1 Introduction

Depuis quelques années, les modèles géométriques tridimensionnels sont de plus en plus omniprésents. Ils sont apparus avec le développement de l'infographie et de la vision tridimensionnelle, qui ont permis une modélisation 3D d'objets de forme complexes. Ces modèles peuvent être créés par des systèmes de modélisation (logiciels de Conception Assistée par Ordinateur (CAO), ...), ou obtenus grâce à des procédés de numérisation tridimensionnelle d'objets physiques. Ils sont généralement représentés sous forme de maillages surfaciques (surface linéaire par morceaux composée de polygones) ou volumiques (représentation de l'objet par un ensemble de polyèdres). Ces modèles peuvent aussi être décrits par des nuages de points, des formes à pôles (courbes de Bézier, NURBS, ...) ou des surfaces implicites [2].

I.2 La modélisation 3D des objets complexes

La modélisation est la représentation des formes et des dimensions des objets. Elle peut être filaire, surfacique ou volumique (Figure 1.1). Dans ce qui suit nous allons détailler la modélisation surfacique des objets de notre étude.

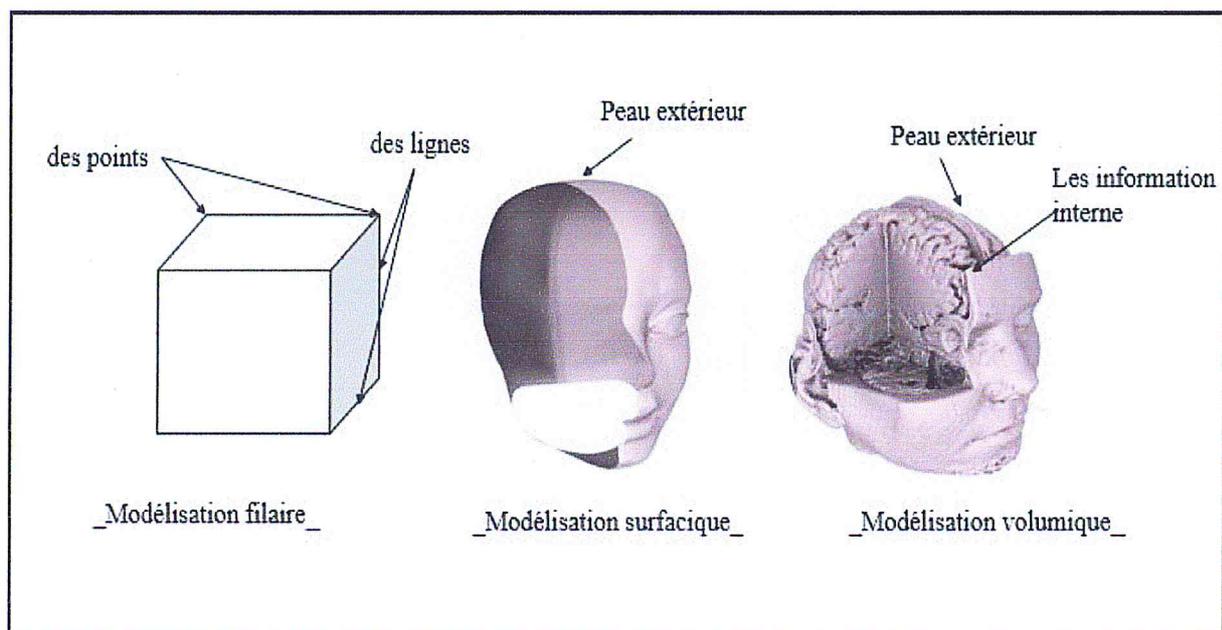


Figure 1.1: Différentes manières de modélisation

I.2.1 Modélisation Filaire

Elle est basée sur des points et des segments, l'objet est décrit par ses sommets (points) et ses arrêtes (segment reliant ces sommets) [3].

I.2.2 Modélisation surfacique

Cette modélisation est le premier outil du concepteur de forme, car elle répond à de nombreux besoins de l'industrie aéronautiques et automobiles. La description de ce modèle est basée sur des concepts mathématiques d'approximation et prend en compte la notion de surfaces simples et surfaces très complexes.

L'objectif de cette modélisation est la définition de l'enveloppe et des surfaces frontières de l'objet. Il existe deux types de modélisation surfacique [4] :

- a) **Surfacique à facette plane** : C'est une forme géométrique simple inscrite ou conçue dans un plan, souvent le triangle est considéré comme la facette la plus simple (Figure 1.2).

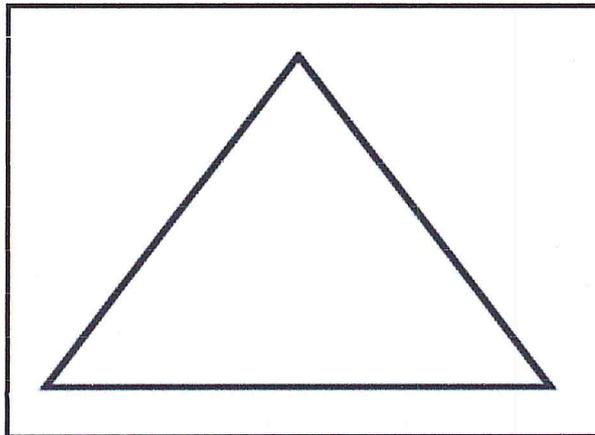


Figure1.2: facette sous forme de triangle

- b) **Surfacique gauche** : cette méthode est employée quand la surface à définir est trop complexe pour être définie par des surfaces simples (planes, cylindriques, sphériques, coniques.).

Une surface est dite gauche dans les cas suivants [5] :

- une surface qui ne peut pas être définie avec une équation mathématique exacte.
- Une surface qui est obtenue par raccordement de plusieurs morceaux (Figure1.3).

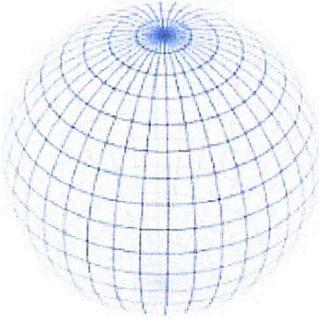
surface simple	Surface gauche
Définit par une équation mathématique? oui	Définit par une équation mathématique? <u>non</u>
	 Plusieurs morceaux

Figure 1.3: Différence entre une surface simple et une surface gauche

I.2.3 Modélisation volumique :

C'est la modélisation la plus complète, elle englobe la modélisation filaire et surfacique (arêtes et surfaces). Elle permet la représentation dans l'espace, avec la notion de matière [5].

I.3 Construction d'un modèle surfacique

Pour construire un modèle approprié à l'application il existe deux possibilités [6] :

- Adaptation d'un modèle préexistant.
- Création d'un modèle à partir des données disponibles.

I.4 Méthodes de modélisation

La modélisation 3D d'un environnement ou d'un objet à partir de données sensorielles, consiste à acquérir des données depuis les capteurs positionnés en divers points de vue, puis à générer une représentation adaptée à la tâche qui exploitera le modèle obtenu. Cela consiste souvent à faire une approximation des points de mesure par des fonctions mathématiques [7].

Dans le cas de la modélisation d'objet, la forme de la surface est déterminée une fois sa représentation mathématique est connue. Pratiquement, dans la plupart des situations de conception de nouvelles surfaces, le concepteur n'a en tête que la forme de la surface qu'il veut obtenir sans connaître ni sa représentation mathématique, ni ses propriétés géométriques.

Pour résoudre ce problème dans le domaine de la conception et de la modélisation des surfaces, il existe plusieurs méthodes de modélisation, celle qui nous intéresse c'est la méthode basée sur l'utilisation des points.

I.4.1 Méthodes basées sur l'utilisation des points

Trois approches peuvent être utilisées pour la génération d'une surface à partir d'un ensemble de points décrivant la forme de la surface [8] :

- a) **Interpolation** : c'est la génération d'une surface qui passe par l'ensemble de points (Figure 1.4).

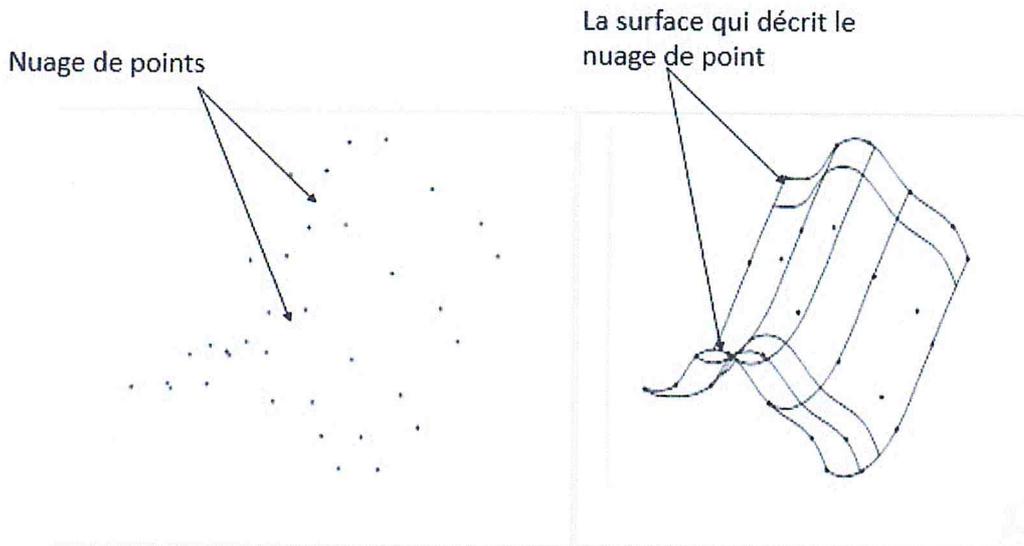


Figure 1.4: Méthode de modélisation de surface par interpolation [8]

- b) **Approximation** : génération d'une surface qui approxime le mieux un ensemble de points mais qui ne passe pas nécessairement par ces points (Figure 1.5).

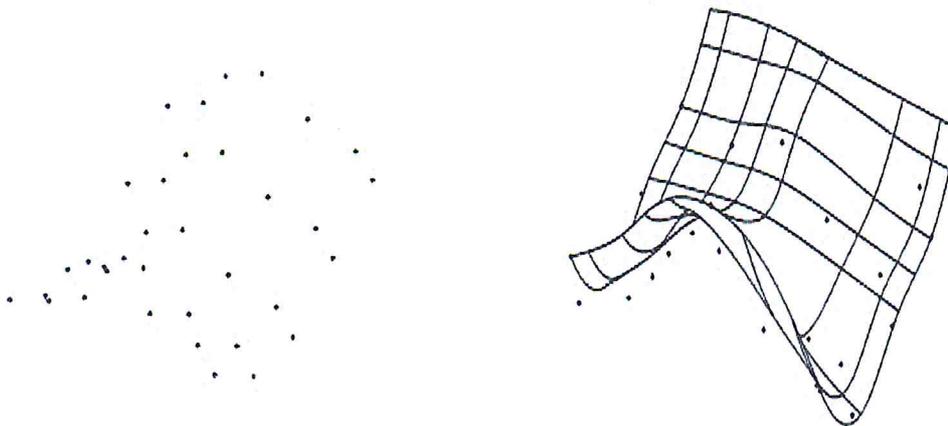


Figure 1.5: Méthode de modélisation de surface par approximation [8]

- c) **Surface Paramétrique** : C'est une surface continue définie par un réseau de contrôle représentant la forme de la surface (Figure 1.6).

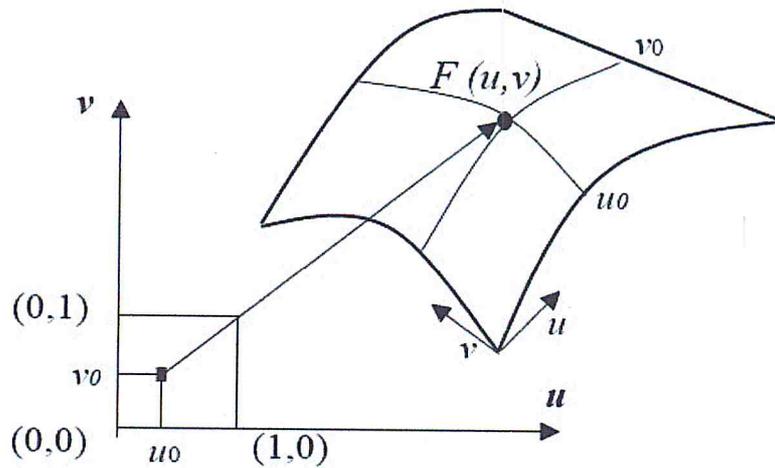


Figure 1.6 : localisation d'un point sur une paramétrique

Généralement, les surfaces paramétriques ne sont pas utilisées séparément, mais plusieurs morceaux de ces surfaces sont joints ensemble pour former une surface plus complexe, Parmi ces surfaces (Bézier, Bézier-rationnelle-Spline et NURBS), La relation qui existe entre les différents types de représentation des surfaces est montrée par la figure suivante (Figure 1.7) [8].

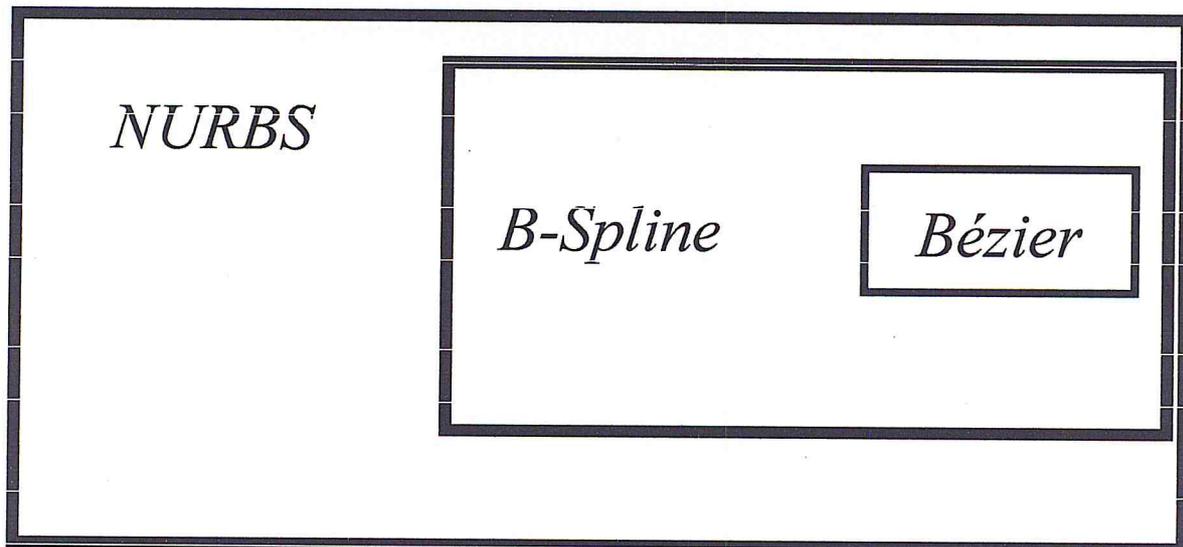


Figure1.7 : Relation entre les différentes représentations [8].

I.5 Avantages et inconvénients

Les méthodes de modélisation vu précédemment (Figure 8) sont flexibles et possèdent plusieurs importantes propriétés, facilitant la conception des courbes et des surfaces, en contrepartie elles ne sont pas aptes à représenter des surfaces très complexes car elles demandent des calculs intensifs. Le principal problème provient de la complexité des surfaces modélisées qui ne répondent pas uniquement aux contraintes fonctionnelles de conception, ceci nous amène à prospecter dans une autre approche dite indirecte (Ingénierie inverse) [1].

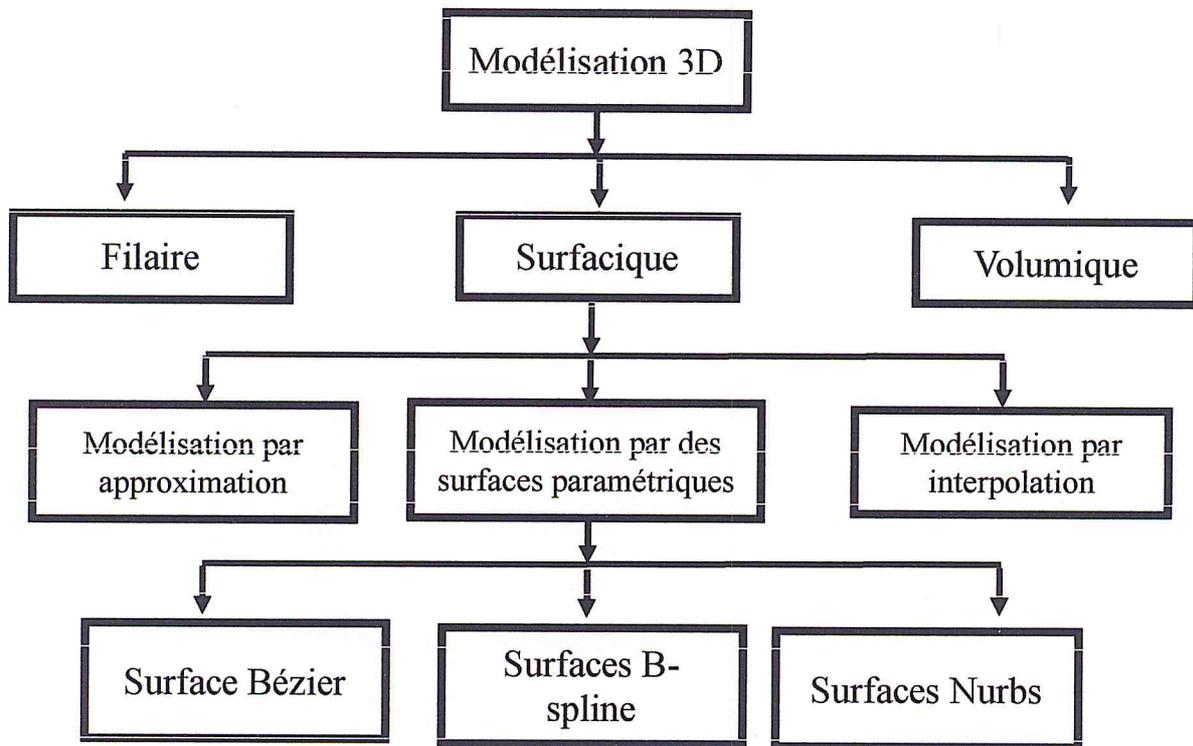


Figure 1.8 : Schéma générale de la modélisation 3D

II. Ingénierie inverse

II.1 Introduction

Les systèmes de conception et de fabrication assistés par ordinateur (CFAO) sont utilisés dans diverses industries afin de créer des objets physiques à partir de modèles numériques, cependant la rétro-ingénierie c'est l'opération inverse qui consiste à conférer un modèle numérique à un objet physique [1].

II.2 Définition

L'ingénierie inverse est le processus de découverte des principes technologiques d'un objet ou d'un composant à travers l'analyse de sa structure et de sa fonction. Une telle analyse peut ensuite être utilisée pour redessiner l'objet très rapidement en utilisant la conception assistée par ordinateur.

L'ingénierie inverse est une technique qui permet de donner une représentation numérique (mathématique) à un modèle physique existant. Cette technique peut être utilisée comme un outil de conception pour produire une copie d'un objet. Son principe repose sur l'acquisition d'un nuage de points issu de la surface de la pièce à digitaliser. Ce nuage de points est traité à travers des logiciels de reconstruction de surfaces, il passera par différentes phases pour pouvoir arriver en fin à l'objet final [9, 10].

II.3 Raisons d'utilisation de l'ingénierie inverse

Cette méthode a été conçue pour plusieurs raisons, nous citons [9, 10] :

- Le produit original n'existe pas, mais le client a besoin d'un produit.
- Le producteur original a arrêté de fabriquer le produit.
- La documentation de conception du produit original est perdue.
- La surface est très complexe pour être conçue par la méthode directe.

II.4 Domaines d'application

La reconstruction d'objets est une discipline relativement jeune, qui intéresse plusieurs domaines d'application et de recherches parmi ceux, nous citons [9] :

- Génie mécanique tel que design et étude de style, clonage et recopiage des œuvres d'art.
- Génie médicale tel que les technologies des prothèses.
- Vision artificiel et robotique.

- L'automobile.
- Aérospatial.
- l'ingénierie médicale.

11.5 Processus de l'ingénierie inverse

Le processus de la reconstruction d'un objet en utilisant l'ingénierie inverse nécessite le passage par trois principales étapes (Figure 1.9) [11] :

Étape 1 : Digitalisation ou la numérisation de l'objet pour l'acquisition d'un nuage de point.

Étape 2 : Traitement du nuage de point.

Étape 3 : Reconstruction du modèle CAO à partir du nuage de points.

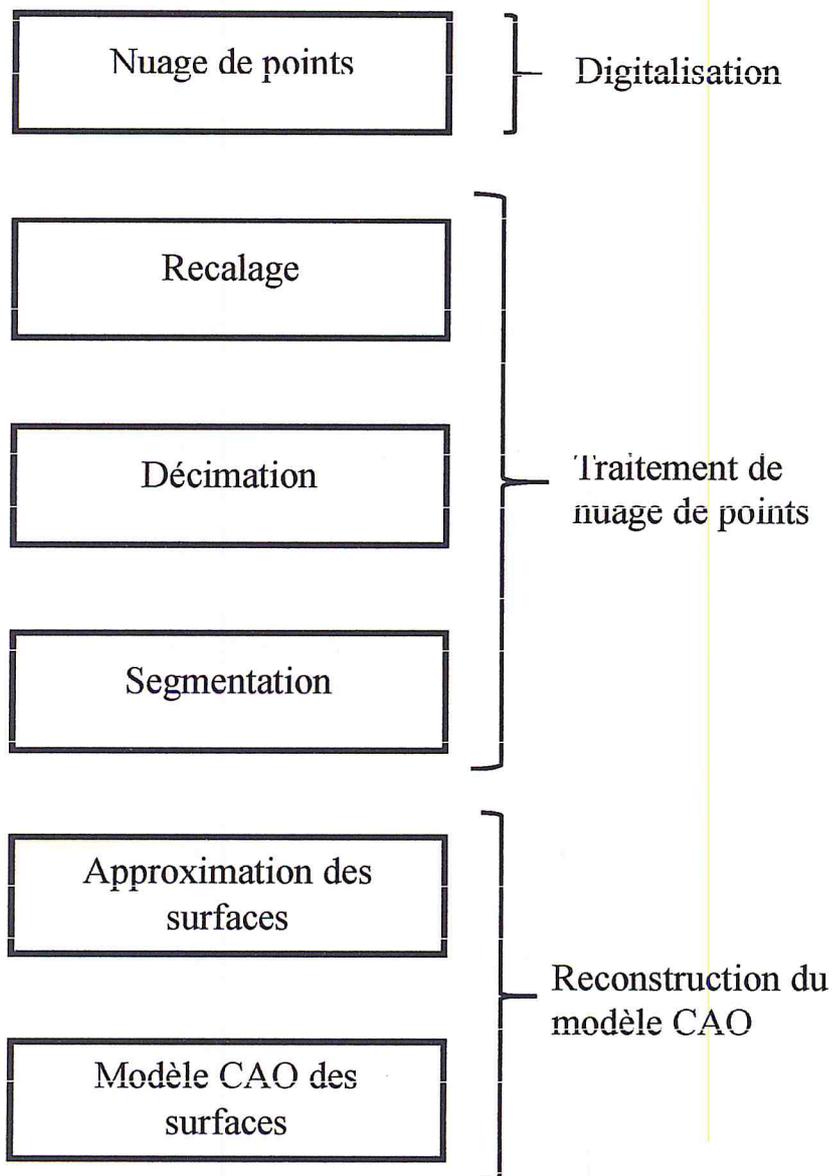


Figure 1.9 : Étapes du processus de l'ingénierie inverse [13].

II.5.1 Numérisation de l'objet

La numérisation est une étape qui permet la mesure d'objets tridimensionnels. Cette mesure se traduit par un ou plusieurs ensembles de point 3D qui, une fois assemblés, forment l'image de profondeur de l'objet. Ces ensembles de points sont constitués de suites de points représentés par les coordonnées cartésiennes tridimensionnelles. L'ensemble de ces triplets offre une définition de la, ou des surfaces de l'objet mesuré sans indication sur son type topologique, sa géométrie, ses frontières ou lignes de séparation [12].

Il existe un grand nombre de méthodes d'acquisition de formes, simples ou complexes, chaque méthode utilise un mécanisme ou un phénomène physique, afin d'interagir avec la surface de l'objet à numériser. Ces méthodes sont classées principalement en deux types [12] :

- a) **Les méthodes à contact :** Les méthodes à contact agissent directement sur l'objet. Au touché, un signal est émis, l'événement point se produit et la prise des coordonnées est exécutée. Dans ce type de méthodes, on retrouve exclusivement les machines à mesurer tridimensionnelles (MMT), la (Figure 1.10) montre le résultat d'une numérisation d'un objet.
- b) **Les méthodes sans contact :** Utilisent soit la lumière, soit le son ou alors les champs magnétique comme agent de mesure à partir des entités physique tel que « capteur CCD source laser ». La récupération de cet agent de mesure dans son état de retour, permet de mesurer les coordonnées localisant le point ciblé. Dans chaque cas, une analyse appropriée doit être menée, afin de déterminer la position des points de la surface à partir de l'entité physique mesurée.

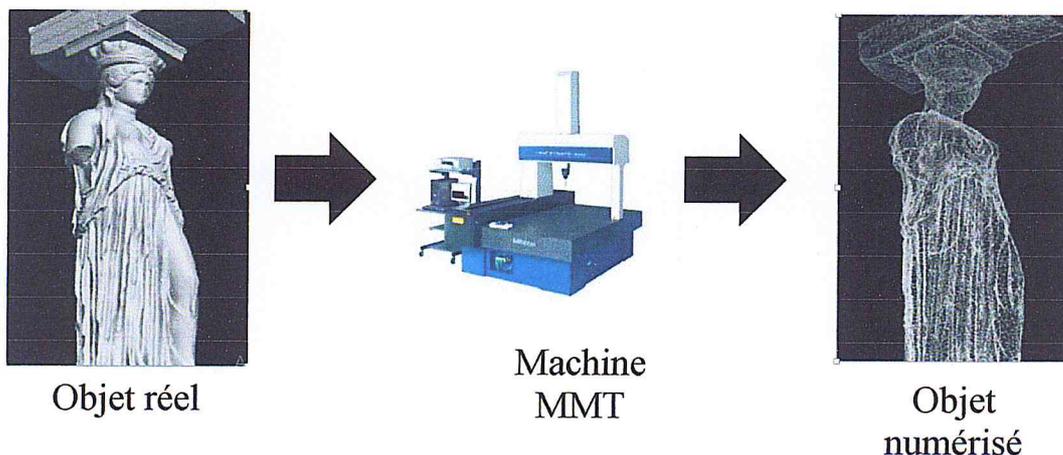


Figure 1.10 : numérisation d'un objet

II.5.2 Recalage

Un objet de forme complexe ne peut pas être numérisé en une seule passe, plusieurs points de vue sont nécessaires. Chaque repositionnement de l'objet induit une nouvelle passe. Cette numérisation donne naissance à plusieurs nappes de points qui se superposent partiellement. Cette étape consiste à aligner les images de profondeur dans un même repère pour ensuite les fusionner, de manière à obtenir un nuage de points répartis sur toute la surface numérisée de l'objet [10].

Donc il est nécessaire de mettre en correspondance ces diverses nappes dans un même repère objet. Afin de recalibrer plusieurs ensembles de données 3D entre eux, la figure suivante (Figure 1.11) montre les différentes acquisitions nécessaires afin de numériser entièrement l'objet placé au centre de la figure. L'objet en question est un casque de sécurité [9].

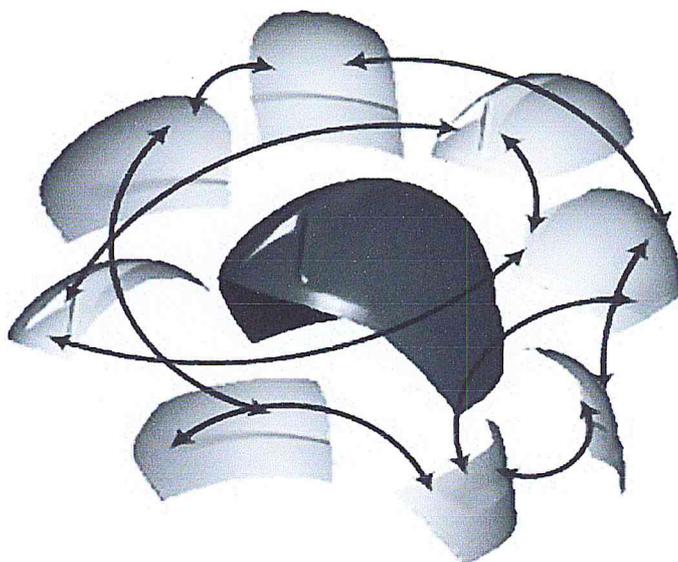
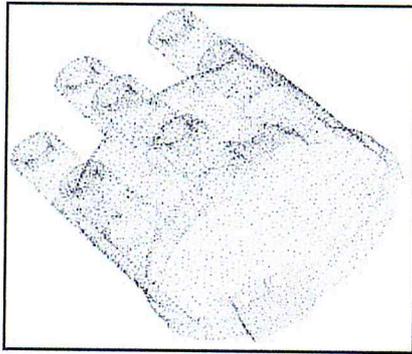


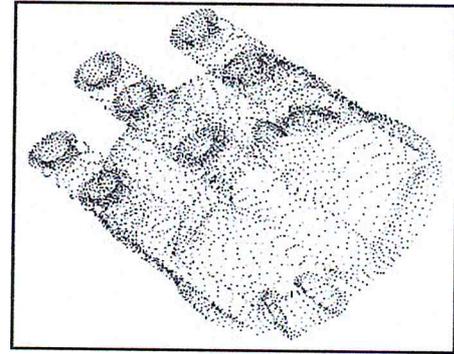
Figure 1.11 : recalage des nuages de points [9].

II.5.3 Décimation

Une simplification s'avère parfois nécessaire afin de rendre le nuage de points manipulable et traitable sur ordinateur avec une puissance de calcul et des ressources mémoire raisonnables, son objectif est de diminuer le nombre de points du nuage et garantir le maintien d'une définition satisfaisante de l'objet, comme le montre la (Figure 1.12) [13] :



Avant décimation



Après décimation

Figure 1.12 : exemple d'un objet avant et après décimation [14]

II.5.4 Segmentation

La segmentation du nuage de points, est l'une des étapes les plus critiques et qui consiste à subdiviser le nuage de points en un nombre fini de régions distincts (régions caractéristiques), en se basant sur un ensemble de critères (Figure 1.13) [9].

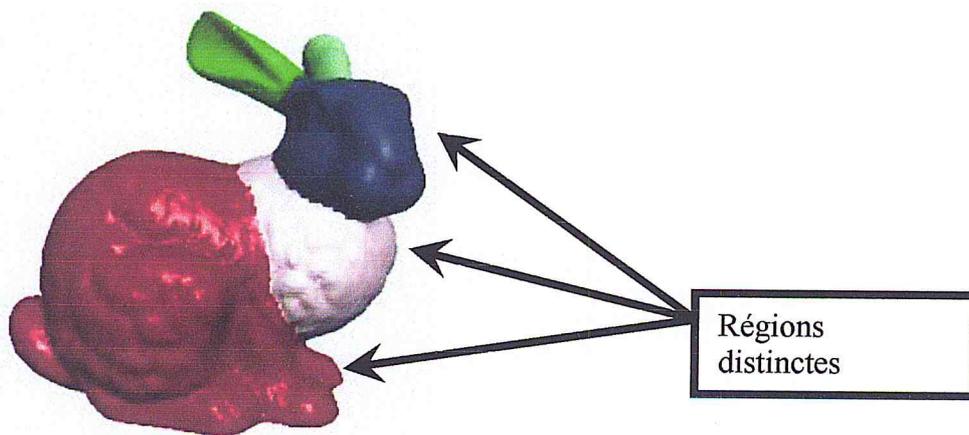


Figure 1.13 : Segmentation d'un lapin

II.5.5 Approximation des surfaces :

C'est l'étape fondamentale, qui consiste à reconstituer la surface de l'objet numérisé à partir de l'échantillon de points. Cette étape de reconstruction de surface est au centre de la problématique de notre projet.

Une fois que les régions sont déterminées, il faut les approximer par des modèles géométriques de surface adéquats, en utilisant des techniques de lissage comme l'interpolation et l'approximation tout en assurant une certaine continuité entre les frontières de ces surfaces,

les surfaces résultante sont des surfaces qui approximent le nuage de point comme le montre la (Figure 1.14).

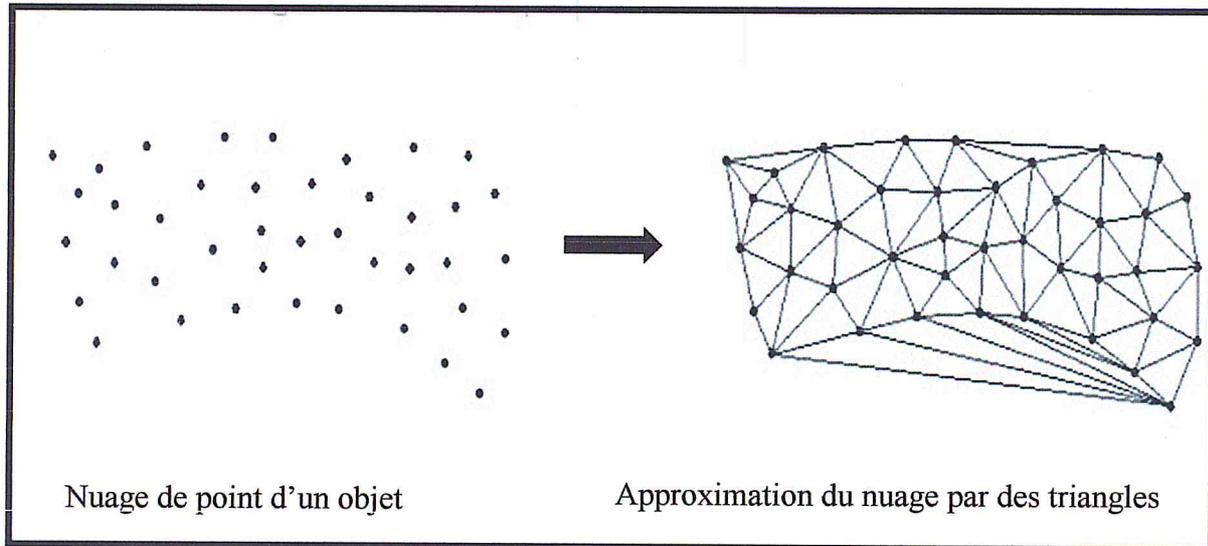


Figure 1.14 : Exemple d'une approximation par des triangles.

II.6 Avantages de l'ingénierie inverse

L'ingénierie inverse présente de nombreux avantages dans les situations suivantes [14] :

- Elle permet de concevoir un objet malgré l'absence de la documentation.
- Permet de soutenir des modifications sur la fabrication de l'objet malgré l'insuffisance du modèle originel de CAO.
- Répond aux besoins de la fabrication de l'objet malgré que le fabricant originel n'existe plus ou ne fabrique plus le produit.
- Renforcement des fonctionnalités
- Amélioration de la performance et/ou les fonctionnalités de produit
- Actualisation des matériaux désuets ou des processus de fabrication désuets.

III. Étude des méthodes de reconstruction des objets 3D

III.1 Introduction

L'objectif final du processus du Reverse Engineering est la reconstruction du modèle informatique « Modèle CAO ». Cependant, cette étape est très importante, fastidieuse et consommatrice de temps et dépend de l'expérience du concepteur.

Afin de faciliter cette tâche, il est préférable d'approximer le nuage de points par un ensemble d'éléments géométriques simples (triangle, tétraèdre, etc.). Plusieurs algorithmes ou techniques ont ainsi été développés pour produire en sortie une surface approchant au mieux la forme de la surface physique de l'objet [15].

III.2 Méthodes de reconstruction

III.2.1 Techniques de reconstruction par approximation

La reconstruction par approximation signifie que la surface finale reconstruite passe à proximité des points, plutôt que sur les points, Cette technique revient à approximer les points par une fonction continue. Elle peut être réalisée par plusieurs méthodes, citons [16] :

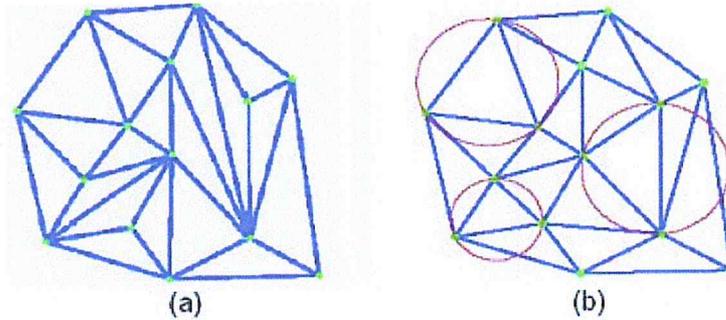
- a) **Liste square** : La méthode Least Square est une méthode d'approximation qui permet de trouver une solution plus proche de la solution réelle du problème indéterminé, Généralement cette méthode est combinée avec d'autres méthodes d'optimisation.
- b) **Poisson surface reconstruction** : c'est une fonction implicite qui fusionne les schémas d'ajustement global et local. Elle permet d'examiner toutes les données en une fois, c'est pourquoi est une solution globale, et aussi peut créer des surfaces très lisses En même temps et couvrir les surfaces vides par remplissage du model existant.

III.2.2 Techniques de reconstruction par interpolation

La géométrie algorithmique a également apporté des contributions importantes au problème de la reconstruction 3D. Les méthodes proposées réalisent généralement une interpolation des points. Il existe plusieurs méthodes de reconstruction de maillages basées sur l'opération de « triangulation » 2D ou 3D.

La triangulation la plus utilisée est celle de Delaunay qui possède des propriétés géométriques remarquables, et qui permet d'optimiser un certain nombre de critères. De plus, cette triangulation définit une représentation unique et proche de la forme réelle [15].

(Figure 1.15) illustre une triangulation 2D pour un ensemble de points [2].



a. triangulation quelconque

b. une triangulation Delaunay

Figure 1.15 : Triangulation quelconque et une triangulation Delaunay [2].

III.3 Triangulation de Delaunay

L'étape de construction de la triangulation étant d'une importance essentielle dans le processus de reconstruction, la triangulation de Delaunay se fait généralement dans un espace à deux, à trois dimensions ou plus. De façon générale, le terme triangulation désigne une triangulation d'un ensemble de points dans un espace 2D, ou une triangulation 3D d'un ensemble de points dans l'espace 3D.

III.3.1 Les dérivés de Delaunay

Il existe plusieurs méthodes basées sur l'utilisation de la triangulation de Delaunay, parmi ceux :

- **Boissonnat** : cette méthode est proposée par le chercheur Boissonnat, elle permet d'utiliser la triangulation de Delaunay pour reconstruire une forme 3D unique en sculptant cette triangulation cependant, cette méthode restait trop vague [16].
- **Alpha Shape** : cette méthode est proposée par Edelsbrunner et Mücke qui permet de construire à partir de la triangulation de Delaunay une famille de formes qui dépendent d'un paramètre alpha [16].

III.3.2 Propriétés de la triangulation de Delaunay

La triangulation de Delaunay est assurée par plusieurs propriétés [17] :

- a) **Maximum de l'angle minimum** : Pour quatre points en position convexe, Delaunay maximise le plus petit angle (Figure 1.16).

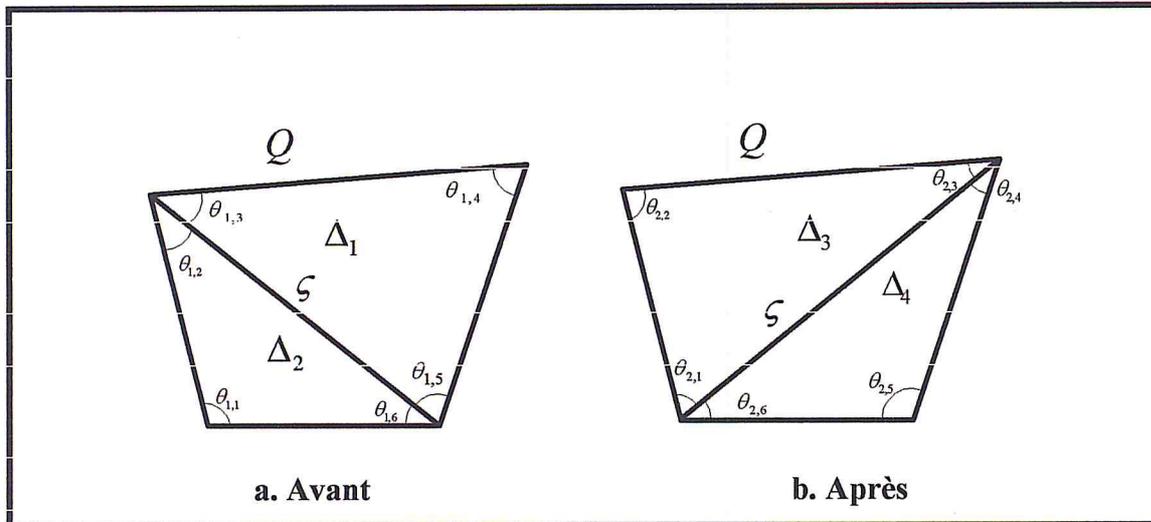


Figure 1.16 : Triangulations possibles d'un quadrilatere Q .

b) Enveloppe convexe : les arêtes extérieures de la triangulation de Delaunay d'un ensemble de points S constituent la frontière de l'enveloppe convexe de S .

c) Cercles circonscrits : Parmi toutes les propriétés, le cercle circonscrit à un triangle jouera un rôle très important dans les algorithmes de triangulation comme nous le verrons plus tard. Ce cercle est défini par son centre et son rayon.

Pour chaque triangle (tétraèdre dans le cas 3D) du maillage, son cercle (sphère dans le cas de 3D) circonscrit ne doit contenir aucun élément de l'ensemble des sommets.

Pour une meilleure visualisation nous allons donner une représentation en 2D de la propriété du cercle circonscrit (Figure 1.17).

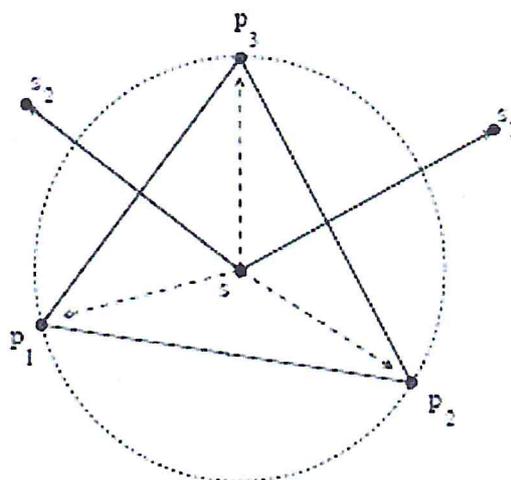


Figure 1.17 : Cercle circonscrit

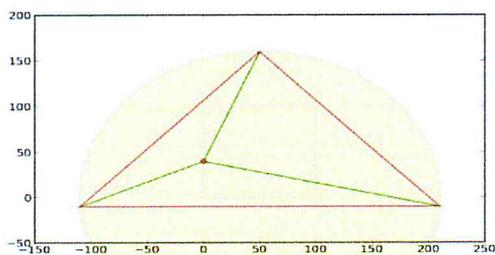
III.3.3 Méthodes de calculs de la triangulation de Delaunay

Il existe plusieurs algorithmes qui génèrent de façon incrémentale la triangulation de Delaunay, On parle de techniques incrémentales lorsque nous partons d'une triangulation et, en ajoutant un point, ces algorithmes sont [13,18] :

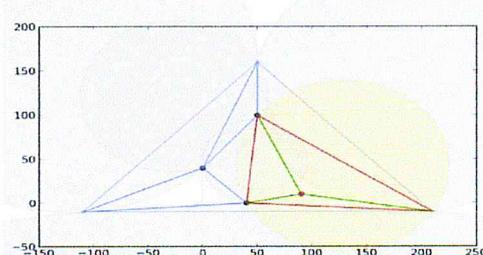
a) **Triangulation par Destruction/Construction** : elle se fait à partir d'une triangulation partielle du semis, elle passe par les étapes suivantes (Figure 1.18.) [19] :

1. Création du super triangle à partir de trois points de départ.
2. Identification des triangles dont le cercle circonscrit contient le point ajouté.
3. Stockage de toutes les arêtes de ces triangles
4. Suppression des arêtes dupliquées.
5. Suppression des triangles dont le cercle circonscrit contient le point ajouté.
6. Création de nouveaux triangles, en utilisant le point inséré courant les arêtes stockées
7. Ajout de nouveaux triangles à la triangulation
8. Suppression des triangles ayant en moins une arête en commun avec le super triangle

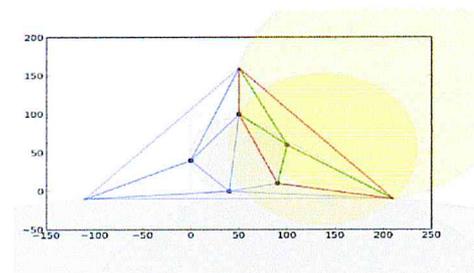
On utilise donc principalement deux méthodes : une fonction récursive qui permet la suppression des triangles (destruction), l'autre fonction permet la création de nouveaux triangles (construction).



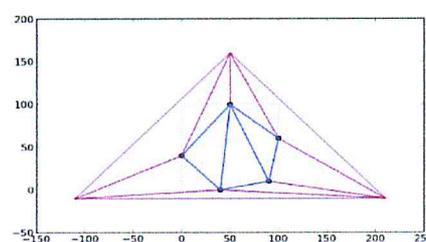
Insertion d'un point dans un « super-triangle »



Insertion 2, 3,4 point.



Suppression des arêtes du super-triangle.



Suppression des arêtes du super-triangle.

Figure 1.18 : les Etapes de l'algorithme destruction et instruction.

b) Algorithme de Flip

Cet algorithme est plus rapide que le premier. C'est une suite de fonctions récursives (qui pourront être implémentées en itératif), vérifient si les côtés sont légaux. Un côté est défini comme étant l'arête commune à deux triangles. Pour savoir s'il est illégal, nous regardons les points qui ne sont présents que dans un des triangles (donc les points opposés).

Un côté est considéré comme étant illégal si au moins un de ces deux points est contenu dans le cercle circonscrit du triangle opposé, Si c'est le cas, le côté illégal est remplacé par un côté relié par les deux points opposés. Et on testera les quatre côtés qui restent pour voir si ce changement a provoqué un autre conflit [17].

IV. Format d'échange de données

IV.1 Introduction

La CAO et la FAO, sont deux différents systèmes, qui nécessitent un échange d'informations, cet échange doit être assuré par un moyen standard, qui facilite le passage de l'information entre ces deux plateformes, Ces translateurs sont nommés « Formats d'échange de données ».

Il existe plusieurs formats d'échange de données permettant l'échange de données entre divers programmes ou logiciels par l'intermédiaire d'un fichier. Plusieurs formats sont apparus, les formats natifs développés par les éditeurs de logiciels, puis les formats neutres et les passerelles comme IGES, STEP, DXF PATRAN, STL etc.

Malgré ces nombreux standards existants, aucune solution universelle n'existe pour transférer un modèle CAO d'un système à un autre, d'une manière complète. Les erreurs de conversion ou les pertes d'informations lors du transfert sont fréquents [20].

IV.2 Modèle IGES

Un fichier IGES est un fichier séquentiel constitué d'une séquence d'enregistrements, il se compose de cinq sections qui doivent figurer dans l'ordre suivant (Début de la section, section global, Section de Paramètres de données, section d'entrée d'annuaire, et fin de la section [21].

IV.2.1 Avantages et inconvénients

- Avantages :
 - Echange de plusieurs formes des modèles de données.
 - L'export des informations porte sur la définition mathématique de la surface ainsi toutes ses propriétés initiales.
- Inconvénients :
 - Limité et obsolète
 - l'export vers un format IGES ne permettra de sauvegarder que l'extérieure du solide
 - Ce format n'est pas adapté au transfert des données non géométriques
 - Trop grand et de taille et complexe

IV.3 Modèle STEP

C'est une norme ISO pour présenter des objets 3D et des informations connexes. Le développement de STEP a commencé en 1984 en tant que successeur d'IGES, le plan initial été d'obtenir une seule édition mais à cause de la complexité, la norme a dû être divisée en plusieurs parties distinctes c.à.d. son interprétation varie d'un éditeur à un autre ce qui entraîne une erreur ou perte d'information sur les modèles. De plus STEP inclut un langage de modélisation de données qui permet de décrire la représentation de données [20].

IV.3.1 Avantages et inconvénients

- Avantage :
 - Indépendance des plates-formes matérielles et logicielles
 - Interopérabilité entre logiciels
 - Longévité de données archivées.
 - Extensibilité et modularité du fichier.
- Inconvénients :
 - Beaucoup d'erreurs et d'échecs à cause de la diversité de son interprétation
 - Grande perte d'information.
 - Le processus d'interprétation des protocoles d'application est décrit de façon informelle.
 - difficultés d'intégration.

IV.4 Modèle STL

Le format STL est le plus largement utilisé, il est initialement conçu pour exporter les modèles CAO vers les machines de prototypage rapide de stéréo lithographie. Ce format a été lancé par 3D système en 1989.

La discrétisation permet d'obtenir des fichiers robustes lisibles sans problèmes sur un grand nombre de plateformes. Le format de fichier STL ne décrit que la géométrie de surface d'un objet en 3D (Figure1.20) et ne comporte pas d'informations concernant la couleur, la texture ou les autres paramètres habituels d'un modèle de conception assistée par ordinateur. Le format STL est spécifié comme [22] :

- Un fichier Binaire plus commun, moins de taille mais non lisible et trop condensé.
- un fichier ASCII (caractère imprimable) qui est plus descriptif et lisible mais gros de taille, ce fichier commence par une ligne de description de (nom, nom de l'auteur...) précédée par le mot réservé « **Solid** » suivie par les composantes de la normal puis les coordonnées des sommets des triangles. Cette partie peut être répétée une ou plusieurs fois. La fin de fichier est indiquée par « **endsolid** » (Figure 1.19).

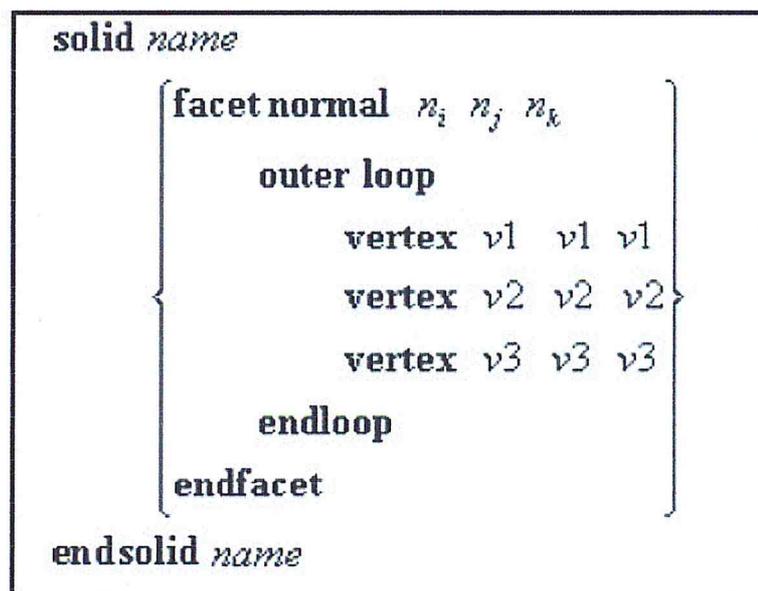


Figure 1.19 : structure d'un fichier STL ASCII

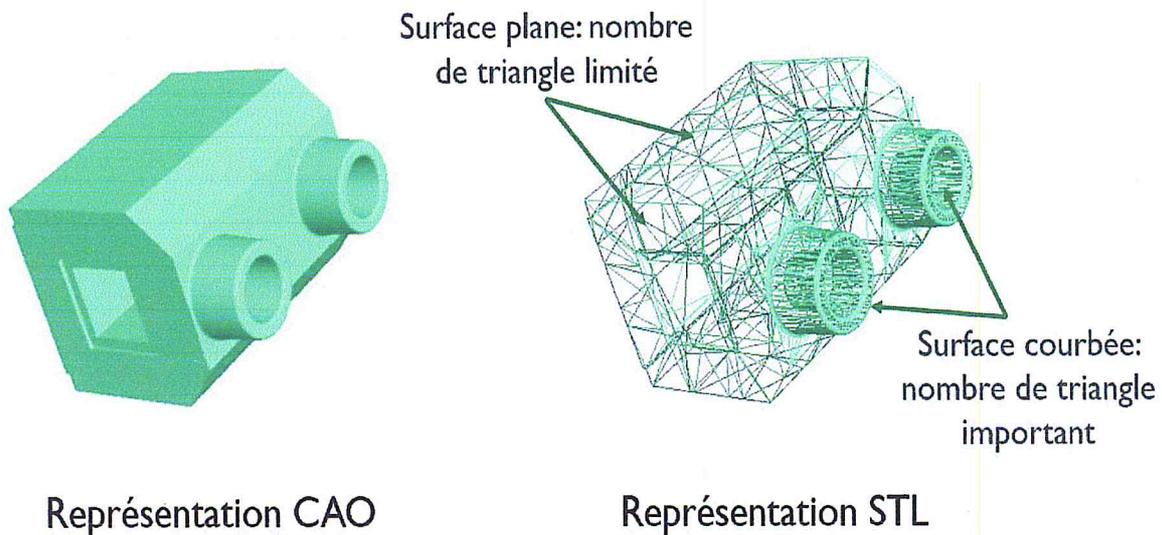


Figure 1.20: Représentation STL d'un objet

IV.4.1 Avantages et inconvénients

➤ Avantages :

- Indépendance des plates-formes matérielles et logicielles
- Interopérabilité entre logiciels.
- Longévité de données archivées.
- Extensibilité et modularité

➤ Inconvénients :

- Supprime les informations de connectivité.
- Le support des couleurs n'est pas standard.

Afin d'utiliser la représentation 3D des objets nous avons présenté les différents formats d'échange de données et nous avons mis l'accent sur le format STL qui est le support de notre étude.

IV.5 Conclusion :

Dans ce chapitre nous nous sommes intéressés au processus de conception des surfaces des objets complexes, ensuite nous avons mis en évidence une méthode inverse dite « l'ingénierie inverse » qui a pour objectif de reconstruire un objet en donnant une représentation numérique (mathématique) à un modèle physique existant.

Diverses méthodes de reconstruction existent, chacune ayant ses artifices et ses limitations. Malgré les Résultats atteints, les étapes de la reconstruction souffrent toutes de difficultés qui font l'objet de Recherche soutenues. Les approches sont nombreuses et parfois très différentes. Elles répondent actuellement certes à un besoin industriel urgent mais demeurent encore onéreuses. Chaque méthode de reconstruction présente des inconvénients propres, d'où l'absence d'un schéma de reconstruction unifié.

Parmi les méthodes d'approximation existantes, il y a celles qui s'appuient sur l'approximation du nuage de points par des éléments géométriques. Pour reconstruire le modèle 3D, nous avons utilisé et adapté la triangulation de Delaunay, celle-ci étant classiquement réservée à la triangulation 2D ou 3D.

Afin d'utiliser la représentation 3D des objets nous avons présenté les différents formats d'échange de données entre la CAO et la FAO et nous avons mis l'accent sur le format STL [15].

CHAPITRE II

SOLUTIONS PROPOSEES

I. Introduction

La reconstruction 3D permet l'obtention d'une représentation en trois dimensions d'un objet à partir d'un ou plusieurs nuages de points.

Le but recherché est de produire en sortie une surface approchant au mieux la forme de la surface physique de l'objet. Pour cela nous avons utilisé dans notre projet la triangulation de Delaunay qui possède des propriétés géométriques remarquables et qui permet d'optimiser un certain nombre de critères. De plus, elle est l'une des rares approches qui définit de manière unique l'objet que l'on veut construire [15].

II. Problématique

Les objets de formes complexes ou les surfaces gauches sont utilisés dans la conception, et la réalisation des pièces mécaniques, médicales, robotique...etc. ces modèles peuvent être obtenus par deux méthodes. La première méthode se base sur l'utilisation des modèles CAO, cependant il n'est pas toujours évident d'obtenir des formes voulues, dans le cas où le modèle CAO n'est pas disponible, une deuxième méthode est utilisée, l'ingénierie inverse qui repose sur l'acquisition d'un nuage de points avec une machine à mesure tridimensionnelle « MMT », et par la suite faire des traitements appropriés pour générer un modèle CAO de l'objet.

Avant d'arriver à la génération du modèle CAO, il existe plusieurs étapes que nous avons citées et détaillées dans le chapitre précédent. (Figure 2.1) montre l'architecture de reconstruction des surfaces réalisées auparavant en deux dimensions par l'équipe CFAO, mais qui est limitée en terme de temps à cause de la taille volumineuse des nuages de points, ce qui demande un temps de traitement très important, et aussi en terme de précision et représentation d'objets car la représentation en deux dimensions ne gère que des points et des lignes sans aucune notion de pièce et ne permet pas de générer le modèle STL.

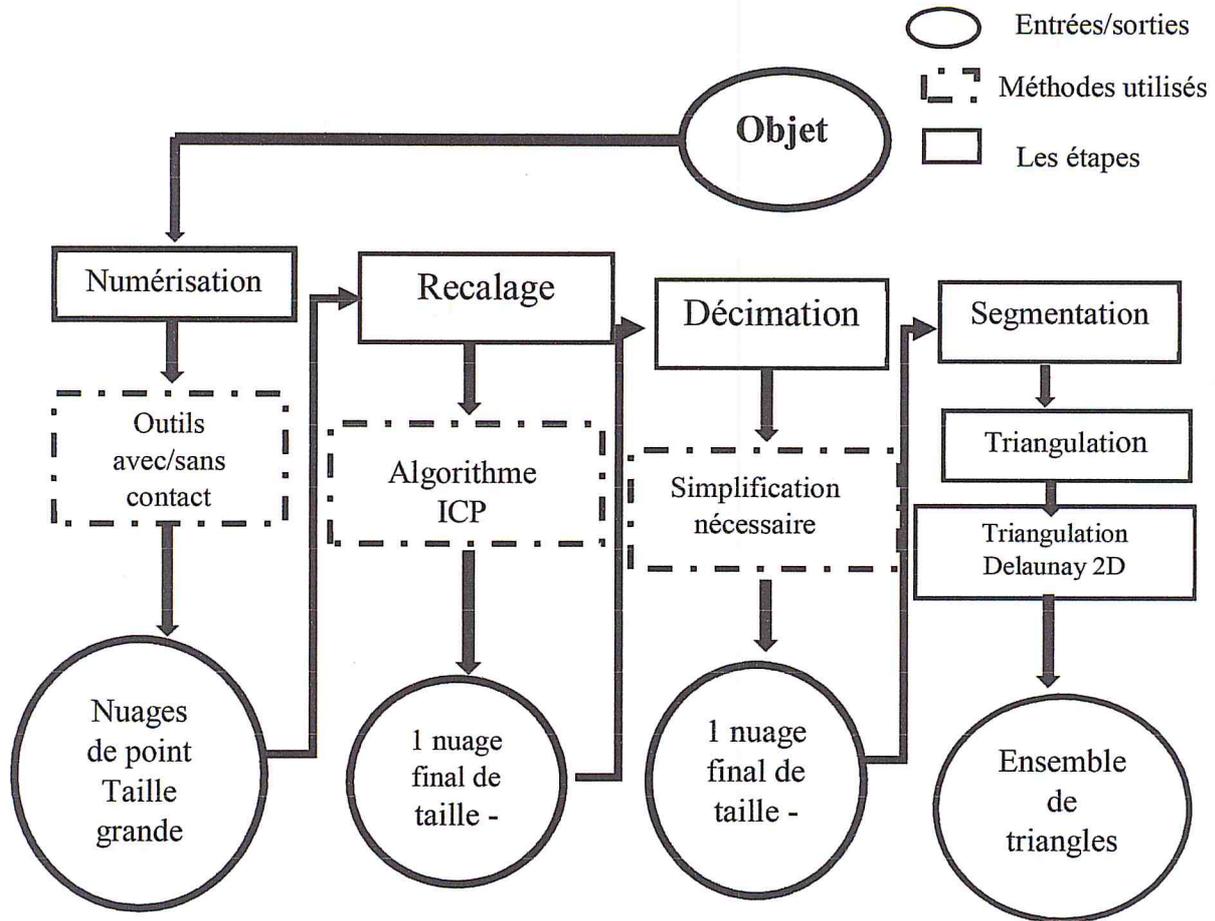


Figure 2.1 : Processus de reconstruction des surfaces en deux dimensions

Le processus existant (représentation des objets en deux dimensions) présente différents problèmes qui sont :

- Problème de précision (2D)
- Problème de lenteur de temps de modélisation (nuages de points volumineux)
- Cette modélisation ne permet pas de générer un modèle STL de l'objet.

Pour éviter ces problèmes nous proposons une nouvelle architecture logicielle (Figure 2.2) qui permettra :

- Augmentation de la précision (3D)
- Activation du processus de modélisation
- Génération du modèle STL de l'objet.

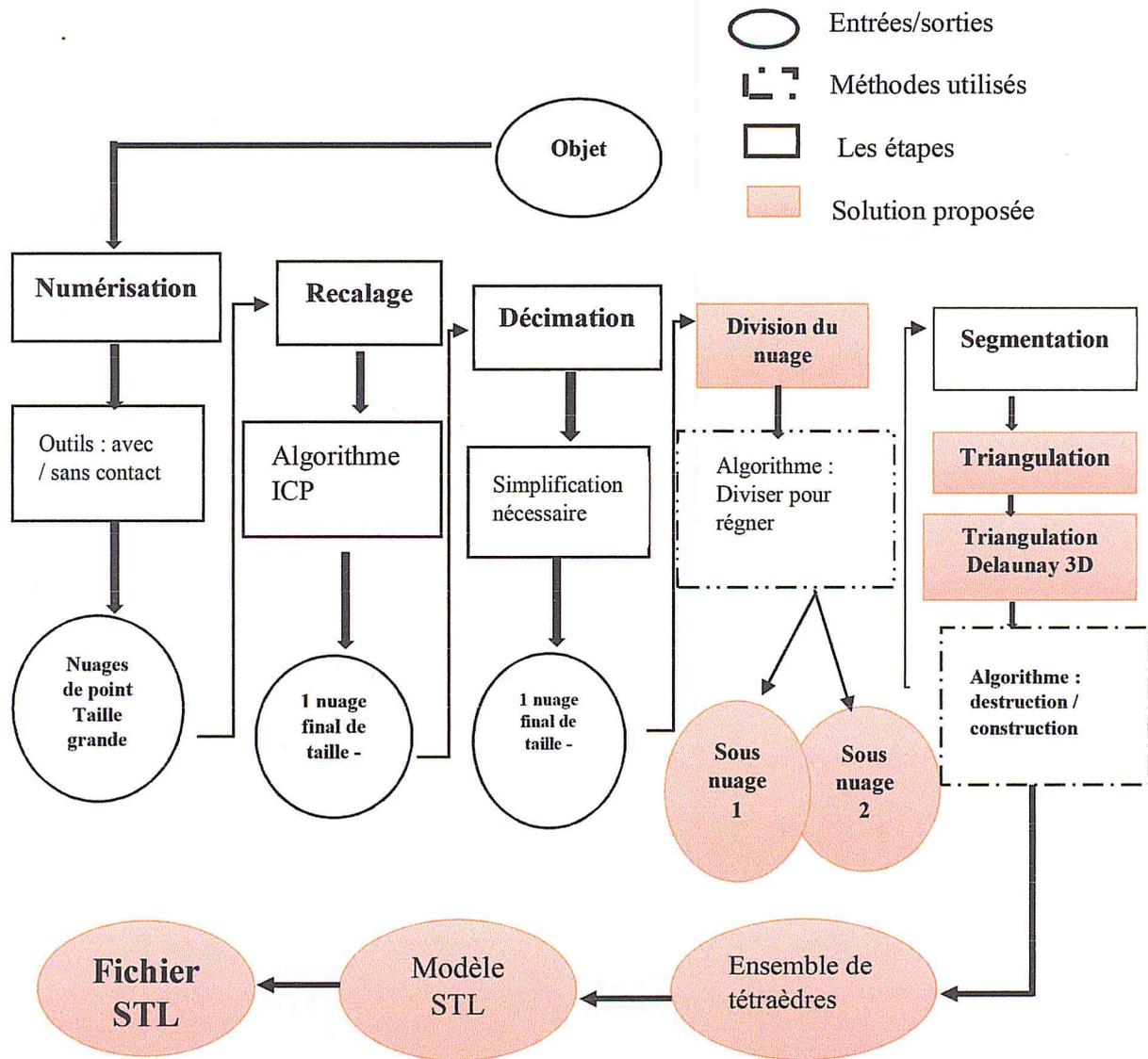


Figure2.2 : Processus proposé pour la reconstruction des surfaces en trois dimensions

III. Objectifs

Notre objectif principal est le développement d'une application distribuée permettant la reconstruction des modèles 3D à partir de plusieurs nuages de points en un temps optimal. Cet objectif peut être subdivisé en plusieurs sous objectifs :

- Développement des algorithmes d'approximation du nuage de points par des tétraèdres en utilisant la triangulation de Delaunay 3D.
- Création d'une méthode qui permettra la reconstruction des nuages de points en un temps optimal.
- Génération du modèle STL d'un objet digitalisé.

Chapitre2 : SOLUTIONS PROPOSEES

Pour atteindre les objectifs prédéfinis nous avons proposé une architecture logicielle, (Figure 2.2) qui permettra la reconstruction des surfaces à partir d'un nuage de points en gagnant dans le temps de modélisation et en taux de précision.

IV. Architecture logicielle pour la reconstruction des surfaces complexes

Notre travail ne portera ni sur la numérisation, ni sur le recalage et la décimation, nous allons nous focaliser sur l'étape de l'approximation des surfaces en trois dimensions, qui est la dernière étape du processus de l'ingénierie inverse. Cependant, cette étape est très importante, fastidieuse et consommatrice de temps et dépend de l'expérience du concepteur. Afin de faciliter cette tâche nous proposons de passer par les étapes présentées dans l'organigramme global de la reconstruction des surfaces en 3D (Figure 2.3) :

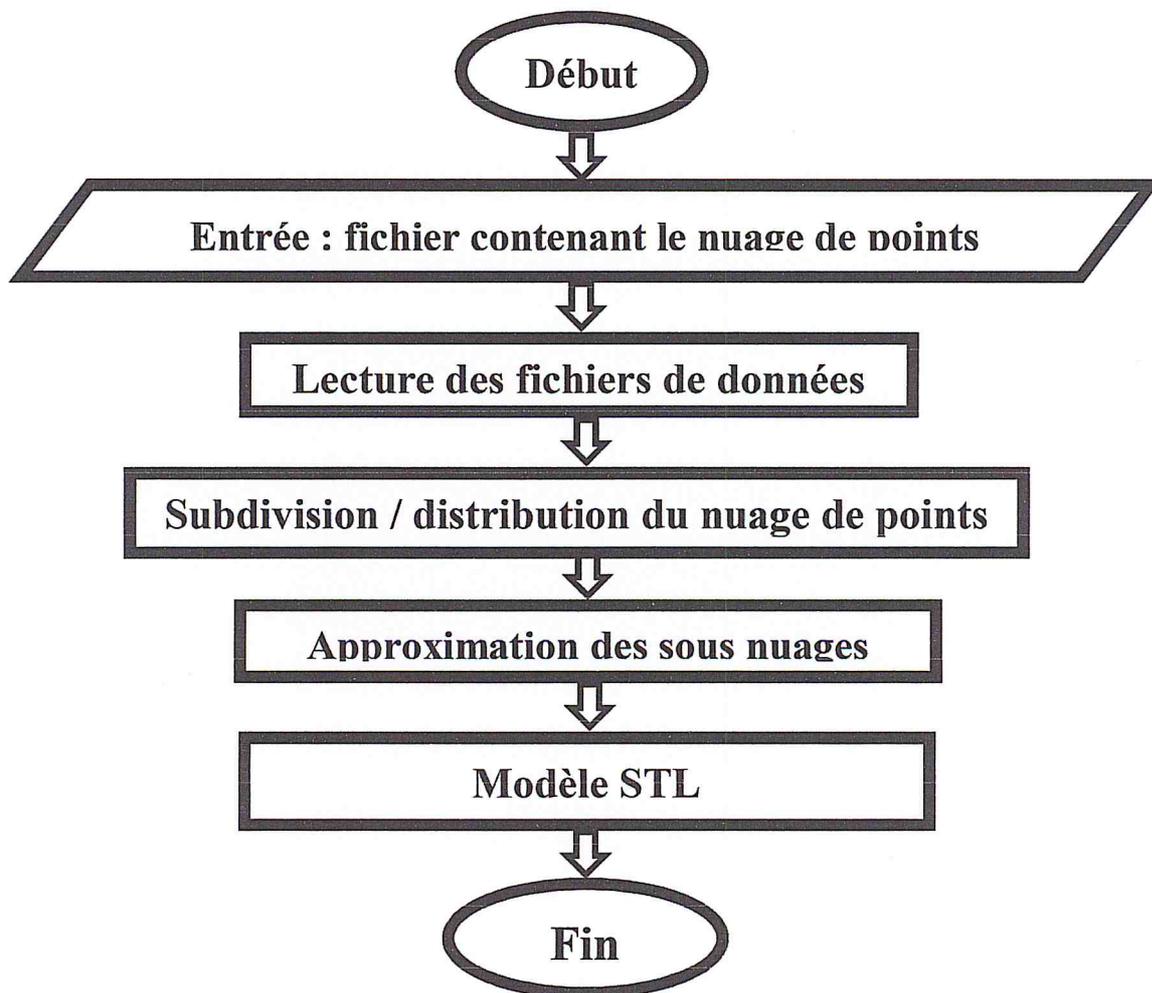


Figure 2.3 : Organigramme global de la reconstruction des surfaces en trois dimensions.

IV.1 Description de l'organigramme global de la reconstruction

L'organigramme global de la reconstruction est divisé en quatre tâches, Chaque tâche principale de l'organigramme global comporte plusieurs sous-tâches :

1. Lecture du nuage de points

- Récupération des points avec leurs coordonnées.
- Elimination des points dupliqués.
- Stockages des points.
- Visualisation du nuage de points.

2. Subdivision et distribution du nuage de points

Après avoir stocké les points nous allons passer à la deuxième étape qui se compose de deux étapes :

- **Etape 1** : Subdivision du nuage de points traité en plusieurs sous-nuages en deux, en quatre, ou en huit sous nuages.
- **Etape 2** : Distribution des sous nuages sur plusieurs corps.

3. Approximation des sous-nuages

- Création du tétraèdre englobant.
- Triangulation de Delaunay.
- Génération du modèle STL de l'objet.

Dans la section suivante nous allons détailler chacune des sous tâches de l'organigramme global.

IV.2 Les tâches de la reconstruction

IV.2.1 Lecture du nuage de point

Lecture des coordonnées des points du nuage à partir d'un fichier de donnée est la première étape dans notre application logicielle. Elle consiste à récupérer les points et leurs coordonnées (x, y, z), qui seront stockés dans des classes et des structures de données en éliminant les points dupliqués pour éviter la redondance et générer par la suite une triangulation correcte.

Pour gagner l'espace mémoire et pour profiter aussi de l'accès rapide aux données, Nous avons utilisé un ensemble de fonctions développées par la suite. La (Figure 2.4) présente les différentes étapes suivies pour la lecture du nuage de point.

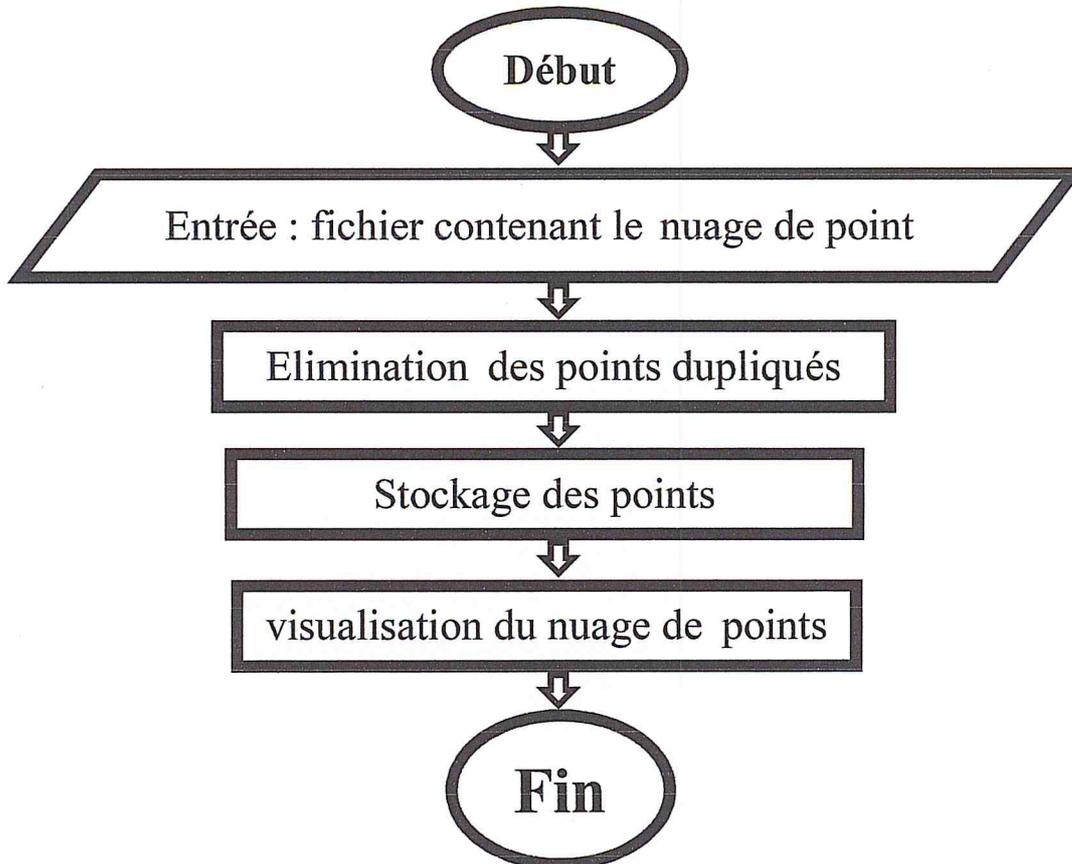


Figure 2.4 : Organigramme de lecture du nuage de points.

IV.2.2 Subdivision/distribution du nuage de points

L'objectif de cette étape est l'optimisation du temps de la reconstruction des surfaces, pour pouvoir atteindre cet objectif nous avons proposé une solution qui est composé de deux étapes :

- Première étape : Subdivision du nuage de point en plusieurs sous nuages (Figure 2.5).

Après avoir stocké le nuage de point traité, cette subdivision peut être réalisée par plusieurs manières, subdivision en deux, quatre, ou huit sous nuage en utilisant l'algorithme de « diviser pour régner ».

L'algorithme « Diviser pour régner » s'applique à de nombreuses constructions algorithmiques. Le principe de la méthode consiste à diviser le problème initial en des sous-problèmes de complexité moindres en termes de taille, c'est l'opération de division. Ces sous-problèmes seront traités, puis la solution du problème initial est construite en fusionnant les solutions des sous-problèmes. C'est l'opération de fusion [21].

- 1^{er} cas : division en deux sous-nuages.
- 2^{ème} cas : division en quatre sous-nuages.

Chapitre2 : SOLUTIONS PROPOSEES

➤ 3^{ème} cas : division en huit sous nuages.

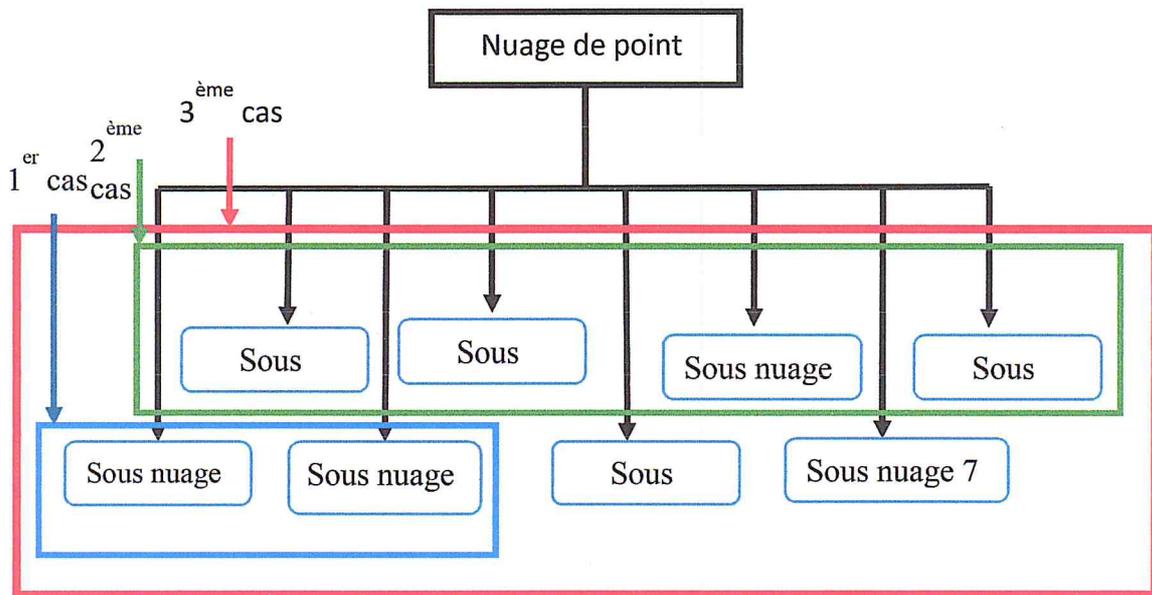


Figure 2.5 : Subdivision du nuage de points.

- Deuxième étape : une fois le nuage de point initial est subdivisé en plusieurs sous nuages, ces derniers doivent être passés par les étapes suivantes (Figure 2.6) :
 1. L'envoi des sous nuages par l'administrateur, aux corps pour pouvoir assurer le parallélisme par la suite.
 2. La modélisation des sous nuages pour générer le modèle STL.
 3. L'envoi des résultats fournis par les corps à l'administrateur.
 4. L'obtention des résultats envoyés par les corps pour pouvoir les concaténer.
 - **Administrateur** : c'est le responsable de la subdivision du nuage de points et l'envoi des sous nuages aux corps.
 - **Corps** : ce sont des objets physiques qui peuvent être plusieurs ordinateurs, ou plusieurs processeurs qui permettent le calcul et la modélisation des sous nuages de points.

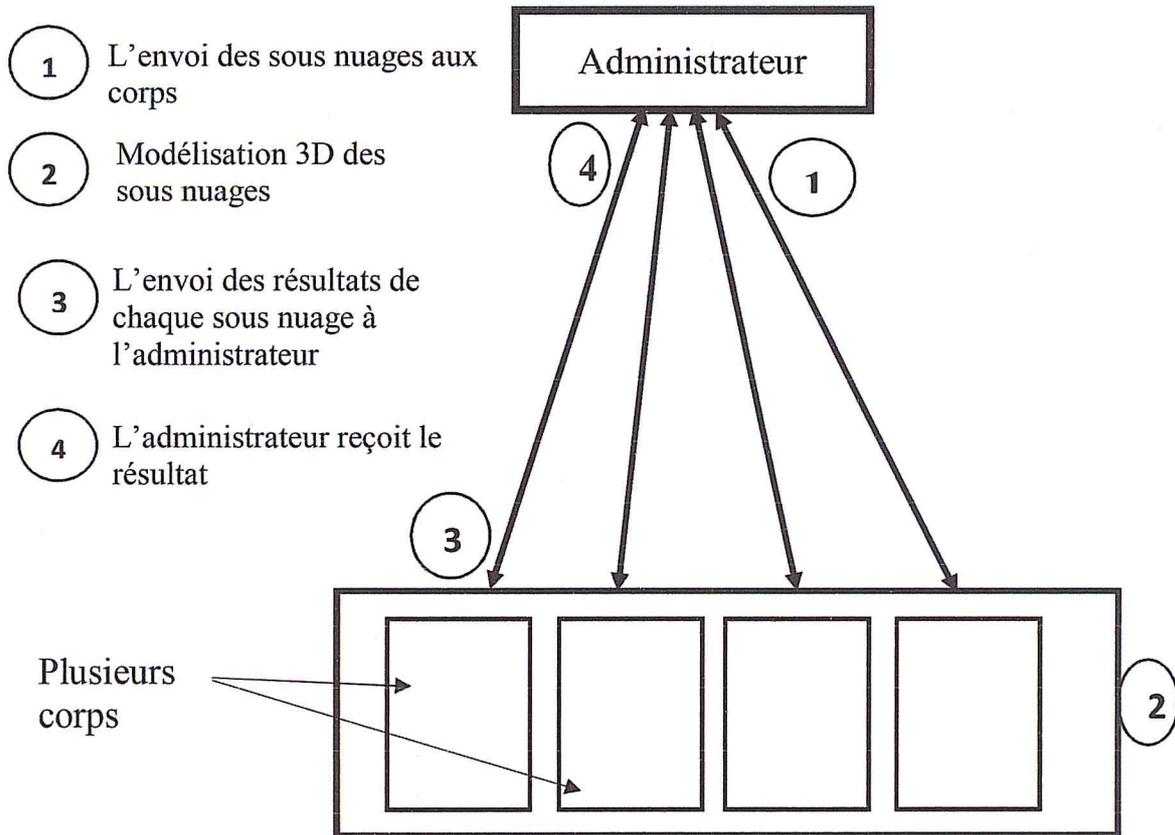


Figure 2.6 : Architecture de la distribution des sous nuages sur plusieurs corps

IV.3 Approximation des surfaces des sous nuages

L'approximation des surfaces gauches est une tâche complexe, Plusieurs approches sont utilisées pour résoudre cette complexité, les plus importantes utilisent la triangulation de Delaunay 3D. Mais le choix est porté sur l'inspiration de l'algorithme destruction et construction [19]. Cet algorithme est une des méthodes de calcul de la triangulation Delaunay, il existe déjà en 2D et nous l'avons translaté vers la 3D, en utilisant l'algorithme incrémental pour l'insertion des points la (Figure 2.7) montre l'organigramme d'approximation des surfaces.

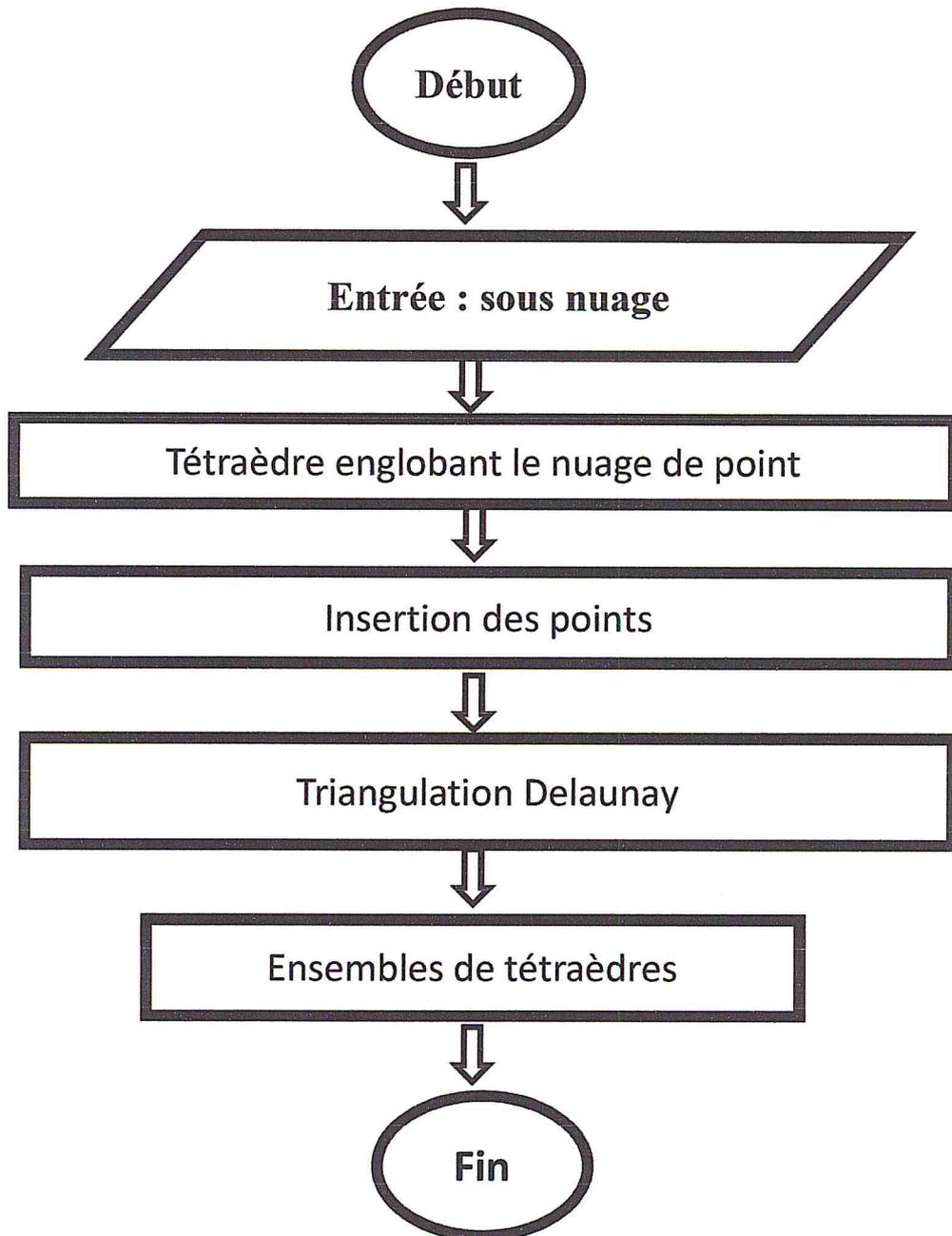


Figure 2.7: Organigramme de l'approximation par des tétraèdres

Chapitre2 : SOLUTIONS PROPOSEES

L'organigramme de l'approximation par des tétraèdres montré ci-dessus, présente plusieurs sous tâches, dans la section suivante nous allons détailler chacune des sous tâches.

IV.3.1 Création du tétraèdre englobant

C'est une sous tâche qui permet la génération d'un tétraèdre englobant l'ensemble des points. Dans notre travail nous allons suivre le même processus de reconstruction sur les sous nuages obtenus après la subdivision comme le montre l'organigramme suivant (Figure 2.8) :

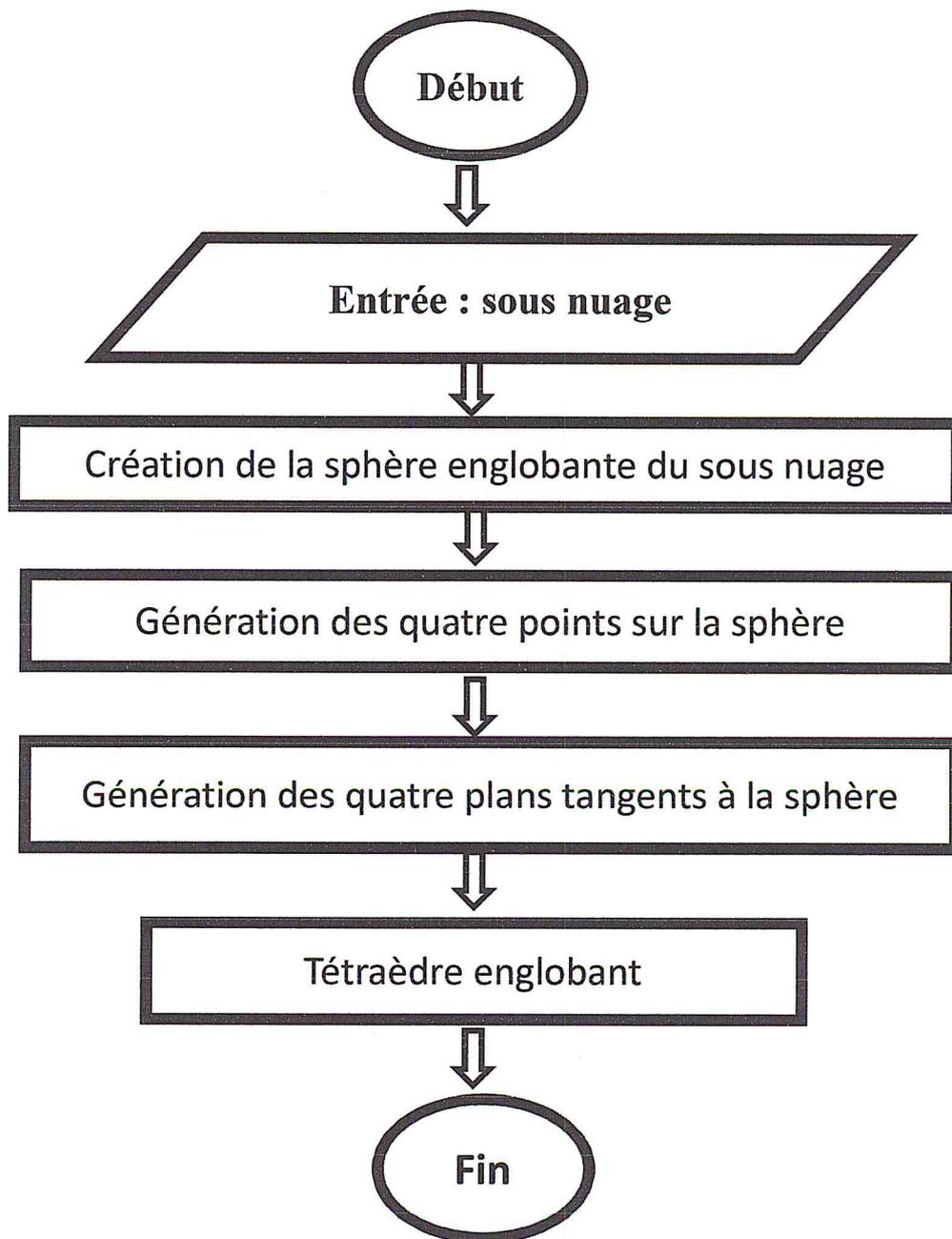


Figure 2.8 : Organigramme du tétraèdre englobant

Chapitre2 : SOLUTIONS PROPOSEES

Afin d'obtenir le tétraèdre englobant nous avons utilisé plusieurs fonctions et méthodes basées sur des calculs mathématiques, ces méthodes sont résumées dans les étapes suivantes :

- a) Création de la sphère englobante du sous nuage, cette méthode est réalisée par :
 - Le Calcul des coordonnées du centre de la sphère.
 - Le Calcul du rayon de la sphère qui est défini par la distance maximale entre le centre de la sphère et les différents points d'extrémités du sous nuage (X_{\min} , X_{\max} , Y_{\min} , Y_{\max} , Z_{\min} , Z_{\max}).
- b) Génération des quatre points sur la sphère qui permettent de définir les plans tangents en utilisant les coordonnées sphériques.
- c) Génération des quatre plans tangents sur la sphère passant par les points définis précédemment.
- d) Création du tétraèdre englobant à partir des points qui sont obtenus par la méthode du pivot de gauss en faisant l'intersection entre les différents plans.

Une fois le tétraèdre englobant est créé nous passons à l'insertion des points

IV.3.2 Insertion des points

Il existe plusieurs techniques pour l'insertion des points, nous avons utilisé l'algorithme incrémental qui consiste à insérer les points un par un. Cette méthode permet d'insérer les points soit d'une manière aléatoire ou séquentielle, dans cas de notre projet nous avons suivis les deux modes d'insertion [7].

IV.3.3 Triangulation de Delaunay

A chaque point inséré, une triangulation doit être faite, cette triangulation doit vérifier le critère de la sphère vide de Delaunay .pour cela nous allons suivre des étapes qui sont basées sur des méthodes de calcul mathématiques (Figure 2.9) :

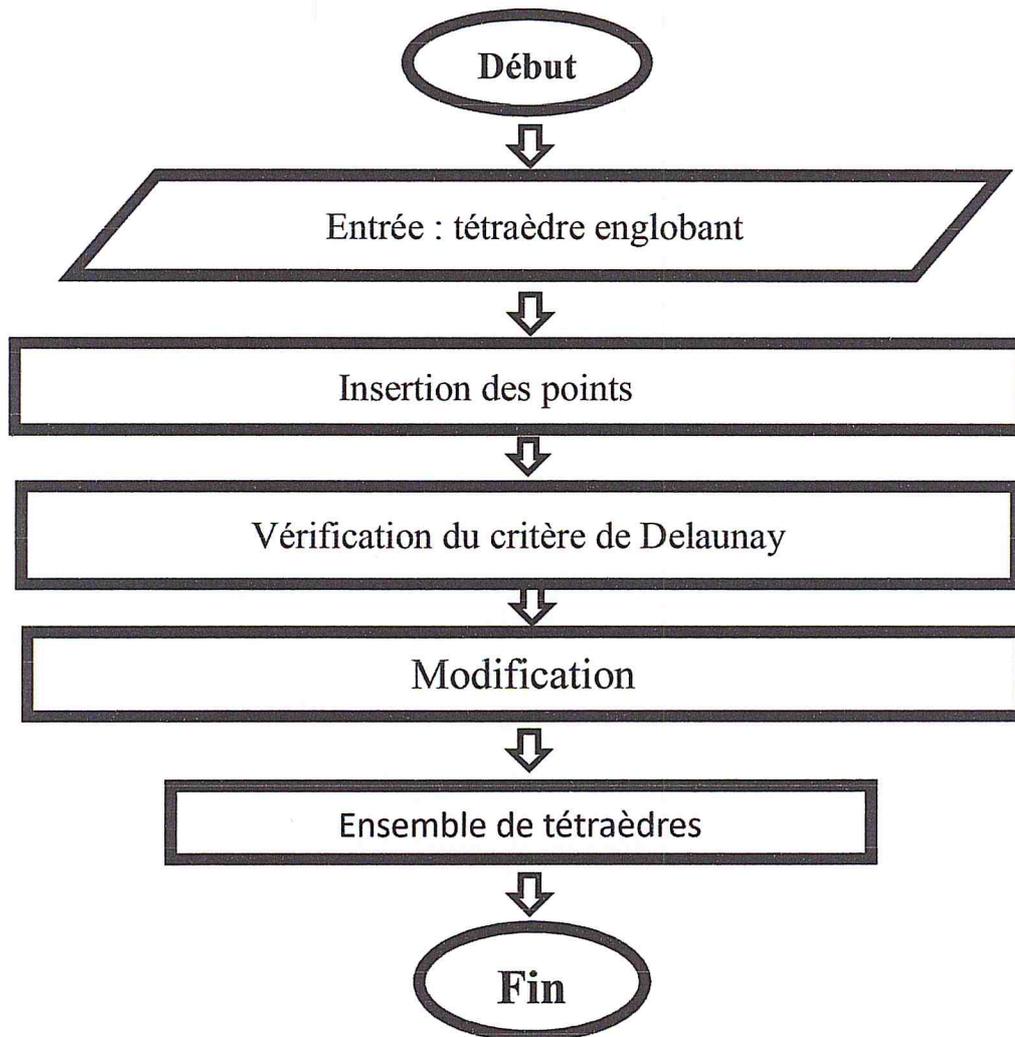


Figure 2.9 : Organigramme de la triangulation Delaunay

1. Pour chaque tétraèdre une création d'une sphère circonscrite est nécessaire :
 - Calcul du centre de la sphère : le centre est calculé par l'intersection des trois plans médiateurs du tétraèdre.
 - Calcul du rayon de la sphère : est défini par la distance entre le centre de la sphère et un des sommets du tétraèdre.
2. Insertion des points un par un (Figure 2.10)
3. Vérification pour chaque tétraèdre le critère de Delaunay.
4. Modification pour chaque tétraèdre ne vérifiant pas le critère de Delaunay :

Chapitre2 : SOLUTIONS PROPOSEES

- Détermination de tous les tétraèdres dont la sphère circonscrite contient le point ajouté.
 - Stockage de toutes les faces des tétraèdres dont la sphère circonscrite contient le point ajouté.
 - Suppression de faces dupliquées.
 - Suppression dans tétraèdres.
 - Création des nouveaux tétraèdres à partir du point ajoutée et les faces stockés des tétraèdres supprimés.
 - Ajout des nouveaux tétraèdres a la liste des tétraèdres.
5. Suppression des tétraèdres contenant au plus un sommet du tétraèdre englobant pour avoir la triangulation exacte du modèle.

IV.3.4 Génération du modèle STL

Le modèle STL est une représentation surfacique de l'objet, car il permet la définition de la peau extérieure du modèle tant dit que la triangulation réalisée est une triangulation volumique qui représente tout l'objet en profondeur. Pour cela nous allons suivre quelque étape présentée dans l'organigramme suivant (Figure 2.10) :

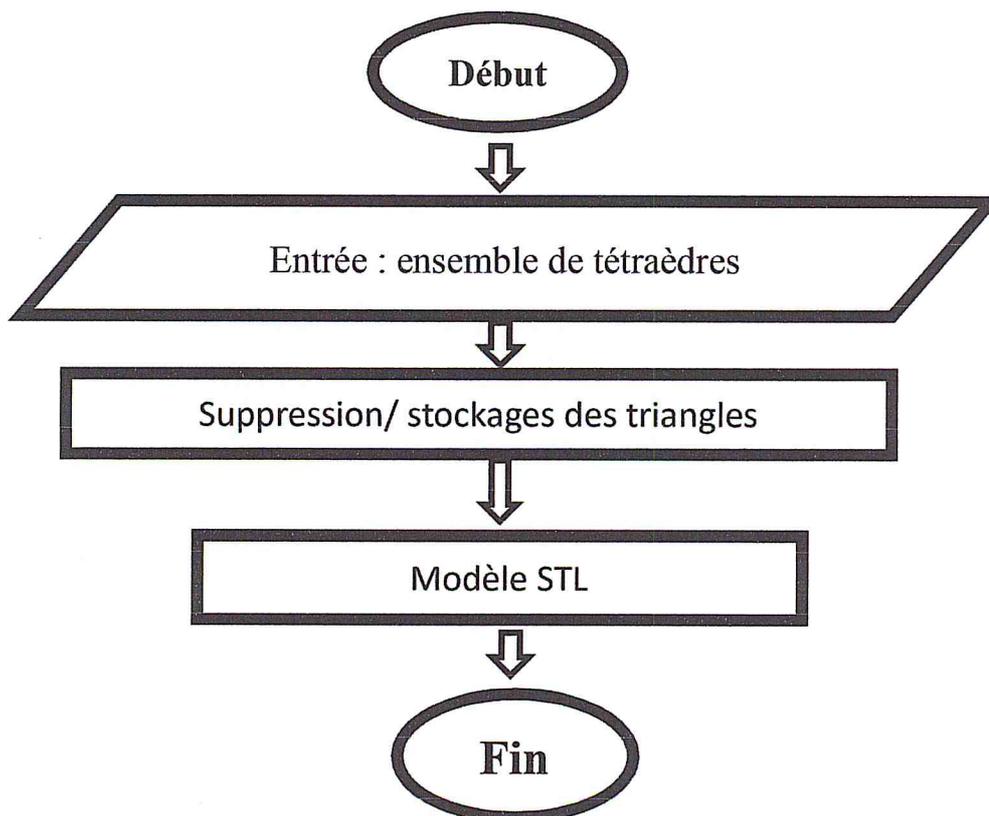


Figure 2.10 : Organigramme de la génération du modèle STL

Afin d'obtenir la surface désirée de l'objet ou le modèle STL, nous allons garder uniquement les faces qui n'ont pas de voisins.

Pour cela nous allons suivre les étapes suivantes :

1. Suppression de toutes les faces des tétraèdres qui ont au moins un voisin.
2. Stockage des faces triangulaires qui n'ont pas de voisin.
3. Visualisation des faces triangulaires (modèle STL).

V. Conclusion

Dans ce chapitre, nous avons défini notre système en commençant par la problématique, où nous avons constaté que le processus de reconstruction en 2D présente des problèmes de précisions, et de lenteur du temps de reconstruction, et ne permet pas d'obtenir le modèle STL.

Dans ce contexte la, nous avons pu amélioré le processus de reconstruction et le temps qu'elle prend, en commençant par détailler toutes les étapes(lecture du nuages de points, subdivision et distribution du nuage de point, approximation des sous nuages , modèle STL), et en donnant des architectures représentatives et des organigrammes de conception, et à travers ça, nous avons su faire une reconstruction 3D à partir d'un nuage de point quelconque et aussi l'optimisation du temps..

Dans le chapitre suivant, nous allons présenter l'implémentation des solutions proposées de notre application ainsi que la phase de tests et validation.

CHAPITRE III
IMPLEMENTATION
ET
TEST-VALIDATION

I. Introduction

Dans le but d'optimiser le temps de traitement, de concevoir et d'automatiser la reconstruction des surfaces gauches à partir d'un nuage de points, une application logicielle graphique et interactive a été développée pour l'équipe de CFAO.

Après l'étude de conception, de proposition des solutions, ainsi que les objectifs fixés dans le chapitre précédent, nous passons maintenant à l'étape d'implémentation. Dans ce chapitre nous allons évoquer les techniques suivies pour la réalisation de notre application, ainsi que la visualisation de chaque approche, ou réside son importance. Et nous l'avons divisé en deux parties.

La première partie concerne l'implémentation de l'application, et la description des principaux algorithmes développés et les démarches suivis.

La deuxième partie permet de présenter les différents tests effectués et la validation du logiciel, par rapport au cahier des charges, aussi pour faire les corrections nécessaires en cas de problèmes qui peuvent apparaître pendant l'exécution.

L'implémentation a été faite en utilisant un langage évolué doté d'une interface graphique qui facilite à l'utilisateur la manipulation et l'exécution des différentes tâches.

II. Implémentation

Pour la réalisation des solutions proposées, nous avons utilisé le C++ Builder et la plateforme Windows 8.1. Complémentaire, Ce travail est un ensemble de fenêtres Windows qui permettent à l'utilisateur d'interagir avec l'application et de visualiser tous les objets manipulés via la bibliothèque graphique OpenGL.

II.1 Langage de programmation

Le langage C++ est actuellement l'un des langages les plus utilisés dans le monde, aussi bien pour les applications scientifiques que pour le développement des logiciels. En tant qu'héritier du langage C, le C++ est d'une grande efficacité. Mais il a en plus des fonctionnalités puissantes, comme par exemple la notion de classe, qui permet d'appliquer les techniques de la programmation-objet [22].

II.2 Environnement de programmation Builder c++

C'est un environnement de développement basé sur C++ proposé par Borland. Tourné vers le développement rapide d'applications sous Windows. En un mot, C++ Builder permet de réaliser de façon très simple l'interface des applications, et de relier aisément le code utilisateur aux événements Windows, quelle que soit leur origine (souris, clavier, événement système, etc.) [23].

II.3 OpenGL

OpenGL (Open Graphics Library) est une bibliothèque graphique très complète qui permet aux programmeurs de développer des applications 2D et 3D assez facilement [24].

II.4 Présentation des fenêtres

Notre application logicielle se compose de plusieurs fenêtres, dans ce chapitre nous allons présenter toutes les fenêtres que nous avons réalisé.

II.4.1 Fenêtre principale

Cette fenêtre est la première fenêtre (Figure 3.1) qui apparait après l'exécution de l'application, elle comprend deux parties :

1. **Barre du menu principal :** La barre de menu principal contient les travaux précédents de l'équipe CFAO, comme notre application est intégrée dans tout le projet on va présenter seulement ce qui s'insère dans notre travail qui est un seul bouton : **la reconstruction 3D**.
2. **Scène de visualisation :** permet l'affichage et la visualisation des résultats obtenus dans un plan cartésien tridimensionnel.

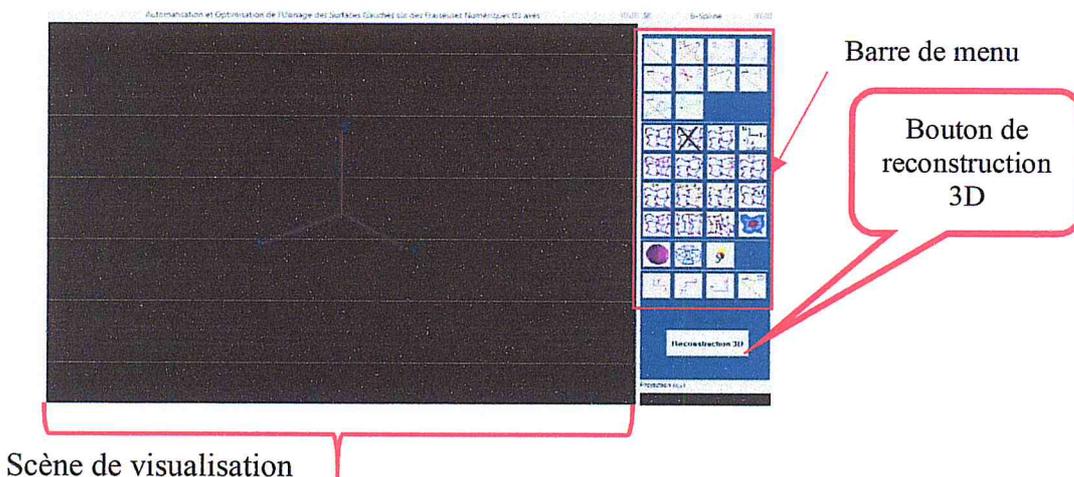


Figure3.1: Fenêtre principale

II.4.2 Fenêtre de «reconstruction 3D»

Pour accéder à la partie que nous avons réalisée, et intégrée dans l'application, nous pouvons cliquer sur le bouton reconstruction 3D.

Pour lancer l'interface du logiciel développés, qui contient une seule fenêtre, qui composée de plusieurs onglets représentant la séquence d'actions à mener pour aboutir au résultat final (Figure3.2).

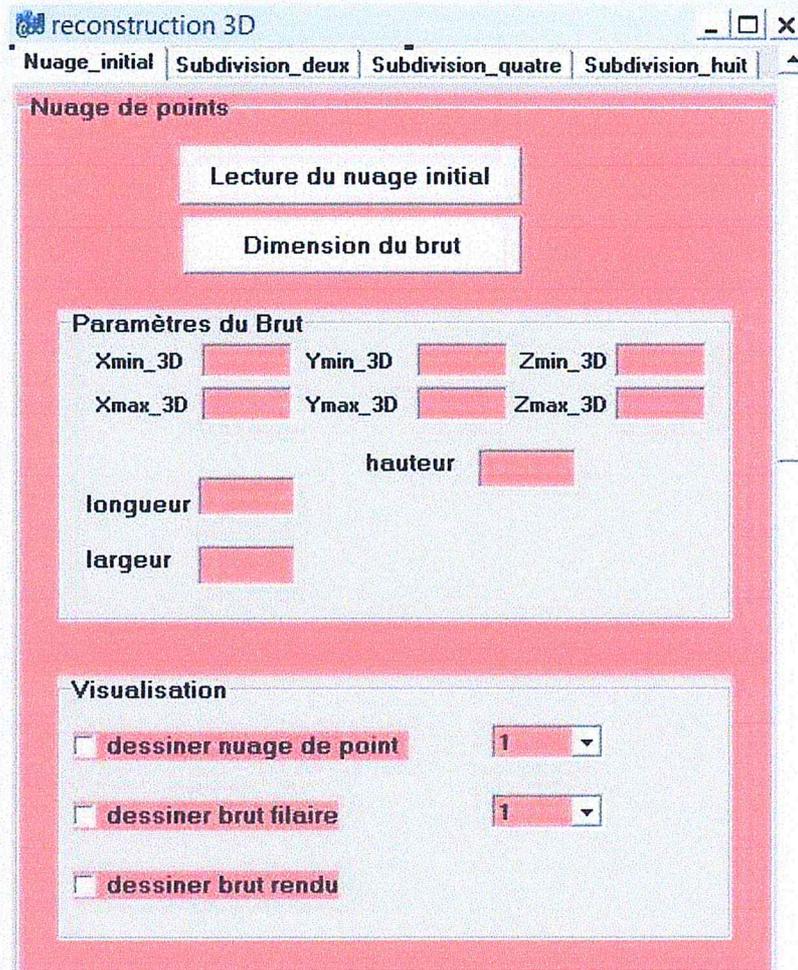


Figure 3.2 : Fenêtre de reconstruction 3D

1. Implémentation des tâches proposées

Comme nous l'avons cité dans le chapitre précédent le processus de reconstruction 3D contient plusieurs tâches qui permettent de définir le chemin à tracer pour arriver au résultat final ces tâches sont :

- Lecture du fichier nuage de point.

- Subdivision du nuage de points.
- Approximation des surfaces des objets complexes.
- génération du modèle STL.

1.1 Lecture du fichier nuage de point

Dans Le premier onglet « nuage_initial » un bouton est consacré à la lecture du nuage de points introduit par l'utilisateur. Pour cela l'utilisateur est invité à cliquer sur le bouton **lire nuage de points**. (figure3.2). les fonctions utilisés sont exprimés par l'algorithme(1)

- **Pseudo Algorithme (1) de lecture du fichier nuage de points :**

Entrée : Fichier de nuage de point

- Création d'une classe « point 3D » définie par trois variable X, Y et Z.
- Classification de données du fichier : un point est récupérer à partir de trois lignes du fichier.
- Stockage de chaque point dans un tableau dynamique de type point_3D
Tab_nuage..
- Vérification des points double : une variable booléenne « existe » est initialisée à false pour le point à ajouter.
- Si existe égal à true, nous passons au point suivant.
- si existe égal à false nous allons insérer le nouveau point dans le tableau avec la fonction push_back

L'utilisateur est invité à cliquer sur le bouton **dimension brut** pour le calcul des dimensions du brut (x_min_3D, x_max_3D, y_min_3D, y_max_3D, z_min et z_max). L'algorithme (2) présente les étapes suivies.

- **Pseudo Algorithme (2) calcul des dimensions de la boite englobante :**

Entrée : nuage de points

- Initialisation des coordonnées d'extrémités par le premier point ;
- Parcours du tableau point par point ;
- Recherche des cordonnées d'extrémité
- Calcul de la longueur, la Largeur et la hauteur

Sortie : point d'extrémités et dimensions du brut.

1.2 Subdivision du nuage de points

1.2.1 Subdivision en deux sous nuages

La subdivision peut être faite par plusieurs manières, comme nous l'avons cité dans le chapitre précédent, tous les sous nuages passent par le même processus de reconstruction, pour cela chaque onglet définit un type de subdivision(en deux, en quatre, en huit), et est composé de plusieurs blocs, ou chacun est un ensemble de boutons qui font appels à des algorithmes et des fonctions qui sont décrit par la suite :

Par exemple le deuxième onglet (Figure 3.3) de type subdivision en deux, l'utilisateur est invité à :

- a. Choisir le mode de subdivision selon (l'axe X, l'axe Y).
- b. Diviser le nuage de point en deux en cliquant sur le bouton **subdiviser_nuage_deux**.

Le bouton `subdiviser_nuage_deux` fait appel des fonctions qui sont exprimé par le pseudo algorithme (3).

➤ **Pseudo algorithme de subdivision en deux (3) :**

Entrée : Tableau de nuage de point

- Création d'une variable `x_centre` tel que
$$x_centre=(x_min_3D+x_max_3D)/2 ;$$
- Parcours du tableau `Tab_nuage`
- Pour chaque `point_3D` si la coordonnée `x` est supérieure à `X_centre`, le point sera ajouté au premier tableau `Tab_nuage1` et si elle est inférieure le point sera ajouté au deuxième tableau `Tab_nuage2`.

Sortie : deux tableaux de nuage de point

1.2.2 Calcul dimension

Une fois l'utilisateur clique sur le bouton de calcul dimension un appel à la fonction calcul dimension est exécuté (Figure 3.3) .Cette fonction a comme paramètres un des tableaux des sous nuages et les paramètres calculés représentant les extrémités du sous nuage (`x_min1`, `y_min1`, `z_min1`, `x_max`, `y_max1` et `z_max1`).

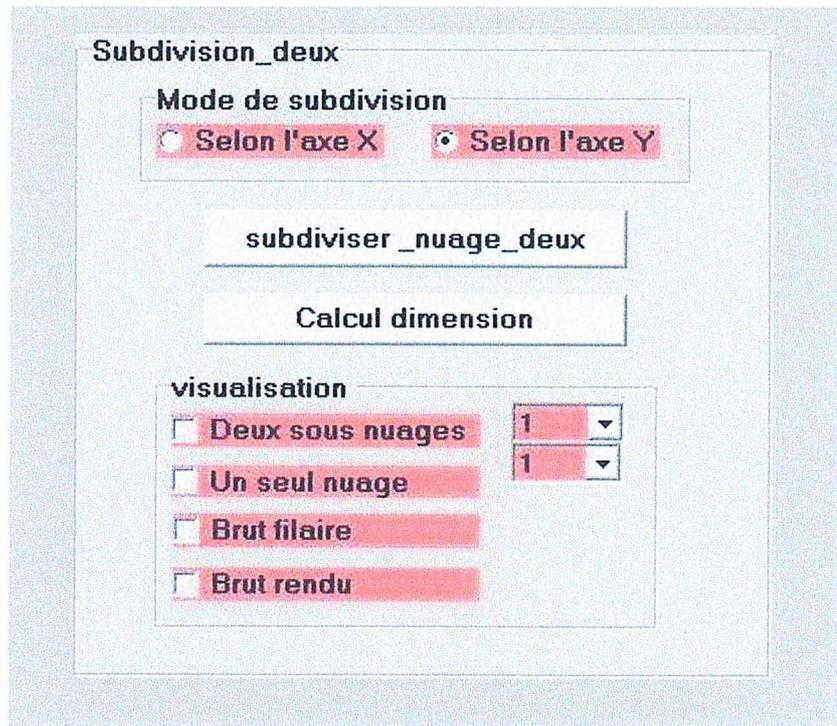


Figure 3.3 : Bloc subdivision en deux

1.3 Approximation des surfaces des objets complexes

L'approximation des surfaces est une tâche ardue et nécessite des connaissances en mathématique en particulier la géométrie dans l'espace, la majorité des algorithmes se base sur des fonctions récursives. Pour cela nous avons utilisé l'algorithme de de destruction et construction que nous l'avons translaté vers la 3D pour diminuer le problème des calculs intensifs.

1.3.1 Création du tétraèdre englobant

Cette tâche est assurée par le bouton tétraèdre_englobant (qui fait appel à des fonctions basée sur le pseudo algorithme (4)

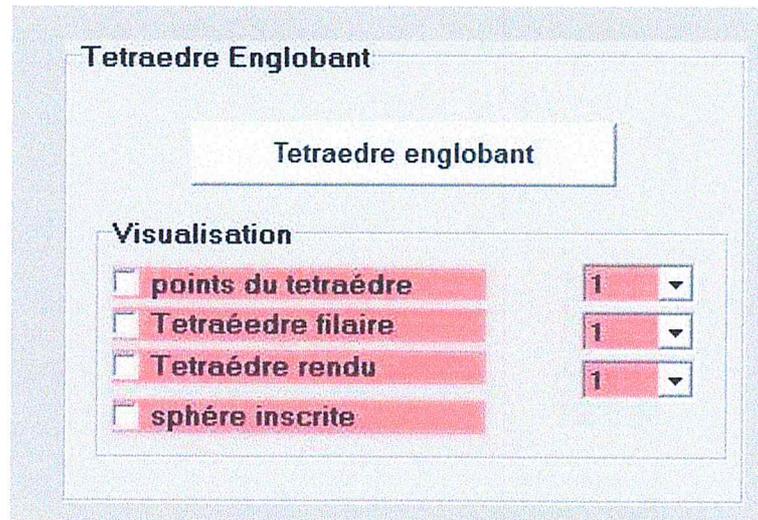


Figure 3.4: Bloc du tétraèdre englobant

➤ Pseudo algorithme du tétraèdre englobant(4) :

Entrée : Sous nuage de point

- Création de la sphère englobante (inscrite) : calcul du rayon de la sphère : La fonction `calculer_rayon` permet de calculer huit distances en faisant la combinaison de tous les points d'extrémités. Le rayon est défini par la distance maximale entre les huit distances.
 - a. Calcul de la distance : la distance est calculée comme suit : la fonction `a` comme paramètre trois coordonnées de type double
 - b. calcul du centre de la sphère (`x_centre1`, `y_centre1` et `z_centre1`).
- Génération des quatre points sur la sphère en utilisant coordonnées sphériques
- Génération des quatre plans tangents à la sphère passant chacun par un des quatre points défini sur la sphère.
- Intersection de trois plans détermine un point du tétraèdre
- Le tétraèdre est défini par quatre point alors quatre intersections de trois plans sont réalisées par la méthode du pivot de gauss à fin d'avoir les quatre point du tétraèdre englobant.
- Création d'un nœud tétraèdre pour stocker le tétraèdre dans la liste de tétraèdre (`liste_globale_tetra1`).

- Réalisation de la sphère circonscrite : chaque noued_tetraedre fait appel à la une fonction calculer_sphère().

Calcul du centre de la sphère : -le centre de la sphère est déterminé par l'intersection de trois plans médiateurs du tétraèdre ; un plan médiateur est un plan qui passe par le point de gravité de la face et le centre d'un plan. Nous avons utilisé le système de cramer basant sur la création de quatre matrices, une est principales et trois secondaires et le calcul du déterminant. Pour éviter le cas du déterminant nul on a développé vingt combinaisons à partir de quatre plans du tétraèdre.

- **Calcul du rayon de la sphère** : le rayon de la sphère circonscrite est la distance entre le centre de la sphère et un des sommets des tétraèdres car la distance est équidistante entre le centre et les quatre sommets

Sorite : tétraèdre englobant et sphère circonscrite

1.3.2 Insertion de point

Avant de passer à l'insertion des points, l'utilisateur est invité à choisir le mode de triangulation (manuel, automatique)

- 1) **Le mode automatique** : permet l'insertion des points d'une manière automatique, et aussi permet la visualisation final du résultat, sans être obligé d'insérer le point un par un et de faire la modification manuellement. (Figure 3.5) :

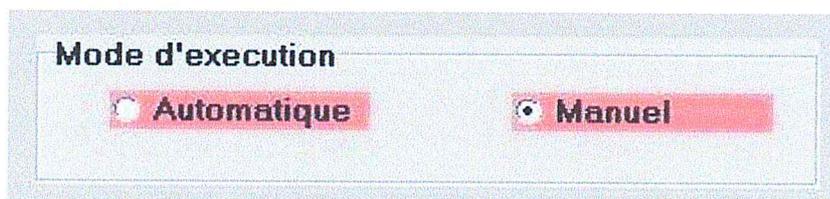


Figure 3.5 : Bloc de modes de triangulation

Une fois le mode automatique est choisi l'utilisateur aura droit à cliquer sur le bouton triangulation de Delaunay (Figure 3.6). Ce dernier permet l'insertion et la modification de tous les points en une seule fois par un appel automatique à la fonction d'insertion et de vérification modification.

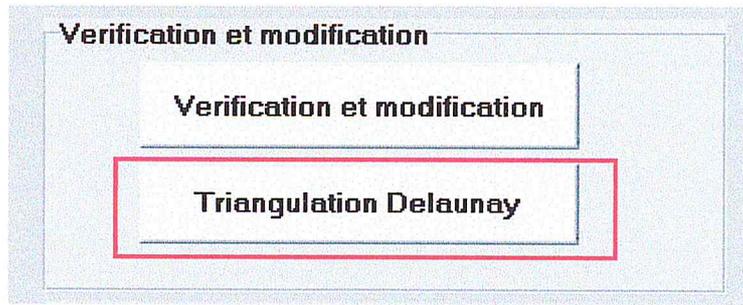


Figure 3.6 : Bouton de triangulation de Delaunay

- 2) Le **mode manuel** : permet l'insertion des points et la modification un par un ainsi la visualisation de la triangulation à chaque insertion (Figure 3.5). Il existe deux modes d'insertion (Figure 3.7).
- Aléatoire : permet l'insertion des points aléatoirement en utilisant random.
 - Séquentiel : permet l'insertion des point un par un en parcourant le tableau contenant les points à inséré.

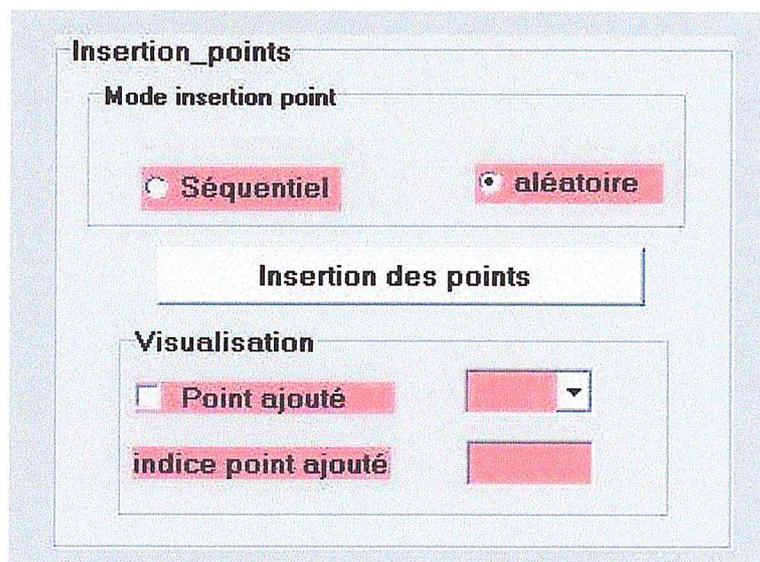


Figure 3.7 : Bloc insertion des points

Une fois l'utilisateur a choisi le mode manuel, il peut choisir le mode d'insertion de point (séquentiel, aléatoire).ensuite appuyer sur le bouton vérification et modification qui permet de vérifier le critère de Delaunay à chaque insertion. Les étapes suivis sont présentées dans le pseudo algorithme (6) (7).

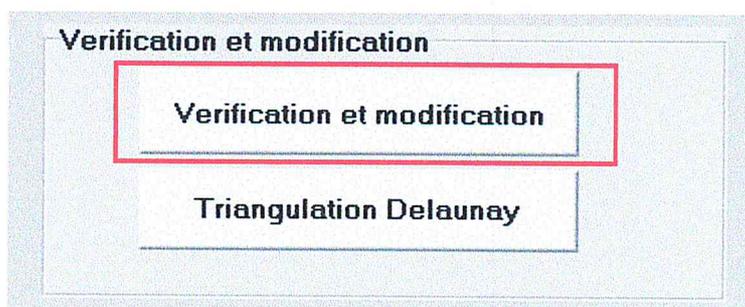


Figure 3.8: Bouton de vérification et modification

L'insertion des points a été réalisée en se basant sur le pseudo algorithme (5)

➤ **Pseudo algorithme d'insertion des points (5) :**

Entrée : sous nuage de point

- Parcours du tableau de nuage
- choix du mode de triangulation
- choix du mode d'insertion

Sortie : Points insérés

➤ **Pseudo Algorithme de vérification du critère de Delaunay(6) :**

Entrée : point ajouté

- Calcul de la distance entre le point ajouté et le centre de la sphère.
- Si la distance est supérieure au rayon de la sphère, le critère de delaunay est vérifié.
- Si la distance est inférieure au rayon de la sphère, le critère de delaunay n'est pas vérifié donc une modification est nécessaire.

Sortie : Etat de la triangulation

➤ pseudo algorithme de modification (7) :

Entrée : Triangulation de delaunay incorrecte

- parcours de la liste liste_globale_tetra1
- si une sphère d'un tétraèdre contient le point ajouté (elle ne respecte pas le critère de la sphère vide)
- sauvegarde des faces de ce tétraèdre dans un tableau Tab_face_commune
- parcours du tableau Tab_face_commune et suppression des faces dupliqués par la fonction erase()
- suppression de ce tétraèdre de la liste avec la fonction supprimer_tetraedre()
- Création de nouveau tétraèdre en faisant l'appel à une fonction crée_nouveau_tetraedre() et qui permet la création des tétraèdres à partir du point ajouté et les faces stockées dans le tableau.
- l'ajout des nouveaux tétraèdres dans la liste des tétraèdres

Sortie : Triangulation conforme au critère de Delaunay

1.3.3 Triangulation Delaunay finale

Après avoir eu la triangulation de Delaunay, une triangulation réelle du sous nuage est faite afin d'avoir la triangulation seulement de la partie qu'on veut reconstruire (Figure 3.9). Les étapes suivies sont expliqués dans le pseudo algorithme(8).

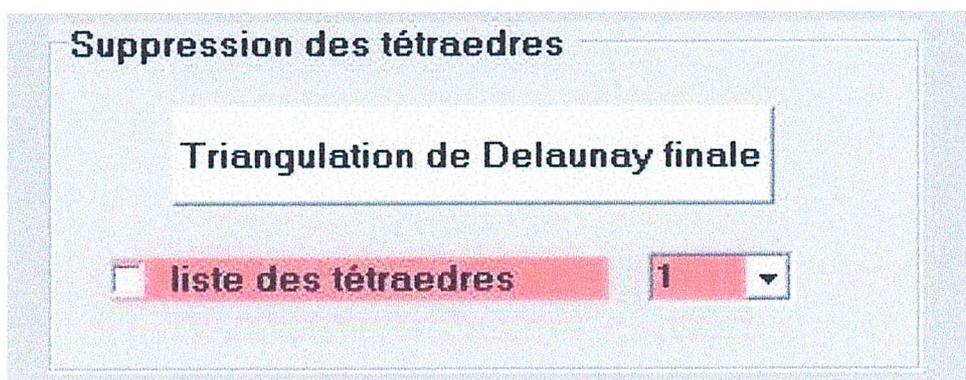


Figure 3.9 : Bouton de triangulation Delaunay

➤ pseudo algorithme de suppression des tétraèdres (8) :

Entrée : Triangulation dans le tétraèdre englobant

- parcours de la liste liste_globale_tetra1
- suppression de tous les tétraèdres ayant en moins un sommet en commun avec le tétraèdre englobant

Sortie : Triangulation du sous nuage (ensemble de tétraèdres)

1.4 Génération du modèle STL

Pour pouvoir générer un modèle STL nous avons implémenté une fonction `determiner_face_a_stocker`, Cette fonction retourne un booléen et prend en paramètres trois point de type `point_3D` (Figure 3.10) les étapes suivies pour cette tâche sont exprimé dans le pseudo algorithme.

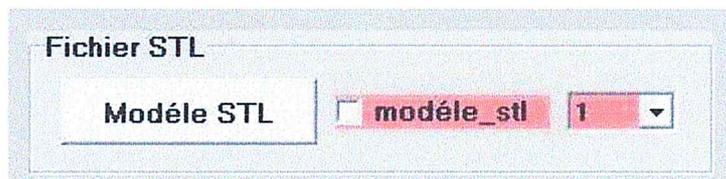


Figure 3.10 : Bouton de génération du modèle STL

➤ Pseudo algorithme du modèle (9) :

Entrée : ensemble de tétraèdres

- Création de la classe `triangle_stl`
- parcours de la liste `liste_globale_tetra1`
- création du tableau `Tab_trinagle-stl` de type `tringle_stl`
- initialisation d'une variable booléenne `existe` à `false`
- vérification pour chaque tétraèdre de la liste s'il a des voisins
- si un tétraèdre a en moins un voisin dans une face la variable `existe` devient à `true`
- si `existe=false` donc la face du tétraèdre n'a aucun voisin
- calcul de la normale de la face.
- stockage de la face dans le tableau `Tab_trinagle-stl`.

Sortie : modèle STL

III. Test des résultats

Après avoir présenté l'implémentation de notre application logicielle, nous passons à la dernière étape qui est l'étape des tests et validations.

Cette phase de tests consiste à tester les approches proposées. Pour cela, nous allons considérer un exemple de nuage de points complet, puis le subdivisé en deux, ce nuage est issu d'une pièce de poterie pour la réalisation d'un moule d'injection de plastique, cette pièce est digitalisée par une machine de mesure tridimensionnelle (MMT).

Nous allons montrer aussi l'exécution et la visualisation de trois exemples de sous nuages subdivisés en deux ensuite, en huit selon des axes différents, et les différentes phases pour arrivera la fin à un modèle STL, Cependant on peut distinguer l'optimisation du temps de traitement et la qualité de la triangulation par à rapport au différentes subdivisions.

Pour la validation nous allons faire une étude entre la modélisation de deux dimensions et la modélisation en trois dimensions et une comparaison entre les différentes méthodes et approches que nous avons suivies.

III.1 Exemple de test

Pour tester notre application, Nous allons faire les tests sur trois nuages de points, issus des pièces réelles digitalisées, Pour cela nous montrons le traitement principal réalisé pour chaque nuage de points.

1. Premier nuage de point

Ce nuage est issu d'une pièce de poterie destinée à être réalisée en grande quantité dans un moule d'injection en plastique. Nous avons commencé par traiter tout le nuage, Ensuite nous l'avons subdivisé en deux pour les traiter séparément dans le but de minimiser le temps de traitement de la triangulation de Delaunay.

Tout le processus de la génération de la triangulation de Delaunay appliqué sur cet exemple, est décrit comme suit :

a. Traitement du nuage initial

▪ Etape1

- Ouverture de l'application et choix du fichier de nuage de point

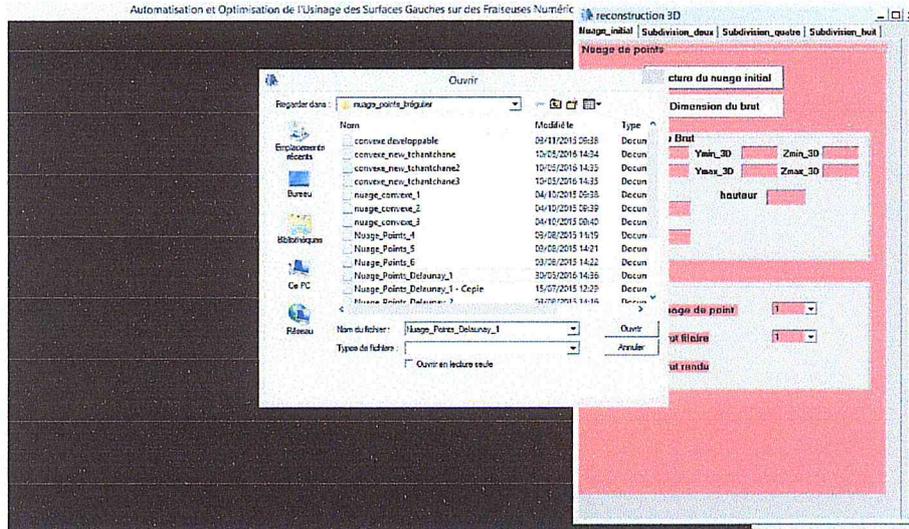


Figure 3.11 : Lecture du nuage de point

▪ Etape 2

- Nuage de point initial obtenu après la numérisation et la lecture du fichier

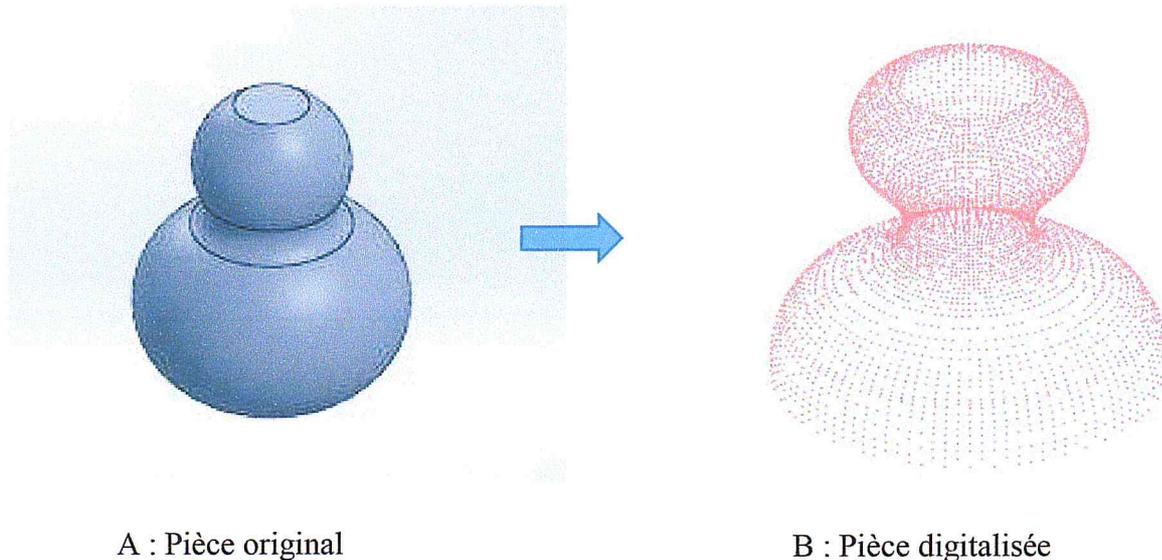


Figure 3.12 : Lecture du nuage de point

▪ Etape 3

- Calcul des dimensions de la boîte englobante

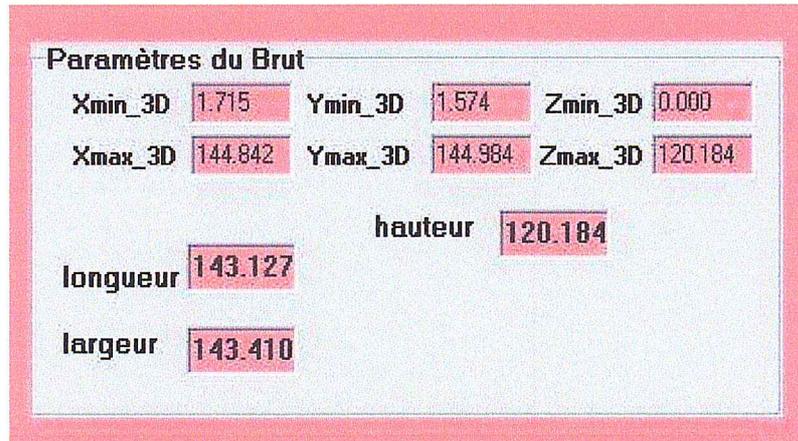


Figure 3.13 : Calcul des dimensions du brut

▪ Etape 4

- Boîte englobante filaire

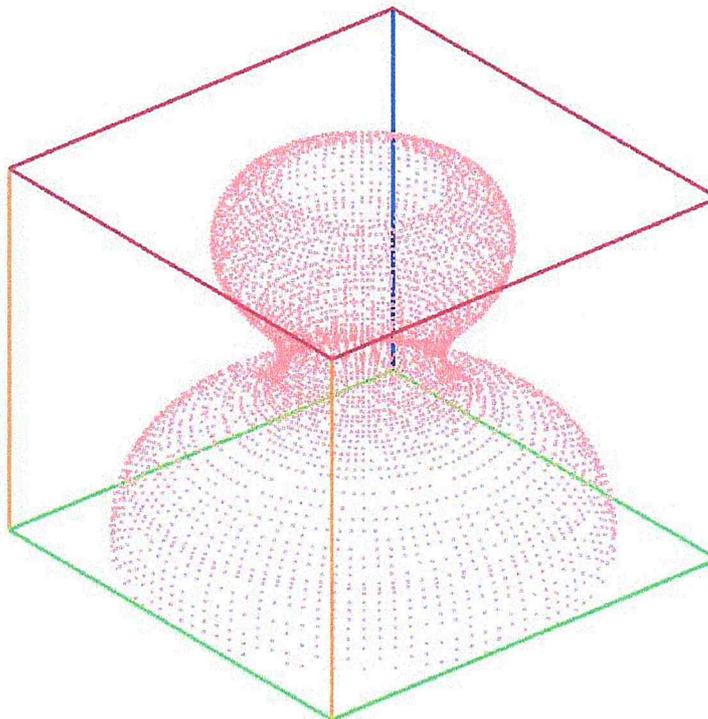


Figure 3.14 : Calcul des dimensions du brut

- Etape 5
 - Boîte englobante rendue

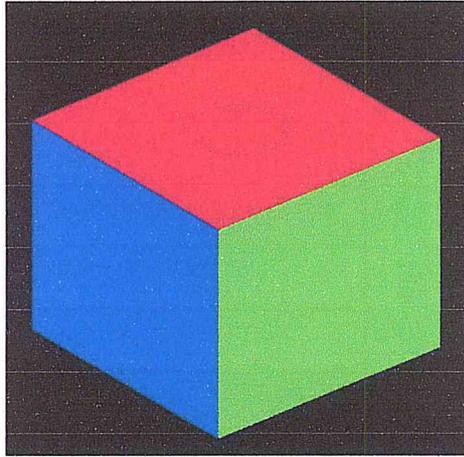


Figure 3.15 : Brut rendu

- Etape 6
 - Quatre Points du tétraèdre englobant

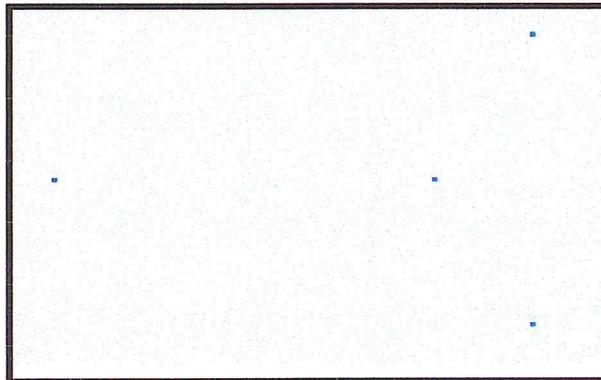


Figure 3.16: Points du tétraèdre englobant

- Etape 7
 - Sphère inscrite englobante du nuage de point et le tétraèdre englobant

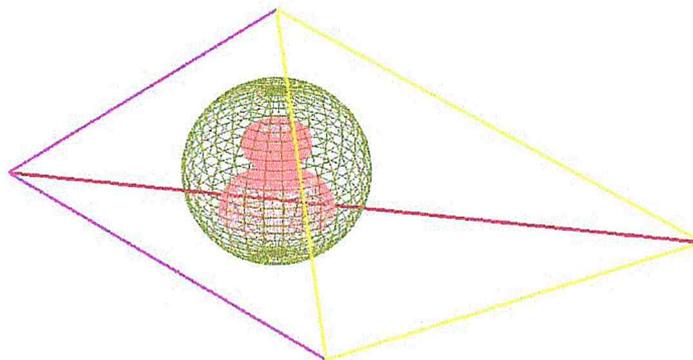


Figure 3.17 : Sphère inscrite du tétraèdre englobant

- Etape 8
 - Triangulation de Delaunay du nuage initial

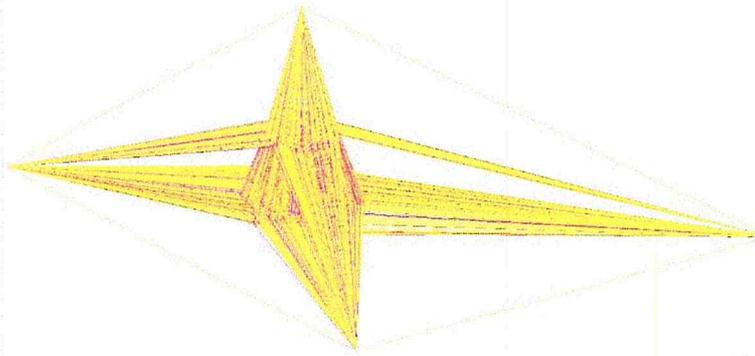


Figure 3.18 : Triangulation de Delaunay

- Etape 9
 - Triangulation de Delaunay finale



Figure 3.19 : Triangulation Delaunay finale

- Etape 10
 - Modèle STL



Figure 3.20 : Modèle STL

b. Traitement des sous nuages

Le nuage initial est subdivisé en deux selon l'axe « x », chaque sous nuage est stocké dans un tableau, le choix de la subdivision n'est pas aléatoire mais c'est fait pour aboutir à une meilleure subdivision par rapport à la forme de la pièce numérisée.

▪ Etape 1

- Subdivision en deux sous nuages

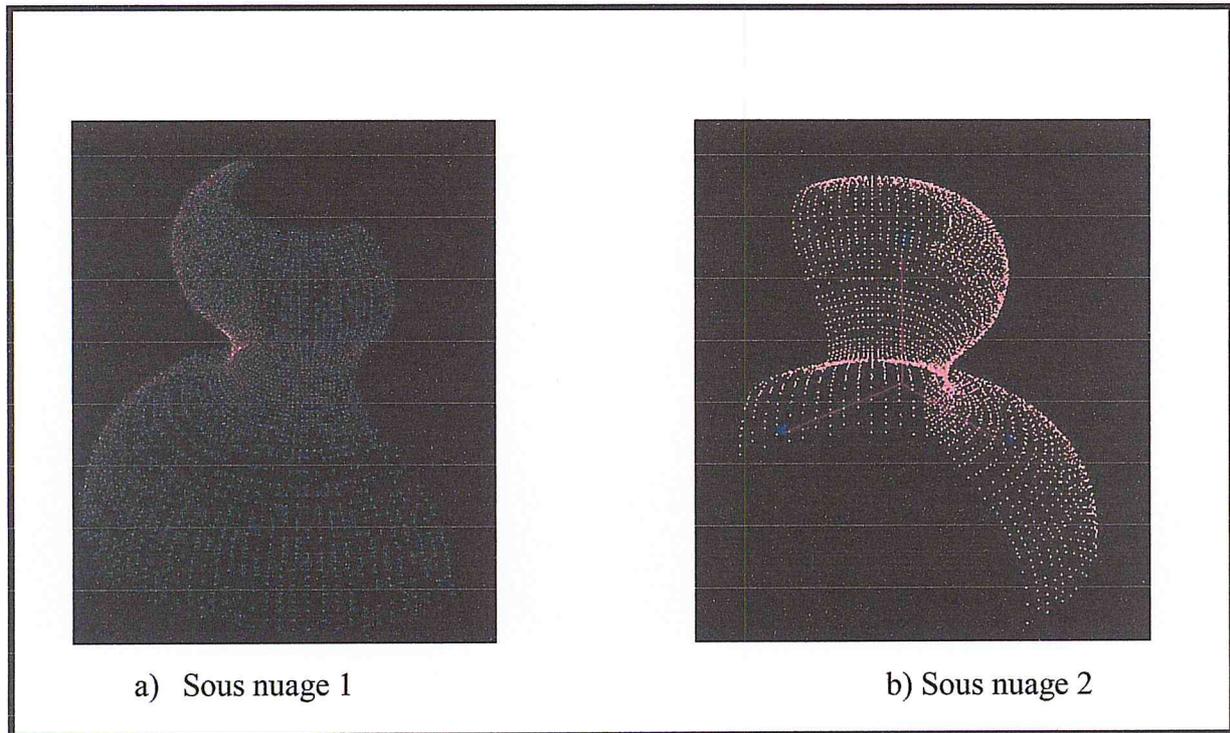


Figure 3.21 : Nuage de point subdivisé en deux

▪ Etape 2

- Triangulation de Delaunay finale des deux sous nuages

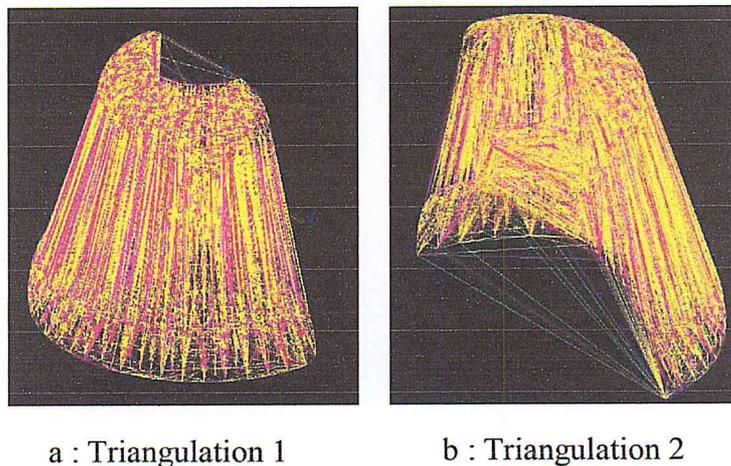


Figure 3.22 : Triangulation Delaunay des deux sous nuages

- **Etape 3**
 - Génération du modèle STL des deux sous nuages

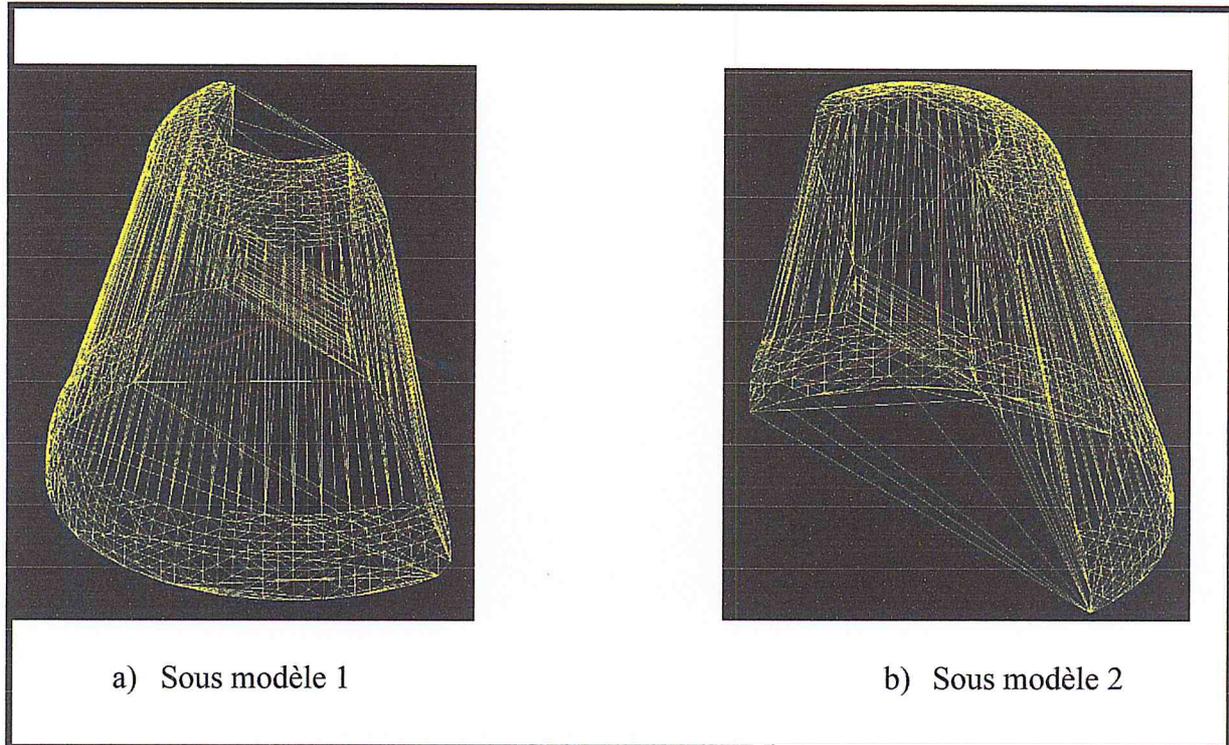


Figure 3.23 : Modèle STL de chaque sous nuage

2. Deuxième nuage de point

Ce nuage est issu de la digitalisation d'une pièce mécanique en acier, nous avons commencé par traiter tout le nuage ensuite nous avons subdivisé ce nuage en huit sous nuages selon l'axe des « y », dans cet exemple nous allons juste présenter les résultats obtenus.

a. Traitement du nuage initial.

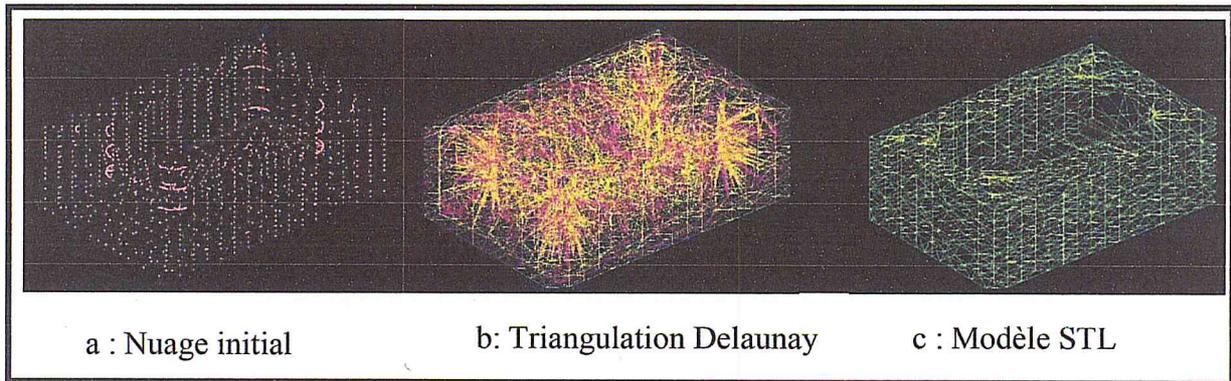


Figure 3.24 : processus de génération du modèle STL à partir d'un nuage de points.

b. Traitement d'un seul sous nuage subdivisé en deux de l'exemple précédent :

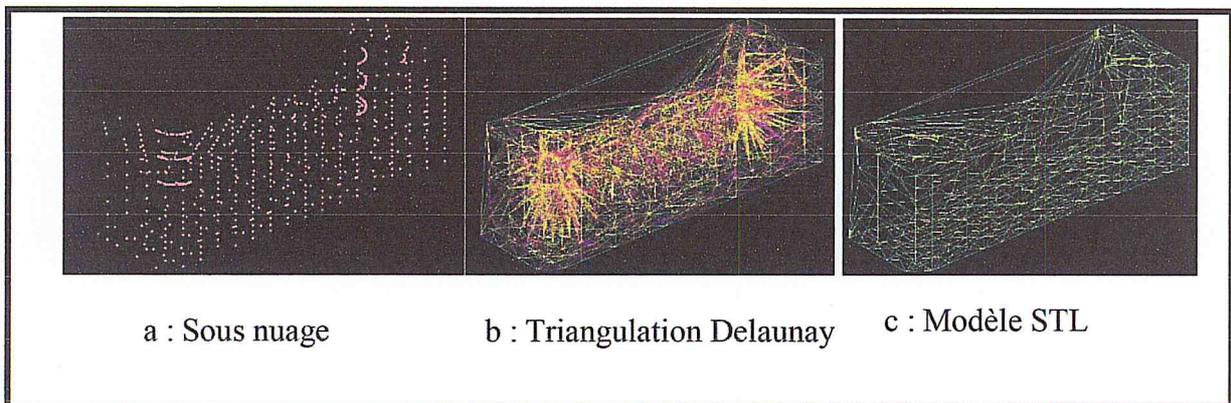


Figure 3.25 : Processus de génération du modèle STL à partir d'un nuage de points.

3. Troisième nuage de point

a. Traitement du nuage initial

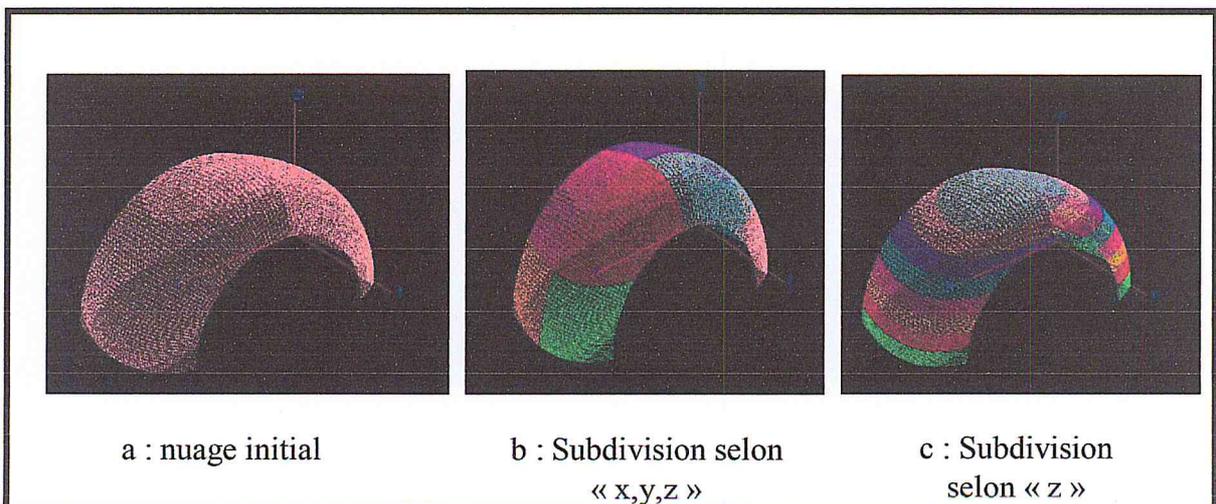


Figure 3.26 : Comparaison entre les différents modes de subdivisions

Chapitre3 : IMPLEMENTATION ET TEST-VALIDATION

Nous avons opté pour la subdivision, afin d'optimiser le temps d l'approximation, pour ce faire il faut considérer plusieurs critères comme le choix d'axe de la subdivision qui doit être étudié pour arriver au résultat souhaité et avoir une meilleure approximation la(Figure 3.26)montre que si une subdivision est faite par rapport à l'axe des z ,l'approximation de l'objet sera mauvaise car elle approxime deux morceaux non continus

III.2 Etude critique

Afin de montrer l'importance et l'utilité de la subdivision nous avons comparé les différents résultats obtenus avant et après la subdivision.

- Exemple du premier nuage

	Nuage initial	Sous nuage1	Sous nuage2
Nombre de points	5755	2953	2796
Nombre de tétraèdre	35230	17891	17075
Temps de triangulation	04 :50 :57	01 :27 :31	01 :15 :37

Tableau 1 : Résultat de test avant et après la subdivision

▪ Exemple du deuxième nuage

1. Subdivision en deux

Nuage Critères	initial	Sous-nuage 1	Sous-nuage 2
Nombre de points	1314	633	681
Nombre de tétraèdres	7849	3691	3000
Temps de triangulation	15min	30 secs	3min

Tableau 2 : Résultat de test avant et après la subdivision en deux

2. Subdivision en huit

Nuage Critères	initial	1	2	3	4	5	6	7	8
Nombre de points	1314	84	190	96	84	64	98	203	268
Nombre de tétraèdres	7849	7418	799	17971	9769	3275	17192	1701	999
Temps de triangulation	15min	40sec	30sec	2min 10sec	1min	19sec	2min	2min	30secs

Tableau 3 : Résultat de test avant et après la subdivision en huit

➤ comparaison

A partir de l'analyse des résultats obtenus (Tableau 1, 2,3) à la fin de la triangulation du nuage initial et des sous nuages, nous avons constaté que le temps de traitement du nuage initial est très grand comparant à la somme des temps de traitement des sous nuages.

IV. Validation de l'application logicielle

Pour valider l'application logicielle, un banc d'essai a été fait sur différents nuages de point obtenus de la numérisation des objets réels pour les reconstruire. Les tests sont effectués sur trois nuages, la validation est faite par rapport à plusieurs paramètres :

IV.1.1 Triangulation en deux et trois dimensions

bien que la triangulation en trois dimensions prend des temps de traitement énormes, car le temps d'exécution peut varier des heures à des jours en prenant en facteur la taille du nuage de point, la capacité de la mémoire et la puissance du processeur. Mais son principal avantage est qu'elle présente des améliorations dans la précision. Parce qu'elle permet la définition l'objet avec ses dimensions réelles. ce qui mène à une bonne approximation de l'objet. aussi elle permet la génération d'un modèle STL qui est un circuit fermé, permet la représentation surfacique de la peau extérieure de l'objet à partir d'un volume et par la suite la génération du fichier STL à partir du modèle. Cela était impossible avec la triangulation en deux dimensions.

Critères de comparaison	2D	3D
Temps de triangulation	Très important	Important
Précision	Mauvaise	Bonne
Qualité d'approximation	Médiocre	Meilleure
Génération du fichier STL	Impossible	possible

Tableau 4 : Comparaison entre la triangulation de deux et trois dimensions

IV.1.2 Modes de subdivisions

Pour remédier au problème des temps de traitement importants, une série de méthodes sont implémentées qui consistent en la subdivision du nuage obtenu en des sous-nuages. La forme des objets diffère d'un objet à un autre ce qui rend impossible de subdiviser les nuages de la même façon, pour cela une création de trois modes a eu lieu pour mieux représenter le nuage et l'approximer à une qualité meilleure.

	Modes de subdivision			
	Selon x	Selon y	Selon z	Selon x,y,z
Premier nuage	×			
Deuxième nuage		×		
Troisième nuage				×

Tableau 5 : Mode de subdivision

IV.1.3 Modes de triangulation

dans la réalisation de l'application nous avons paramétré, en mettant en évidence deux modes de triangulation .le premier est manuel ,il permet l'insertion du point et la modification un par un donc l'exécution de la triangulation pas à pas .Ce mode est bénéfique dans le sens de la vérification et la visualisation de la triangulation à chaque insertion ,ce mode présente un inconvénient qui est l'augmentation du temps de traitement ,pour cela un deuxième mode est réalisé, qui permet la triangulation automatique à chaque insertion .et par la suite la visualisation de la triangulation finale une fois le processus est terminé .

	Automatique	manuel
Visualisation Après chaque insertion		×
Rapidité	×	

Tableau 6 : Mode de triangulation

IV.1.4 Modes d'insertion

Pour insérer un point deux méthodes sont implémentées, une dite séquentielle, elle est avantageuse dans la détection des erreurs. L'autre est aléatoire ses avantage qu'elle est efficace pour sortir du voisinage du point inséré, aussi le temps de traitement est minime par rapport à la première méthode.

	aléatoire	séquentielle
Insertion au voisinage		×
Rapidité	×	

Tableau 7: Mode d'insertion

V. Conclusion

Dans ce chapitre, nous avons présenté les différentes étapes réalisées sur un nuage de point depuis sa lecture et jusqu'à la phase finale qui est le modèle STL, dans la partie des tests nous avons pris trois exemples de nuages de points qui différent dans la taille, le nombre de tétraèdres après la triangulation et le temps d'exécution.

CONCLUSION GENERALE

CONCLUSION GENERALE

Le travail réalisé au sein de la division productique et robotique du CDTA s'articule autour de développement d'une application, pour la reconstruction des surfaces gauches à partir d'un nuage de point quelconque, il s'agit précisément de la réalisation d'une approximation des nuages de points par des éléments géométriques (tétraèdres).

La première étape évoquée dans ce mémoire est l'étude bibliographique qui permet d'étudier en premier lieu les méthodes de conception et modélisation des surfaces gauches, Ensuite, d'étudier le processus de l'ingénierie inverse, et les différentes méthodes de reconstruction en appuyant sur la triangulation 3D de Delaunay.

Par la suite, nous avons proposé des architectures représentatives, et des organigrammes conceptuels, la solution proposée est composée de plusieurs tâches qui sont basées sur plusieurs approches reposées sur la triangulation Delaunay 3D.

A la fin, nous avons montré les différentes fonctions offertes par l'application logicielle développée suivi d'un test de validation.

Le résultat de notre travail est une application logicielle permettant :

- Subdivision du nuage de points en plusieurs morceaux pour pouvoir les distribuer sur différents corps pour avoir l'optimisation en temps d'exécution.
- Génération d'une triangulation 3D (ensemble de tétraèdres) qui approxime le mieux un nuage de points en utilisant la triangulation 3D de Delaunay.
- Génération d'un modèle STL à partir de d'un ensemble de tétraèdres obtenu de la triangulation de Delaunay.

Perspectives

En vue d'améliorer notre travail, tout travail n'est pas parfait, il est intéressant de rajouter quelques fonctionnalités à notre proposition comme :

- le développement de nouvelles méthodes de traitement pour améliorer la qualité de la triangulation comme la méthode de flip.
- La concaténation des sous nuages traités pour former l'ensemble de l'objet.

Le but de ce travail est de concevoir et d'implémenter une application distribuée pour la reconstruction des modèles 3D à partir d'un nuage de points quelconque, malheureusement, nous avons eu beaucoup de difficultés concernant l'intégration des bibliothèques dans l'environnement (MPI,...).

Ce projet nous a permis d'appliquer et approfondir nos connaissances théoriques et pratiques dans le domaine de développement d'une application complexe.

REFERENCES BIBLIOGRAPHIQUES

[1] Introduction au Reverse Engineering Titre In

http://www-igm.univ-mlv.fr/~dr/XPOSE2013/reverse_engineering/generalite.html.

[2] Céline ROUDET, Contribution de la reconstruction 3D à la compression de maillages surfaciques triangulaires, Rapport de stage – Master M2 Recherche Informatique, Université Claude Bernard Lyon 1, p. 13-14.

[3] Philippe Vanackère, LES MODÉLISATIONS GÉOMÉTRIQUES UTILISÉES DANS LES LOGICIELS DE DAO/CAO, revue, Lycée technique d'Armentières, Titre In

<http://www.epi.asso.fr/revue/dossiers/d12p123.html>, 2004

[4] Franck Rolland. Representation tridimensionnelle et reconstruction 3D a partir de coupes 2D. Modelisation et simulation. Université Joseph-Fourier - Grenoble I, 1991. Français. P.11-33

[5] Christine Potier, “Modélisation - Mastère Multimédia”, Ecole nationale supérieure des télécommunications, Janvier 2002.p.1-24.

[6] Moron, Mise en correspondance de données 3D avec modèle CAO: application a l'inspection automatique v. (1996, Décembre) p.17-19.

[7] Juan Andres Restrepo Specht. Modelisation d'objets 3D par construction incrementale d'un maillage triangulaire, dans un contexte robotique. Automatique /Robotique. Université Paul Sabatier-Toulouse III, 2005. Français. P.5-11.

[8] Bey Mohamed, Modelisation des courbes et des surfaces, Laboratoire de Robotique et d'Intelligence Artificielle Equipe Systèmes Robotisés de Production, Avril 2000. P.21-27.

[9] Maria, A. N. Optimisation de l'opération de finition des surfaces de formes libres par l'intégration de différentes formes d'outils et de stratégies d'usinage à partir d'un modèle STL . centre de developpement des technologies avancées (2011).p. 27-35.

[10] Rémi ALLÈGRE, These de doctorat, “Contributions à l'introduction de l'expressivité dans la reconstruction et l'édition de modèles 3D”, l'université Claude Bernard Lyon 1 decembre 2006, p.2-32.

[11] Mohanad Makki, Christophe Tournier, François Thiébaud, Claire Lartigue, Reverse Engineering et Copiage rapide de nuages de points numérisés, IUT de Cachan, Université Paris-Sud 11, août 2007.p.3-5.

- [12] M.O. Challali, I. Belaidi, K. Mohammedi, B. Boucherit, Reconstruction d'objets 3D Démarche globale et triangulation de Delaunay, Groupe Transfert de Technologie, LMMC, Université de Boumerdes, Mars 2004.p.2-7.
- [13] M.O. Challali, I. B. Magister_ Challali_Reconstruction d'objets discets 3 D Application de la triangulation de Delaunay. p. 13-104
- [14] LE REVERSE ENGINEERING, Titre In <http://www.msc-scanning.com/le-reverse-engineering.htm>
- [15] V.T.Rajan. Optimality of the Delaunay triangulation in Rd. Discrete Computational Geometry 12; 1994.p. 189-202
- [16] Seng Poh Lim, Habibollah Haron, Artificial Intelligence Review, June 2012,
- [17] P.L.George, H. Bourouchaki. Triangulation de Delaunay et maillage, application aux éléments finis. Editions Hermes, Paris 1997.p 7-15
- [18] Jean Christophe Beyler Benjamin Isenmann, Génération de terrain et triangulation de Delaunay, Mai 2005. P 3-13.
- [19] C.A.Arens, The Bowyer-Watson algorithm, Faculty of Civil Engineering and Geosciences Delft University of Technology, 2002,p.7-9.
- [20] Formats d'échanges. (s.d.). Titre In portail d'information de l'école centrale de paris: <http://cao.etudes.ecp.fr/?page=formats.htm#STL>
- [21] Vérification des fichiers STL, Titre In <http://www.drim3d.com/STL>
- [22] O. Marguin, C++ : LES BASES, 2003, p.1-2.
- [23] Borland C++ Builder Tutorial, 2002, cs.nyu.edu/~campbell/Tutorial.pdf p.1-2
- [24] Introduction a Open GL, Titre In <https://openclassrooms.com/courses/creez-des-programmes-en-3d-avec-opengl/introduction-a-opengl>.