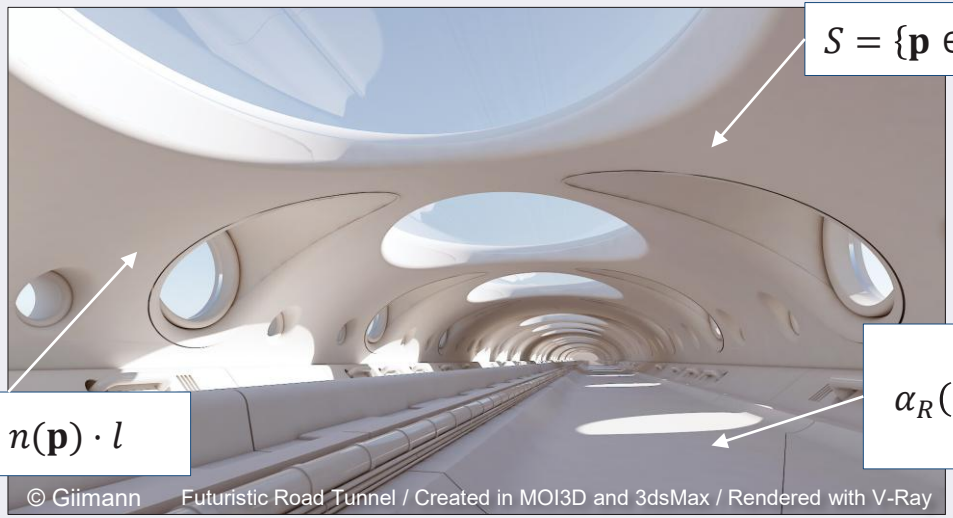
Computer Graphics

From mathematics …



… to the screen

E. Galin
Université Lyon 1

Computer Graphics

Mathematics
Modeling
Color and Texturing
Shading
Realistic Rendering
Acceleration
Animation

Structures accélératrices

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Objectif

Eviter certaines opérations coûteuses, lorsque c'est possible
Les traitements supplémentaires de sélection ou de réjection devront être minimisés

Intersection	Appartenance
$\Delta \cap A = \emptyset \ ? \ A \cap B = \emptyset \ ?$	$\mathbf{p} \in A \ ? \ A \subset B \ ?$

Rendu : lancer de rayon Animation : détection de collision

Typologie

Subdivision de l'espace, ou classification d'objet

Spatial partitioning	**Object partitioning**
Régions disjointes	Objets disjoints
Octrees, **B**inary **S**pace **P**artitioning, grilles	Hiérarchies de volumes englobants

Analyse

Réduire le coût de nombreuses requêtes

Optimisation d'algorithmes	Hiérarchie de la scène

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Computer Graphics

Low level optimization

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Exemple : rotation

Algorithme nécessitant de trouver k point sur le cercle trigonométrique

```
For k ∈ [0, n − 1]
    Let θ = 2π k/n
    Compute point p(cos θ , sin θ)
    Do something …
```

$2n$ appels à /

$2n$ appels à sin ou cos

$c \approx 25n$

Optimisation par le calcul de la matrice de rotation $\mathbf{R}$ d'angle $2\pi/n$

$$\mathbf{R} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

2 appels à sin ou cos

```
Initialize R(2π/n) and set p(1,0)
For k ∈ [0, n − 1]
    Do something …
    Incremental rotation p = R p
```

$4n \times$ et $2n +$

$c \approx 20 + 6n$

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Exemple : intersection

Sphère S de centre $\mathbf{c}$ et rayon r
Equation paramétrique $\mathbf{p}(t)$ du rayon Δ
Résoudre l'équation de second degré $f \circ \delta(t) = 0$

$$S = \{\mathbf{p} \in \mathbf{R}^3, f(\mathbf{p}) = (\mathbf{p}-\mathbf{c})^2 - r^2 = 0\}$$

$$\Delta = \{\mathbf{p}(t) = \mathbf{o} + \mathbf{d}t, t \in \mathbf{R}^+\}$$

Direction normalisée $|\mathbf{d}| = 1$

$$f \circ \mathbf{p}(t) = \mathbf{d}^2 t^2 + 2(\mathbf{o}-\mathbf{c}) \cdot \mathbf{d}\, t + (\mathbf{o}-\mathbf{c})^2 - r^2 = 0$$

$c \approx 21$
$c \approx 34$

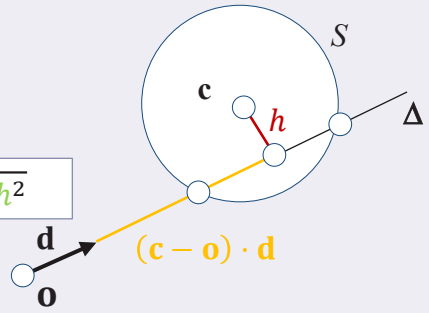
Compute coefficients $at^2 + bt + c = 0$
Compute $\delta = b^2 - 4ac$
If $\delta < 0$ no intersection
Otherwise roots t = $(-b \pm \sqrt{\delta})/2a$

1 appel à /

Méthode géométrique : calcul de $h^2 = \mathrm{d}(\mathbf{S}, \Delta)^2$ et comparaison à r^2
Moins de calculs, notamment avant le test

$$h^2 = (\mathbf{c}-\mathbf{o})^2 - \big((\mathbf{c}-\mathbf{o}) \cdot \mathbf{d}\big)^2$$

$$t = (\mathbf{c}-\mathbf{o}) \cdot \mathbf{d} \pm \sqrt{r^2 - h^2}$$



$c \approx 15$
$c \approx 21$

Compute h^2
If $h^2 > r^2$ no intersection
Otherwise $t = (\mathbf{c}-\mathbf{o}) \cdot \mathbf{d} \pm \sqrt{r^2 - h^2}$

Lyon1 Université Claude Bernard — cnrs — LIRIS
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Computer Graphics

Bounding volumes

Volumes englobants

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Principe

Englober une géométrie complexe G par un volume simple V
Précéder les requêtes sur G d'une requête équivalente filtrante sur V

Surcoût faible

Appartenance
Si $\mathbf{p} \notin V$ alors faux
Sinon déterminer si $\mathbf{p} \in G$

Couteux

Lancer de rayon
Si $\Delta \cap V = \emptyset$ alors faux
Sinon calculer $\Delta \cap G$

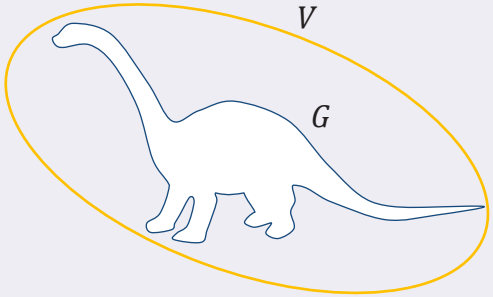
Affichage
Si $E \cap V = \emptyset$ alors faux
Sinon afficher G

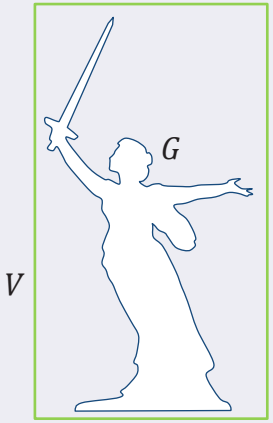
Critères de choix de volume

Mémoire, efficacité des algorithmes
Qualité d'englobement $V \supset G$

$$\rho = v(G)/v(V)$$

Ratio des volumes





eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Fundamentals of Computer Graphics, Steve Marschner, Peter Shirley, ISBN 9780367505035, 2021 by A K Peters/CRC Press

Volumes englobants

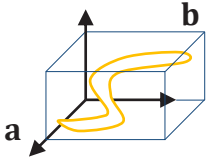
Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Types

Volumes de géométrie simple
Critères de compacité, vitesse de traitement, forme approximant la géométrie

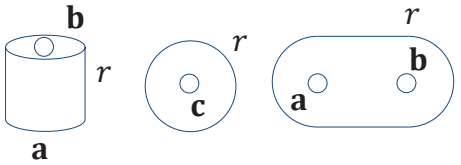
Boites

Compacte
Traitements **rapides**
Parfois **peu** englobant



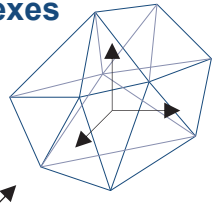
Formes arrondies

Compactes
Moins rapides
Fitting moyen



Convexes

Plus complexe
Moins rapides
Meilleur **fitting**



Arbre à k-dimensions : plans de directions spécifiques

Type	Mémoire	$\Delta \cap V$	$V \cap V$
AABB	Moyen	**Rapide**	**Rapide**
OBB	Coûteux	Coûteux	Coûteux
Sphère	**Faible**	**Rapide**	**Rapide**
Capsule	Moyen	Moyen	Moyen
Convexe	Coûteux	Coûteux	Coûteux

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Plans

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

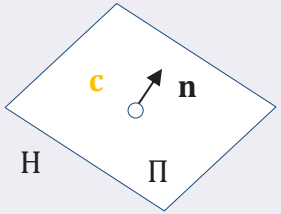
Structure

Un plan Π est caractérisé par une direction $\mathbf{n}$ (unitaire) et un point $\mathbf{c}$

$$\Pi = \{\mathbf{p}, (\mathbf{p} - \mathbf{c}) \cdot \mathbf{n} = 0\}$$

Le demi est espace associé H est :

$$\Pi = \{\mathbf{p}, (\mathbf{p} - \mathbf{c}) \cdot \mathbf{n} < 0\}$$



Slab

Un slab S est l'intersection de deux demi plans de directions opposées $\pm\mathbf{n}$
On peut le définir par exemple un point $\mathbf{c}$, une épaisseur e et une direction $\mathbf{d}$

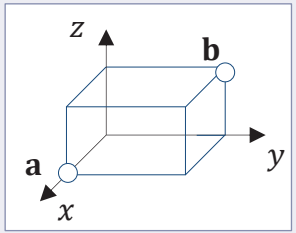
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Boites

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Structure

Boite alignées sur les axes du repère $B(\mathbf{a}, \mathbf{b})$ compacte en mémoire



Requêtes

Appartenance

Appartenance

$$\mathbf{p} \in B \Leftrightarrow x > x_a \wedge x < x_b \wedge \ \ldots$$

Intersection avec une droite

Intersection

$$[t^-, t^+] = (S_x \cap \Delta) \cap (S_y \cap \Delta) \cap (S_z \cap \Delta)$$

Si $t^- > t^+$, alors $B \cap \Delta = \emptyset$

Intersection des intervalles

Intersections de Δ avec les slabs S

Intersection entre deux boites

Intersection

$$B \cap B' = \emptyset \Leftrightarrow x_a > x'_b \vee x_b < x'_a \wedge \cdots$$

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Sphère

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Structure

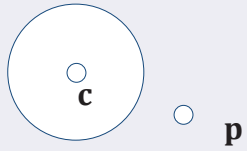
$S(\mathbf{c}, r)$ compact en mémoire ← 4 réels

Requêtes

Appartenance

Appartenance
$\mathbf{p} \in S \Leftrightarrow |\mathbf{p} - \mathbf{c}| < r$

→ Accélération $|\mathbf{p} - \mathbf{c}|^2 < r^2$

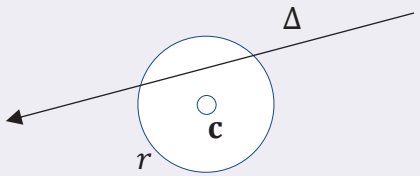


Intersection avec une droite

Intersection
$\Delta \cap S \Leftrightarrow |\delta(t) - \mathbf{c}| = r$

Equation du rayon
$\delta(t) = o + dt$

Equation de second degré en t



Intersection entre deux sphères

Collision
$S_a \cap S_b = \emptyset \Leftrightarrow |\mathbf{c}_a - \mathbf{c}_b| > r_a + r_b$

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Convexes et polytopes

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Convexe

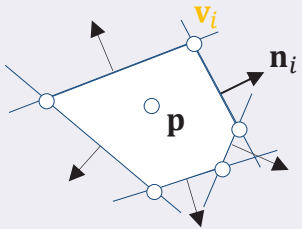
Intersection d'un ensemble de demi plans $\pi_i(\mathbf{v}_i, \mathbf{n}_i)$ de direction $\mathbf{n}_i$ et passant par $\mathbf{v}_i$

$$\mathbf{p} \in C \Leftrightarrow \forall i \in [0, k-1\,](\mathbf{p} - \mathbf{v}_i) \cdot \mathbf{n}_i < 0$$

$\Leftrightarrow \mathbf{p} \cdot \mathbf{n}_i - \mathbf{v}_i \cdot \mathbf{n}_i < 0$: pour éviter de stocker $\mathbf{v}_i$, on peut stocker $d_i = \mathbf{v}_i \cdot \mathbf{n}_i$

Polytopes

Convexe avec k directions choisies
Equivalent à l'intersection de k demi-plans π_i, i $\in [0, k-1]$



Appartenance

$$\mathbf{p} \in P \Leftrightarrow \forall i \in [0, k-1\,]\ \mathbf{p} \cdot \mathbf{n}_i < d_i$$

Simplification du produit scalaire pour $\mathbf{n} \in \{(\pm 1, \pm 1, 0)/\sqrt{2}, (\pm 1, 0, 0), \ldots\}$

Intersection avec une droite

Intersection

$$\Delta \cap P \Leftrightarrow \bigcap_{i=0}^{k-1} (\Delta \cap \pi_i)$$

Lyon1 Université Claude Bernard — cnrs — LIRIS

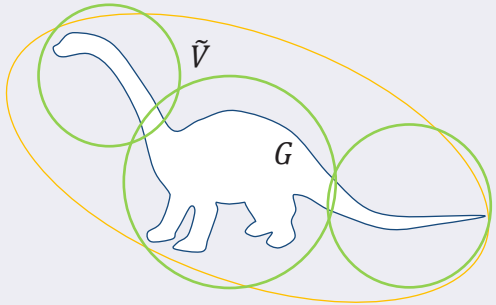
eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

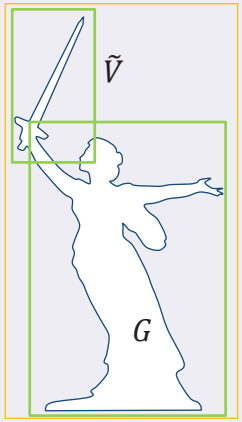
Volumes englobants multiples

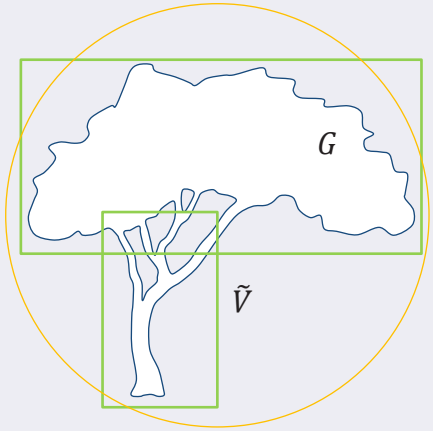
Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Principe

Union d'un petit nombre ($n \approx 2-3$) de volumes englobants $\tilde{V}$ simples
Souvent préférable à un volume complexe







eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Object partitioning

Hiérarchies de boites englobantes

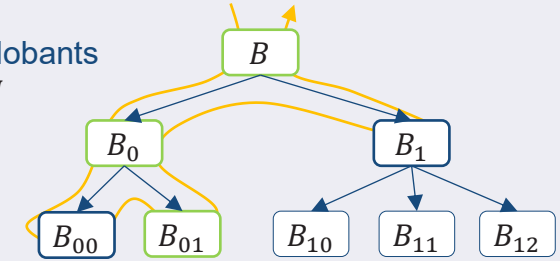
Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Structure

Les volumes englobants ne réduisent pas la complexité !
Une répartition hiérarchique dans un arbre la réduit à $\sim O(\ln n)$ [Marschner2021]

Volumes englobants aux nœuds N

Les feuilles avec objets O



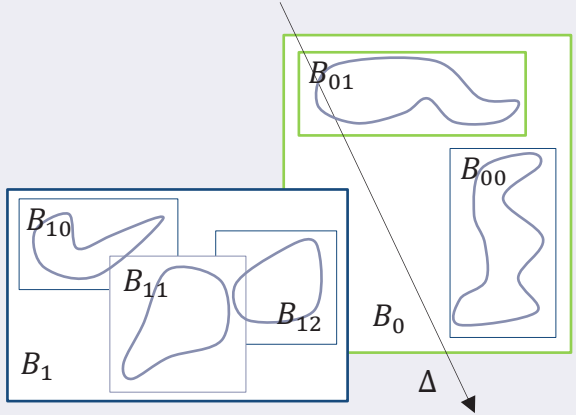
Requêtes

Appartenance

Si $\mathbf{p} \notin B$
  Pas d'appartenance
Sinon
  Si N terminal : test $\mathbf{p} \in O$ pour O de N
  Sinon descente récursive sur N_k

Intersection avec une droite

Si $\Delta \cap B = \emptyset$
  Pas d'intersection
Sinon
  Si N terminal : test $\Delta \cap O$ pour O de N
  Sinon descente récursive sur N_k



eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

Fundamentals of Computer Graphics, Steve Marschner, Peter Shirley, ISBN 9780367505035, 2021 by A K Peters/CRC Press

Construction

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Algorithme

Différentes heuristiques de construction [Wald2007] pour un ensemble d'objets $P = \{P_k\}$

Approche récursive : partant de P, trouver une plan de séparation π, décomposer jusqu'à $\#P = 1$

In general **x**, **y**, **z**

Construction

For all directions **d**
- Sort P along chose direction **d**
- Split P into two sub-sets P^- and P^+
- Compute volumes $V(P^-)$ and $V(P^+)$
- Evaluate the cost c of the split

Keep direction with the least cost c

Recurse with sub-sets P^- and P^+

eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin

I. Wald, S. Boulos, P. Shirley. Ray Tracing Deformable Scenes using Dynamic Bounding Volume Hierarchies. *ACM Transactions on Graphics*, **6**(1), 2007

I. Wald. On fast Construction of SAH based Bounding Volume Hierarchies. *Eurographics / IEEE Symposium on Interactive Ray Tracing*, 2007

Computer Graphics

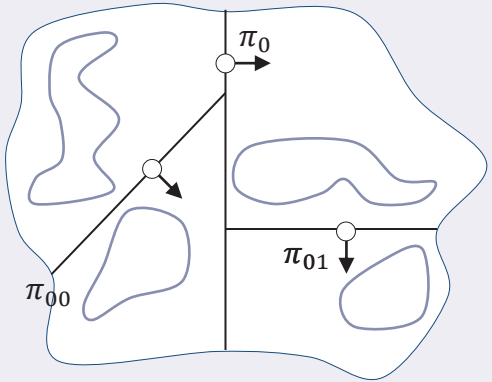
Space decomposition

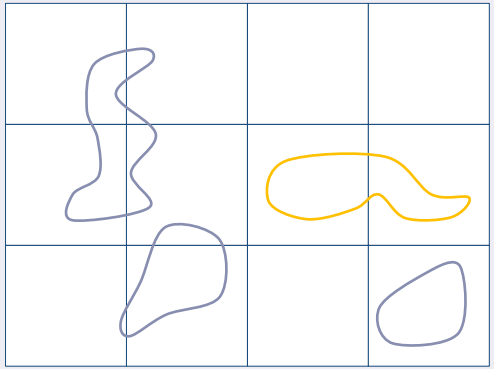
Décomposition spatiale

Overview
Query optimization
Bounding volumes
Object hierarchies
Space decomposition

Structure

Subdivision régulière (grille), adaptative (octree, **B**inary **S**pace **P**artitionning tree, kd tree)





eric.galin@liris.cnrs.fr
http://liris.cnrs.fr/~egalin