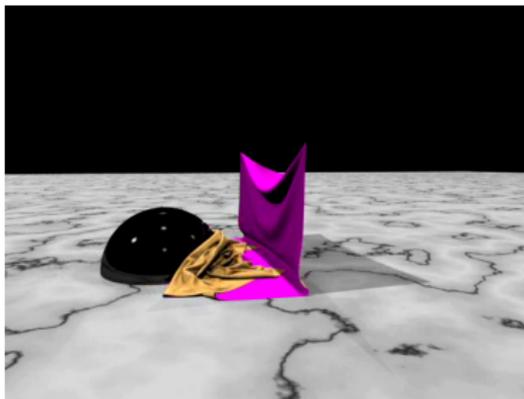


Collisions

SIA Ensimag 3A

Estelle Duveau



Robert Bridson

- **Andrew Witkin et David Baraff - Physically Based Modeling cours à Siggraph 2001**
- Cours de Marie-Paule Cani, François Faure, Nicolas Holzschuch

Plan

- 1 Traitement des collisions
- 2 Exemple 1 : Sphère/Plan
- 3 Exemple 2 : Sphère/Sphère
- 4 Détection des interpénétrations
- 5 Expérimentation

Plan

- 1 Traitement des collisions
- 2 Exemple 1 : Sphère/Plan
- 3 Exemple 2 : Sphère/Sphère
- 4 Détection des interpénétrations
- 5 Expérimentation

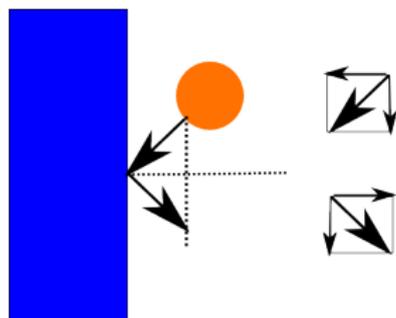
Traitement des collisions

Processus :

- 1 Détection des pénétrations
- 2 Modélisation du contact
- 3 Réponse aux collisions

Réponse à une collision :

- Forces de pénalité
- Impulsion :
 - ▶ Vitesse tangentielle inchangée
 - ▶ Vitesse normale retournée



Plan

- 1 Traitement des collisions
- 2 Exemple 1 : Sphère/Plan
- 3 Exemple 2 : Sphère/Sphère
- 4 Détection des interpénétrations
- 5 Expérimentation

Masse (position q , vitesse v)/Plan (normale n , position p)

- 1 Critère de pénétration :

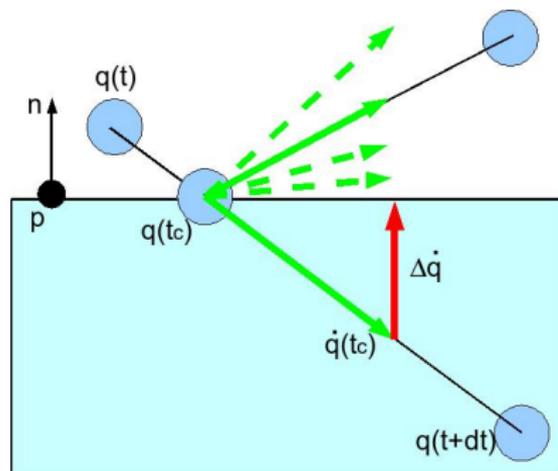
$$pq.n < 0$$

- 2 Retour à la position de collision
- 3 Pour une collision non-élastique, la vitesse est modifiée de

$$\Delta v = -(v.n)n$$

- 4 Application d'un **coefficient de rebond** ϵ :

$$v = v + (1 + \epsilon)\Delta v$$



⇒ **Problème : multiples retours aux positions de collision**

Masse (position q , vitesse v)/plan (normale n , position p)

- 1 Critère de pénétration :

$$pq.n < 0$$

- 2 La position est projetée à la surface : la position est modifiée de

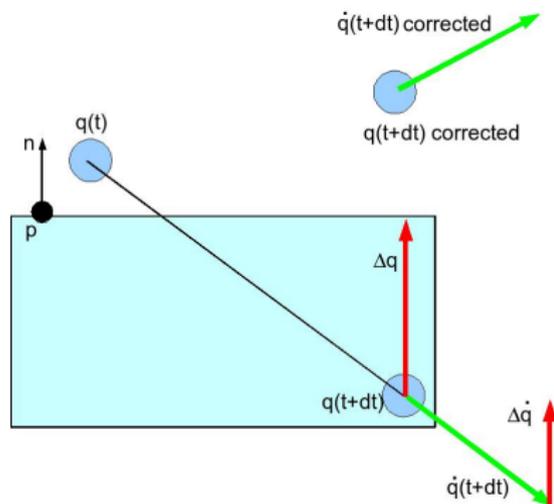
$$\Delta q = -(pq.n)n$$

- 3 Pour une collision non-élastique, la vitesse est modifiée de

$$\Delta v = -(v.n)n$$

- 4 Application d'un coefficient de rebond ϵ :

$$q = q + (1 + \epsilon)\Delta q \quad v = v + (1 + \epsilon)\Delta v$$



⇒ **Toutes les collisions sont traitées en même temps**

Plan

- 1 Traitement des collisions
- 2 Exemple 1 : Sphère/Plan
- 3 Exemple 2 : Sphère/Sphère**
- 4 Détection des interpénétrations
- 5 Expérimentation

Sphère/Sphère -1/2

Sphère i définie par position q_i et rayon r_i

- 1 Critère de pénétration :

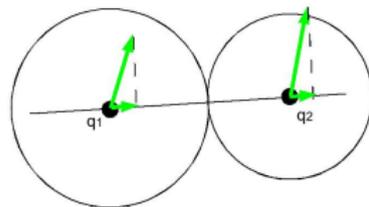
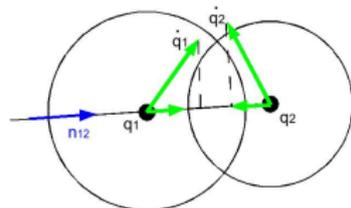
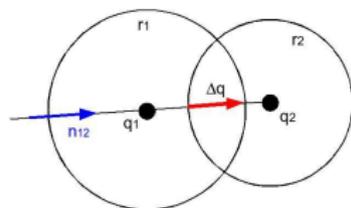
$$\|q_1 q_2\| < r_1 + r_2$$

- 2 Pour une collision non-élastique, leurs positions sont modifiées de

$$\Delta q = (r_1 + r_2 - \|q_1 q_2\|) q_1 q_2$$

et leurs vitesses de :

$$\Delta v = ((v_2 - v_1) \cdot q_1 q_2) q_1 q_2$$



Exemple : Sphère/Sphère -2/2

Sphère i définie par position q_i et rayon r_i

- 3 Utilisation des masses pour conserver le centre de masse :

$$\Delta q_1 = \frac{m_1^{-1}}{m_1^{-1} + m_2^{-1}} \Delta q$$

$$\Delta v_1 = \frac{m_1^{-1}}{m_1^{-1} + m_2^{-1}} \Delta v$$

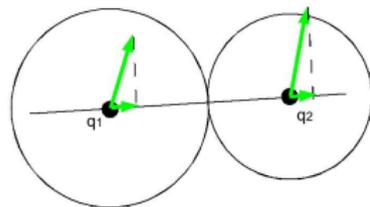
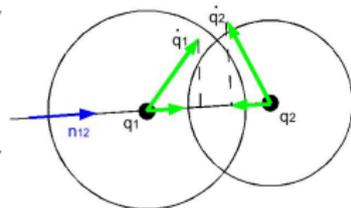
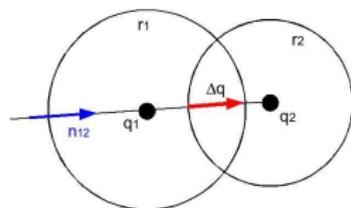
$$\Delta q_2 = \frac{m_2^{-1}}{m_1^{-1} + m_2^{-1}} \Delta q$$

$$\Delta v_2 = \frac{m_2^{-1}}{m_1^{-1} + m_2^{-1}} \Delta v$$

- 4 Application d'un coefficient de rebond ϵ :

$$q_i = q_i + (1 + \epsilon) \Delta q_i$$

$$v_i = v_i + (1 + \epsilon) \Delta v_i$$

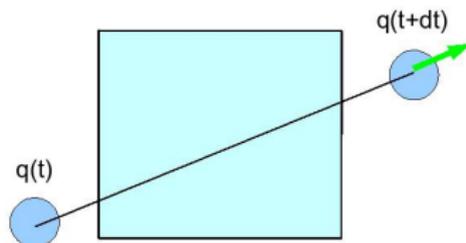


Plan

- 1 Traitement des collisions
- 2 Exemple 1 : Sphère/Plan
- 3 Exemple 2 : Sphère/Sphère
- 4 Détection des interpénétrations**
- 5 Expérimentation

Détection continue

Discrétisation du temps : la détection peut ne pas être détectée

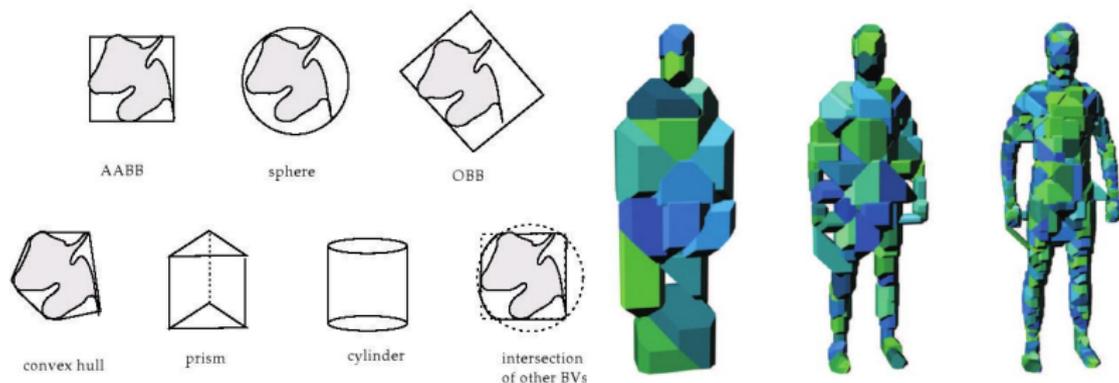


⇒ **Solution exacte :**

- 1 Intersection des trajectoires : résolution d'équations de type intersection point/triangle ou arête/arête pour trouver le temps de collision
- 2 Retour à l'instant de contact

Volumes englobants

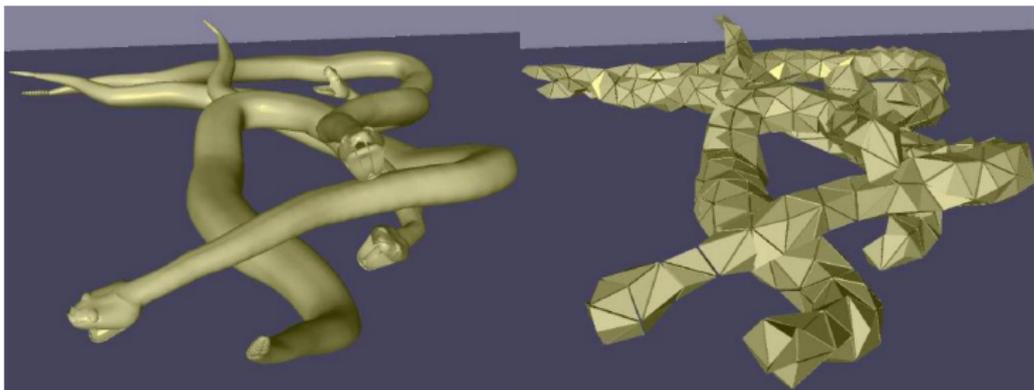
- Non-intersection des volumes englobants \Rightarrow non-intersection des objets
- **Volumes englobants** : boîtes parallèles aux axes (AABB), sphères, boîtes orientées (OBB)...
Tout objet dont les intersections sont rapides à calculer!
- Encore mieux : **hiérarchie de volumes englobants** : plus rapide mais mise-à-jour de la hiérarchie coûteuse pour les objets déformables



Multiresolution

Toujours mieux : plonger une géométrie complexe dans une géométrie plus grossière

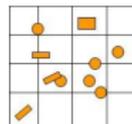
- Application de la simulation (dynamique + collisions) à la géométrie grossière
- Rendu de la géométrie fine



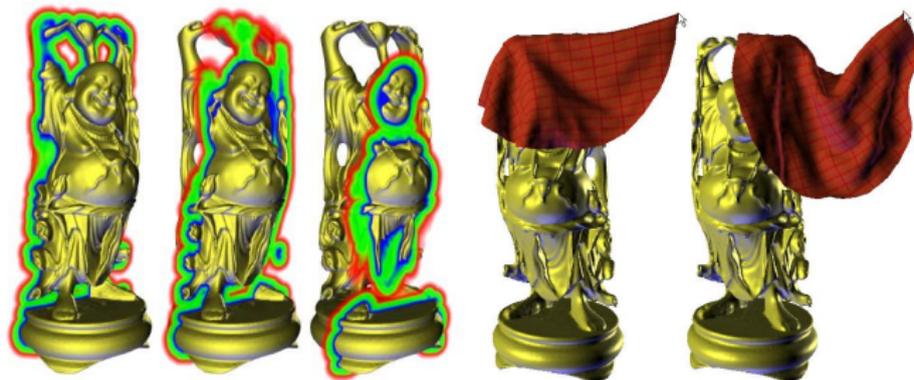
Discretisation de l'espace

- Diminuer le nombre de paires d'objets à tester :

Plonger les objets dans une grille de l'espace : tester les objets d'une même case ou de cases voisines

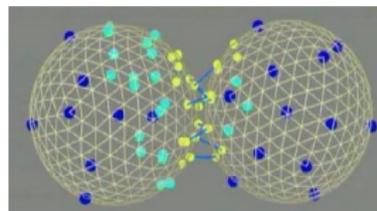
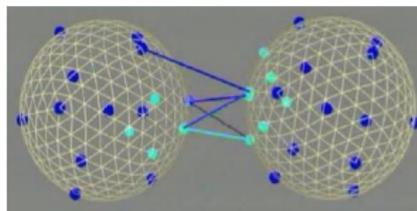
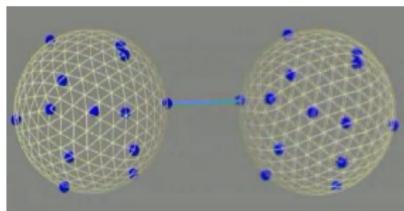


- **Champ de distance** : fonction retournant le point de la surface le plus proche en projetant chaque particule sur la surface



Méthodes stochastiques

- Tester la distance entre paires d'arêtes aléatoires
- Raffiner les tests aux endroits où les objets sont proches



Plan

- 1 Traitement des collisions
- 2 Exemple 1 : Sphère/Plan
- 3 Exemple 2 : Sphère/Sphère
- 4 Détection des interpénétrations
- 5 **Expérimentation**

TP : Collisions

- Dans `animate()` de `Dynamics`, les forces sont calculées, les vitesses et positions sont mises à jour et la fonction `treatCollisions()` est appelée. Comme son nom l'indique, `treatCollisions()` s'occupe du traitement des collisions.
- Pour toutes les masses, `treatCollisions()` traitent les collisions de la masse en question avec les plans (resp. les sphères) définis dans `initDynamics()` et stockés dans le vecteur `planes` (resp. `balls`).
- **Question 1** : Ecrire `collisionMassPlane(int massI, int planeJ, double rebound)` qui traite les collisions entre la masse `massI` et le plan `planeJ` avec le coefficient de rebond `rebound`.
- **Question 2** : Ecrire `collisionMassBall(int massI, int ballJ, double rebound)` qui traite les collisions entre la masse `massI` et la sphère `ballJ` avec le coefficient de rebond `rebound`.