

Física per a videojocs

Professors de VJ

Física per videojocs

És un tema molt ampli:

- Cinemàtica del sòlid rígid.
- Cinemàtica inversa.
- Dinàmica del sòlid rígid.
- Detecció de col·lisions.
- Resposta a la col·lisió.
- Molles.
- Fluids.

Motors

És un tema complex.

Habitualment es fan servir motors:

- Box2D: <http://www.box2d.org/>
- Bullet: <http://www.bulletphysics.org/>
- ODE (Open Dynamics Engine): <http://www.ode.org/>
- PhysX: <http://developer.nvidia.com/physx>
- Havok: <http://www.havok.com/>

Conceptes

Dos costats:

- Recerca → Precisió
- Jocs → Aparença

Objectes:

- Estàtics
- Dinàmics

Cinemàtica bàsica

Estudia el moviment sense analitzar les causes.

Càlcul de la posició

$$p = p_0 + v \cdot \Delta t$$

On:

- $\Delta t = t - t_0$
- En 2D: $p = (x, y)$, $p_0 = (x_0, y_0)$, $v = (v_x, v_y)$
- En 3D: $p = (x, y, z)$, $p_0 = (x_0, y_0, z_0)$, $v = (v_x, v_y, v_z)$

Cinemàtica bàsica

Afegint acceleració:

$$a = dv / dt \rightarrow dv = a \cdot dt \rightarrow \Delta v = v - v_0 = a \cdot \Delta t$$

$$v = v_0 + a \cdot \Delta t$$

És el que es coneix com a integració d'Euler.

Podríem fer servir l'equació de posició:

$$x = x_0 + v_0 \cdot \Delta t + \frac{1}{2} \cdot a \cdot \Delta t^2$$

per tenir en compte l'acceleració entre frames.

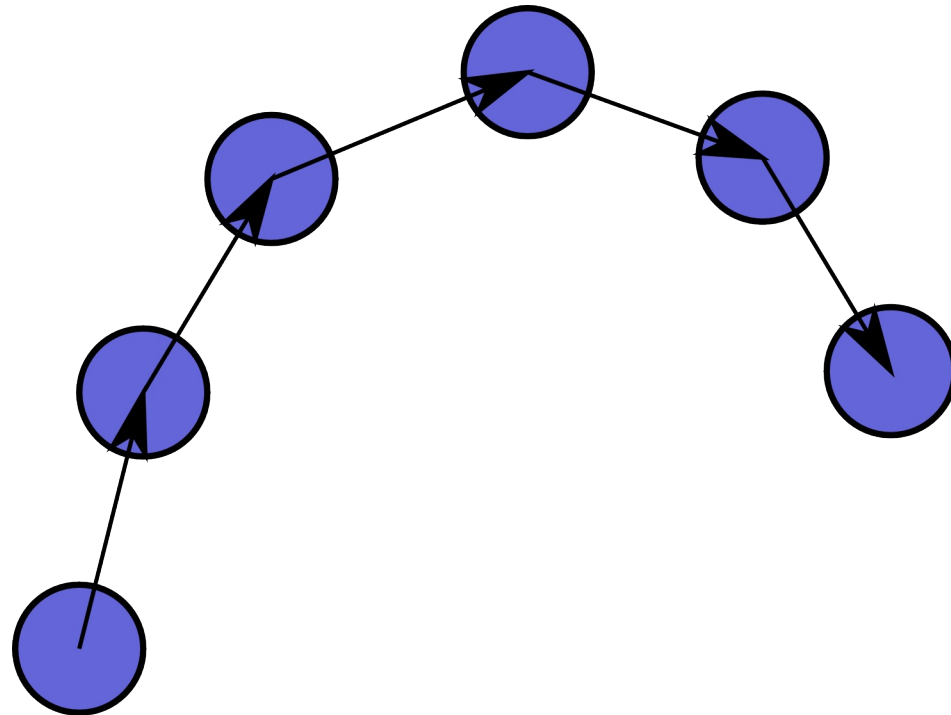
Per problemes de complexitat i d'estabilitat no ho fa ningú.

Cinemàtica bàsica

- La velocitat canvia a cada frame degut a l'acceleració.
- Però entre frames es manté constant.

Conseqüència:

Trajectòria →



Dinàmica

Estudia les causes del moviment → Forces

- S'expressen com a vectors:
 - Tenen direcció.
 - En 2D: $F = (F_x, F_y)$
 - En 3D: $F = (F_x, F_y, F_z)$
- Es poden sumar per obtenir la força total aplicada a l'objecte:
 - Es fa per components.
 - $F_{Tx} = F_{1x} + F_{2x} + F_{3x}$
 - $F_{Ty} = F_{1y} + F_{2y} + F_{3y}$
 - $F_{Tz} = F_{1z} + F_{2z} + F_{3z}$

Dinàmica

La força total aplicada a cada objecte F_T permet calcular l'acceleració:

$$F_T = m \cdot a$$

$F_T \equiv$ Força total, $m \equiv$ Massa, $a \equiv$ Acceleració

Aquesta acceleració és fa servir en la simulació cinemàtica.

- Modifica la velocitat.

Col·lisions

Dos parts en el tractament de col·lisions

- Detecció de col·lisions (collision detection).
- Resposta a la col·lisió (collision response).

Poden ser entre:

- Objecte estàtic / Objecte dinàmic
 - Simple
- Objecte dinàmic / Objecte dinàmic
 - Més complicat

Dinàmica

Queden un parell de conceptes útils pel càlcul de la resposta de col·lisions.

- Moment lineal (p)

$$p = m \cdot v$$

- Energia cinètica (E_v)

$$E_v = m \cdot v^2 / 2$$

$m \equiv$ Massa, $v \equiv$ Velocitat

Col·lisions

Suposicions:

- Ignorem fricció.
- No hi ha deformació.
 - Simulació de sòlids rígids.

Resultats:

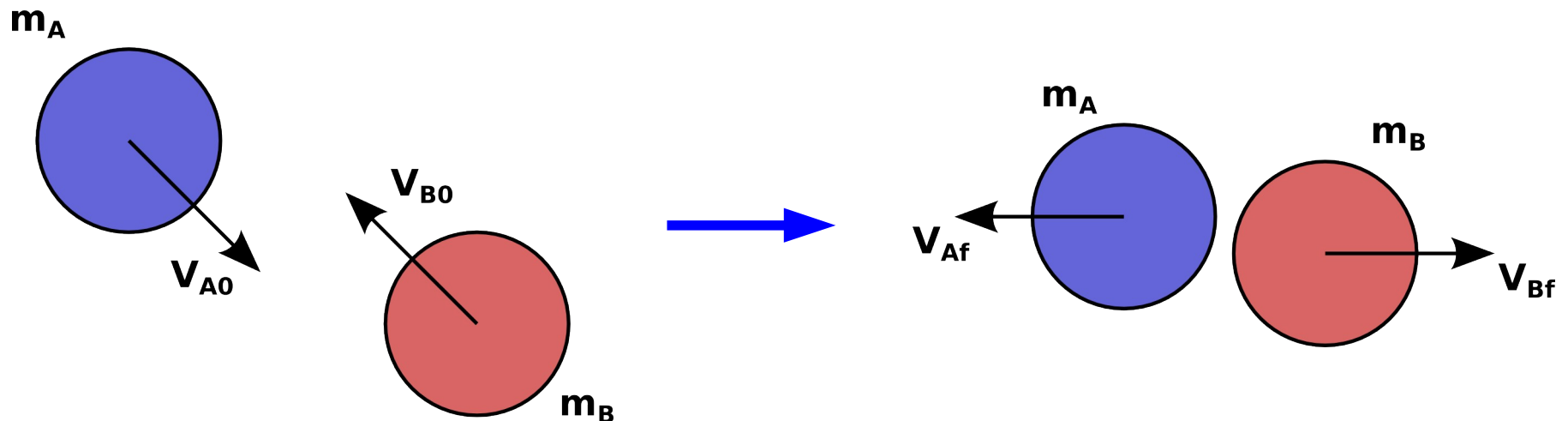
- Col·lisió perfectament elàstica
 - Es conserva el moment lineal total.
- Col·lisió elàstica imperfecta
 - Part del moment lineal es perd en la col·lisió.

Col·lisions

Posem que tenim dos objectes col·lisionant:

- Objecte A amb massa m_A i velocitat v_{A0} .
- Objecte B amb massa m_B i velocitat v_{B0} .

Hem de poder calcular les velocitats resultants de la col·lisió: v_{Af} , v_{Bf} .



Col·lisions

Si tenim col·lisions elàstiques perfectes:

- El moment linial total s'ha de conservar.

$$p_{A0} + p_{B0} = p_{Af} + p_{Bf}$$

$$m_A \cdot v_{A0} + m_B \cdot v_{B0} = m_A \cdot v_{Af} + m_B \cdot v_{Bf}$$

Tenim 1 equació, 2 incognites.

- Cal conservar també l'energia cinètica.

$$m_A \cdot v_{A0}^2 / 2 + m_B \cdot v_{B0}^2 / 2 = m_A \cdot v_{Af}^2 / 2 + m_B \cdot v_{Bf}^2 / 2$$

Això permet trobar la solució 1D.

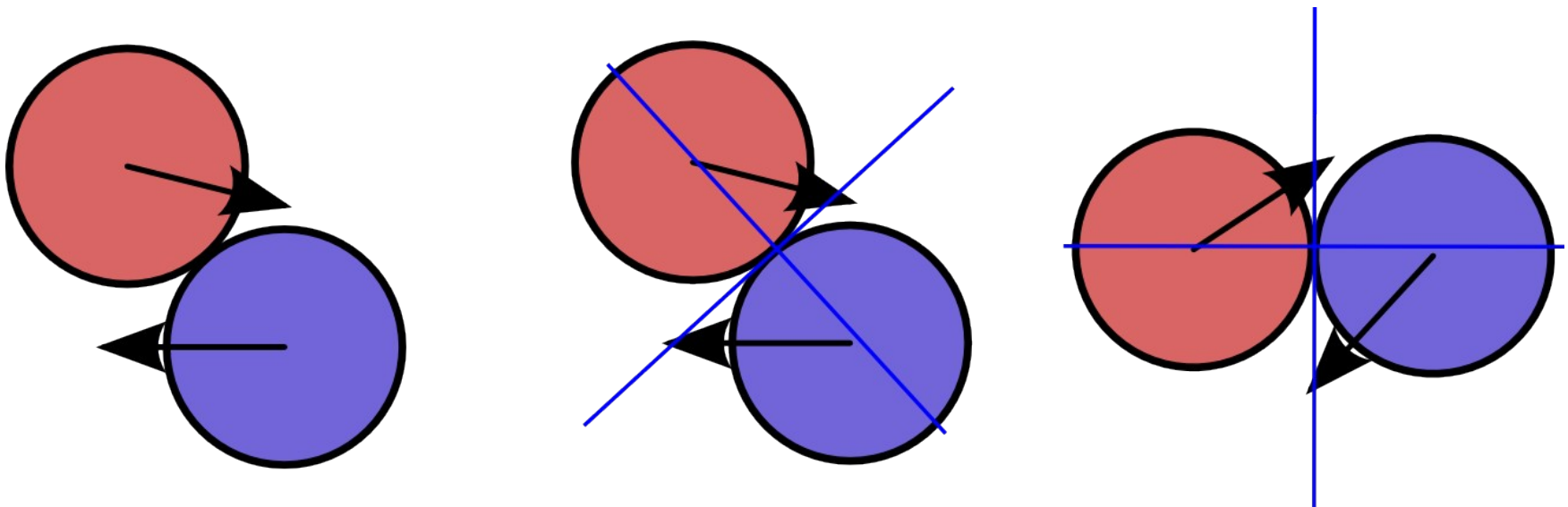
$$v_{Af} = \frac{v_{A0}(m_A - m_B) + 2m_B v_{B0}}{m_A + m_B}$$

$$v_{Bf} = \frac{v_{B0}(m_B - m_A) + 2m_A v_{A0}}{m_A + m_B}$$

Col·lisions

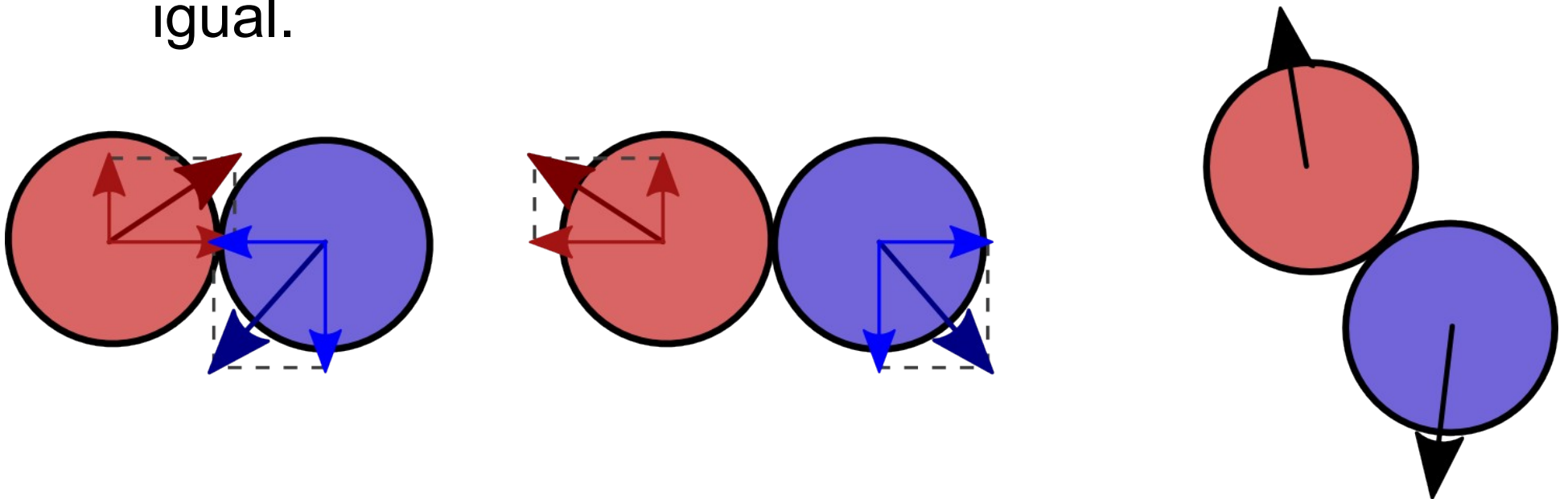
Les velocitats són vectors (2D o 3D).

- Es tracta de transformar el problema en 1D.
- Rotem fins que la col·lisió estigui alineada amb l'eix OX.
- Entren en joc el punt de contacte C i la normal de contacte N.



Col·lisions

- Les velocitats rotades tenen dos components en el nou sistema:
 - Pel component X (component normal), calculem col·lisió 1D i invertim direcció.
 - El component Y (component tangencial) el deixem igual.



Col·lisions

En el cas de la col·lisió sigui entre objecte estàtic i objecte dinàmic:

- Les velocitats tan inicial com final de l'objecte estàtic són zero.
- Les equacions queden molt més simples.

$$v_{A0} = 0, v_{Af} = 0$$

$$m_B \cdot v_{B0} = m_B \cdot v_{Bf} \rightarrow v_{B0} = v_{Bf}$$

1D: Només cal invertir la direcció de la velocitat.

2D: Invertim el component normal, sense alterar el component tangencial.

Col·lisions

Quan es fa la simulació s'ha d'anar amb compte en que els errors no introdueixin energia al sistema (increment de moment lineal).

Propietats necessàries per tot objecte dinàmic:

- Massa.
- Posició.
- Velocitat.
- Acceleració (produïda per la suma de forces).
- Mètode per a detectar col·lisions.

Detecció de col·lisions

Ens centrem en comprovar la col·lisió (intersecció) d'una parella d'objectes.

- El problema es torna més simple.
- Dóna lloc a una primera solució:
 - Comprovar col·lisió entre tota parella d'objectes.
 - Costós.
 - Cal descartar tants objectes com sigui possible (atribut bWall als TBE).

Detecció de col·lisions

En ordre incremental de:

- Cost.
- Precisió.

Tenim tècniques per preguntar:

- Hi ha la possibilitat de que aquests dos objectes col·lisionin?
- Col·lisionan aquests dos objectes?
- Quan col·lisionen?
- On col·lisionen (punt d'impacte)?

Detecció de col·lisions

No volem haver de comprovar tots els objectes dos a dos.

Dues fases:

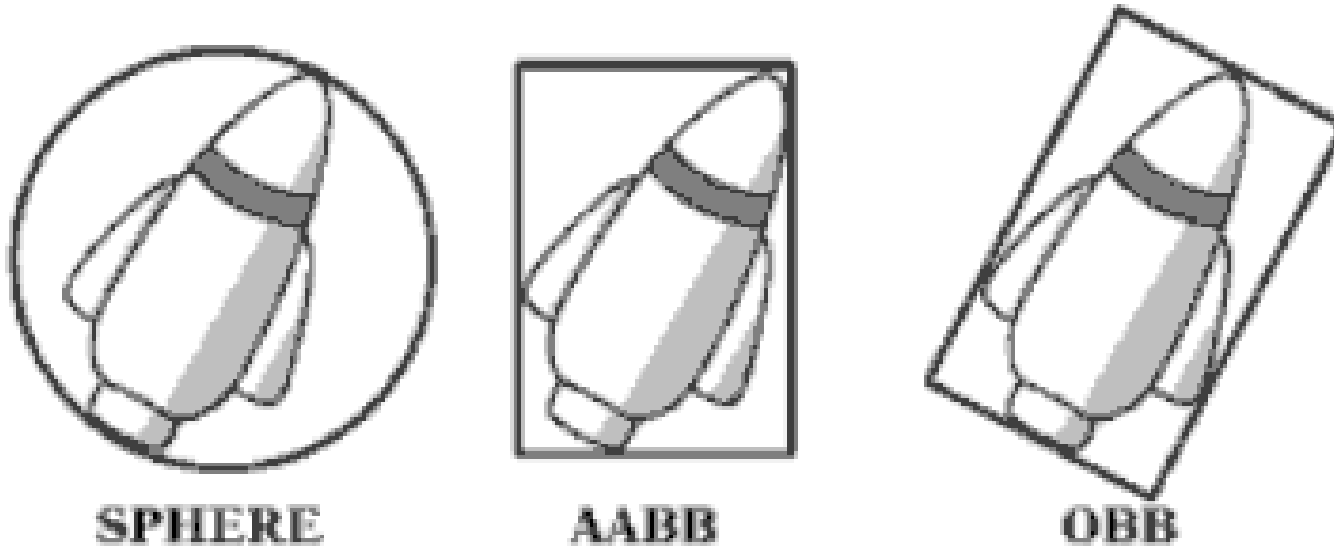
- Detectar candidats a col·lisió.
 - Fent servir Estructures de Partició Espacial.
- Comprovar col·lisions de forma més precisa.
 - Interseccions entre volums envolupants.
 - Càlcul de punts i normals de contacte.

Detecció de col·lisions

Volums envolupants

- Model molt simple que conté l'objecte
 - Serveix per simplificar càlcul de col·lisions.
- Propietats importants:
 - Càlcul d'intersecció eficient.
 - Que s'ajusti al model.
 - Poc consum de memòria.
 - Fàcil de transformar.
 - Fàcil de calcular.

Detecció de col·lisions



Esferes

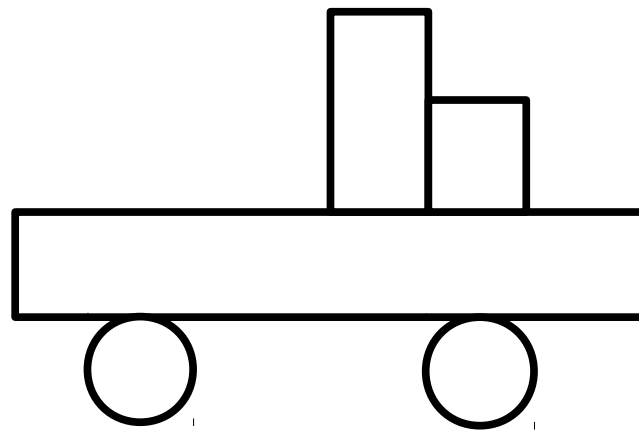
AABB: Axis Aligned Bounding Box
Capses alineades amb els eixos

OBB: Oriented Bounding Box
Capses amb orientació arbitrària

Detecció de col·lisions

Jeràrquies de volums envolupants

- Els mètodes més simples no s'ajusten bé a models complexos.
 - Podem fer servir diversos volums envolupants.
 - Quan cerquem col·lisions no cal comprovar interseccions entre ells.



Detecció de col·lisions

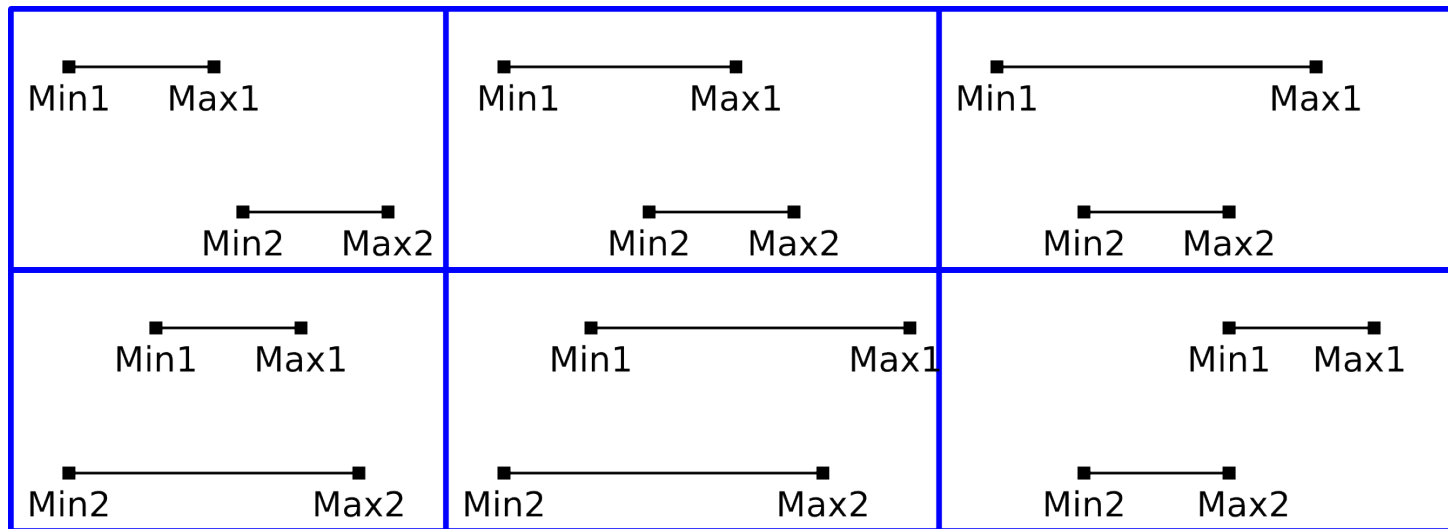
Per reduir el nombre de comprovacions:

- Estructures de partició espacial
 - Octrees, quadrees.
 - Graella uniforme.
 - Es divideix el mon en una graella de cel·les del mateix tamany.
 - S'associa cada objecte amb les cel·les que “trepitja” la seva capsula envolupant.
 - Només els objectes que es troben en una mateixa cel·la es comparen.
 - En el cas d'un TBE els tiles defineixen la graella de forma automàtica.

Detecció de col·lisions

Intersecció de capses \rightarrow Intersecció d'interval·ls

1D



Intersecció si es compleix que (totes dues):

- $\text{Min1} < \text{Max2}$
- $\text{Min2} < \text{Max1}$

Detecció de col·lisions

2D

- Capsa 1: $(\text{minx1}, \text{miny1}) - (\text{maxx1}, \text{maxy1})$
- Capsa 2: $(\text{minx2}, \text{miny2}) - (\text{maxx2}, \text{maxy2})$

Intersecció si és compleix (totes dues):

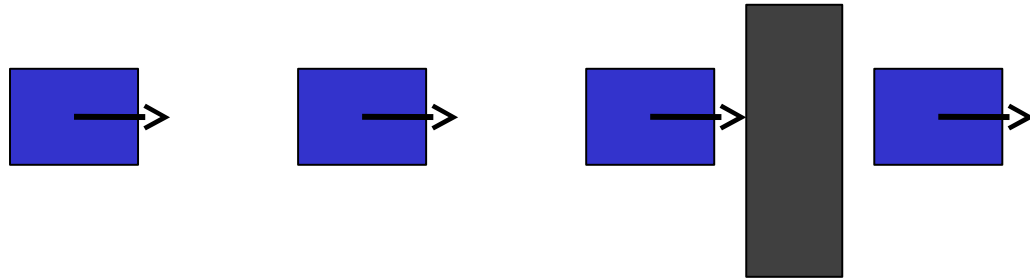
- $(\text{minx1} < \text{maxx2}) \ \&\& \ (\text{minx2} < \text{maxx1})$
- $(\text{miny1} < \text{maxy2}) \ \&\& \ (\text{miny2} < \text{maxy1})$

3D

- Extendre una coordenada més.

Detecció de col·lisions

Cal anar amb compte amb l'efecte túnel.



Tenir refresc prou alt (es recomana un mínim de 60 fps per la física) o ...

Detecció de col·lisions

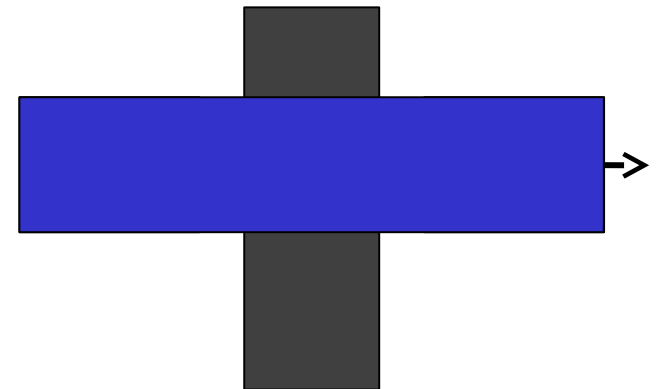
Canviar el sistema de coordenades per a que un dels dos objectes sigui estàtic:

- Sistema de coordenades mòbil.
- Complicat.



0 ...

Calcular formes d'escombrat.



Molles

Les molles es poden simular fent servir la llei de Hooke:

$$F = - k \cdot x$$

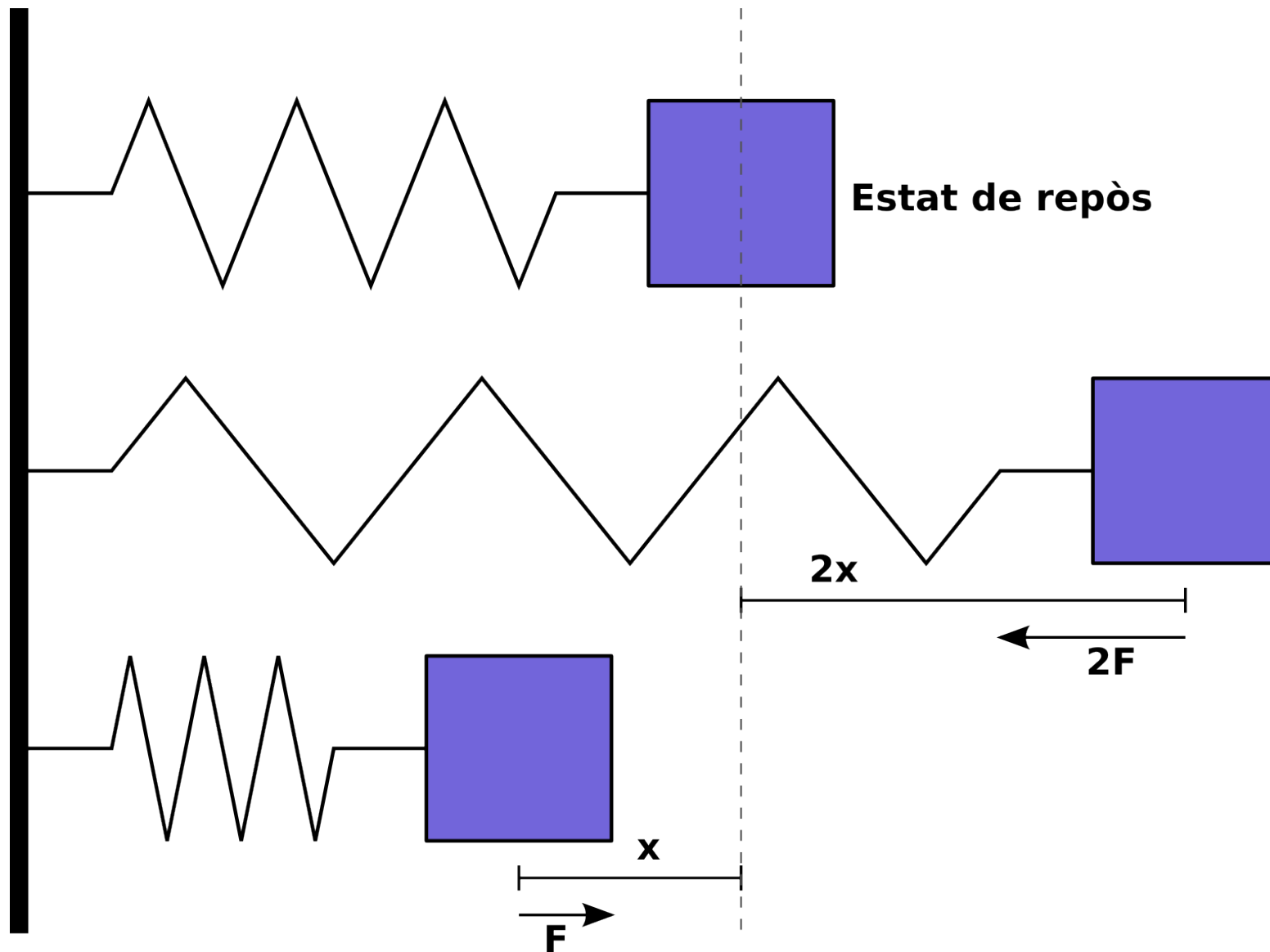
On:

- $F \equiv$ Força exercicida per la molla.
- $x \equiv$ Distància des del punt d'equilibri.
- $k \equiv$ Constant de la molla.

Per tant:

- La força es proporcional a la compressió de la molla.
- Amb la constant k controlem com de tibant és.

Molles



Molles

Sense res més oscil·laria indefinidament.

- Cal afegir amortiment.

$$F = -k \cdot x - b \cdot v$$

On:

- $v \equiv$ Velocitat relativa dels dos extrems de la molla.
- $b \equiv$ Coeficient d'amortiment de la molla.

Molles

Calen integradors especials per millorar precisió i estabilitat:

- Integració de Verlet, RK4, ...

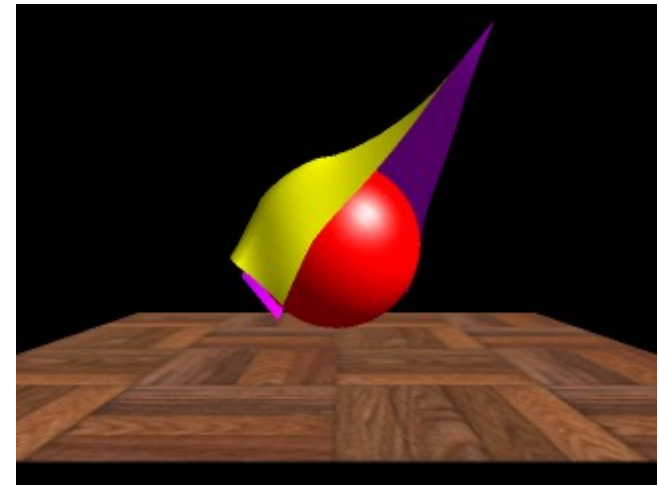
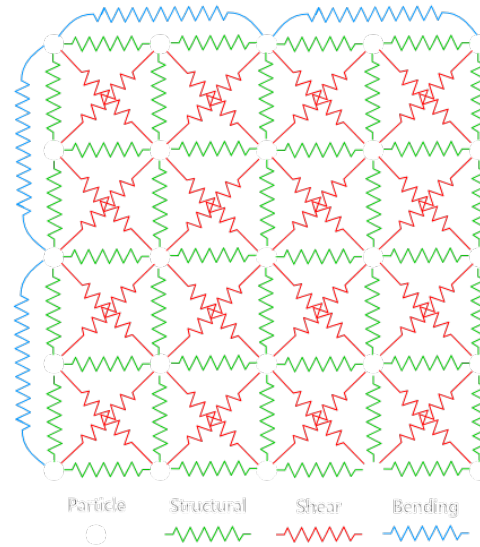
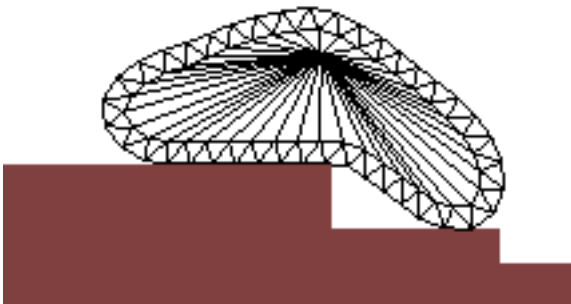
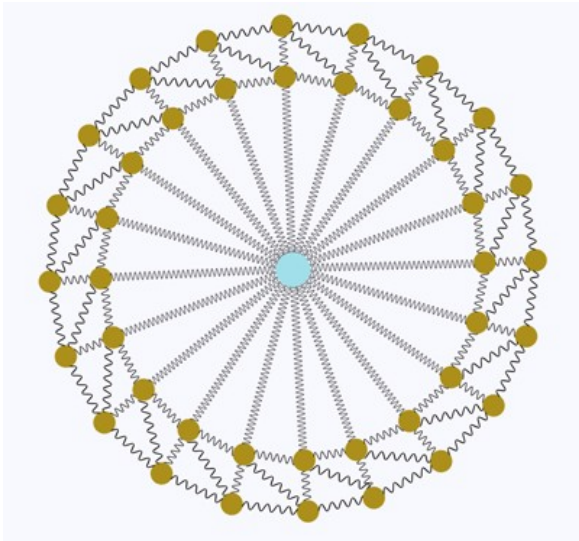
Les molles permeten simular efectes:

- Restriccions entre punts.
- Objectes deformables.
- Roba.

Molles

Deformables

Roba



Fluids

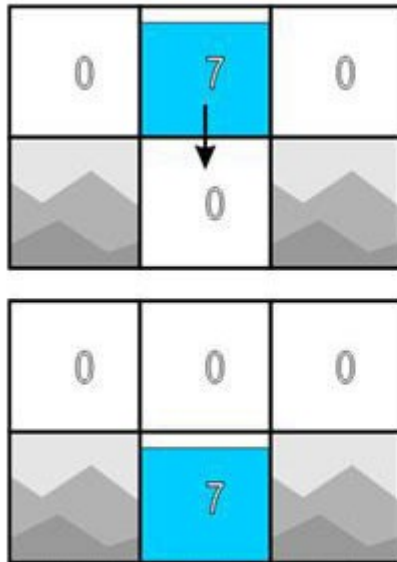
Multitud de maneres de simular fluids.

En jocs els més usats són:

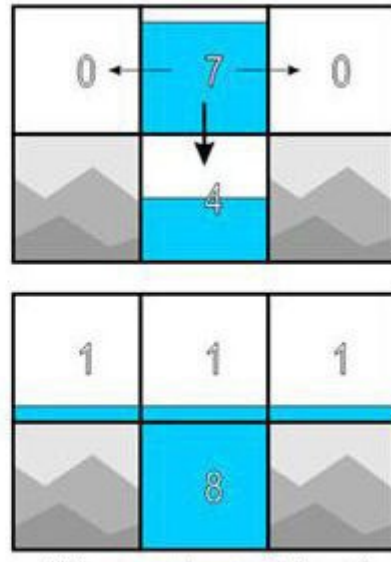
- Autòmates celulars.
- Smoothed Particle Hydrodynamics (SPH).
- Mètodes procedurals.

Fluids

Autòmates celulars (Minecraft)

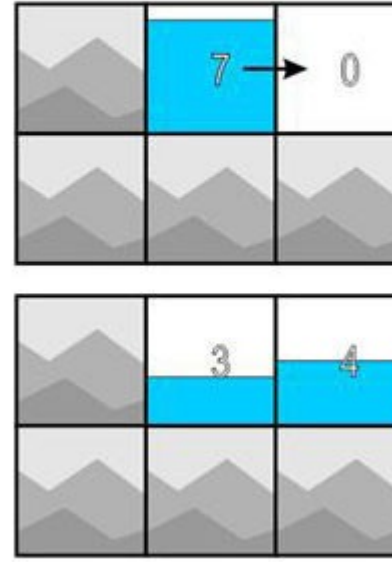


Simplest case: unsupported water simply falls down.

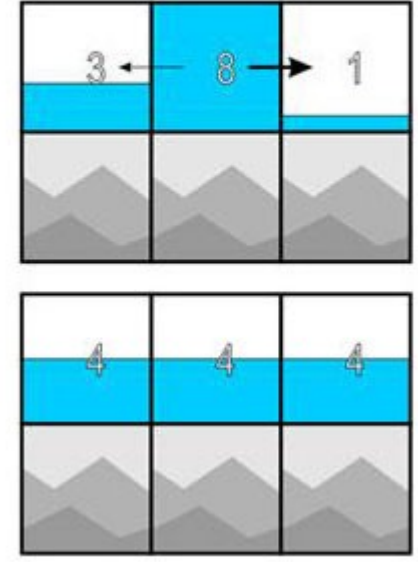


A little more complex: water fills lower cell, then spreads evenly at its own layer.

Pretty basic: on the same plane, water spreads evenly to adjacent cells. Excess fluid flows, giving water a "restless" quality.

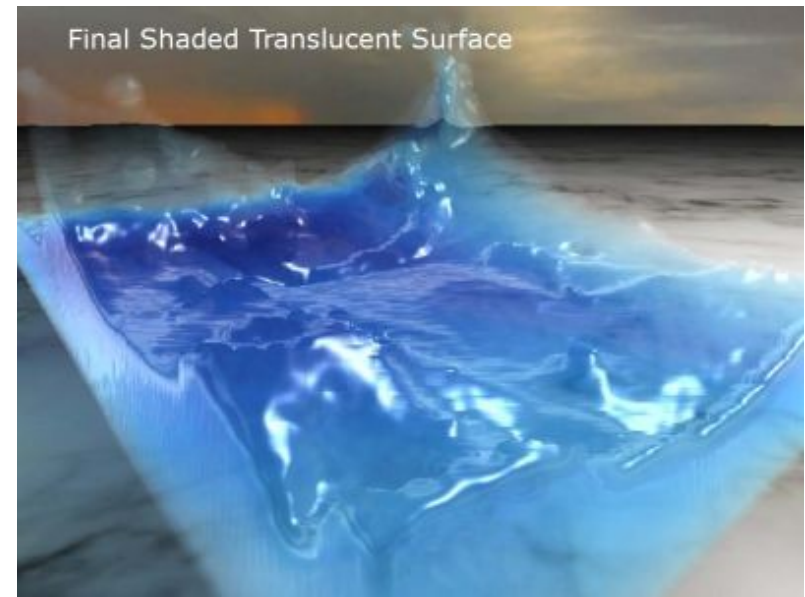
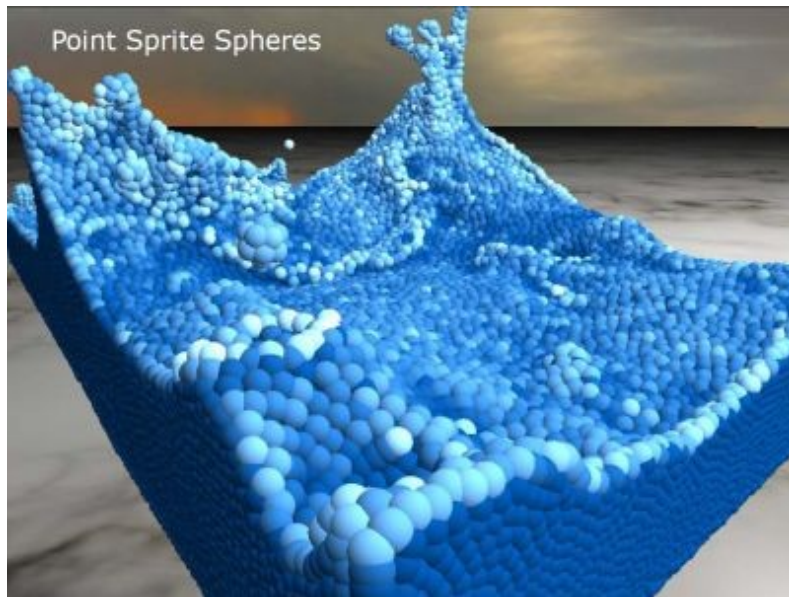


Water spreading in both directions at its own level tries to average levels between destinations and source.



Fluids

Smoothed Particle Hydrodynamics (PhysX)



Fluids

Mètodes procedurals

- Suma d'ones (sinusoidals, ...)

