

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab Blida

N° D'ordre :



Faculté des sciences

Département d'informatique

Réalisé par :

TAZI Asmaa HAOUCHINE Chahinez

En vue d'obtenir le diplôme de master

Domaine : Mathématique et informatique

Filière : Informatique
Spécialité : Informatique
Option : Ingénierie de logiciel

Sujet :

Reconstruction des frontières d'objets 3D à partir une triangulation de Delaunay

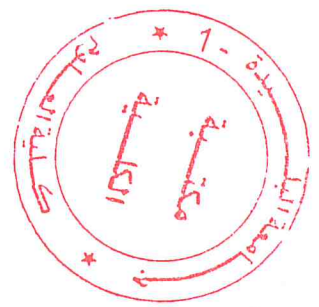
Soutenu le : 30/06/2018,

devant :

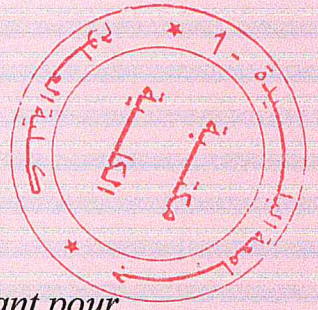
Mr. HAMMOUDA.
Mme. TOUBALINE
Mme TCHANTCHANE Zahida.
Mr. BEY Mohamed.
Mr. BENDIFALAH Hassen.
Mr. KAMECHE Abdallah Hichem.

Président
Examinatrice
Encadreuse.
Encadreur.
Encadreur.
Promoteur.

Promotion
2017 / 2018



Remerciement



Tout d'abord nous tenons à rendre grâce à **DIEU** tout puissant pour nous avoir donné le courage et la détermination nécessaires pour finaliser ce travail et le mener à terme.

En premier lieu, Nous tenons à remercier nos encadreurs **Mme TCHANTCHAN Zahida, Mr BENDIFALAH Hassen et Mr BEY Mohamed** pour leur encadrement, disponibilité, patience, et conseils qu'ont constitués un apport considérable.

Nous voudrions également remercier **Mr. KAMECHE Abdallah Hichem** qui nous a fait l'honneur d'être notre promoteur.

Nous tenons à remercier les membres de jury pour nous avoir fait l'honneur de juger notre travail.

Nous tenons à remercier nos très chers parents pour leur soutien et leur encouragement illimité durant ces longues années d'études.

Un grand Merci à nos amis, à tous ceux qui nous ont soutenu et à tous ceux qui ont participé de près ou de loin à l'aboutissement de ce travail.

Asmaa & Chahinex.

Dédicaces Asma

Par la grâce de dieu je viens de terminer mon projet de fin d'étude.

Que de dédie :

A mes parents, qui grâce à leurs encouragements et leurs grands sacrifices, ils ont pu créer le climat affectueux et propice à la poursuite de mes études.

Aucune dédicace ne pourrait exprimer mon respect ma considération et mes profonds sentiments envers eux.

*A mes beaux frères : **Mohamed et Khalil** et à ma chère sœur **Maroia** à qui je souhaite pleine de réussite.*

A toute ma famille paternelle et maternelle.

*A mon binôme **Chahinez** et à sa petite famille.*

*A mes chères amis plus particulièrement: **Samia, Manel, Maroia, Ahlem, Amina, Sabrina, lillia, Sofia, Nour El Houda, Mohamed, Diae Elddine, Abd El Kader, Sid ahmed.***

A tous ceux qui m'ont soutenu.

ملخص :

هذا العمل جزء من تطوير منصة برمجية لإنتاج أسطح من الأشكال المعقدة التي بدأها فريق التصميم والتصنيع بمساعدة الكمبيوتر CFAO التابعة لقسم الإنتاج والروبوتات DPR التابعة لمركز التطوير التقنيات المتقدم CDTA.

في هذا المشروع ، نحن مهتمون بتوليد تقنية التثليث التقريبي للجدد الخارجي للجزء ثلاثي الأبعاد من الشكل المعقد من تثليث دلوناي. يتكون العمل من اقتراح منهجية لإعادة بناء حدود الكائن الثلاثي الأبعاد باستخدام تقنيات تقريب الحدود مثل " أشكال ألفا " و "غابرييل غراف" و "كروست" و ... إلخ. الهدف من هذا العمل هو تطوير وحدة برمجية رسومية وتفاعلية تحت نظام ويندوز.

كلمات مفتاحية إعادة الإعمار, التثليث دلوناي, نموذج STL ، أشكال ألفا.

Présumé

Ce travail s'insère dans le cadre du développement d'une plateforme logicielle pour la production des surfaces de formes complexes initiée 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 à la génération d'une triangulation approximant la peau extérieure d'objet 3D de forme complexe à partir d'une triangulation de Delaunay. Le travail consiste à proposer une méthodologie permettant la reconstruction des frontières d'objet 3D en utilisant des techniques d'approximation de frontière telle que «Alpha Shape», «Graph de Gabriel», «Crust»,...etc. Le but de ce travail est le développement d'un module logiciel graphique et interactif sous Windows automatisant cette tâche.

Mots-Clés : Reconstruction, Triangulation de Delaunay, Modèle STL, Alpha forme.

Abstract

This work is a part of the software developed by “CFAO” team of the Center for the Development of Advanced Technologies “CDTA” dedicated to the machining of free form surfaces on CNC milling machines.

In this project we are interested in the generation of a triangulation approximating the outer skin of 3D object of complex shape from a Delaunay triangulation. The work consists of proposing a methodology for the reconstruction of 3D object boundaries using boundary approximation techniques such as "Alpha Shape", "Gabriel Graph", "Crust", ... etc. The goal of this work is the development of a graphical and interactive software module under Windows automating this task.

Keywords: Reconstruction, Delaunay Triangulation, STL Model, Alpha shape.

Sommaire

Introduction Générale.....	1
Chapitre 1 : Etat de l'art	3
1.introduction	4
2.Généralités sur la CFAO	4
3.Format d'échange de données	5
3.1Modèle STL	5
3.1.1.Propriétés	5
3.1.2.Types de fichier STL	6
3.1.3.Avantages.....	9
3.1.4.Inconvénients.....	9
4.Modélisation géométriques	9
4.1.Modèle paramétrique	10
4.2.Modèle polyédrique	10
4.3.Modèle de subdivision.....	11
5.Représentation des surfaces	11
5.1.Définition.....	11
5.2.Modélisation Filaire.....	12
5.3.Modélisation surfacique.....	12
5.4.Modélisation volumique	13
6.Reverse Engineering	13
6.1.Définition.....	13
6.2.Pourquoi utiliser le Reverse Engineering.. ..	14
6.3.Processus de l'ingenierie inverse.....	14
6.3.1.Numérisation de l'objet	15
6.3.2.Recalage.....	16
6.3.3.Décimation	16
6.3.4.Segmentation.....	17
6.3.5.Définition continue de l'objet.....	17
6.4.Avantages de l'ingenierie inverse.....	17
7.Reconstruction d'objet 3D	18
7.1.Définition.....	18
7.2.Méthodes de reconstruction	18
7.2.1.Reconstruction par approximation	18
7.2.2.Reconstruction par interpolation.....	19
8.Triangulation de Delaunay	20
8.1.Définition.....	20
8.2.Propriétés de Delaunay	20
8.3.Triangulation Delaunay en 3D.....	21
8.4.Méthodes d'implimentation de Delaunay.....	21
8.4.1. Méthode Destruction-Construction.....	22
8.4.2.Algorithme de flip	22

9.Méthodes de la reconstruction des frontieres d'objet.....	23
9.1.Graphe de Gabriel.....	23
9.1.1.Description.....	24
9.2.CRUST.....	24
9.2.1 Description.....	25
9.2.2.Avantages.....	25
9.2.3.Inconvénients.....	26
9.2.4.Complexité.....	26
9.2.5. Diagramme de Voronoi.....	26
9.3.Alpha Shape.....	27
9.3.1.Alpha shape 3D.....	27
9.3.2.Alpha complexe.....	27
9.3.3.Description.....	28
9.3.4.Complexité.....	28
9.3.5.Propriétés de alpha shape.....	29
9.3.6.Avantages.....	29
9.3.7. Inconvénients.....	29
10. Conclusion.....	30
Chapitre II : Conception de l'application.....	31
1. Introduction.....	32
2. Architecture globale du système.....	32
2.1.Vérification du fichier.....	33
2.2.Récupérer limites.....	34
2.3.Créer cellules.....	34
2.4.Lire les tetraedres.....	35
2.5.Affecter les points aux cellules.....	36
2.6.Déterminer voisins des tetraedres.....	36
2.7.Déterminer tetraédres et facettes frontieres.....	38
2.8.Appliquer Alpha shape.....	38
2.9.Modifier Triangulation de Delaunay.....	39
2.10.Générer fichier STL.....	40
3.Conclusion.....	40
Chapitre III :	41
1.Introduction.....	42
2.Implémentation.....	42
2.1.Language de programmation utilisé.....	42
2.1.1. Embarcadero c++ builder.....	42
2.1.2.OpenGl.....	43
3.Présentation des fenêtres.....	43
3.1.Creation des cellules.....	44
3.2.Lecture des points.....	45
3.3.Voisins des Tetraedres.....	46
3.4.Alpha Shape.....	46
4.Validation.....	47

4.1.Lecture de fichier	47
4.2.Creation des cellules	48
4.3.Lecture des points	49
4.4.Creation des voisins des tetraedres	51
4.5.Détermination des facettes et tétraedre frontières.....	51
4.6.Application de Alpha shape	54
5.Conclusion.....	56
Conclusion générale	58
Références bibliographiques	60

Liste des figures

Figure 1: : règles d'autorités.....	3
Figure 2: exmple de STL Format ASCII.....	3
Figure 3: exemple de STL Format binaire.....	4
Figure 4: structure du fichier STL.....	5
Figure 5: Localisation d'un point sur une surface paramétrique.....	6
Figure 6: Exemple d'une géométrie paramétrique polyédrique.....	7
Figure 7: Organigramme de modélisation de surface.....	8
Figure 8: Modélisation FIL de FER.....	8
Figure 9: Exemple d'une modélisation surfacique.....	9
Figure 10: Processus de l'ingénierie inverse.....	11
Figure 11: Numérisation d'un objet.....	12
Figure 12: Segmentation d'un lapin.....	13
Figure 13: Approximation de la surface.....	15
Figure 14: Interpolation de la surface.....	15
Figure 15: La triangulation de Delaunay.....	16
Figure 16: Tétraèdre 3D.....	17
Figure 17: Sphère circonscrite.....	17
Figure 18: Les Etapes de l'algorithme destruction et instruction.....	18
Figure 19: Un côté illégal (gauche), le flip (droite).....	19
Figure 20: Exemple de Graphe de Gabriel.....	19
Figure 21: processus du Graphe de Gabriel.....	20
Figure 22: Exemple de CRUST.....	21
Figure 23: Diagramme de Voronoï.....	22
Figure 24: Alpha complexe.....	24
Figure 25: Exemple de quelques Alpha Shape.....	25
Figure 26: Organigramme de l'Architecture générale.....	33
Figure 27: Organigramme de vérification du fichier.....	34
Figure 28 : Organigramme de création des cellules.....	35
Figure 29: création des cellules.....	36
Figure 30: Affectation des points aux cellules.....	36
Figure 31: Caractéristique d'un tétraèdre.....	37
Figure 32: Exemple de centre de la sphère circonscrite d'un triangle.....	39
Figure 33: Sphère circonscrite, rayon et centre de la sphère.....	39
Figure 34: Organigramme de génération du fichier STL.....	40
Figure 35: Onglets de la fenêtre principale.....	44
Figure 36: Onglet création des cellules.....	44
Figure 37: Onglet lecture des points.....	45
Figure 38: Onglet voisins de tétraèdre.....	46
Figure 39: Onglet alpha shape.....	47
Figure 40: choisir un fichier.....	48

<i>Figure 41: Paramètres du Brut.....</i>	<i>48</i>
<i>Figure 42: Visualisation de toutes les cellules</i>	<i>49</i>
<i>Figure 43: Visualisation d'une seule cellule.....</i>	<i>49</i>
<i>Figure 44: Visualisation des points globale</i>	<i>50</i>
<i>Figure 45: Visualisation de la Tétraédrisation.....</i>	<i>50</i>
<i>Figure 46: le Modèle dans les cellules</i>	<i>50</i>
<i>Figure 47: Tétraèdres partagent un point commun.....</i>	<i>50</i>
<i>Figure 48: Visualisation de tétraèdre</i>	<i>51</i>
<i>Figure 49: Voisins de tétraèdre</i>	<i>51</i>
<i>Figure 50: Facettes frontières en filaire.....</i>	<i>52</i>
<i>Figure 51: Facettes frontières en rendu</i>	<i>52</i>
<i>Figure 52: Normale de Facette frontière.....</i>	<i>52</i>
<i>Figure 53: Cercle de Facette frontière</i>	<i>53</i>
<i>Figure 54: sphère de Facette Frontière.....</i>	<i>53</i>
<i>Figure 55: : tétraèdres frontières en filaire.....</i>	<i>53</i>
<i>Figure 56:.. tétraèdre frontières en rendu.....</i>	<i>53</i>
<i>Figure 57:.. Alpha = 5.....</i>	<i>54</i>
<i>Figure 58:.. Alpha =9.5</i>	<i>54</i>
<i>Figure 59:.. Alpha12.....</i>	<i>55</i>
<i>Figure 60: Alpha20.....</i>	<i>55</i>
<i>Figure 61: Alpha =60.....</i>	<i>55</i>
<i>Figure 62: Alpha =100.....</i>	<i>55</i>

Introduction générale

Introduction générale:

Problématique

La reconstruction 3D, désigne la technique qui permet d'obtenir une représentation en trois (03) dimensions d'un objet à partir d'un nuage de points dont les coordonnées 3D sont saisies par un dispositif de mesure prises sous différents points de vue de l'objet.

Le processus de la reconstruction se fait en deux étapes, la triangulation et la reconstruction des frontières d'objet. Le résultat de la triangulation donne l'enveloppe convexe.

L'objectif de ce travail est de mettre en place une application logicielle permettant de faire la correction de l'enveloppe convexe pour avoir les frontières de l'objet. La solution à ce problème, n'est pas triviale, c'est une tâche ardue et couteuse en temps.

Présentation du sujet

La reconstruction 3D, désigne la technique qui permet d'obtenir une représentation en trois dimensions d'un objet soit par l'utilisation des outils de conception (logiciel de conception assistée par ordinateur) soit à partir d'un nuage de points définis par leurs coordonnées. Les coordonnées sont souvent obtenues par digitalisation d'objets physiques à l'aide de moyens appropriés tel que les machines à mesurer tridimensionnels (MMT). Le nuage de points obtenu est souvent représenté sous forme de maillages surfaciques ou volumique. Ces objets ainsi digitalisés sont utilisés dans divers domaines tel que l'aéronautique, l'industrie de l'automobile, les moules, les matrices, etc....

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-conception ou Reverse Engineering c'est l'opération inverse qui consiste à conférer un modèle numérique à un objet physique.

L'objectif final du processus du Reverse Engineering est la reconstruction du modèle informatique « modèle CAO ».

Le passage des nuages de points vers un modèle facettisé est réalisé en utilisant des techniques de reconstruction d'objet basé sur le concept de la triangulation. La plus fiable des méthodes de la triangulation est celle de Delaunay. Cette dernière génère l'enveloppe convexe de l'objet physique. Cette méthode à elle seule, ne peut représenter un objet 3D sans certaines irrégularités surtout au niveau de la jonction de surfaces constituant l'objet 3D.

Chapitre I

ETAT DE L'ART

Chapitre I : Etat de l'art

1. Introduction

Dans ce chapitre nous présentons d'une manière générale le système de Conception et de Fabrication Assisté par Ordinateur (CFAO), ainsi que les différentes méthodes de modélisation des surfaces en CAO, ensuite une présentation des phases de la Rétro Conception ainsi que la présentation de la triangulation de Delaunay sont détaillées et les méthodes de la reconstruction des frontières d'objet sont recensées. Enfin la description des formats d'échange des données et en particulier le format STL sont introduits.

2. Généralités sur la CFAO [1]

LA CFAO est la combinaison de la CAO avec la FAO (conception assistée par ordinateur/fabrication assistée par ordinateur). Ce terme fait référence à un logiciel utilisé à la fois pour la conception et la fabrication de produits.

Le terme "CAO" fait référence à l'utilisation des technologies informatiques pour la conception et la documentation des conceptions. Les applications de CAO/FAO sont utilisées pour concevoir les produits et pour planifier les processus de fabrication. Les logiciels de FAO utilisent les modèles et les ensembles créés dans les logiciels de CAO pour générer des trajectoires d'usinage sur lesquelles s'appuient les machines qui convertissent les conceptions en pièces physiques. Les logiciels de CAO/FAO sont généralement utilisés pour l'usinage des prototypes et des pièces finies.

La CAO et la FAO, sont deux différents systèmes, qui nécessitent un échange d'informations hétérogènes, 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 le passage de l'information entre divers programmes ou logiciels par un fichier intermédiaire. Parmi les premiers, Les formats natifs développés par les éditeurs des logiciels, puis les formats neutres et les passerelles comme (IGES, STEP, DXF PATRAN, STL...). Chaque application nécessite

l'utilisation d'un format spécifique en fonction des données à échanger du fait que chaque format d'échange est caractérisé par la nature de l'information translatée.

Les applications de CFAO utilisent le plus souvent le format 'STL ' qui convient le plus à des applications d'usinage ou de reconstruction d'objet. De la conception à la production, le procédé de CFAO est constitué d'une suite d'activités importantes en passant par les deux étapes CAO et FAO.

3. Format 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 fréquentes.

3.1 Modèle STL

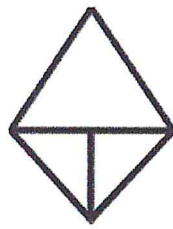
Un fichier STL stocke des informations sur les modèles 3D. Ce format décrit uniquement la géométrie de la surface d'un objet tridimensionnel sans aucune représentation de couleur, de texture ou d'autres attributs de modèle commun.

Ces fichiers sont généralement générés par un programme de conception assistée par ordinateur (CAO), en tant que produit final du processus de modélisation 3D avec une extension STL. Le format de fichier STL, est le plus couramment utilisé pour l'impression 3D. Lorsqu'il est utilisé avec un trancheur 3D il permet à un ordinateur de communiquer avec le matériel de l'imprimante 3D. Depuis ses débuts modestes, le format de fichier STL a été adopté et pris en charge par de nombreux autres progiciels de CAO et est aujourd'hui largement utilisé pour le prototypage rapide, l'impression 3D et la fabrication assistée par ordinateur. Les amateurs et les professionnels l'utilisent de la même manière.

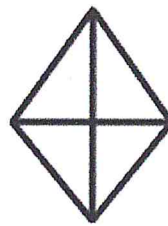
3.1.1 Propriétés [2]

- Approximation de la frontière par des facettes triangulaires.

- Le fichier « STL » est volumineux. Le modèle CAO est remplacé par des triangles dont le nombre dépend de la précision désirée, de la forme de l'objet et de ses dimensions.
- Le fichier « STL » ne contient pas l'information d'échelle et les coordonnées sont dans des unités arbitraires.
- Règle d'autorité : chaque triangle doit partager deux sommets avec chacun de ses triangles adjacents (Figure 12.b). Ainsi, un sommet d'un triangle ne peut pas être localisé sur le côté d'un autre triangle.



a). Non vérifier



b). vérifier

Figure 1 : règles d'autorités

3.1.2 Types de fichier STL

Il existe deux types de fichier STL :

a) STL Format ASCII

Le format ASCII est principalement destiné à tester de nouvelles interfaces CAO. La grande taille de ses fichiers le rend peu pratique pour un usage général. La syntaxe d'un fichier STL en ASCII est la suivante (Figure 2) :

```

solid name
  {
    facet normal ni nj nk
      outer loop
        vertex v1 v1y v1z
        vertex v2 v2y v2z
        vertex v3 v3y v3z
      endloop
    endfacet
  } +
endsolid name

```

Figure 2 : exemple de STL Format ASCII.

Le visage en gras indique un mot-clé. Ceux-ci doivent apparaître en minuscules. Notez qu'il y a un espace dans "facette normal" et dans "boucle externe", alors qu'il n'y a pas d'espace dans les mots-clés commençant par "fin". L'indentation doit être avec des espaces, les onglets ne sont pas autorisés. La notation "{...} + " signifie que le contenu des accolades peut être répété une ou plusieurs fois. Les symboles en italique sont des variables qui doivent être remplacées par des valeurs spécifiées par l'utilisateur. Les données numériques dans les lignes normales et de sommet de facette sont des flottants de précision simple, par exemple, 1.23456E + 789.

STL Format binaire

Le format binaire utilise l'entier IEEE et la représentation numérique en virgule flottante. La syntaxe d'un fichier STL binaire est la suivante (Figure 3) :

Bytes	Data type	Description	
80	ASCII	Header. No data significance.	
4	unsigned long integer	Number of facets in file	
<div style="display: inline-block; vertical-align: middle; font-size: 3em; line-height: 1;"> { </div>	4	float	i for normal
	4	float	j
	4	float	k
	4	float	x for vertex 1
	4	float	y
	4	float	z
	4	float	x for vertex 2
	4	float	y
	4	float	z
	4	float	x for vertex 3
	4	float	y
	4	float	z
	2	unsigned integer	Attribute byte count

Figure 3 : exemple de STL Format binaire [40].

La notation "{...}+" signifie que le contenu des accolades peut être répété une ou plusieurs fois. La syntaxe d'attribut n'est pas documentée dans la spécification formelle. Il est spécifié que le nombre d'octets d'attribut doit être défini sur zéro.

Comment un fichier STL stocke-t-il des informations sur les facettes?

Le format de fichier STL fournit deux façons différentes de stocker des informations sur les facettes triangulaires qui recouvrent la surface de l'objet. Ceux-ci sont appelés le *codage ASCII* et le *codage binaire*. Dans les deux formats, les informations suivantes de chaque triangle sont stockées:

1. Les coordonnées des sommets.
2. Les composants de l'unité du vecteur normal au triangle. Le vecteur normal doit pointer vers l'extérieur par rapport au modèle 3D.

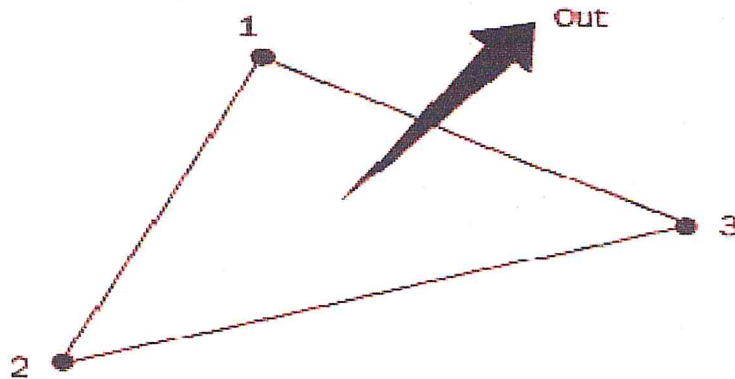


Figure 4 : structure du fichier STL [40].

Un fichier STL stocke les coordonnées des sommets et des composantes du vecteur normal de l'unité aux facettes

3.1.3 Avantage

- Format de fichier simple.
- Pris en charge par plusieurs logiciels de modélisation 3D.
- Génération rapide des maillages.

3.1.4 Inconvénients

- format STL décrit seulement la géométrie, d'autres informations sur l'objet telles que la couleur, matières ne sont pas décrites.
- Une fois un fichier exporté en STL, il est ensuite difficilement modifiable.
- Suppression des informations de connectivité
- Perte d'informations sur la géométrie de l'objet.

4. Modélisation géométriques

En informatique graphique, on utilise de nombreuses représentations des objets réels (données médicales issues de scanner, modèles CAO, jeux vidéo, ...). La façon la plus courante de qualifier une forme est de décrire son enveloppe extérieure : c'est la modélisation surfacique. Ces modèles sont notamment utilisés dans les systèmes de CAO/FAO :

- Modèle paramétrique.
- Modèle polyédrique.
- Modèle de subdivision.

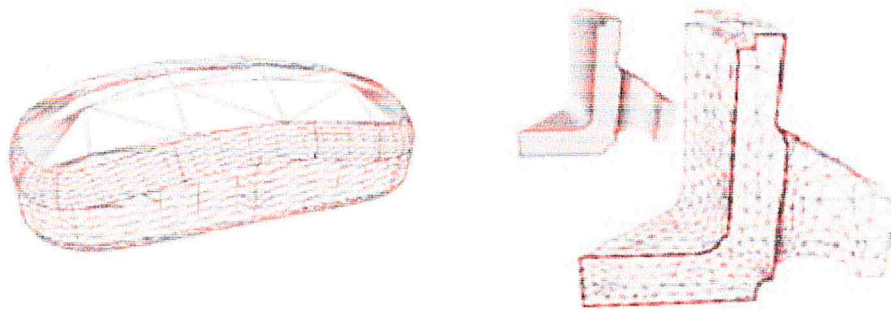


Figure 6: Exemple d'une géométrie paramétrique polyédrique.

4.3 Modèle de subdivision [7]

D'autres modèles existent pour représenter un objet 3D, notamment les modèles de subdivision. Ces modèles s'appuient sur les modèles polyédriques et permettent de modéliser des surfaces complexes et « lisses » par des étapes de subdivision.

5 Représentation des surfaces

5.1 Définition [8]

Les surfaces complexes, appelées aussi surfaces gauches, sont caractérisées par une topologie complexe composée d'un ensemble de surfaces de formes concaves, convexes, en selle de cheval ou à double courbure. Outre la complexité de l'opération de représentation de ces surfaces, s'ajoute la difficulté de considérer l'ensemble des formes complexes et du design. La production de ces surfaces est un domaine complexe et fait sans cesse confronté à de nouveaux défis: des exigences de précision de plus en plus grandes, des formes de plus en plus complexes, une pression relative aux coûts croissants, une meilleure efficacité et une plus grande productivité.

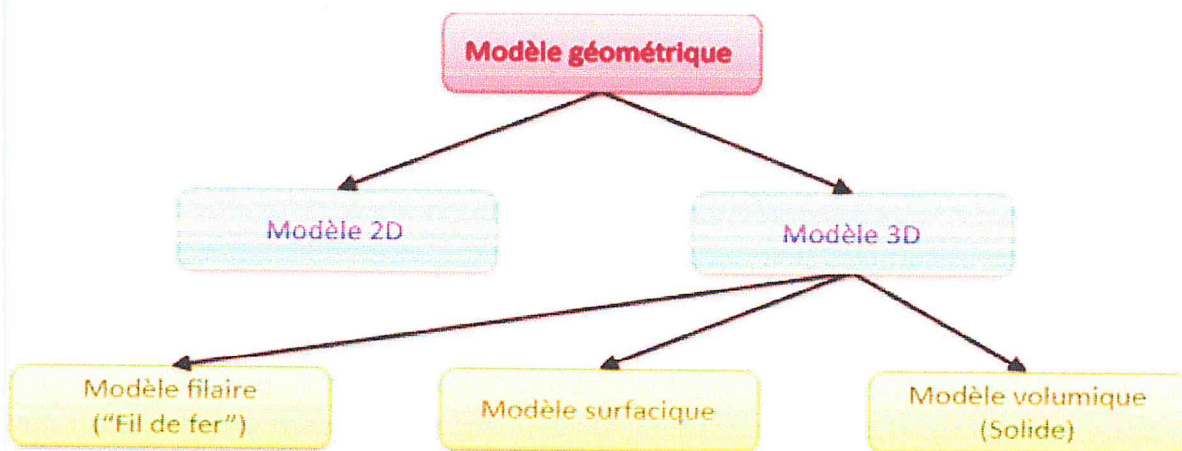


Figure 7 : Organigramme de modélisation de surface.

5.2 Modélisation Filaire [9]

Ce type de modèle est le premier à avoir été mis en œuvre. On ne conserve que les coordonnées X, Y, Z des sommets et des arêtes qui les joignent. Plusieurs ambiguïtés peuvent découler de son utilisation car plusieurs interprétations d'un même modèle peuvent être faites. Il est impossible de réaliser sur ces modèles, le calcul des propriétés physiques (volume, poids, centre de gravité, inertie, etc. ...). Son utilisation présente des avantages tels que la création, la visualisation rapide du modèle, une faible utilisation du processeur, une modification aisée des points et des arêtes et une faible capacité de mémoire.

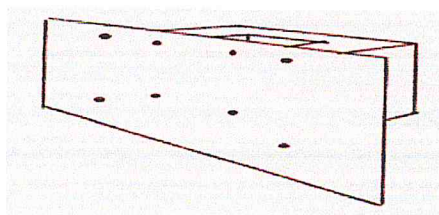


Figure 8: Modélisation FIL de FER.

5.3 Modélisation surfacique [10]

Historiquement, c'est la deuxième technique apparue mais c'est en fait la première qui a permis une approche 3D plus profonde que la représentation fil de fer.

La modélisation des surfaces est un aspect technique très important dans de nombreux domaines (conception des formes en automobile, aéronautique, ...), surtout lorsque les

surfaces des objets sont définis à partir des réseaux de points obtenus d'après des maquettes réalisées par machine à mesurer.

Les avantages de l'utilisation des surfaces sont :

- Une description exacte de l'enveloppe de la pièce.
- Une meilleure visualisation de la pièce par l'utilisation de techniques de rendu réaliste.

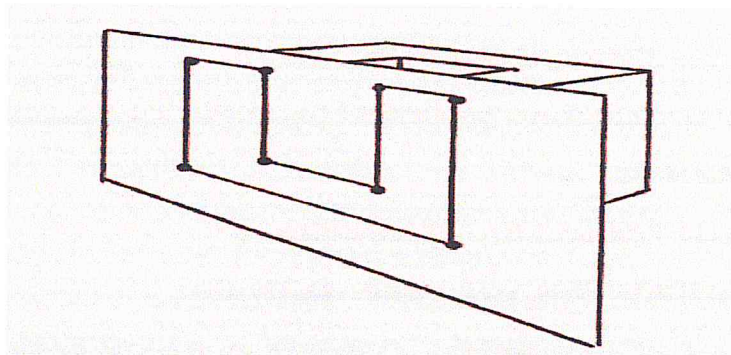


Figure 9 : Exemple d'une modélisation surfacique.

5.4 Modélisation volumique [10]

C'est la technique de représentation d'un objet la plus utilisée. Ainsi les modélisateurs volumiques ou solides peuvent calculer les propriétés mécaniques, réaliser des vérifications de collision, fournir les vues complètes ou en coupe de pièces, avec élimination des lignes cachées.

Le solide se caractérise par son aspect homogène 3D, ses limites (le solide occupe un espace fini) et ses frontières qui définissent un intérieur et un extérieur au volume. Les principaux modèles de solide sont : la représentation par occupation spatiale, la représentation par les frontières.

6 Reverse Engineering (ingénierie inverse)

6.1 Définition

L'ingénierie, désigne l'ensemble des processus nécessaire pour le développement d'un produit, de l'idée initiale à la conception et au développement jusqu'au produit final.

La rétro ingénierie désigne la discipline qui consiste à analyser un produit final pour comprendre son fonctionnement ou sa production dans le but de le dupliquer. Il s'agit donc, de faire de l'ingénierie inversée (on part du produit et on arrive à l'étape de construction).

Dans une procédure d'ingénierie directe, le concepteur du produit élabore un dessin détaillant les modalités de fabrication d'un objet puis ce dernier est mis en production et fabriqué conformément au dessin de conception.

Le Reverse Engineering suit la procédure inverse. Dans un premier temps, les ingénieurs identifient les composants du système et leurs interactions. L'objet est démonté afin d'en découvrir sa structure et son fonctionnement. La duplication peut alors se faire en capturant les dimensions physiques, les fonctionnalités et les propriétés matérielles de l'objet.

Après un dessin CAO est généré pour le système analysé. La phase suivante consiste à reproduire un modèle qui représente le produit original avec une précision conformément au dessin établi.

6.2 Pourquoi utiliser le Reverse Engineering [8]

Le Reverse Engineering est utilisé et adopté dans les cas suivants :

- Le producteur original n'existe pas, mais le client a besoin de ce produit.
- Le producteur original a arrêté de fabriquer le produit.
- La documentation de conception du produit original est perdue.
- Inspection et contrôle d'une pièce produite avec son modèle CAO.
- Analyse des produits des concurrents.
- Création des données en 3D d'un modèle.
- Documentation architecturale, construction et de mesure.

6.3 Processus de l'ingénierie inverse

Le processus de la reconstruction d'un objet en utilisant l'ingénierie inverse nécessite le passage par cinq principales étapes :

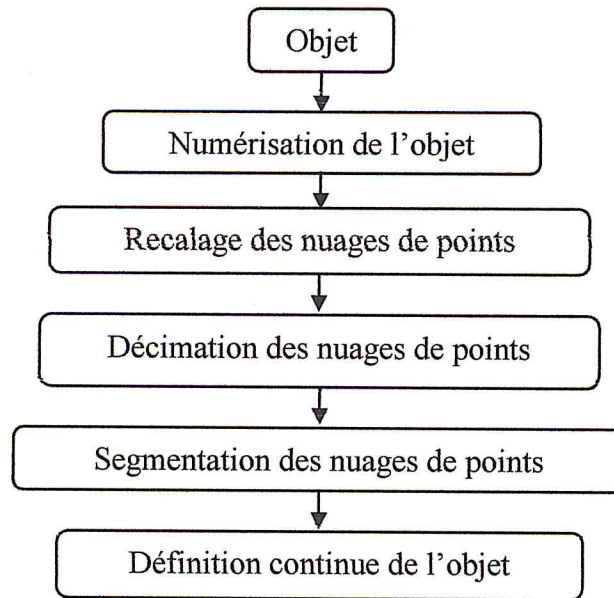


Figure 10 : Processus de l'ingénierie inverse.

6.3.1 Numérisation de l'objet [12]

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 en 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.

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:

- 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), (Figure 7).
- 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.

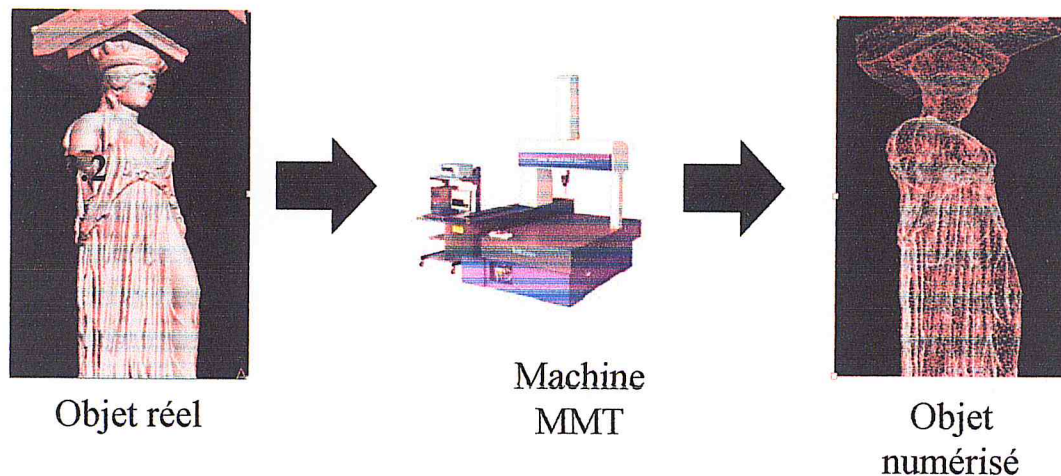


Figure 11: Numérisation d'un objet.

6.3.2 Recalage [13]

Un objet de forme complexe ne peut pas être numérisé en prenant en compte une seule face, plusieurs points de vue sont nécessaires. On aura dans ce cas plusieurs nappes de points. Le recalage consiste à mettre en correspondance ces derniers, deux types de recalage existent :

- Recalage simple : (classique) permet un recalage deux par deux. Deux ensembles de points sont recalés afin de former un seul, et ce dernier va être recalé avec le troisième et ainsi de suite.
- Recalage globale : permet de faire une correspondance simultanée de plusieurs ensembles de points.

6.3.3 Décimation [14]

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.

6.3.4 Segmentation [15]

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.

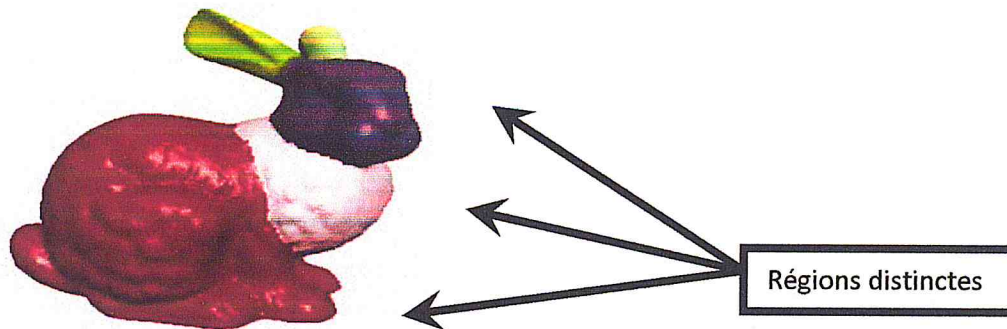


Figure12 : Segmentation d'un lapin.

6.3.5 Définition continue de l'objet

Cette étape est celle qui présente les plus grandes difficultés dans le processus de reconstruction. Dès que les régions sont identifiées, une approximation est faite par des modèles géométriques.

6.4 Avantages de l'ingénierie inverse [16]

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

- 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 production 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.

7. Reconstruction d'objet 3D

7.1 Définition

La reconstruction 3D, désigne la technique qui permet d'obtenir une représentation en trois dimensions d'un objet à partir d'un nuage de points. Elle peut aussi être définie, comme étant le processus au moyen duquel, une représentation 3D est créée dans la mémoire d'un ordinateur, afin de décrire les caractéristiques géométriques d'un objet réel (dimensions, volume, forme...). Il existe en perception artificielle, une multitude de techniques de reconstruction 3D, en fonction de la méthode d'acquisition de mesures sur l'objet, et de la représentation 3D finale.

Pour des objets de forme libre, on applique généralement une approche globale, car la construction incrémentale de maillage 3D est très complexe. Un algorithme de création d'un maillage 3D à partir d'un nuage de points 3D doit être capable d'effectuer la connexion de l'ensemble de points représentatifs de l'objet sous forme d'éléments de surface plane, que ce soient des triangles, des hexagones, des carrés ou tout autre forme géométrique 2D.

Les algorithmes développés jusqu'à présent, résultent d'un compromis entre le coût de calcul, le volume occupé en mémoire par le modèle, et la qualité du maillage obtenu.

Plusieurs algorithmes permettent de générer un maillage à partir d'un nuage de points 3D: Marching Cubes, Ball Pivoting, Surfaces déformables, Alpha shapes, et de nombreuses méthodes dérivées de la triangulation de Delaunay. L'efficacité de chacune est définie par le temps de calcul pour traiter un nuage de taille donnée et par la qualité finale du maillage. [17]

7.2 Méthodes de reconstruction

7.2.1 Reconstruction par approximation

L'approximation signifie que la surface est très proche des points échantillonnés mais ne passe pas obligatoirement par eux.

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. [18]

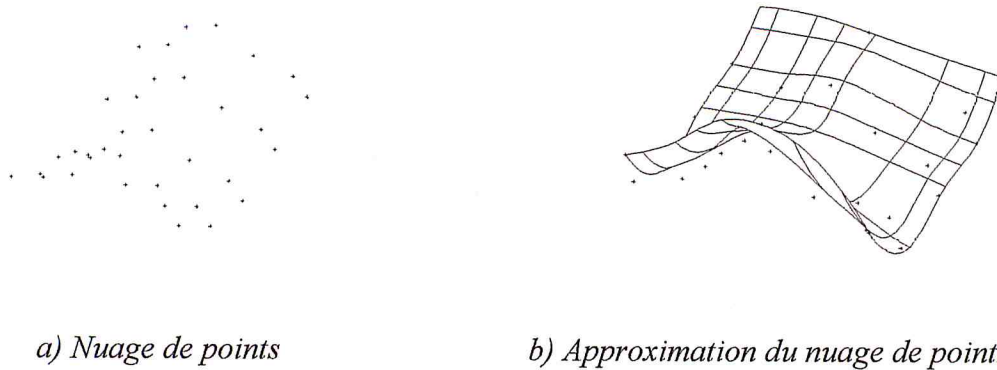


Figure13 : Approximation de la surface. [19]

7.2.2 Reconstruction par interpolation

L'interpolation signifie que la surface reconstruite préserve le nuage de points original, les points échantillonnés sur la surface de l'objet feront partie de la surface reconstruite.

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. [18]

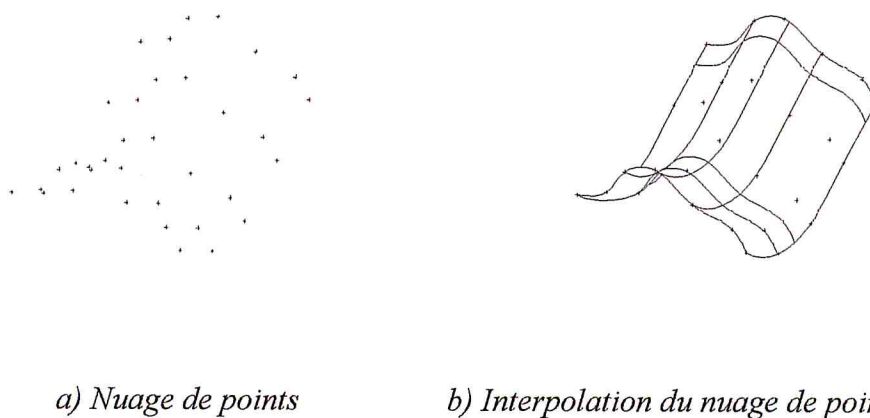


Figure 14 : Interpolation de la surface [19]

8. Triangulation de Delaunay

8.1 Définition

En mathématiques et plus particulièrement en géométrie algorithmique, la triangulation de Delaunay d'un ensemble 'P' de points du plan, est une triangulation $DT(P)$ telle qu'aucun point de 'P' n'est à l'intérieur du cercle circonscrit d'un des triangles de $DT(P)$.

Soit S un ensemble de points en position général. On appelle triangle de Delaunay de S , un triangle dont les sommets sont des points de S et qui est inscriptible dans un cercle vide, c'est-à-dire un cercle qui ne contient aucun point de S en son intérieur.

On appelle triangulation de Delaunay de S toute triangulation de S dont tous les triangles sont des triangles de Delaunay de S .

8.2 Propriétés de Delaunay [21]

- chaque triangle est entouré d'un cercle vide qui passe par les sommets du triangle et ne contient aucun autre site ou sommet à l'intérieur.
- Les sommets et côtés ouverts de l'ensemble des triangles de Delaunay sont deux à deux disjoints.
- La triangulation de Delaunay est parmi toutes les triangulations de S , celle qui maximise le minimum des angles des triangles.
- Le maillage de Delaunay interpole l'ensemble des points de S .

La triangulation de Delaunay de S est la seule triangulation de S dont toutes les arêtes sont légales.

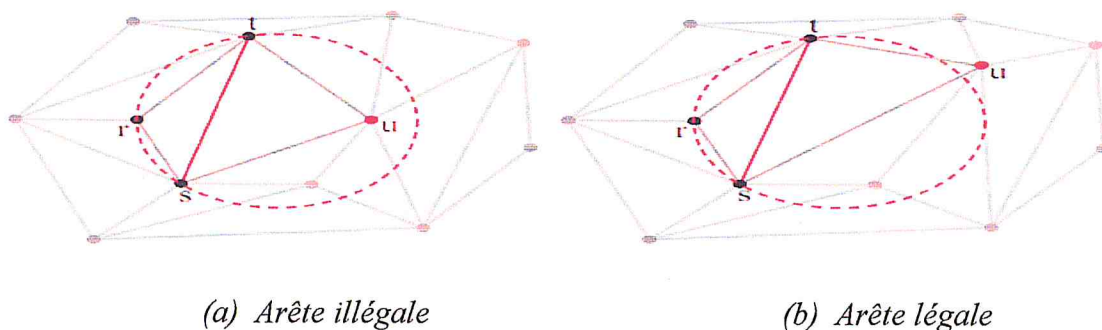


Figure 15: La triangulation de Delaunay.

8.3 Triangulation Delaunay en 3D [22]

En trois dimensions, la triangulation de Delaunay est une tétraédrisation de l'ensemble de points. Dans ce cas, le nuage de points définit la surface d'un objet ainsi que son intérieur. Chaque tétraèdre possède une sphère circonscrite vide, ces dernières ne se chevauchent jamais et chaque tétraèdre est adjacent à exactement quatre tétraèdres (sauf ceux qui sont situés sur la frontière de l'ensemble de points).

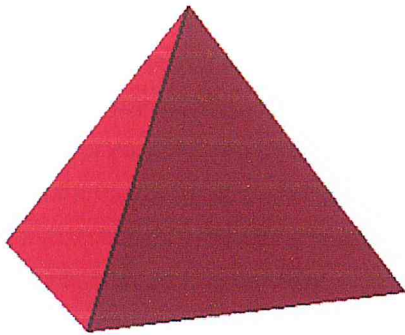


Figure 16: Tétraèdre 3D.

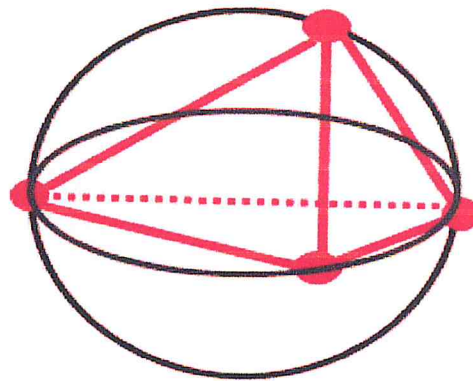


Figure 17 : Sphère circonscrite.

8.4 Méthodes d'implémentation de Delaunay

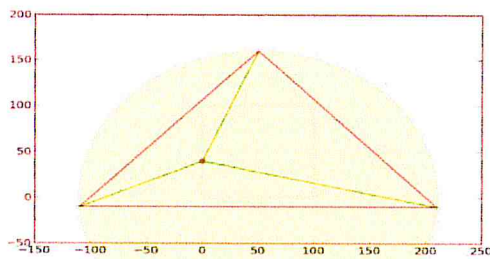
Il existe plusieurs algorithmes qui génèrent de façon incrémentale la triangulation de Delaunay. On parle de techniques incrémentales, lorsqu'une triangulation préliminaire est enrichie par l'ajout de points du nuage. Parmi les méthodes existantes On cite les algorithmes suivants :

- Méthode Destruction-Construction.
- Algorithme de Flip

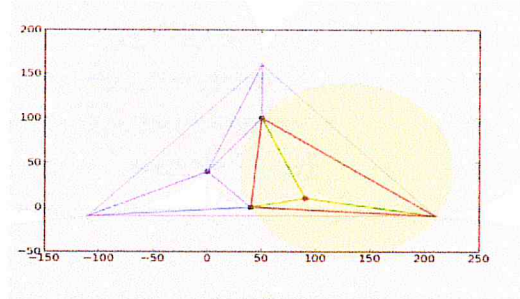
8.4.1 Méthode Destruction-Construction [23]

Elle se fait à partir d'une triangulation partielle du semis, et se poursuit par la triangulation existante augmentée d'un point. Donc, partant de quatre points de départ formant deux triangles (donc quatre points déjà triangulés), l'algorithme rajoute des points qui sont intérieurs à la triangulation et qui sont triangulés au fur et à mesure.

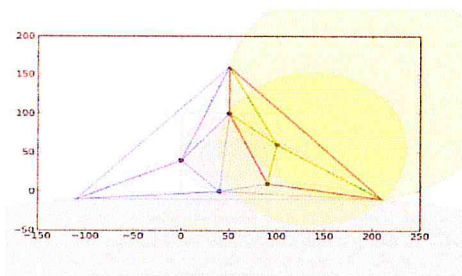
On procède ainsi : on utilise un triangle et un point choisi comme indiqué ci-dessus, puis tous les triangles dont le cercle circonscrit contient le point considéré sont détruits. Une fois les triangles illégaux détruits, on construit ceux qui sont engendrés par ce nouveau point, en ajoutant des arêtes reliant au point inséré les extrémités de celles qui n'appartenaient qu'à un seul triangle présent dans la liste de ceux qui ont été supprimés.



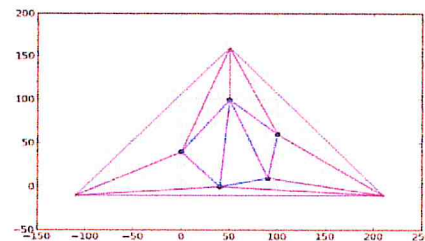
Insertion d'un point dans un « super-triangle »



Insertion 2, 3, 4 point.



Suppression des arêtes du super-triangle.



Suppression des arêtes du super triangle.

Figure 18 : Les Etapes de l'algorithme destruction et instruction.

8.4.2 Algorithme de flip [8]

Cet algorithme est plus rapide, en principe. C'est une grande fonction récursive (qui pourrait être implémentée en itératif), qui vérifie si les côtés sont légaux. Un côté est défini

comme étant l'arête commune à deux triangles. Pour savoir s'il est illégal, seuls les points qui ne sont présents que dans un des triangles sont considérés, donc les points opposés

Si c'est le cas, le côté illégal est remplacé par un côté relié par les deux points opposés. Et les quatre côtés restants sont testés pour voir si ce changement a provoqué un autre conflit. L'image qui suit montre un côté illégal et le flip qui permet de rendre ce côté légal.

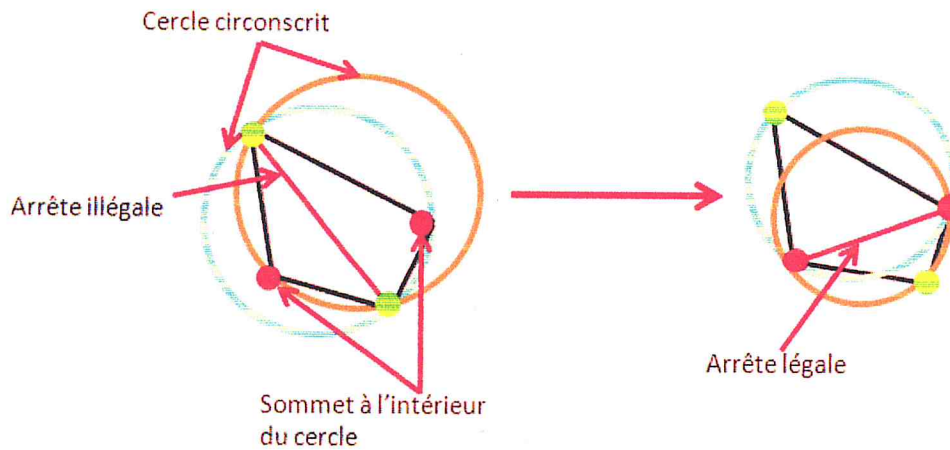


Figure 19 : Un côté illégal (gauche), le flip (droite).

9. Méthodes de la reconstruction des frontières d'objet

9.1 Graphe de Gabriel

Le graphe de Gabriel est un sous-graphe de la triangulation de Delaunay.

Deux points de l'ensemble d'échantillonnage S , p et q définissent une arête de graphe de Gabriel $G(S)$ si et seulement si le cercle (sphère) du diamètre pq ne contient pas de point de S . $G(S)$ est clairement inclus dans la triangulation de Delaunay de S $DT(S)$ et peut être extrait de $DT(S)$ en temps linéaire. [24]

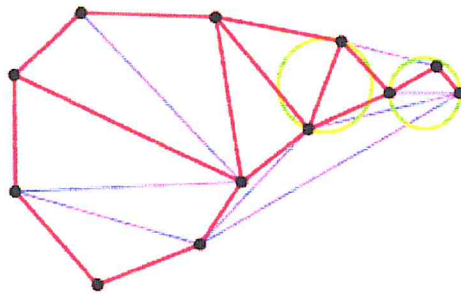


Figure 20 : Exemple de Graphe de Gabriel.

9.1.1 Description

L'exemple ci-dessous illustre le processus de Graphe de Gabriel.

En (a), la courbe est initialisée sur l'enveloppe convexe de l'ensemble de points. L'arête courante, circonscrite par une demi-sphère de Gabriel non vide, forme un triangle de Delaunay (gris foncé) avec le point. Ce triangle devient externe, la courbe est mise à jour en (b), et celle-ci continue de se contracter. Le résultat final apparaît en (d) avec des demi-sphères de Gabriel vides. [25].

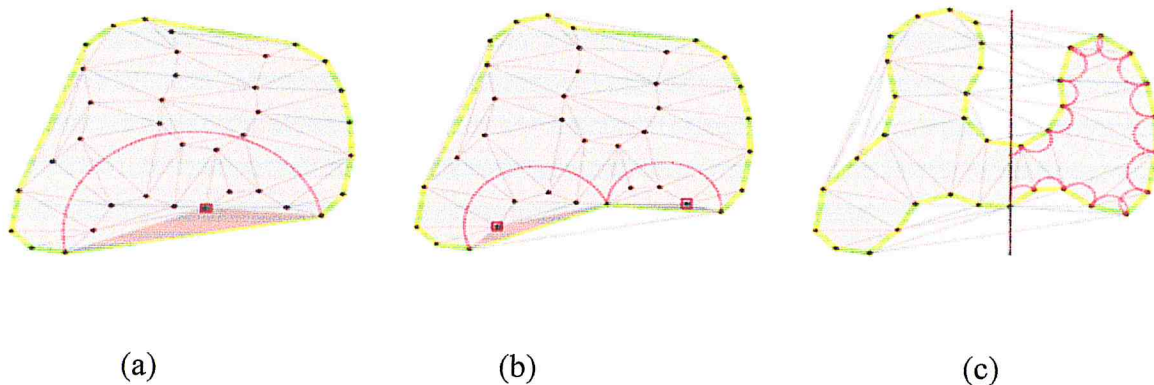


Figure 21 : processus du Graphe de Gabriel.

8.2 CRUST

Le CRUST est une méthode pour reconstruire une approximation du contour d'un objet 2D ou de la surface d'un objet en 3D à partir d'un ensemble de points mesurés sur un objet. En 3D. [26]

Étant donné un ensemble de points d'échantillonnage P , on détermine $T(P)$ la triangulation de Delaunay de P . Puis on calcule l'ensemble C des centres des cercles circonscrits aux triangles de $T(P)$, et on construit une triangulation de Delaunay $TD(P \text{ et } C)$ entre des points d'échantillonnage P plus l'ensemble des centres des cercles circonscrite C .

Le CRUST est l'ensemble des arrêtes de triangulation de Delaunay formées uniquement de points de d'échantillonnage P . [27]

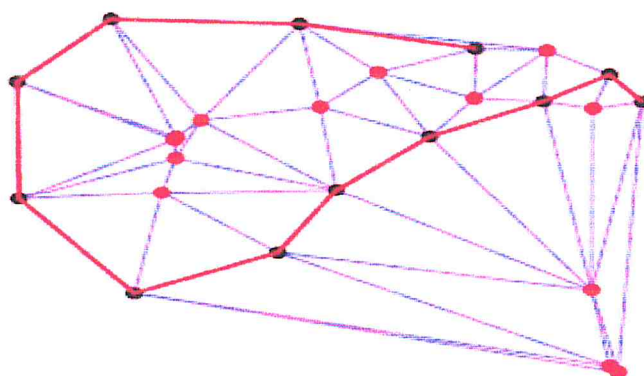


Figure 22 : Exemple de CRUST (les points P en noire et les points C en rouge).

8.2.1 Description

La méthode de CRUST consiste à :

- Acquisition Nuage de points de départ.
- Calcul du diagramme de Voronoï.
- Calcul des pôles des cellules de Voronoï.
- Construction de la triangulation de Delaunay de l'union des points du nuage et des pôles définis dans l'étape précédente.
- Garder uniquement les triangles dont les trois sommets sont des points de l'échantillon de départ.

8.2.2 Avantage :

- Crust Garantie de converger vers une surface qui reconstruit correctement.
- Permet une reconstruction surfacique sans besoin de spécifier des informations sur le processus d'échantillonnage.

8.2.3 Inconvénient :

- La surface de sortie peut contenir des trous et des défauts au niveau de la reconstruction, si l'exigence de densité de l'échantillon n'est pas respectée.
- Le temps d'exécution de l'algorithme dépend de l'ordre des points d'entrée.
- Ne peut pas gérer les virages serrés et les coins.

8.2.4 Complexité

La complexité asymptotique de l'algorithme de Crust est $O(n^2)$ où $n=|S|$, puisque c'est le pire temps nécessaire pour calculer une triangulation de Delaunay en trois dimensions.

Comme cela a été fréquemment observé, la complexité du pire cas pour le Delaunay en trois dimensions ne se produit presque jamais dans la pratique. Toutes les autres étapes sont le temps linéaire. [28]

8.2.5 Diagramme de Voronoï

Le diagramme de Voronoï pour un ensemble S de points, appelés sites, dans l'espace euclidien R_d de dimension d , est la subdivision de cet espace en régions, nommées cellules de Voronoï, qui contiennent tous les points de l'espace près du site générateur de la région que des autres sites [41].

Pour chaque sommet du diagramme de Voronoï, le cercle passant par les points de S voisins de ce sommet ne contient aucun autre point de S . [42]

Les polyèdres de Voronoï sont convexes, d'intérieurs non vide et forment un recouvrement tel que deux éléments distincts ont une intersection vide ou réduite à un point ou une arête.

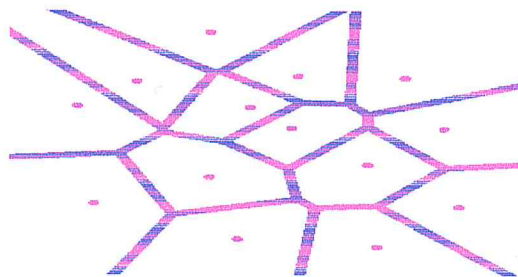


Figure 23 : Diagramme de Voronoï.[43]

8.3 Alpha Shape

La méthode des formes alpha était à l'origine motivée pour l'étude des points dans l'avion [29].

Soit $P := \{p_1, \dots, p_N\} \subset \mathbb{R}^3$ un ensemble fini de points. Intuitivement, l' α -forme de P est constituée de l'ensemble des arêtes et des triangles, ayant pour sommets les points de P , et pour lesquels il existe une boule circonscrite dont l'intérieur ne contient aucun point de P [30]

Alpha Shape permettent de décrire la forme d'un ensemble de points P , paramétré par la valeur de α [31], en utilisant un ensemble de points de contrôle $A \subset \mathbb{R}^2 \setminus P$, et la propriété de disque vide entre les points de P et les points de A .

Pour une valeur donnée de $\alpha \in [0; \infty]$, les boules alpha sont des boules de rayon α autour des points de P . Le complexe alpha correspondant de P est la triangulation de Delaunay de P restreinte à la balle alpha. [32]

Étant donné un ensemble $S \subset \mathbb{R}^r$ ($r = 2; 3$) en position générale, la triangulation de Delaunay de S est le sous complexe $DT(S)$ constitué uniquement de :

1. Tous les d -simplices $\triangleleft T$ avec $T \subset S$ tels que la boule circonscrite de T (pour $r = 2$ cela dégénère au grand cercle de T) ne contient aucun autre point de S .
2. Tous les k -simplexes qui sont des faces d'autres simplexes dans $DT(S)$.

8.3.1 Alpha Shape 3D

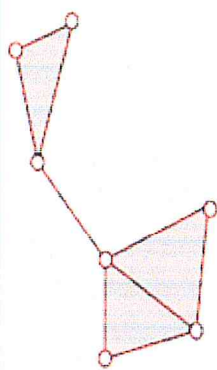
Alpha Shape est une généralisation de l'enveloppe convexe d'un ensemble fini de points.

Soit S un ensemble de points de \mathbb{R}^3 . Soit un réel $\alpha > 0$. Trois points $p, q, r \in S$ forment une facette de Alpha Shape de S si et seulement s'il existe un cercle ouverte de rayon α vide de tout point de S et dont la frontière contient p, q, r . [33]

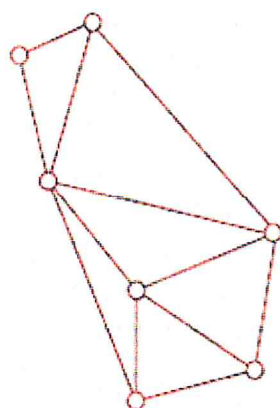
8.3.2 Alpha complexe

Afin de calculer la forme, le α -complexe est utilisé. Le complexe alpha C_α est un sous-complexe du Delaunay triangulation $DT(S)$. Le sous-ensemble de $DT(S)$ est déterminé par α .

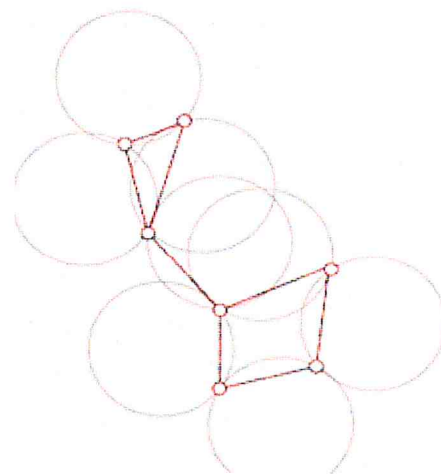
Chaque k -simplex $\Delta T \in DT(S)$ est dans le complexe alpha C_α si le cercle circonscrit de T de rayon $r < \alpha$ est vide. Alors la frontière S_α se compose de tous les simplices de s qui sont exposés. [34]



C_α



$DT(S)$



S_α

Figure 24 : Alpha complexe.

8.3.3 Description [35]

Cette méthode consiste à :

1. Calculer la triangulation de Delaunay.
2. Générer le complexe alpha (L'algorithme calcule le complexe pour tous valeurs).
3. Extraire la frontière de la forme du complexe.

Les étapes 1 et 2 peuvent être pré calculées pour un P donné.

8.3.4 Complexité

- Triangulation de Delaunay dans le pire des cas : $O(n^2)$. [36]
- La triangulation de Delaunay est faisable en temps : $O(n \log n)$. [37]
- Générer α -complexe: $O(m \log m)$. [35]
- Extraire les frontières de Alpha Shape : $O(m)$. [34]

8.3.5 Propriétés de alpha Shape

- $S_{\infty}(P) = \text{conv}(P)$
- $S_0(P) = P$
- $S_{\alpha_1}(P) \subset S_{\alpha_2}(P)$ if $\alpha_1 < \alpha_2$

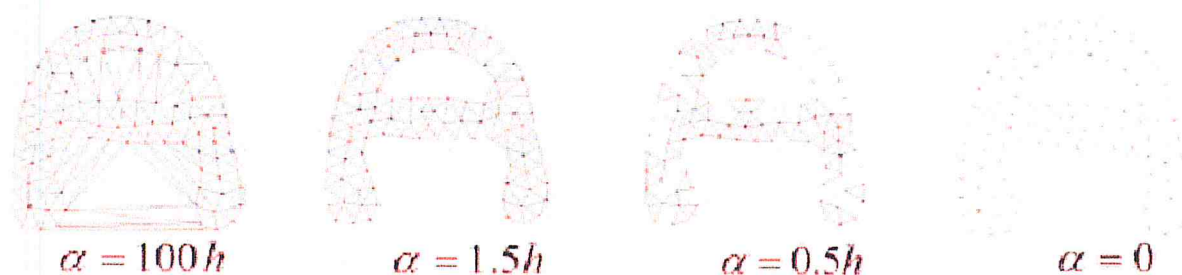


Figure 25 : Exemple de quelques Alpha Shape.

Il y a des ensembles de points pour lesquels il n'y a pas de α satisfaisant. Ceci se traduit par le fait que tous les alpha Shape ne donnent pas intuitivement de bonnes approximations de la surface de l'objet. C'est le plus souvent le cas lorsque les points ne sont pas uniformément échantillonnés. Ce problème découle du fait que les alpha Shape sont basées sur les distances entre les points afin de décider quels points se connecter par des triangles ou lignes. Les points non uniformes sont donc relativement inappropriés. [34]

8.3.6 Avantages

- Grâce à cette méthode, il devient dès lors possible d'extraire les points de contour de ces ensembles de points.
- permet d'extraire des surfaces avec l'ensemble de α .
- α -Shape interpole l'ensemble de points.

8.3.7 Inconvénients

- La valeur globale d'Alpha peut ne pas exister pour une reconstruction correcte dans certains cas.
- La reconstruction peut ne pas être étanche.
- La reconstruction peut être disjointe.
- Un calcul de Delaunay est fait sur l'ensemble de points qui est coûteux en 3D.

10 Conclusion

L'objectif de ce chapitre était de se familiariser avec les concepts liés à la Conception et Fabrication Assisté par Ordinateur (CFAO) et en particulier à la phase de reconstruction d'objet 3D utilisé dans le domaine du reverse engineering. Nous avons, à cet effet, essayé de donner un aperçu général sur les différentes méthodes de modélisation des surfaces en CAO. En outre, nous avons présenté une présentation des phases de la Rétro Conception et la présentation de la triangulation de Delaunay. Enfin nous avons vu les différents algorithmes de restitution de contours d'un objet à partir une triangulation de Delaunay.

L'algorithme définie, Graphe de Gabriel consiste seulement à reconstruire le contour d'un objet il ne peut pas trouver l'intérieur et déterminer les surfaces cruse.

Puis le temps d'exécution de l'algorithme de crust dépend de l'ordre des points d'entrée.

C'est pourquoi il nous a paru intéressant de voir le troisième algorithme celle de l'alpha-Shape, qui se chargeait de raccorder les points de manière à retrouver la forme originelle pour qu'on puisse l'afficher.

Le chapitre suivant est dédié à la conception de notre application.

Chapitre II

Conception de l'application

Chapitre II : Conception de l'application.

1. Introduction

L'objectif de la reconstruction des frontières d'objet en trois dimensions à partir d'une triangulation de Delaunay est de produire une représentation fidèle de l'objet. Cette représentation est une description de la peau de l'objet sous forme de modèle triangulé.

Toutes les méthodes de reconstruction d'objet, souffrent d'insuffisance aux niveaux des jonctions des surfaces frontières de l'objet 3D.

Dans cette partie, une approche visant à améliorer la qualité des frontières de l'objet est présentée. Dans un premier temps, la méthode proposée, est détaillée à travers des algorithmes, décrivant les différentes étapes suivies pour résoudre le problème de l'insuffisance de la représentation des facettes et des frontières de l'objet 3D reconstruit.

2. Architecture globale du système

Le travail concerne la reconstruction des frontières d'objets 3D à partir d'une triangulation de Delaunay. La reconstruction consiste à modifier la triangulation de Delaunay 3D en utilisant la méthode Alpha Shape. L'objectif final est la génération d'un modèle STL exploitable par les applications de l'usinage de surfaces discrètes.

L'architecture générale de l'application est divisée en sept 08 parties illustrées par l'organigramme suivant (Figure 26) :

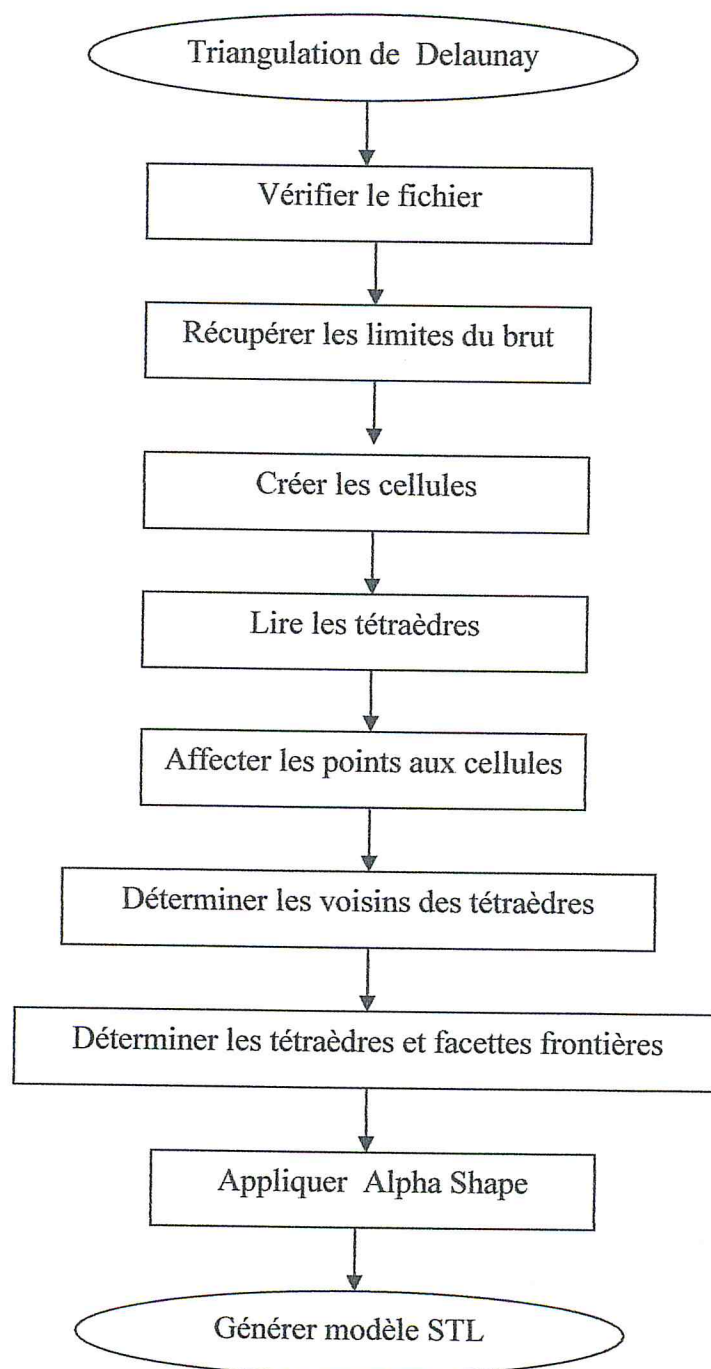


Figure 26 : Organigramme de l'Architecture générale.

2.1. Vérifier du fichier

La vérification du fichier d'entrée consiste à vérifier la conformité syntaxique du fichier de base et leur extension. Si le fichier n'est pas valide, la triangulation n'est pas traité (Figure 27). La procédure de vérification est illustrée ci-dessous :

1. Sélection d'un fichier texte.

Conception de l'application

- Vérification du fichier s'il contient des caractères non conformes (informations autres que les coordonnées des points):
 - Si oui, le fichier n'est pas valide.
 - Si non, lecture du fichier:

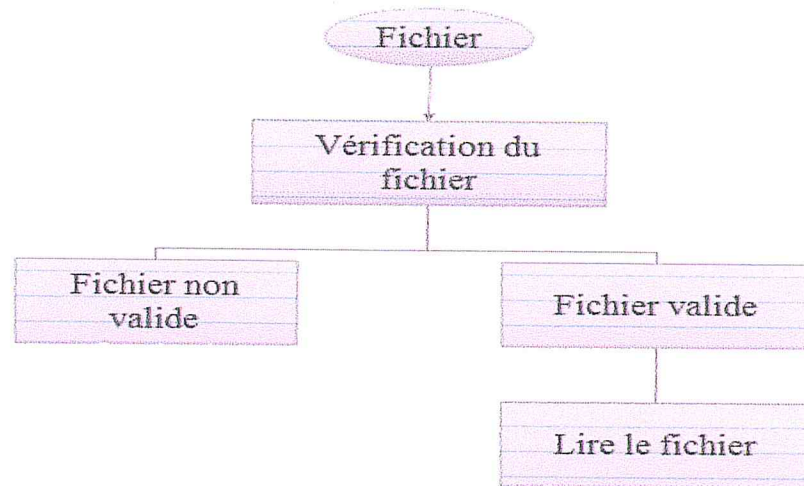


Figure 27: Organigramme de vérification du fichier.

2.2 Récupérer les limites du brut

- Lire les limites du brut.
- Affecter les coordonnées selon l'ordre (X_min, X_max, Y_min, Y_max, Z_min, Z_max).

2.3 Créer les cellules

L'objectif de cette fonction, est l'optimisation du temps de reconstruction des surfaces. La figure suivante illustre ses principales étapes.

- Récupération des dimensions du brut (X_min, X_max, Y_min, Y_max, Z_min, Z_max).
- Calcul des dimensions du brut à savoir ; la hauteur, la longueur et la largeur.
 - longueur = get_x_max()- get_x_min().
 - largeur = get_y_max()- get_y_min().
 - hauteur = get_z_max()- get_z_min().
- Subdivision de la hauteur, de la longueur et de la largeur par le nombre de cellules pour calculer les pas suivant les trois axes X, Y et Z.
 - pas_x= longueur/getnbr_cellule_x();

```
pas_y= largeur/getnbr_cellule_y());
```

```
pas_z= hauteur/getnbr_cellule_z());
```

4. Création des cellules et leurs enregistrements dans un tableau dynamique à trois dimensions.

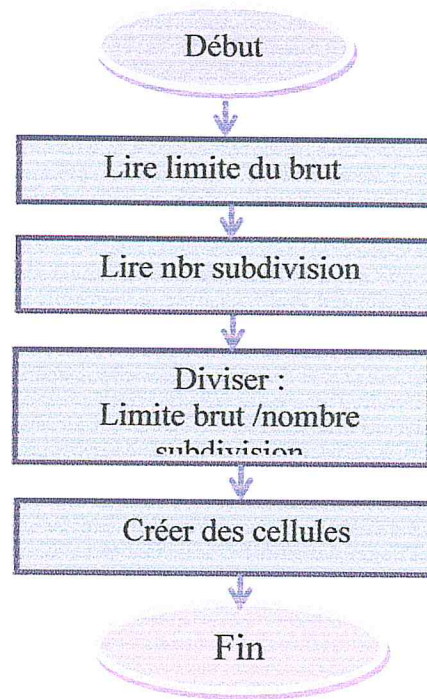


Figure 28 : Organigramme de création des cellules.

Le processus de création des cellules est illustré ci-dessous :

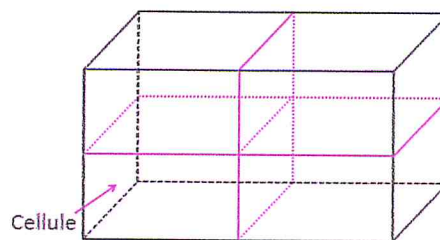


Figure 29: création des cellules.

2.4 Lire les tétraèdres

- Récupérer les coordonnées des points (x, y, z) des tétraèdres.
- Affecter chaque trois valeur à un point de type «Point_frontière».

Conception de l'application

- Stocker le point dans un tableau « tableau_point ».
- affecter chaque trois point à une facette de type « facette_frontière »
- Stocker la facette dans un tableau « tableau_facette ».
- affecter chaque quatre points dans à un tétraèdre de type « tétraèdre_frontière »
- Enregistrement du tétraèdre dans un tableau « tableau_tétraèdre ».

2.5 Affecter les points aux cellules

L'affectation des points aux cellules consiste à déterminer pour chaque cellule les points appartenant à l'intérieur de son enveloppe en se basant sur des considérations purement géométriques (Figure30).

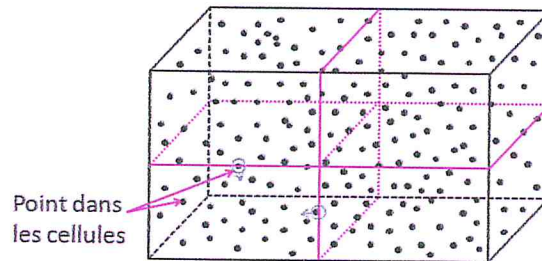


Figure 30: Affectation des points aux cellules.

La procédure est illustrée ci-dessous :

- Parcourir le tableau des cellules.
Pour chaque cellule récupérer les points.
 - Vérifier les points doubles.
Affecter le point à la cellule selon i,j,k .

2.6 Détermination des voisins des tétraèdres

Chaque tétraèdre est caractérisé par :

1. Quatre (04) points et chaque trois point représentent une facette (Figure a).
2. Entre un (01) et quatre (04) voisin (voisin 1, 2, 3 et 4) (Figure b).
3. Quatre (04) normales dirigées vers l'extérieur (normalisée). (Figure c)

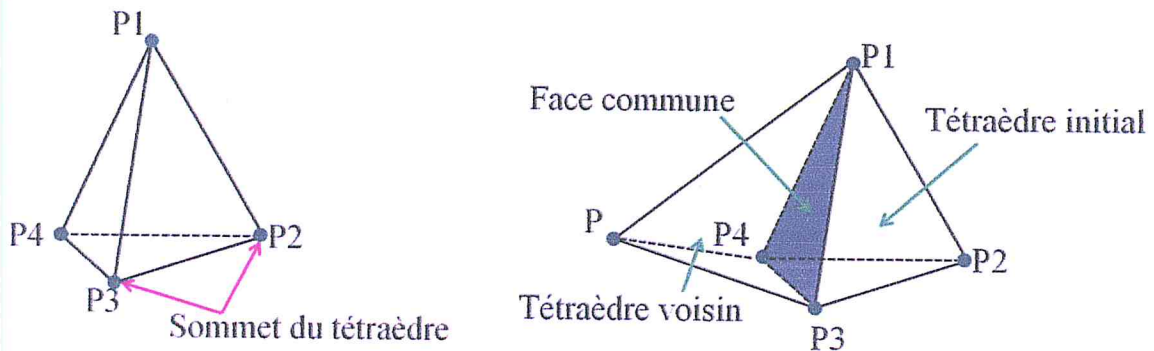


Figure a : Sommets d'un tétraèdre. Figure b : Représentation d'un tétraèdre voisin

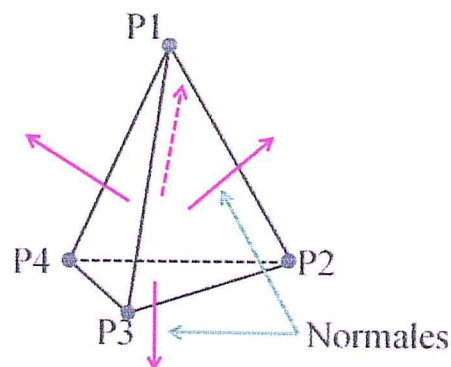


Figure c : Normales d'un tétraèdre.

Figure 31: Caractéristique d'un tétraèdre.

La procédure de Détermination des voisins de tétraèdre est décrite ci-dessous :

1. Pour chaque tétraèdre (A) d'indice j
 - Récupérer les points
 - Pour chaque point
 - Récupérer tab_indice tétraèdre
 - Récupérer le tétraèdre (B).
 - Récupérer les facettes de tétraèdre.
2. Comparer chaque facette de tétraèdre (B) avec les facettes de tétraèdre (A)
 - Si facette de (A) = facette de (B) Alors le tétraèdre B est voisin de A.
3. Si la facette numéro (1) est la facette commune alors (B) est voisins1.
4. Si la facette numéro (2) est la facette commune alors (B) est voisins2.
5. Si la facette numéro (3) est la facette commune alors (B) est voisins3.

Conception de l'application

6. Si la facette numéro (4) est la facette commune alors (B) est voisins4.

2.7 Déterminer tétraèdres et facettes frontières

Cette fonction permis d'identifier tous les tétraèdres et les facettes frontières de l'objet.

La procédure de Déterminer les voisins de tétraèdre est décrite ci-dessous :

- Récupérer les tétraèdres de tab _ tétraèdre.

Pour chaque tétraèdre récupérer ses voisins.

- Si un de ses voisin = -1 alors ajouter au tableau des tétraèdres frontières

Récupérer le voisins telle que voisin = -1.

Ajouter facette frontière.

2.8 Appliquer Alpha Shape

La procédure d'application d'Alpha Shape est illustrée ci-dessous :

- saisie la valeur de alpha souhaité.
- Pour chaque facette frontière
 - Calculer le centre du cercle circonscrit de la facette qui est l'intersection des trois diamètres de la facette.

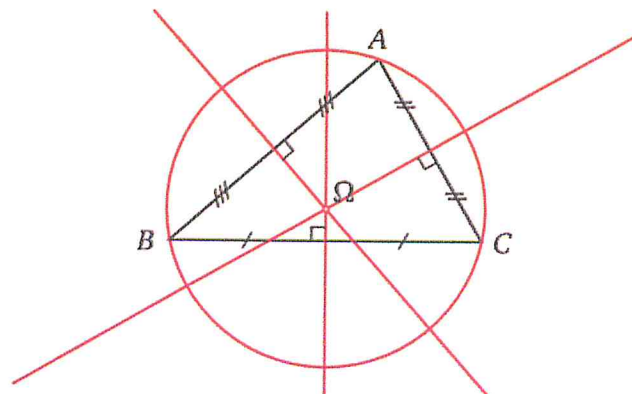


Figure32 : Exemple de centre de la sphère circonscrite d'un triangle.

Une fois le cercle paramétré, on procède comme suit :

- calcul du rayon du cercle circonscrit de la facette.
- Affectation de la valeur alpha au rayon sphère de la facette.
- Détermination du centre de sphère de la facette :

1. Trouver la ligne perpendiculaire à la facette (parallèle à la normale de la facette) et passant par le centre du cercle circonscrite.
2. Sur cette ligne, trouvez un point M du côté de la normale et distant de la valeur de alpha par rapport au centre du cercle circonscrite, ce point représente le centre de la sphère tel que :

$$M_x = \text{centre_cercle_circon.getx()} + \alpha * \text{normal.getx()}$$

$$M_y = \text{centre_cercle_circon.gety()} + \alpha * \text{normal.gety()}$$

$$M_z = \text{centre_cercle_circon.getz()} + \alpha * \text{normal.getz()}$$

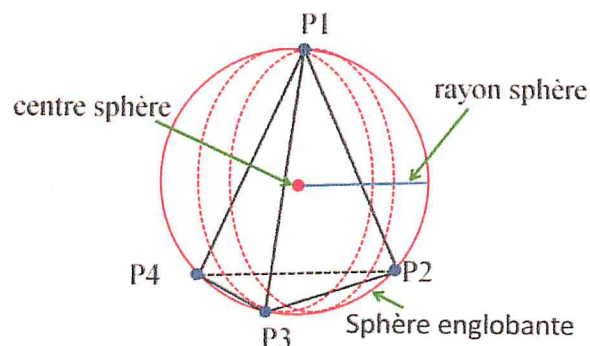


Figure33 : Sphère circonscrite, rayon et centre de la sphère.

2.9 Modifier la Triangulation Delaunay

Cette procédure permet de faire une mise à jour de la triangulation après l'application de la méthode Alpha Shape.

La procédure des modifications de Triangulation Delaunay est décrite ci-dessous :

- Calculer la distance entre le centre de la sphère et les points.
- Comparer le rayon de la sphère avec la distance calculé :
 - Si rayon de sphère < distance :
 - ✓ Supprimer la facette frontière.
 - ✓ Supprime le tétraèdre frontière de la facette frontière.
- Vérifier si la sphère de la facette est vide :
 - ✓ Si oui, garder la facette et le tétraèdre.
 - ✓ Sinon supprimer le tétraèdre frontière de la facette.

- Si rayon de sphère > distance :
 - ✓ Garder la facette et le tétraèdre frontière.

2.10 Générer fichier STL

Une fois la modification de la triangulation du modèle de l'objet digitalisé est validée, le modèle STL de l'objet est généré. C'est l'ultime phase de l'approche proposée. Elle constitue l'objectif principal de l'application (Figure 34):

1. Recherche des faces dont les voisins sont nuls (le tétraèdre ne partage cette face avec aucun autre tétraèdre).
2. Sauvegarde de la face identifiée.
3. Sauvegarde des faces en suivant la structure du fichier STL.
4. Visualisation du modèle STL.

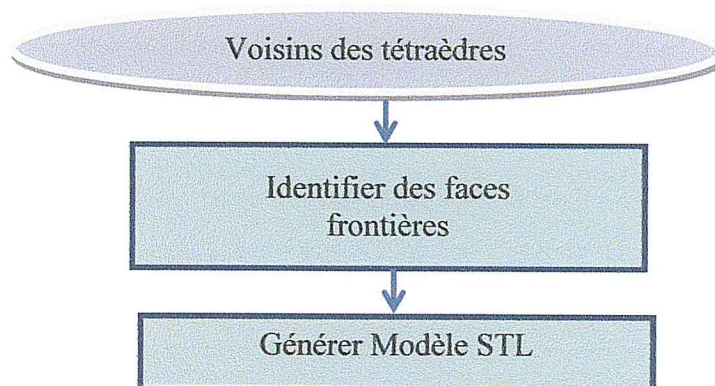


Figure34 : Organigramme de génération du fichier STL.

3. Conclusion

Dans ce chapitre, nous avons défini la conception de notre application en présentant les différents objectifs à atteindre et tous les détails nécessaires à sa mise en œuvre avec les organigrammes et algorithmes utilisés dans le développement de ce travail.

Dans le chapitre suivant, nous allons présenter l'implémentation des solutions proposées ainsi que la validation du travail réalisé à travers un exemple concret de pièce.

Chapitre III

Implémentation et validation

1. Introduction

Après avoir exprimé les objectifs de l'application à développer, la mise en œuvre de la proche proposée vient pour conclure ce travail. Ainsi ce chapitre a pour objectif de donner une vue assez complète du travail réalisé en présentant les informations manipulées à travers les sessions d'utilisation développées.

La première partie du chapitre, est réservée à la présentation du langage utilisé pour implémenter l'application ainsi que les différentes fenêtres de l'application. Quand à la deuxième partie, La validation du travail effectué est vérifier à travers un exemple concret d'une pièce triangulé avec la méthode de Delaunay et raffiner par l'approche proposée dans ce travail.

2. Implémentation

La réalisation des solutions proposées a été faite par l'utilisation d'Embarcadero C++Builder 10 Seattle. 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.

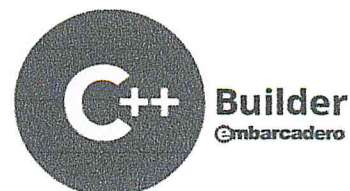
2.1 Langage de programmation utilisé

Afin de se conformer à la majorité des système de CFAO, et afin de permettre l'intégration de notre application dans l'environnement de reconstruction d'objet développée par l'équipe CFAO du CDTA, le travail réalisé a été implémenté en utilisant le langage C++.

Crée en 1983 par Bjarne Stroustrup, 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 de logiciels. En tant qu'héritier du langage C, le C++ est d'une grande efficacité. De plus, il a des fonctionnalités puissantes, comme par exemple la notion de classe, qui permet d'appliquer les techniques de la programmation orientée objet.

2.1.1 Embarcadero c++ builder

Embarcadero C++ Builder toute la puissance du langage C++ oriente objet, il offre la possibilité de développer rapidement des applications (RAD) sous Windows grâce à ses

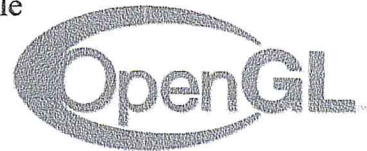


Implémentation et validation

différentes bibliothèques. Il permet la création instantanée des interfaces utilisateurs pour les applications desktop, applications mobiles, ... etc, car il offre une gestion de l'interface. Le compilateur C++ qui est inclus "Borland C++ Compiler" est un compilateur et un optimiseur de code, de haut rendement et multithread, qui agit en second plan. Sans arrêter le travail, les applications se compileront et s'exécuteront plus rapidement.

2.1.2 OpenGL

Open Graphic Library ou OpenGL**** OpenGL est le premier environnement pour le développement d'applications graphiques 2D et 3D portables et interactives. Depuis son introduction en 1992.



OpenGL est devenu l'interface de programmation d'applications graphiques (API) 2D et 3D la plus utilisée et la plus prisée de l'industrie, apportant des milliers d'applications à une grande variété de plates-formes informatiques. OpenGL favorise l'innovation et accélère le développement d'applications en intégrant un large éventail de fonctions de rendu, de mappage de texture, d'effets spéciaux et d'autres puissantes fonctions de visualisation.

3. Présentation des fenêtres

L'application principale est composée d'un environnement qui englobe dans ses onglets toutes les applications du projet.

La fenêtre principale de notre projet est composée de quatre (04) onglets. Chaque onglet est dédié à une des phase(s) du processus de reconstruction des frontières d'objet 3D à partir une triangulation de Delaunay :

- Onglet 1 : Création des cellules.
- Onglet 2 : Lecture des points.
- Onglet 3 : Voisins des tétraèdres.
- Onglet 4 : Alpha Shape.

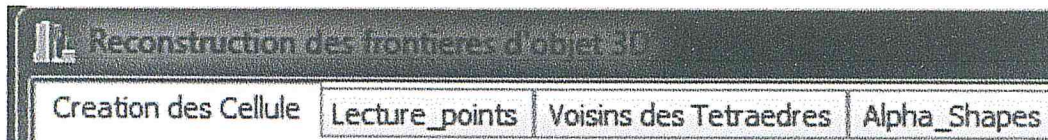


Figure 35 : Onglets de la fenêtre principale.

3.1 Création des cellules

Le premier onglet « Création des cellules » (Figure) est activé au lancement de l'application, il est divisé en deux parties :

La première partie, permet la lecture de la première ligne du fichier de triangulation contenant les limites du brut par un simple clic sur le bouton « Lecture limite ». par la suite, les paramètres du brut (X_{min} , X_{max} , Y_{min} , Y_{max} , Z_{min} , Z_{max}) sont affichés.

La seconde partie est consacrée à la création des cellules dans les trois plans XY, XZ et YZ, après la saisie du nombre de subdivision voulu. Une fois la création est terminée, l'utilisateur peut visualiser la subdivision des cellules en filaire et en rendu.

Après l'affichage des résultats, l'utilisateur peut passer au deuxième onglet pour effectuer la lecture des points.

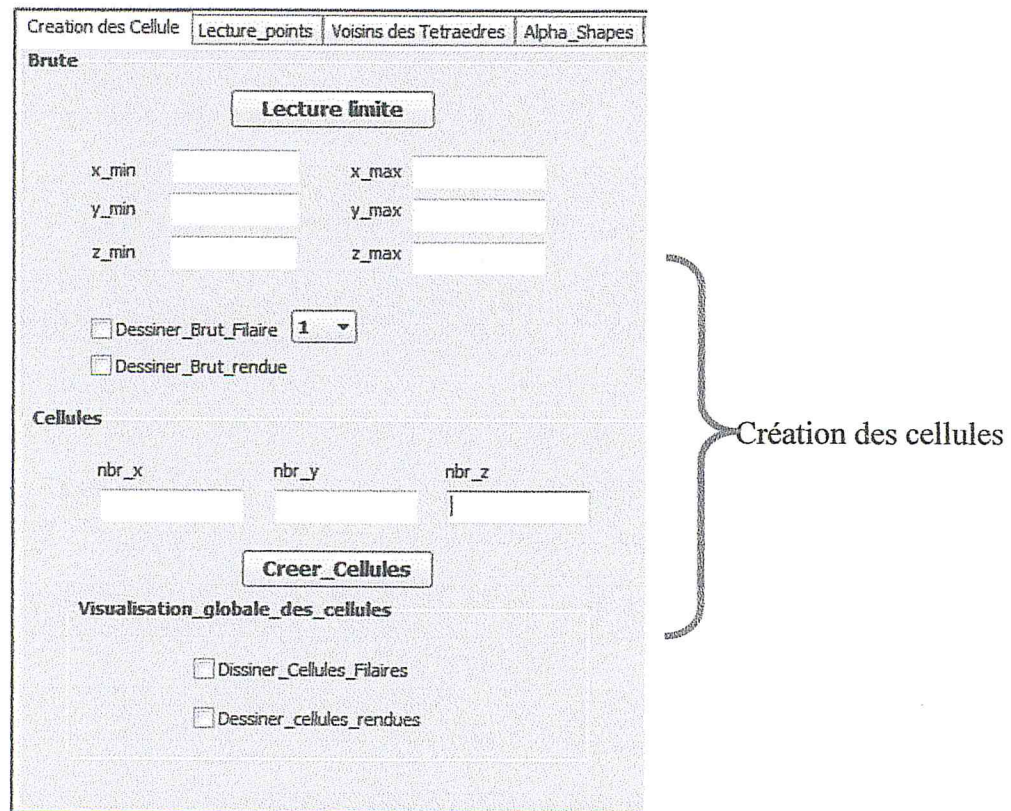


Figure 36 : Onglet création des cellules.

3.2 Lecture des points

Cet onglet est consacré premièrement à lecture des points du fichier telque chaque ligne contient les points representent un tetraedre en plus de l'affectation des points aux tétraedres et aux facettes et l'affectation des facettes aux tetraedres. Apres cette lecture, le nombre de points du fichier est affiché et l'utilisateur peut visualiser :

- ✓ les cellules en filaire et en rendu selon les X, Y et Z.
- ✓ les points d'une/des cellule(s).
- ✓ Tous les points du modèle triangulé.
- ✓ La tétraédrisation globale du modèle triangulé en filaire et en rendu.
- ✓ Les tétraèdres partagent chaque point.

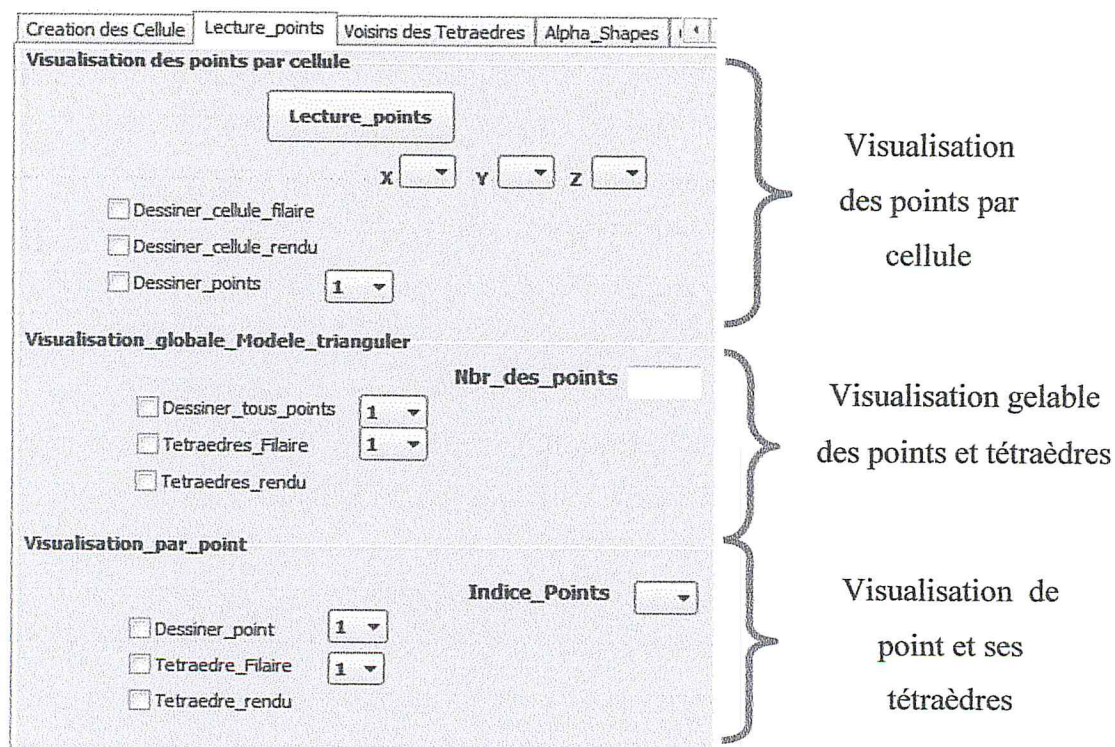


Figure 37: Onglet lecture des points.

Une fois les résultats sont obtenus, l'utilisateur peut passer au troisième onglet pour déterminer des voisins des tétraèdres.

3.3 Voisins des tétraèdres

Cet onglet est consacré à déterminer des voisins des tétraèdres. Après la détermination des voisins, l'utilisateur peut visualiser pour chaque tétraèdre l'ensemble des tétraèdres voisins en filaire et en rendu, ainsi que chaque tétraèdre voisin à part.

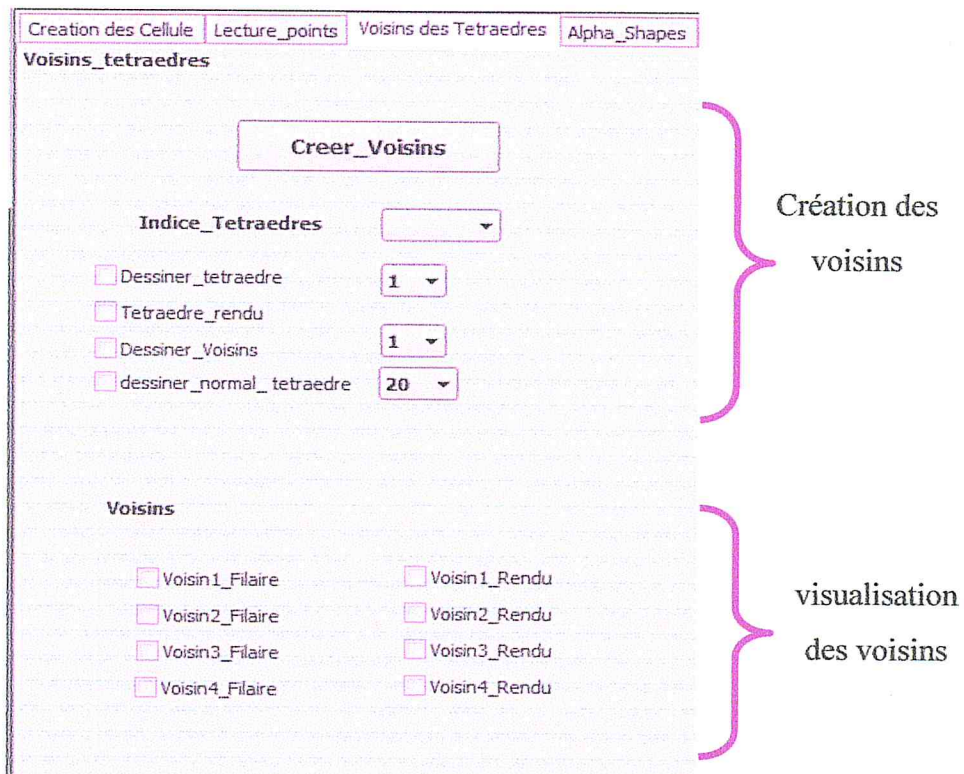


Figure 38: Onglet voisins de tétraèdre.

Une fois les résultats sont obtenue l'utilisateur peut passer au quatrième onglet pour faire la reconstruction des frontières.

3.4 Alpha Shape

Cet onglet dédié aux frontières du modèle triangulé, est divisé en deux phases :

La première est dédiée à la détermination des facettes et des tétraèdres frontières, et la deuxième à la reconstruction des frontières.

L'utilisateur peut visualiser globalement ou partiellement les tétraèdres et les facettes frontières en filaire et en rendu et il peut aussi ; pour chaque facette frontière visualiser :

- ✓ le tétraèdre de la facette en filaire et en rendu.
- ✓ Les points de la facette.
- ✓ Le centre et le cercle circonscrit.

Implémentation et validation

- ✓ La normale.
- ✓ La ligne perpendiculaire à la facette.
- ✓ Le centre et la sphère.

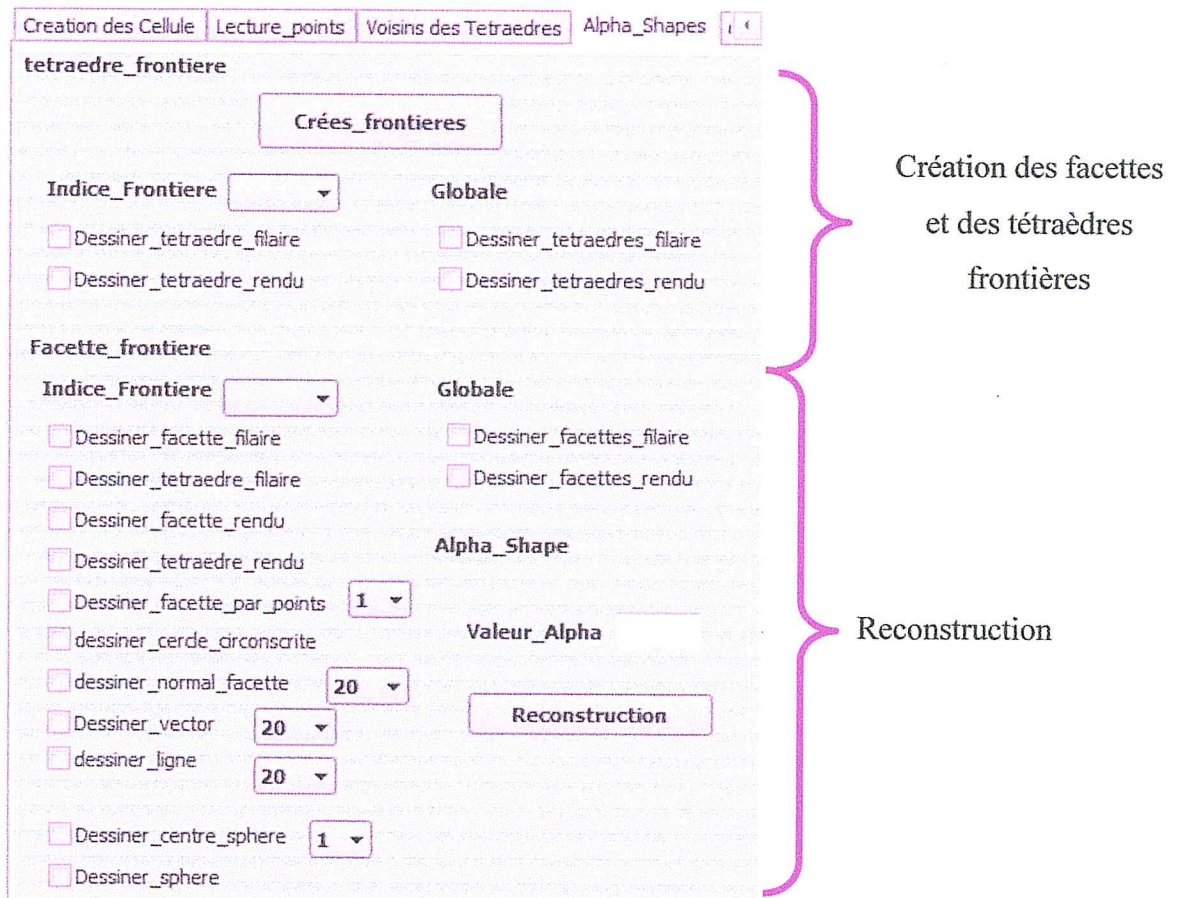


Figure 39: Onglet alpha shape.

4. Validation

La validation de notre travail est effectuée sur l'exemple de statut présenté ci-dessous. Le traitement effectué sur la triangulation Delaunay jusqu'à la génération d'un modèle STL représentant d'une manière plus ou moins fidèle l'objet cible est présenté dans la suite.

4.1 Lecture du fichier

Dans l'onglet lecture fichier Txt, l'utilisateur doit cliquer sur le bouton lecture limites pour pouvoir sélectionner un fichier (.txt) à ouvrir.

Ouverture de l'application et choix du fichier.

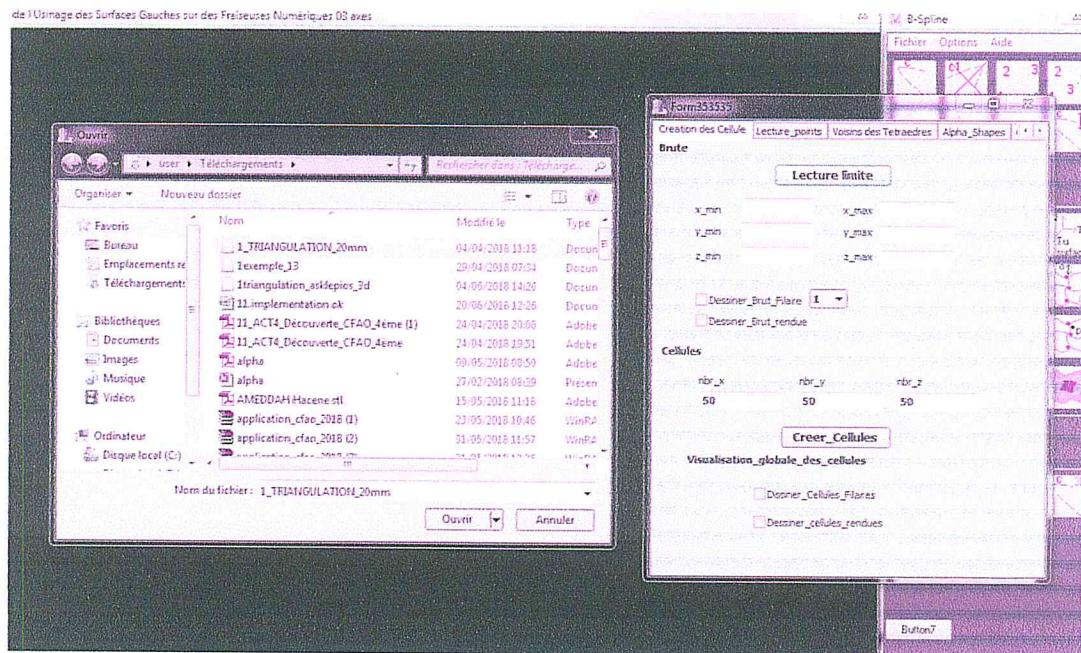


Figure 40: choisir un fichier.

- Les paramètres de brute sont :

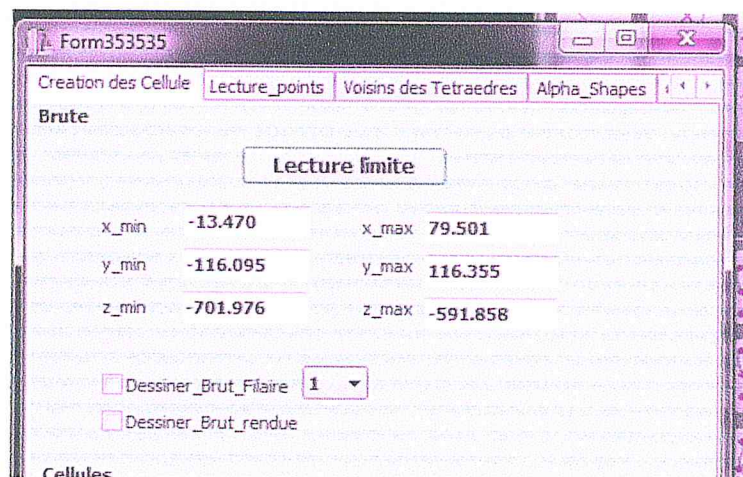
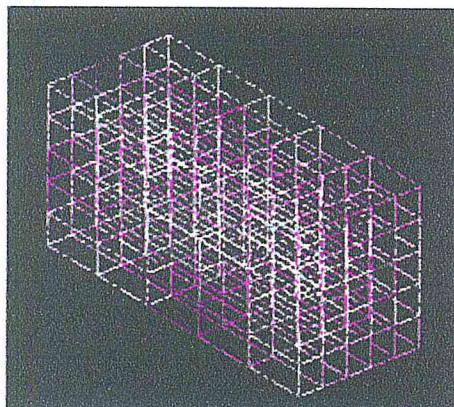


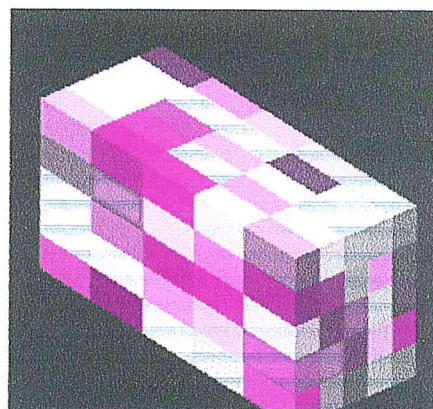
Figure41 : Paramètres du Brut.

4.2 Création des cellules

Après la lecture de fichier, l'utilisateur doit sélectionner le nombre de subdivision des cellules selon X, Y et Z. Le test a été effectué pour 5 cellules selon X, 5 cellules selon Y et 5 cellules selon Z.

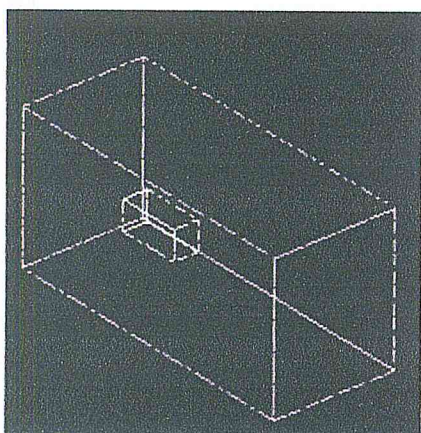


a. Cellules en filaire

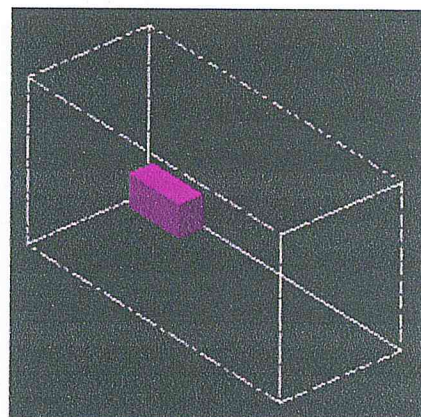


b. Cellules en rendu

Figure42: Visualisation de toutes les cellules.



a. Une cellule en filaire.



b. Une cellule en rendu.

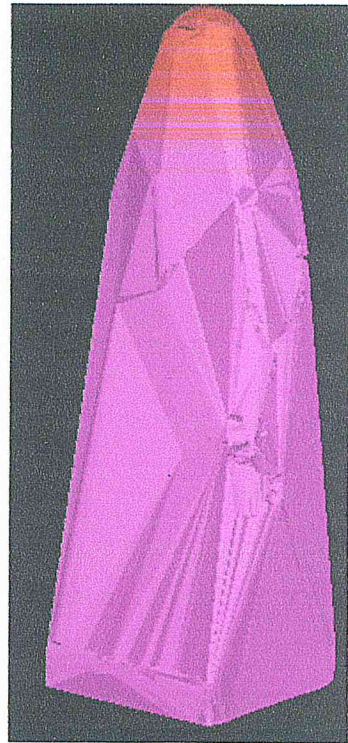
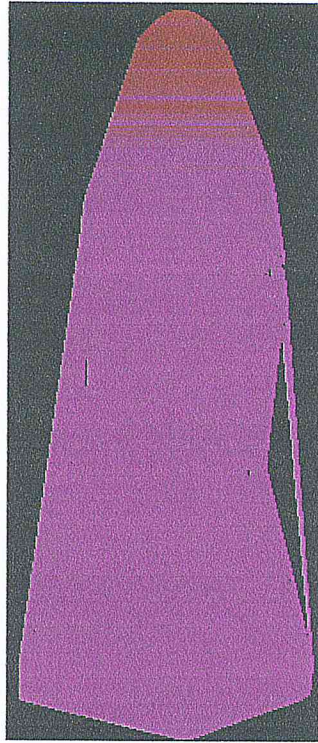
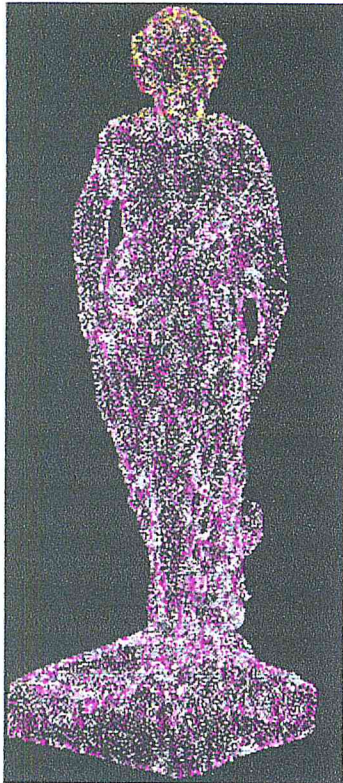
Figure43 : Visualisation d'une seule cellule.

4.3 Lecture des points

La triangulation de modèle contient 28689 points, alors 193895 tétraèdres.

L'utilisateur doit cliquer sur le bouton lecture points pour lire les points. Il peut visualiser :

- Le modèle triangulé globale en filaire et en rendu.
- Pour chaque point, les tétraèdres qui le partage.



a. Modèle filaire

b. Modèle en rendu.

Figure44: Visualisation des points globale.

Figure45: Visualisation de la Tétradratisation.

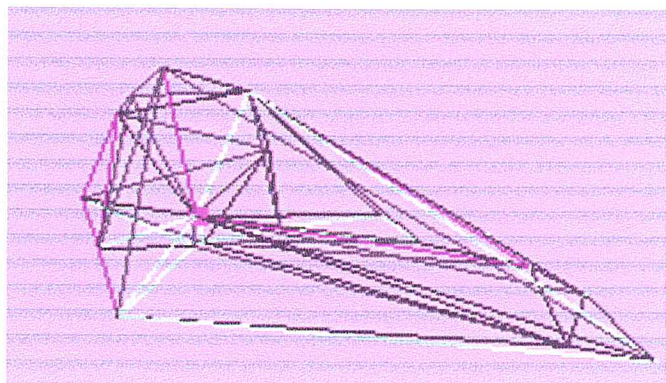
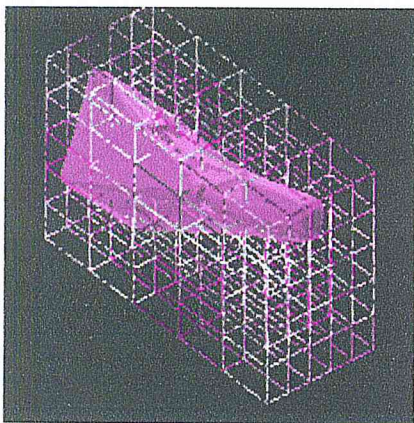
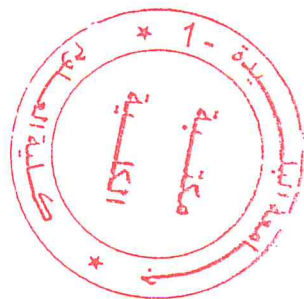


Figure46 : Le modèle dans les cellules

Figure47: Tétraèdres partageant un point commun



4.4 Création des voisins de tétraèdre

Dans cette étape, les voisins de chaque tétraèdre sont créés. Pour cela, l'utilisateur doit cliquer sur le bouton `Crées voisin`. L'utilisateur peut visualiser l'ensemble des voisins obtenu en filaire, en rendu et les normales de tétraèdre.

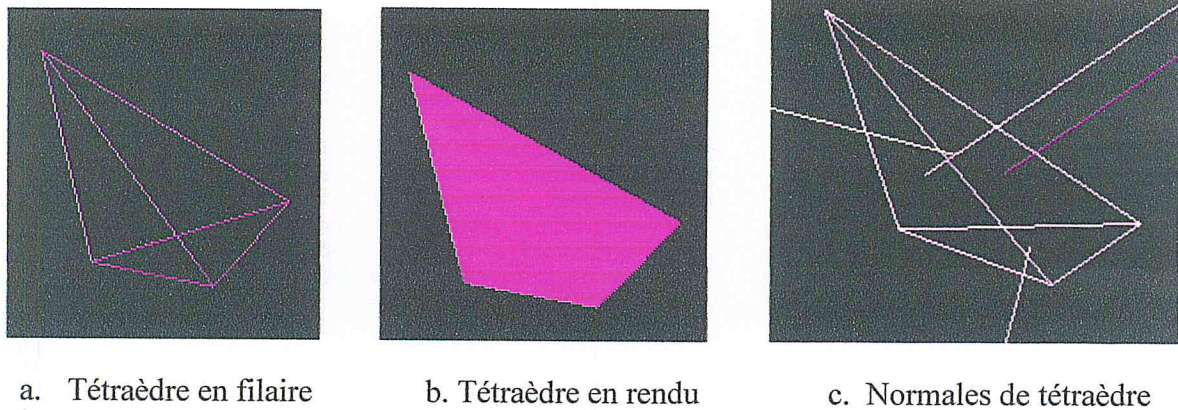


Figure48: Visualisation de tétraèdre.

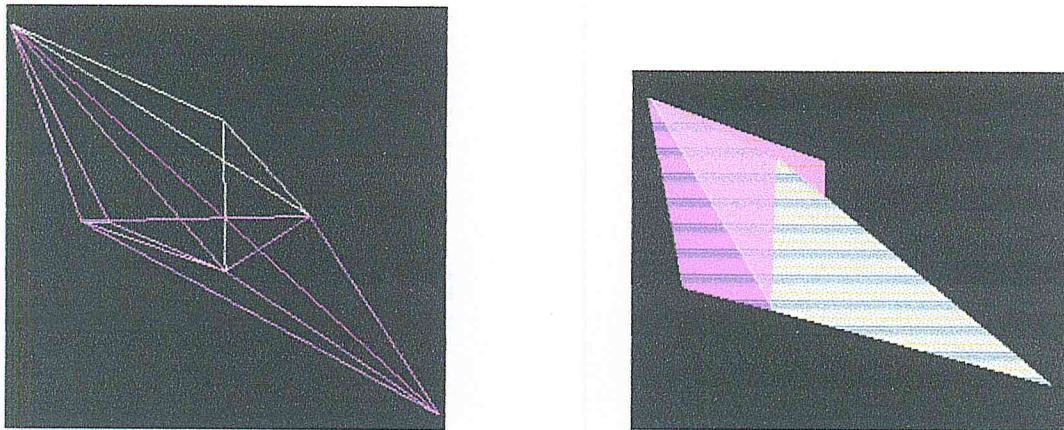


Figure 49: Voisins de tétraèdre.

4.5 Détermination des facettes et tétraèdres frontières

Dans cette étape, on procède à la détermination des facettes et des tétraèdres frontière pour appliquer Alpha

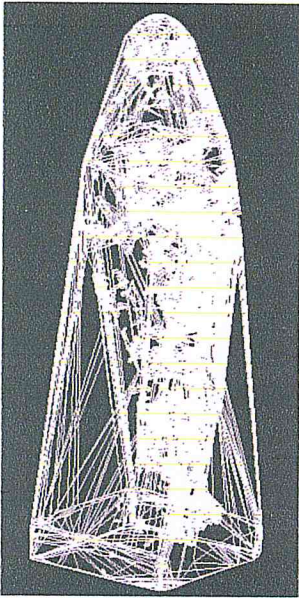


Figure50: Facettes frontières en filaire.

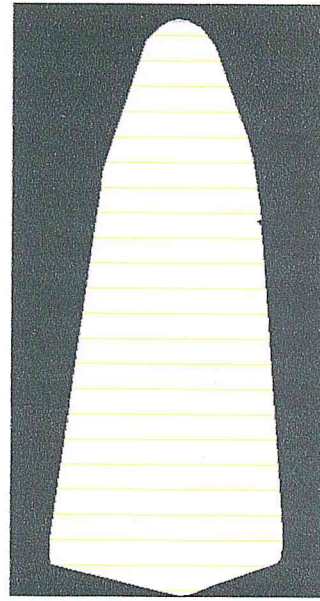


Figure51 : Facettes frontières en rendu.

Pour chaque facette frontière on peut visualiser sa normale, son cercle circonscrit et sa sphère.

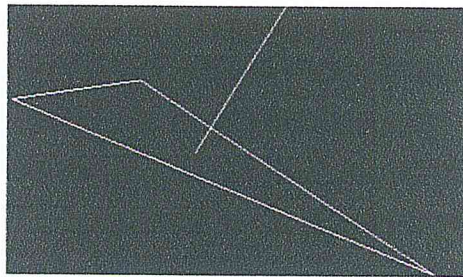


Figure52 : Normale de Facette frontière.

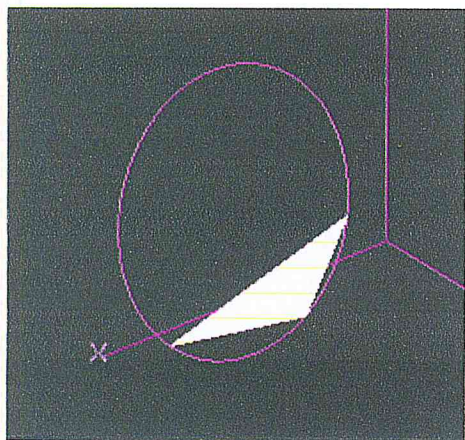


Figure53 : Cercle de Facette frontière.

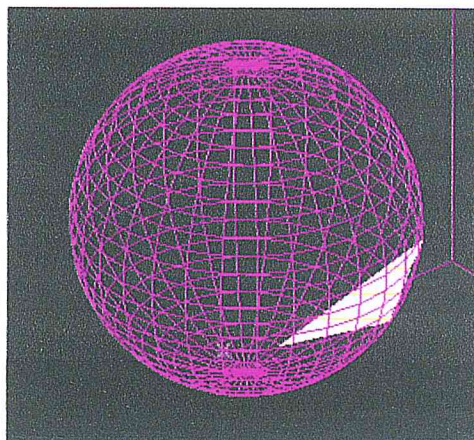


Figure54: sphère de Facette Frontière.

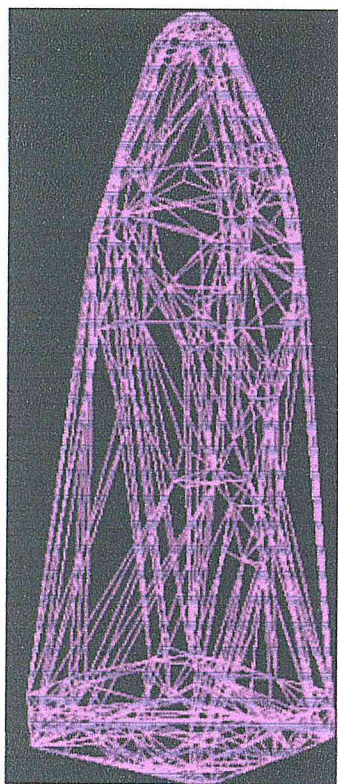


Figure55 : tétraèdres frontières en filaire.



Figure56 : tétraèdre frontières en rendu.

4.6 Application de Alpha Shape

Après L'extraction de toutes les facettes frontières, on applique le rayon alpha. Pour tester la méthode Alpha Shape.

Pour des raisons techniques (performance du PC) le nombre de points a été diminué de 28689 points jusqu'à 864 points. Dans cette partie, le mode d'exécution automatique est adopté. L'utilisateur saisie la valeur de alpha souhaitée ensuite il active le bouton Reconstruction.

Plusieurs valeurs d'alpha ont été testées pour valider l'Alpha Shape. Par visualisation on peut remarquer à partir de nuage des points que la valeur optimale d'alpha pour cet exemple est 12.



Figure57 : Alpha = 5.



Figure58 : Alpha = 9.5

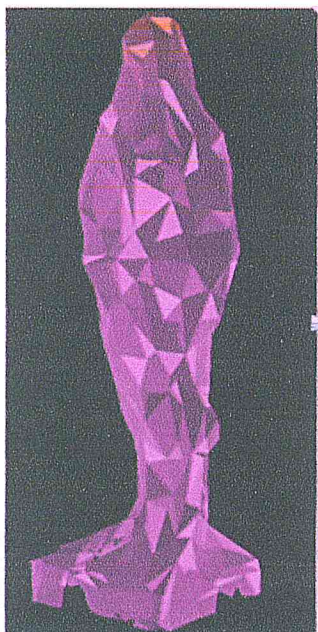


Figure59 : Alpha=12.



Figure60 : Alpha =20.

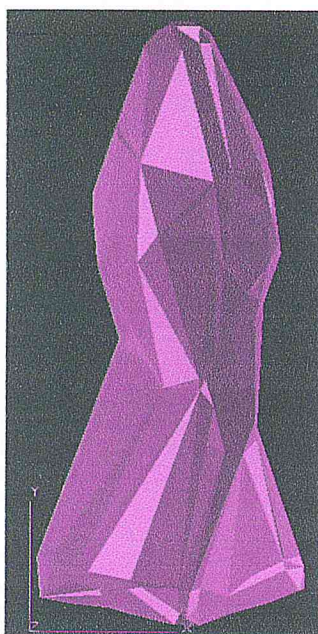


Figure61 : Alpha =60.



Figure62 : Alpha =100.

5. Conclusion

Dans ce chapitre, les différentes étapes de notre application logicielle ont été présentées et le travail réalisé a été validé à travers un exemple concret de statut obtenu par la triangulation Delaunay. L'application a été testée depuis la lecture du fichier (.txt) jusqu'à la modification de la triangulation de Delaunay.

Les performances du PC sur lequel a été implémentée l'application, a constitué un obstacle pour la validation du travail à travers une triangulation Delaunay avec un nombre très important de points. Néanmoins, l'approche a été validée à travers un exemple concret ce qui constitue un Maillon de plus dans l'environnement de reconstruction d'objets développé par l'équipe de CFAO du CDTA.

Conclusion générale:

Le travail réalisé au sein de l'équipe CFAO de la Division Productique et Robotique du CDTA s'articule autour de développement d'une application logicielle graphique et interactive pour la reconstruction des frontières d'objet 3D à partir une triangulation de Delaunay. Il s'agit précisément de la réalisation d'une application logicielle pour reconstruire les objets par des éléments géométriques simples (triangle) en utilisant la méthode alpha Shape.

Lors de la réalisation de ce projet, nous avons commencé par une étude bibliographique sur le processus de CFAO, le format d'échange de données « STL », les différentes méthodes de modélisation de surface, le processus du Reverse Engineering et plus précisément au niveau de la phase de reconstruction d'objets 3D et présenté les différentes méthodes d'implémentation de Delaunay.

Le deuxième chapitre, a été réservé à la conception de notre application (les algorithmes développés), le troisième est l'implémentation informatique de notre application logicielle ainsi que des tests de validation.

Nous avons conçu réaliser une application logicielle qui prend en charge : La reconstruction des frontières d'objet 3D en utilisant la technique de Alpha Shape.

Alpha-Shape, se chargeait de raccorder les points de manière à retrouver la forme originelle pour qu'on puisse l'afficher.

Seulement, après des tests sur divers objets échantillonnés, et après plusieurs essais de valeurs pour l'alpha-Shape, cette méthode s'est avérée assez peu fidèle même si la forme générale était respectée.

En vue d'améliorer notre travail, pour cela nous proposons les perspectives suivantes :

- Trouver une méthode permet de déterminer la valeur optimale de Alpha.
- Reconstruction des frontières d'objet 3D par la méthode Crust.
- Reconstruction des frontières d'objet 3D par la méthode Graphe de Gabriel.

References Bibliographique:

- [1] K. Bouhadja, "Fabrication des surfaces de forme gauche". Rapport de recherche CFAO/CDTA. 2013
- [2] M.E.H.Bendifallah. « Approche pour la Finition des Surfaces Gauches par la Stratégie 'Gradient Conjugué' à Partir de Modèle STL ». Mémoire de Master, Faculté Des Sciences et Sciences de L'ingénieur- Université Amar Telidji Laghouat, 2011.
- [3] Farin, G. E., Hoschek, J., and Kim, M.-S. Handbook of Computer Aided Geometric Design. North Holland. (2002).
- [4] J-C. LEON, Modélisation et construction de surfaces pour la CFAO, Hermès, Paris, 1999
- [5] [Chambelland, 2006] Chambelland, J. C. (2006). Reconstruction de surfaces à pôles pour la conversion et la simplification de modèles B-Rep. PhD thesis, Université de la Méditerranée.
- [6]. S.Kumar et D. Manocha, Efficient rendering of trimmed NURBS surfaces, Computer Aided Design, 27(7): 509-521, 1995
- [7] Warren, J. and Weime, H. Subdivision methods for geometric design: A constructive approach. The Morgan Kaufmann Series in Computer Graphics. (2001).
- [8] Saffar KH & Zekkari S, Rapport conception et développement d'une application distribuée pour Reconstruction de surface d'objet 3D à partir d'un Nuage de Points par la méthode de Flip.. Master 2 USDB. (2017).
- [9] Schulte Richard M., Padmanabhan Srikanth, Devgun Mohan S., Feature-driven, processbased approach to the integration of CAD/CAM in wireframe modèles, Internationale Journal of Production,
- [10] AMEDDAH, Hacène. *Modélisation CAO et Stratégies d'usinage pour la réalisation des surfaces à géométrie compliquée (Surfaces Libres)*. Thèse de doctorat. Université Mustapha Ben Boulaid Batna 2. 2013.
- [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] Rémi ALLÈGRE, These de doctorat, "Contributions à l'introduction de l'expressibilité dans la reconstruction et l'édition de modèles 3D", l'université Claude Bernard Lyon 1 decembre 2006, p.2-32.

- [14] Challali, M. O., Belaidi, I., Mohammedi, K., Belaidi, A., & Ishiomin, G. Application et adaptation de la triangulation de Delaunay pour la reconstruction de surfaces. Surface reconstruction using and adapting Delaunay triangulation.
- [15] 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.
- [16] Zouaoui, O & Sissani, D, (2016). Rapport conception et développement d'une application distribuée pour la reconstruction des modèles 03D à partir d'un nuage de points quelconque. Master 2 USDB.
- [17] DE JESUS TRUJILLO-ROMERO, Felipe. *Modélisation et reconnaissance active d'objets 3D de forme libre par vision en robotique*. 2008. Thèse de doctorat. Institut National Polytechnique de Toulouse-INPT.
- [18] AbdErahim, L & Zeghoudi, S. (2011). Décomposition d'objet 3d composé de mailles triangulaires par la courbe directe.
- [19] 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.
- [20] Si, Hang. TetGen, a Delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software (TOMS)*, 2015, vol. 41, no 2, p. 11.
- [21] P.L.George, H. Bourouchaki. Triangulation de Delaunay et maillage, application aux éléments finis. Editions Hermes, Paris 1997.p 7-15
- [22]: Gold, H. L. C. (2005). La modélisation de données océanographiques à l'aide du diagramme de Voronoï tridimensionnel. *Revue internationale de géomatique*. Volume, 1(1)..
- [24] MATULA, David W. et SOKAL, Robert R. Properties of Gabriel graphs relevant to geographic variation research and the clustering of points in the plane. *Geographical analysis*,
- [25] L. U. Claude and B. Lyon, "L'université claudes bernard – lyon 1," 2006.
- [26] [P. Alliez, "1 Iteration de Lloyd en 2D 3 Diagramme de Voronoi borné 2D," pp. 1–5.
- [27] S. Linux, "Reconstruction & G . A . Crust - Courbes / Surfaces isovaleur," pp. 2011–2012, 2012.
- [28] N. Amenta and U. T. Austin, "A New Voronoi-Based Surface Reconstruction Algorithm."

[29] EDELSBRUNNER H., KIRKPATRICK D. G., SEIDEL R.: On the shape of a set of points in the plane. *IEEE Trans. Inform. Theory* IT.29 (1983), 551-559.

[30] [Edelsbrunner et al., 1983, Mucke, 1993, Edelsbrunner et Mucke, 1994]

[31] H. Edelsbrunner and P. M. Ernst, "Three-Dimensional α Shapes," vol. 13, no. September, 2005.

[32] F. Cazals, J. Giesen, M. Pauly, and A. Zomorodian, "Conformal alpha shapes," *Proc. Eurographics/IEEE VGTC Symp. Point-Based Graph. 2005.*, pp. 55–61, 2005.

[33] EDELSBRUNNER, Herbert et MÜCKE, Ernst P. Three-dimensional alpha shapes. *ACM Transactions on Graphics (TOG)*, 1994, vol. 13, no 1, p. 43-72.

[34] K. Fischer, "Introduction to Alpha Shapes," *Dep. Inf. Comput. Sci. Fac. Sci. Utr. Univ.*, pp. 1–17, 2000.

[35] H. Edelsbrunner and P. M. Ernst, "Three-Dimensional α Shapes," vol. 13, no. September, 2005.

[36] E. Fließwasser, "Surface Reconstruction with Alpha Shapes," no. December, 2015.

[37] J. Lundgren, "Alpha shapes," 2010.

[40] http://www.fabbers.com/tech/STL_Format

[41] 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.

[42] http://www.mathom.fr/mathom/sauvageot/Modelisation/Graphes/Voronoi_telephone.pdf

[43] <https://www-sop.inria.fr/prisme/fiches/Voronoi/index.html.fr>

