

Modélisation en synthèse d'images

Cours de master – M1 (Option: I.V.A.)

Université Mohamed Khider Biskra

2019 – 2020

Dr: Zerari Abd El Mouméne

Notions et Techniques de base

- 2D versus 3D
- Points et vecteurs
- Opérations importantes sur les vecteurs
- Droites, segments de droite et modélisation
- Modélisation de polygones

Notions et Techniques de base

2D versus 3D

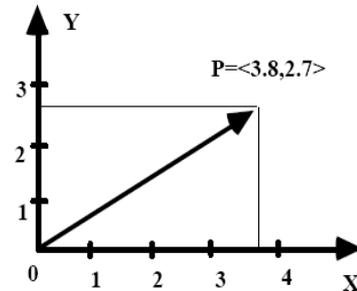
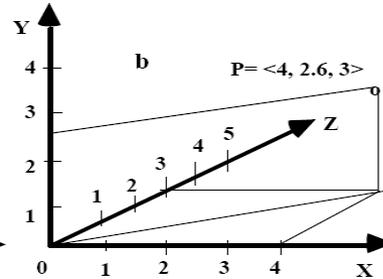
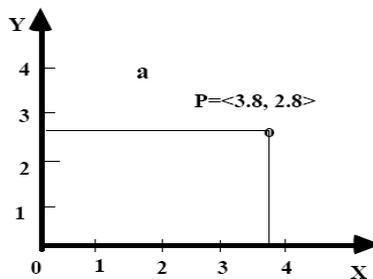
- L'être humain vit dans un monde à trois dimensions, mais lorsqu'il dessine, il utilise généralement des feuilles de papier qui n'ont que deux dimensions.
- Il se trouve donc confronté à un problème de représentation en deux dimensions d'un monde à trois.
- Deux solutions s'offrent alors:
 - ✓ représenter seulement une face plane des objets, par exemple la façade avant d'une maison ou le dessus d'une table
 - ✓ tenter de dessiner la scène choisie en tenant compte de lois de projection telle que la perspective.

Notions et Techniques de base

Points et vecteurs

L'objet graphique le plus simple est évidemment le point caractérisé par ses coordonnées et noté $P = \langle P_x, P_y, P_z \rangle$. Nous utiliserons une notation semblable pour représenter les vecteurs qui jouent un rôle fondamental en informatique graphique.

Un vecteur sera considéré comme la direction donnée par la flèche reliant l'origine du système d'axes au point donné par les composantes du vecteur.



Notions et Techniques de base

Opérations importantes sur les vecteurs

- **norme :**
 - $|\mathcal{V}| = \sqrt{V_x^2 + V_y^2 + V_z^2}$
- **addition :**
 - $V_1 + V_2 = \langle V_{1x} + V_{2x}, V_{1y} + V_{2y}, V_{1z} + V_{2z} \rangle$
- **produit scalaire:**
 - $V_1 \cdot V_2 = \|V_1\| \|V_2\| \cos \alpha$ où α est l'angle entre les 2 vecteurs
 - il faut noter que le résultat est un nombre réel et que dans un système orthonormé, on a :
 - $V_1 \cdot V_2 = V_{1x}V_{2x} + V_{1y}V_{2y} + V_{1z}V_{2z}$
 - ce qui permet de déduire l'angle entre les vecteurs.
- **produit vectoriel:**
 - $V_1 \times V_2 = \langle V_{1y}V_{2z} - V_{1z}V_{2y}, V_{1z}V_{2x} - V_{1x}V_{2z}, V_{1x}V_{2y} - V_{1y}V_{2x} \rangle$
- L'intérêt principal du produit vectoriel est qu'il fournit un vecteur perpendiculaire au plan des deux vecteurs intervenant dans le produit. On peut encore noter que
 - $\|V_1 \times V_2\| = \|V_1\| \|V_2\| \sin \alpha$ où α est l'angle entre les 2 vecteurs V_1 et V_2 .

Notions et Techniques de base

Droites, segments de droite et modélisation

- La droite est une figure très courante bien que l'on utilise plutôt le segment de droite.
- La différence est simple, une droite passe par deux points, tandis qu'un segment de droite est limité par 2 points.
- Pour tracer un segment de droite AB, nous utiliserons deux instructions:
 - ✓ une pour se positionner au point A et
 - ✓ une pour tracer le segment de A à B

Notions et Techniques de base

Modélisation de polygones

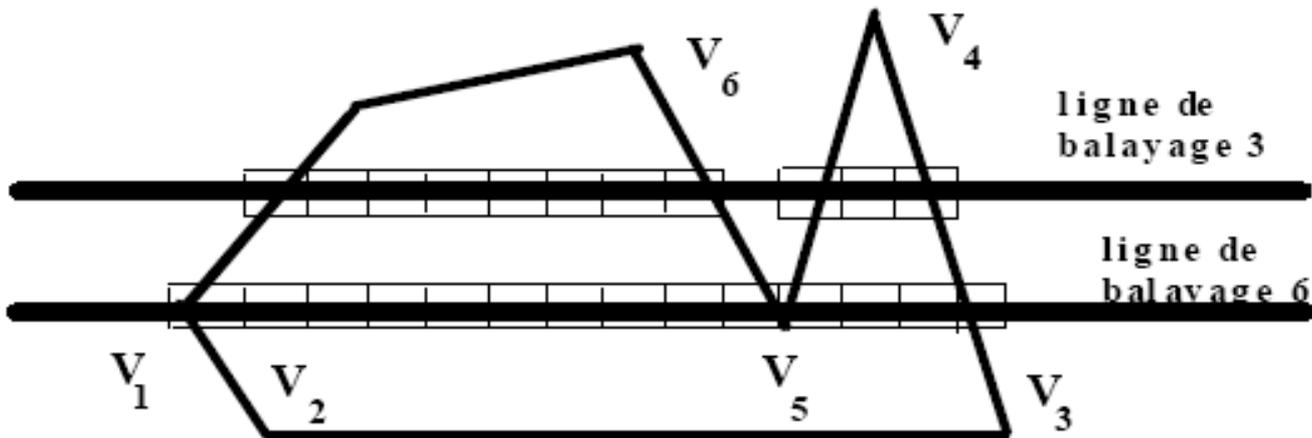
Le concept de polygone

- ❑ Qu'on travaille en deux dimensions ou en trois, le polygone joue un rôle extrêmement important.
- ❑ Nous entendons par polygone une figure plane définie par une liste de points (les sommets) reliés par des segments de droite (les arêtes).
- ❑ Les sommets sont supposés tous différents, les arêtes ne doivent pas se croiser et une arête relie le dernier sommet au premier.
- ❑ Un polygone est concave s'il existe au moins un angle interne supérieur à 180° ;
- ❑ Il est convexe, s'il n'est pas concave.
- ❑ Les polygones sont plus utilisés à deux dimensions, car en les remplissant de couleurs, on construit rapidement des images attrayantes.

Notions et Techniques de base

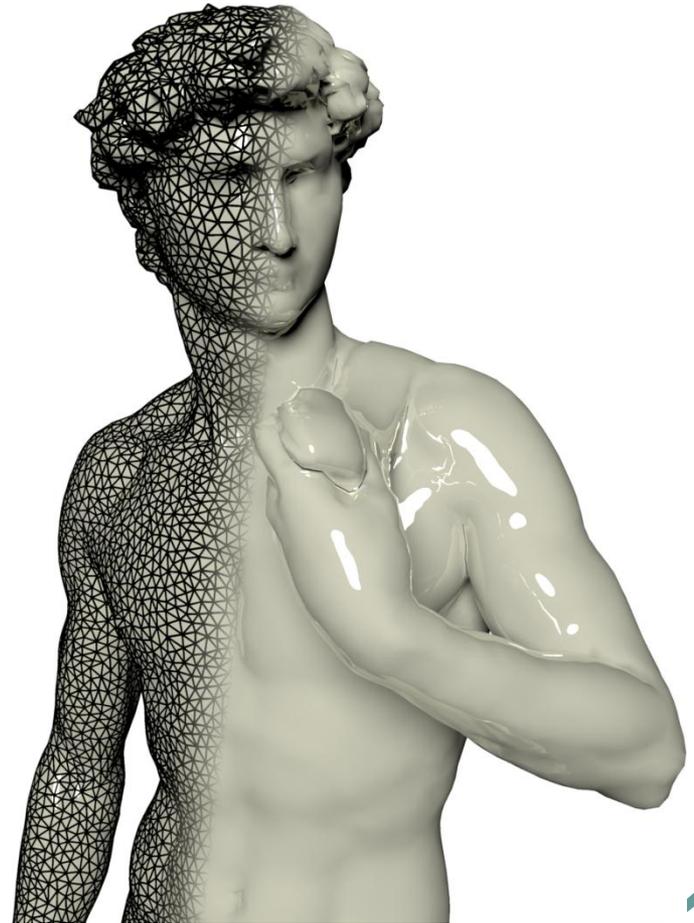
Remplissage de polygones (1)

- ❑ L'algorithme de base le plus populaire pour remplir un polygone consiste à balayer le polygone par des lignes horizontales (lignes de balayage).
- ❑ A chaque ligne, on répète les étapes suivantes:
 1. trouver les intersections de la ligne de balayage avec toutes les arêtes du polygone
 2. trier ces intersections dans l'ordre croissant des coordonnées x
 3. mettre à la valeur donnée tous les pixels situés entre les paires d'intersection.



Modélisation : But du cours

- Comment modélise t'on un objet 3D, inventaire ?
- Quel modèle pour quelle application ?



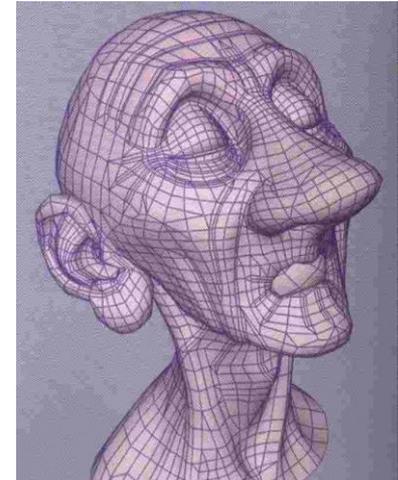
Approches de modélisation

Notion de « modèle géométrique »

- Modèle mathématique de l'objet virtuel (équation de sa surface)

Comment créer ce modèle ?

- Qu'il s'affiche vite ?
- Qu'il n'occupe pas trop de mémoire ?
- Qu'on puisse facilement le modifier ?



La modélisation

- Deux types d'objets à modéliser
 - des objets virtuels / inventés
 - les objets réels
- Plusieurs modes de création
 - **interactive** par des graphistes
 - **automatique** à partir du réel
 - scanner 3D, données médicales, etc...
 - **procédurale**
 - scènes complexes, terrain, etc...
 - Ou un mélange de ces modes...
- Différentes utilisations
 - affichage, animation, simulation physique, etc...

La modélisation

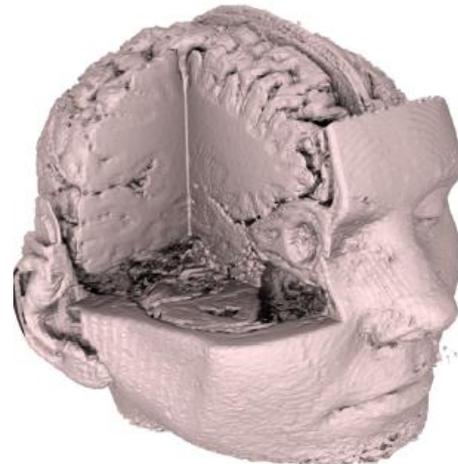
- Construction des objets de la scène
 - Primitives géométriques
 - Opérations de transformations (placer les objets)
- Difficultés
 - Scène 3D / écran 2D
 - Placement des objets
 - Modification des objets (maillages)

Les modèles géométriques

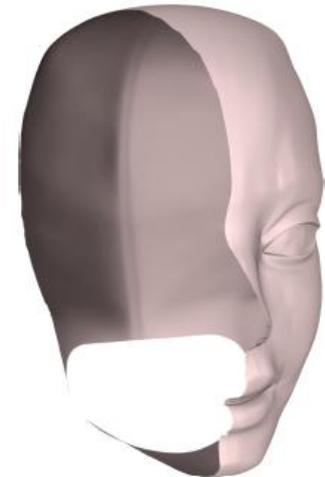
- Fil de fer
- Modèles **non structurés**
 - points
- Modèles **surfacciques**
 - maillages
 - surfaces paramétriques
 - surfaces de subdivision
 - surfaces implicites
- Modèles **volumiques** (Solides)
 - octrees
 - CSG
- Modèles **procéduraux**
 - Fractales
 - Grammaires / L-systems
 - Système de particules
- Modèles **à base d'images**
 - Acquisition et rendu



■ Modélisation volumique



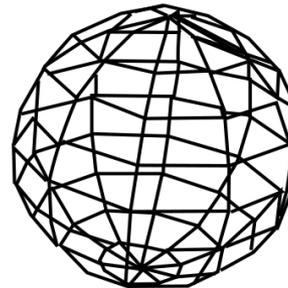
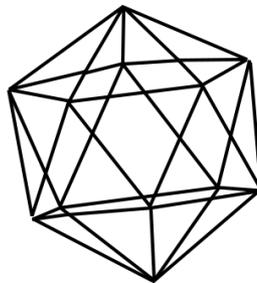
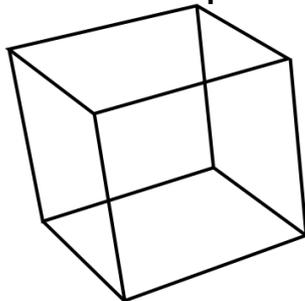
■ Modélisation surfaccique



Les modèles géométriques

Modélisation en fil de fer

- Ce modèle ne conserve que les coordonnées (x, y, z) des sommets et les arêtes qui les joignent.
- Il fait apparaître l'objet comme étant un ensemble de points reliés par des traits,
- Avantages: Ce modèle:
 - Nécessite un minimum de puissance de calcul.
 - Offre un gain d'espace mémoire.
 - Assure une visualisation très rapide (temps réel).
 - Représente la forme générale de l'objet.
 - Il est très simple. .



Les modèles géométriques

Modélisation en fil de fer

- ***Inconvénients***
 - L'ambiguïté, rien ne distinguant le vide du plein.
 - L'impossibilité d'obtenir des images réalistes.
 - La possibilité de créer des objets sans aucun sens physique.
 - La difficulté de résoudre le problème de l'élimination des parties cachées.

Les modèles géométriques

Modèles non structurés

Nuages de points

- Modèles obtenus
 - par digitalisation manuelle ou par scanner
 - par reconstruction à partir d'images
 - par échantillonnage d'un autre modèle
- Avantages
 - "naturels" i.e. conception atomiste
 - simples à afficher
 - théorie puissante pour l'antialiasing
 - rendu adaptatif
- Inconvénients
 - coût mémoire
 - difficilement éditable
 - Calcul de la normale



Les modèles géométriques

Modélisation surfacique

- Maillages
- Surfaces paramétriques
- Surfaces de subdivision
- Surfaces implicites

Les modèles géométriques

Modélisation surfacique

□ Ce modèle permet de définir les objets par l'association des morceaux des surfaces (tranches, carreaux) à des contours (bords) délimités par les arêtes.

□ L'approche surfacique est supérieure à l'approche fil de fer. Elle relève cependant des faiblesses du modèle précédent.

Avantages:

La modélisation surfacique:

- Est plus développée que l'approche fil de fer.
- Est assez utilisée en CFAO.
- Permet de créer des objets lisses.
- Permet de construire des objets très complexes

• **Inconvénients:**

- Plus coûteuse en espace mémoire
- Le temps écoulé est grand que celui du modèle fil de fer.
- Problème de "rattachement", chaque surface est décrite indépendamment aux autres.

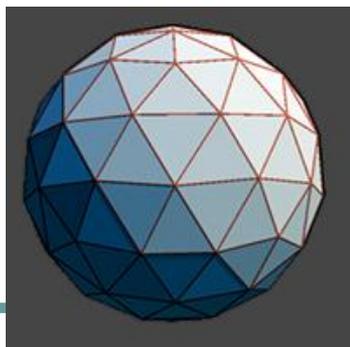
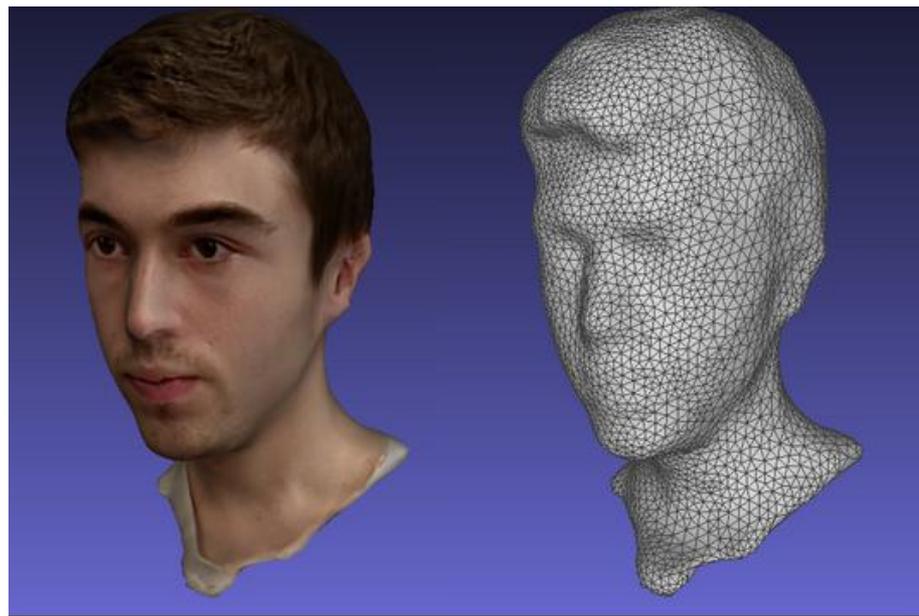
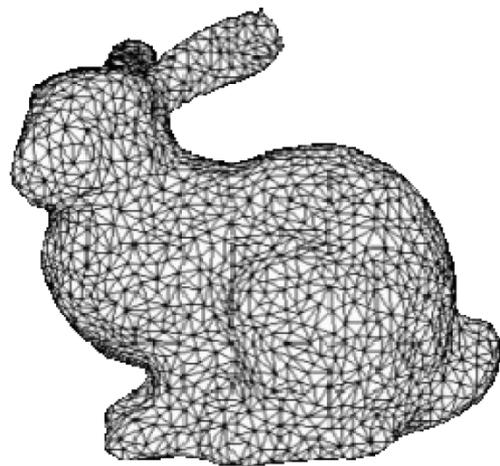
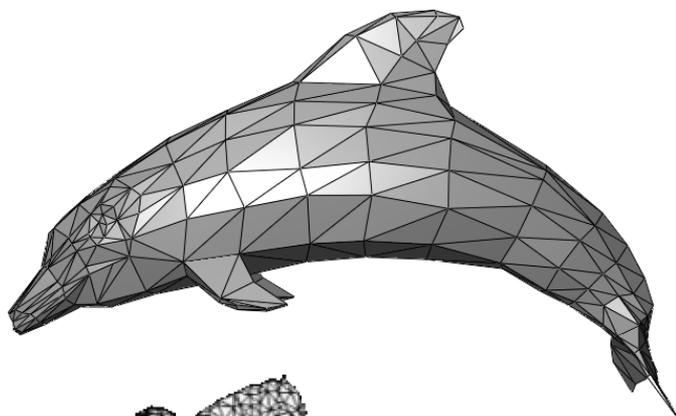
Modélisation surfacique: Maillages de polygones

Maillages de polygones

- Un **mesh** ou **maillage** est un objet tridimensionnel constitué de sommets, d'arêtes et de faces organisés en polygones sous forme de fil de fer.
- Les faces se composent généralement de triangles, de quadrilatères ou d'autres polygones convexes simples, car cela simplifie le rendu.
- Les faces peuvent être combinées pour former des polygones concaves plus complexes, ou des polygones avec des trous.
- On peut effectuer de nombreuses opérations sur les *meshes*, comme la logique booléenne, le lissage, etc.
- Affichage facile avec OpenGL
- Des outils de modélisation graphique tels que Blender ou 3D Studio Max permettent de réaliser ces objets filaires.

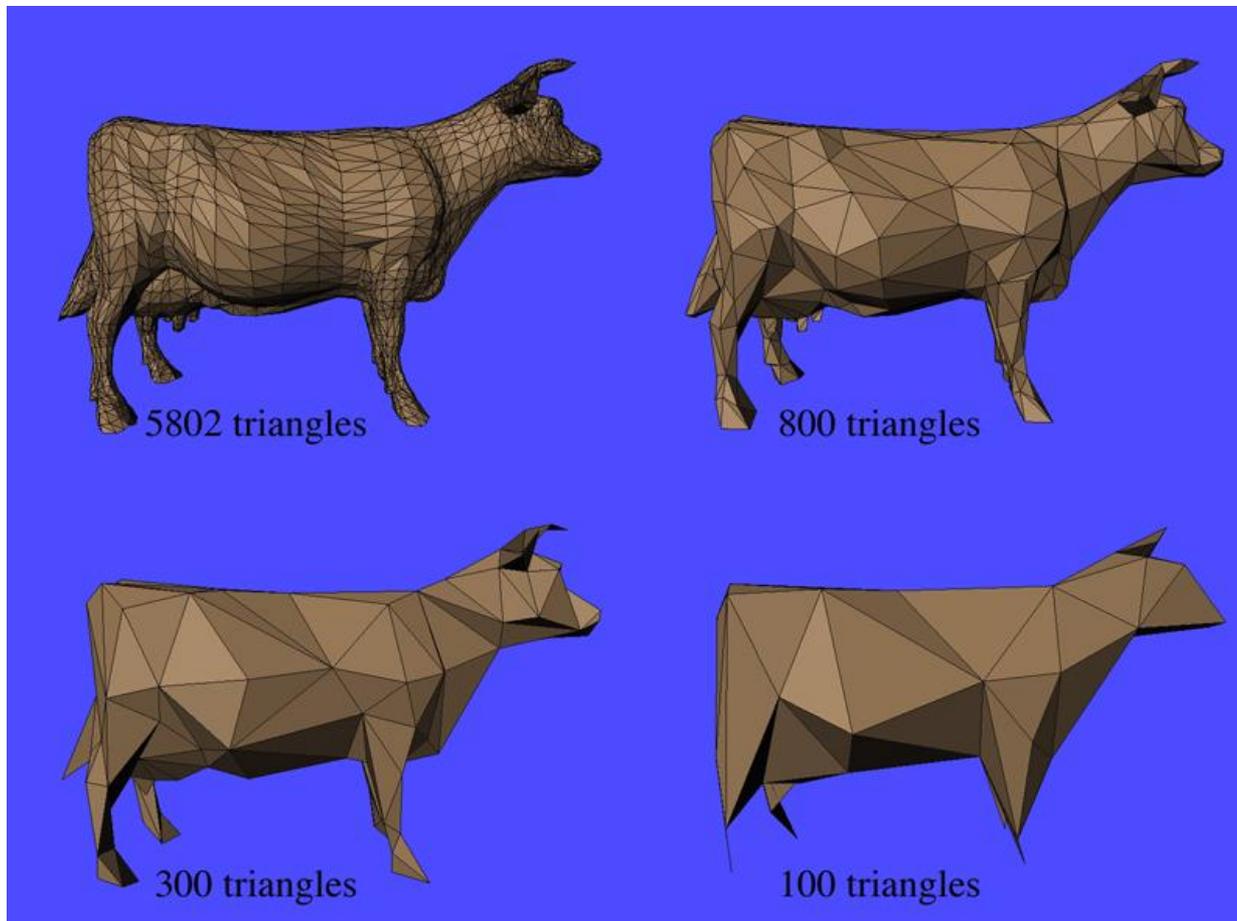
Modélisation surfacique: Maillages de polygones

Maillages de polygones



Modélisation surfacique : Niveaux de détails des Surfaces Triangulées LOD (Level Of Details)

Différentes représentations du même objet sont calculées, chacune a un niveau de détail différent. Pendant l'exécution de l'application, une représentation de l'objet est sélectionnée et visualisée.



Modélisation surfacique : Niveaux de détails des Surfaces Triangulées LOD (Level Of Details)

Objectif

• Observations :

- objet détaillé → énormément de polygones
- coût de rendu = fonction du nombre de polygones
- taille des polygones en espace image = fonction de la distance à l'observateur.

• Idée :

- adapter la résolution (le nombre de polygones) du maillage en fonction du point de vue

• Défis :

- comment calculer des versions simplifiées du maillage ?
- comment choisir la résolution adaptée ?
- comment rendre cela efficace du point de vue du GPU ?

Modélisation surfacique: Surfaces paramétrées

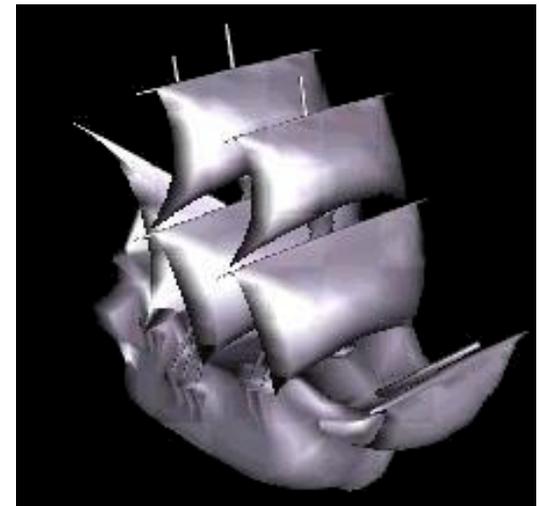
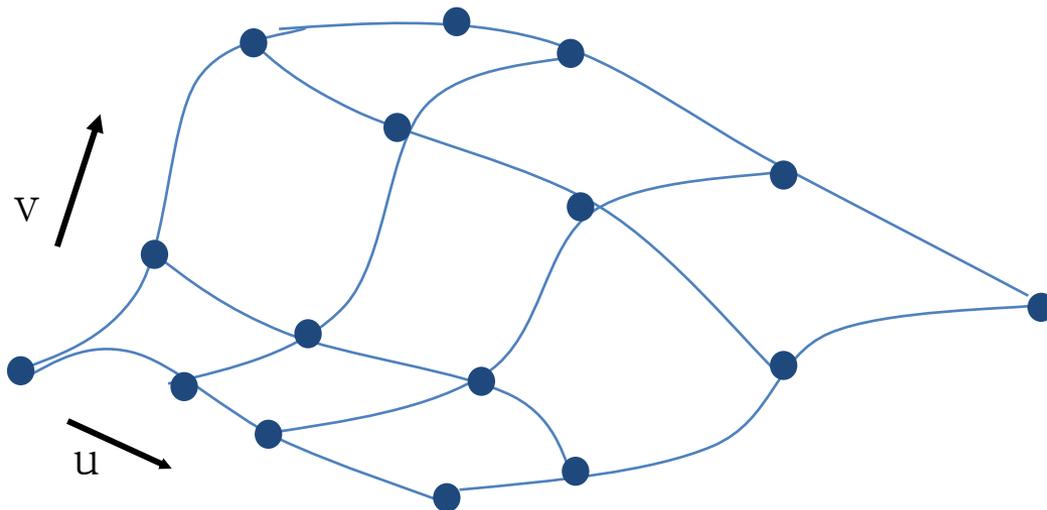
La modélisation paramétrique est un type de modélisation explicite qui utilise des fonctions.

Cela sert de manière historique à décrire des courbes et des surfaces. On peut facilement identifier les propriétés de continuité grâce à une analyse différentielle des fonctions mises en œuvre. Les fonctions utilisées sont souvent des polynômes pour leur caractère lisse. Plus le degré du polynôme est élevé, plus la courbe ou la surface sont lisses.

Modélisation surfacique: Surfaces paramétrées

Facettes: Ensemble de carreaux

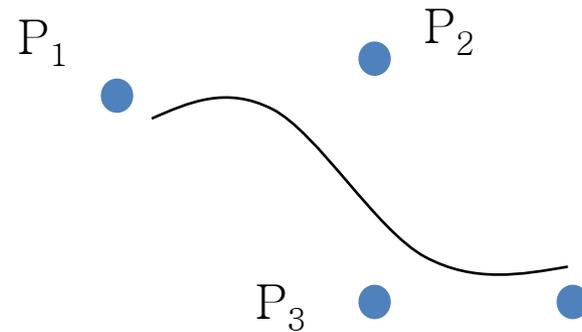
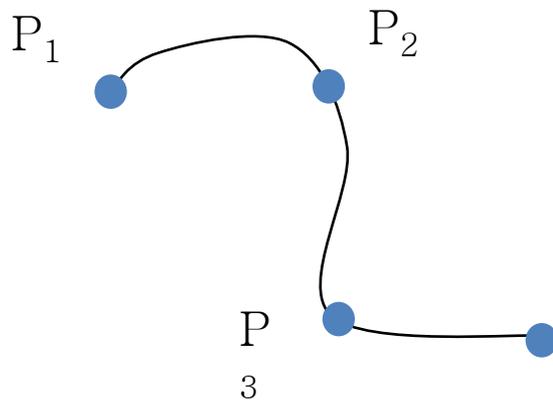
Carreau: un morceau de surfaces paramétrique (Bezier, Bsplines, COONS...)



Modélisation surfacique: Surfaces paramétrées

Facettes: Ensemble de carreaux

- définition à partir de points de contrôle
- Interpolation ou approximation



Modélisation surfacique: Surfaces paramétrées

Surfaces de formes libres

□ La représentation paramétrique d'une courbe dans l'espace est de la forme :

$$\square \mathbf{C}(t) = (\mathbf{X}(t), \mathbf{Y}(t), \mathbf{Z}(t))$$

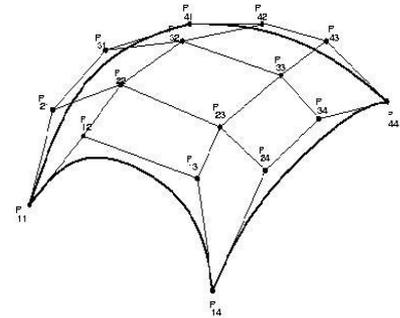
$$\square \text{ où } t \in [t_{\min}, t_{\max}]$$

□ Ainsi que celle d'une surface :

$$\square \mathbf{S}(u, v) = (\mathbf{X}(u, v), \mathbf{Y}(u, v), \mathbf{Z}(u, v))$$

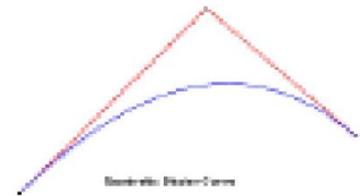
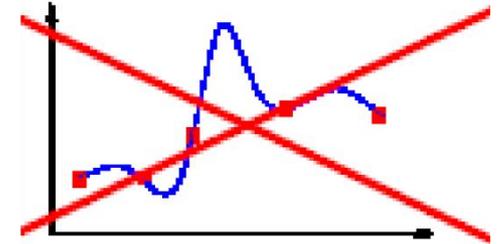
$$\square \text{ où } u \in [u_{\min}, u_{\max}], \quad v \in [v_{\min}, v_{\max}]$$

□ Chaque valeur de t (pour les courbes) et de u et de v (pour les surfaces) correspond à un point dans la courbe (surface).

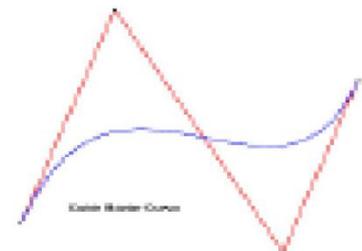


Modélisation surfacique: Surfaces paramétrées

- Courbes et surfaces de formes libres
- Définition par des points de contrôle
 - pas de polynôme de haut degré
 - forme imprévisible et pas souhaitable
 - influence globale de chaque point (instable)
 - plutôt un contrôle local
 - on construit des patches (carreaux de surface) qui ne dépendent que des points voisins...
 - ...et qui sont des polynômes de faible degré
 - on s'arrange pour que les patches se raccordent de façon C^1 ou plus (la continuité paramétrique est vérifiée à l'ordre 1 (C^1 -continuité)).



quadratic



cubic

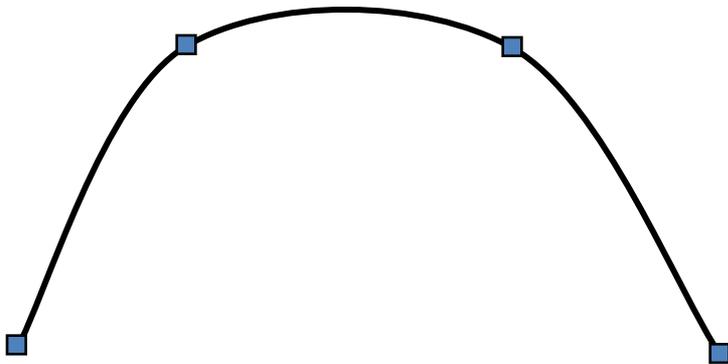
Modélisation surfacique: Surfaces paramétrées

Une courbe ou une surface est généralement donnée par un ensemble de points. On distingue alors 2 types de méthodes:

Les approximations: la courbe ou la surface est définie par les points, mais ne passe pas nécessairement par eux. Ce sont les ajustement et les méthodes de points de contrôle. (Approximations de Bézier - Approximations B-spline)

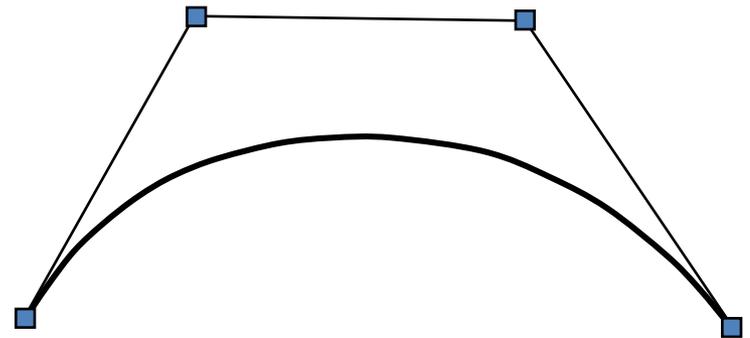
• **Les interpolations:** la courbe ou la surface passe par les points.

Interpolation



La courbe passe par les points de contrôle

Approximation

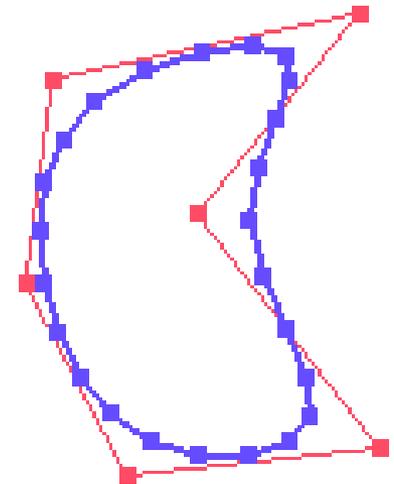


La courbe est attirée par les points de contrôle

Modélisation surfacique: Surfaces paramétrées

Interpolation

- Méthodes globales:
 - Interpolation de Lagrange
 - Interpolation d'Hermite
 - Inconvénients en CAO
 - Trop de calculs, résolution de systèmes linéaires
 - Résultats parfois mauvais: trop d'ondulations
 - Modification d'un point?
- Fonction spline cubique d'interpolation inconvénients:
 - Calculs longs
 - Modifications pas complètement locales
 - Ondulations



spline cubique

Modélisation surfacique: Surfaces paramétrées

Approximation

Définition

À partir des n points P_1, \dots, P_n , la courbe d'approximation est définie par :

$$P(t) = \sum_{i=1}^{i=n} P_i \varphi_{i,k}(t)$$

où $\varphi_{i,k}(t)$ pour $i=1$ à n sont les fonctions de base et t le paramètre

Pour obtenir une bonne approximation, les fonctions de base doivent:

être à support le plus local possible

le plus lisse possible

$$\sum_i \varphi_{i,k}(t) = 1 \text{ en tout point}$$

Approximations classiques

si les $\varphi_{i,k}(t)$ sont les polynômes de Bernstein => approximation de Bézier

si les $\varphi_{i,k}(t)$ sont les B-splines d'ordre k => approximation B-spline

Modélisation surfacique

Surfaces paramétrées: Carreaux de Bézier

- **Produit tensoriel**

- 2 paramètres u et v
- Réseau de points de contrôle $P_{i,j}$

- Surface de Bézier: est calculée à partir d'une grille de $M \times N$ points de contrôle selon les formules suivantes:

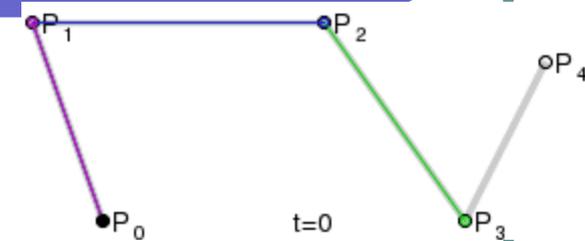
$$P(u,v) = \sum_{i=0}^n \sum_{j=0}^m P_{i,j} B_{i,n}(u) B_{j,m}(v)$$

$B_{i,j}$: Polynômes de Bernstein

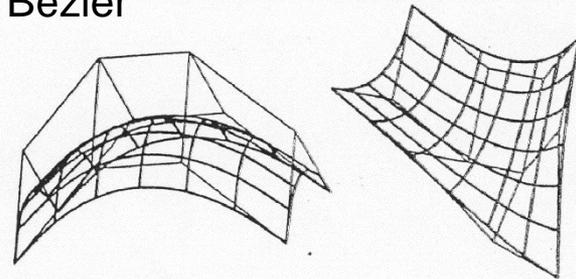
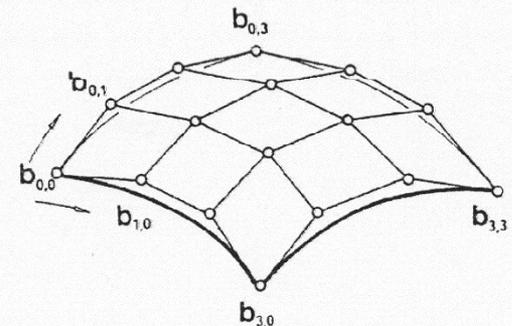
$P_{i,j}$: points de contrôle

- **Propriétés :**

- les frontières du carreau sont les courbes de Bézier
- la surface appartient à l'enveloppe convexe, définie par $(n+1)(m+1)$ points de contrôle $P_{i,j}$



Exemple de construction de courbe de Bézier.



Modélisation surfacique

Surfaces paramétrées: Recollement des surfaces de Bézier

- soit 2 surfaces : Q définie par q_{ij} ,

R définie par r_{ij}

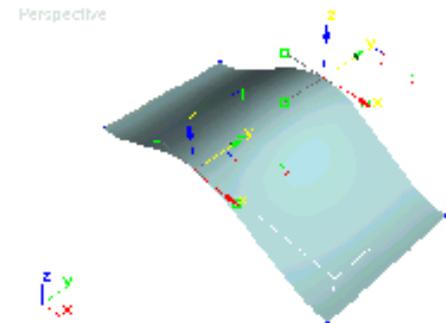
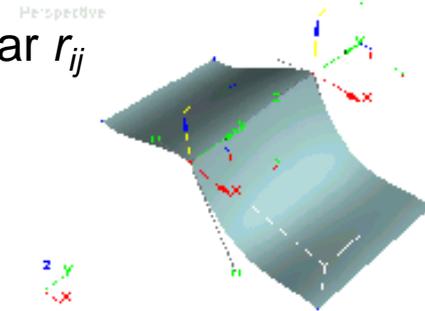
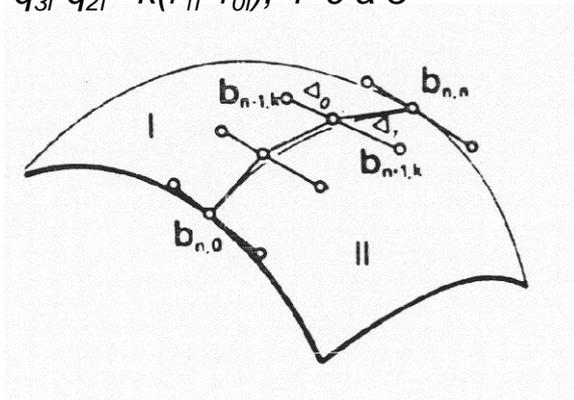
- continuité d'ordre zéro

- $q_{3i}=r_{0i}$, $i=0$ à 3

- continuité d'ordre un:

- alignement des tangentes

- $q_{3i}-q_{2i}=k(r_{1i}-r_{0i})$, $i=0$ à 3



Modélisation surfacique

Surfaces paramétrées: Surfaces B-splines

- Produit tensoriel

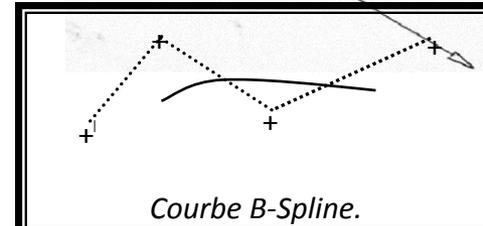
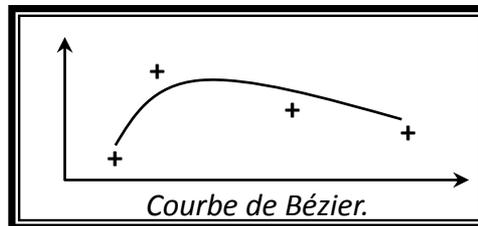
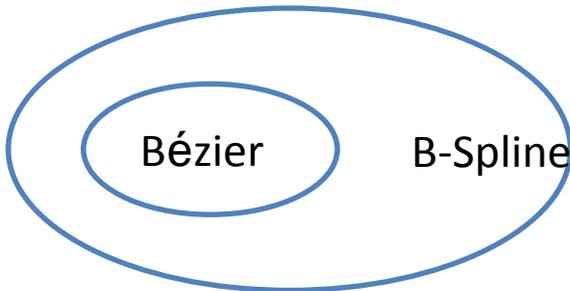
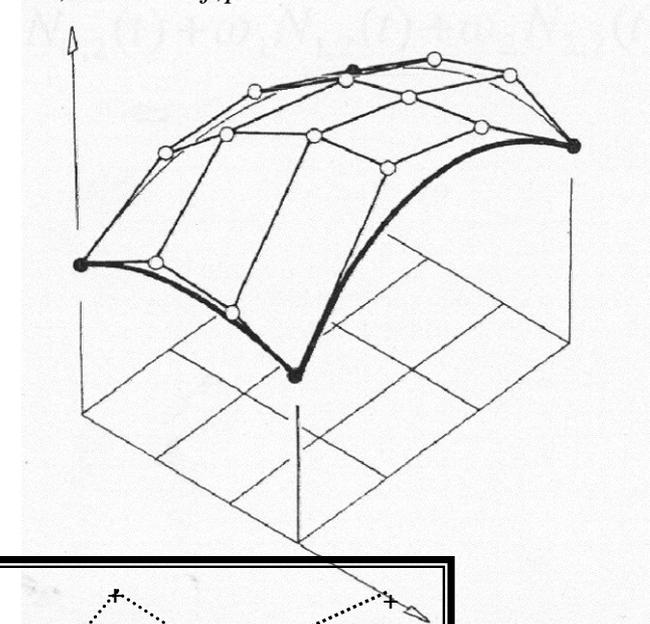
- 2 paramètres u et v

- Réseau de points de contrôle $P_{i,j}$

- Surface B-spline :
$$P(u,v) = \sum_{i=0}^{i=n} \sum_{j=0}^{j=m} P_{i,j} N_{i,k}(u) N_{j,p}(v)$$

Les B-splines vérifient une relation de récurrence:

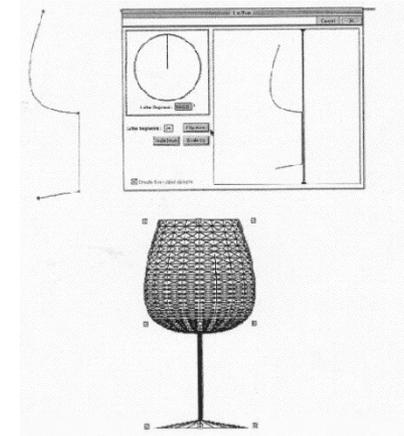
$$N_{i,1}(t) = \begin{cases} 1 & \text{si } t_i \leq t < t_{i+1} \\ 0 & \text{sinon} \end{cases}$$



Modélisation surfacique

Surfaces paramétrées: Surfaces créées à partir de courbes

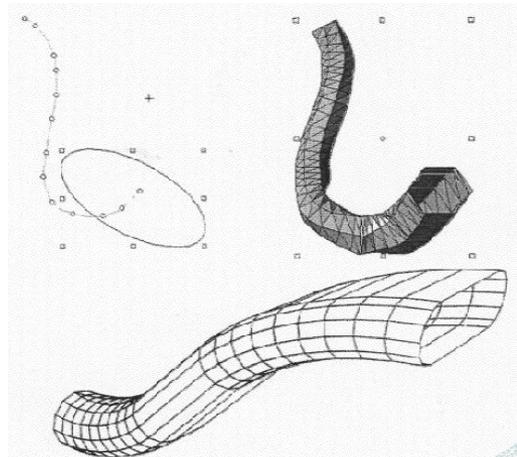
- Surfaces de révolution: Surface créée à partir
 - d'une courbe
 - un axe de rotation
 - position de la courbe par rapport à l'axe de rotation
 - un angle de rotation



- Surfaces extrudées: Surface créée à partir d'une courbe plane en lui donnant de l'épaisseur



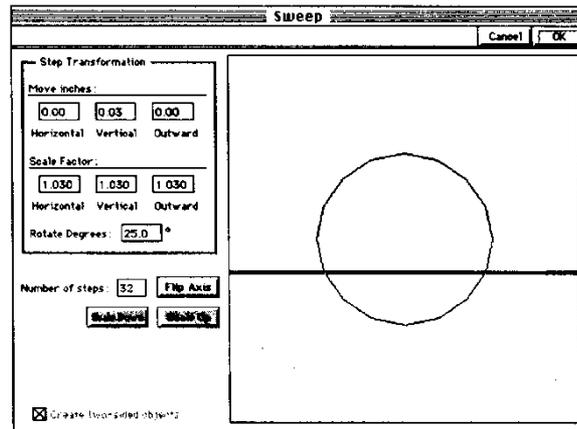
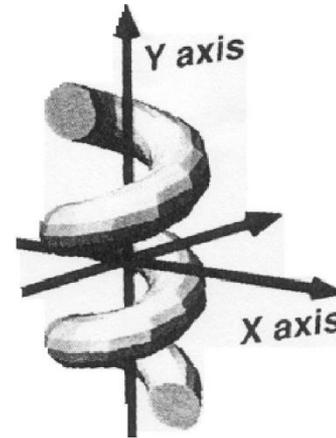
- **Extrusion généralisée**
 - Une courbe plane fermée
 - une trajectoire
 - position et modification de la courbe plan le long de la trajectoire



Modélisation surfacique:

Surfaces paramétrées: Surfaces créées à partir de courbes

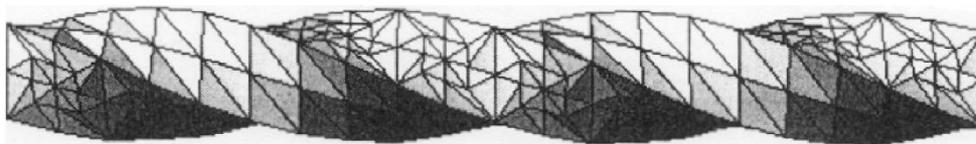
- Sweeping: Construction par déplacement
 - Une courbe plane
 - Un axe de rotation
 - Un angle de rotation
 - Un déplacement



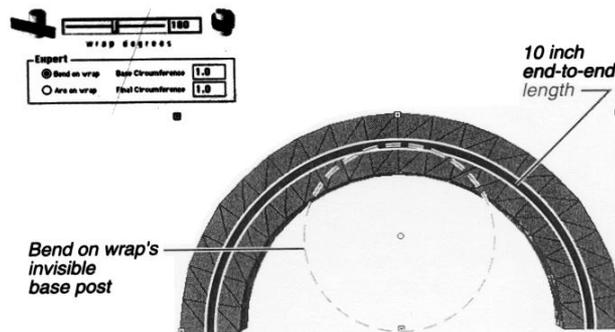
Modélisation surfacique:

Surfaces paramétrées: Construction par déformation (warping)

- Construction par Déformation
 - Torsion



- Enroulement



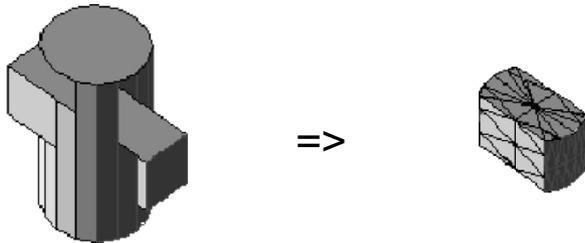
Modélisation surfacique

Surfaces paramétrées: Composition booléenne de volumes

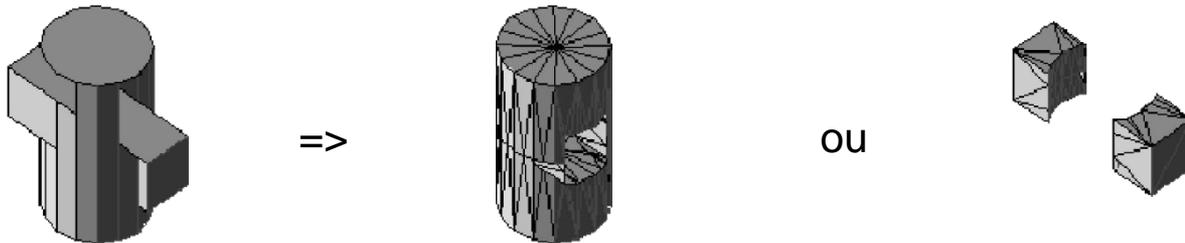
- Opérateur booléen: Union



- Intersection

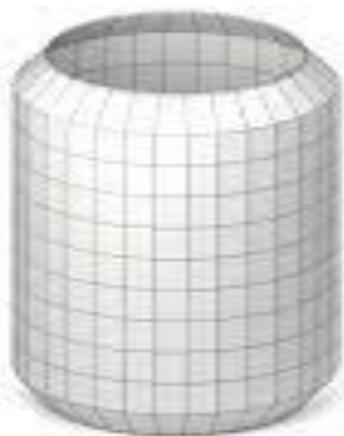


- Différence



Modélisation surfacique

Surfaces de subdivision



Modélisation surfacique

Surfaces implicites

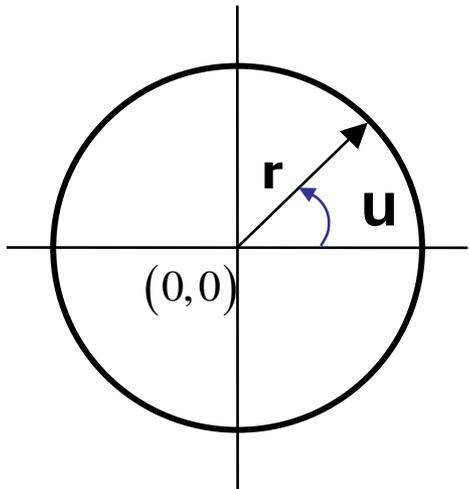
□ Une surface implicite tridimensionnelle est une surface constituant de points $P(x, y, z)$, satisfaisant la fonction implicite $f(x, y, z)=0$.

Modélisation surfacique

Surfaces implicites

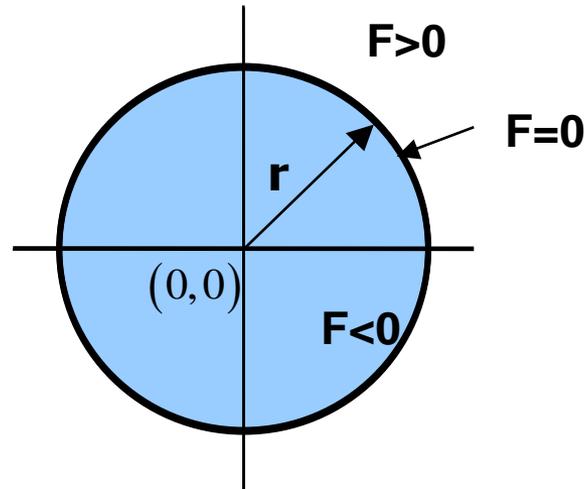
□ Une surface implicite tridimensionnelle est une surface constituant de points $P(x, y, z)$, satisfaisant la fonction implicite $f(x, y, z)=0$.

Paramétrique



$$\begin{aligned}x(u) &= r \cos(u) \\y(u) &= r \sin(u)\end{aligned}$$

Implicite



$$F(x,y) = x^2 + y^2 - r^2 = 0$$

Modélisation surfacique

Surfaces implicites

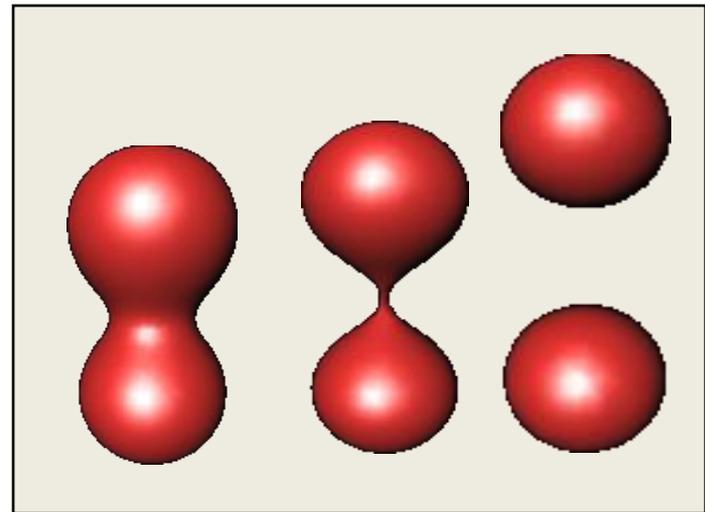
$$S = \{ P(x,y,z) / f(x,y,z) = iso \}$$

Intérêt : Combiner des éléments

union : $f = \max(f_1, f_2)$

Intersection : $f = \min(f_1, f_2)$

« mélange » : $f = f_1 + f_2$



Modélisation surfacique

Surfaces implicites

Avantages :

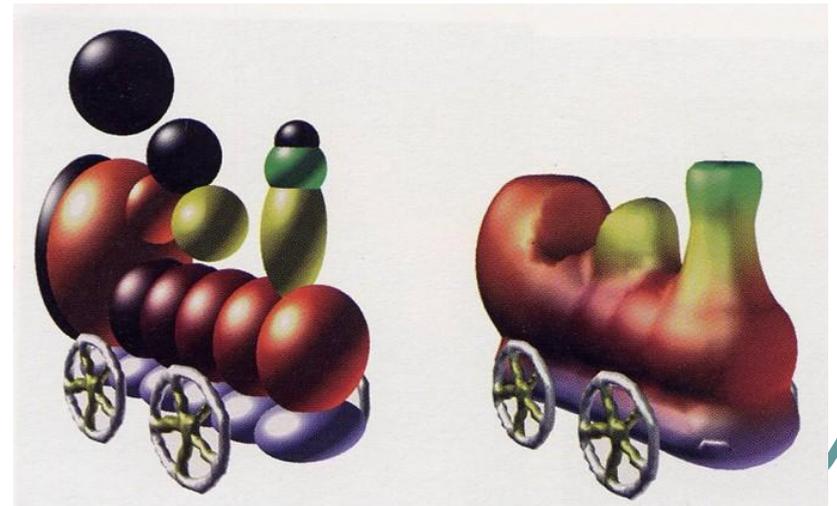
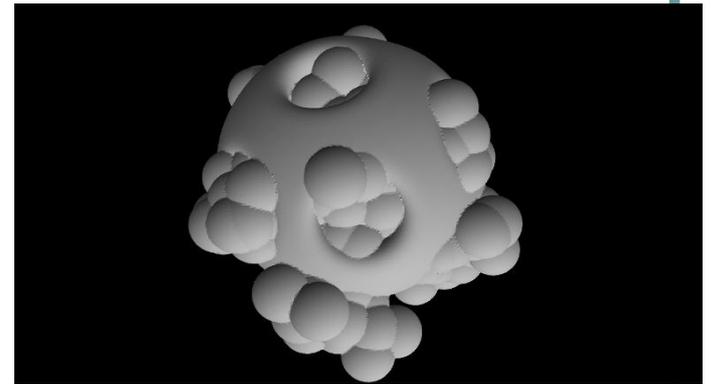
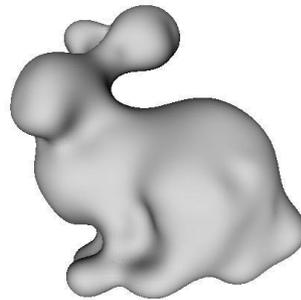
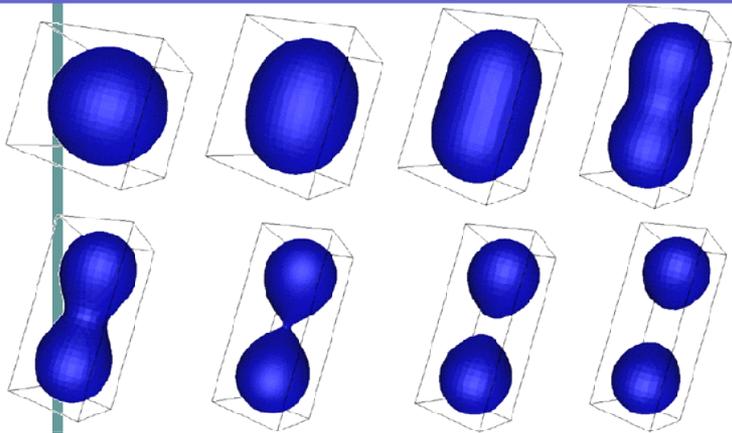
- La possibilité de créer des objets plus lisses.
- Permet de fusionner plusieurs formes.
- Définie une surface avec peu de paramètres, par rapport à la surface paramétrique.

Inconvénients:

- Rendu coûteux (en temps de calcul).
- Difficulté de placage de texture (nécessite une polygonalement de la surface).

Modélisation surfacique

Surfaces implicites



Modélisation volumique

- ❑ Les objets 3D ne sont pas traités comme un assemblage de polygones, mais comme des entités entières (volumes).
- ❑ Elle consiste à fournir des informations sur l'espace occupé par l'objet.
- ❑ Pour définir et manipuler un objet en volume, il doit traduire leurs propriétés (sommets, arêtes et rayon) en formules mathématiques.
- ❑ Cette approche décrit et visualise des objets solides.
- ❑ Parmi les types de modélisation volumique on distingue les méthodes (CSG, BREP, Hybride).

Modélisation volumique

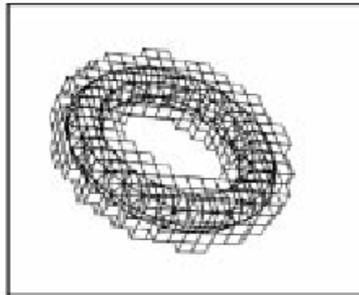
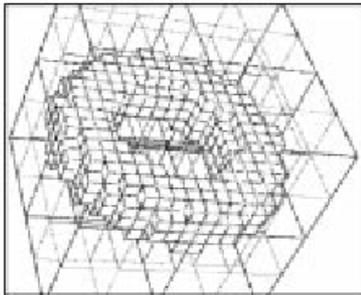
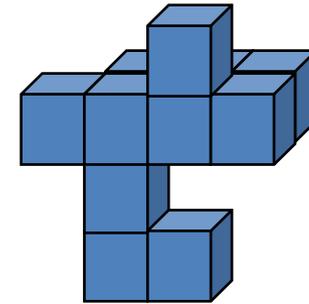
Volumes discrets: voxels

Voxels = éléments d'une grille 3D

Présence ou absence de matière

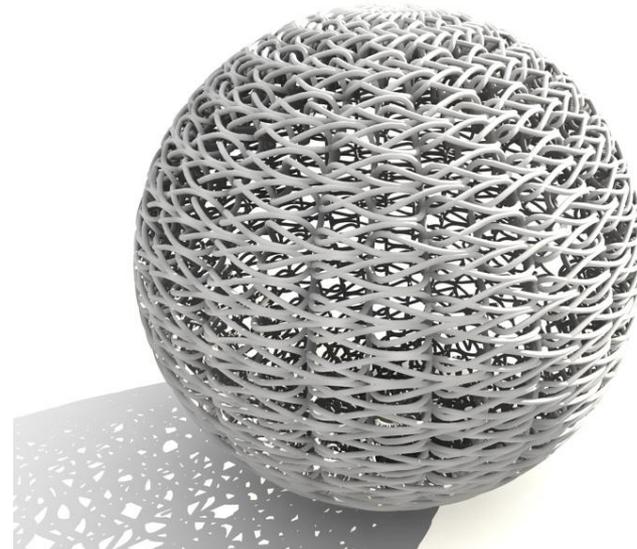
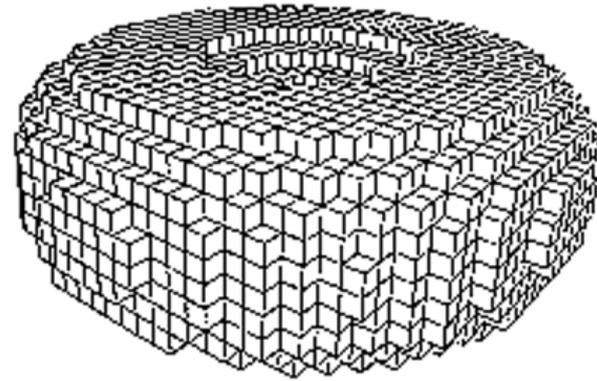
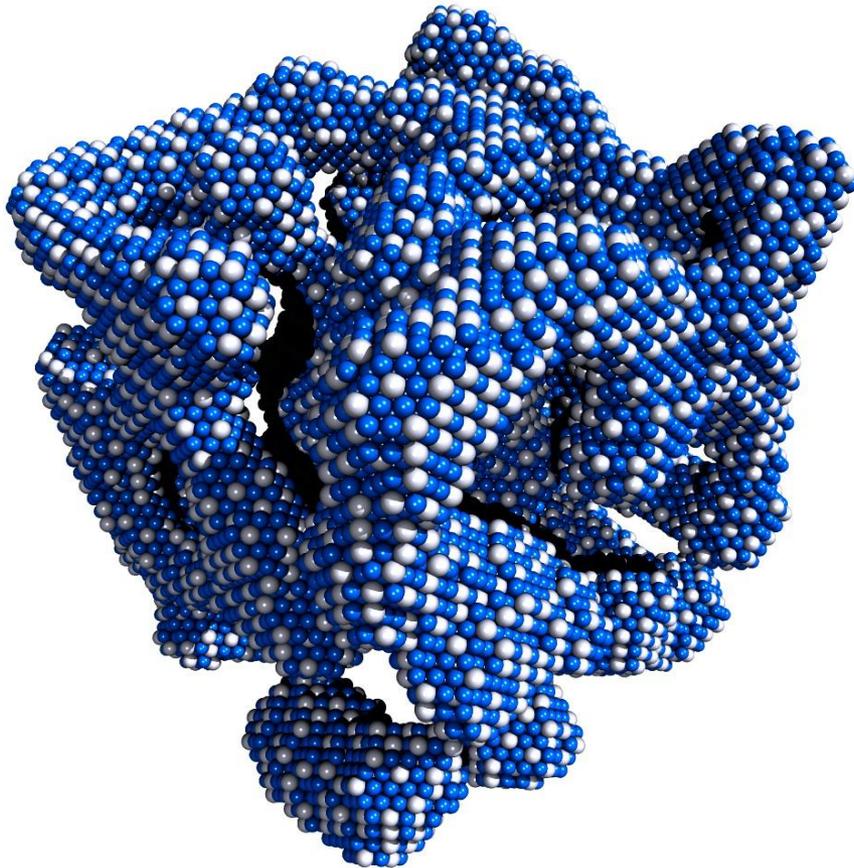
une définition –dite voxalisée- de V

est donnée par :

$$V = \sum_{i=1}^N v_i$$


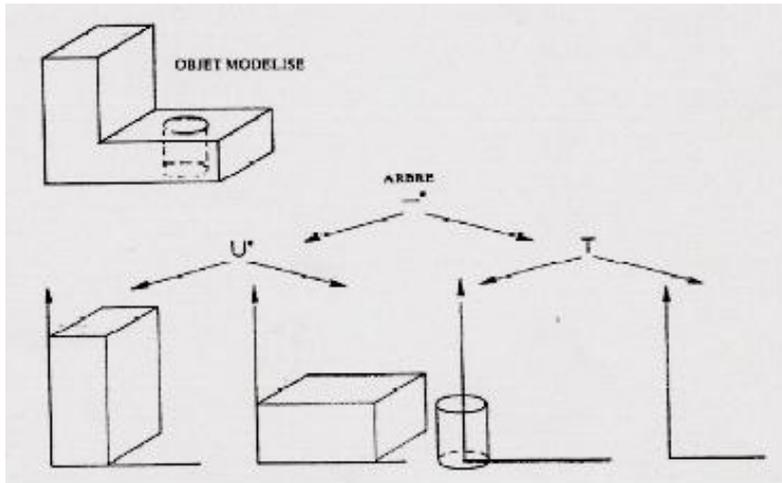
Modélisation volumique

- ❑ **Constructive Solid Geometry (CSG)**
- ❑ **Octrees**



Modélisation volumique: Constructive Solid Geometry (CSG)

- Représentation par arbre de construction:CSG



- Représentation par une grammaire
Exp -> prim / transf prim / op exp exp
Prim -> cube / sphere / cone / ...
Transf -> translation / homothétie / rotation
Op -> union / intersection / différence

Modélisation volumique: Constructive Solid Geometry (CSG)

Générer des formes complexes à l'aide de primitives.

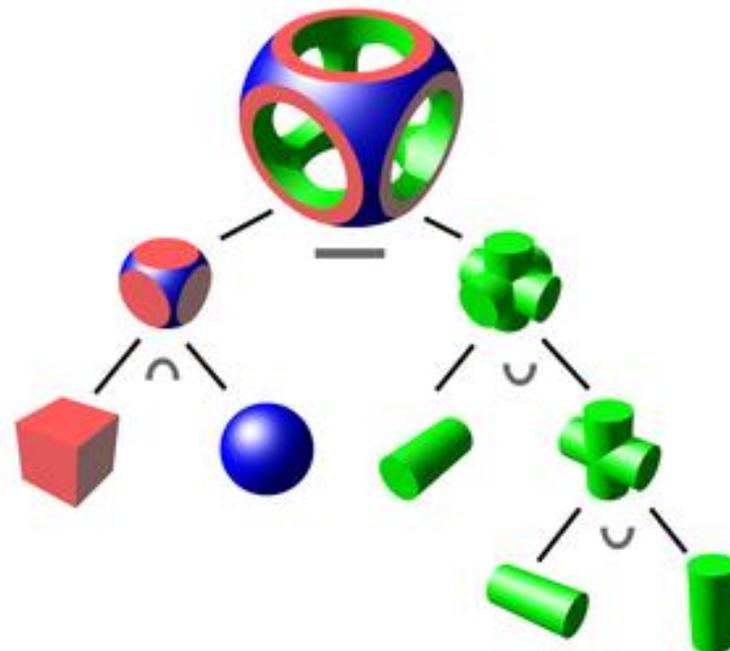
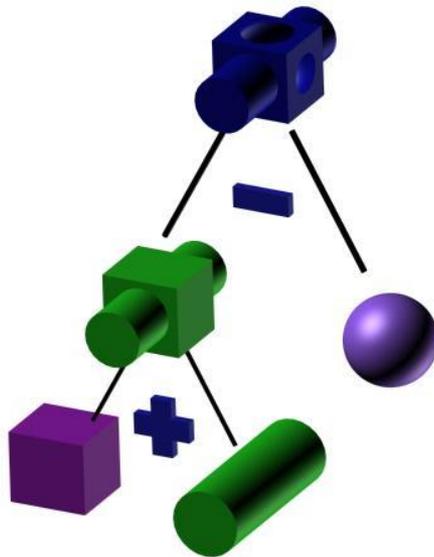
Dessiner un objet : rogner des parties, percer des trous, ...

Coller des pièces entre-elles

Utilisé généralement dans la CAO.

Operation possibles

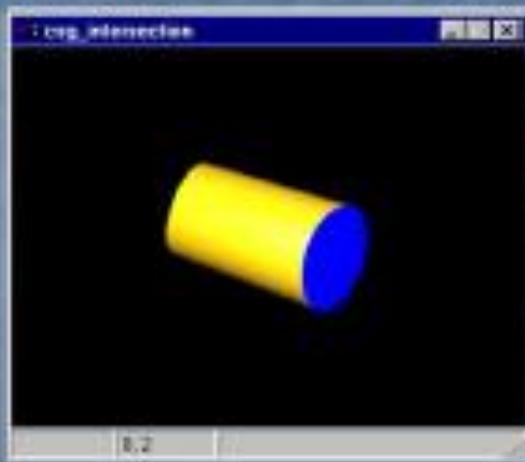
- union
- intersection
- soustraction



Modélisation volumique: Constructive Solid Geometry (CSG)

Opérateurs booléens

- L'intersection (\cap)
 - un objet complexe est constitué par le volume résultant de l'intersection de 2 objets plus simples
 - Exemples :

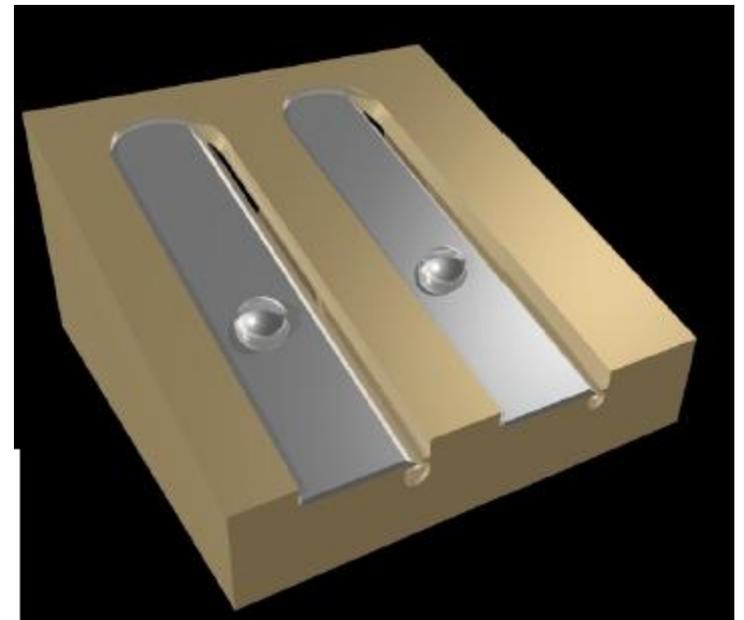
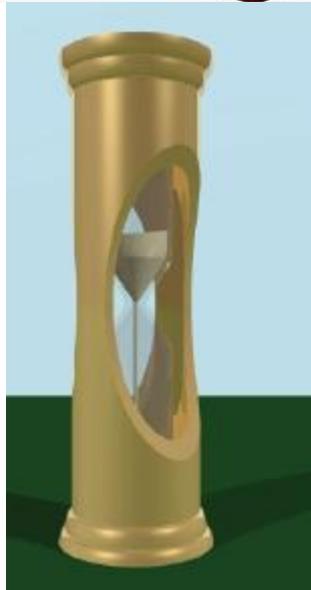
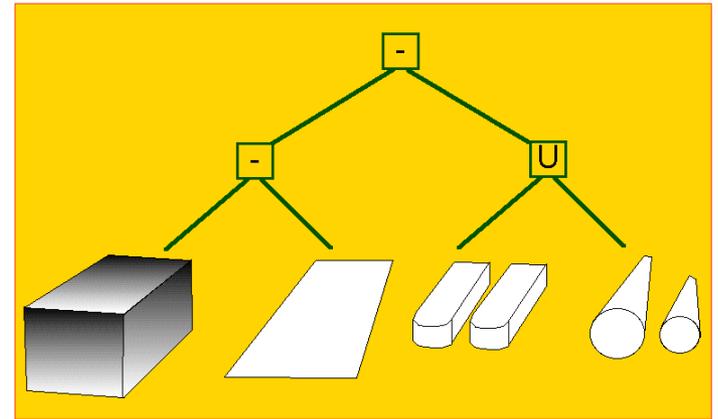
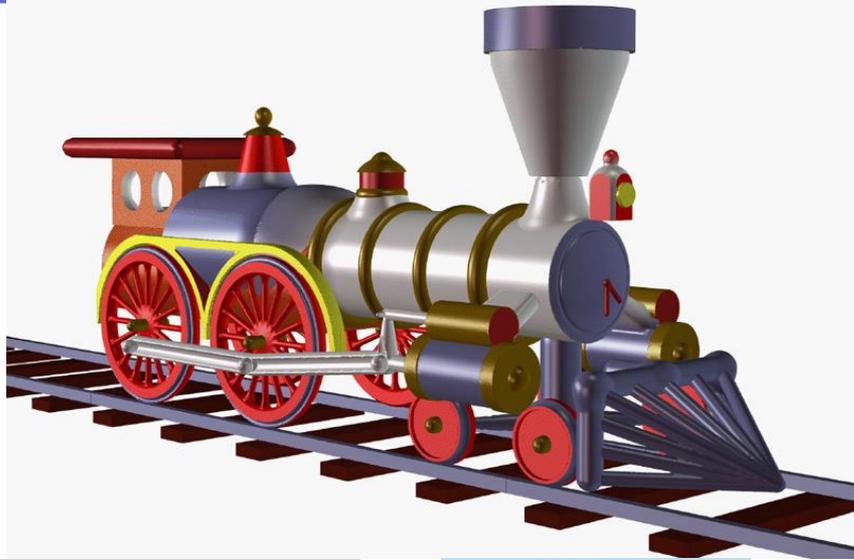


Sphère \cap Cylindre



Cube \cap Sphère

Modélisation volumique: Constructive Solid Geometry (CSG)



Modélisation volumique: Octrees

- Les solides sont représentés par une listes de voxels (éléments de volume). Les voxels sont des cellules spatiales occupées par le solide; ce sont généralement des cubes de taille fixe réparties selon une grille.
- La principale méthode de décomposition en cellules est l'**Octree** qui est une généralisation des **Quadtrees**.
- Un **quadtree** est une représentation d'un objet 2D basée sur la subdivision récursive d'un carré en 4 carrés de même taille (quadrants)

Avantages des Octrees

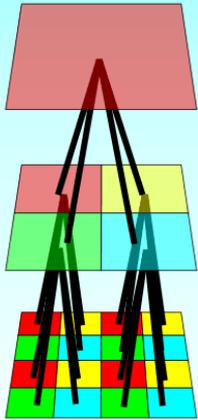
- Facilité de validation.
- Simplicité de l'accès à un point donné et l'unicité spatiale assurée.

Inconvénients des Octrees

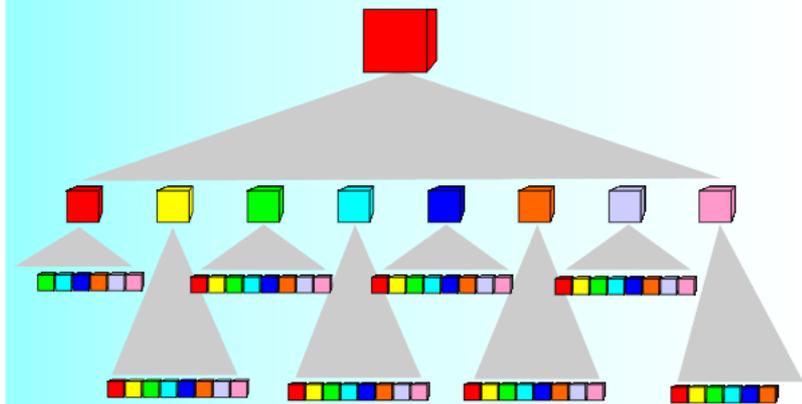
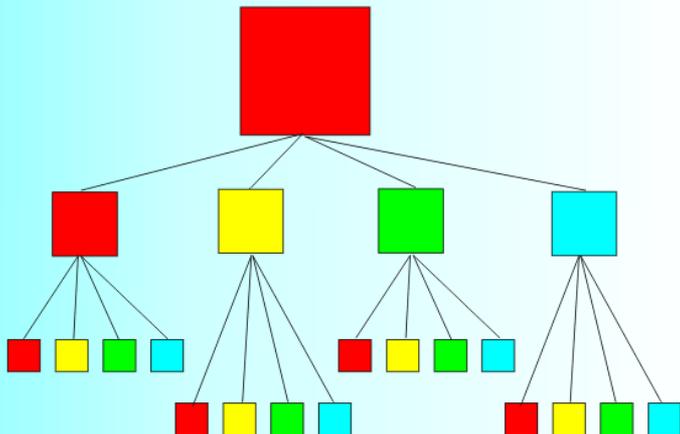
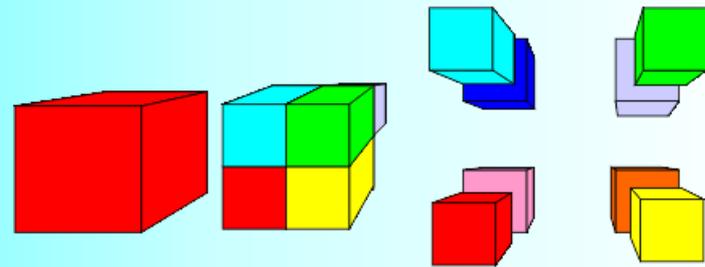
- Pas de structure entre les différentes parties d'un objet.
- Coûteux en termes de mémoire.
- Il n'est pas possible de représenter des objets très complexes.

Modélisation volumique: Octrees

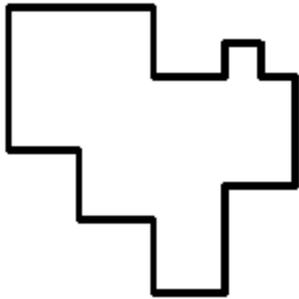
Quadtrees



Octrees



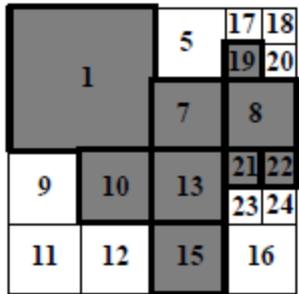
Modélisation volumique: Octrees



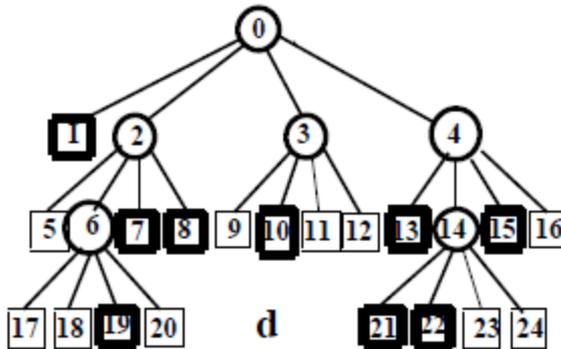
a

1	1	1	1	0	0	0	0
1	1	1	1	0	0	1	0
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	0	0
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0

b

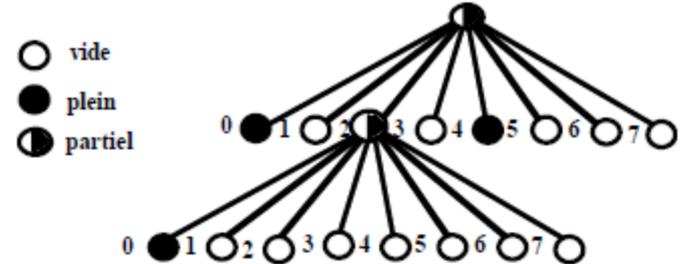
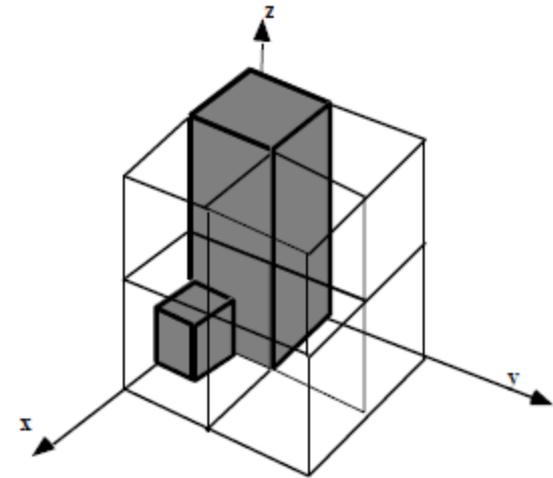


c



d

- noeud représentant une région pleine
- noeud représentant une région vide
- noeud avec descendants



Exemple de quadtree. a: l'objet 2D; b: la représentation en matrice binaire correspondante; c: la décomposition en quadrants; d: la représentation en quadtree

Octree representation of a 3D objet

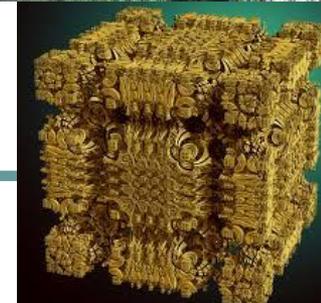
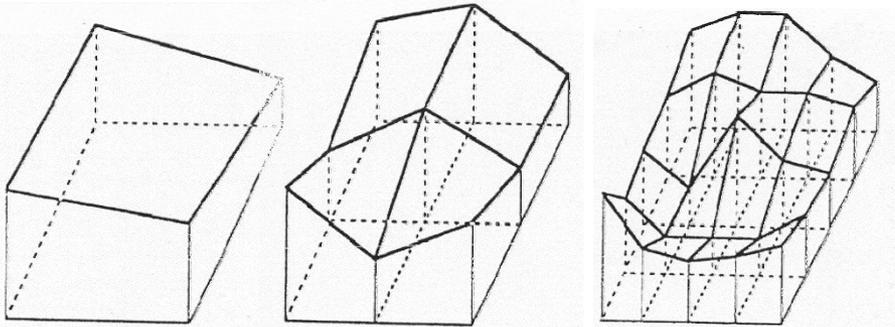
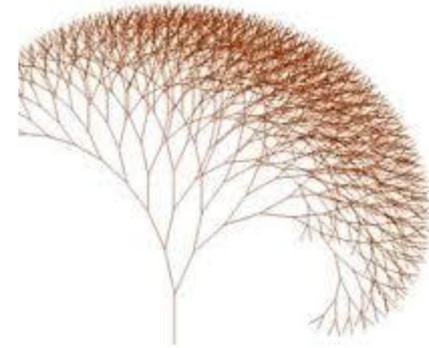
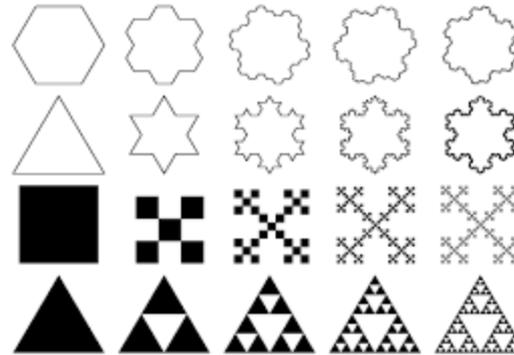
Modélisation procédural:

- ❑ Consiste à représenter des objets naturels complexes comme les arbres, nuages, plantes, feu ...etc.
- ❑ Ses objets sont conçus par l'utilisation de l'un des nouveaux modèles suivants: récurrents (Fractales, Graftales), déformations, Physiquesetc.).

Modélisation procédural: Fractales

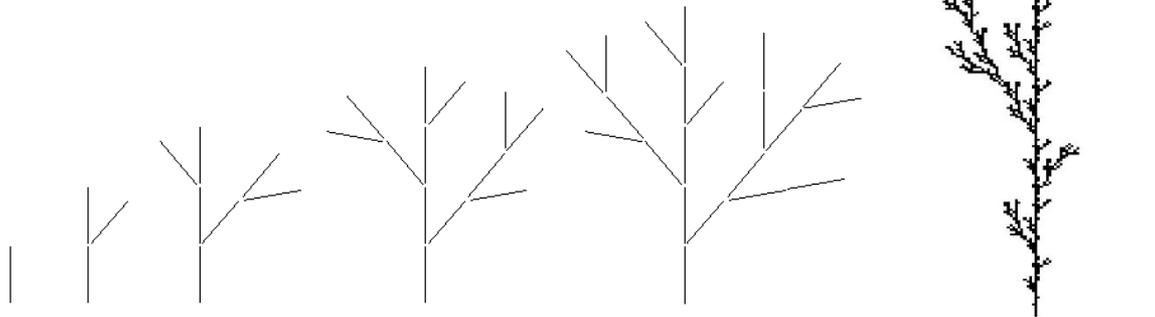
- Montagnes fractales

- Construction récursive du terrain
- Modèle statistique
- axiome
- générateur aléatoire



Modélisation procédural: L-systèmes

- Modélisation de plantes
 - Grammaire décrivant la croissance
 - L'axiome
 - les règles de production
 - un angle
 - le nombre d'itérations
- Exemple:
 - axiome : F
 - règle: $F \rightarrow F[+F]F[-F]F$
 - $\alpha = \pi/7$



Modélisation procédural: L-systèmes

•axiome : S L F F F

•règles:

S \longrightarrow [+++G][---G]TS

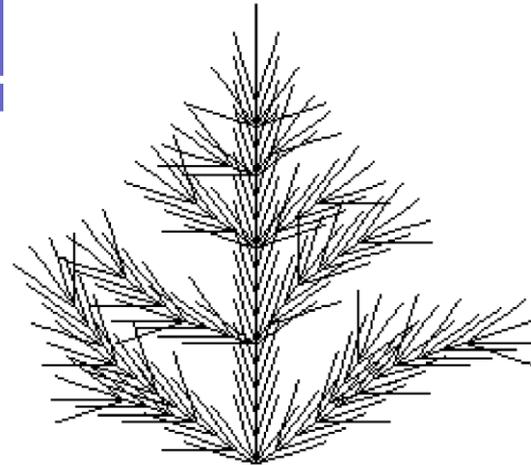
G \longrightarrow +H [-G]L

H \longrightarrow -G [+H]L

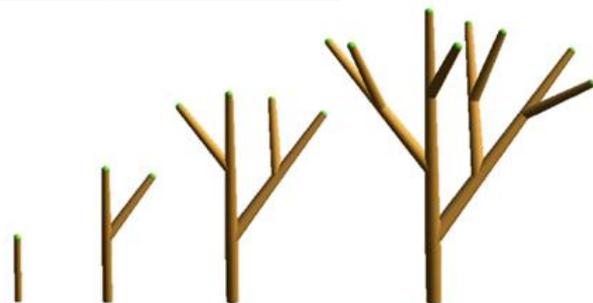
T \longrightarrow TL

L \longrightarrow [-FFF][+FFF]F

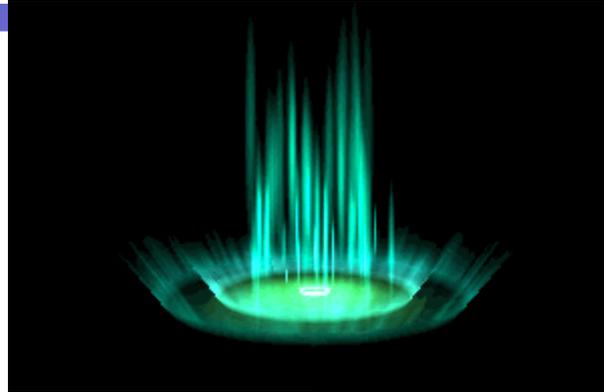
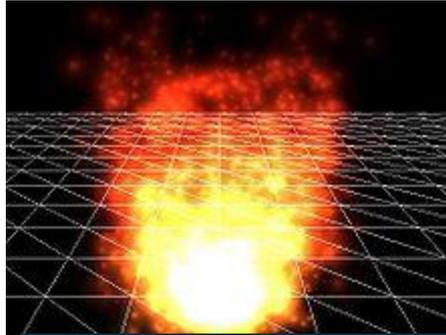
• $\alpha = \pi / 10$



Some deterministic 3D branching plants.



Modélisation procédural: Systèmes de particules



Modélisation procédural:

Systemes de particules

Attributs de la particule : déterministes ou aléatoires

- Ensemble de particules
 - Position, vitesse, masse, paramètres de forme, couleur, transparence
- Déplacement: équations de la dynamique
 - Somme des forces = masse * accélération
- *Générateur* : source de particules
- *Durée de vie* limitée
- Très utiles pour :
 - Poussière, fumée, étincelles, flammes, liquides...

Modélisation procédural:

Systemes de particules

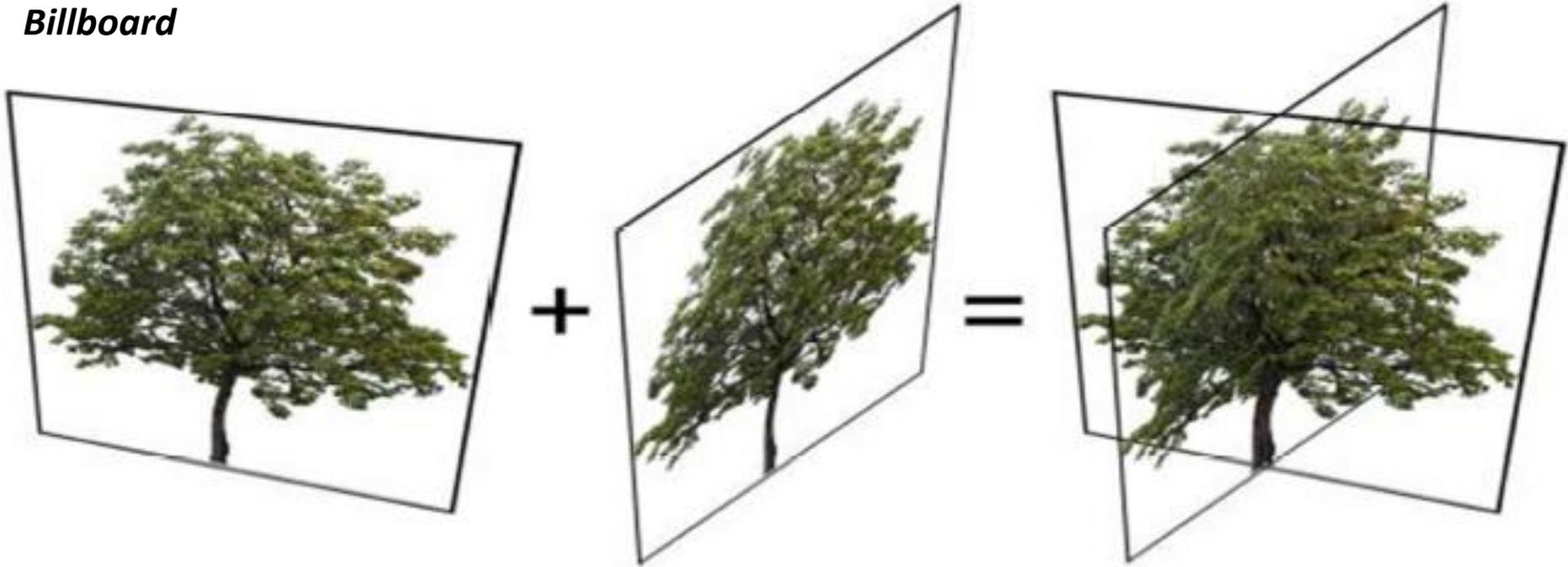
Cycle de vie des particules:

pour chaque plage **temporelle** faire :

- 1. générer N nouvelles particules
- 2. donner des attributs aux nouvelles particules
- 3. détruire les particules dont la durée de vie est passée
- 4. animer les particules restantes : gestion collision+ombre
- 5. les collisions peuvent être destructives,
- 6. dessiner les particules restantes

Modèles à base d'images

Billboard



vue panoramiques

