

## Exercice 1 : Etude de méthodes de simplification avec MeshLab

### 1) Quadric Edge-Collapse decimation

On choisit un seuil de manière arbitraire. Ce seuil va permettre à l'algorithme de choisir des arêtes ayant une taille inférieure à ce seuil. Pour toutes les arêtes sélectionnées, il y a deux méthodes possibles :

- Edge-Collapse : On calcul la position du point au milieu de l'arête et on crée un sommet à cette position. On supprime ensuite les deux sommets ayant permis de calculer sa position et on relie les sommets annexes à notre nouveau sommet.
- Half Edge-Collapse : On choisit un des deux sommets de l'arête pour le supprimer. L'autre sommet devient le nouveau sommet reliant tous les sommets annexes.

Le but est donc de supprimer les plus petites arêtes de sorte à garder que les triangles les plus gros et simplifier la forme.

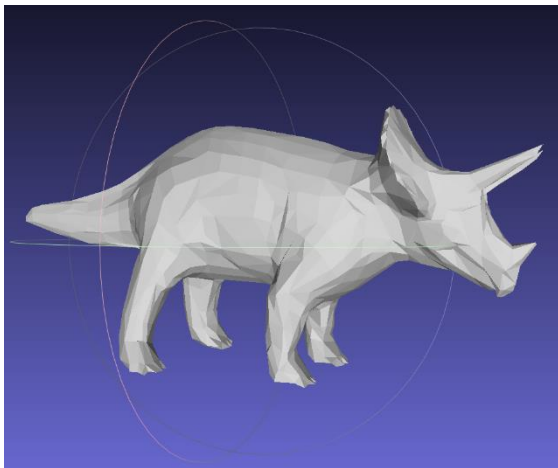


Figure 1 : avant simplification – 998 faces

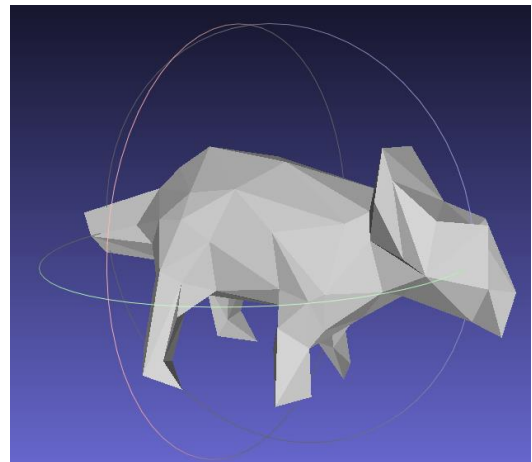


Figure 2 : Après simplification – 200 faces

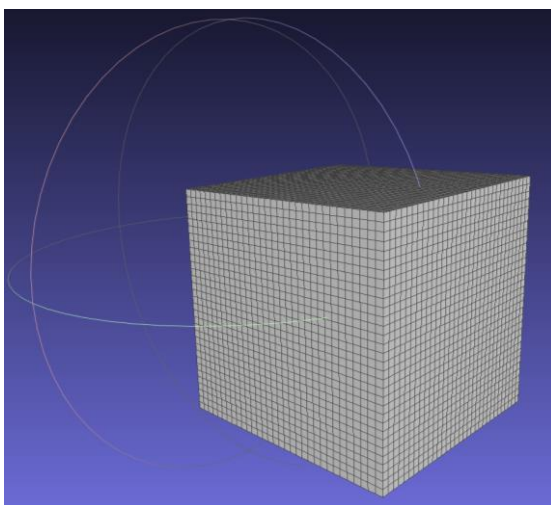


Figure 4 : Avant simplification

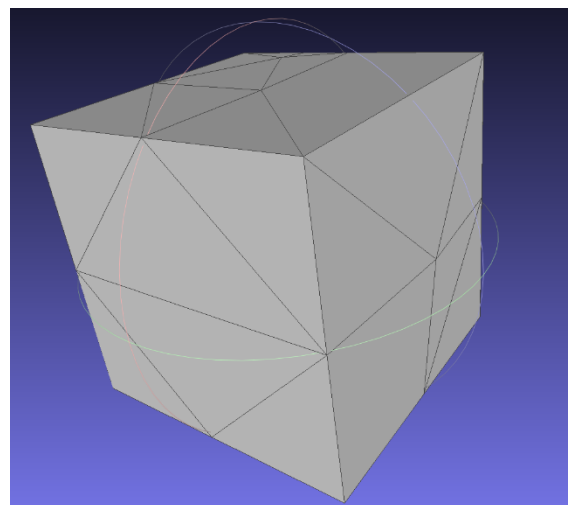


Figure 3 : Après simplification

Il est possible de jouer sur plusieurs paramètres pour faire varier le nombre de face :

Simplification: Quadric Edge Collapse Decimation

*Simplify a mesh using a quadric based edge-collapse strategy. A variant of the well known Garland and Heckbert simplification algorithm with different weighting schemes to better cope with aspect ratio and planar/degenerate quadrics areas.*  
*See:*  
*M. Garland and P. Heckbert.*  
**Surface Simplification Using Quadric Error Metrics** ([pdf](#))  
*In Proceedings of SIGGRAPH 97.*

Target number of faces

Percentage reduction (0..1)

Quality threshold

Preserve Boundary of the mesh ☐

Boundary Preserving Weight

Preserve Normal ☐

Preserve Topology ☐

Optimal position of simplified vertices ☒

Planar Simplification ☐

Planar Simp. Weight

Weighted Simplification ☐

Post-simplification cleaning ☒

Simplify only selected faces ☐

Default Help

Close Apply

- Target number of faces : donne le nombre de face à atteindre
- Pourcentage reduction : donne le nombre de face que l'on veut supprimer
- Quality threshold : Aucun changement constaté

Sur les arêtes vives, comme les arêtes du bord du cube, aucune modification n'a été apportée. L'algorithme ne marche pas sur les arêtes vives.

## 2) Clustering decimation

On crée une grille contenant des cellules de tailles égales et arbitraires. Pour chaque cellule de la grille, on calcule un sommet représentatif en réalisant une moyenne de tous les sommets présents dans la cellule. On connecte ensuite deux sommets entre eux si dans le maillage d'origine les deux sommets étaient reliés. Ce qui fait la précision de la forme est la taille de chaque cellule de la grille. En effet, plus les cellules sont petites, plus la forme possède de point et donc plus la forme est précise.

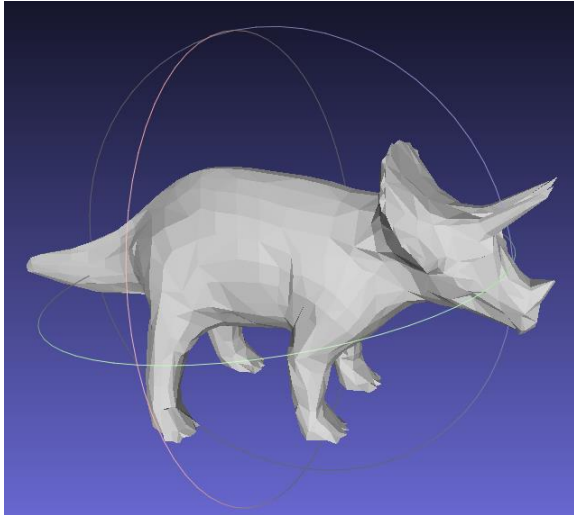


Figure 7 : Avant simplification - taille de cellules = 0.20161

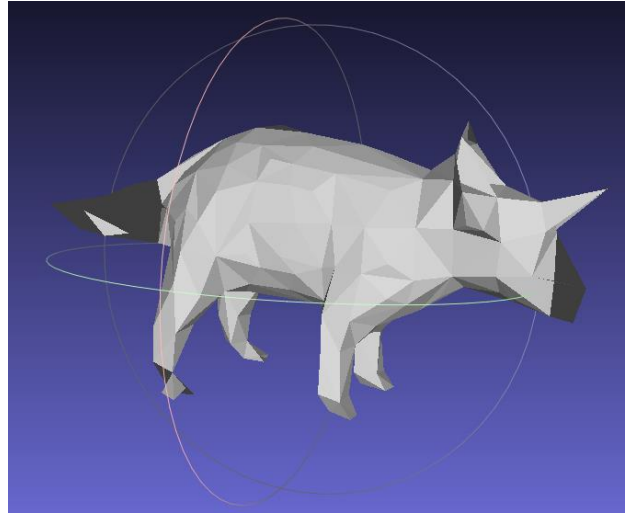


Figure 8 : Après simplification - taille des cellules = 1.00804

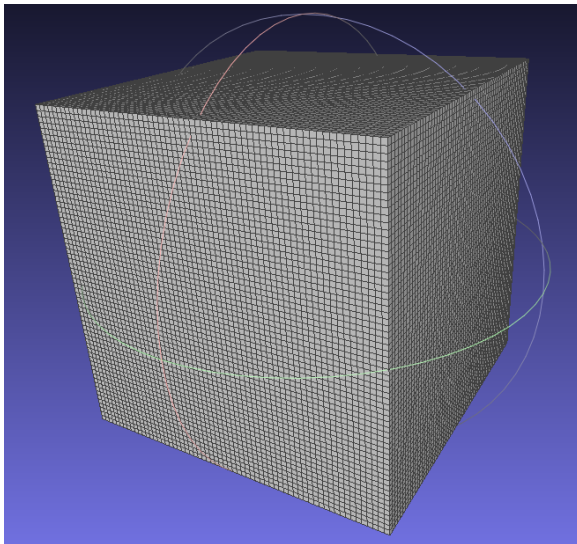


Figure 6 : Avant simplification - taille des cellules : 0.034641

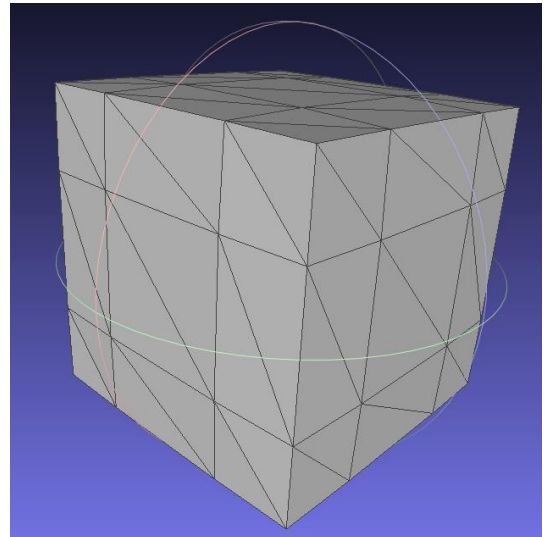


Figure 5 : Après simplification - taille des cellules : 1.044051

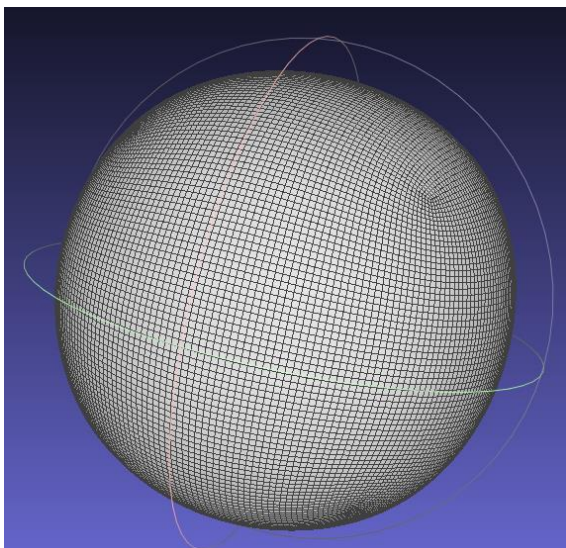


Figure 10 : Avant simplification - taille des cellules = 0.29081

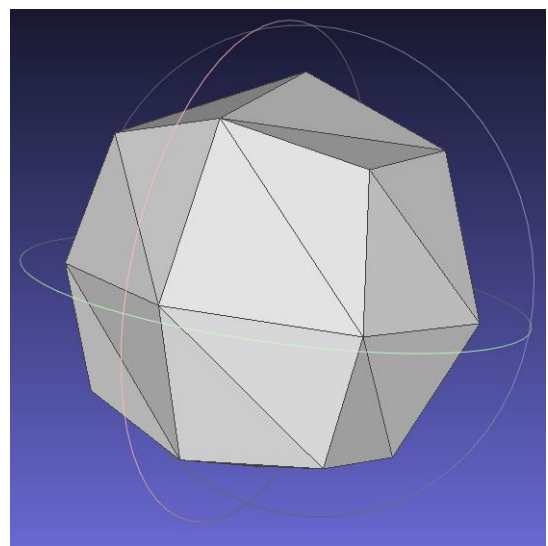
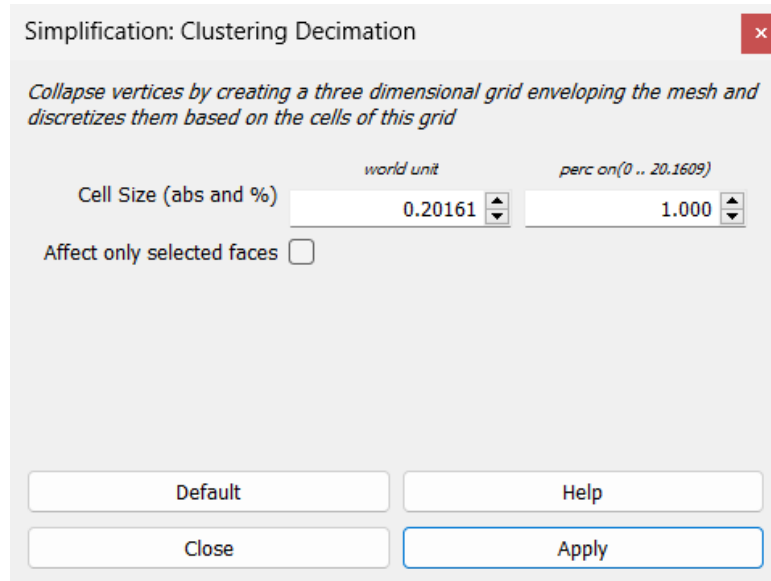


Figure 9 : Après simplification - taille des cellules = 0.523464

Note : crash du logiciel lorsque la taille des cellules est trop grosse pour le cube et la sphère.

Un seul paramètre peut être changé :



- Cell size : la taille des cellules de la grille. Comme dit précédemment, plus la cellule est grande, moins la forme sera précise.

### 3) Comparaison des deux méthodes de simplification

	Point positif	Point négatif
Quadric Edge-Collapse decimation	<ul style="list-style-type: none"><li>- Conserve le plus possible la forme originelle</li><li>- Possède plus de réglages permettant de conserver la topologie, les normales et autre...</li></ul>	<ul style="list-style-type: none"><li>- Algorithme lourd et lent avec des crash du logiciel qui surviennent lorsque l'on a une forme avec beaucoup de triangles.</li></ul>
Clustering decimation	<ul style="list-style-type: none"><li>- Algorithme plus léger car utilise énormément de calculs rapides à exécuter</li><li>- Plus simple à utiliser car peu de paramètres</li></ul>	<ul style="list-style-type: none"><li>- Ne conserve pas la forme originelle car agit sur toutes les arêtes</li></ul>