Chapitre 8 : Animation

Modélisation 3D et Synthèse

Fabrice Aubert
fabrice.aubert@lifl.fr

Université Lille1
Sciences et Technologies

IEEA - Master Info - Parcours IVI

2012-2013

1 Introduction

Animation

- Animation = succession d'images.
- Domaine : artistique (liberté de la caméra, des formes), simulation (outils pédagogiques ou aide au geste : chirurgie, assemblage, pilotage), ludique (jeux vidéos).
- Génération à temps différé : pré-calcul des images, puis diffusion de ces images ("film").
  - Pas de contrainte de temps (permet d'obtenir le réalisme nécessaire).
  - Pas d'interaction avec l'animation.
- Génération dynamique (dite abusivement "à temps réel" ou à temps interactif) : chaque image calculée est immédiatement affichée.
  - Contrainte de temps : 10 images/s pour la perception d'une continuité, 25/30 images pour un confort visuel.
  - Interaction possible (simulation avec intervention humaine).
- Temps réel : l'image calculée avec un paramètre t correspondant au "vrai" temps t (pas nécessairement à une fréquence élevée...).

Exemple d'objets animés

- Mouvement de caméra : interaction sur l'orientation, la position, interpolation selon plusieurs positions clefs, suivi d'une courbe
- Mouvement de personnage : défini à la "main" (interpolation), motion-capture, génération automatique de marche.
- Mouvement rigide d'objets : soumis à un poids, chocs $\Rightarrow$ réalisme physique.
- Déformation d'objets : déformation géométrique, simuler des objets "mous".
- Déformation topologique : briser les objets.
- Fluides : simulation des mouvement d'eau.

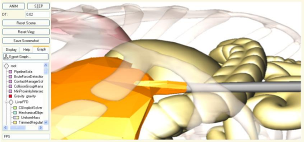
2 Techniques d'animation

Descriptive

- L'animation est décrite explicitement (scénario, scripts, ...) en utilisant des outils (principalement) géométriques.
- Exemples : courbe b-spline pour la trajectoire de caméra (avec contrôle de vitesse), la trajectoire des objets, fonctions mathématiques de déformation des objets, outils d'expressions des visages, d'aide à la conception de la marche, etc...
- Outre ces outils, la technique maîtresse du descriptif est l'interpolation : l'animateur donne des positions clés, et le logiciel d'animation doit générer les positions intermédiaires pour obtenir du 25 images/seconde.
- $\Rightarrow$ appelée "key-framing" (interpoler les paramètres entre 2 positions clefs).
- L'interpolation peut être linéaire (souvent insuffisante) ou ... plus complexe (interpolation spline).
- On peut mettre dans le descriptif des outils tels que : le morphing (technique d'interpolation), le motion capture (acquisition de données).
- Avantage : liberté "totale" dans la conception des animations.
- On oppose généralement animation descriptive à animation basée physique.

Animation basée physique

- Faire suivre aux objets les lois mécaniques.
- Avantage : réalisme, le mouvement est généré sans intervention humaine (seules les positions initiales des objets et les paramètres des lois sont fixés).
- Inconvénient : corps déformables avec rupture difficilement accessible au temps interactif...

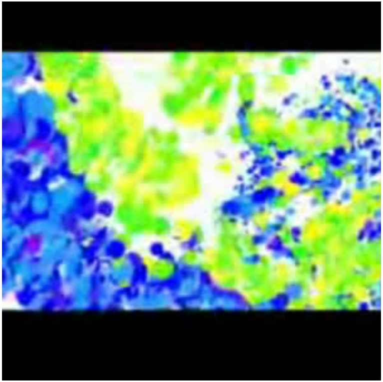


Comportementale

- Chaque entité de l'animation répond à des stimulis qui vont définir son comportement lors de l'animation.
- Exemple : simulation d'un banc de poisson (chaque poisson répond principalement selon des critères de proximité), vol d'oiseau.

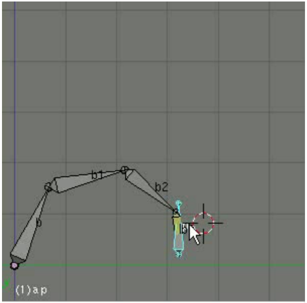
Particule

- Uniquement basé sur l'animation de points qui peuvent subir des lois physiques (simples) et souvent accompagnés de lois stochastiques de mouvement.
- Exemples : animation de feu et chute d'eau.



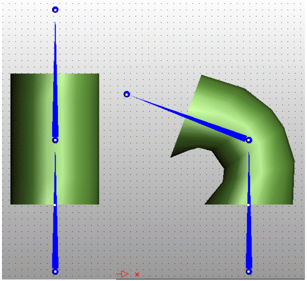
Animation par contraintes

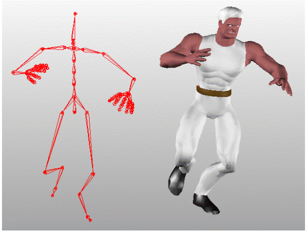
- On fixe un objectif à un objet soumis à des contraintes (géométrique, définition des degrés de liberté, obstacles).
- Exemples :
  - cinématique inverse pour un bras articulé (objectif : main qui saisit une balle ; contraintes : position des articulations fixées par les liaisons).
  - path-finding : robot muni de capteurs dans un labyrinthe qui doit trouver la sortie.



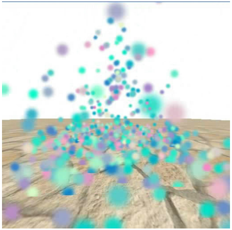
Habillage

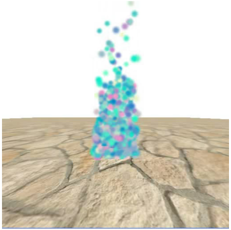
- On se contente d'animer un "squelette" d'un objet (ensemble de points, ou de segments inclus dans l'objet). L'animation de l'habillage (souvent une surface géométrique) est déduit de l'animation du squelette.
- Exemple : "skinning" pour les personnages (par exemple : surface implicite pour la peau).

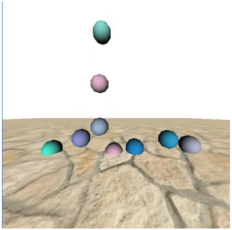




3 Animation basée Physique







Mécanique du point

- La scène à animer est constituée d'un ensemble de points (ou particules).
- Problème : trouver la position x des points à chaque instant t.
- Vitesse à l'instant $t \Rightarrow \frac{\partial x}{\partial t}(t) = v(t)$ (dérivée de la position par rapport au temps).
- Notation : les dérivées temporelles sont notées avec un point $\frac{\partial x}{\partial t}t = \dot{x}(t)$
- Accélération à l'instant t : $a(t) = \dot{v}(t)$ (dérivée de la vitesse)
- Autrement dit : $a(t) = \ddot{x}(t)$ (dérivée seconde de la position).

Exemple avec accélération constante

- $a(t) = a$ à tout instant (accélération constante) :
- Comme $\dot{v}(t) = a(t) = a$, on trouve $v(t)$ en trouvant la primitive de a : $v(t) = at + Cte$.
- $Cte = v_0$: vitesse initiale à l'instant $t = 0$ (i.e. en début de simulation) qui doit être donnée.
- $\dot{x}(t) = v(t) = at + v_0$. La position x est donnée par la primitive de $at + v_0 \Rightarrow$
  $x(t) = \frac{1}{2}at^2 + v_0 t + x_0$

position à l'instant t =0

Relation Fondamentale de la Dynamique (RFD)

- le mouvement est généré par des forces (poids, attraction, frottement, liaison ressort, contact, ...).
- on note $F_A(t)$ l'ensemble des forces appliquées au point A à l'instant t.
- on appelle quantité de mouvement du point A la grandeur $p_A(t) = m_A v_A(t)$ (m_A : masse, $v_A(t)$: la vitesse).
- on calcule alors l'accélération du point A grâce à la Relation Fondamentale de la Dynamique (RFD) :

$$F_A(t) = \dot{p_A}(t) \text{ où } \dot{p_A}(t) = \frac{\partial m_A v_A}{\partial t}(t) = m_A a_A(t) = m_A \ddot{x_A}(t)$$

- On connait donc l'accélération :

$$\ddot{x_A}(t) = \frac{F_A(t)}{m_A}$$

deriver second de la position

- On trouve la position de A en résolvant $\ddot{x_A}(t) = \frac{F(t)}{m}$ (équation différentielle).
- Dans la suite on note les formules sans préciser t (ne pas oublier que les valeurs dépendent de t).

Intégration du mouvement

- Trouver x à partir de $\ddot{x} = \frac{F}{m}$ s'appelle « intégrer le mouvement »(résolution d'une équation différentielle).
- Cette équation admet difficilement une solution formelle (sauf dans des cas très particuliers : chute d'un corps, pendule, masse suspendue à un ressort,...).
- $\Rightarrow$ adoption de méthodes de résolutions approchées et itérative.
- Exemple d'une méthode d'intégration simple : Euler explicite.

Euler explicite

- On suppose connue x_t et v_t à l'instant t.
- A l'instant 0, les positions x_0 et les vitesses v_0 initiales sont données.
- On cherche la nouvelle position $x_{t+\delta t}$ (δt est appelé le pas de temps de la simulation).
- On effectue alors les approximations suivantes sur les dérivées : $a_t = \dot{v}_t = \frac{v_{t+\delta t} - v_t}{\delta t} = (\frac{F_t}{m}$ d'après RFD) et $v_t = \dot{x}_t = \frac{x_{t+\delta t} - x_t}{\delta t}$
- Ce qui donne, avec $a_t = \frac{F(t)}{m}$:

$$\boxed{\begin{aligned} v_{t+\delta t} &= v_t + \frac{F}{m}\delta t \\ x_{t+\delta t} &= x_t + v_t\delta t \end{aligned}}$$

1

- Problème intuitif : si le pas δt est trop grand, le mouvement est faux par rapport à la RFD. Si le pas est trop petit, on a des erreurs numériques qui donnent également un mouvement faux. Le choix d'un « bon »pas de temps est conditionné par le rapport $\frac{F}{m}$.
- $\Rightarrow$ étude des problèmes de convergences et de stabilité des méthodes d'intégration
- Nombreuses autres méthodes d'intégration : Runge-Kutta (tenir compte de plusieurs instants pour faire une meilleure approximation), méthodes dites implicites (couteuses, mais stables) $\Rightarrow$ voir le domaine de l'analyse numérique.

Boucle de simulation

Principe :

position et vitesse pour chaque objet a gérer

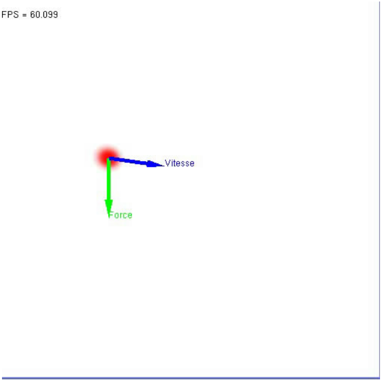
```
t=0
e_t=(x_t,v_t)=(x_0,v_0); // e_t = vecteur d'etat
Tant que simulation non finie faire
  t=t+delta_t
  Résoudre collision // correction/génération de forces
  Calculer F_t/m
  e_t=e_t+delta_t*(F_t/m,v_t) // Euler explicite
  Afficher // découplage souhaitable (multi-threading).
fin tant que
```

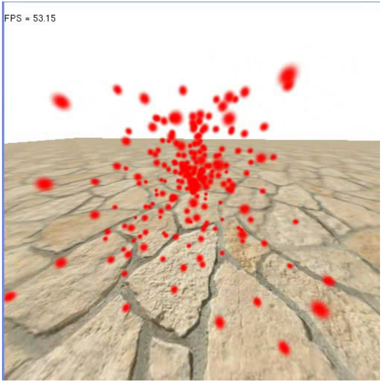
force

voir 1 slide précédant

Exemple

- Une force unique : le poids mg avec $g = 9.81$ et m la masse du point (donnée).





Collision

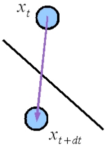
- Détecter la collision :
  - Détection statique : déterminer si à l'instant t les objets sont en intersection. Principal problème : risque de traverser sans détecter.
  - Détection dynamique : intersection de trajectoire (intersection entre les positions à t et à $t + dt$) pour déterminer l'instant de collision. Principal problème : peut s'avérer très délicat et/ou couteux.
- Répondre à la collision simplement ?
  - Corriger les vitesses/positions (impulsion)
  - Insérer des forces de répulsion (méthode dite à pénalité)
  - $\Rightarrow$ difficile de conserver un raisonnement "physiquement" correct (réalisme).

Autre approche possible : résolution d'un problème sous contrainte (trouver les positions en leur interdisant des zones de l'espace) $\Rightarrow$ plus complexe à mettre en oeuvre.

Collision point/plan infini

Détection statique : le point traverse t'il le plan ?

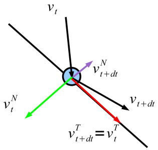
- Plan donné par (A, n). Particule en mouvement donnée par sa position P_t.
  - Instant t : calcul de $\overrightarrow{AP_t} \cdot n$ (position relative de la particule au plan).
  - Instant $t + dt$: calcul de $\overrightarrow{AP_{t+dt}} \cdot n$
  - Si changement de signe $\Rightarrow$ la particule a traversé le plan.
  - Pour être physiquement correct, il faudrait trouver l'instant du contact (pour avoir la vitesse et la position au moment du contact) $\Rightarrow$ trouver l'instant de collision.
  - $\Rightarrow$ dans la suite on se contente de réagir en prenant la position et la vitesse à l'instant $t + dt$.

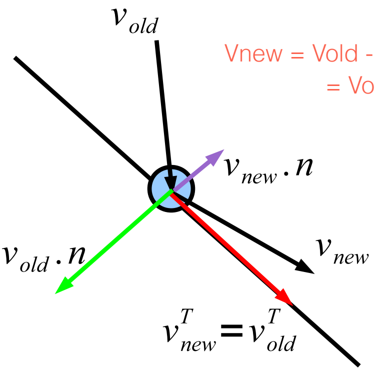


Collision point/plan infini

Réponse : impulsion.

- Changement de la vitesse spontanément (principe d'impulsion instantanée) :
  - On décompose la vitesse v en vitesse tangentielle v^T et normale v^N (i.e. $v = v^T + v^N$ et $v^N = (v \cdot n)n$ si n normé).
  - La vitesse tangentielle au contact reste inchangée.
  - La vitesse normale est opposée, en lui appliquant un coefficient de restitution $\varepsilon \in [0,1]$ (traduit le rebond).





vert violet

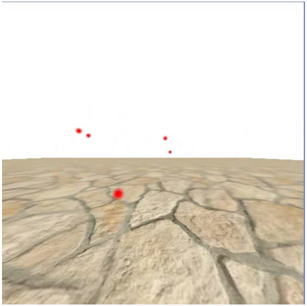
Vnew = Vold - (Vold.n)n - Ɛ(Vold.n)n
= Vold - [(1-Ɛ)Vold.n]n

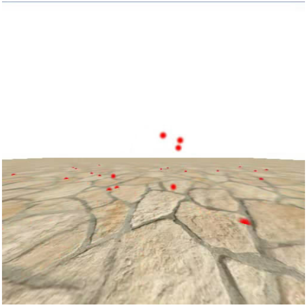
- Soient v_{old} et v_{new} les vitesses avant et après la collision de la particule sur un objet **statique**.

coef de restitution

- Exercice : connaissant v_{old}, n et ε quelle est la vitesse finale v_{new} ?

Exemple

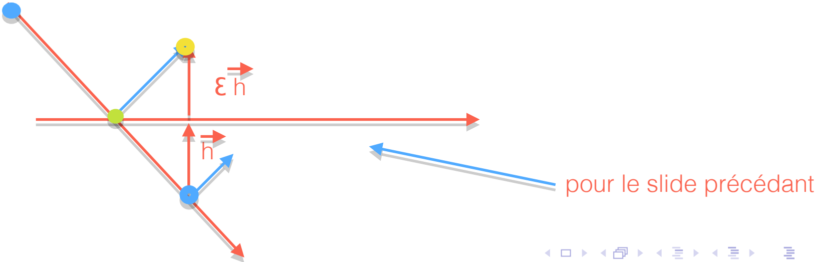




- Il faut également corriger la position.
- Par impulsion : $x_{new} = x_{old} + (1+\varepsilon)(H - x_{old})$ avec H projection de x_{old} sur le plan.

Exemples de forces sur les particules

- Amortissement/frottement : $F = -kv$.
- Ressort entre 2 particules x_1 et x_2 : $F_{1\to 2} = -k(x_2 - x_1 - l_{repos})$
- Force d'attraction gravitationnelle entre 2 particules : $F_{1\to 2} = -G\frac{m_1 m_2}{\|x_2 - x_1\|}(x_2 - x_1)$



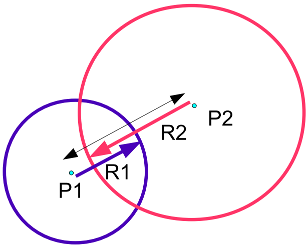
Autres collisions

- Collision point/sphère (localiser la particule par rapport à une sphère).
- Collision point/triangle (plan infini puis localisation de l'intersection à l'intérieur).
- Collision point/volume (intersection demi-droite avec bord).
- Collision point/bsp

Animation basée physique avec des sphères

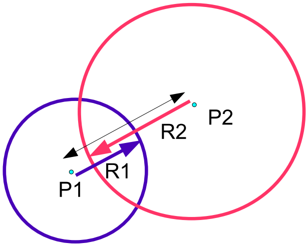
- position/vitesse = position/vitesse du centre
- la sphère considérée comme un point (application de la physique à son centre).
- On applique les même calculs que pour les particules
- **Mais** on gère la collision entre les sphères (prise en compte d'un rayon donné).
- Ce qui change ?
  - détection des collisions entre toutes les sphères deux à deux.
  - calcul de l'impulsion lors d'une collision entre 2 sphères.
- Remarque : on ne gère pas le roulement.

Collisions sphère/sphère



- Détection : $(p_2.position() - p_1.position()).length() < p_2.radius() + p_1.radius()$
- Réponse par impulsion : correction des vitesses :
  - On calcule un vecteur impulsion j.
  - Soit N le vecteur normal au point de contact : on peut prendre (approximation) le vecteur joignant les deux centres.
  - $v_{1\,new} = v_{1\,old} - \frac{j}{m_1} N$ et $v_{2\,new} = v2_{old} + \frac{j}{m_2} N$ avec j le coefficient d'impulsion.
  - On calcule que $j = \frac{-(1+\varepsilon)(v_{2\,old} - v_{1\,old}) \cdot N}{(\frac{1}{m_1} + \frac{1}{m_2}) N \cdot N}$

Collisions sphère/sphère : correction des positions



- Eloigner les sphères l'une de l'autre dans la direction $N = p_2.position() - p_1.position()$
- Distance de recouvrement des 2 sphères : $D = (p_2.position() - p_1.position()).length() - p_1.radius() - p_2.radius()$ (négative si en collision).
- Tenir compte des masses m_1 et m_2 des sphères pour distribuer le déplacement D entre p_1 et p_2.

$$p_{1\,new} = p_{1\,old} + (1+\varepsilon) * \frac{m_2}{m_1+m_2} D \frac{N}{N \cdot N}$$
$$p_{2\,new} = p_{2\,old} - (1+\varepsilon) * \frac{m_1}{m_1+m_2} D \frac{N}{N \cdot N}$$

Problèmes/Autre utilisation des sphères

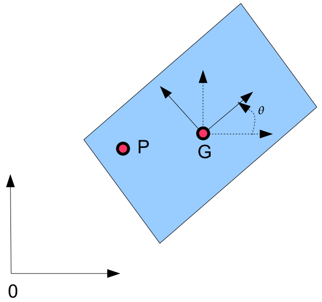
- La résolution de la collision entre 2 sphères peut regénérer des collisions avec les autres sphères (problème des multi collisions : très délicat à résoudre en statique).
- Utilisation possible des sphères : modèle de collision.
- Souvent utilisées avec un principe de pénalité comme réponse.
- Si nombreuses sphères $\Rightarrow$ nécessité d'optimiser pour éviter les tests entre toutes les paires de sphères.

Animation basée physique avec des boites (cas 2D)

Mécanique des solides rigides

- Pour une particule :
  - trouver la position $x(t)$:
  - donner/calculer la somme des forces F qui s'appliquent à la particule.
  - puis résoudre la RFD avec intégration d'Euler par exemple :
    - $v_{new} = v_{old} + \frac{F}{m} dt$
    - $x_{new} = x_{old} + v_{old} dt$
- Pour un solide :
  - trouver la position $x(t)$ **ET** trouver l'orientation $\theta(t)$ (rappel : le placement de tout objet 3d se réduit à une composition *TR*).
  - Pour la position $x(t)$: on peut appliquer la RFD au centre de masse G (i.e. considérer l'objet comme un point réduit à son iso-barycentre).
  - Pour l'orientation $\theta(t)$: il faut faire intervenir d'autres notions mécaniques pour la calculer $\Rightarrow$ moment des forces, moment cinétique, moment d'inertie.
  - $\Rightarrow$ pour simplifier, on ne considère que le cas 2D dans la suite.

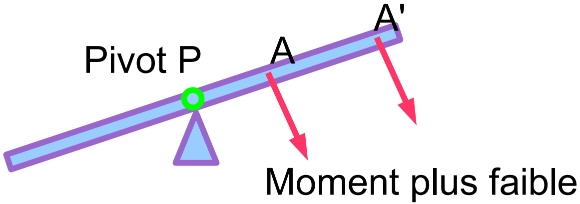
Vitesse angulaire



- La position d'un solide à l'instant t est définie par la position x de son centre de masse G, et de sa rotation θ autour de ce centre.
- On définit la vitesse angulaire par $\omega = \dot{\theta}$
- On définit le vecteur vitesse angulaire par $\vec{\omega} = \omega \vec{u}$ où u est l'axe de rotation (dépend de l'instant t).
- Vitesse d'un point P quelconque du solide : $v_P = v_G + \overrightarrow{PG} \times \vec{\omega}$
- Comment trouver l'angle $\theta(t)$? $\Rightarrow$ on applique le théorème du moment cinétique.

Moment d'une force

- Moment d'une force (analogie avec la force pour la position) :
  - c'est l'aptitude d'une force appliquée en A à faire tourner un solide autour d'un pivot P donné.
  - $M_{F/P} = F \times AP$ (produit vectoriel). En 2D (i.e. $z = 0$) : une seule composante (le z du produit vectoriel est le déterminant).



Moment d'inertie et moment cinétique

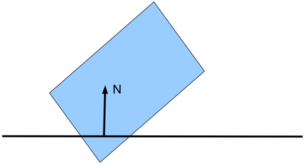
- Moment d'inertie (analogie avec la masse) en 2D :
  - Par rapport à un point P : $I_P = \int\int_{A \in S} (AP)^2 dm$
  - On choisit généralement le centre de masse $P = G$.
  - Exemple : rectangle de masse totale M et de dimension (l, h) : $I_G = \frac{1}{12} M(l^2 + h^2)$.
- Moment cinétique d'un point A du solide (analogie avec quantité de mouvement) en 2D :
  - Par rapport à un point P : $L_{A/P} = p \times AP$ (où p : quantité de mouvement = mv).
  - On montre que $L_{A/G} = I_G \omega$ (i.e. indépendant de A).

Théorème du moment cinétique

- $M_{F/G} = \frac{\partial L_{A/G}}{\partial t} = I_G \dot{\omega}$ (analogie avec $F = ma$).
- On a donc une relation pour trouver l'angle à partir de l'accélération angulaire : $\dot{\omega} = \frac{M_{F/G}}{I_G}$
- Application d'Euler :

$$\begin{aligned}
\overrightarrow{\omega}_{new} &= \overrightarrow{\omega}_{old} + \frac{M_{F/G}}{I_G} dt \\
\theta_{new} &= \theta_{old} + \omega_{old} dt
\end{aligned}$$

Collision Polygone/Plan infini



- Déterminer le sommet le plus loin.
- Appliquer une impulsion au point de contact P et considérer la normale N au contact.
- L'impulsion induit un changement de vitesse à P, qu'on doit répercuter sur v_G et sur $\overrightarrow{\omega}$.
- Rappel : $v_P = v_G + \overrightarrow{PG} \times \overrightarrow{\omega}$
- Le calcul de la correction est :

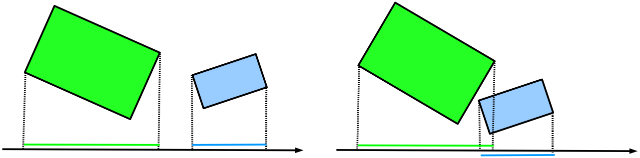
$$v_{G_{new}} = v_{G_{old}} + \frac{j}{m} N$$
$$\overrightarrow{\omega}_{new} = \overrightarrow{\omega}_{old} + (\overrightarrow{PG} \times jN)$$

Avec

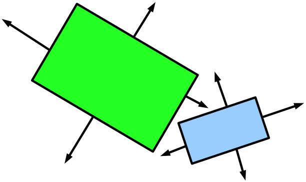
$$j = \frac{-(1+\varepsilon) v_{P_{old}} \cdot N}{\frac{1}{m} + \frac{N \cdot ((PG \times N) \times PG)}{I_G}}$$

- v_G est la vitesse du centre de masse, v_P est la vitesse du point au contact.

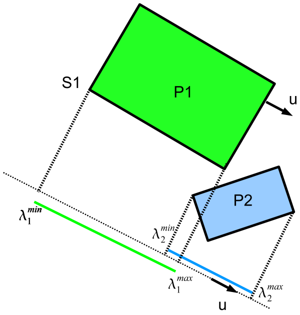
Détections de collision entre polygones



- Détection par axe séparateur pour des polygones $P1$ et $P2$ **convexes** :
  - Un vecteur est dit axe séparateur si les projections des deux polygones sur ce vecteur sont disjointes.
  - S'il existe un axe séparateur alors les polygones sont disjoints.
  - Pour deux polygones (convexes), s'il existe un axe séparateur, alors un axe orthogonal à une arête des polygones est nécessairement séparateur.
  - $\Rightarrow$ recherche d'un axe séparateur pour la détection de collision.



Test d'axe séparateur



- Projeter sur l'axe u ? $\Rightarrow$ projeter tous les sommets S sur u : $\lambda = P \cdot u$.
- Disjoints ? Comparez les λ : $[\lambda_1^{min}, \lambda_1^{max}]$ intersecte $[\lambda_2^{min}, \lambda_2^{max}]$?
- On appelle distance de séparation sur l'axe la distance de superposition (sur l'exemple : $\lambda_1^{max} - \lambda_2^{min}$).
- Remarques :
  - raisonnement valide pour tout polygone convexe (nombre quelconque de sommets)
  - se généralise en 3D (les axes à tester sont les normales aux faces, et les normales aux couples d'arêtes de $P1$ et $P2$).

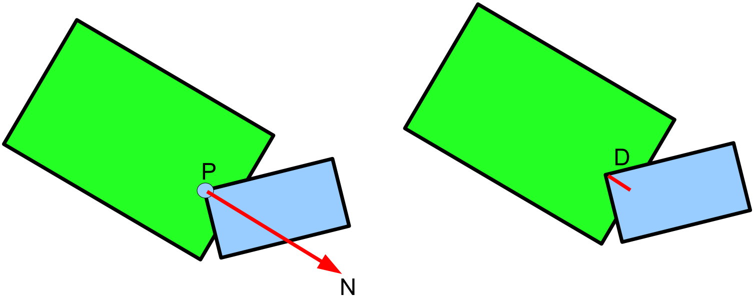
Réponse à la collision

- Par impulsion au point de contact P, avec un contact de normal N (normalisé).
- Le polygone 1 subit : $v^1_{Gnew} = v^1_{Gold} + \frac{j}{m^1} N$ et $\overrightarrow{\omega^1}_{new} = \overrightarrow{\omega^1}_{old} + (\overrightarrow{PG^1} \times jN)$ (similaire pour 2 mais avec $-j$).

$$j = \frac{-(1+\varepsilon)(v^1_{Pold} - v^2_{Pold}) \cdot N}{\frac{1}{m^1} + \frac{1}{m^2} + \frac{N \cdot ((P^1G^1 \times N) \times P^1G^1)}{I^1_G} + \frac{N \cdot ((P^2G^2 \times N) \times P^2G^2)}{I^2_G}}$$

- Remarque : la vitesse de P n'est pas la même pour 1 et 2.
- Problème : P ? N ?

Réponse à la collision



- Plusieurs solutions, toutes plus ou moins... incorrectes.
- Rappel : il faudrait répondre au contact, et non à une situation d'intersection ! (trouver l'instant t qui précède la collsion, puis trouver les points les plus proches).
- Proposition : prendre l'axe (de l'algorithme de détection) qui génère la plus **petite** superposition comme normale, et prendre les sommets en intersection comme points de contacts (en tp : moyenne de tous les sommets inclus).
- Comme on répond à une situation en intersection $\Rightarrow$ correction de la position :
  - $G^1{}_{new} = G^1{}_{old} + \frac{m^1}{m^1+m^2} DN$ et $G^2{}_{new} = G^2{}_{old} - \frac{m^2}{m^1+m^2} DN$ (moyenne pondérée par les masses).

Accélération de la détection de collisions

- Généralement décomposer en 2 phases : narrow-phase (détection à proprement parler) et broad-phase (accélération).
- Narrow-Phase :
  - détection "précise" entre 2 corps (sphère/sphère, polygone/polygone, objet/objet, sphère/polygone, AABB/..., etc).
  - Principaux problèmes : algorithme géométrique d'intersection, mais surtout fournir les informations nécessaires à la réponse choisie (point de contact, normale au contact, distance de superposition, etc).
- Broad-Phase :
  - déterminer des critères entre tous les corps de la scène pour éviter la narrow-phase.
  - Exemples :
    - tests d'intersections sur boites englobantes (sphère, AABB, OBB, etc).
    - axe séparateur sur l'axe x de la scène (projection de tous les objets de la scène).
    - subdivision de la scène (grille régulière, octree dynamique).

Conclusion sur l'animation physique

Référence web : http ://www.cs.cmu.edu/afs/cs/user/baraff/www/pbm/pbm.html (Witkin/Baraff/Kass 97).
Autre : "Game Physics", David Eberly.

- gestion du contact (stabilité), contacts/collisions multiples
- gestion des contraintes (par exemple rigides articulés).
- modèles par contraintes (coordonnées généralisées).
- En 3D : pas nécessairement plus compliqué, mais plus lourd : algorithmes de collisions 3D ; angle de rotation qui devient axe de rotation ; moment d'inertie qui devient une matrice d'inertie.
- modèles déformables (approximation avec ressorts ; éléments finis).
- modèles de rupture.