

# Travailler avec les maillages

## ENSIIE2 : Option RVIG

Jean-Yves Didier

`didier@ufrst.univ-evry.fr`

## 1 Maillages (*mesh*) et pavages (*tessellation*)

- Définitions
- Représentation des maillages
- Exemple d'algorithme de tessellation
- Applications de la tessellation

## 2 Génération des surfaces maillées

- Par interpolation : les NURBS
- Par triangulation d'un nuage de points

## 3 Ressources

# Maillages (*mesh*) : définition

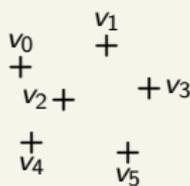
## Définitions

- Maillage** Ensemble de sommets, arêtes et faces qui définissent la forme d'un objet polyédrique en 3D ;
- Sommet** Une position couplée à d'autres informations telles que la couleur, la normale à la surface, et les coordonnées de texture ;
- Arête** Connection entre 2 sommets ;
- Contour** Liste ordonnée d'arêtes liées entre elles 2 à 2 ;
- Face** Ensemble de contours fermés définissant une surface ;

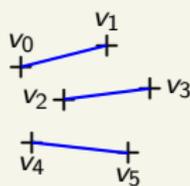


## Représentation des maillages (1/2)

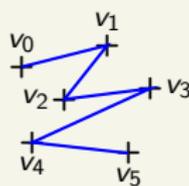
### Modes de représentation en OpenGL



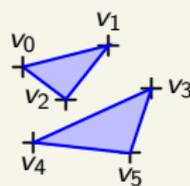
GL\_POINTS



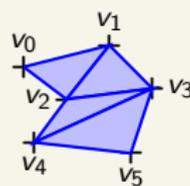
GL\_LINES



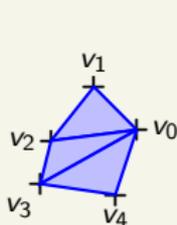
GL\_LINE\_STRIP



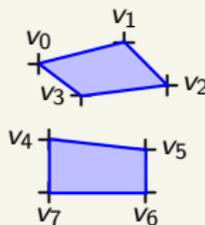
GL\_TRIANGLES



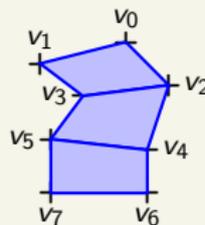
GL\_TRIANGLE\_STRIP



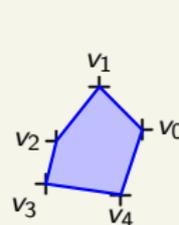
GL\_TRIANGLE\_FAN



GL\_QUADS



GL\_QUADS



GL\_POLYGON

## Représentation des maillages (2/2)

### Modes d'alimentation des informations

- Mode direct : les coordonnées des sommets sont spécifiées une à une, ainsi que chaque information annexe (couleur, normale, ...) attachée à ces derniers ;
- Mode indexé : les données sont préparées dans des tableaux auxquels on fait référence lors de la construction du maillage.

### Exemple d'indexation dans Three.js

```
var geometry = new THREE.Geometry();  
geometry.vertices.push(new THREE.Vector3(-10, 10, 0));  
geometry.vertices.push(new THREE.Vector3(-10, -10, 0));  
geometry.vertices.push(new THREE.Vector3(10, -10, 0));  
geometry.faces.push(new THREE.Face3(0, 1, 2));
```





# Exemple d'algorithme de tessellation

## La tessellation en polygones monotones

### Principales étapes

Etant donné un polygone quelconque auquel est associé un contour :

- 1 Calculer une valeur approchée de la normale au polygone ;
- 2 Projeter le polygone sur un plan n'englobant pas la normale ;
- 3 Partitionner la projection en régions x-monotones en utilisant une ligne de balayage verticale ;
- 4 Trianguler les régions x-monotones ;
- 5 Regrouper les triangles en rubans (*strip*) ou éventails (*fan*).

# Calcul approché de la normale

## Calcul approché de la normale

- Prendre 3 sommets non colinéaires et largement séparés ;
- Calculer la normale au triangle formé ;
- Vérifier si la normale est cohérente par rapport au contour :
  - ▶ Sinon, inverser la normale !

# Projection des sommets

## Projection des sommets

- Calcul rapide approché :
  - ▶ Sélectionner l'axe des coordonnées dont le produit scalaire avec la normale est le plus grand ;
  - ▶ Projeter le long de cette direction sur un plan qui lui est perpendiculaire.
- Calcul exact :
  - ▶ Projeter le long de la normale sur un plan perpendiculaire à cette dernière.

## Partition en régions x-monotones

### Description

- Algorithme en  $O(n \log n)$  [GJPT78] ;
  - ▶ Décomposition trapézoïdale par balayage d'une ligne verticale ;
  - ▶ Création d'une subdivision x-monotone ;
  - ▶ Triangulation.
- Version de complexité en  $O(n)$  [Chazelle91] existante.

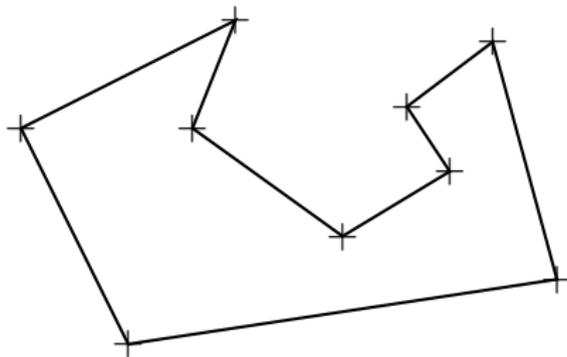
### Polygone L-monotone : définition

Polygone tel que chaque ligne perpendiculaire à L coupe le polygone au plus deux fois.

# Décomposition trapézoïdale

## Le balayage de lignes

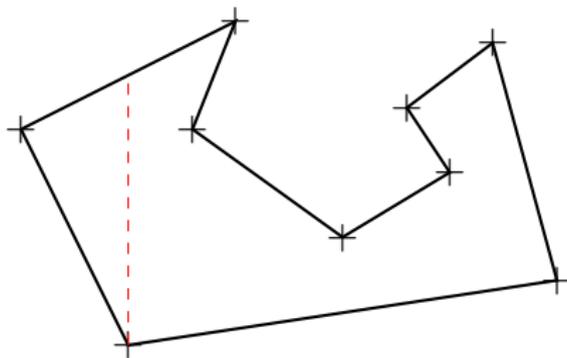
- Une ligne verticale balaye le polygone ;
- Pour chaque sommet, on crée une arête qui coupe le contour ;
- Le polygone est décomposé en trapézoïdes (parfois dégénérés en triangles).



# Décomposition trapézoïdale

## Le balayage de lignes

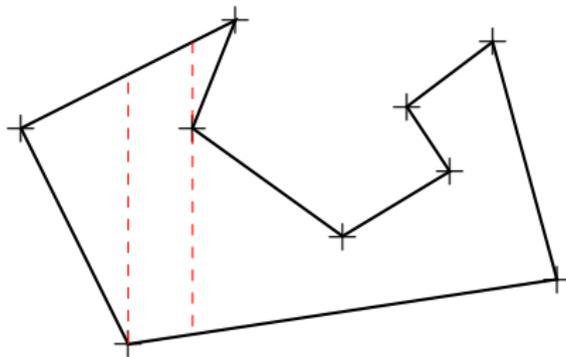
- Une ligne verticale balaye le polygone ;
- Pour chaque sommet, on crée une arête qui coupe le contour ;
- Le polygone est décomposé en trapézoïdes (parfois dégénérés en triangles).



# Décomposition trapézoïdale

## Le balayage de lignes

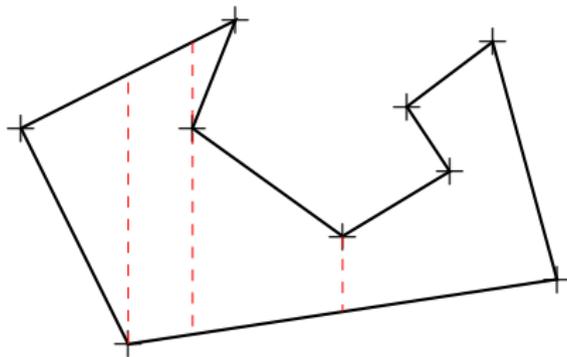
- Une ligne verticale balaye le polygone ;
- Pour chaque sommet, on crée une arête qui coupe le contour ;
- Le polygone est décomposé en trapézoïdes (parfois dégénérés en triangles).



# Décomposition trapézoïdale

## Le balayage de lignes

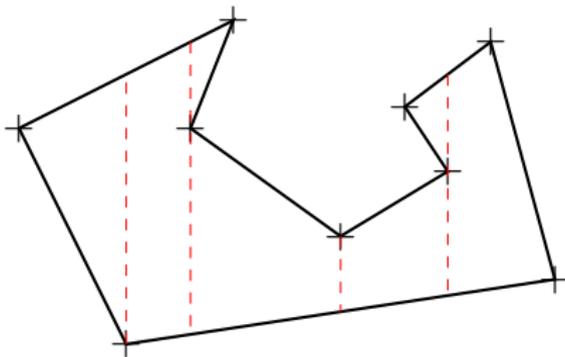
- Une ligne verticale balaye le polygone ;
- Pour chaque sommet, on crée une arête qui coupe le contour ;
- Le polygone est décomposé en trapézoïdes (parfois dégénérés en triangles).



# Décomposition trapézoïdale

## Le balayage de lignes

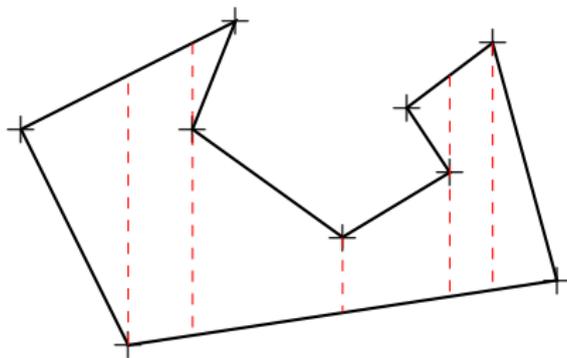
- Une ligne verticale balaye le polygone ;
- Pour chaque sommet, on crée une arête qui coupe le contour ;
- Le polygone est décomposé en trapézoïdes (parfois dégénérés en triangles).



# Décomposition trapézoïdale

## Le balayage de lignes

- Une ligne verticale balaye le polygone ;
- Pour chaque sommet, on crée une arête qui coupe le contour ;
- Le polygone est décomposé en trapézoïdes (parfois dégénérés en triangles).

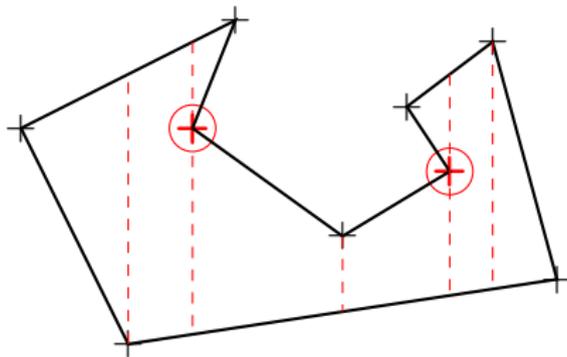


## Raison de la rupture de monotonicité

### Constat

Cause : sommets de trapèzes (non dégénérés) avec les arêtes du même côté :

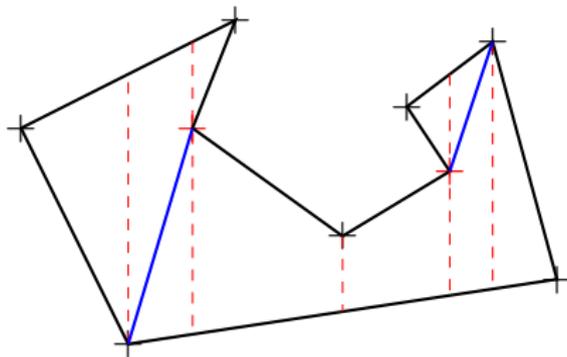
- Arêtes à droite : sommets de partition (*split*) ;
- Arêtes à gauche : sommets de fusion (*merge*) ;



## Création des régions x-monotones

### Par ajout d'arêtes supplémentaires

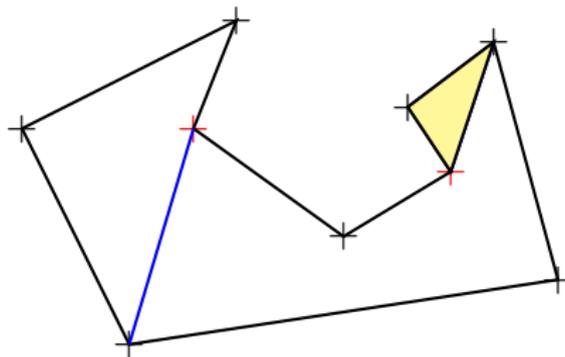
- *split* : diagonale joignant le sommet du trapèze gauche ;
- *merge* : diagonale joignant le sommet du trapèze droit ;



# Triangulation des zones x-monotones

## Description informelle

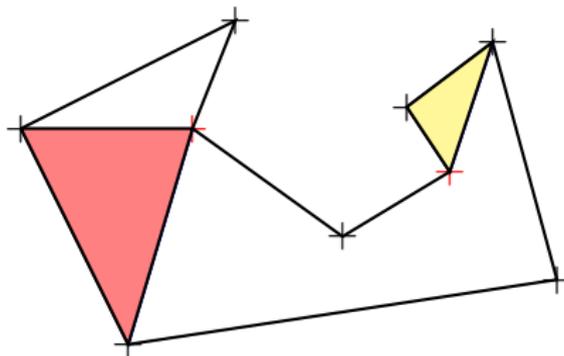
- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.



# Triangulation des zones x-monotones

## Description informelle

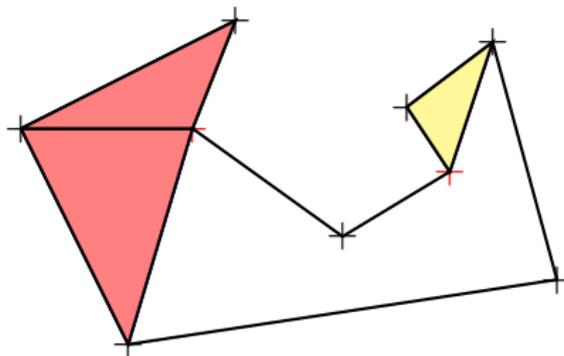
- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.



# Triangulation des zones x-monotones

## Description informelle

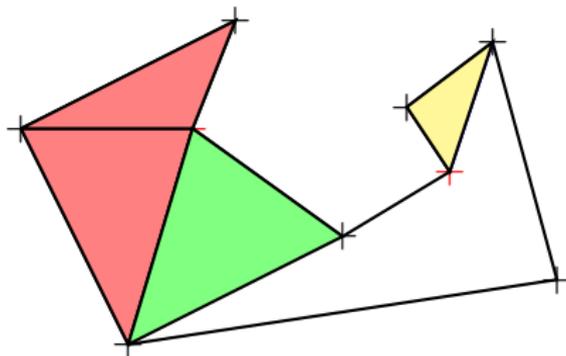
- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.



# Triangulation des zones x-monotones

## Description informelle

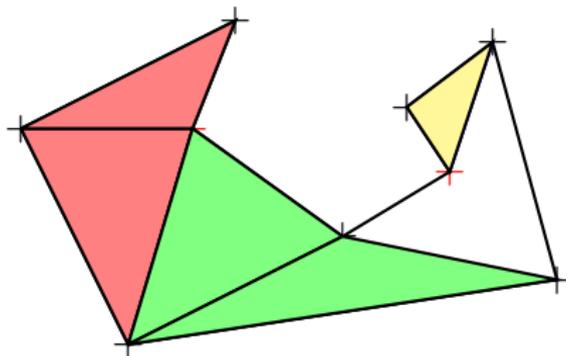
- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.



# Triangulation des zones x-monotones

## Description informelle

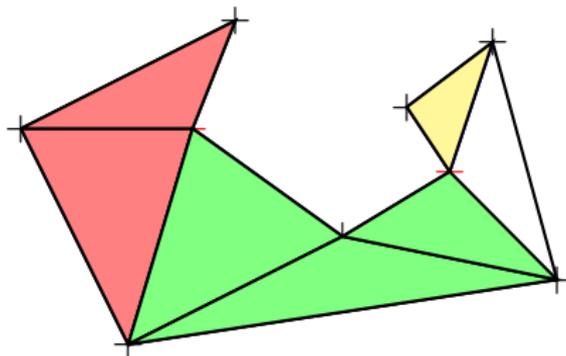
- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.



# Triangulation des zones x-monotones

## Description informelle

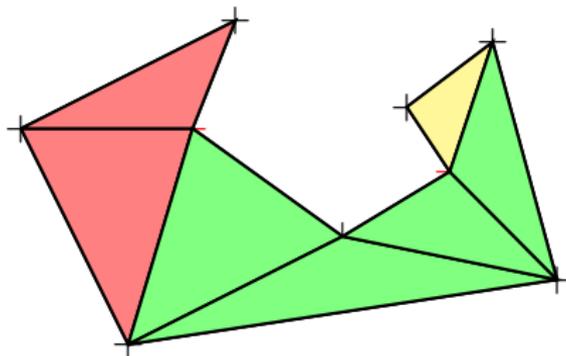
- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.



# Triangulation des zones x-monotones

## Description informelle

- Parcours des sommets de gauche à droite ;
- Constitution d'éventails (*fan*) d'abord, rubans (*strip*) sinon.

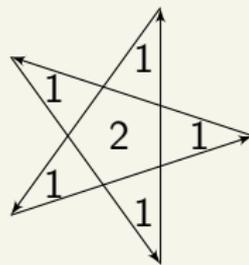
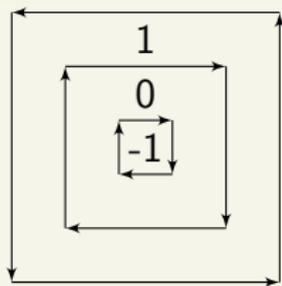
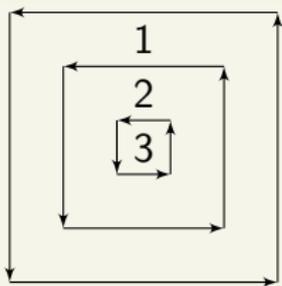


## Applications de la tessellation (1/3)

### Triangulation de polygones quelconques

- Algorithme précédent utilisé par la bibliothèque GLU (*GL Utility*) avec des ajouts :
  - ▶ Contours orientés (*winding*) ;
  - ▶ Zones de l'espace associés à un *winding number* permettant de distinguer les zones à l'intérieur et à l'extérieur du polygone.

### *winding* et *winding number* [redbook]

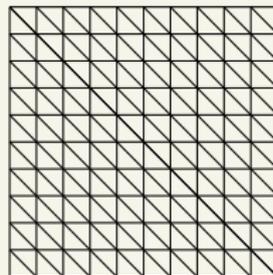
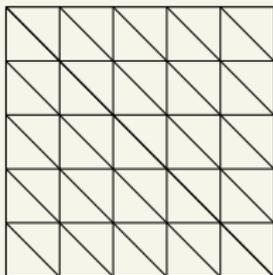
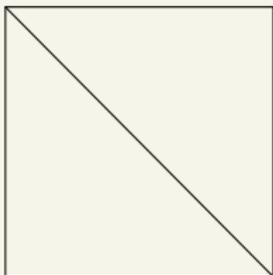


## Applications de la tessellation (2/3)

### Raffinement de maillages

- Pour augmenter le niveau de détail lorsqu'on se rapproche ;
- Souvent combiné à une carte d'élévation (*height map*) ;
- Permet de modifier la géométrie de surface (impression de relief renforcée).

### Exemples de raffinement de maillages



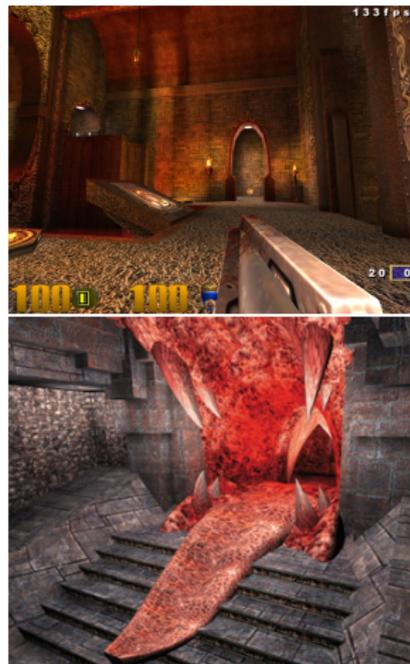


- 1 Maillages (*mesh*) et pavages (*tessellation*)
  - Définitions
  - Représentation des maillages
  - Exemple d'algorithme de tessellation
  - Applications de la tessellation
- 2 Génération des surfaces maillées
  - Par interpolation : les NURBS
  - Par triangulation d'un nuage de points
- 3 Ressources

# NURBS

## NURBS : *Non-uniform rational basis spline*

- Modèle mathématique utilisé pour générer et représenter des courbes surfaciques ;
- Définies par un ordre, un ensemble pondéré de points de contrôles et un vecteur nodal ;
- Généralisation des courbes de Béziérs car fonctionnent en coordonnées homogènes.
- Introduites pour la première fois dans un jeu vidéo en temps réel : *Quake III Arena* (1999).



## NURBS : principe (1/2)

### Combinaison de fonctions de bases

- Mélange de fonctions de bases : fonctions linéaires, quadratiques ou cubiques ;
- Ordre : détermine le degré du polynôme des fonctions de base (degré = ordre - 1) ;
- Point de contrôle : déterminent la forme de la courbe ou de la surface. Chaque point de l'entité est calculé par une somme pondérée des points de contrôle ;
- Vecteur nodal : séquence de paramètres indiquant où et comment les points de contrôle affectent la forme de la NURBS.

## NURBS : principe (2/2)

### Le vecteur nodal dans le détail

- Le nombre de noeuds est égal au nombre de points de contrôle additionné du degré de la courbe additionné à un ;
- Les valeurs du vecteur nodal sont croissantes ;
- Les valeurs consécutives peuvent avoir la même valeur : dans ce cas plusieurs points de contrôle affectent la courbe en même temps ;
- Souvent non représenté car peu intuitif à utiliser lors de l'édition des surfaces.

## Construction des NURBS (1/3)

Basées sur une série de fonctions de base : les B-splines

Notées  $N_{i,n}(u)$  et obtenues par :

$$N_{i,n} = f_{i,n}N_{i,n-1} + g_{i+1,n}N_{i+1,n-1}$$

$i$  :  $i$ ème point de contrôle ;

$n$  : degré de la courbe ;

$u$  : paramètre d'interpolation (souvent sur  $[0, 1]$ ).

Avec les fonctions d'accompagnement  $f$  et  $g$  :

$$f_{i,n}(u) = \frac{u - k_i}{k_{i+n} - k_i} \text{ et } g_{i,n}(u) = \frac{k_{i+n} - u}{k_{i+n} - k_i}$$

Où  $k_i$  désigne la  $i$ ème composante du vecteur nodal.

## Construction des NURBS (2/3)

### Forme générale des courbes

$$C(u) = \sum_{i=1}^k R_{i,n}(u)P_i \text{ et } R_{i,n}(u) = \frac{N_{i,n}(u)w_i}{\sum_{j=1}^k N_{j,n}(u)w_j}$$

$k$  : nombre de point de contrôle ;

$P_i$  : point de contrôle ;

$w_i$  : poids associé au point de contrôle.

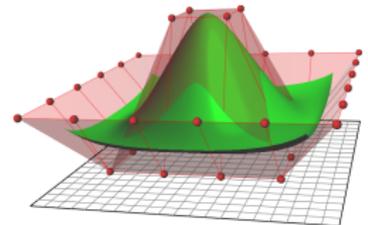
# Construction des NURBS (3/3)

## Forme générale des surfaces

Généralisation des courbes sur 2 dimensions

$$C(u) = \sum_{i=1}^k \sum_{j=1}^l R_{i,j}(u, v) P_{i,j}$$

$$R_{i,j}(u, v) = \frac{N_{i,n}(u) N_{j,m}(v) w_{i,j}}{\sum_{p=1}^k \sum_{q=1}^l N_{p,n}(u) N_{q,m}(u) w_{p,q}}$$



source : wikipedia

## Intérêt à l'utilisation des NURBS

- Donnent un formalisme mathématique commun pour les formes standards et les formes libres ;
- Permettent de définir une large palette de formes ;
- Se calculent rapidement par des algorithmes numériquement stables ;
- Sont invariantes par transformation perspective et projection.

# Triangulation de Delaunay

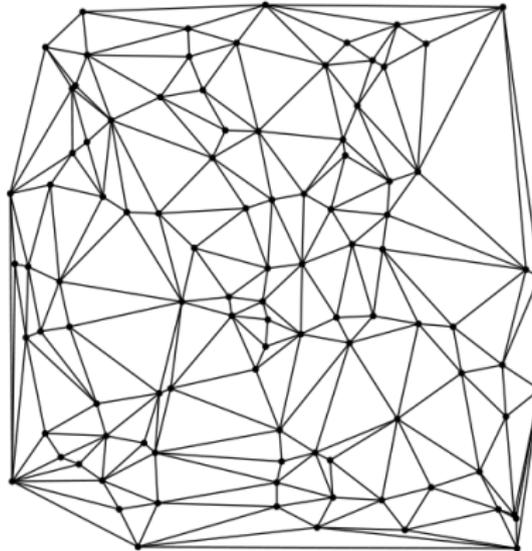
## Présentation et définition

- Créée en 1934 (Boris Delaunay) ;
- Définition : une triangulation de Delaunay d'un ensemble de points  $P$  du plan est une triangulation telle qu'aucun point de  $P$  n'est à l'intérieur du cercle circonscrit d'un triangle de la triangulation ;
- Transcription à une dimension supérieure : aucun point ne doit se trouver dans la sphère circonscrite à un tétraèdre.

## Propriétés

- Problème dual du diagramme de Voronoï ;
- L'union de tous les triangles constitue l'enveloppe convexe des points.

# Exemple de triangulation



Triangulation de 100 points disposés aléatoirement (source : Wikipedia).

# Algorithme de construction (1/3)

## Approche itérative

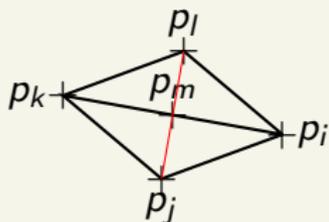
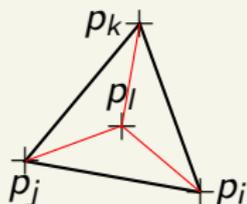
**Entrée** : Ensemble de points dans le plan  $P$

**Sortie** : Triangulation de Delaunay de  $P$

- Initialiser avec un triangle contenant l'ensemble des points à trianguler ;
- Pour chaque point à ajouter dans la triangulation :
  - ▶ Trouver le triangle contenant le nouveau point ;
  - ▶ Si le point est strictement à l'intérieur,
    - Alors ajouter 3 arêtes des sommets vers le point.
    - Sinon, le point est sur une arête commune : couper les triangles en deux.
  - ▶ Régulariser les arêtes nouvellement créées.

## Algorithme de construction (2/3)

### Ajout des arêtes



### Arête illégale : définition

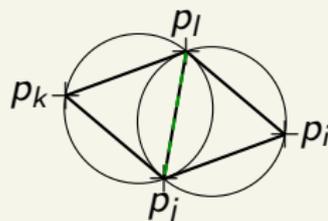
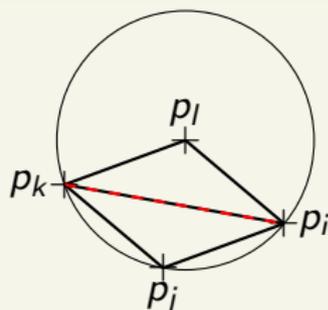
Dans une triangulation de Delaunay, une arête est illégale si le troisième sommet de chacun des triangles de part et d'autre de l'arête est compris dans le cercle circonscrit de l'autre triangle.

### Régularisation des arêtes

Si la nouvelle arête est illégale, alors réaliser une inversion d'arête.

## Algorithme de construction (3/3)

### Régularisation d'arête par l'exemple



- A gauche,  $p_i p_k$  est illégale :  $p_l$  est dans le cercle circonscrit au triangle  $p_i p_j p_k$  ;
- A droite, régularisation :  $p_i p_k$  est basculée, c'est à dire remplacée par  $p_j p_l$ .

## Bibliographie

- [GJPT78] GAREY, Michael R., JOHNSON, David S., PREPARATA, Franco P., et al. *Triangulating a simple polygon*. Information Processing Letters, 1978, vol. 7, no 4, p. 175-179.
- [Chazelle91] CHAZELLE, Bernard. *Triangulating a simple polygon in linear time*. Discrete & Computational Geometry, 1991, vol. 6, no 1, p. 485-524.
- [redbook] SCHREINER, Dave. *Open GL Programming Guide : The Official Guide to Learning OpenGL, Version 4.3*. Addison Wesley, 2013.
- [BKO00] DE BERG, Mark, VAN KREVELD, Marc, OVERMARS, Mark, et al. *Computational geometry*. Springer Berlin Heidelberg, 2000.

# Webographie

- <http://www.glprogramming.com/red/index.html>  
Bible du programmeur OpenGL (redbook).
- <http://geometrie.foretnik.net/files/NURBS-en.swf>  
Démonstration interactive du fonctionnement de NURBS
- <http://en.wikipedia.org/wiki/NURBS>  
Page de wikipedia anglophone sur les NURBS