

Chapitre 12

Subdivision de maillages

G. Gesquière

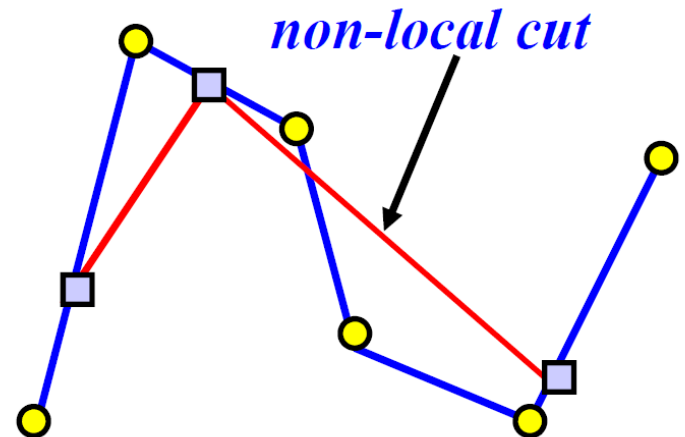
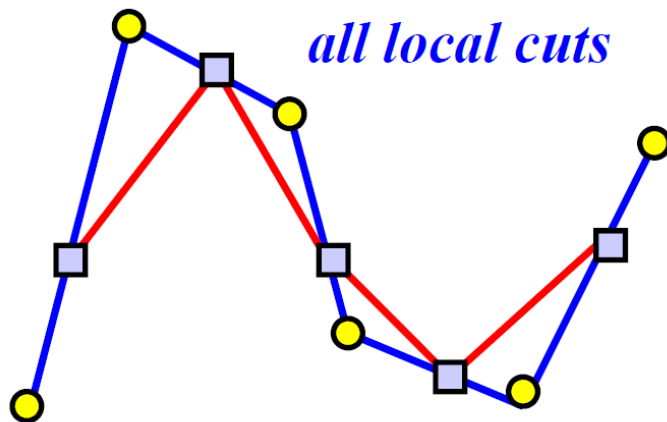
- Ce cours est une **compilation** :
 - Cours de Loïc Barthe, Modélisation géométrique (IRIT-UPS Toulouse; Equipe Vortex)
 - Cours S. Lanquetin, Université de Bourgogne
 - Cours G. Gesquière
 - Cours Ching-Kuang Shene

Simplifier ou subdiviser ?

- But : Avoir accès au même objet, mais avec des représentations ayant un nombre différent de polygones. Il s'agit donc de créer une hiérarchie de maillages. Ces hiérarchies peuvent être :
 - "Bottom-top": On part du modèle détaillé (les feuilles de la hiérarchie) et on va jusqu'à la forme la plus simplifiée. Les approches que l'on a vu jusqu'à maintenant, vont dans ce sens.
 - "Top-down": On part de la version simplifiée (la racine) et on ajoute progressivement des détails jusqu'à la représentation la plus fine. Des approches par surfaces de subdivision ou d'ondelettes suivent cette approche.

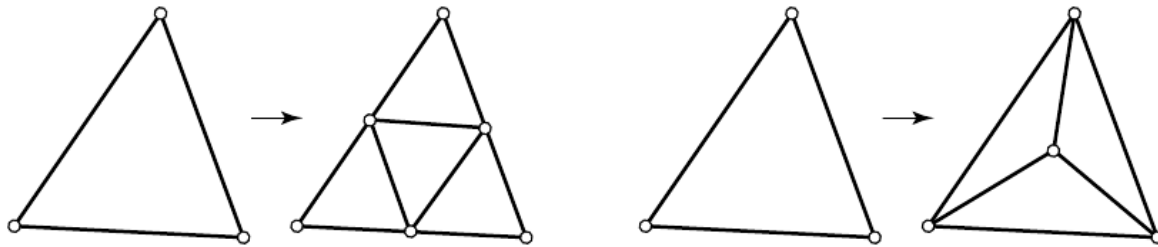
Un premier exemple : Curve Corner Cutting

- Take two points on different edges of a polygon and join them with a line segment. Then, use this line segment to replace all vertices and edges in between. This is corner cutting!
- Corner cutting can be local or non-local.
- A cut is **local** if it removes exactly one vertex and adds two new ones. Otherwise, it is **non-local**.

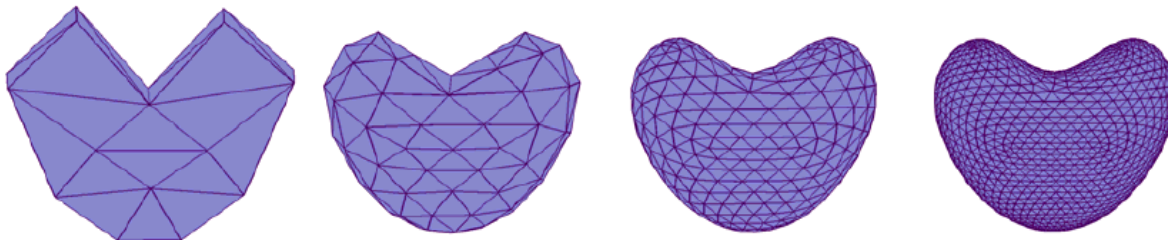


Raffinage/ Subdivision

- Raffinage par découpage de triangles

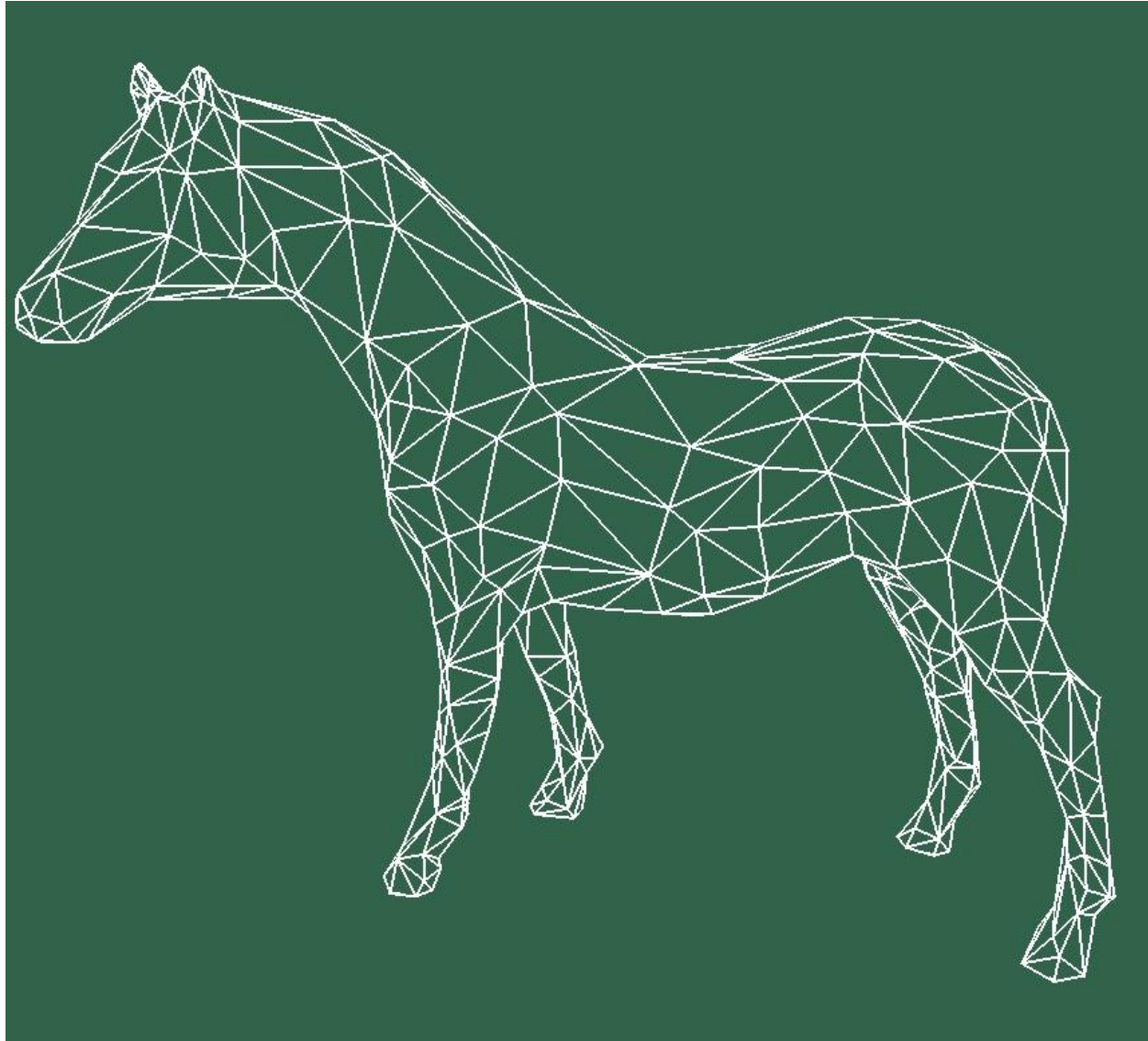


- Subdivision (règle permettant de positionner les points d'un objet suite à un raffinement)



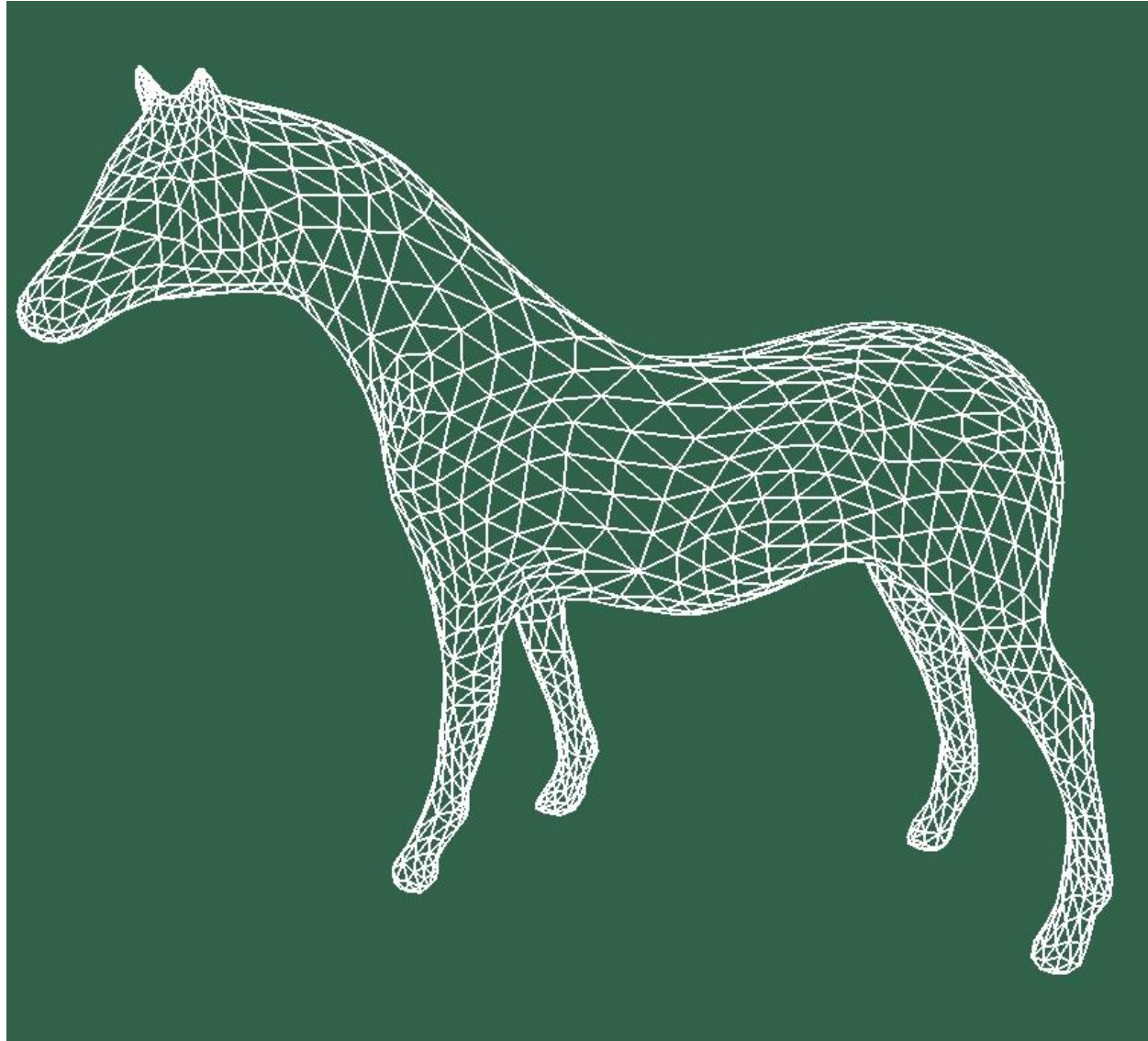
Exemple de subdivision

- Maillage initial



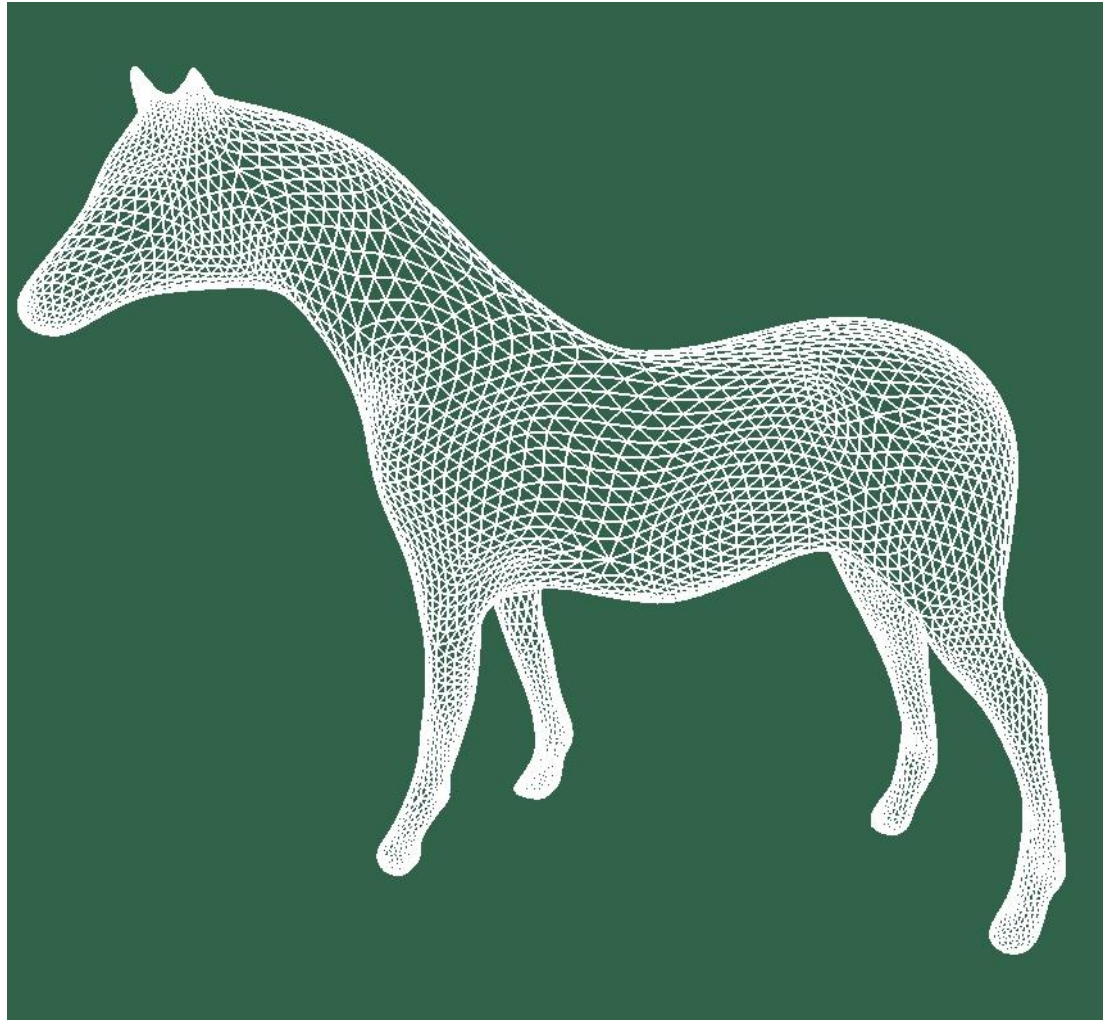
Exemple de subdivision

- Le maillage après un pas de subdivision



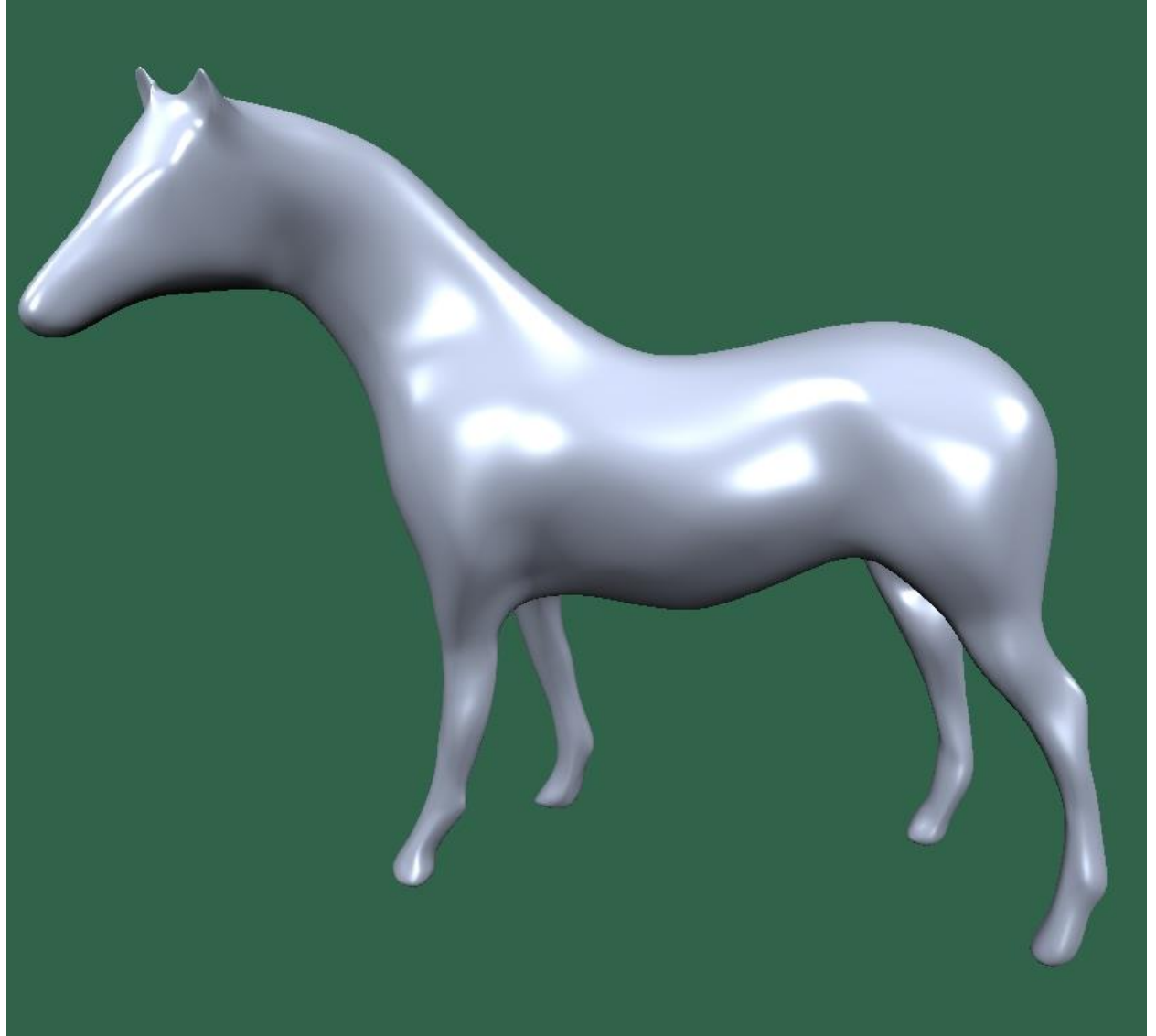
Exemple de subdivision

- Le maillage après deux pas de subdivision



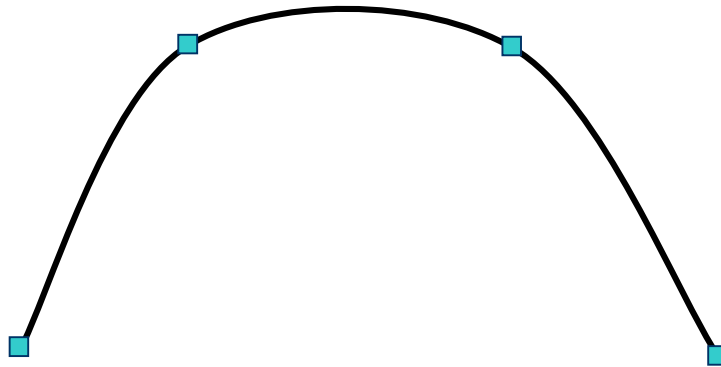
Exemple de subdivision

- La surface limite



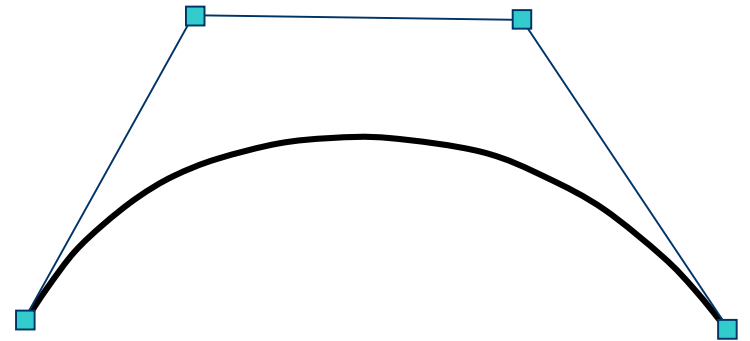
Rappel : Interpolation / Approximation

Interpolation



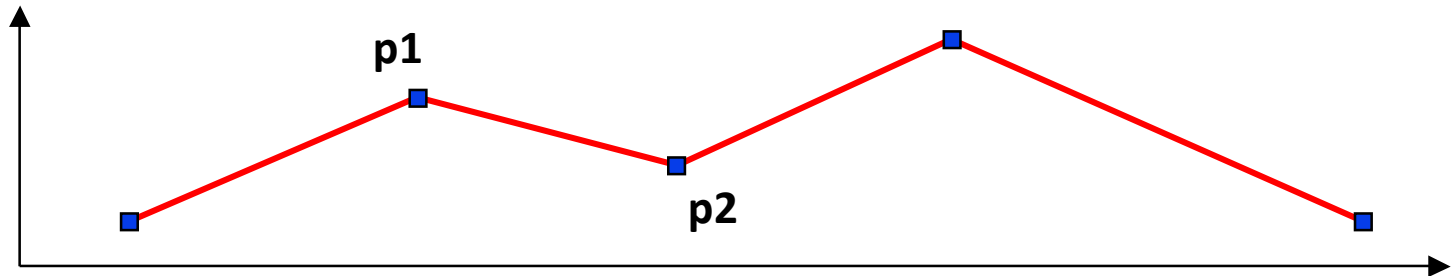
La courbe passe par
les points de contrôle

Approximation



La courbe est attirée par
les points de contrôle

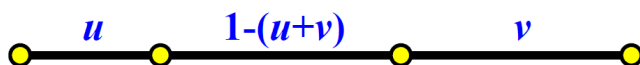
Rappel : Polynômes linéaires par morceaux



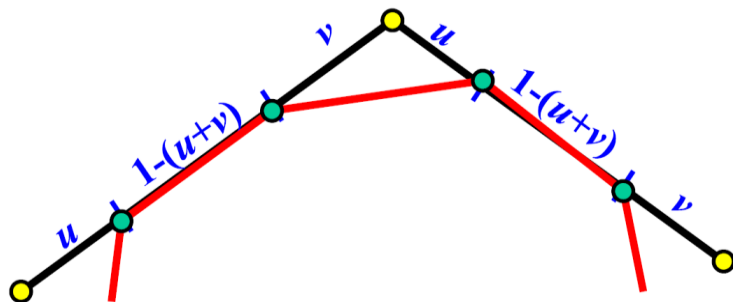
$$p(u) = up_1 + (1 - u)p_2$$

Simple Corner Cutting: 1/5

- On each edge, choose two numbers $u \geq 0$ and $v \geq 0$ and $u+v \leq 1$, and divide the edge in the ratio of $u: 1-(u+v): v$.



- Here is how to cut a corner.



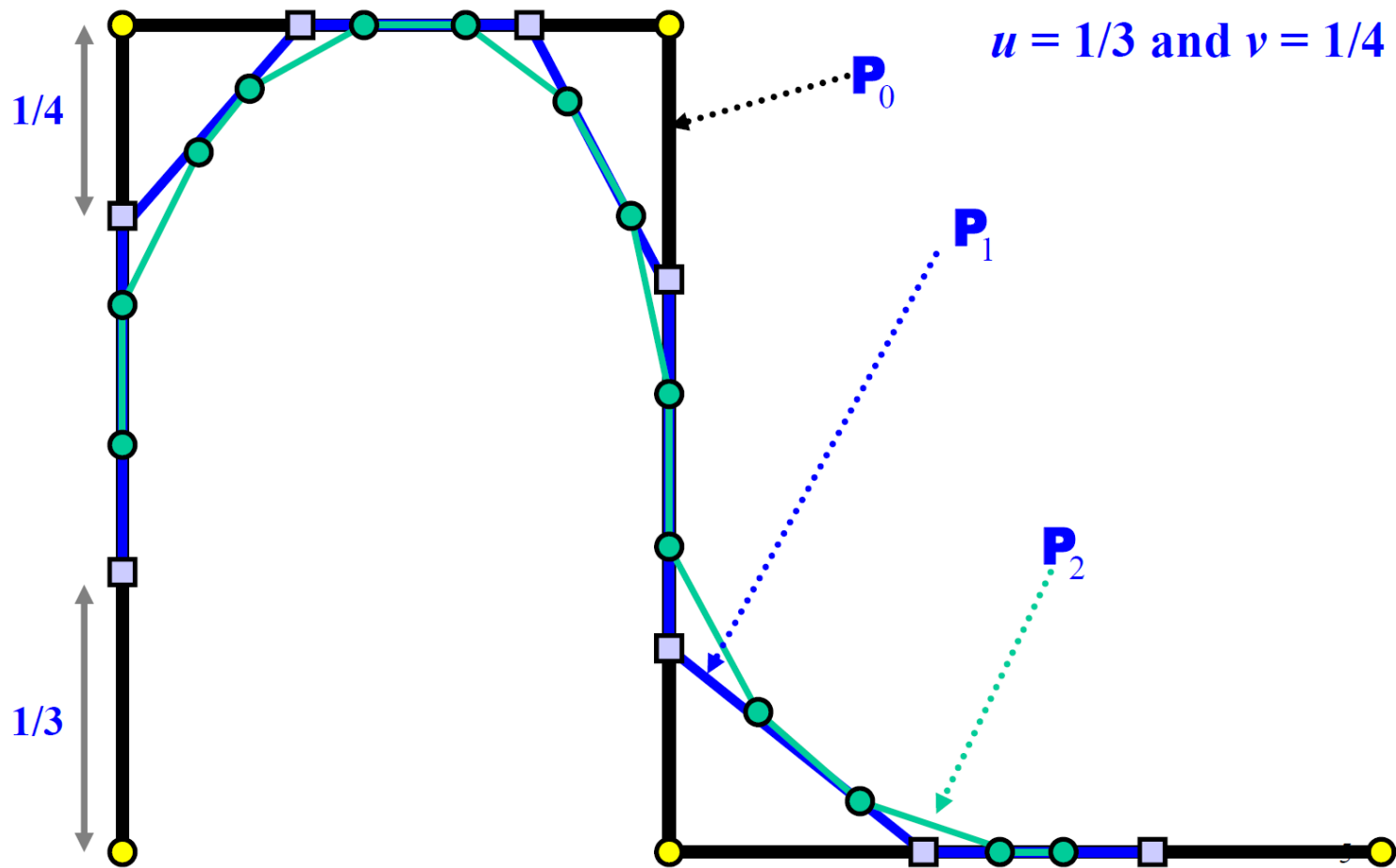
Simple Corner Cutting: 2/5

- Suppose we have a polyline P_0 . Divide its edges with the above scheme, yielding a new polyline P_1 .
- Dividing P_1 yields P_2 , ..., and so on. What is

$$\mathbf{P}_\infty = \lim_{i \rightarrow \infty} \mathbf{P}_i$$

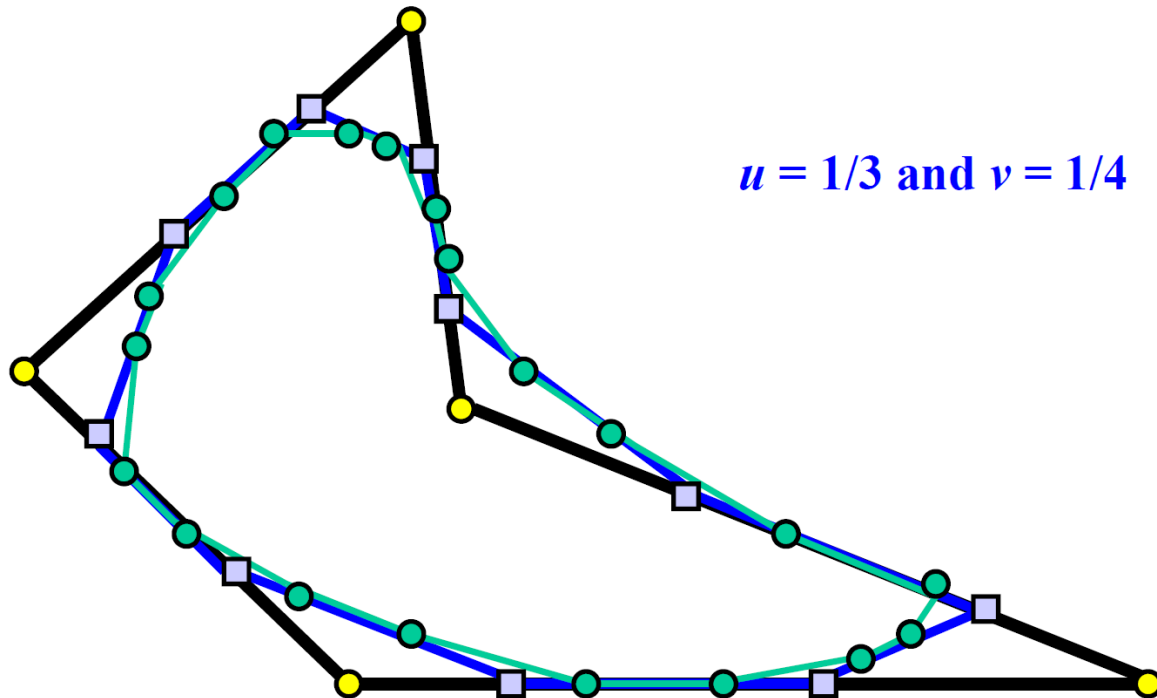
- The u 's and v 's do not have to be the same for edge. Moreover, the u 's and v 's used to divide P_i do not have to be equal to those u 's and v 's used to divide P_{i+1} .

Simple Corner Cutting: 3/5

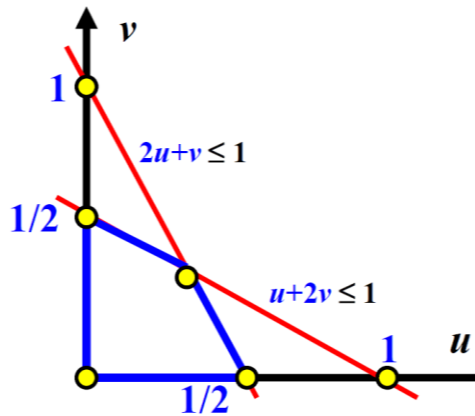


Simple Corner Cutting: 4/5

- For a polygon, one more leg from the last point to the first must also be divided accordingly.



Simple Corner Cutting: 5/5



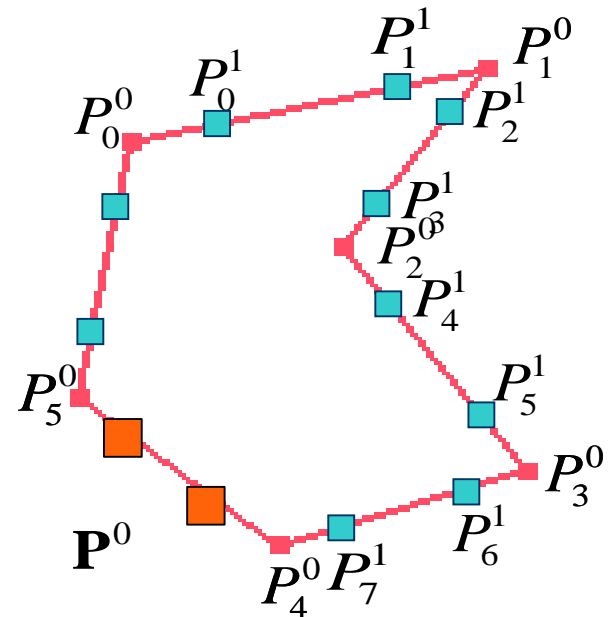
Chaikin used $u = v = 1/4$

- The following result was proved by Gregory and Qu, de Boor, and Paluszny, Prautzsch and Schäfer.
- If all u 's and v 's lies in the interior of the area bounded by $u \geq 0$, $v \geq 0$, $u+2v \leq 1$ and $2u+v \leq 1$, then P_∞ is a C^1 curve.
- This procedure was studied by Chaikin in 1974, and was later proved that the limit curve is a B-spline curve of degree 2.

Exemple: Algorithme de Chaikin

$$P_{2i}^{k+1} = \frac{3}{4}P_i^k + \frac{1}{4}P_{i+1}^k$$

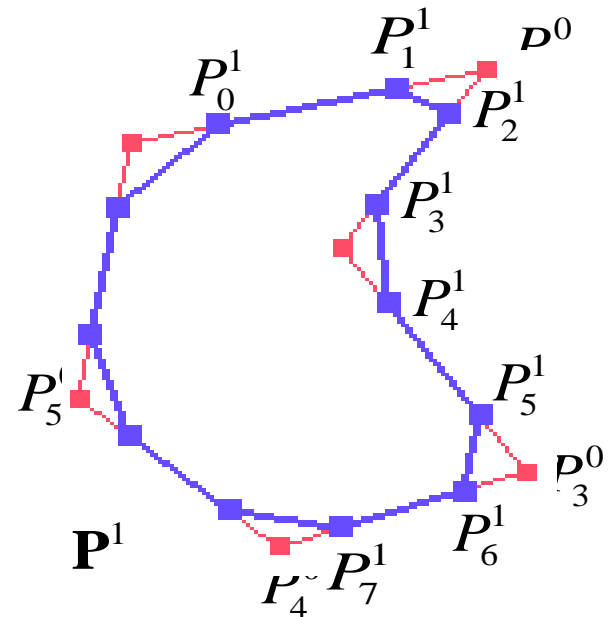
$$P_{2i+1}^{k+1} = \frac{1}{4}P_i^k + \frac{3}{4}P_{i+1}^k$$



Exemple: Algorithme de Chaikin

$$P_{2i}^{k+1} = \frac{3}{4}P_i^k + \frac{1}{4}P_{i+1}^k$$

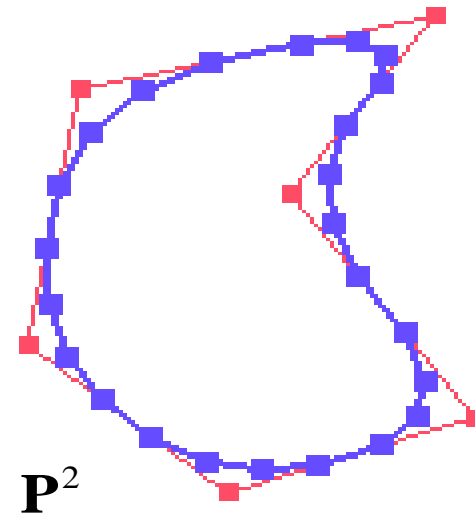
$$P_{2i+1}^{k+1} = \frac{1}{4}P_i^k + \frac{3}{4}P_{i+1}^k$$



Exemple: Algorithme de Chaikin

$$P_{2i}^{k+1} = \frac{3}{4}P_i^k + \frac{1}{4}P_{i+1}^k$$

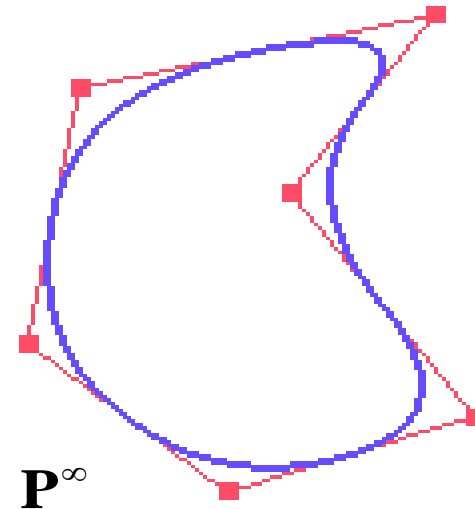
$$P_{2i+1}^{k+1} = \frac{1}{4}P_i^k + \frac{3}{4}P_{i+1}^k$$



Exemple: Algorithme de Chaikin

$$P_{2i}^{k+1} = \frac{3}{4}P_i^k + \frac{1}{4}P_{i+1}^k$$

$$P_{2i+1}^{k+1} = \frac{1}{4}P_i^k + \frac{3}{4}P_{i+1}^k$$



Facts about Subdivision Surfaces

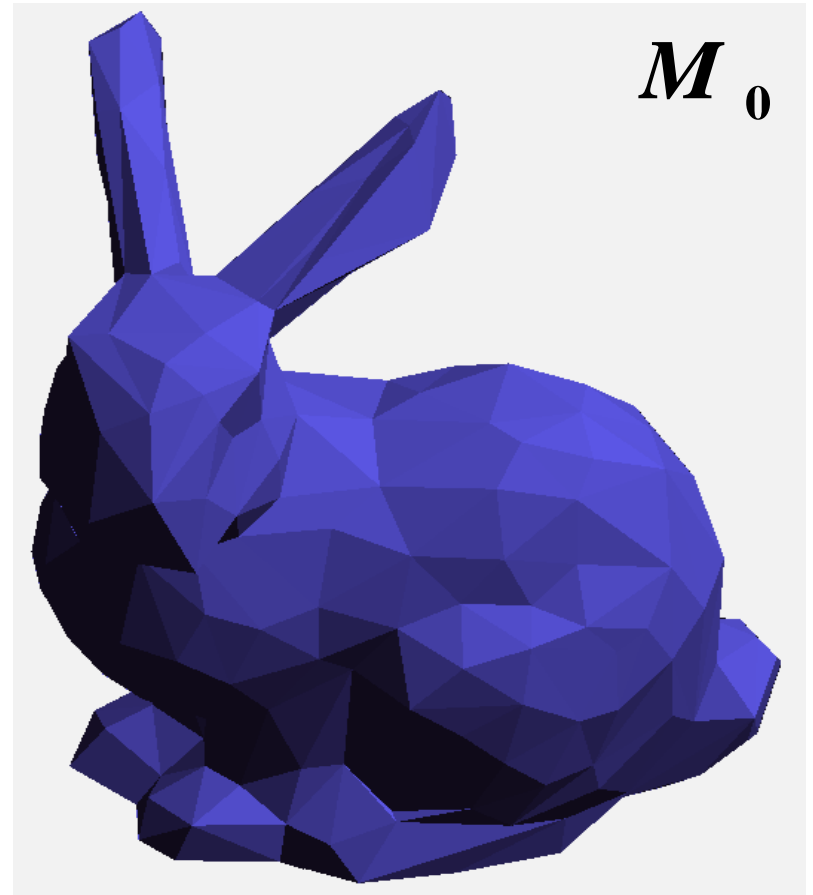
- Subdivision surfaces are limit surfaces:
 - It starts with a mesh
 - It is then refined by repeated subdivision
- Since the subdivision process can be carried out infinite number of times, the intermediate meshes are **approximations** of the actual subdivision surface.
- Subdivision surfaces is a simple technique for describing complex surfaces of arbitrary topology with guaranteed continuity.
- Also supports Multiresolution.

What Can You Expect from ...?

- It is easy to model a large number of surfaces of various types.
- Usually, it generates smooth surfaces.
- It has simple and intuitive interaction with models.
- It can model sharp and semi-sharp features of surfaces.
- Its representation is simple and compact (e.g., winged-edge and half-edge data structures, etc).
- We only discuss 2-manifolds without boundary.

Exemple : Surface de subdivision

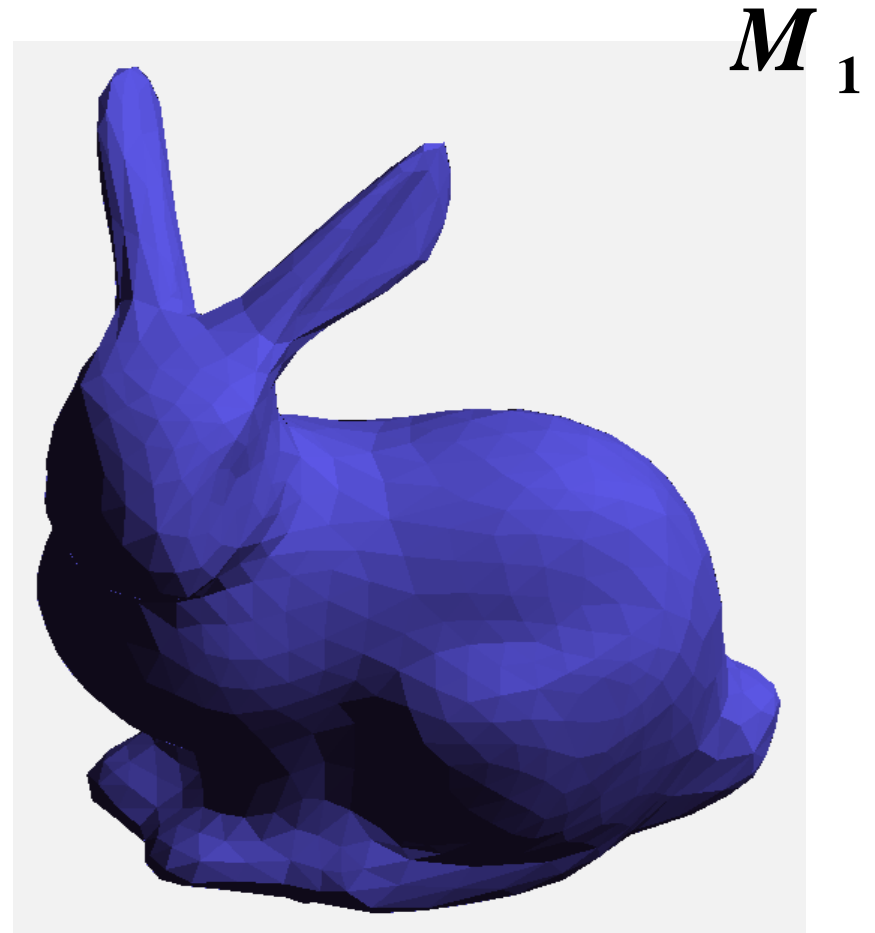
- Maillage initial
- Règles de subdivision



Exemple : Surface de subdivision

- Maillage initial
- Règles de subdivision

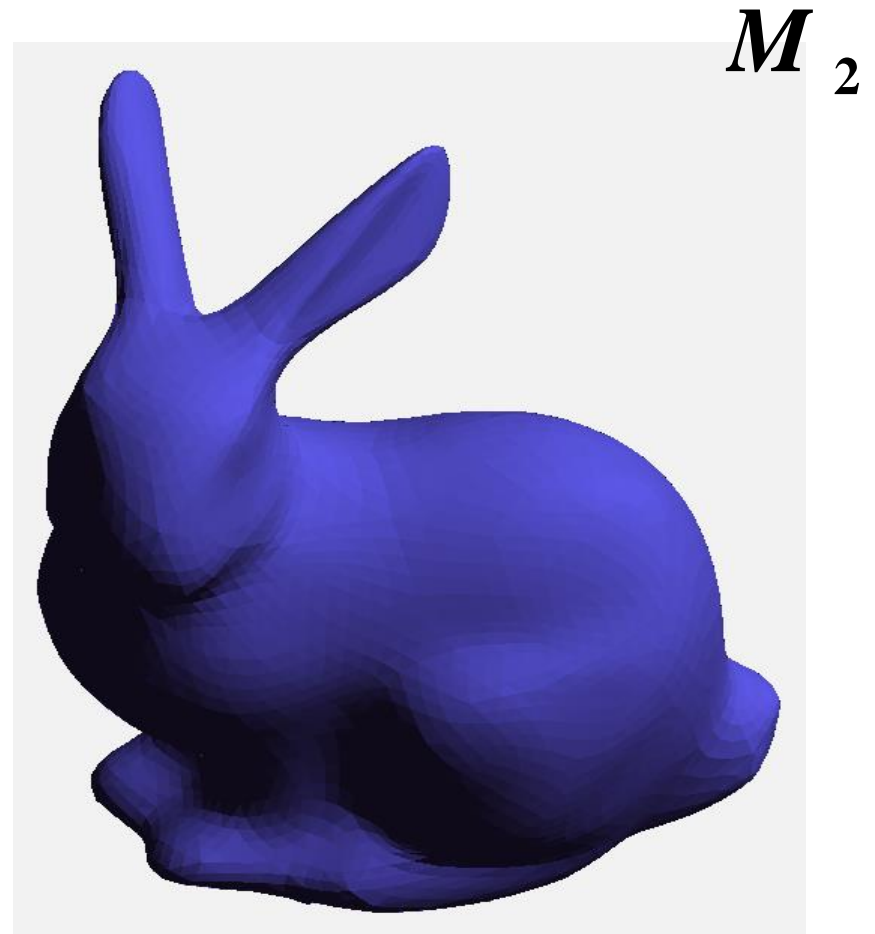
$$M_1 = S \cdot M_0$$



Exemple : Surface de subdivision

- Maillage initial
- Règles de subdivision

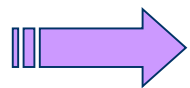
$$\begin{aligned} M_2 &= S \cdot M_1 \\ &= S_2 \cdot M_0 \end{aligned}$$



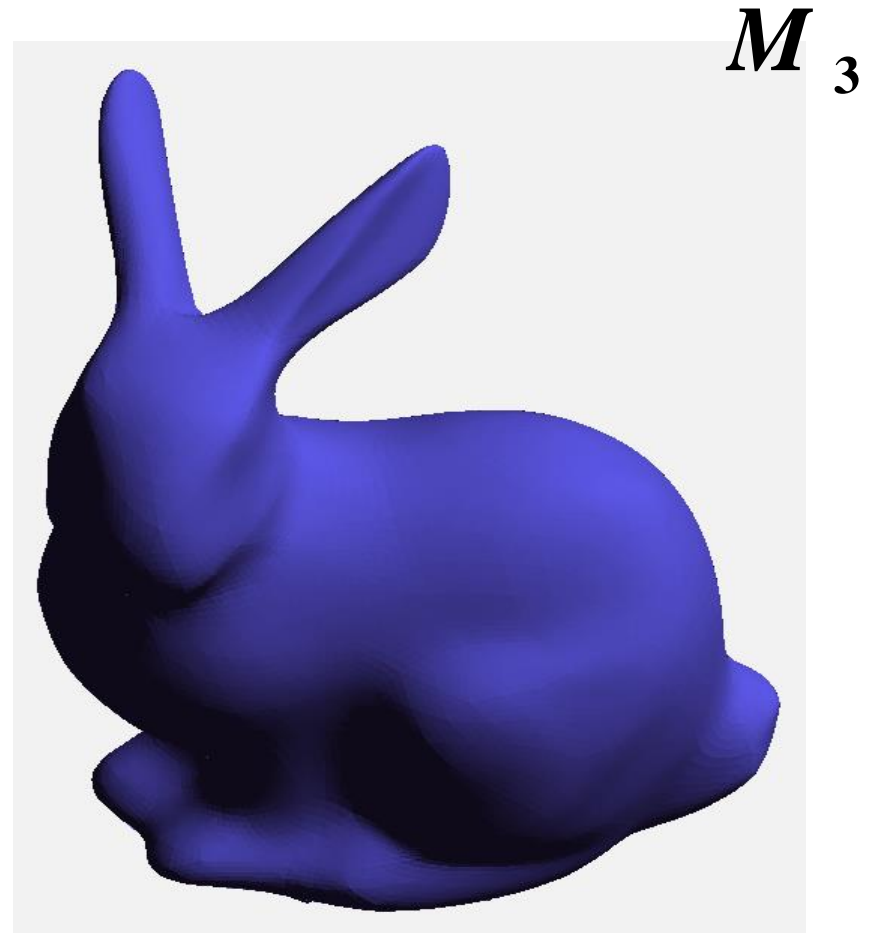
Exemple : Surface de subdivision

- Maillage initial
- Règles de subdivision

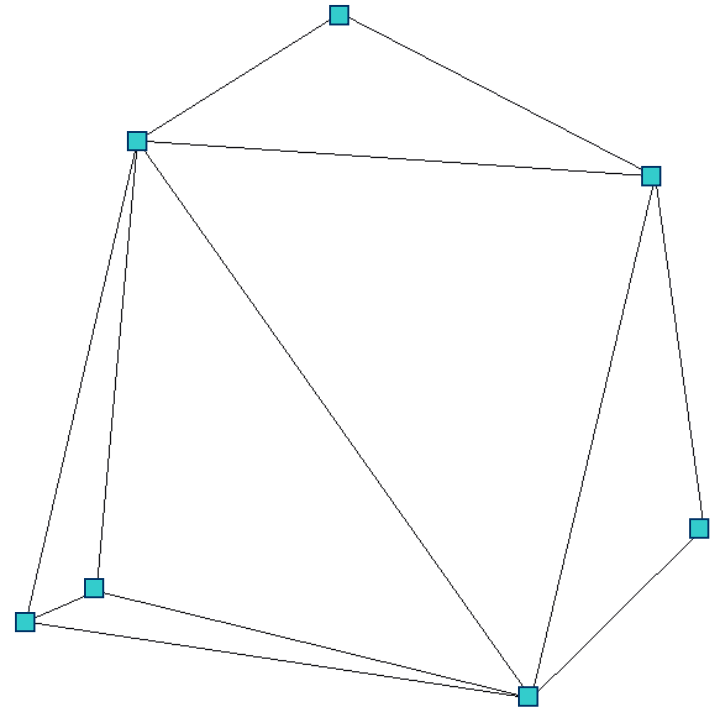
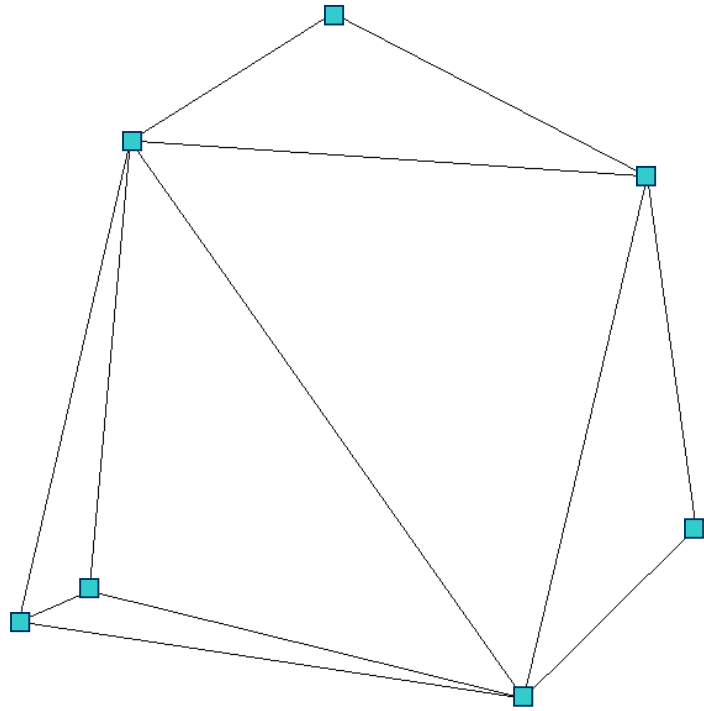
$$M_{k+1} = S \cdot M_k$$



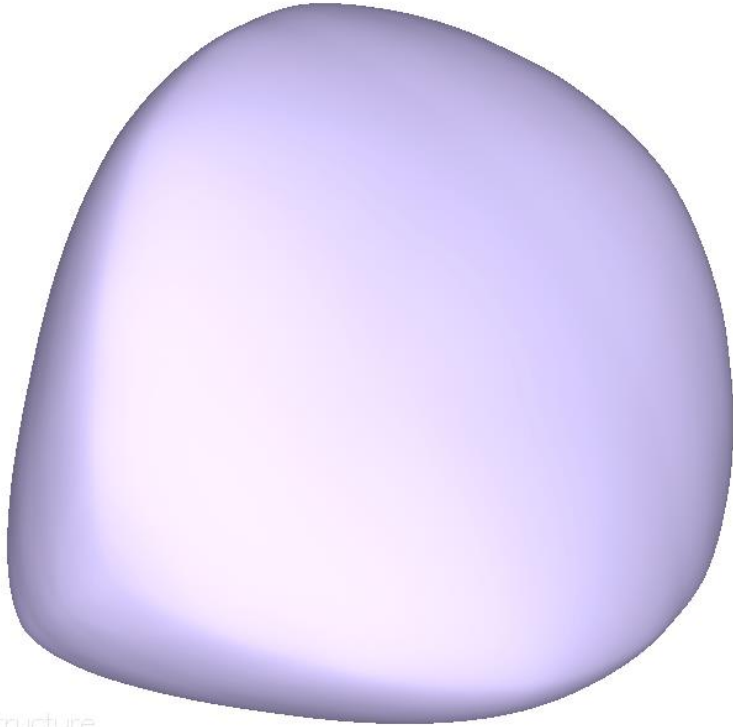
Surface lisse



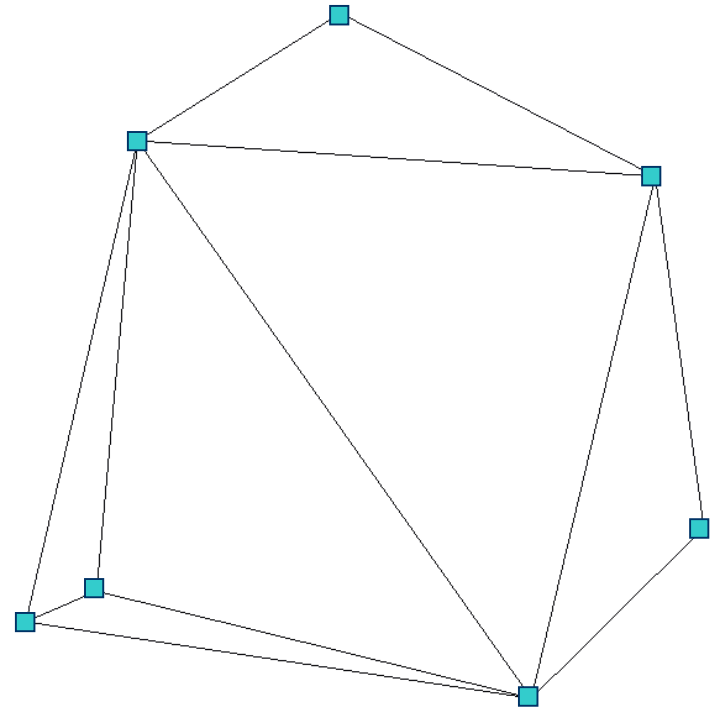
Interpolation / Approximation



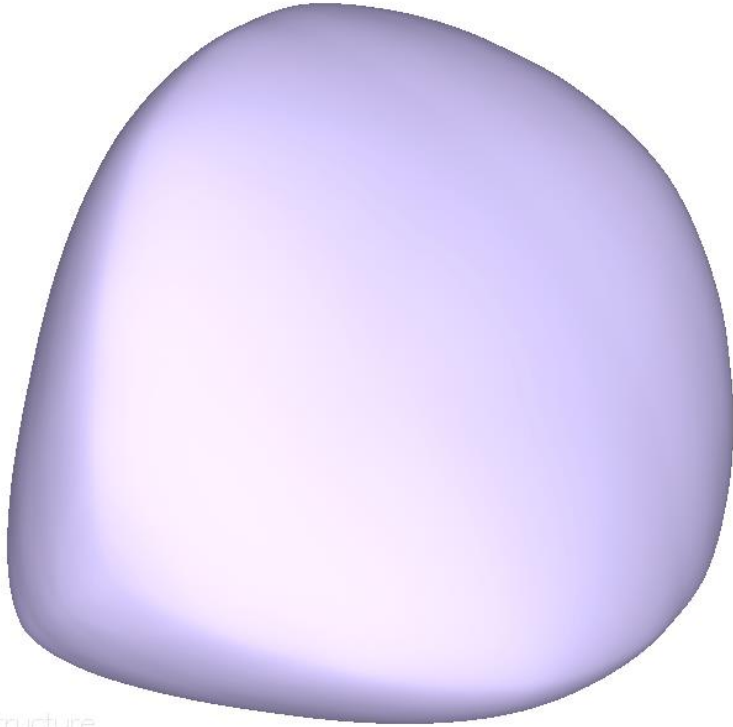
Interpolation / Approximation



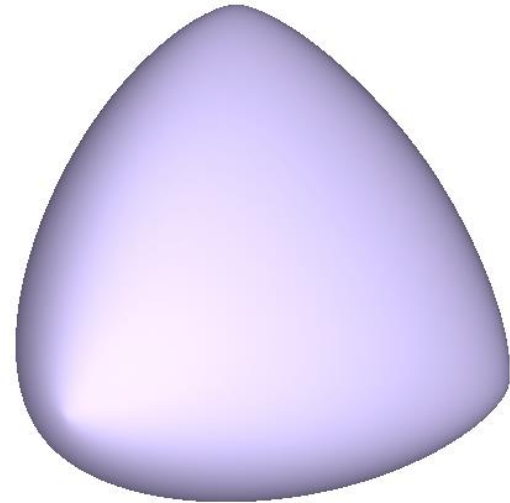
structure



Interpolation / Approximation



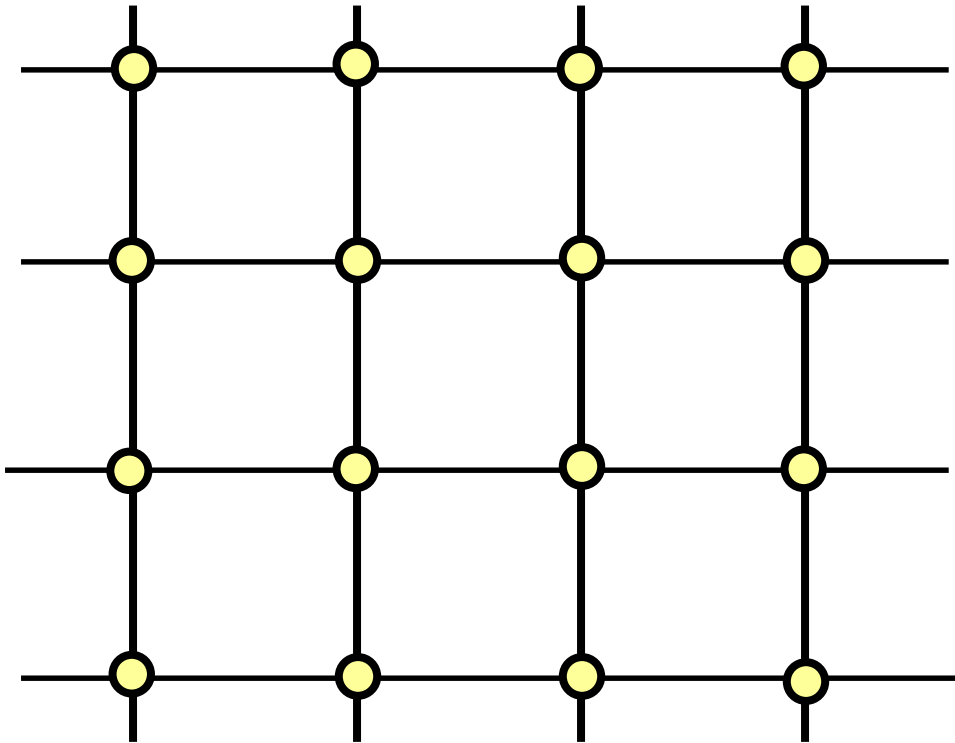
structure



structure

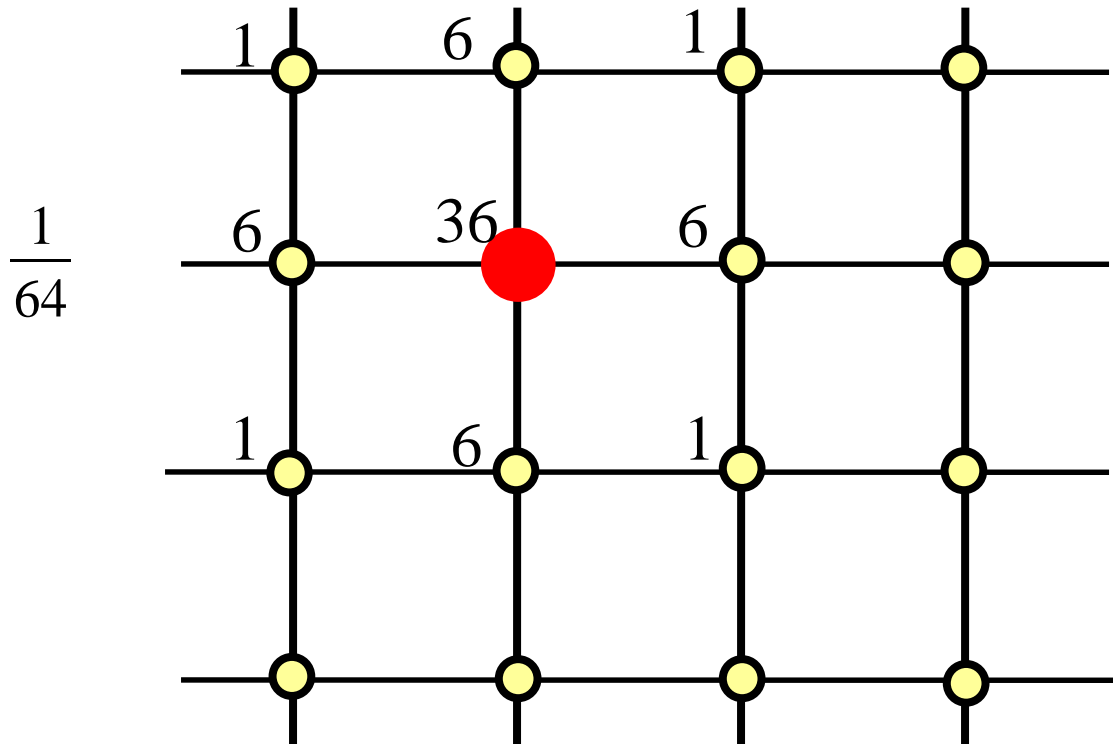
Principe de subdivision

- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque



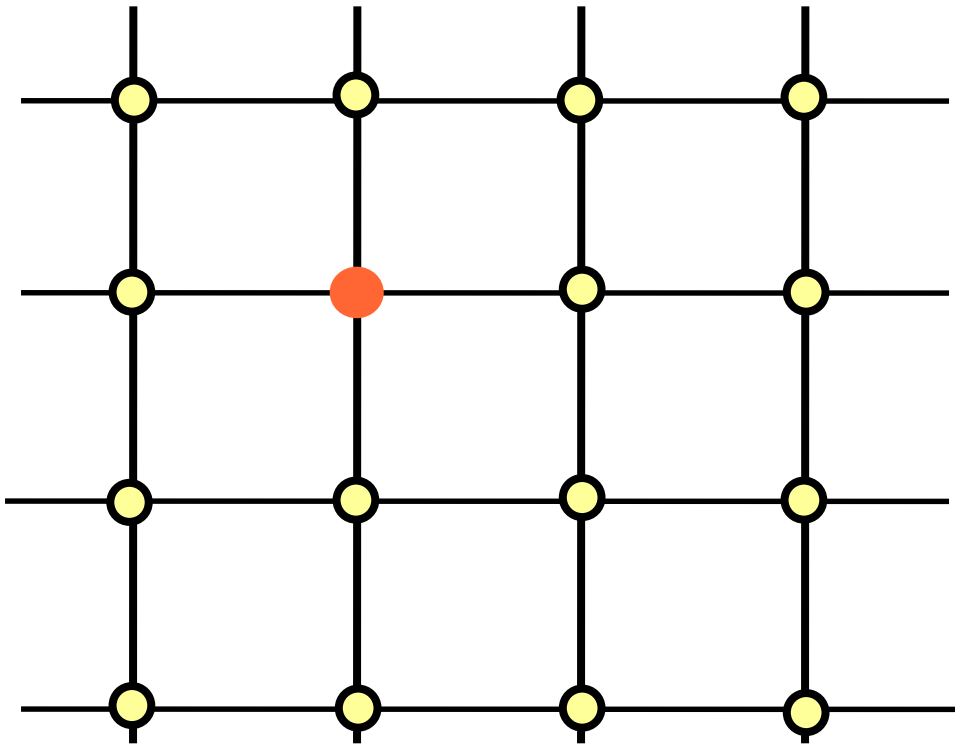
Principe de subdivision

- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque
 - Les sommets sont **déplacés** par une combinaison affine des sommets du maillage



Principe de subdivision

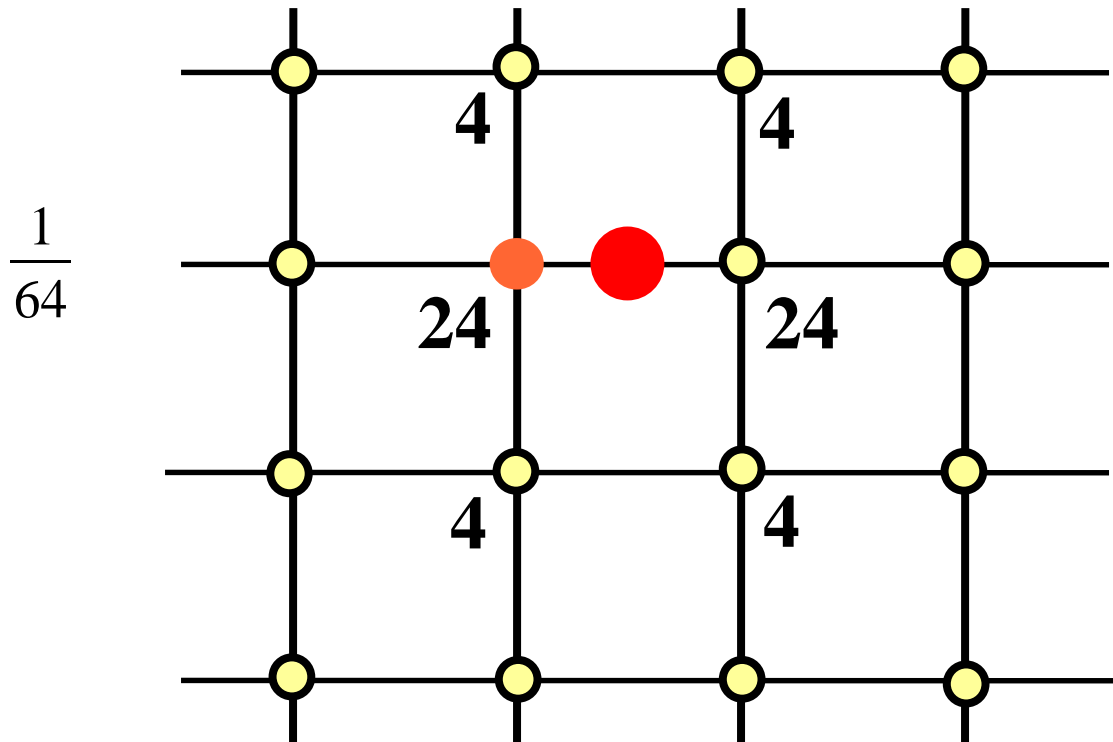
- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque



Principe de subdivision

- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque

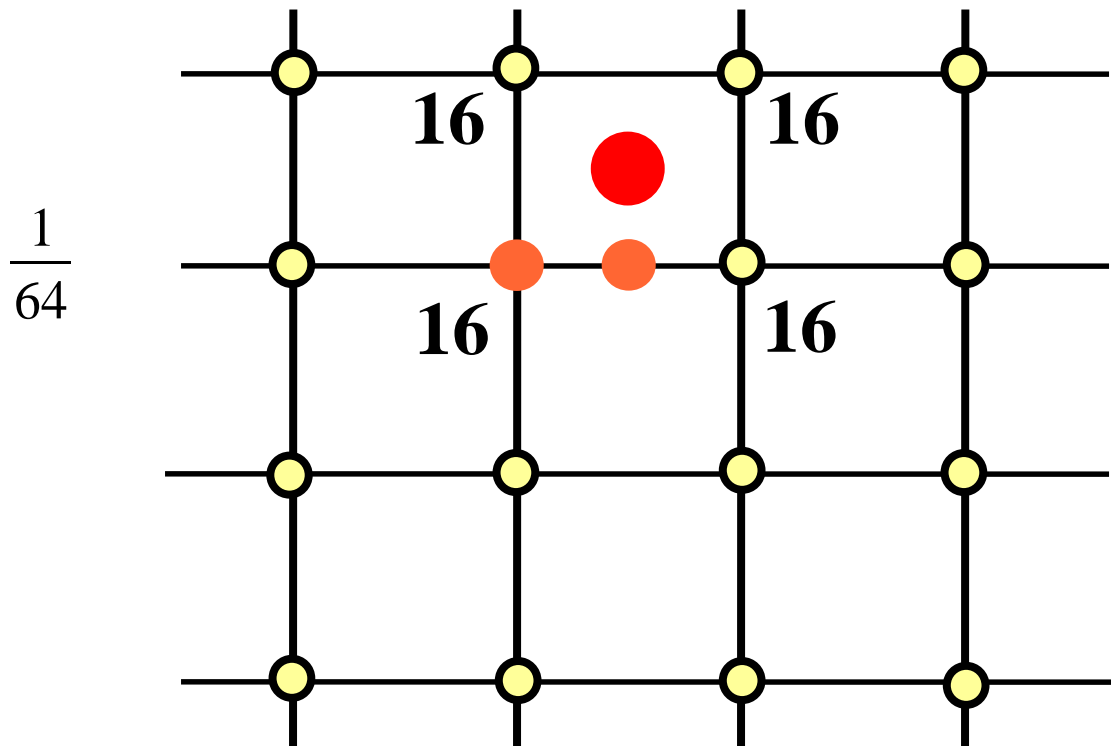
Des **nouveaux** sommets sont **ajoutés** par une combinaison affine des sommets du maillage



Principe de subdivision

- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque

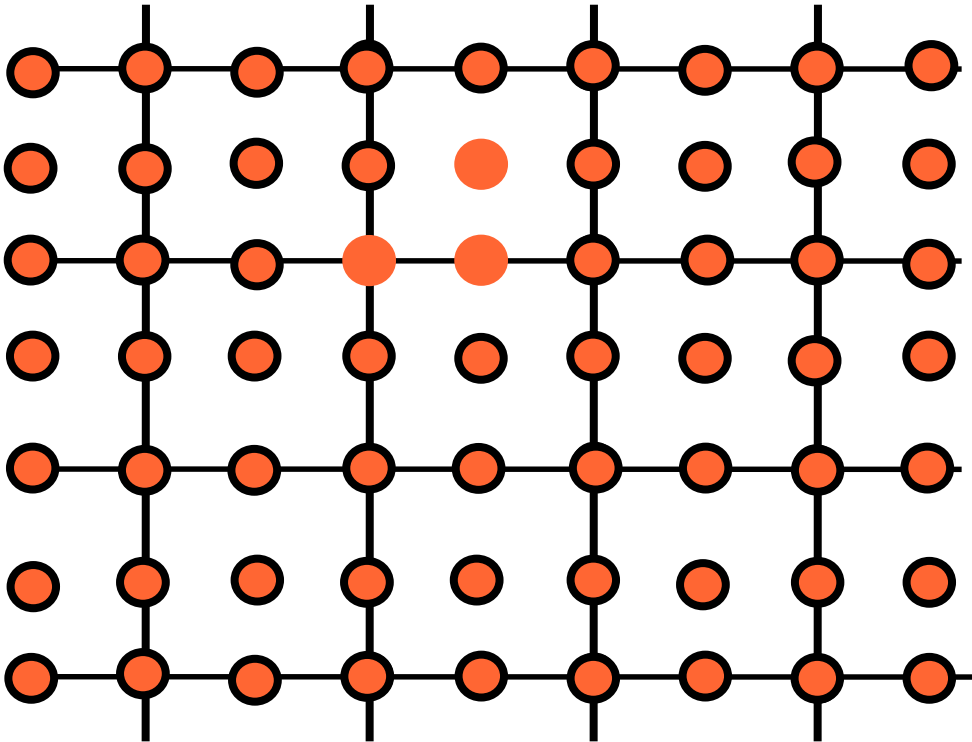
Des **nouveaux** sommets sont **ajoutés** par une combinaison affine des sommets du maillage



Principe de subdivision

- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque

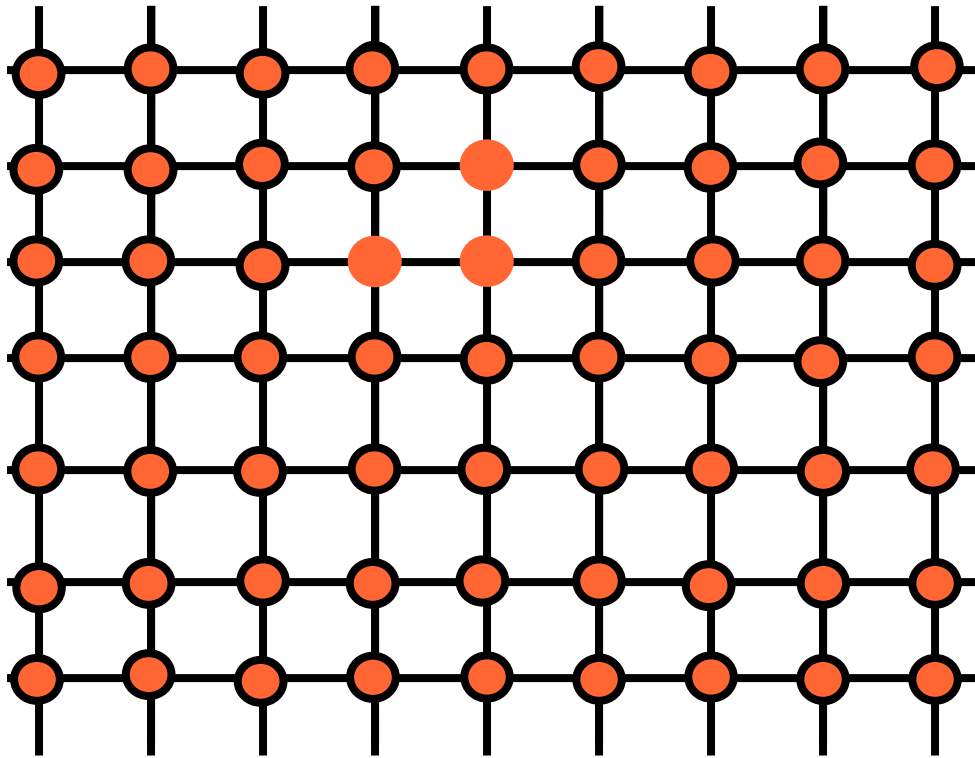
Le processus de déplacement / insertion est répété sur l'ensemble du polygone initial



Principe de subdivision

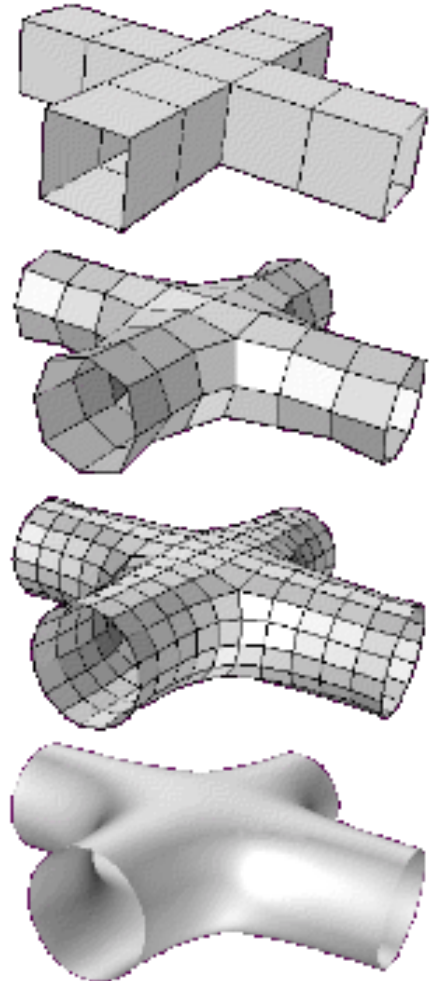
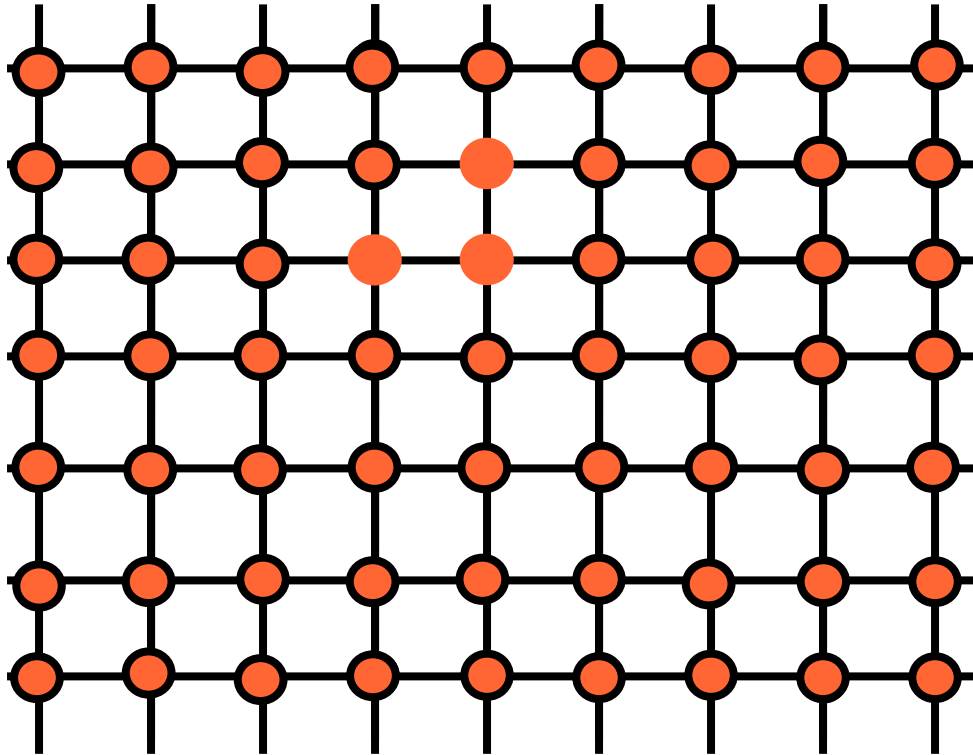
- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque

Les nouveaux points sont maillés



Principe de subdivision

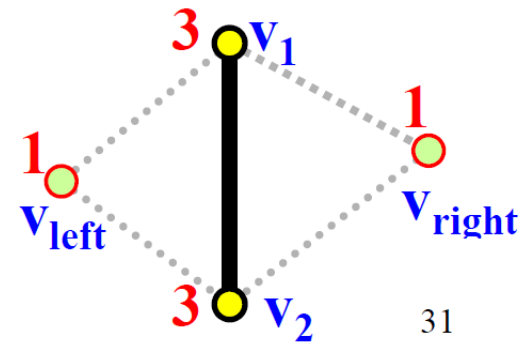
- Généralisation des surfaces paramétriques à des polygones de contrôle quelconque



Loop's Algorithm: 1/5

- Loop's (i.e., Charles Loop's) algorithm only works for triangle meshes.
- Loop's algorithm computes a new edge point for each edge and a new vertex for each vertex.
- Let v_1v_2 be an edge and the other vertices of the incident triangles be v_{left} and v_{right} . The new edge point e is computed as follows.

$$\mathbf{e} = \frac{3}{8}(\mathbf{v}_1 + \mathbf{v}_2) + \frac{1}{8}(\mathbf{v}_{\text{left}} + \mathbf{v}_{\text{right}})$$



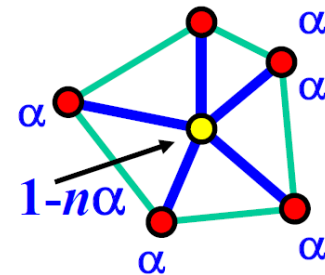
Loop's Algorithm: 2/5

□ For each vertex \mathbf{v} , its new vertex point \mathbf{v}' is computed below, where $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ are adjacent vertices

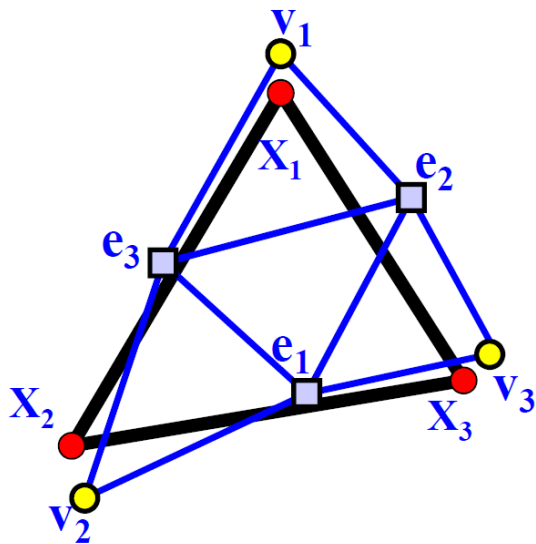
$$\mathbf{v}' = (1 - n\alpha)\mathbf{v} + \alpha \sum_{j=1}^n \mathbf{v}_j$$

where α is

$$\alpha = \begin{cases} \frac{3}{16} & n=3 \\ \frac{1}{n} \left[\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right] & n>3 \end{cases}$$



Loop's Algorithm: 3/5



- Let a triangle be defined by X_1 , X_2 and X_3 and the corresponding new vertex points be v_1 , v_2 and v_3 .
- Let the edge points of edges v_1v_2 , v_2v_3 and v_3v_1 be e_3 , e_1 and e_2 . The new triangles are $v_1e_2e_3$, $v_2e_3e_1$, $v_3e_1e_2$ and $e_1e_2e_3$. This is a 1-to-4 scheme.
- This algorithm was developed by Charles Loop in 1987.

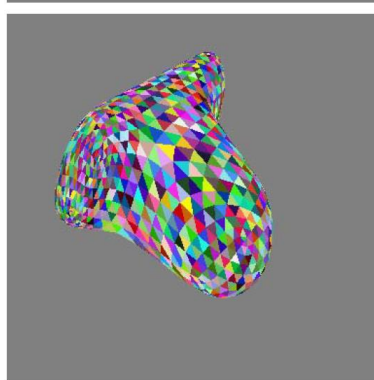
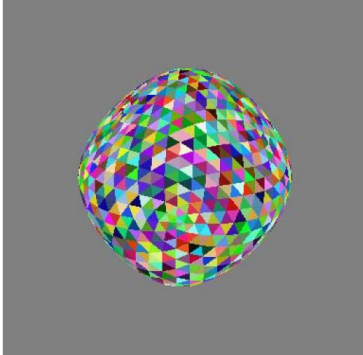
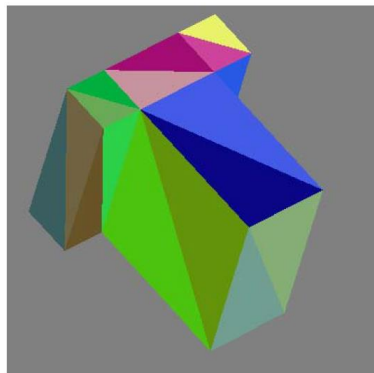
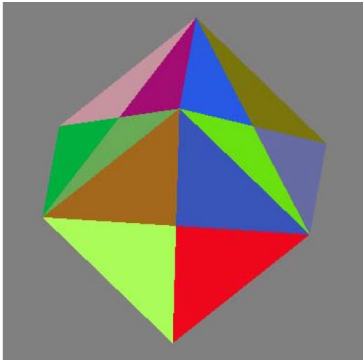
Loop's Algorithm: 4/5

- Pick a vertex in the original or an intermediate mesh. If this vertex has n adjacent vertices $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$, it converges to \mathbf{v}_∞ :

$$\mathbf{v}_\infty = \frac{3 + 8(n-1)\alpha}{3 + 8n\alpha} + \frac{8\alpha}{3 + 8n\alpha} \sum_{j=1}^n \mathbf{v}_j$$

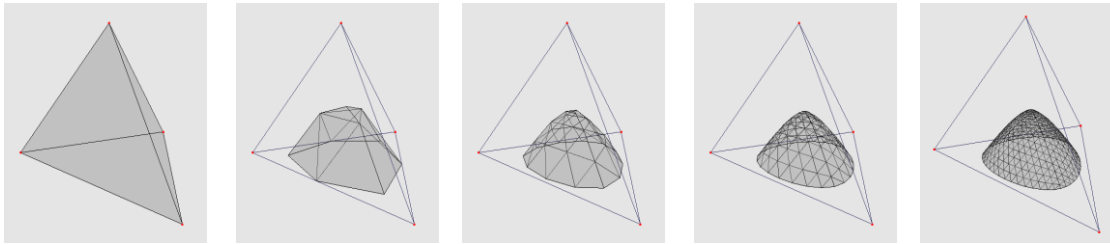
- If all vertices have valency 6, the limit surface is a collection of C^2 Bézier triangles.
- However, only a torus can be formed with all valency 6 vertices. Vertices with different valencies converge to extraordinary vertices where the surface is only G^1 .

Loop's Algorithm 5/5

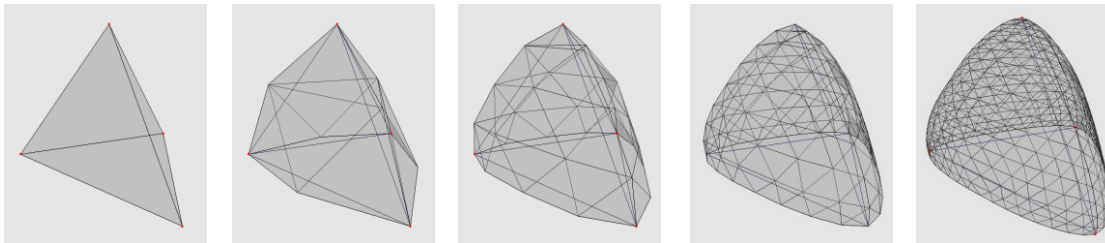


Principe de subdivision

Loop



Butterfly



Subdivision adaptative

- Principe de subdivision adaptive ou non-uniforme :

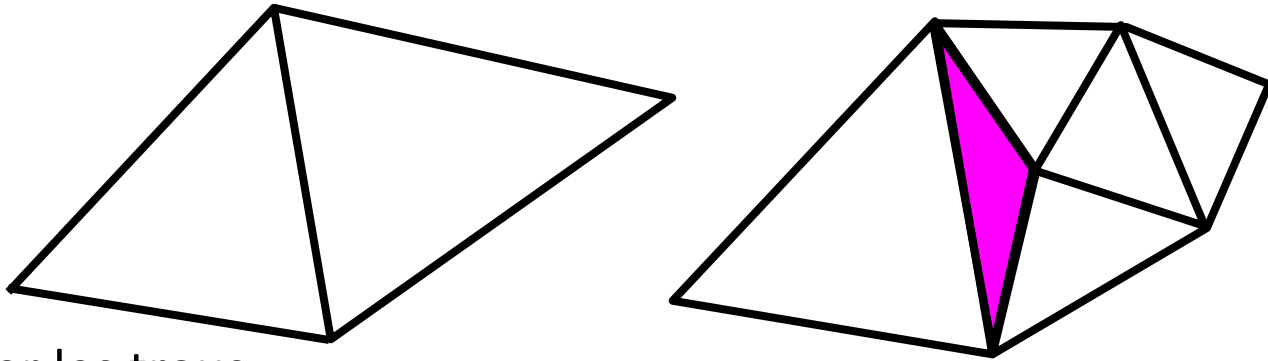
- Où subdiviser ?

Critère de subdivision

- Comment subdiviser ?

Règles de subdivision

Problème de la subdivision adaptative

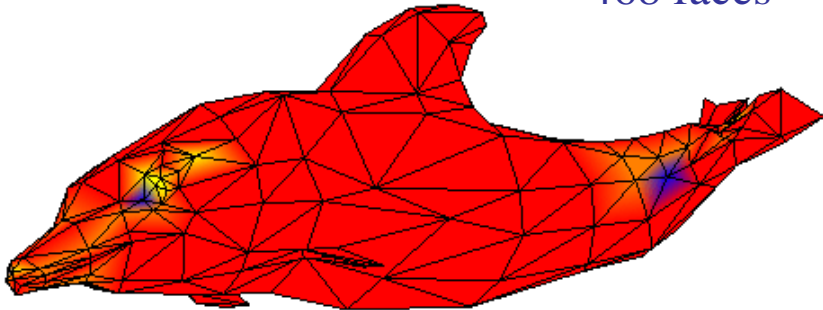


- Éviter les trous
- Générer un "petit" nombre de faces
- Obtenir un maillage progressif

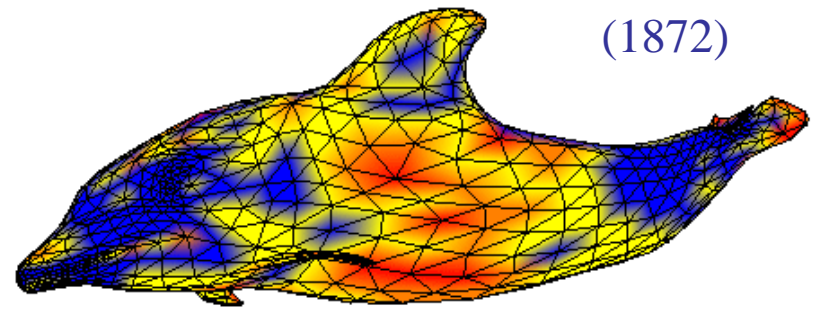
Subdivision Adaptive

- Utiliser des règles pour subdiviser seulement certaines zones en fonction de critères
- Résultats

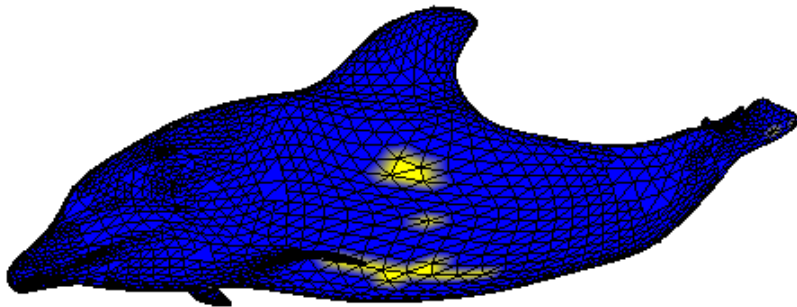
468 faces



1692 faces
(1872)



5022 faces
(7488)



5133 faces
(29952)

