

---

# CSI : PCLTTM

---

Angel Lagrange

Amélie Rouge

Ainji Joma'a

Patick Quevillon

## 1 Pipeline complète du papier *Progressive Compression for Lossless Transmission of Triangle Meshes* (Alliez & Desbrun, SIGGRAPH 2001)

### 1.1 1. Prétraitement

Entrée : maillage triangulaire orientable et manifold (variété 2D plongée en 3D).

Objectif : encoder séparément mais de manière synchronisée la **connectivité** (topologie des triangles) et la **géométrie** (positions des sommets), de façon **progressive** et **sans perte**.

**Définition — Manifold :** Un maillage triangulaire est *manifold* si et seulement si chaque arête est adjacente à au plus deux faces et que le voisinage de chaque sommet est topologiquement équivalent à un disque (ou demi-disque pour une frontière).

Les opérations de décimation et de retriangulation manipulent des *patches*, c'est-à-dire les 1-rings centrés sur un sommet.

### 1.2 2. Structures fondamentales

- **Patch** : ensemble de faces incidentes à un sommet de valence  $d$ .
- **Independent set of patches** : ensemble de patches disjoints couvrant le maillage, chaque face appartenant à au plus un patch.
- **Null patch** : face ne faisant partie d'aucun patch (signalé par un code spécial  $N$ ).
- **Gate** : arête orientée pointant vers une face (*front face*) et un sommet (*front vertex*). Sert à parcourir le maillage.
- **Étiquettes (tags flags)** :
  - États de sommet/face : *free*, *conquered*, *to be removed*.
  - **Retriangulation tag** : signe + ou - indiquant s'il faut localement **maximiser** ou **minimiser** la valence lors du retriangulage.

### 1.3 3. Décimation guidée par la valence

**Principe :** On supprime successivement les sommets de valence  $\leq 6$  (ou  $\leq 4$  sur une frontière) pour réduire la redondance topologique. Chaque suppression crée un patch dont la topologie est connue.

**Conditions de sécurité :**

1. Ne pas violer la propriété **manifold** (pas de duplication d'arêtes existantes).
2. Éviter le **normal flipping**.

**Définition — Normal flipping :** Phénomène où la suppression d'un sommet et le retriangulage local inversent l'orientation des normales, provoquant un retournement de surface local.

3. Respecter les contraintes géométriques facultatives (basées sur volume ou distance patch).

**Décision :**

- Si valence  $v > 6$  : ne pas supprimer.
- Si valence  $v \leq 6$  : supprimer le sommet et retriangler le patch.

#### 1.4 4. Conquête décimante (Decimating Conquest)

**But :** parcourir la surface et supprimer les sommets admissibles tout en produisant les codes nécessaires à la reconstruction.

Données encodées :

- Pour chaque sommet supprimé : son **code de valence**  $v \in \{3, 4, 5, 6\}$ .
- Pour chaque face non couverte : un **code Null Patch (N)**.
- Ces codes forment la séquence  $A_i$  de la couche  $i$ .

**Algorithme :**

1. Initialiser une file FIFO de *gates* avec une arête de départ.
2. Tant que la file n'est pas vide :
  - (a) Si la *front face* est déjà conquise → ignorer.
  - (b) Si le *front vertex* est libre et  $v \leq 6$  :
    - Sortir le code  $v$ .
    - Taguer les voisins comme *conquered*.
    - **Retrianguler** localement le patch.
    - Effectuer l'**encodage géométrique** du sommet supprimé.
    - Ajouter les  $v - 1$  gates issues du patch dans la file.
  - (c) Sinon (vertex conquis ou  $v > 6$ ) :
    - Sortir un code  $N$  (null patch).
    - Pousser les deux gates restantes du triangle.

**Remarque :** le flux de valences  $\{v, N, v, N, \dots\}$  encode intégralement la topologie.

#### 1.5 5. Retriangulation adaptative

**But :** reconstruire localement les trous créés de manière **déterministe, réversible** et **régulière**. Cette étape décide comment refermer les patches après suppression d'un sommet tout en maintenant une connectivité compatible avec la décompression future.

**Principe général :** Lorsqu'un sommet de valence  $v$  est supprimé, il laisse un trou à  $v$  côtés. Ce trou peut être retriangulé de plusieurs façons : pour garantir que le même choix soit reproduit à la décompression, l'algorithme utilise une **table de retriangulation fixe** basée sur des **tags** + et - attribués aux sommets du bord.

**Définition — Table de retriangulation +/- :** Chaque arête du bord du patch porte une orientation locale. Les deux sommets extrémités de cette arête reçoivent chacun un **tag** :

- + : indique que le sommet doit **garder une valence élevée** après retriangulation (on favorise la création d'arêtes incidentes).
- - : indique que le sommet doit **réduire sa valence locale** (on évite de créer de nouvelles arêtes incidentes).

Ces tags sont dérivés du contexte local (valence avant suppression, état de conquête, type de gate). La combinaison des tags (+, +), (+, -), (-, +) ou (-, -) pour la gate active définit **de manière unique** le schéma de retriangulation à appliquer selon la table suivante (stockée symétriquement chez l'encodeur et le décodeur).

### Fonctionnement :

- Pour chaque patch supprimé, la table est consultée à partir des tags des deux sommets extrémités de la *gate courante*.
- La configuration (tag gauche, tag droit,  $v$ ) renvoie un unique modèle de triangulation prédéfini.
- Le retriangulage est effectué selon ce modèle, sans aucune décision géométrique.

**Définition — Régularité locale :** Un maillage est localement régulier si la distribution des valences reste concentrée autour de 6, garantissant des triangles équilibrés et bien formés.

### 1.6 6. Conquête nettoyante (Cleaning Conquest)

**But :** supprimer toutes les valences 3 créées précédemment. **Symétrie :** même mécanisme que la décimation, mais limité aux sommets  $v = 3$ .

### Données encodées :

- Un code 3 pour chaque sommet supprimé.
- Un code  $N$  pour chaque face non couverte.
- Cette séquence forme  $B_i$  pour la couche  $i$ .

**Résultat :** Une paire **Décimante + Nettoyante** correspond à une simplification  $3 \rightarrow 1$  (équivalent inverse d'une subdivision  $\sqrt{3}$ ).

### 1.7 7. Boucle de couches et compression du flot de codes

- On répète les paires (Décimante  $\rightarrow$  Nettoyante) jusqu'à obtenir la maille la plus grossière possible.
- Chaque paire  $(A_i, B_i)$  est encodée séquentiellement :

$$B_n, A_n, B_{n-1}, A_{n-1}, \dots, B_1, A_1$$

- Avant encodage :
  - Les codes 3 de  $B_i$  sont remplacés par 6 (meilleure statistique de probabilité).
  - Une simulation de décodage élimine  $\approx 10\%$  des codes  $N$  redondants.
- Compression via un **encodeur arithmétique adaptatif d'ordre 0**.

### 1.8 8. Encodage géométrique

Pour chaque sommet supprimé (c'est-à-dire chaque code de valence émis) :

1. **Quantification** : coordonnées des sommets sur 8 à 12 bits.
2. **Ordre implicite** : l'ordre des suppressions (issu de la conquête) définit l'ordre d'encodage.
3. **Prédiction barycentrique** :

$$\hat{v} = \frac{1}{n} \sum_{i=1}^n v_i$$

4. **Cadre de Frenet local** :

$$\begin{aligned} n &= \text{normale moyenne pondérée du patch,} \\ t_1 &= \text{projection de la gate sur le plan tangent,} \\ t_2 &= n \times t_1. \end{aligned}$$

5. **Coordonnées projetées** :

$$v_r = b + \alpha t_1 + \beta t_2 + \gamma n,$$

où  $(\alpha, \beta, \gamma)$  sont quantifiés puis encodés par l'arithmetic coder avec leur intervalle de variation.

## 1.9 9. Décodage progressif (client)

Le décodeur reçoit :

- les séquences compressées de valences et null patches ( $B_n, A_n, \dots, A_1$ ),
- les coordonnées géométriques correspondantes.

### Principe de reconstruction :

1. Le décodeur démarre d'une maille très grossière (dernière couche encodée).
2. À chaque étape, il **insère** les sommets supprimés précédemment, en suivant l'ordre inverse des conquêtes :  
$$(\text{Uncleaning } B_n) \rightarrow (\text{Undecimation } A_n) \rightarrow \dots \rightarrow (\text{Undecimation } A_1)$$
3. Pour chaque code de valence  $v$  :
  - on **découvre le patch** associé (grâce aux tags et à la structure FIFO),
  - on insère un nouveau sommet connecté aux  $v$  sommets du bord,
  - si le patch est nul ( $N$ ), on saute et attend le prochain code.
4. Le processus est entièrement synchronisé : chaque vertex inséré correspond à un code lu.

### Reconstruction géométrique :

- À chaque insertion :
  1. recalculer le barycentre et le cadre de Frenet local ( $t_1, t_2, n$ ) ;
  2. décoder  $(\alpha, \beta, \gamma)$  et reconstruire :
$$v_r = b + \alpha t_1 + \beta t_2 + \gamma n.$$
- Le maillage est progressivement raffiné, un sommet à la fois, jusqu'à retrouver la géométrie exacte originale.