

---

# Crypto-Compression 3D

Compte Rendu 7

---

Potin Clément, Fournier Romain  
Master 2 IMAGINE  
Université de montpellier  
2021



# Problématique

L'objectif est de développer un algorithme qui, de manière conjointe, compresse avec ou sans pertes et chiffre un objet 3D. Les performances de cette méthode seront ensuite mesurées en termes de taux de compression et de qualité de reconstruction de l'objet 3D.

## Edgebreaker : finalisation

Cette semaine a vu l'aboutissement de notre travail sur l'algorithme de compression et décompression de maillages 3D "*Edgebreaker*".

Cet algorithme, si efficace pour ce pour quoi il a été conçu, mais tellement complexe à mettre en place, aura occupé une bonne partie de notre temps ces dernières semaines. La partie décompression, sur laquelle nous étions bloqués depuis bientôt 3 semaines, aura nécessité de nombreuses implémentations infructueuses ainsi que de multiples réécritures de l'algorithme de compression (afin d'adapter son fonctionnement et ses données de sortie aux implémentations de l'algorithme de décompression).

Ce temps de développement particulièrement important aura été en partie notre faute : nous avons visé trop haut dès le début de l'implémentation de l'algorithme. Plutôt que de nous contenter des versions les plus "simples" d'*Edgebreaker*, nous avons décidé d'implémenter une version adaptée aux maillages troués, non manifolds, et non homéomorphiques à une sphère, par peur de devoir réécrire entièrement l'algorithme pour prendre en compte ces cas particuliers (ce qui aurait en effet été le cas). Mais ces versions, très complexes et manquant d'exemples et cas d'implémentation concrets, n'auront pas abouti.

Finalement, notre algorithme *Edgebreaker* ne traite donc pas les maillages :

- Troués
- Non manifolds
- Non homéomorphiques à une sphère

Toutefois, nous veillerons à tout de même proposer une compression de ces maillages, bien que sans utiliser *Edgebreaker*. En effet, nous tâcherons de détecter tous ces cas particuliers et nous contenterons d'une compression "basique" pour ceux-ci, à savoir quantifier les positions des sommets et coordonnées de texture, indexer les normales dans un dictionnaire précalculé, les associer à un codage de *Huffman* adapté à chacun

de ces types de données, et enregistrer le tout sous format binaire pour optimiser l'espace de stockage utilisé.

## Pré-traitement des maillages

Nous avons déjà pu en parler la semaine dernière, mais certains maillages nécessitent un prétraitement avant de pouvoir être utilisables par nos algorithmes de compression.

Nos prétraitements suppriment maintenant les cas :

- Où un sommet est référencé plusieurs fois dans un même triangle
- Où plusieurs triangles sont identiques (référencent les mêmes 2 sommets)
- Où plusieurs sommets sont identiques (au mêmes coordonnées)
- Où des arêtes sont non manifolds (sont au bord de plus de 2 triangles)
- Où des sommets ne sont pas utilisés par des arêtes/triangles

Et permettent d'adapter au maximum les maillages à nos algorithmes, tout en supprimant des données inutiles (comme des sommets non utilisés, ou des sommets ou triangles dupliqués, notamment).

## Travail restant

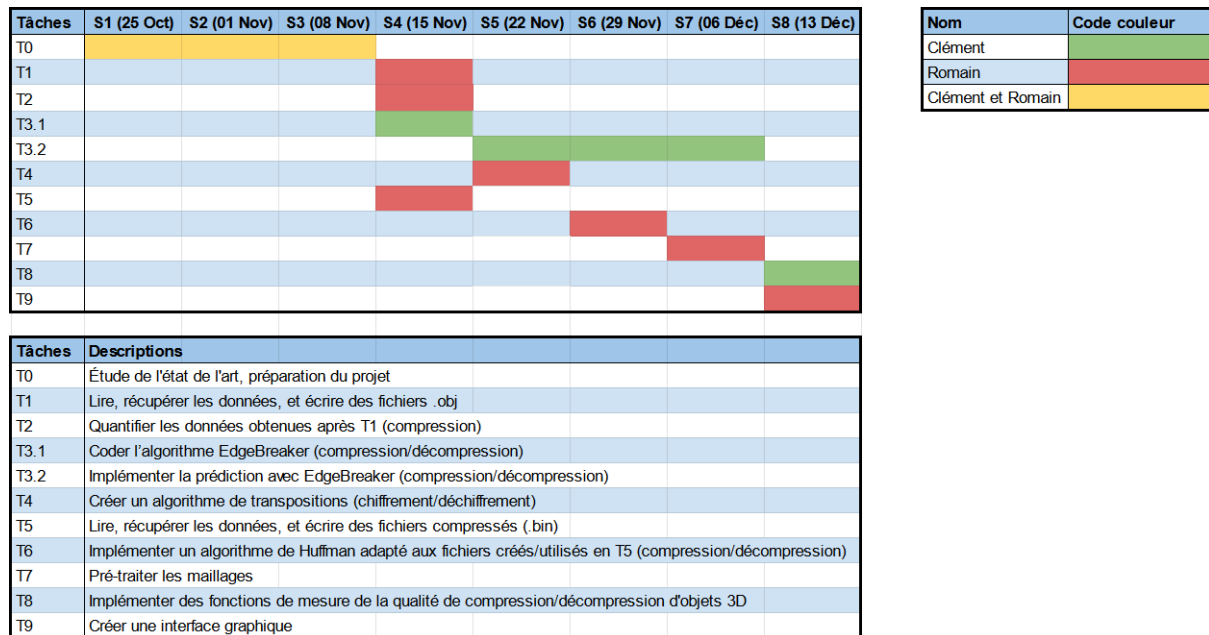
Conscients d'avoir encore un certain travail à fournir avant de pouvoir proposer une version distribuable de ce projet, nous sommes malgré tout fiers des algorithmes et techniques de compression que nous avons réussi à mettre en place.

Pour proposer une version satisfaisante de notre projet, nous devrions nous acquitter des tâches suivantes :

- Lier *Edgebreaker* aux autres modules de compression
- Adapter nos méthodes d'évaluation de la compression et qualité de reconstruction d'objets 3D à nos données
- Proposer une interface graphique permettant de compresser/décompresser des objets, et affichant diverses informations sur les objets (poids avant/après compression), indices de qualité de reconstruction de l'objet, ...
- Prendre en compte les coordonnées de texture de l'objet (si le fichier .obj en contient), y compris au sein de l'algorithme *Edgebreaker*

Pour améliorer les performances de nos algorithmes, nous avons envisagé l'implémentation de multiprocessing. Toutefois, notre algorithme le plus coûteux en temps, *Edgebreaker*, nécessite l'identification de la configuration courante avant de pouvoir passer à la suivante, et n'est donc pas adapté à ce genre d'optimisations.

## Gantt final



*Décomposition finale des tâches*

## Lien du Git

Notre travail sera mis à jour au lien suivant :

<https://github.com/Romimap/3D-CryptoCompression/>

# Références

## Demos & Softwares

1. Vladimir Agafonkin, “Edgebreaker, the Heart of Google Draco” :  
<https://observablehq.com/@mourner/edgebreaker-the-heart-of-google-draco>
2. Google Draco:  
<https://github.com/google/draco>

## Papers

3. Michael Deering, “Geometry Compression”, sun Microsystems, 1995 :  
[http://web.cse.ohio-state.edu/~shen.94/Su01\\_888/deering.pdf](http://web.cse.ohio-state.edu/~shen.94/Su01_888/deering.pdf)
4. Chandrajit L Bajaj, Valerio Pascucci, Guozhong Zhuang, “Single Resolution Compression of Arbitrary Triangular Meshes with Properties”, University of Texas, 1997:  
<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.14.946&rep=rep1&type=pdf>
5. Mike M. Chow, “Optimized Geometry Compression for Real-time Rendering”, Massachusetts institute of Technology, May 1997:  
<https://www.semanticscholar.org/paper/Optimized-geometry-compression-for-real-time-Chow/39babe9519e6b58bae4350e4f57be86dc45ce6f9>
6. Jarek Rossignac, “Edgebreaker: Connectivity compression for triangle meshes”, Georgia Institute of Technology, 1999 :  
<https://www.cc.gatech.edu/~jarek/papers/EdgeBreaker.pdf>
7. Daniel Cohen-Or, David Levin, Offir Remez, “Progressive Compression of Arbitrary Triangular Meshes”, Tel Aviv University, 1999:  
<https://www.tau.ac.il/~levin/vis99-dco.pdf>
8. Jarek Rossignac, Alla Safonova, Andrzej Szymczak, “3D Compression Made Simple: Edgebreaker on a Corner-Table”, College of Computing and GVC Center, Georgia Institute of Technology, 2001:  
[https://www.researchgate.net/publication/3896746\\_3D\\_compression\\_made\\_simple\\_Edgebreaker\\_with\\_ZipandWrap\\_on\\_a\\_corner-table](https://www.researchgate.net/publication/3896746_3D_compression_made_simple_Edgebreaker_with_ZipandWrap_on_a_corner-table)

9. Pierre Alliez, Mathieu Desbrun, "Progressive Compression for Lossless Transmission of Triangle Meshes", University of Southern California, February 2002:  
[https://www.researchgate.net/publication/2534417\\_Progressive\\_Compression\\_for\\_Lossless\\_Transmission\\_of\\_Triangle\\_Meshes](https://www.researchgate.net/publication/2534417_Progressive_Compression_for_Lossless_Transmission_of_Triangle_Meshes)
10. Jarek Rossignac, "3D mesh compression", College of Computing and GVV Center Georgia institute of Technology, January 2003:  
[https://www.researchgate.net/publication/27521282\\_3D\\_Mesh\\_Compression](https://www.researchgate.net/publication/27521282_3D_Mesh_Compression)
11. Esam Elsheh, A. Ben Hamza, "Secret sharing approaches for 3D object encryption", Concordia Institute for Information Systems Engineering, Concordia University, Montréal, QC, Canada, 2011:  
<https://www.sciencedirect.com/science/article/abs/pii/S095741741100724X>
12. In-Ho Lee and Myungjin Cho, "Optical Encryption and Information Authentication of 3D Objects Considering Wireless Channel Characteristics", Department of Electrical, Electronic, and Control Engineering, Hankyong National University, Ansong 456-749, Korea, October 2013:  
[https://www.osapublishing.org/DirectPDFAccess/766AED72-4C7E-47EE-BB28209C333886A8\\_276786/josk-17-6-494.pdf](https://www.osapublishing.org/DirectPDFAccess/766AED72-4C7E-47EE-BB28209C333886A8_276786/josk-17-6-494.pdf)
13. Marc Éluard, Yves Maetz, and Gwenaél Doërr, "Geometry-preserving Encryption for 3D Meshes", Technicolor R&D France, November 2013:  
[https://www.researchgate.net/profile/Gwenael-Doerr/publication/273257218\\_Geometry-preserving\\_Encryption\\_for\\_3D\\_Meshes/links/54fc4b660cf2c3f52422a624/Geometry-preserving-Encryption-for-3D-Meshes.pdf](https://www.researchgate.net/profile/Gwenael-Doerr/publication/273257218_Geometry-preserving_Encryption_for_3D_Meshes/links/54fc4b660cf2c3f52422a624/Geometry-preserving-Encryption-for-3D-Meshes.pdf)
14. Xin Chen, Jingbin Hao, Hao Liu, Zhengtong Han and Shengping Ye, "Research on Similarity Measurements of 3D Models Based on Skeleton Trees", School of Mechatronic Engineering, China University of Mining and Technology, Daxue Road 1, Xuzhou 221116, China, State Key Laboratory of Materials Forming and Mould Technology, Huazhong University of Science and Technology, Luoyu Road 1037, Wuhan 430074, China, April 2017:  
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwi91fmtwZj0AhUI2BoKHYcKA1AQFnoECAMQAOQ&url=http%3A%2F%2Fwww.mdpi.com%2F2073-431X%2F6%2F2%2F17%2Fpdf-vor&usg=AOvVaw0lfo-8VxxYFuVQTyCt6JI>
15. Ying Zhou, Lingling Wang, Lieyun Ding, Cheng Zhou, "A 3D model Compression Method for Large Scenes", Huazhong Univ. of Science and Technology, 2018:  
<https://www.iaarc.org/publications/fulltext/ISARC2018-Paper207.pdf>