

Faculté des Sciences Exactes & de l'Informatique
Département de Mathématiques et d'Informatique
Filière Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : Systèmes d'information géographiques

Simplification du maillage triangulaire

Présenté par :

- Mlle. MESBAH Touatia.
- Mlle. BRAHIMI Soumia.

Encadré par:

- Mme. Soraya SIDI YKHLEF.

RESUME :

Le domaine des maillages est très vaste vu qu'on peut représenter chaque objet 3D en un maillage.

Il existe plusieurs applications des maillages parmi elles les Modèles Numériques de Terrain (MNT), qui est une représentation informatisée du terrain réel ce dernier présente des problèmes lors de la manipulation de ses données à cause du gros volume de ces dernières.

Dans ce mémoire, nous avons fait une étude approfondie de l'état de l'art des méthodes de simplification du maillage et nous avons pu implanter une application qui sert à simplifier un MNT (et un maillage triangulaire) par deux méthodes différentes afin de rendre ce dernier plus souple à manipuler dans divers applications grâce à sa taille diminuée.

Mot clés :

Tessellation, Maillage triangulaire, Algorithme géométrique, simplification de maillage.

SOMMAIRE :

| | |
|--|------------------------------------|
| Introduction générale | 5 |
| Chapitre I : maillage et triangulation | 7 |
| I.1. Définition de maillage | 8 |
| I.2. Les classes de maillages | 8 |
| I.3. Les maillages triangulaires | 9 |
| I.3.1. Définition de maillage triangulaire..... | Erreur ! Signet non défini. |
| I.3.2. Définition mathématique | 10 |
| Chapitre II : état de l'art sur la simplification | 12 |
| II.1. Définition de la simplification | 13 |
| II.2. Contrainte de la simplification..... | 13 |
| II.3. Méthodes de simplification..... | 14 |
| II.3.1. Décimation | 14 |
| II.3.2. Simplification par enveloppes | 15 |
| II.3.3. Regroupement de sommets | Erreur ! Signet non défini. |
| II.3.4. Re-échantillonnage (Re-tiling)..... | Erreur ! Signet non défini. |
| II.3.5. Décomposition en ondelettes..... | Erreur ! Signet non défini. |
| II.3.6. Contraction itérative | Erreur ! Signet non défini. |
| II.3.7. Fusion de régions (Region Merging) | 20 |
| II.3.8. Effondrement (Progressive mesh) | 22 |
| II.3.9. Filtres volumiques (voxelisation)..... | 23 |
| II.3.10. Simplification par préservation de l'apparence..... | 24 |

| | |
|--|------------------------------------|
| Chapitre III : mise en œuvre et implémentation..... | 26 |
| III.1.Introduction | 27 |
| III.2.Représentation de l'application | 27 |
| III.2.1. Intérêt du projet | 27 |
| III.2.2. Ressources utilisées | 27 |
| III.3. Conception..... | 29 |
| III.3.1. Architecture générale du programme | 29 |
| III.3.2. Diagramme de cas d'utilisation..... | 30 |
| III.3.3.Description des algorithmes..... | 30 |
| III.3.3.1. Triangulation de Delaunay (TD)..... | 30 |
| III.3.3.2. Méthodes de simplification utilisées | 32 |
| III.3.4.Structure de données utilisées | 35 |
| III.4.Implémentation | 37 |
| III.4.1.Interface principale | Erreur ! Signet non défini. |
| III.5.Conclusion | 48 |
| Conclusion générale | 49 |
| Bibliographie..... | 50 |
| Webgraphie | 52 |

INTRODUCTION GENERALE :

Les systèmes d'information géographique (SIG) regroupent différentes méthodes et technique informatiques, permettant de modéliser, de saisir sous forme numérique, de stocker, de gérer, de consulter, d'analyser, de représenter des objets ou des collections d'objets géographiques, avec la particularité essentielle de prendre en compte les caractéristiques spatiales de ces objets au même titre que les attributs descriptifs qui y sont attachés.

Les systèmes d'information géographique font appel à de nombreux domaines scientifiques et technique et à de nombreuses méthodes de simplification des géométries « Maillage triangulaire, particulièrement les Modèles Numérique de Terrain (MNT) ».

La modélisation du terrain (la réalisation des cartes, l'étude d'un terrain ou le déplacement d'un robot dans un monde extérieur) exige la connaissance de la variation de l'altitude dans un terrain. Le développement des techniques graphiques du virtuel inspiré du réel offre des outils de plus en plus performants permettant la visualisation des modèles numériques sur des écrans d'ordinateurs.

Ces modèles numériques sont des représentations synthétiques du monde réel qui nous entoure ou même de celui imaginaire inventé par l'esprit de l'homme. La prolifération des données géographiques ainsi que l'essor de l'informatique et des outils de construction de Modèles Numérique de Terrain (MNT) ont conduit à un usage très répandu des MNT dans des thématiques différentes.

Cette multiplicité des utilisations, basées sur l'extraction de différents objets et paramètres à partir du plan altimétrique, pose des problèmes de qualité, la définition de la qualité étant fondamentalement liée au besoin particulier d'un utilisateur. En ce qui concerne la qualité des MNT, celle-ci est habituellement exprimée par rapport à la seule altitude, par une mesure statistique des écarts altimétriques à une surface de référence.

Ces dernières années, de nombreux algorithmes de reconstruction ont été proposés dans le but d'extraire des approximations linéaires par morceaux (*i.e.*, des triangulations) directement à partir des nuages de points ou des images numériques. En général, ces maillages possèdent le même degré de complexité que les données d'origine.

Malheureusement, comme l'a bien exprimé *Luebke* [1], « le nombre de polygones désiré dépasse toujours le budget (de polygones) autorisé ». C'est pourquoi, la complexité des modèles polygonaux doit être réduite de manière drastique. Ce processus inclut notamment les techniques de simplification des maillages.

Ces méthodes numériques visent à réduire, autant que possible, le nombre de polygones du modèle, tout en préservant le degré de précision géométrique de celui-ci. En outre, des requis spécifiques peuvent être imposés sur les maillages résultants, concernant par exemple la qualité en forme et en taille des éléments.

Notre mémoire s'articule autour des chapitres suivants :

- Chapitre I : Généralités sur le maillage et la triangulation.
- Chapitre II : Etat de l'art sur la simplification.
- Chapitre III : Mise en œuvre et implémentation d'une application de simplification.

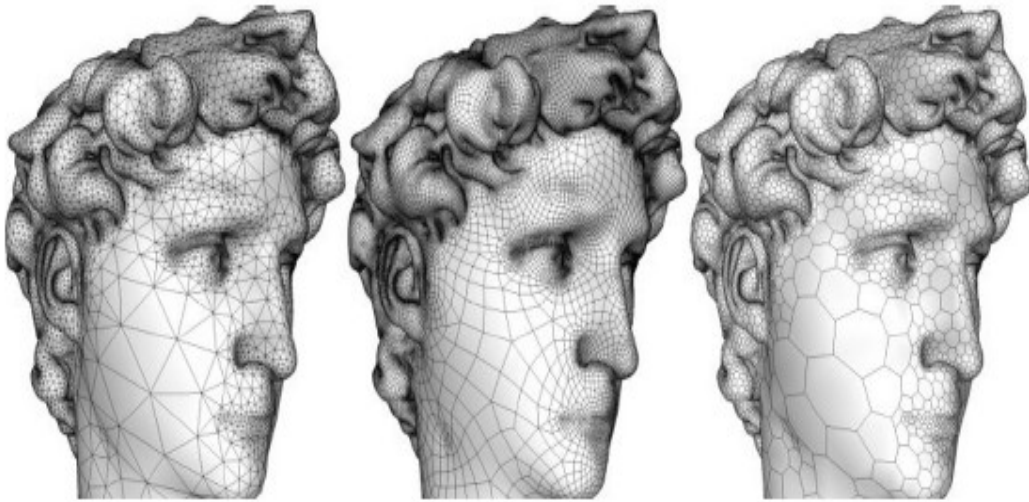
Table des figures

| | |
|--|----|
| Figures I.1 : La géométrie surfacique du David..... | 8 |
| Figure I.2: Différents types de maillages..... | 9 |
| Figure I.3: Un extrait de maillage et sa représentation numérique..... | 9 |
| Figure II.1 : Simplification géométrique d'un objet | 13 |
| Figure II.2 : Décimation de sommet..... | 14 |
| Figure II.3 : Simplification par enveloppes..... | 16 |
| Figure II.4 : Regroupement de sommets..... | 17 |
| Figure II.5 : Simplification par re-tiling..... | 18 |
| Figure II.6 : Décomposition en ondelettes..... | 19 |
| Figure II.7 : Contraction de sommets..... | 19 |
| Figure II.8.1 : Le processus de fusion pour objet..... | 20 |
| Figure II.8.2 : Simplification par fusion de faces..... | 21 |
| Figure II.9: Progressive mesh..... | 22 |
| Figure II.10 : simplification par voxelisation..... | 24 |
| Figure II.11 : Simplification avec préservation de l'apparence..... | 25 |
| Figure III.1: L'IDE C++ Builder..... | 28 |
| Figure III.2 : Architecture générale de l'application..... | 29 |
| Figure III.3 : Diagramme de cas d'utilisation..... | 30 |
| Figure III.4 : les deux cas d'insertion des points..... | 31 |
| Figure III.5: Test de cercle vide..... | 31 |
| Figure III.6 : Démonstration de la contraction d'arête..... | 33 |

| | |
|---|----|
| Figure III.7 : Suppression d'un triangle..... | 34 |
| Figure III.8 : exemple d'un fichier .OFF..... | 36 |
| Figure III.9 : Interface principale de l'application..... | 37 |
| Figure III.10 : l'interface de la version 1..... | 38 |
| Figure III.11 : Chargement d'une triangulation..... | 40 |
| Figure III.12 : Insertion et vérification..... | 40 |
| Figure III.13 : Simplification par contraction d'arête..... | 41 |
| Figure III.14 : Triangulation simplifiée par contraction d'arête..... | 41 |
| Figure III.15 : Tracement des cercles circonscrits..... | 42 |
| Figure III.16 : Simplification par suppression de triangles..... | 42 |
| Figure III.17 : Triangulation simplifiée par méthode 2 après la confirmation..... | 43 |
| Figure III.18 : Confirmation par les cercles circonscrits..... | 43 |
| Figure III.19 : Interface de la version 2..... | 44 |
| Figure III.20 : Simplification du lapin par la méthode 1..... | 45 |
| Figure III.21 : Simplification du lapin par la méthode 2..... | 46 |
| Figure III.22 : Simplification combinée du lapin..... | 47 |

I.1. Définition de maillage :

Un maillage (en anglais mesh) est une structure de données géométriques permettant de représenter des subdivisions de surface à l'aide d'un ensemble de polygones. Les maillages sont particulièrement utilisés en infographie pour représenter des surfaces ou en modélisation pour discrétiser un espace continu. [2]



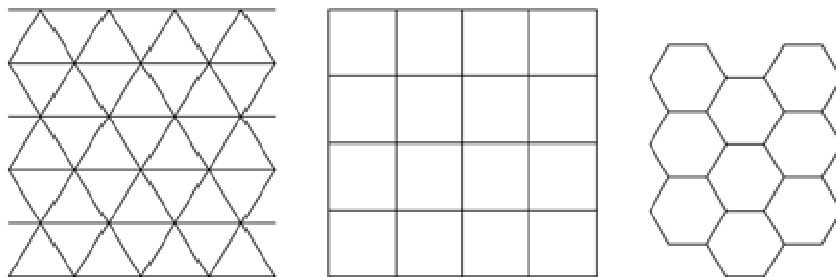
Figures I.1 : La géométrie surfacique du David maillée par des triangles, des quadrangles et des polygones. [2]

I.2. Les classes de maillages :

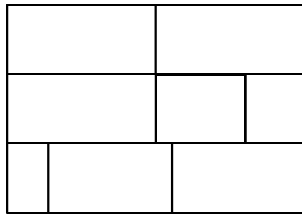
I.2.1. Maillage régulier : toutes les mailles ont le même motif et la même taille.

I.2.2. Maillage irrégulier : toutes les mailles ont le même motif mais de tailles différentes.

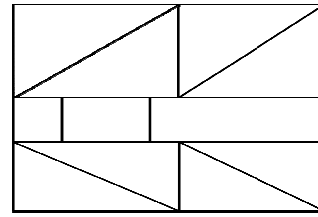
I.2.3. Maillage hybride : les mailles sont formées de motifs différents.



(a) Maillage régulier



(b) Maillage irrégulier



(c) Maillage Hybride

Figure I.2: Différents types de maillages en fonction de leur régularité.

I.3. Les maillages triangulaires :

I.3.1 Définition de maillage triangulaire :

Un maillage triangulaire est constitué d'un ensemble de triangles qui contient :

- **Les sommets:** sont représentés par des points de l'espace.
- **Les arêtes:** sont des connexions entre deux sommets, et sont situés sur la frontière d'une face.
- **Les faces:** sont des triangles dont les sommets et les arêtes appartiennent au maillage.

Voici un exemple démonstratif d'un maillage triangulaire qui contient les informations sur les deux composantes principales du maillage : sa géométrie et sa connectivité.

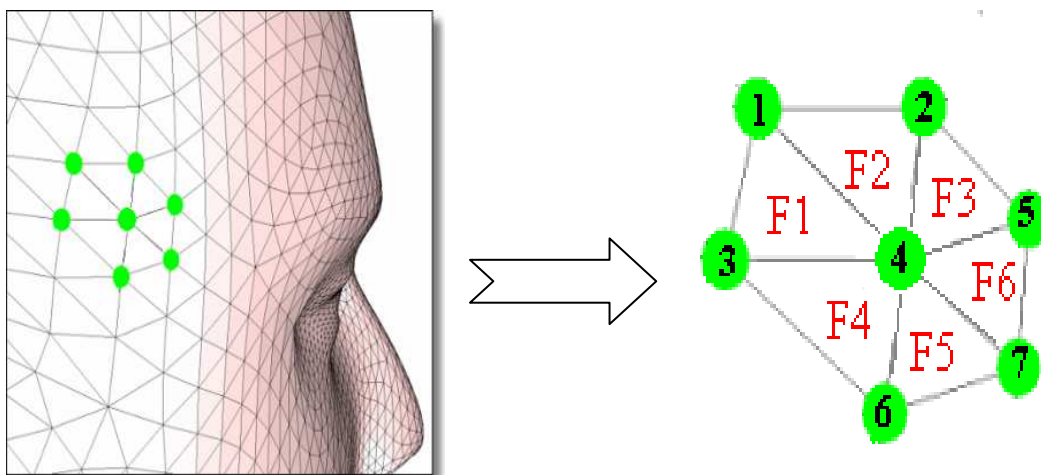


Figure I.3: Un extrait de maillage et sa représentation numérique [2]

La géométrie correspond à une liste de points, les sommets et des triangles. Cette liste décrit les coordonnées tridimensionnelles (x, y, z) des points.

La connectivité correspond à une liste de faces (i.e. : les connexions entre les sommets).

| Sommets (géométrie) | Faces (connectivité) |
|---------------------|----------------------|
| 1 (x1;y1;z1) | F1 (1; 3; 4) |
| 2 (x2;y2;z2) | F2 (1; 2; 4) |
| 3 (x3;y3;z3) | F3 (2; 4; 5) |
| 4 (x4;y4;z4) | F4 (3; 4; 6) |
| 5 (x5;y5;z5) | F5 (4; 6; 7) |
| 6 (x6;y6;z6) | F6 (4;7 ; 5) |
| 7 (x7;y7;z7) | |

Tableau I.1 : les deux composantes principales du maillage.

Dans le tableau ci-dessus, la première colonne correspond aux coordonnées des sommets (i.e. la géométrie), et la deuxième correspond aux sommets qui composent chaque face (la connectivité).

I.3.2) Définition mathématique : [2]

On dit qu'un ensemble de triangles constitue une triangulation M de la surface D qu'ils représentent si les conditions suivantes sont vérifiées :

- a. La surface D est l'union de N triangles :

$$D = \bigcup_{mi \in N} mi \text{ (} mi \text{ étant un triangle quelconque)}$$

- b. Quelque soit les triangles m_i appartenant à la Triangulation M , différent de l'ensemble vide :

$$\forall m_i \in M, m_i \neq \emptyset$$

- c. L'intersection de deux triangles (m_i, m_j) appartenant à une triangulation M donne soit : ensemble vide, un point ou un côté :

$$\forall m_i \in M, m_j \in M, m_i \cap m_j \equiv \begin{cases} \emptyset \\ \mathbf{1 \text{ point}} (m_i \neq m_j) \\ \mathbf{1 \text{ côté commun}} \end{cases}$$

II.1.Définition de la simplification :

La simplification d'un maillage réduit la quantité de données géométriques et topologiques. Le but est de réduire le nombre de sommets et de faces. La réduction d'information engendre une modification de la surface représentée par le maillage. Dans la plupart des cas, la surface simplifiée est différente de la surface originale.

La Figure II.1 montre les différentes versions d'un modèle de base composé de 2100 polygones.

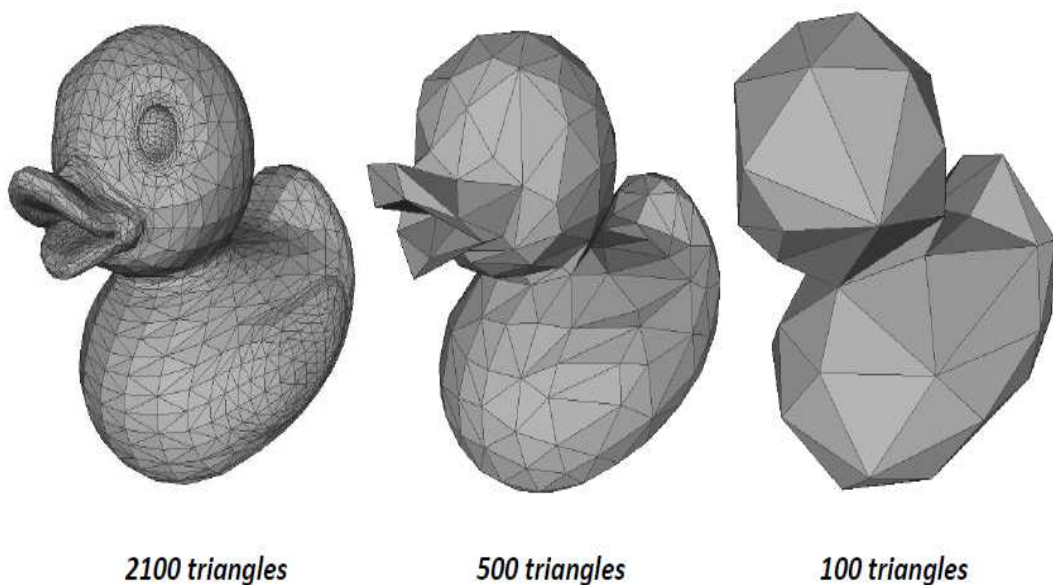


Figure II.1 : Simplification géométrique d'un modèle de 2100 triangles en 500, puis 100 triangles [3]

II.2.Contraintes de la simplification :

a- Respect de la forme :

Un maillage doit représenter convenablement la forme de l'objet ou terrain. La simplification ne doit pas altérer la forme initiale. Le maillage doit être le plus fidèle possible pour restituer la forme ou le relief.

b- La précision souhaitée :

Tout maillage engendre une précision propre dans l'interpolation. La simplification diminue cette précision. Si la précision est définie alors elle constitue

une contrainte supplémentaire dans la simplification. On ne peut pas simplifier au delà de la précision exigée.

c- La qualité des produits dérivés :

Le maillage généré après simplification doit proposer des produits dérivés de qualité acceptable.

II.3.Méthodes de simplification :

II.3.1. Décimation : [4]

La décimation consiste à supprimer un élément (sommet, arête, face, tétraèdre) du maillage basé sur un critère géométrique local et optimal pour cet élément. Après suppression, le trou résultant est comblé par triangulation (tétraédrisation en 3D), souvent coûteuse. Ces méthodes ne peuvent être utilisées que pour des surfaces manifold car la suppression d'un sommet nécessite la projection de la surface sur un plan. De par sa définition la décimation préserve la topologie.

L'approche initiale effectue plusieurs boucles sur le maillage en supprimant les sommets qui satisfont un critère d'angle ou de distance (critères géométriques et topologiques locaux).

La décimation peut aussi porter sur des suppressions d'arêtes, de faces, ou de tétraèdres.

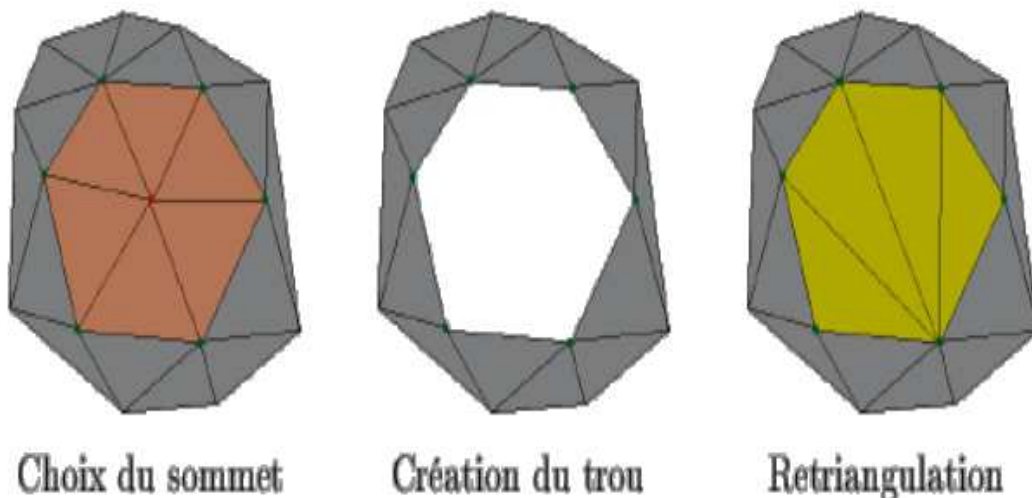


Figure II.2 : Décimation de sommet. [5]

Pseudo-code (Décimation de sommet) :

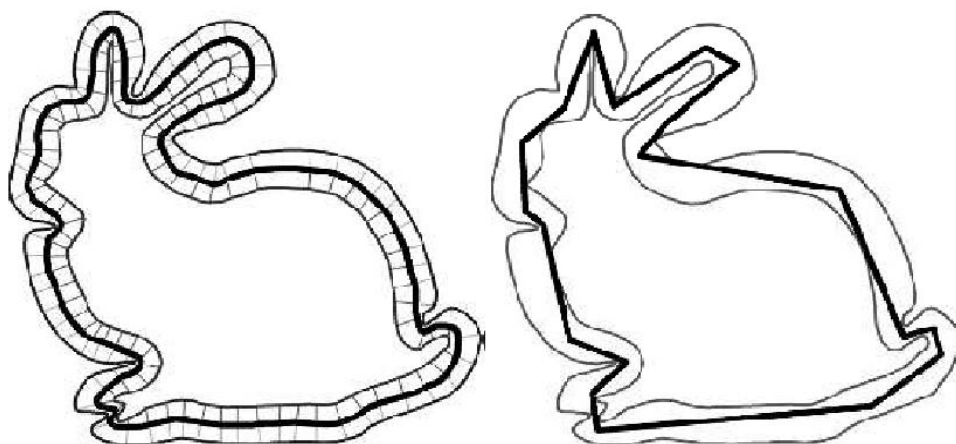
- Classer les sommets que soit amovible ou non (basé sur la topologie locale / géométrie et précision nécessaire)
- Faire
 - Choisir un sommet amovible v
 - Supprimer v et ses faces incidentes
 - Re-trianguler le trou

Jusqu'à ce que

Il n'existe plus de sommet amovible, ou taux de simplification atteint.

II.3.2. Simplification par enveloppes : [6]

Cette méthode permet de délimiter le déplacement géométrique maximum entre la surface originale et simplifiée. Le principe consiste à créer une enveloppe intérieure et extérieure afin d'enserrer la surface d'origine.



Exemple 2D de simplification par enveloppe

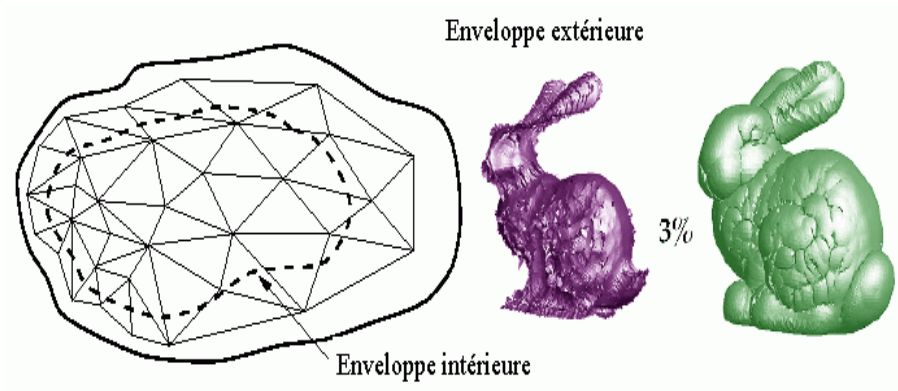


Figure II.3 : Simplification par enveloppes. [7]

La surface originale (en gras) est déplacée selon le seuil d'erreur le long de son vecteur normal afin de calculer l'enveloppe intérieure et extérieure de la surface. Cet algorithme n'est efficace que sur un maillage manifold orienté, et permet de conserver la topologie de l'objet.

Pseudo-code :

- Soit S le maillage d'entrée
- Construire deux enveloppes $-S$ et $+S$ à distance $-e$ et $+e$ à partir de S .
- Simplifier S suivant l'approche de décimation de sommet (mais les choix des sommets sont faits de façon à ce que le maillage résultant n'intersecte ni $-S$ ni $+S$).

II.3.3. Regroupement de sommets (Vertex Clustering) : [4]

Basée sur la proximité géométrique, cette approche découpe le maillage en cluster et unifie tous les sommets d'un même cluster en un unique sommet représentatif.

La figure II.4 montre les "clusters" en pointillé. Après fusion il ne reste bien qu'un seul sommet dans chaque cluster. Cette méthode est rapide et peut être appliquée à des surfaces

non manifold. Cependant elle ne préserve pas la topologie et produit des simplifications très grossières pénalisant considérablement les petits détails. Des optimisations ont été apportées récemment afin d'améliorer la qualité visuelle des maillages simplifiés.

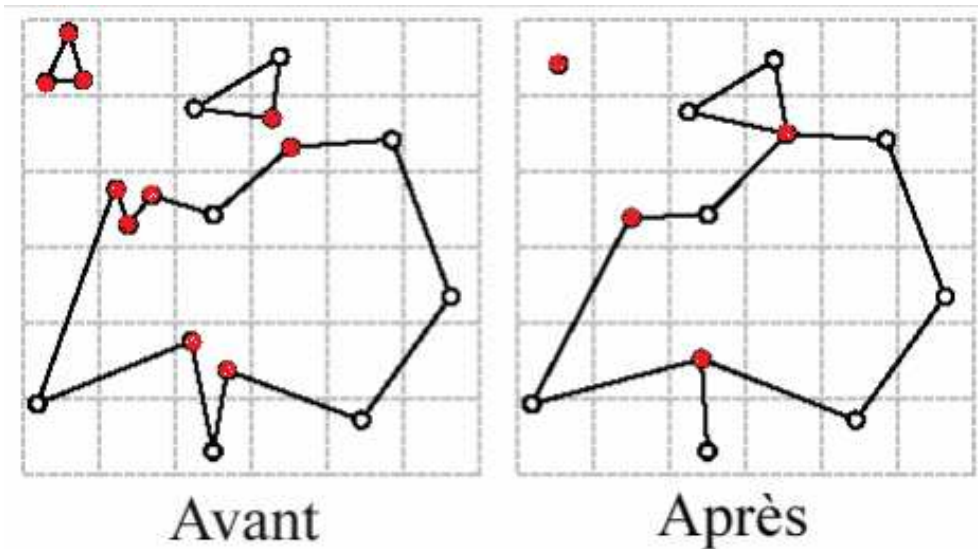


Figure II.4 : Regroupement de sommets. [8]

Pseudo-code :

- Superposer une grille sur le maillage.
- Détecter les groupes de sommets qui se trouvent dans chaque cellule.
- Unifier chaque groupe de sommets en un seul sommet.

II.3.4. Re-échantillonnage (Re-tiling) :

De nouveaux sommets sont insérés de façon aléatoire sur la surface originale du maillage et déplacés le long de la surface vers les régions à plus forte courbure. Les sommets initiaux sont alors supprimés itérativement et un maillage re-triangulé, construit avec ces nouveaux sommets, est généré. Le résultat de [9] présenté figure II.5 montre bien ce choix stratégique de point avant la retriangulation (illustration du milieu).

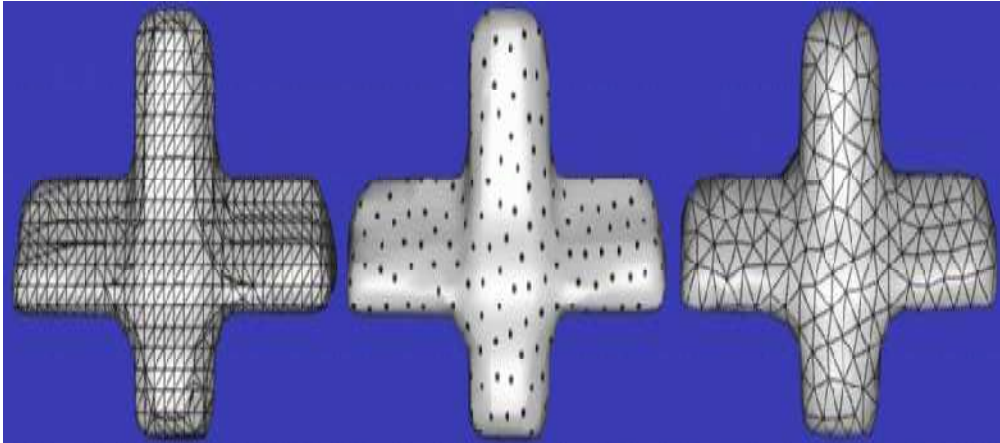


Figure II.5 : Simplification par re-tiling. [9]

Pseudo-code :

- Distribuer une nouvelle série de sommets sur le maillage original
(points positionnés sur les fortes courbures)
- Supprimer les sommets d'origine.
- Re-trianguler.

II.3.5. Décomposition en ondelettes : [10]

Cette approche qui semble intéressante pour la simplification de maillage en fournissant en plus la multirésolution, souffre de la nécessité d'une hiérarchie régulière pour supporter cette décomposition.

Cette méthode a été conçue initialement pour des maillages réguliers mais elle a été étendue pour des maillages plus généraux. Elle préserve la topologie mais est altérée par les erreurs introduites lors de la construction du maillage de base qui se retrouvent lors de la reconstruction.

Le but de cette méthode permet d'extraire les données pertinentes.

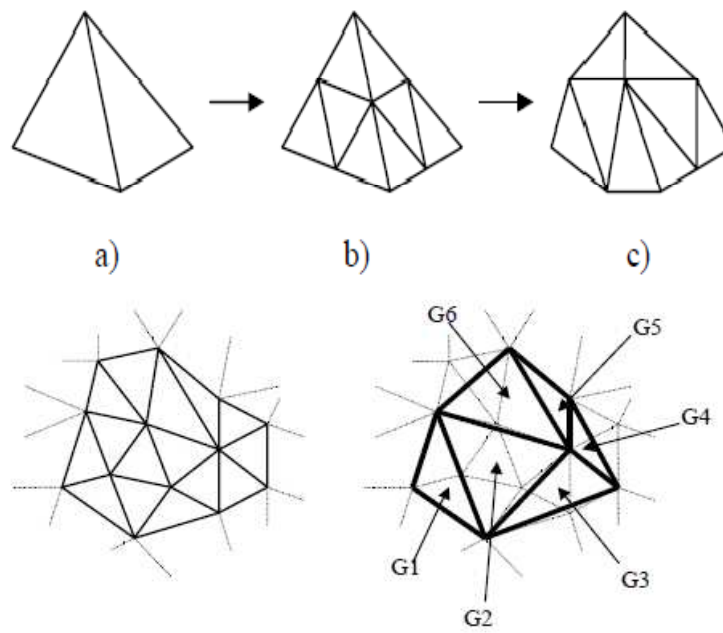


Figure II.6 : Décomposition en ondelettes. [10]

II.3.6. Contraction itérative :

La plus grande classe d'algorithmes est basée sur la contraction itérative d'éléments (paire de sommets, arête, triangle, tétraèdre). La liste de toutes ces méthodes est longue mais le schéma général est sensiblement le même. A noter que la contraction de triangles peut être ramenée à la contraction successive de plusieurs arêtes.

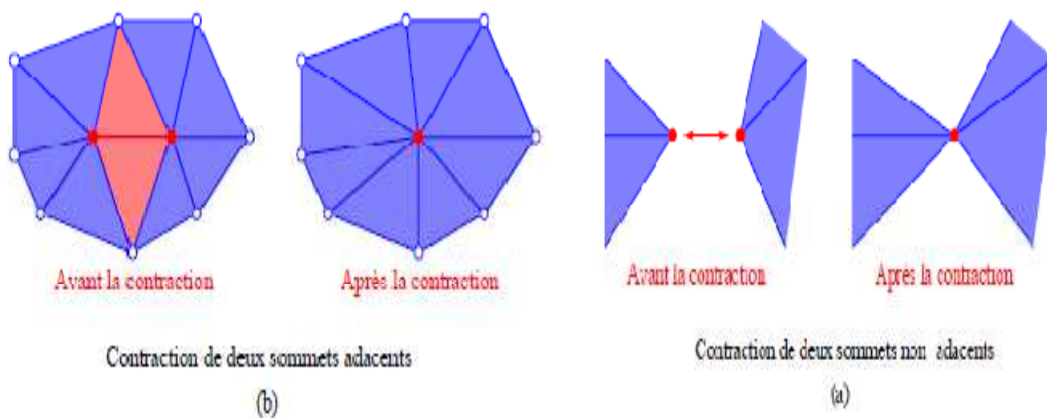


Figure II.7 : Contraction de sommets. [4]

Pseudo-code : La contraction d'une paire de sommets

$(v_i;v_j)$

- Soit deux sommets v_i et v_j du maillage M
- Déplacer les sommets v_i et v_j vers la position v
- Remplacer toutes les occurrences de v_i par v_j
- Supprimer v_i et toutes les faces dégénérées.

II.3.7. Fusion de régions (Region Merging) :

Ces méthodes de simplification de surfaces fusionnent des groupements de triangles et de sommets inscrits dans une certaine région. Certaines méthodes détectent les triangles qui sont quasi-coplanaires en comparant leur normale, les fusionnent et re-triangulent le trou résultant [11].

Ces méthodes préservent la topologie des surfaces mais ne permettent pas de borner l'erreur d'approximation.

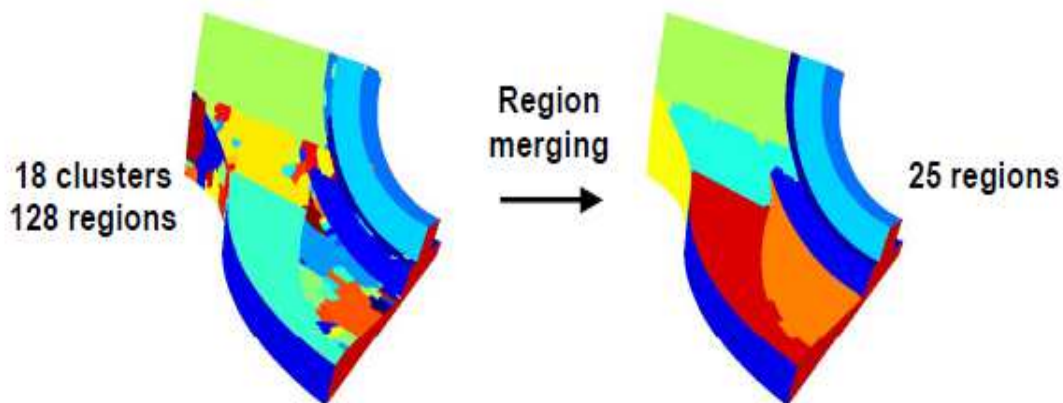


Figure II.8.1 : Le processus de fusion pour l'objet "Fandisk". [12]

La figure (II.8.1) présente le résultat de la fusion pour l'objet "Fandisk". Partant de 128 régions, le processus de fusion avec un seuil fixé à 5, donne 25 régions finales caractérisées par leurs valeurs de courbure.

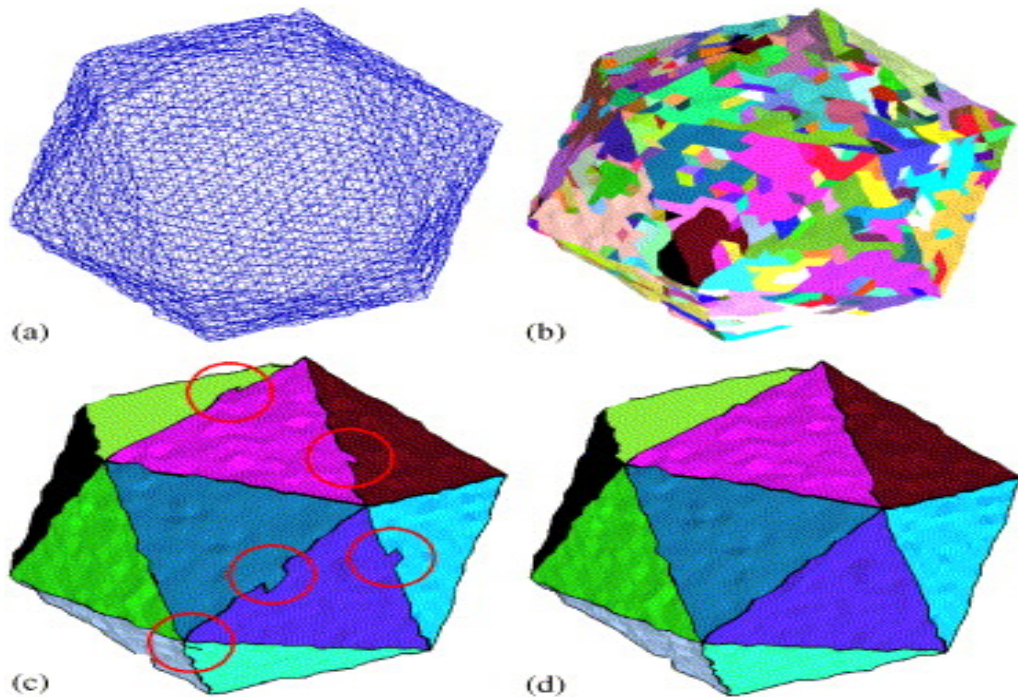


Figure II.8.2 : Simplification par fusion de faces. [13]

(a) Maillage d'entrée. (b) Fusion itérative de triangles (650 régions). (c) un processus itératif de fusion de région (20 régions). (d) correction des frontières. (20 régions).

Pseudo-code :

- Maillage d'entrée M
- Fusion itérative de triangles (Partitionner le maillage en régions en fusionnant les triangles basés sur un critère de planéité)
- Fusion itérative de régions
- Re-trianguler la région résultante.
- Correction des frontières.

II.3.8. Effondrement (Progressive mesh) :

Cette méthode est mieux décrite par *Hugues Hoppe* [14], consiste à itérer une transformation, dite effondrement d'arête, où l'on fusionne les deux extrémités d'une arête en un point arbitrairement placé, entraînant la disparition d'un point et de jusqu'à deux arêtes et faces, si les faces voisines sont des triangles. La difficulté est de mettre à jour les informations d'adjacence, fortement perturbées par l'opération.

Le choix de l'arête à effondrer résulte de la minimisation sur l'ensemble des arêtes d'une fonction d'énergie que l'on se donne, et qui va servir à caractériser la fidélité du maillage simplifié au maillage initial. On peut également minimiser cette fonction sur l'espace des positions acceptables du point destination, ce qui conduit à de meilleures simplifications.

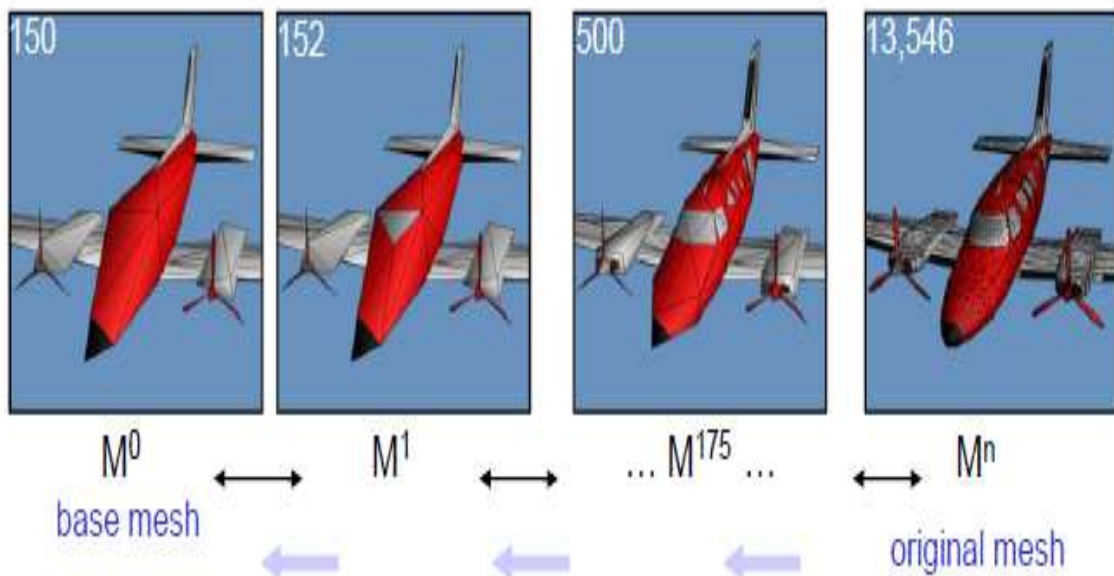


Figure II.9: Progressive mesh. [14]

L'utilisation de cette méthode permet également, l'effondrement d'arête étant réversible si l'on dispose des informations nécessaires, un codage progressif des maillages. En effet, en partant d'un maillage simplifié et en appliquant la « fission » réciproque du dernier effondrement opéré, affiner progressivement le maillage. Ce point est particulièrement intéressant pour la transmission par réseau, permettant au récepteur de visualiser l'objet avant qu'il ne soit entièrement reçu.

Pseudo-code :

- On simule l'effondrement de chaque arête pour déterminer celle qui entraîne la plus faible variation en cas d'effondrement.
- On effondre ainsi l'arête trouvée. (*)
- On re-simule l'effondrement uniquement pour les arêtes proches de celle qui a été effondrée.
- On trouve ainsi une nouvelle arête à effondrer et on reprend en (*)

II.3.9. Filtres volumiques (voxelisation) :

Ces méthodes globales, transforment un maillage polygonal en un champ scalaire défini sur une grille volumique et simplifient ce champ pour créer une approximation du maillage initial.

Pour cela, une grille régulière est superposée au maillage et une valeur scalaire est affectée par cellule. A l'aide de filtres de convolution, ce champ est simplifié et utilisé pour extraire un nouveau maillage triangulaire à l'aide d'un algorithme standard du *marching cube*. Plusieurs techniques existent pour la création du champ scalaire.

He, L. Hong, A. Varshney, et S. Wang [15] assigne une valeur booléenne à une cellule de la grille régulière (*voxel*) étant zéro si le centre de la cellule est à l'intérieur de l'objet sinon, un.

Nooruddin et G. Turk [16] associe à chaque *voxel* sa plus petite distance à l'objet et affine ce premier choix par des méthodes dites de dilatation et d'érosion.

Ces méthodes sont utilisées pour modifier volontairement la topologie du maillage initial afin d'effectuer des simplifications à basse résolution.

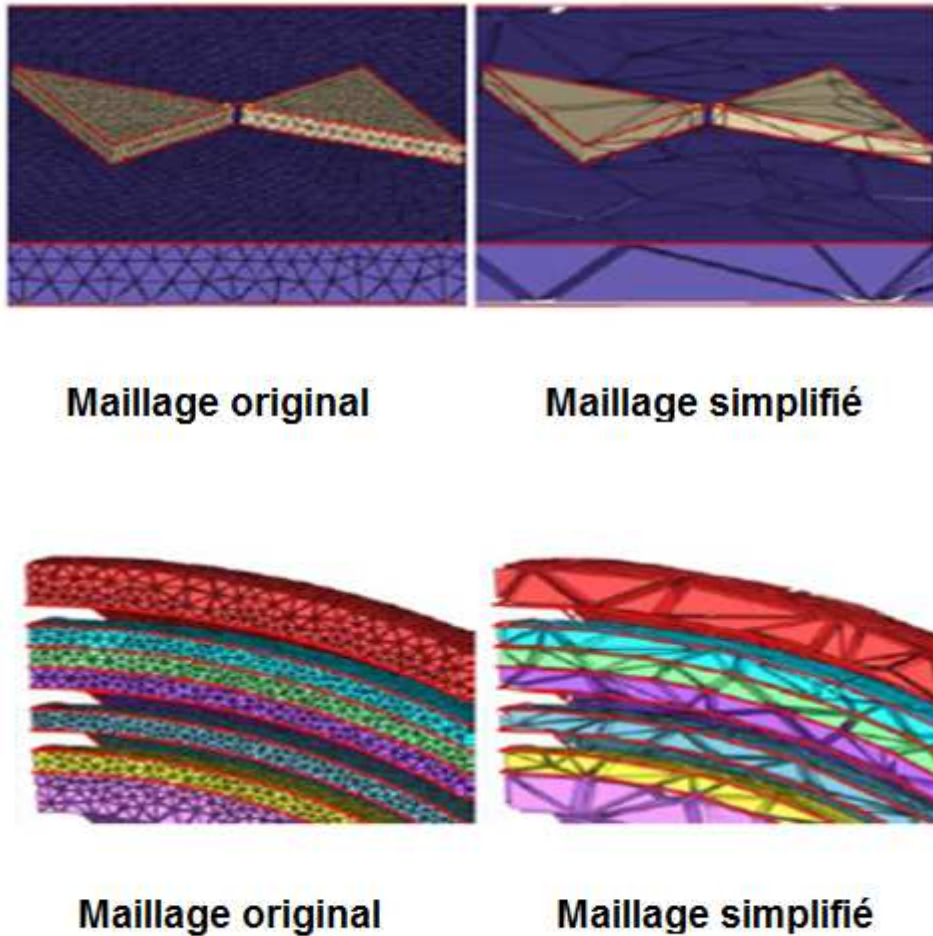


Figure II.10 : simplification par voxelisation. [11]

II.3.10. Simplification par préservation de l'apparence :

Cette technique de [17] peut être vue comme une extension de la méthode de simplification par enveloppe. Au lieu de simplement contraindre la déviation de la surface à un intervalle donné, la déviation (erreur) maximale est basée sur plusieurs critères visuels de la surface simplifiée : la variation de la position de sa surface, la variation de sa couleur et la variation de sa courbure. La variation de couleur correspond aux distorsions induites entre une texture plaquée sur le maillage originel et la même texture plaquée sur le maillage simplifié. La variation de courbure correspond à la perte de détails géométriques. Afin de minimiser la variation de ces deux derniers paramètres lors de la simplification, la couleur et la courbure du maillage originel sont enregistrées dans des textures utilisées par la suite pour atténuer l'erreur

visuelle de la simplification. La courbure est représentée par l'ensemble des normales à la surface.

Finalement, le processus de simplification consiste à créer un maillage simplifié par réduction d'arêtes et à plaquer la texture de couleur et de normales sur celui-ci afin de minimiser les erreurs visuelles commises. Le choix des arêtes à réduire est fait de façon à rester dans les limites autorisées.

Cette technique donne de très bons résultats visuels (figure. II.11) car le placage de la carte de normales redonne un aspect de relief saisissant.

Néanmoins, cette méthode ne permet pas une très grande simplification du modèle, car l'estimation d'erreur basée sur des critères visuels est très contraignante.

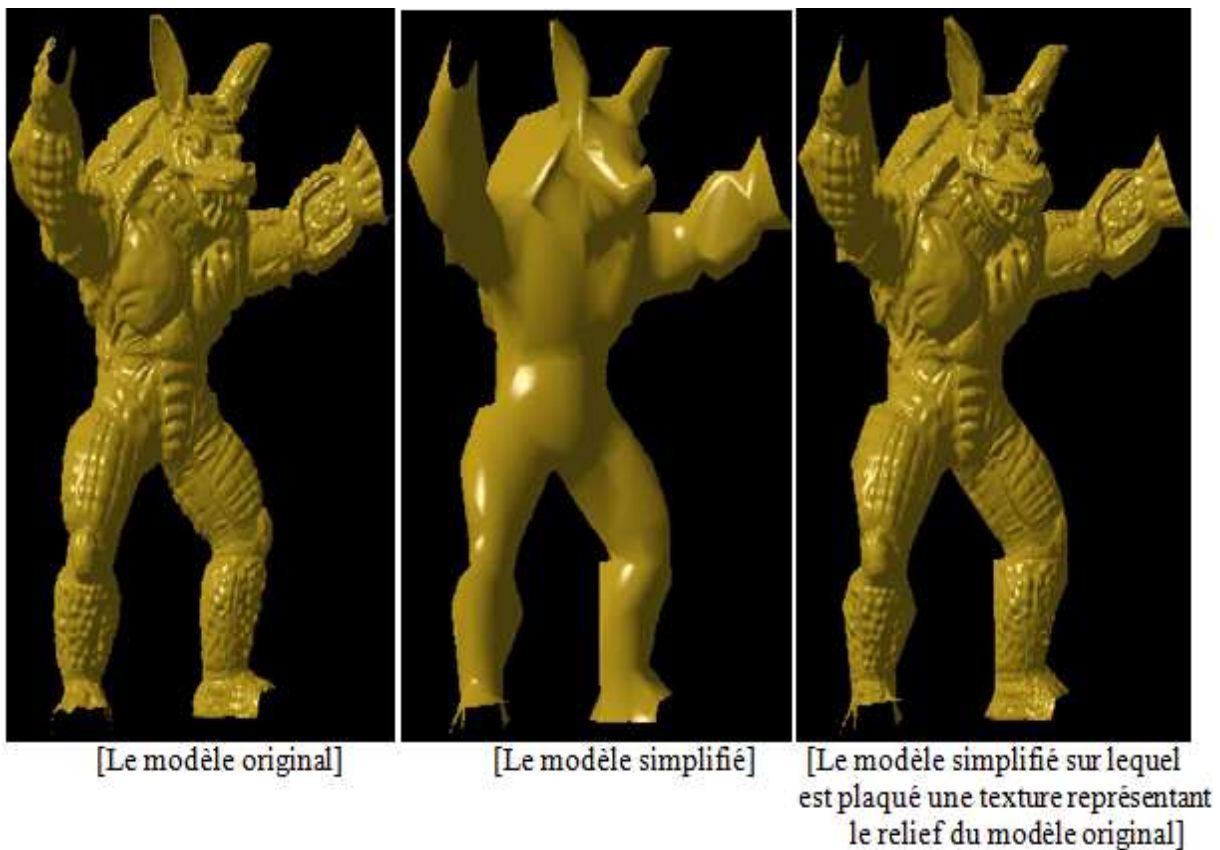


Figure II.11 : Simplification avec préservation de l'apparence. [17]

III.1.Introduction :

Dans ce chapitre, nous présentons l'implémentation de notre application qui sert à simplifier un maillage dans le cas général, et qui peut être appliquée en particulier au Modèle Numérique du Terrain (MNT), donné par l'utilisateur sur un fichier (au format.OFF) en premier lieu, l'application va s'occuper de l'affichage du maillage (MNT), puis l'utilisateur choisira l'une des méthodes de simplification que nous avons implémenté.

A la fin, le modèle simplifié peut être enregistré dans un fichier sous format (.OFF).

Nous avons opté pour implémenter deux méthodes de simplification, la première basée sur la contraction d'arête et la deuxième basée sur la suppression de triangles selon des critères qui servent à garder la forme générale de l'objet à simplifier et choisir une condition.

III.2.Représentation de l'application :

III.2.1. Intérêt du projet :

L'intérêt de notre projet est de pouvoir simplifier un maillage triangulaire ou un MNT (Modèle Numérique de Terrain) par plusieurs méthodes afin de rendre ce dernier plus facile à manipuler dans les différentes applications grâce à sa taille démunie.

III.2.2. Ressources utilisées :

On va présenter les ressources matérielles et logicielles utilisées.

- **Ressources matérielles :** en général, il faut que l'ordinateur soit assez puissant (processeur de plus de 2.00 GHz, mémoire vive de plus de 2.00 GO) puisque c'est une application assez lourde, avec beaucoup d'itérations et de composantes graphiques.
- **Ressources logicielles :**

Pour l'implémentation de notre application, on a choisi de travailler avec le C++ comme langage de programmation sous l'IDE C++ Builder (version 6.7).

C++ Builder est un IDE, un environnement de développement intégré. Il regroupe tout un ensemble d'outils permettant d'effectuer un maximum de tâches de développement au sein du même environnement de travail.

C++ Builder est de plus un environnement de développement visuel C++ RAD (Rapid Application Development). Il permet de construire rapidement des applications en utilisant des composants et de simplifier au maximum l'écriture du code et la réalisation de l'interface.

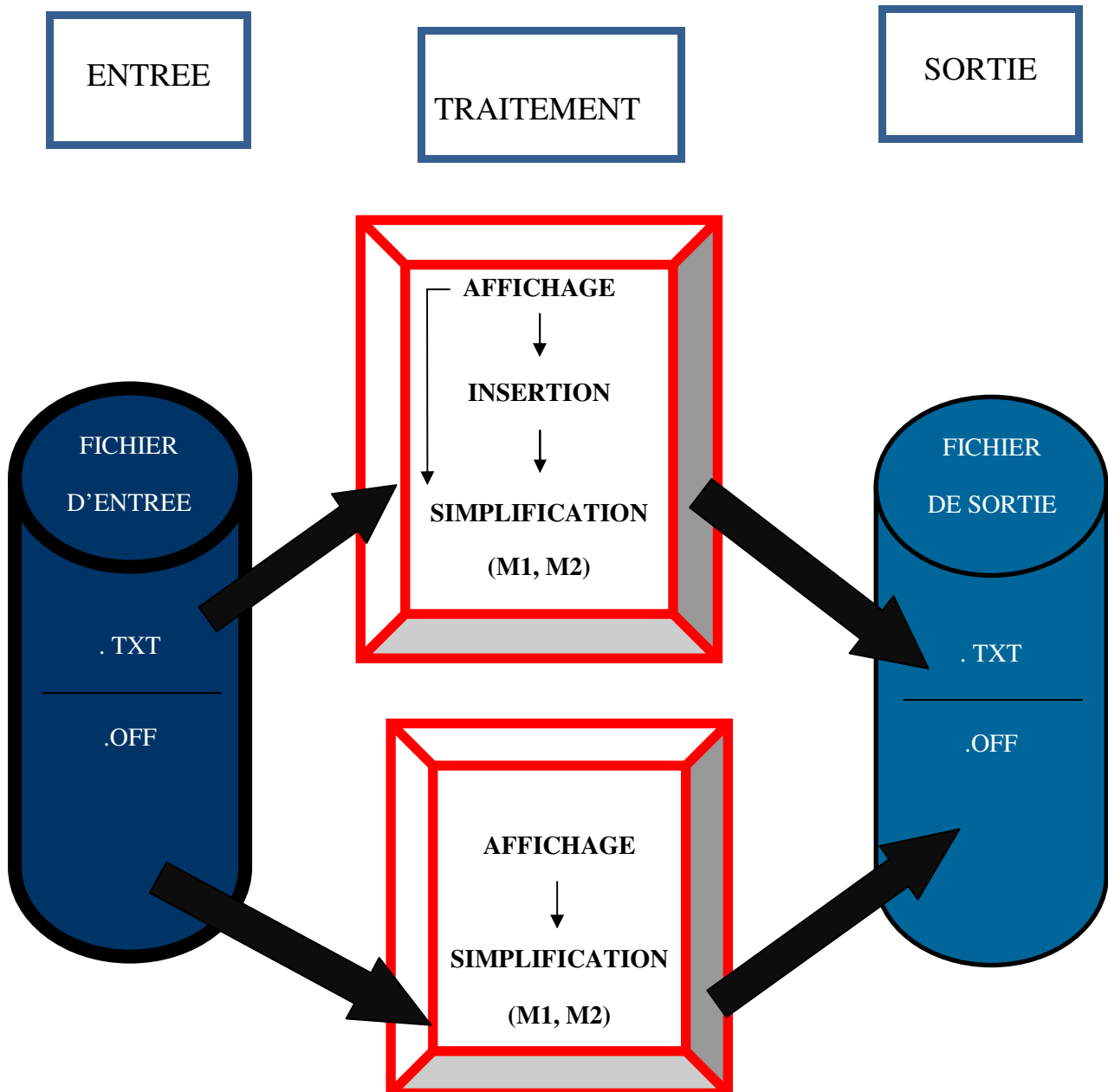
On peut ainsi très rapidement se consacrer à la partie « métier » du code (le code réellement utile de l'application).



Figure III.1: L'IDE C++ Builder

III.3. Conception:

III.3.1. Architecture générale du programme :



M1 : Méthode 1(Contraction d'arête)

M2 : Méthode 2(Suppression du triangle)

Figure III.2 : Architecture générale de l'application.

III.3.2. Diagramme de cas d'utilisation :

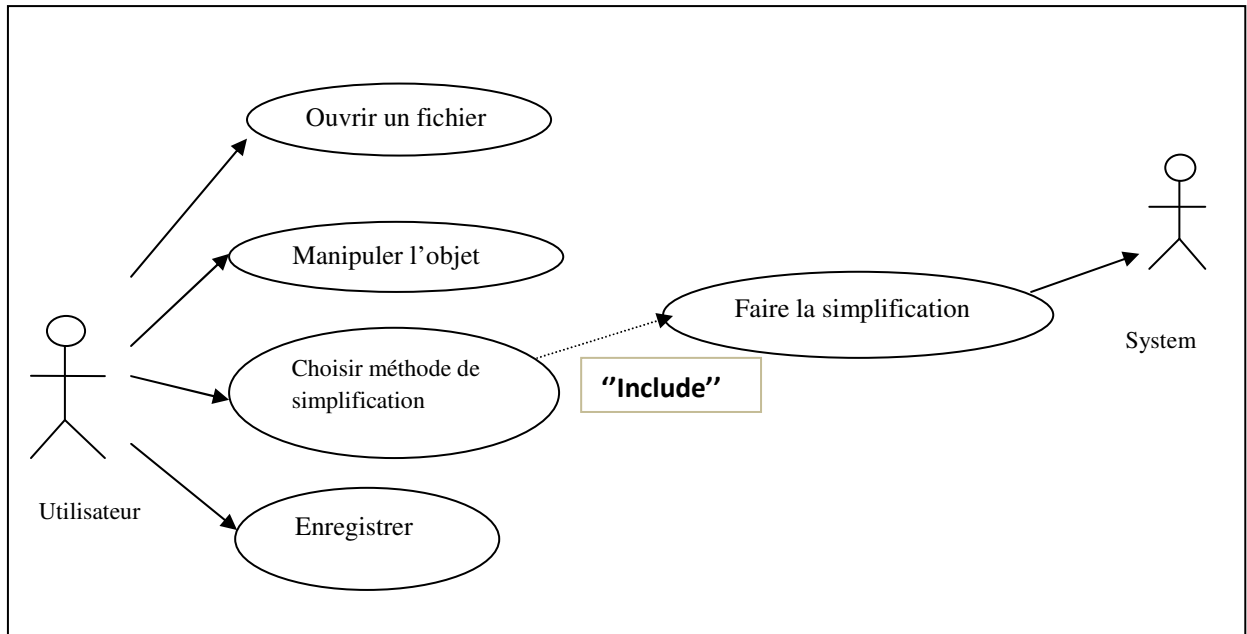


Figure III.3 : Diagramme de cas d'utilisation

III.3.3. Description des algorithmes :

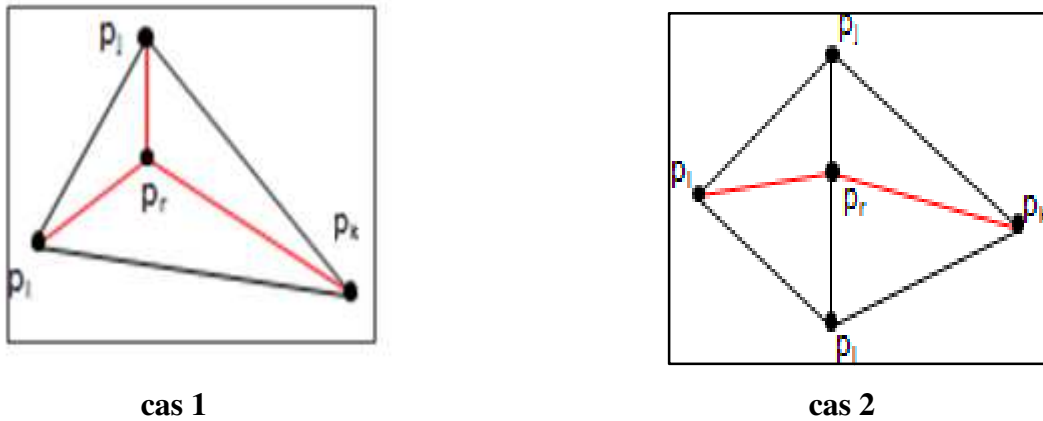
III.3.3.1. Triangulation de Delaunay (TD) :

Nous avons choisit d'implémenté la méthode incrémentale qui est la meilleure façon de construire une TD grâce à son caractère dynamique.

Le principe de cet algorithme est d'ajouter un à un les points et de modifier par conséquence la triangulation.

On a deux cas d'insertion dans la triangulation :

- *Premier cas* : inserer le point à l'interieur d'un triangle.
- *Deuxième cas* : inserer le point sur l'arête .

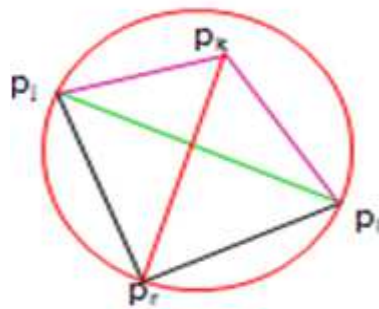


cas 1

cas 2

Figure III.4 : les deux cas d'insertion des points

Comme critère de vérification de la TD, nous avons utilisé la propriété caractéristique de la triangulation de Delaunay qui dit : « le cercle circonscrit à un triangle ne contient aucun autre point de la triangulation que ceux du triangle lui-même ». Si se n'est pas vide fait le retournement d'arête (comme la figure illustré $[p_i p_j]$ sert $[p_k p_r]$).

*Figure III.5:* Test de cercle vide.

Les modifications sont donc exécutées de façon à conserver cette propriété pour que la triangulation générée soit celle de Delaunay.

On trouve par la suite le pseudo code de l'algorithme incrémental

$T = (x_1, x_2, x_3, x_4)$;

T= trianguler T ;

1. Initialiser DT(p) par T triangulé ;
2. Pour r de 1 jusqu'au n faire

Trouver le triangle p_i, p_j, p_k de T, qui contient p_r ;

Si p_r est à l'intérieur de triangle p_i, p_j, p_k

Diviser p_i, p_j, p_k ;

Légaliser p_r, p_i, p_j ;

Légaliser p_r, p_i, p_k ;

Légaliser p_r, p_j, p_k ;

Si p_r se trouve sur une arête de p_i, p_j, p_k

Diviser $p_i, p_j, p_i, p_k, p_j, p_i$;

Légaliser p_r, p_i, p_i ;

Légaliser p_r, p_i, p_k ;

Légaliser p_r, p_j, p_i ;

Légaliser p_r, p_j, p_k ;

III.3.3.2.Méthodes de simplification utilisées :

a. Contraction d'arêtes :

Cette méthode attaque les liens entre les sommets, la suppression des arêtes est faite après vérification de certains critères (vérifier si la suppression de lien ne déformera pas la forme générale de l'objet et la longueur inférieure au seuil introduit par l'utilisateur).

Le traitement est fait comme suit : on supprime un sommet et on remplace l'arête contractée par le deuxième sommet après la modification de ses coordonnées.

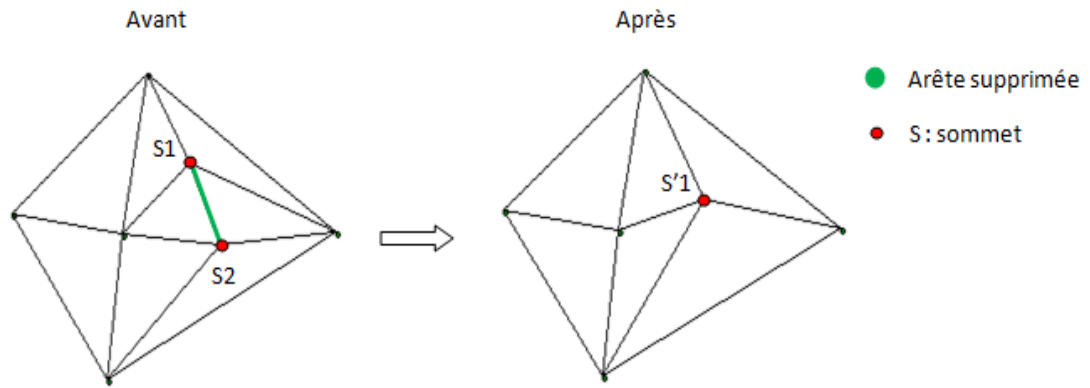


Figure III.6 : Démonstration de la contraction d'arête.

Au dessous, on a le pseudo code de la méthode 1 :

```

Void methode1 ( )
{
Pour chaque triangle faire
{
Calculer la longueur de liens
Si (le lien n'est pas lien de bordure) Et (longueur < Seuil)

Supprimer le lien

Mise à jour des structures de données

Mise à jour de l'affichage
}
}

```


b. Suppression de triangle :

Avec cette méthode, le choix des triangles à supprimer est fait après avoir vérifié si le triangle n'est pas un triangle de bordure, avec la condition (surface de triangle inférieur au seuil introduit par l'utilisateur).

On fait le même traitement que la méthode précédente, mais au lieu de supprimer un sommet, on supprime deux et on remplacera le triangle supprimé par le sommet modifié.

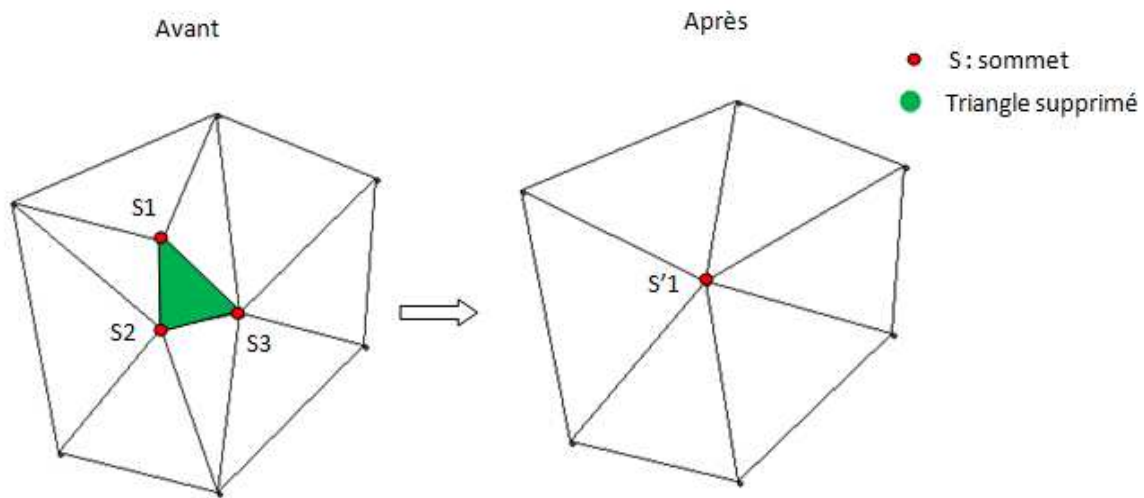


Figure III.7 : Suppression d'un triangle.

Au dessous, on a le pseudo code de la méthode 2 :

```

Void methode2 ()
{
Pour chaque triangle faire
{
Calculer la surface

```

```
    Si (le triangle n'est pas un triangle de bordure) Et (surface < Seuil)

        Supprimer le triangle

        Mise à jour des structures de données

        Mise à jour de l'affichage

    }

}
```

III.3.4. Structure de données utilisées :

Nous avons utilisé pour la structure de données deux matrices pour charger les données à partir d'un fichier .OFF ; une première sert à stocker les coordonnées X ,Y, Z de chaque sommet et une deuxième matrice qui va stocker les indices des sommets formant un triangle ainsi le nombre de voisins de ce triangle suivi des indices de ses voisins.

Points :

| | x | y | z |
|----|---|---|---|
| P1 | | | |
| P2 | | | |
| P3 | | | |

Triangle :

| | P1 | P2 | P3 | NBR-VOISIN | VOISIN 1 | VOISIN 2 | VOISIN 3 |
|----|----|----|----|------------|----------|----------|----------|
| T1 | | | | | | | |
| T2 | | | | | | | |

Et voici un exemple du fichier d'entrée .OFF.

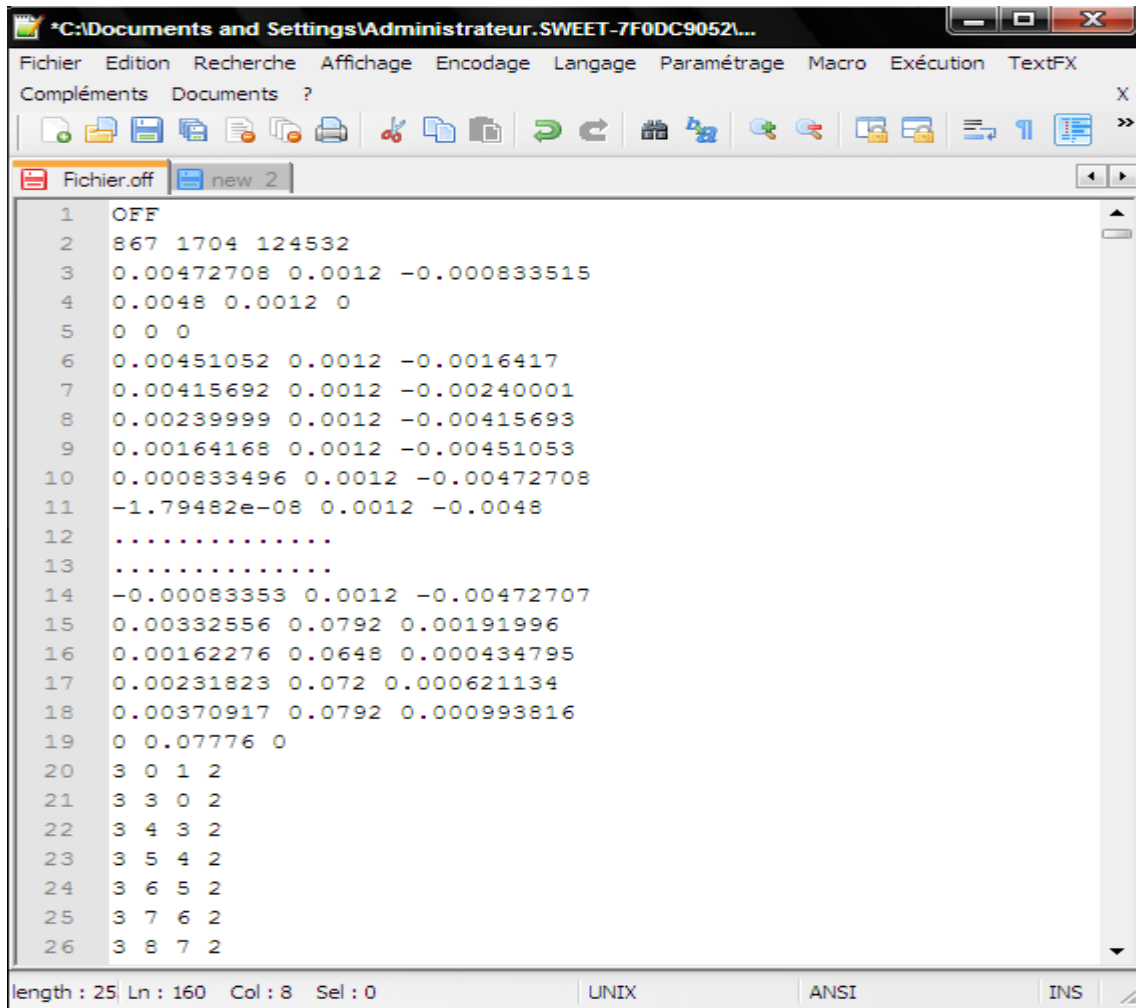


Figure III.8 : exemple d'un fichier .OFF

III.4.Implémentation :

III.4.1 : Interface principale :



Figure III.9 : Interface principale de l'application

Pour mieux illustrer les principes des méthodes de simplification utilisées on a opté pour deux versions pour notre application :

- ✓ Version 1
- ✓ Version 2

❖ **Version 1 :**

En cliquant sur la Version1 Version 1 , on aura la fenêtre suivante :

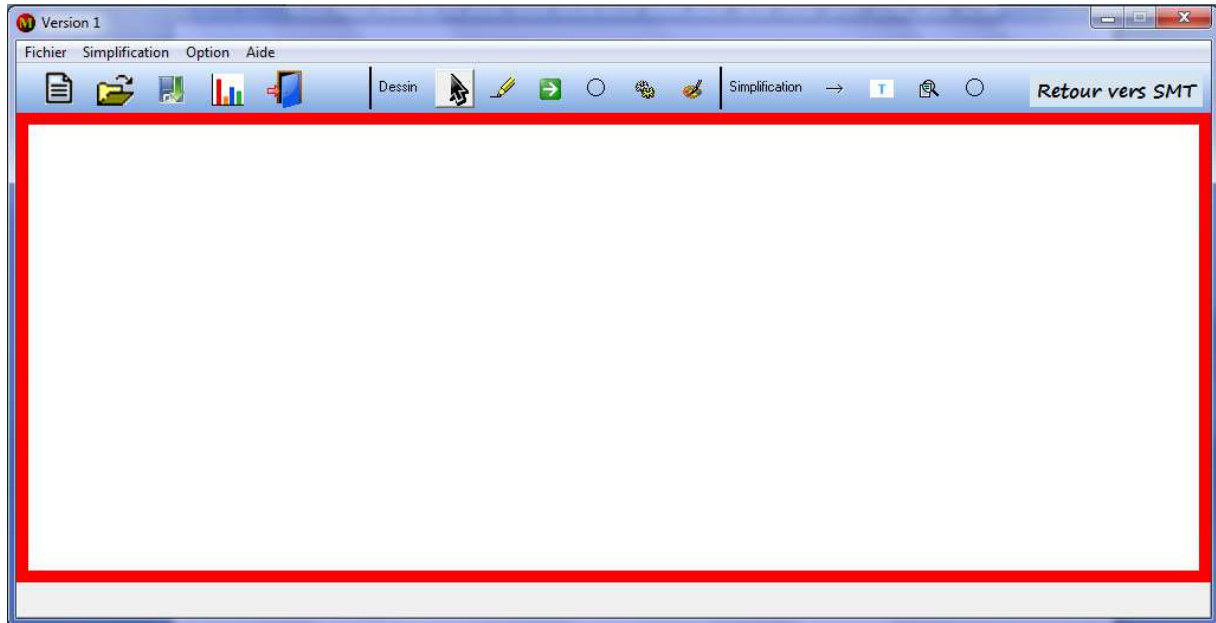


Figure III.10 : l'interface de la version 1.

Elle comporte :

➤ **Menu :**

Fichier :

- Nouveau (vider l'espace réservé au dessin ou affichage).
- Ouvrir (charger un fichier texte).
- Enregistrer (permet l'enregistrement sous format .txt).
- Fermer (quitter la version 1).

Simplification :

- Arête (contraction d'arête).
- Triangle (suppression de triangle).











Option :

- Couleur (permet la modification de l'arrière plan).

Aide :

- A propos (c'est une boîte de dialogue affiche les informations qui concerne l'éditeur du programme).

➤ **Définition des boutons et leurs fonctionnements :**

| Boutons | Fonctions |
|---|-------------------------------------|
|  | Nouveau fichier |
|  | Ouvrir fichier |
|  | Enregistrer fichier |
|  | Voir les statistiques |
|  | Exit |
|  | Empêcher l'insertion |
|  | Insérer un point |
|  | Actualiser |
|  | Tracer les cercles circonscrits |
|  | Confirmer Delaunay |
|  | Changer couleur |
|  | Méthode 1 : Contraction d'arête |
|  | Méthode 2 : Suppression de triangle |
|  | Afficher les deux méthodes |
| Retour vers SMT | Revenir à l'interface principale |

On Charge une triangulation et on confirme qu'elle est de Delaunay en traçant les cercles circonscrits.

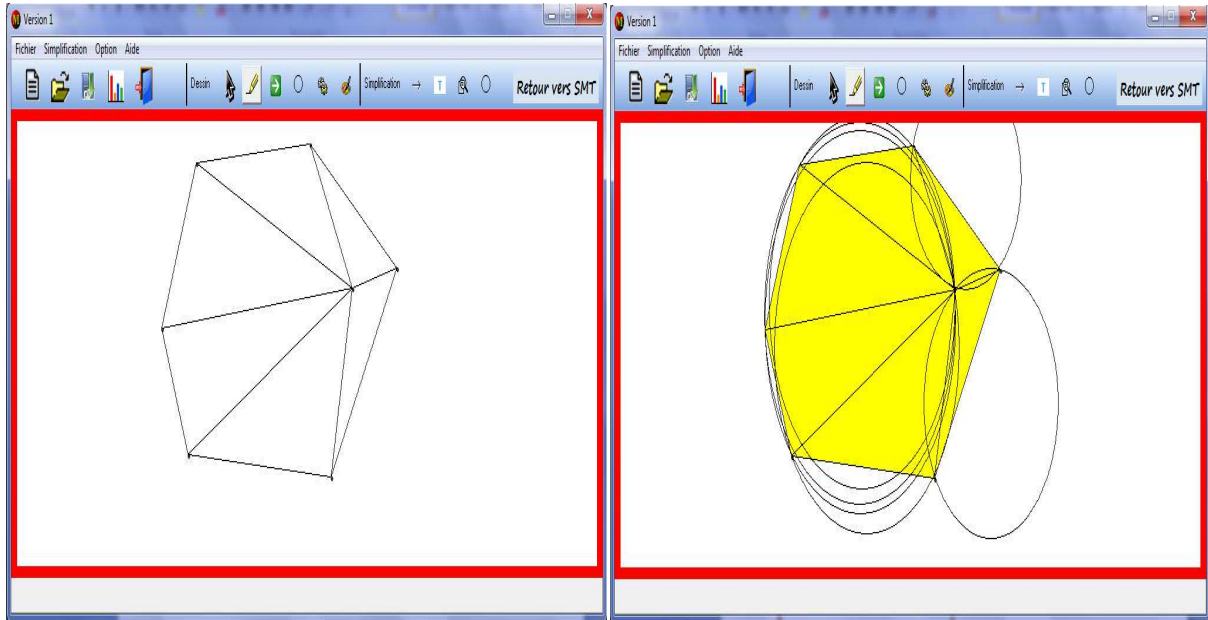


Figure III.11 : Chargement d'une triangulation.

Après l'insertion des points, on aura une autre triangulation mais elle respecte la contrainte de Delaunay comme la figure suivante :

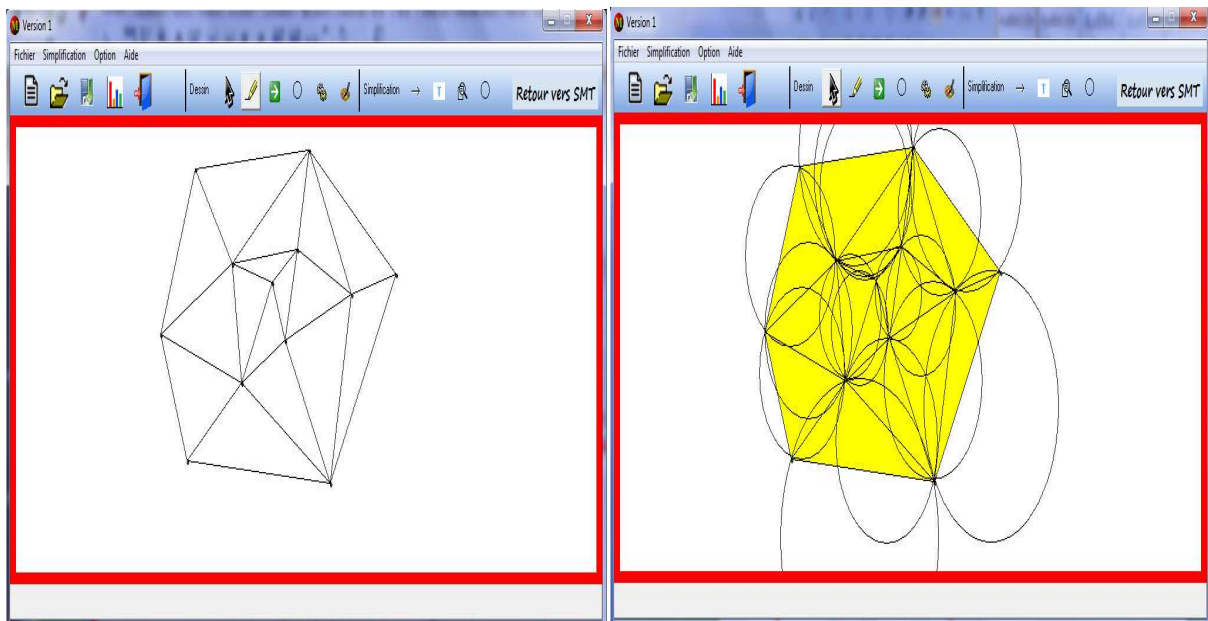


Figure III.12 : Insertion et vérification.

Passant à la simplification, premièrement, par contraction d'arête en cliquant sur (1) de la figure (III.13), une boîte de dialogue s'apparaît pour introduire le seuil.

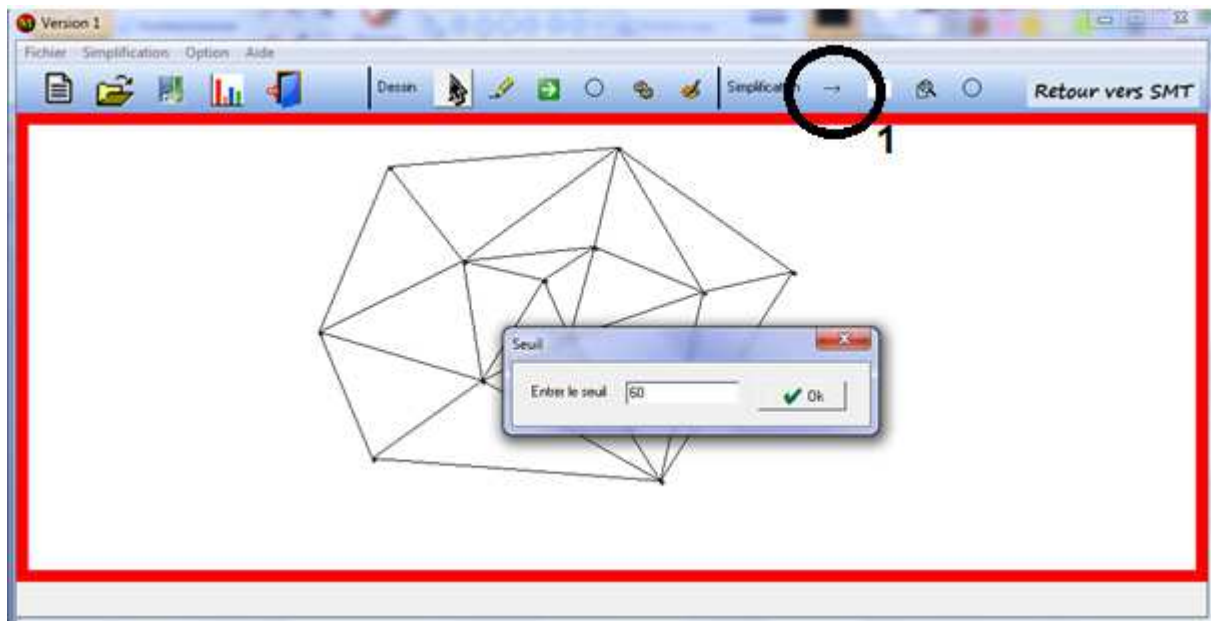


Figure III.13 : Simplification par contraction d'arête.

Après la confirmation, on aura le résultat suivant :

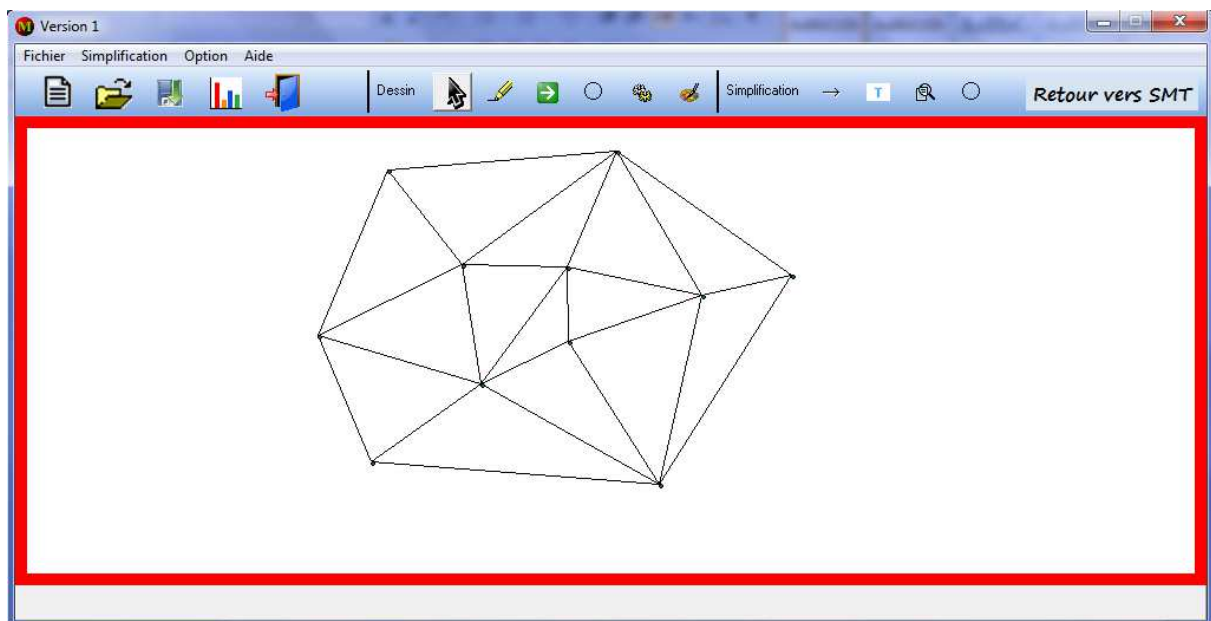


Figure III.14 : Triangulation simplifiée par contraction d'arête.

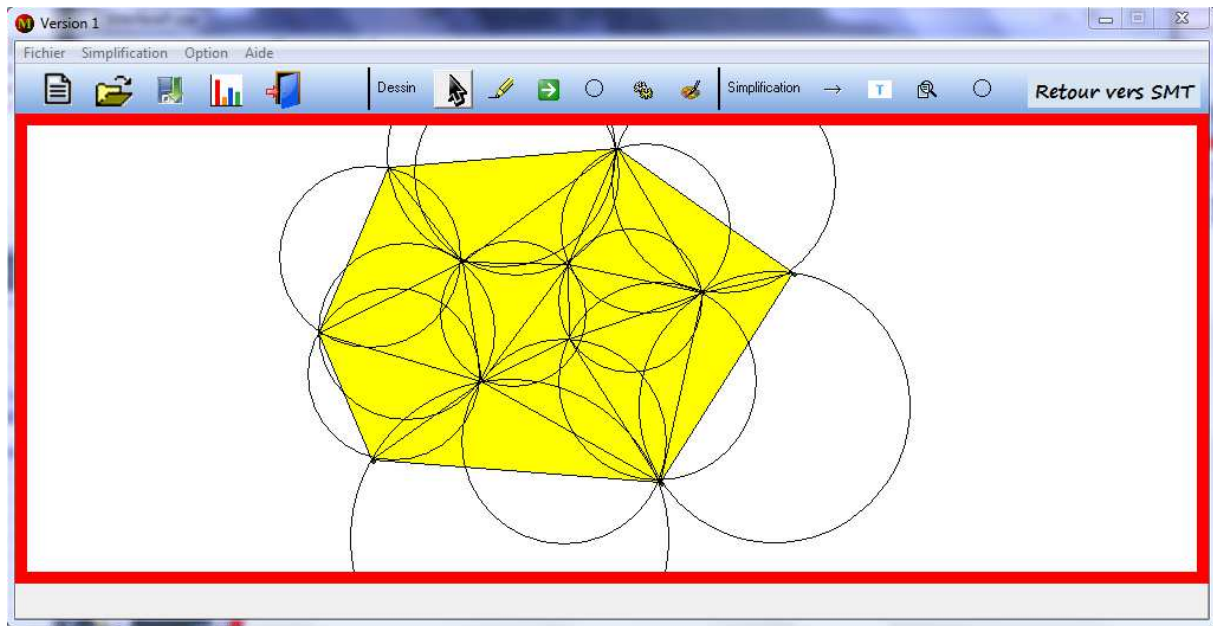


Figure III.15 : Tracement des cercles circonscrits.

Remarque : d'après les figures (III.14) et (III.15), on remarque que la simplification par contraction d'arête garde la fidélité (contrainte de Delaunay).

On passe à la deuxième méthode (suppression de triangles) en cliquant sur (2) de la figure (III.16), une boîte de dialogue qui permet d'entrer le seuil s'affiche.

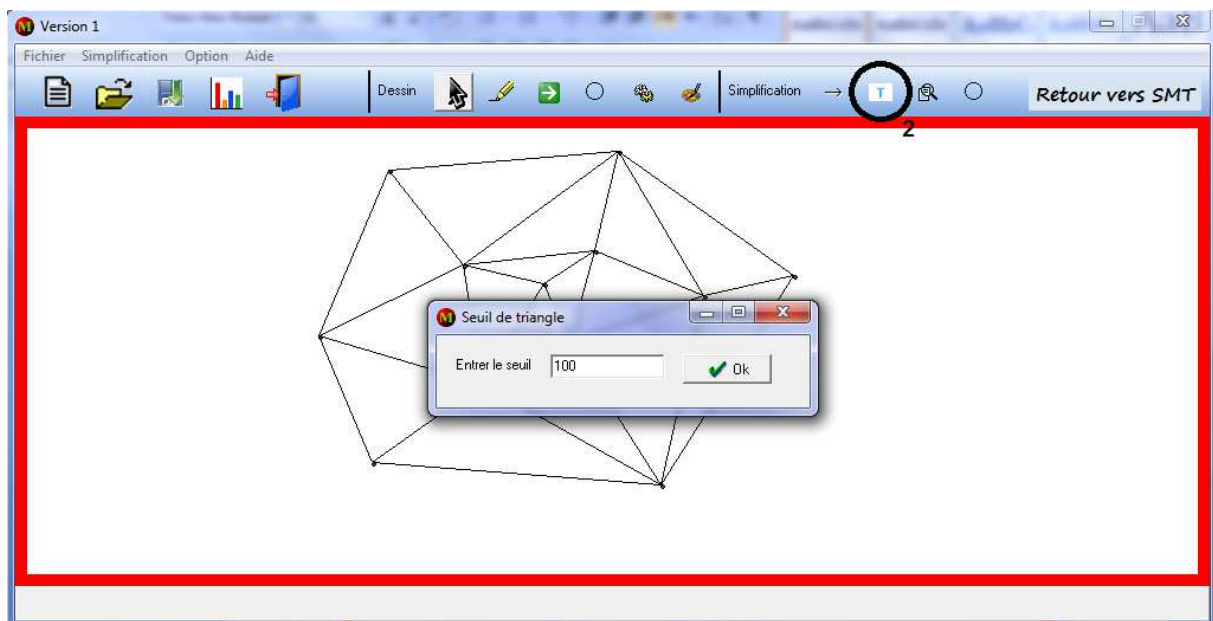


Figure III.16 : Simplification par suppression de triangles.

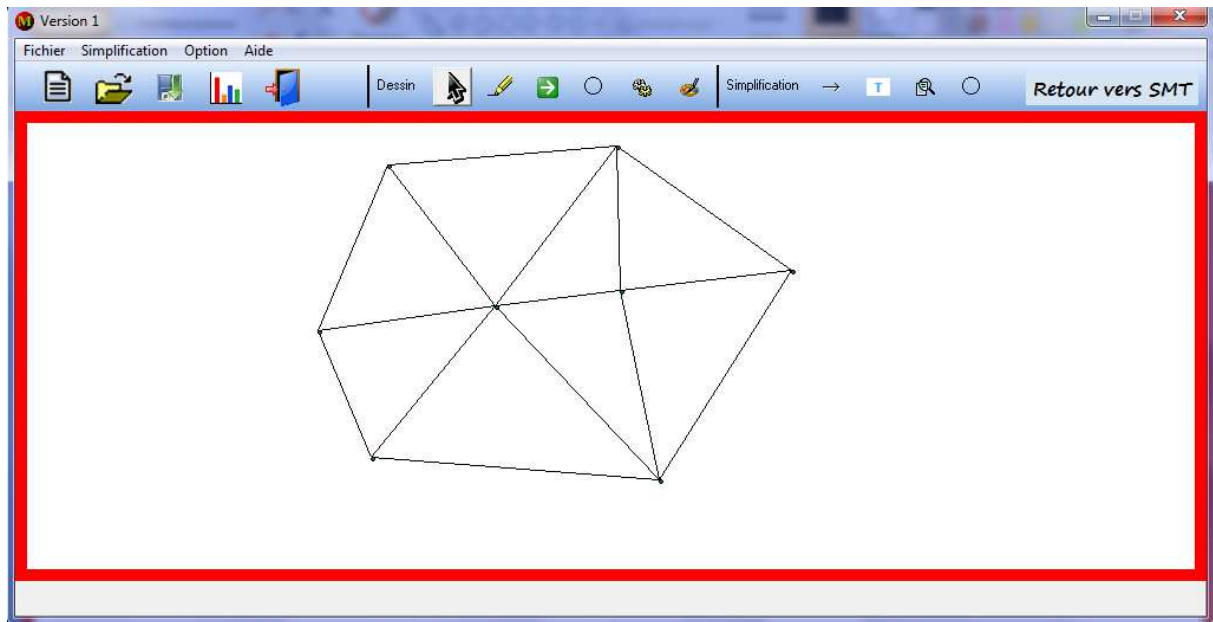


Figure III.17 : Triangulation simplifiée par suppression de triangles après la confirmation.

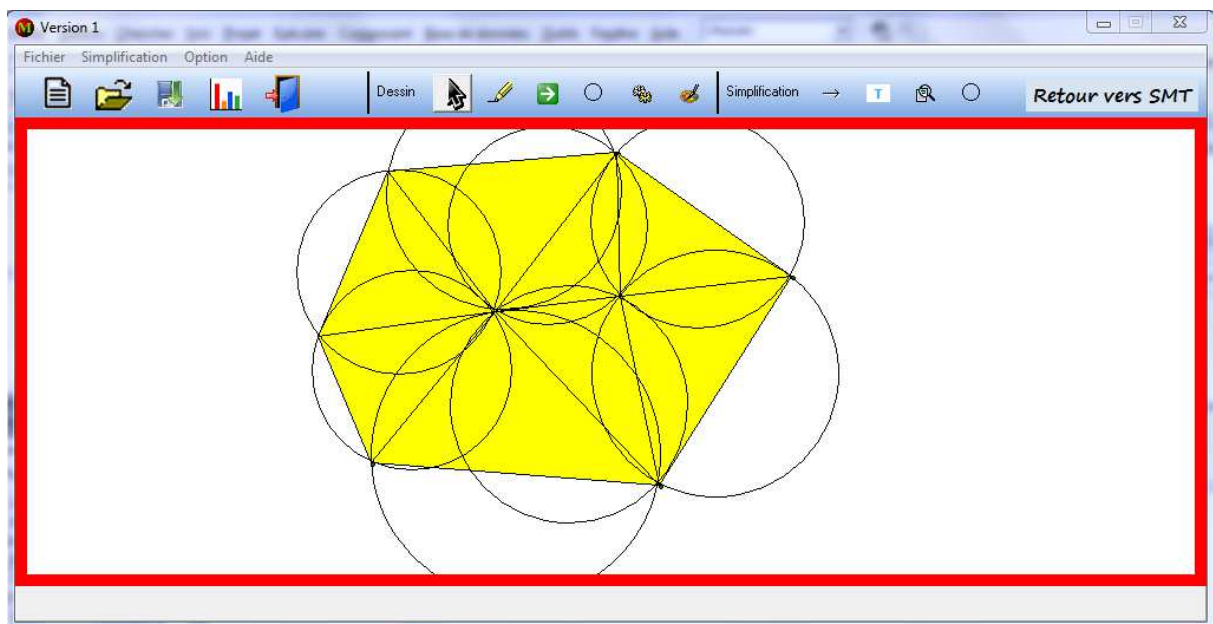


Figure III.18 : Confirmation par les cercles circonscrits.

Remarque : la simplification par suppression de triangles respecte aussi la contrainte de Delaunay.

❖ Version 2 :

En cliquant sur la Version 2 **Version 2**, l'interface suivante va apparaître :

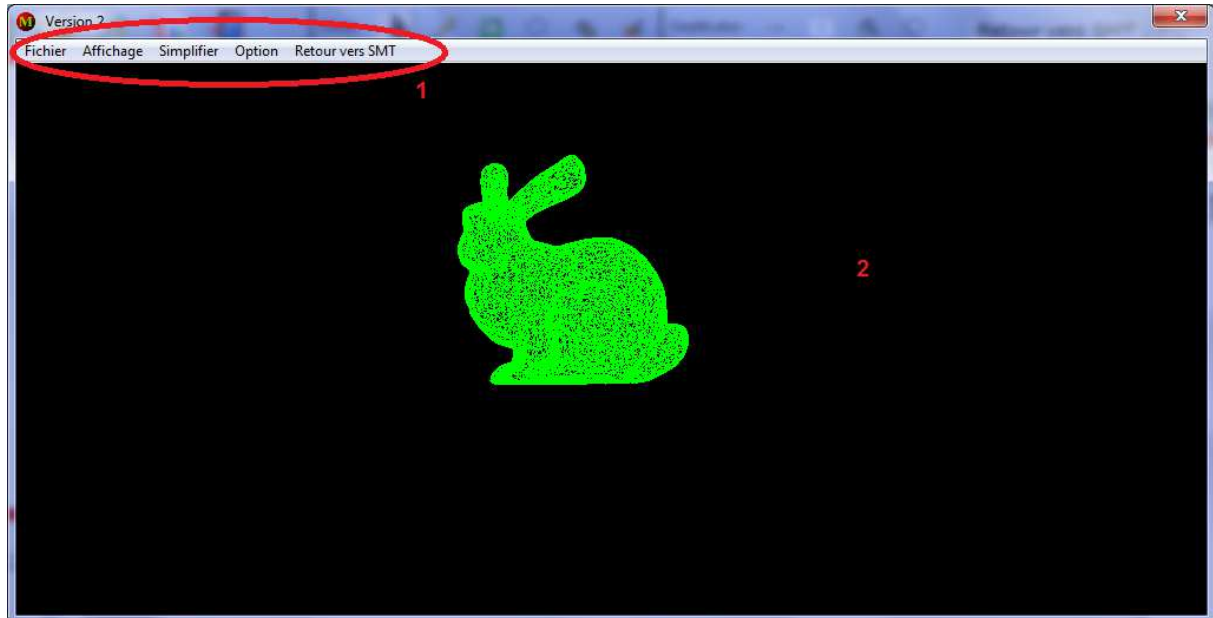


Figure III.19 : Interface de la version 2.

« 1 » Menu :

Fichier :

- Ouvrir (charger un fichier en extension .OFF).
- Enregistrer (permet l'enregistrement sous format .OFF).
- Quitter (fermer la version 2).

Affichage :

- Zoom (zoom + "avant", zoom - "arrière").
- Couleur (Noir, Blanc, Gris).

Simplification :

- Par triangle (suppression de triangle).
- Par arête (contraction d'arête).

Option :

- Seuil.
- Statistiques (permet d'afficher les points et les triangles avant et après la simplification).
- Temps d'exécution.

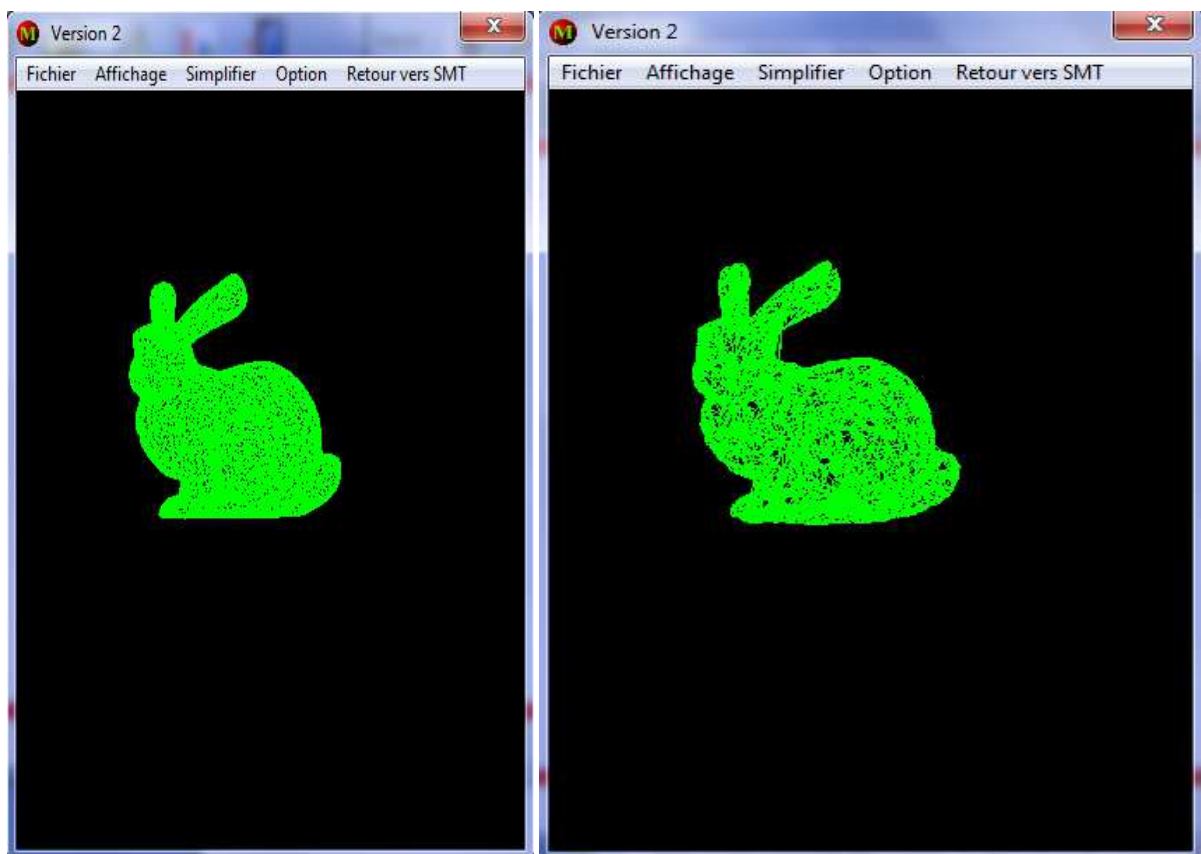
Retour vers SMT: permet de revenir à l'interface principale.

« 2 » : espace réservé pour l'affichage

Pour faire le test des deux méthodes, on va choisir un objet triangulé (lapin) contenant 4052 points et 8100 triangles.

a. La méthode 1 : la simplification du lapin par la contraction d'arête :

Dans cette première expérience, on va essayer d'appliquer la méthode 1 pour le lapin.



Maillage original avec
4052 points et 8100 triangles

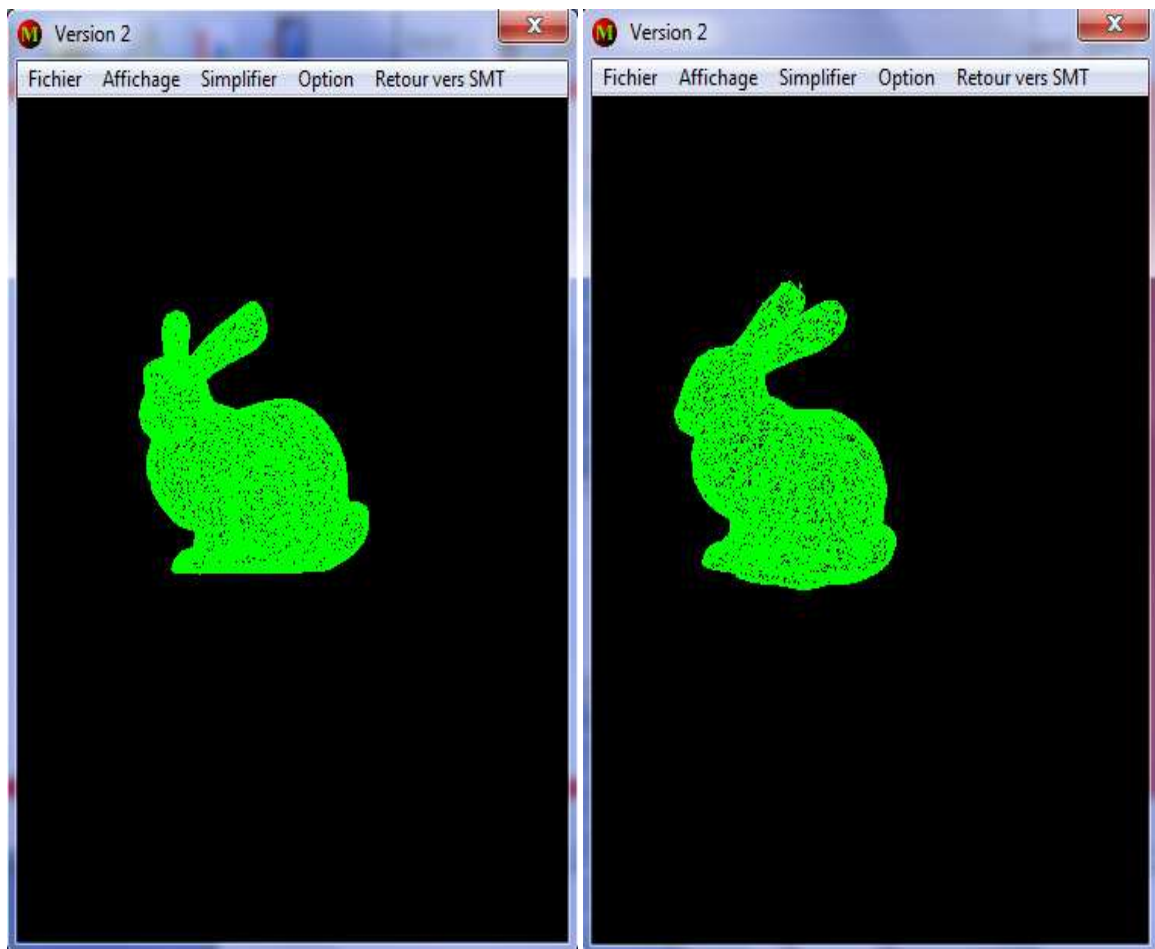
maillage simplifié avec
566 points et 4691 triangles

Figure III.20 : Simplification du lapin par la méthode 1.

Remarque : Après la simplification par la méthode 1, on a obtenu 566 point et 4691 triangles, la forme commence à se détruire.

b. La méthode 2 : la simplification du lapin par la suppression de triangles:

Dans la deuxième expérience, on va essayer d'appliquer la méthode 2 pour le lapin.

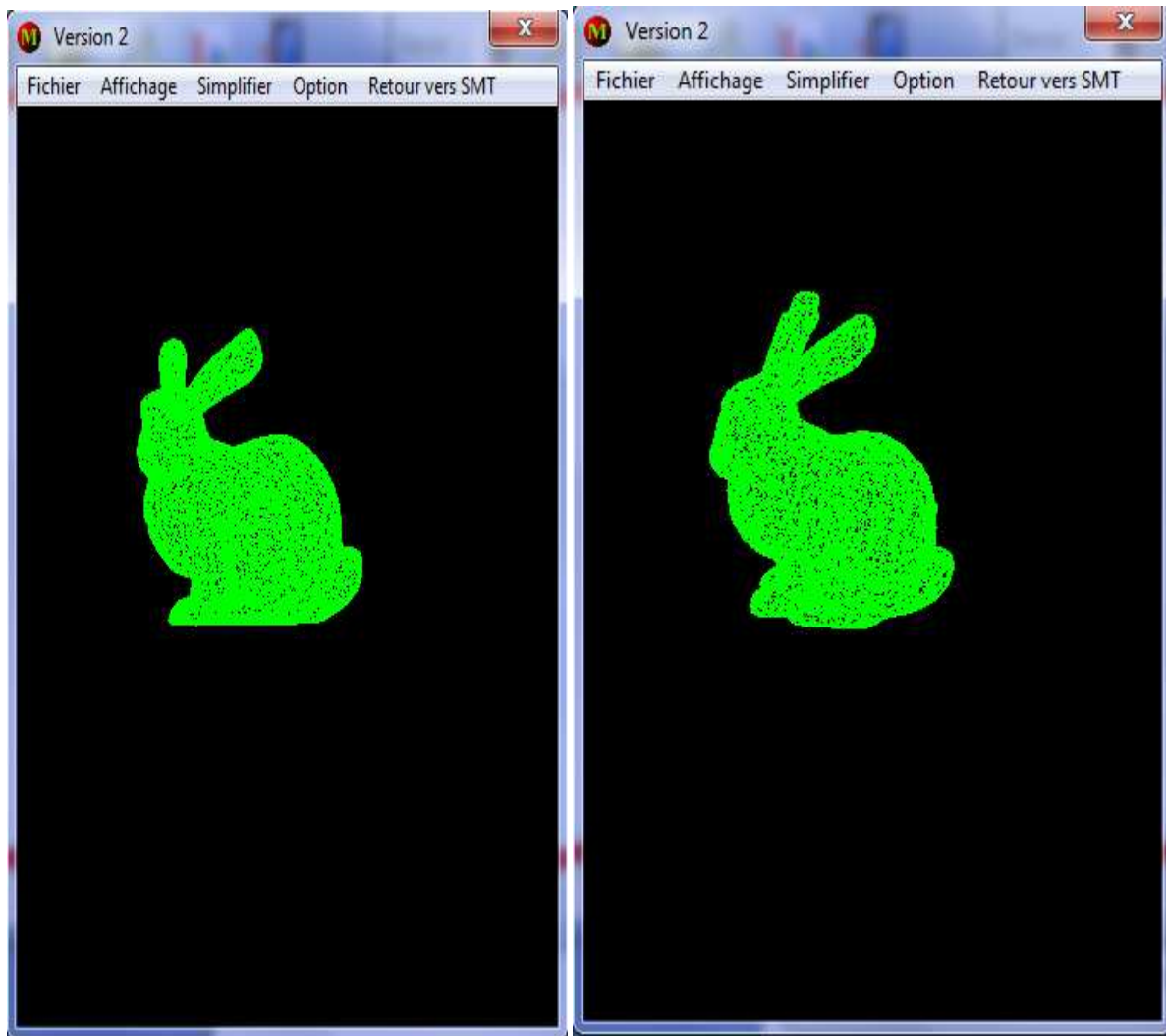


Maillage original avec
4052 points et 8100 triangles

Maillage simplifié avec
2810 points et 5616 triangles

Figure III.21 : Simplification du lapin par la méthode 2.

Remarque : Après la simplification par la méthode 2, on a obtenu 2810 point et 5616 triangles, la forme commence à se détruire.

c. Méthode combinée : (contraction d'arête et suppression de triangle)

Maillage original avec
4052 points et 8100 triangles

Maillage simplifié avec
3038 points et 6408 triangles

Figure III.22 : Simplification combinée du lapin.

Remarque : on remarque que la simplification combinée de deux méthodes précédentes déforme l'objet à partir de 3038 points et 6408 triangles.

III.5.Conclusion :

On conclu que :

- ✓ La simplification par la contraction d'arête est plus rapide par rapport à la simplification par la suppression de triangles.
- ✓ Le nombre de points supprimé en méthode 1 est grand par rapport à la méthode 2.
- ✓ La combinaison des deux méthodes déforme l'objet à partir d'un nombre de points et de triangles plus élevé que pour chaque méthode à part.

CONCLUSION GENERALE :

Le domaine de la simplification et de la visualisation de gros maillages de données dans lequel s'inscrit notre projet a connu un essor conséquent lors de la dernière décennie. De nombreux algorithmes résultent des années de recherche permettant de réaliser des simplifications de maillages puis ensuite de les visualiser ou de les stocker de manière efficace.

Après la simplification du maillage dans le SIG, on obtient pas mal des gains tels que la réduction du nombre d'éléments (espace de stockage) et la qualité des produits dérivés.

Nous avons présenté dans ce mémoire une brève étude sur le maillage et on a parlé des principales méthodes de simplification de ce dernier, par la suite, nous avons choisi d'implémenter deux méthodes (contraction d'arête et suppression de triangles) pour que l'on puisse faire une comparaison entre elles et donner plus de choix à l'utilisateur. Les méthodes implémentées ont donné des résultats satisfaisants au moins sur l'aspect visuel qui reste un facteur élémentaire dans le jugement de l'efficacité de la méthode. Par rapport au temps d'exécution et le nombre de points et de triangles supprimés, la contraction d'arête est meilleurs que la suppression de triangles.

Ce projet nous a permis de :

- Découvrir un domaine très vaste qui est les maillages.
- Se perfectionner dans le C++ Builder.
- S'initialiser à la géométrie algorithmique et l'OpenGL.

ANNEXE

Description de format de fichier .OFF :

Le format .OFF est un format général utilisé pour décrire un maillage quelconque (triangulaire ou autres) sa description est comme suit :

| | |
|--|--|
| OFF | La chaîne de caractères « OFF » |
| Nb_S Nb_F Nb_A | Nombre de sommet, de face et 'arrêtes |
| X1 Y1 Z1 | Coordonnées 1 ^{er} sommet |
| X2 Y2 Z2 | Coordonnées 2 ^{eme} sommet |
| | Les coordonnées des autres sommets |
| Nb1 S ₁₁ S ₁₂ S ₁₃ ...S _{1nb1} | Nombre de sommets constituant la 1ere face suivi des indices des sommets |
| Nb2 S ₂₁ S ₂₂ S ₃₃ ...S _{1nb1} | Même chose pour la 2 ^{eme} face |
| | Même chose pour les autres faces |

BIBLIOGRAPHIE :

- [1] D.P. LUEBKE, A developer's survey of polygonal simplification algorithms, IEEE. Computer Graphics and Applications. 2001

- [2] Benhamada khadidja. « Compression des Maillages Triangulaire Méthode Kd-tree » mémoire de licence en informatique 2007 - 2008.

- [3] Tamy Boubekour : «Reconstruction, Simplification et Subdivision de surfaces 3D De l'acquisition à la compression des objets 3D – GdR ISIS – Avril 2011.

- [4] Fabien VIVODTZEV : « Simplification et visualisation de gros ensembles volumiques » Rapport de stage; 18 juin 2002

- [5] Sébastien BARBIER ; « Exploration interactive de gros maillages surfaciques et volumiques » ; Image Vision et Robotique 2005 - 2006

- [6] Jonathan Cohen, Amitabh Varshney, Dinesh Manocha, Greg Turk, Hans Weber, Pankaj Agarwal, Frederick Brooks, and William Wright. Simplification envelopes. Computer Graphics, 30 (Annual Conference Series) : 119–128, 1996.

- [7] Florian Blanc et Clément Ramel, Projet industriel ; Mémoire de projet : « Visualisation de maillages par LODs » ; Master Sciences de l'Information et des Systèmes spécialité Imagerie Numérique ; 2010 - 2011

- [8] BENABADJI Mehdi, DELMAS Gabriel, LOMBARD Matthieu et OCHMAN Cecile ; « Navigation interactive dans un paysage urbain » ; Rapport du TER FMIN200 du Master IFPRU Informatique ; 2009

- [9] Greg Turk. Re-tiling polygonal surfaces. Computer Graphics, 1992.

- [10] Sébastien Valette, Fabien Thibon, Yun-Sang Kim, Ho-Youl Jung, Isabelle Magnin et Rémy Prost : « Décomposition en ondelettes de maillages triangulaires 3D irrégulièrement subdivisés ». Application à la compression. Dix-septième colloque GRETSI, Vannes, 13-17 septembre 1999
- [11] Fabien VIVODTZEV « Hiérarchisation et visualisation multirésolution de résultats issus de codes de simulation » : Thèse présentée pour l'obtention du titre de Docteur de l'Université Joseph Fourier Spécialité Informatique ; soutenue le 5 décembre 2005.
- [12] G. Lavoue, F. Dupont and A. Baskurt, "Constant Curvature Region Decomposition of 3D-Meshes by a Mixed Approach Vertex-Triangle.", *Journal of WSCG*, vol. 12, 2004.
- [13] Pattern Recognition: "Boundary-trimmed 3D triangular mesh segmentation based on iterative merging strategy", Volume 39, Issue 5, May 2006.
- [14] Hugues Hoppe. Progressive meshes. *Computer Graphics*, 30(Annual Conference Series): 1996.
- [15] T. He, L. Hong, A. Varshney, and S. Wang. Controlled topology simplification. *IEEE Transactions on Visualization and Computer Graphics*, June 1996.
- [16] F. Nooruddin and G. Turk. Simplification and repair of polygonal models using volumetric techniques. *IEEE Transactions on Visualization and Computer Graphics*, April 2003.
- [17] Jonathan Cohen, Marc Olano, and Dinesh Manocha. Appearance-preserving simplification. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM Press, 1998.

WEBGRAPHIE:

- <http://research.microsoft.com/en-us/um/people/hoppe/>
- <http://www.cc.gatech.edu/~turk/pubs.html>
- <http://www.informatik.uni-trier.de/~ley/pers/hd/h/Hoppe:Hugues.html>
- <http://www.commentcamarche.net/>
- <http://www.codes-sources.com/>
- <http://www.siteduzero.com/>
- <http://www.developpez.com/>