

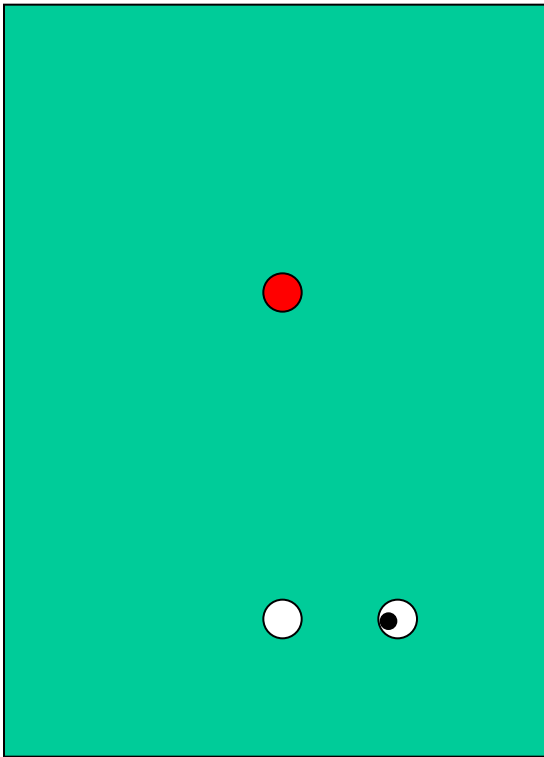
LO19 : Génie Logiciel :

De l'expression des besoins à la réalisation

**Projet : Modélisation objet avec UML d'un jeu de billard
français**

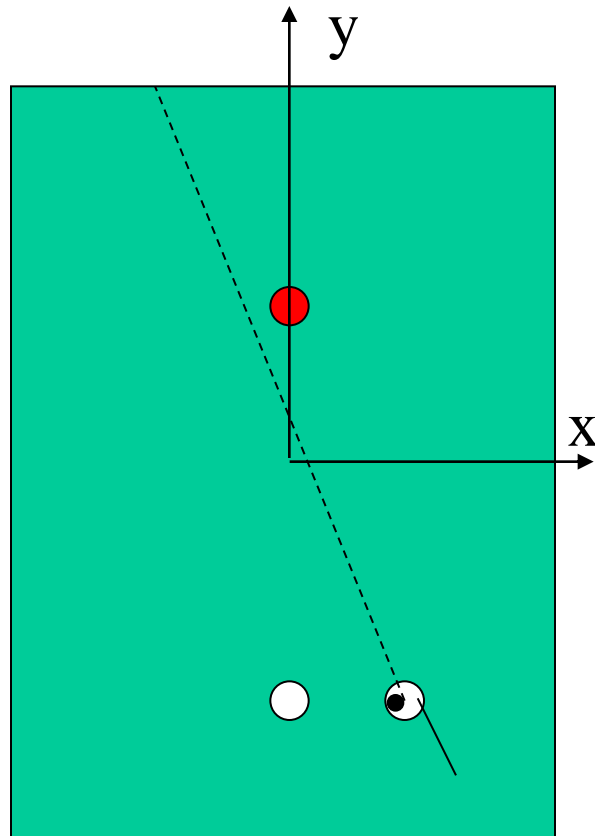
Walter SCHÖN

Configuration du jeu



- Tapis sans trous ni obstacles, dimensions 2845 x 1423 mm
- 3 boules, masse 210 g, diamètre 61 mm
- 1 boule blanche par joueur (l'une est distinguée par un point noir)
- 1 boule rouge, cible commune aux deux autres

Configuration du jeu

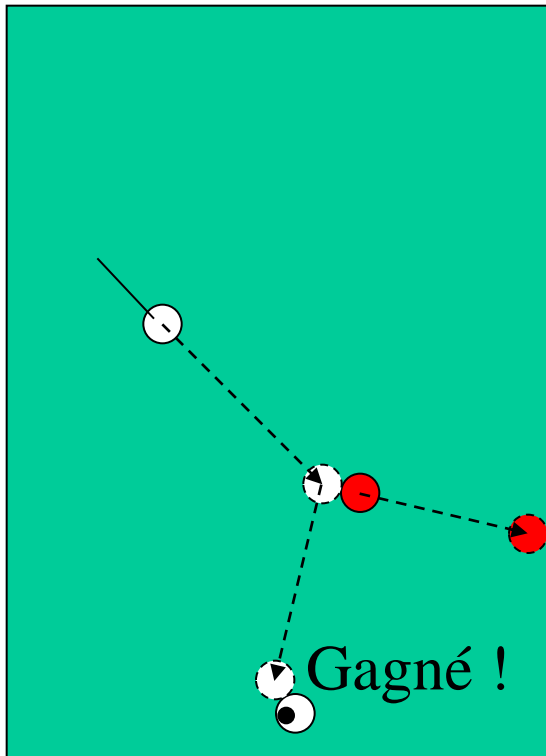


Positions initiales :

- Blanche : $x = 0$ mm,
 $y = -711,5$ mm
- Blanche pointée : $x = 182,5$
mm, $y = -711,5$ mm
- Rouge : $x = 0$ mm,
 $y = 711,5$ mm

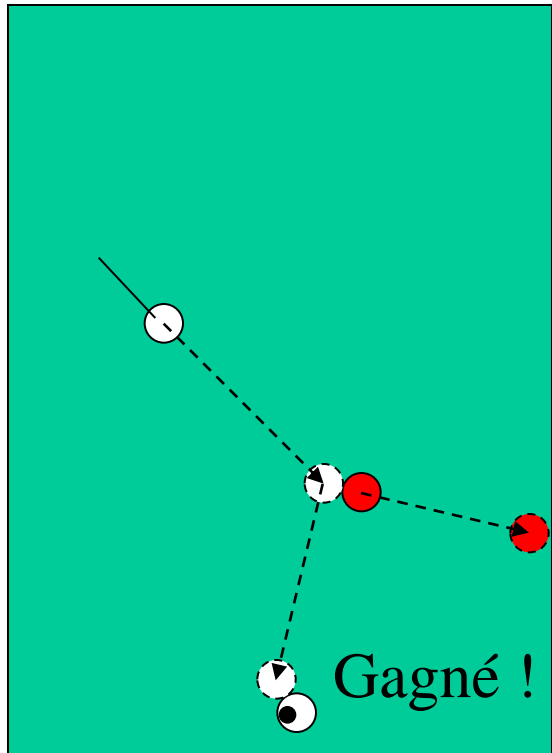
Le joueur jouant la boule
blanche pointée joue en
premier

But du jeu



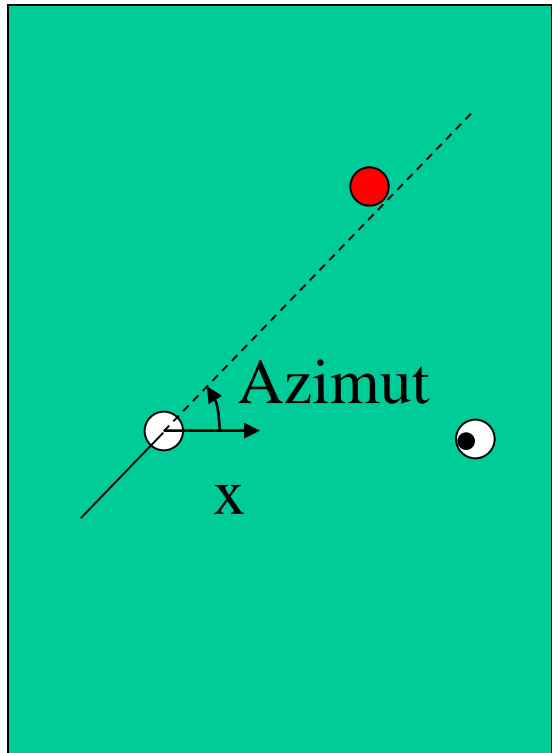
- Chaque joueur à son tour envoie sa boule avec une queue
- Point réussi : la boule joueur choque dans un même coup la boule rouge et la boule adverse
- Ordre des chocs indifférent
- Nombre de rebonds sur les bandes indifférents (hauteur des bandes au point d'attaque d'une boule 36mm)

But du jeu



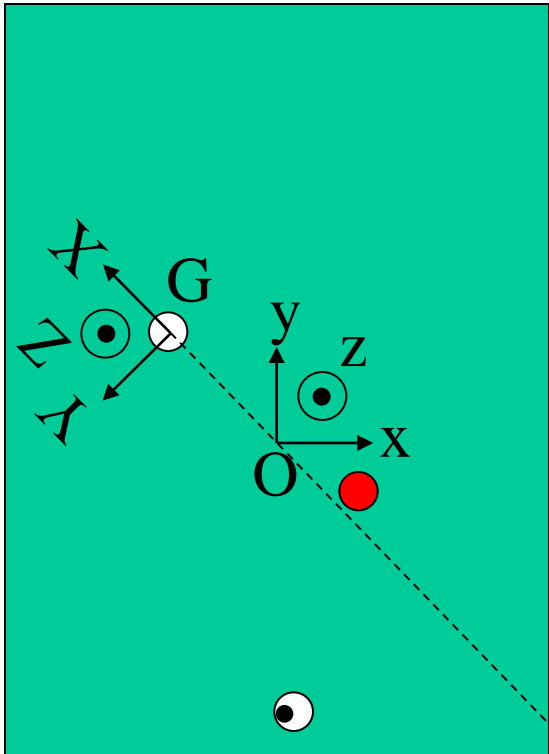
- Après un coup gagnant un joueur rejoue (et ainsi de suite jusqu'à un coup perdant)
- Ronde : ensemble coup(s) joueur 1, coup(s) joueur 2
- La partie se joue en un nombre de rondes convenu à l'avance
- Le vainqueur est celui ayant le plus de points à l'issue de la partie

Réglage d'un tir : azimut



- Les paramètres d'un tir sont au nombre de 5
- Le premier est la direction du tir "azimut" (α : angle $[0-2\pi]$ entre axe x et prolongement de la queue)
- Par commodité de visée, ce prolongement devra être affiché à l'écran lors du réglage

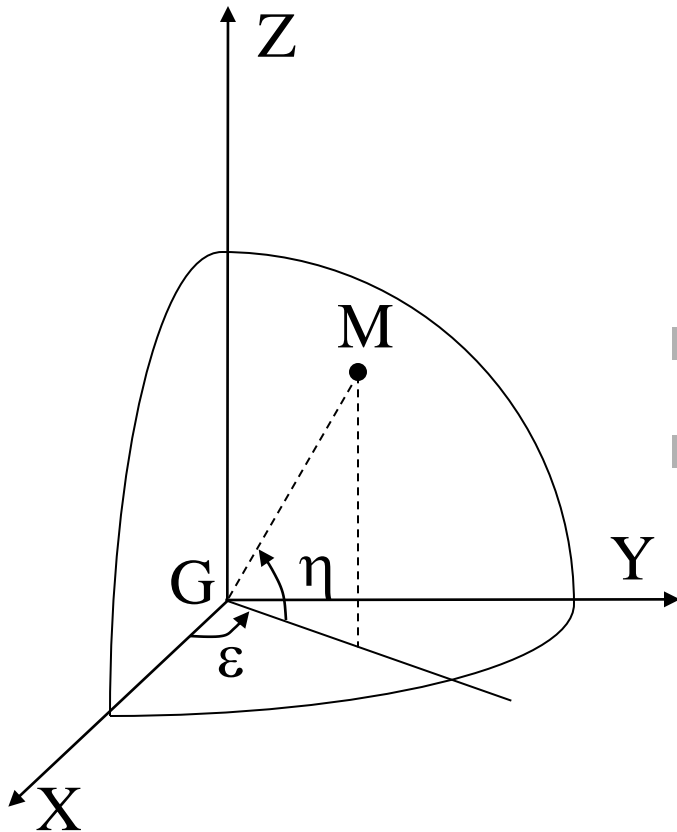
Réglage d'un tir : repère local



Les autres paramètres sont définis dans un repère local :

- G est le centre de masse
- X est horizontal opposé à la direction azimutale (dirigé vers le joueur)
- Z est la verticale ascendante
- Y est horizontal de telle sorte que le repère soit direct

Réglage d'un tir : vue dans le repère local

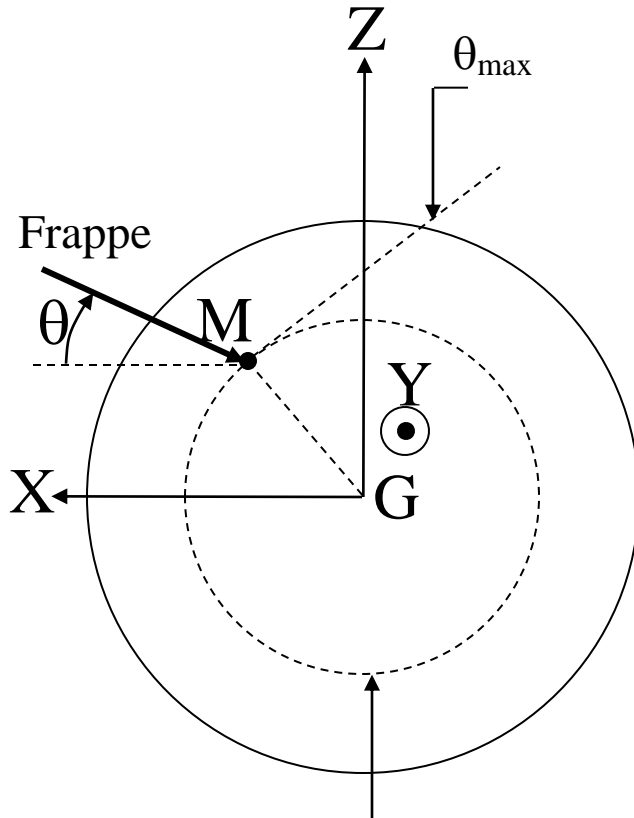


Position du point d'impact M :
deux angles en coordonnées
sphériques entre $-\pi/2$ et $+\pi/2$

- η : angle de hauteur
- ε : angle d'excentrage

Un autre paramètre est
évidemment la force du tir
(module du vecteur
percussion), nous en sommes
donc à 4 paramètres...

Réglage d'un tir : vue en projection sur XGZ



5ème paramètre : en projection sur le plan vertical parallèle à la frappe (vue "de profil")

- θ : angle d'inclinaison
- θ varie entre 0 (percussions "ascendantes" interdites) et θ_{\max} (frappe tangente au petit cercle)

Petit cercle (éventuellement caché) : trajectoire du point d'impact à excentrage constant

Cahier des charges :

Le logiciel permet à deux joueurs humains de s'affronter dans une partie simulée donc :

- De régler tour à tour les paramètres de tir
- D'animer de manière réaliste un tir réglé
- D'évaluer le coup et de rendre la main au joueur dont c'est le tour
- D'arbitrer : comptabiliser les points et proclamer le vainqueur

Cahier des charges :

Le logiciel conserve intégralement l'historique d'une partie donc :

- Enregistrement possible des parties et reprise ultérieure possible
- Retours en arrière possibles
- En cas de reprise sur un coup antérieur un nouvel historique se substitue à l'historique mémorisé => confirmation à demander

Cahier des charges :

Le logiciel permet de régler en début de partie les paramètres

- De jeu : nombre de rondes
- Physiques : coefficients de frottement et de rebond
- De simulation : rapidité de l'animation (pour ces derniers, il doit être possible de les changer en cours de partie)

Conception de l'animation

- Gérée par pas élémentaires : déplacement lors d'un pas proportionnel à la vitesse, évolution vitesse et rotation selon les équations du mouvement
- Effectuer successivement un pas pour chaque boule et ainsi de suite

Conception de l'animation

Un pas pour une boule :

- Calculer l'état après un pas de mouvement libre
- Tester le contact éventuel avec bandes ou autres boules, calculer l'état après tous chocs ou rebonds éventuels
- La boule devant à nouveau être libre de tout contact, modifier alors seulement la position affichée (effacement/réaffichage)

Objectifs du projet

Modélisation UML de cette application :

- Cas d'utilisation et scénarios associés
- Diagrammes de classes
- Diagrammes de collaboration/séquence
- Spécification des méthodes

Documenter le modèle (notes, commentaires)

Codage : facultatif mais obligatoirement conforme au modèle, si réalisé

Organisation du projet

- Nominalement effectuées avec le binôme de TP. Rendu (informatique+papier) dix jours avant la date du final
- Équipes plus importantes : admises si leur objectif est la réalisation et si une gestion de projet est mise en place (note d'organisation nominative, planning)
- La gestion de projet doit dans ce cas être cohérente avec le modèle (découpage en équipes, en tâches élémentaires)

Si vous codez l'application

- Code C++ ou Java lisible, commenté, cohérent avec le modèle, code et modèle si possible en Anglais : publication visée dans le monde du logiciel libre.
- Aucune valeur numérique "en dur" dans le code pour des raisons de maintenabilité (modifiable au pire par changement d'une définition de constante nécessitant au pire une recompilation).
- Encapsuler les primitives d'affichage pour des raisons de portabilité. Dans l'idéal moteur de jeu et bibliothèques graphiques dans des modules séparés.
- Éviter l'usage de bibliothèque trop exotiques dont la pérennité est douteuse.
- Viser au maximum la portabilité

Si vous codez l'application

4 primitives contiennent la mécanique :

- Calcul des conditions initiales d'une frappe
- Un pas de mouvement libre
- Un choc avec une autre boule
- Un rebond sur une bande

Elles vous seront fournies en version complète, si vous parvenez à faire tourner une version simplifiée

Pour en savoir plus : le site de la Fédération Française de Billard

Pour aller plus loin dans :

- Amélioration de l'interface graphique (repères obligatoires sur le tapis...)
- Règles du jeu plus complètes (zones d'exclusion dans les coins...)
- Et pourquoi pas, aborder les autres jeux de billard

<http://www.promo-billard.asso.fr>

Variables d'état d'une boule

- Position : Vecteur $(x,y,z?)$ ($z?$ possibilité de mouvements verticaux ?)
- Vitesse : Vecteur $(x,y,z?)$ (vitesse du centre de masse, $z ?$ possibilité de mouvements verticaux ?)
- Rotation ω : Vecteur (x,y,z) (composante z appelée pivotement nécessaire pour la prise en compte de rebonds à effets sur bandes.

Liste des coefficients physiques

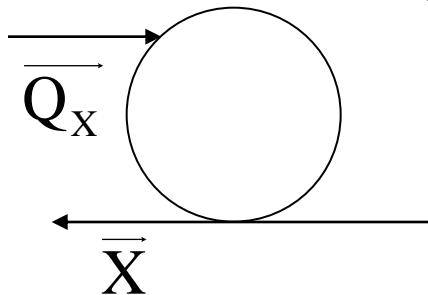
- Constante de gravité à la surface terrestre g
- Coefficients de restitution d'énergie (% de l'énergie restituée) :
 - Boule/Bande : amortissement lors de rebonds sur bande
 - Boule/Boule ? (très voisin de 100% choc élastique)
 - Boule/Tapis ? (très voisin de 0% choc mou+traitement mouvement verticaux ?)
- Coefficients de frottement (R_t/R_n)
 - Boule/Tapis : pour les effets (« friction »)
 - Boule/Bande : pour les effets (rebonds à effet comme au tennis)
 - Boule/queue ? Effet « fausse queue » : dérapage à la surface : réalisable mais complexe : de base considéré comme infini=accroche parfaite
 - Boule/Boule ? (voisin de 0 surface vernies, effet complexe...)
 - Frottement de roulement : amortissement par décélération constante d'un roulement sans glissement : indispensable et différent du coefficient de frottement Boule/Tapis
 - Frottement de pivotement : amortissement par décélération constante d'un pivotement : indispensable car seul phénomène amortissant le pivotement

Conditions initiales : version simplifiée (sans effets)

Pour une version très simplifiée ne tenant pas compte des effets, il suffit de :

- Ne pas tenir compte de la rotation (ω reste égal à 0)
- Ne pas tenir compte de la composante verticale de percussion

La boule est donc considérée comme un point matériel G dont la vitesse initiale est proportionnelle à la composante horizontale de la percussion :



$$\overrightarrow{v}_G = \frac{\overrightarrow{Q}_X}{m} \quad \overline{\omega} = \overline{0}$$

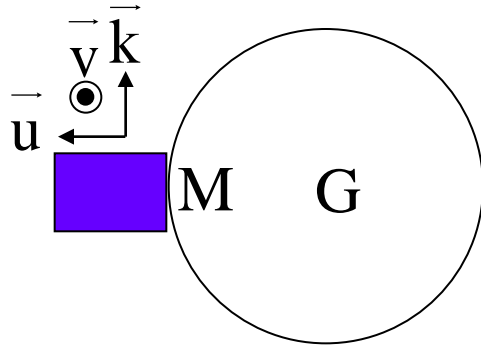
Mouvement libre : version simplifiée (sans effets)

En version simplifiée la boule est considérée comme un point matériel, tous les effets sont ignorés, ce qui revient de fait à considérer qu'il y a toujours roulement sans glissement.

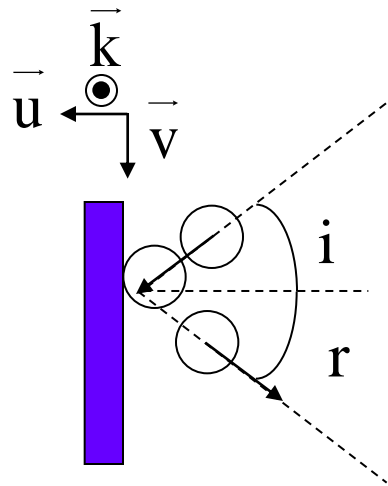
Il faut par contre conserver le frottement de roulement pour ne pas générer de mouvement perpétuel. Donc en version simplifiée :

$$\frac{d\overrightarrow{v_G}}{dt} = -\frac{5}{7r} \delta_R g \frac{\overrightarrow{v_G}}{\|\overrightarrow{v_G}\|} = \overline{Cte}$$

Rebonds sur les bandes : version simplifiée (sans effets)



En version simplifiée, on ne conserve que le terme de lois de Descartes (inversion de la composante de vitesse selon le vecteur u dirigé vers la bande), ce que l'on peut écrire :



$$\vec{v}'_G = \vec{v}_G - 2\vec{v}_G^u$$

$$\vec{v}'_G^u = -\vec{v}_G^u$$

$$\vec{v}'_G^v = \vec{v}_G^v$$

$$r = i$$

Chocs entre boules : version simplifiée (sans effets)

La version simplifiée est le choc élastique et sans frottement,

soit :

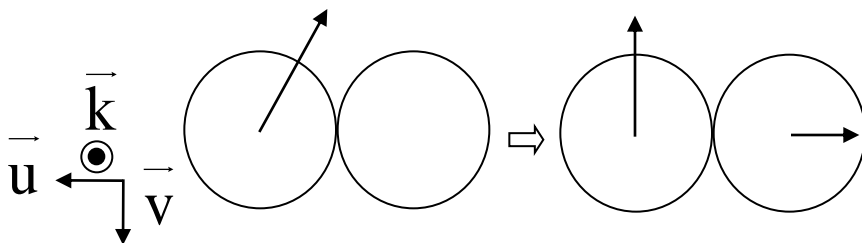
$$\vec{v}'_{G2} - \vec{v}_{G2} = \vec{v}^u_{G1} - \vec{v}^u_{G2} \quad \vec{v}'_{G1} - \vec{v}_{G1} = \vec{v}^u_{G2} - \vec{v}^u_{G1}$$

Dans le cas particulier de la cible immobile on trouve par

exemple :

$$\vec{v}'_{G2} = \vec{v}^u_{G1} \quad \vec{v}'_{G1} = \vec{v}_{G1} - \vec{v}^u_{G1}$$

Interprétation : le projectile cède à la cible la composante normale de sa vitesse et conserve la composante tangentielle
 \Rightarrow les vitesses après le choc sont orthogonales.



Si choc de plein fouet, le projectile s'arrête net

Mouvement libre : conseils de conception de l'animation

Afin que les paramètres correspondent aux grandeurs physiques usuelles, effectuer tous les calculs dans le système MKSA (distances en mètres, temps en secondes) et non en unités arbitraires. Deux paramètres d'échelle sont alors nécessaires :

- Un paramètre d'échelle spatiale genre nombre de pixels représentant 1m du terrain réel.
- Un paramètre d'échelle temporelle genre nombre de pas de temps par seconde. C'est ce paramètre, qui détermine la vitesse de l'animation, qui est à fixer en fonction des performances de la machine.

Mouvement libre : conseils de conception de l'animation

Le paramètre d'échelle spatiale sert à réaliser, juste au moment d'afficher, la conversion :

$$\text{Distance_affichée} = \text{Nb_Pixels_par_m} * \text{Distance_réelle}$$

Le paramètre d'échelle temporelle fixe le pas de temps élémentaire :

$$\Delta t = \frac{1}{\text{Nb_pas_s}}$$

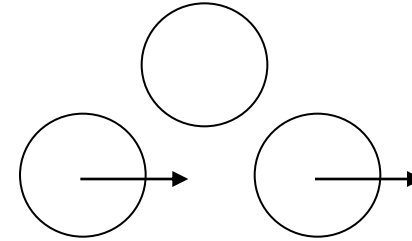
A chaque pas la position est incrémentée de $\vec{v}\Delta t$

et si nécessaire vitesse et rotation respectivement de

$$\frac{d\vec{v}}{dt} \Delta t \quad \text{et} \quad \frac{d\vec{\omega}}{dt} \Delta t$$

Détection des chocs et rebonds

- Le seul test en fin de pas n n'est en fait pas suffisant (un objet rapide peut en « traverser » un autre :



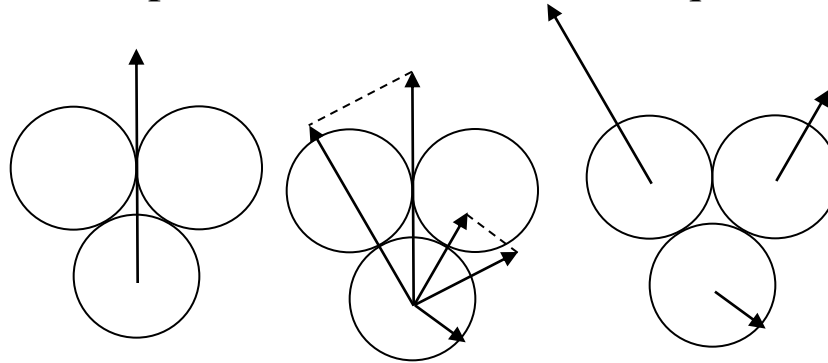
Position à t Position à $t+\Delta t$

- Les mouvements étant rectilignes uniformes la recherche d'interactions éventuelles entre objets ne représente que la résolution d'équations du premier degré (à une inconnue : instant de l'interaction qui doit être compris entre t et $t+\Delta t$), en nombre $3*4$ (3 boules, 4 bandes) + 3 (3 paires de boule) = 15 equations (pas grand-chose donc pour nos machines actuelles...)
- Si plusieurs interaction trouvées, effectuer la première rencontrée (et éventuellement si nécessaire raccourcir l'attente avant affichage de la durée ad-hoc)

Chocs symétriques

- Dernière difficulté : en cas de chocs véritablement simultanés : nécessite des situations très symétrique comme on en rencontre parfois pour les « casses » dans les jeux à plus de trois boules, il n'est pas toujours possible de les traiter l'un après l'autre car on peut trouver plusieurs solutions respectant les équations de la physique (conservation énergie, quantité de mouvement, moment cinétique) mais violant la symétrie, ce qui donne parfois des mouvement choquants à la simulation :

D'abord gauche puis droite



D'abord droite puis gauche donne une autre solution tout aussi dissymétrique : il faut donc traiter ce cas de choc symétrique d'une boule sur deux autres comme un cas particulier en imposant que la solution soit symétrique.