Computer Animation
8-Collisions
SS 13

Prof. Dr. Charles A. Wüthrich,
Fakultät Medien, Medieninformatik
Bauhaus-Universität Weimar
caw AT medien.uni-weimar.de

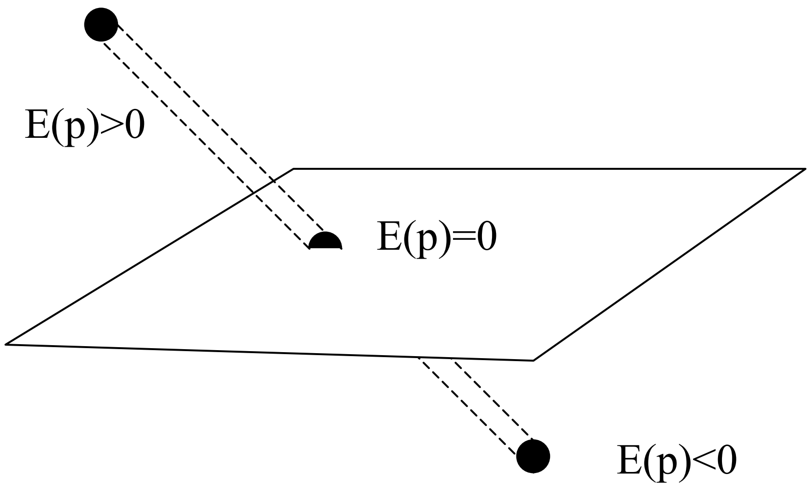
Collisions

- When objects start to move, they actually collide
- Two issues must be addressed:
  - Detecting collision
  - Computing appropriate response
- Detecting collision: two main approaches
  - Penalty method: calculate the reaction after collision has occurred
    - when more particles involved, assume they collided at same instant
    - Imprecise but often acceptable
  - Back up time to first instant of collision and compute appropriate response
    - By heavy no of collisions, quite time consuming
- Computing the appropriate response to collision (depends on physics and distribution of mass of the object)
  - Kinematic response
  - Penalty method: introduce a nonphysical force to restore non penetration but compute it at time of collision
  - Calculation of impulse force

Kinematic response

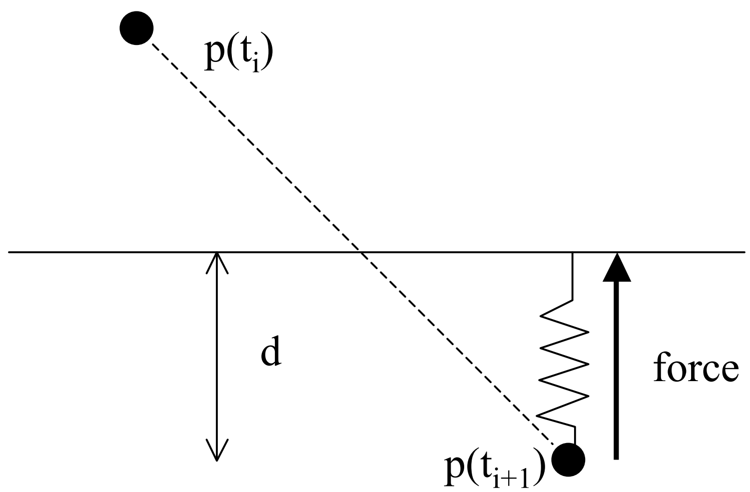
- A simple case is a particle moving at constant velocity and impacting a plane
- Questions:
  - When is the impact?
  - How does it bounce off?
- Use plane equation E(p): ax+by+cz+d
- If normals correct, then
  - If E(p)=0 then p plane point
  - If E(p)>0 then p above plane
  - If E(p)<0 then below plane
- The particle moves with equations:
  $p(t_i)=p(t_{i-1})+t \cdot v_{ave}(t)$

- When $E(p(t_i))$ switches to ≤0 then we had a collision
- Now the component of the velocity parallel to the normal to the plane is negated
- Some damping factor N is added
  $v(t_{i+1})=v(t_i)-v(t_i)N-kv(t_i)N$
  $=v(t_i)-(1+k)v(t_i)N$



Penalty method

- Here we construct a reaction to the collision
- A spring with zero rest length is attached at the instant of collision
- The closest point on the surface to the penetrating point is used as attachment point
- The spring obeys Hooke's law: F=-kd
- The approach needs to assign arbitrary masses and constant, and therefore is not ideal
- Moreover, for fast moving points it might take a few steps to push back the obj
- For polyhedra, it might also generate torque



Polyhedras colliding

- Shape can be complicated for complex objects
- Thus, collisions can be tested before on bounding boxes
- Or by adding hierarchical bounding boxes
- Testing a point to be inside a polyhedron is not easy
- But for a polyhedron one needs to test all vertices for the two objects
- And each point has to be tested against all the planes of the faces of the polyhedron
- This works only for convex polyhedra
- For concave polyhedra, one can use a similar method to the point in polygon test
- Construct a semi-infinite ray from the point towards the polyhedron, and check no of intersections
  - If they are even, then the point is outside
  - If they are odd, then it lies inside
- Of course correct counting double points has to be done
- In some cases, for solids of simple shape and moving with an easy movement, the volume of it can be swept along its trajectory

Impulse force of collision

- To do accurate computations, time has to be backed to the instant of collision
- Then the exact reaction can be computed
- If a collision appeared between t_i and t_{i+1}, then
  - recursive bisection of the time step between these two timepoints will eventually yeld the exact time of the impact
  - Alternatively, a linear approximation of the velocity can be used to simplify the calculations
- At the time of the impact, the normal component of the point velocity can be modified to reflect the bounce
- This normal can be multiplied by a scalar to model the degree of elasticity of the impact
- This scalar is called coefficient of restitution

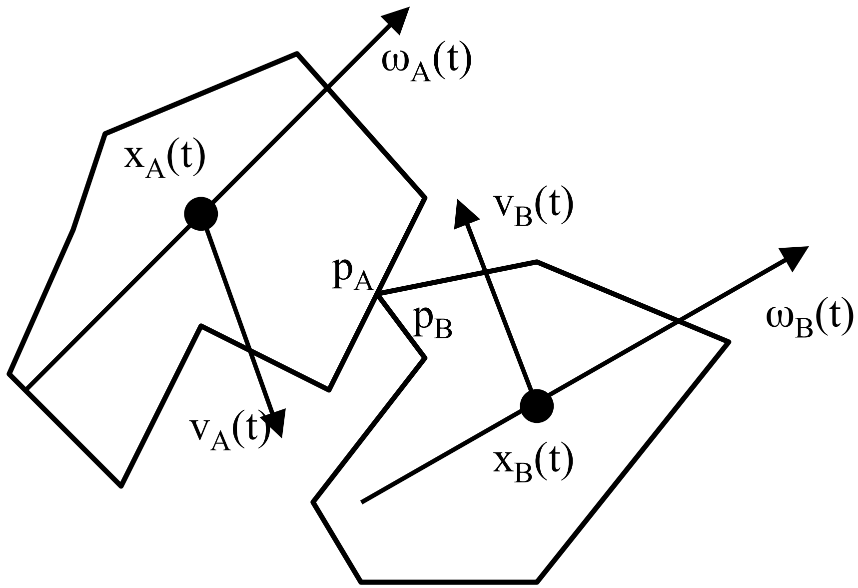
Impulse forces

- Once the simulation is backed up to the time of the collision, the reaction can be computed
- By working back from the desired change in velocity, the required change in momentum can be deduced
- This equation uses the a new term, the impulse, expressed in units of momentum
  $J=F\Delta t=Ma\Delta t=M\Delta v=$
  $\Delta(Mv)=\Delta P$
- J can be seen as a large force acting in a short time interval
- This allows computing the new momentum
- To characterize elasticity, the coeff. of restitution, ε is computed ($0\leq\varepsilon\leq1$)
- The velocities along the normal before and after the impact are related by $v^{+}_{rel}=-\varepsilon v^{-}_{rel}$

Impulse forces

- Assume that the collisions of the two objects A and B has been detected at t
- Each obj Ob has position of mass center $x_{Ob}(t)$, lin. velocity $v_{Ob}(t)$ and ang. velocity $\omega_{Ob}(t)$
- At the point of intersection, the normal to the surface of contact is determined (note, it can be a surface, but also a point)
- Let r_A and r_B be the relative positions of the contact points WRT the center of mass
- Relative velocities of the contact points WRT center of mass and the velocities of the contct points are computed as

- $r_A=p_A-x_A(t)$
  $r_B=p_B-x_B(t)$
  $v_{rel}=(p_A{}^\circ(t)-p_B{}^\circ(t))$
  $p_A{}^\circ(t)=v_A(t)+\omega_A(t)\times r_A$
  $p_B{}^\circ(t)=v_B(t)+\omega_B(t)\times r_B$



Impulse forces

- Linear and angular velocities of the objects before the collision v_{ob}^{-} ω_{ob}^{-} are updated v_{ob}^{+} ω_{ob}^{+}
  $v_A^{+}=v_A^{-}+jn/M_A$
  $v_B^{+}=v_B^{-}+jn/M_B$
  $\omega_A^{+}=\omega_A^{-}+I_A^{-1}(t)(r_A\times j\cdot n)$
  $\omega_B^{+}=\omega_B^{-}+I_B^{-1}(t)(r_B\times j\cdot n)$
  where the impulse J is a vector quantity in the direction of the normal
  $J= j\cdot n$
- To find the impulse, the difference between the velocities of the contact points after collision in the direction of the normal to the surface of collision is formed
- $v_{rel}^{+}=n\cdot(\dot{p}_A^{+}(t)-\dot{p}_B^{+}(t))$
  $v_{rel}^{+}=n\cdot(v_A^{+}(t)+\omega_A(t)\times r_A - v_B(t)+\omega_B(t)\times r_B)$
- Substituting previous equations one obtains

$$j=\frac{-((1+\varepsilon)\cdot v_{rel}^{+})}{\frac{1}{M_A}+\frac{1}{M_B}+n\cdot(I_A^{-1}(t)(r_A\times n))\times r_A+(I_B^{-1}(t)(r_B\times n))\times r_B}$$

- Contact between two objects is defined by the point on each involved and the normal to the surface of contact
- If the collision occurs, the eq. above is used to compute the magnitude of the impulse
- The impulse is then used to scale the contact normal, and update linear and angular momenta

Friction

- An object resting on another one has a resting contact with it
- This apples a force due to gravity which applies to both objects and can be decomposed along the directions parallel F_{Pa} to the resting surface and F_N perpendicular to it
- The static friction force is proportional to F_N:

  $$F_s=\mu_s F_N$$

- Once the object is moving, there is a kinetic friction taking place. This friction creates a force, opposite to the direction of travel, and again proportional to the normal

  $$F_k=\mu_k F_N$$

Resting contact

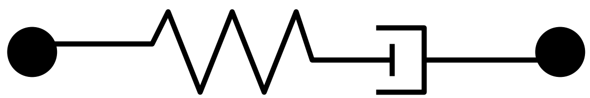
- It is difficult to compute forces due to the resting contact
- For each contact point, there is a force normal to the surface of contact
- All these forces have to be computed for all objects involved in resting contact
- For each contact point, a torque is also generated on it.
- If bodies have to rest, all those forces and torques have to be zero
- Solutions to this problem include quadratic programming, and are beyond the scope of this course

Constraints

- One problem occuring in animation is the fact that variables are not free.
- Constraints are usually set on objects and limit the field of the independent variables.
- There are two types of constraints:
  - hard constraints: strictly enforced
  - soft constraints: the system only attempts to satisfy them

Flexible objects

- To simulate elastic objects, Spring-mass-damper model is most used approach
- Springs: work with Hooke's law: the force applied is $F_{i,j}=-F_{j,i}=k_s(d_{i,j}(t)-len_{i,j})v_{i,j}$ where
  - d_{ij} distance between the two points
  - len_{ij} rest length of the spring
  - k_s spring constant
  - v_{ij} unit vector from point i to point j
- The flexible model is modelled as a net of points with mass and springs and dampers between them
- A damper can impart a force in the direction opposite to the velocity of the spring length and proportional to that velocity
  $F^d_i=-k_d\ v_i(t)$
- One can also introduce angular dampers and springs between faces
- Additional internal springs have often to be added to add stability to the system



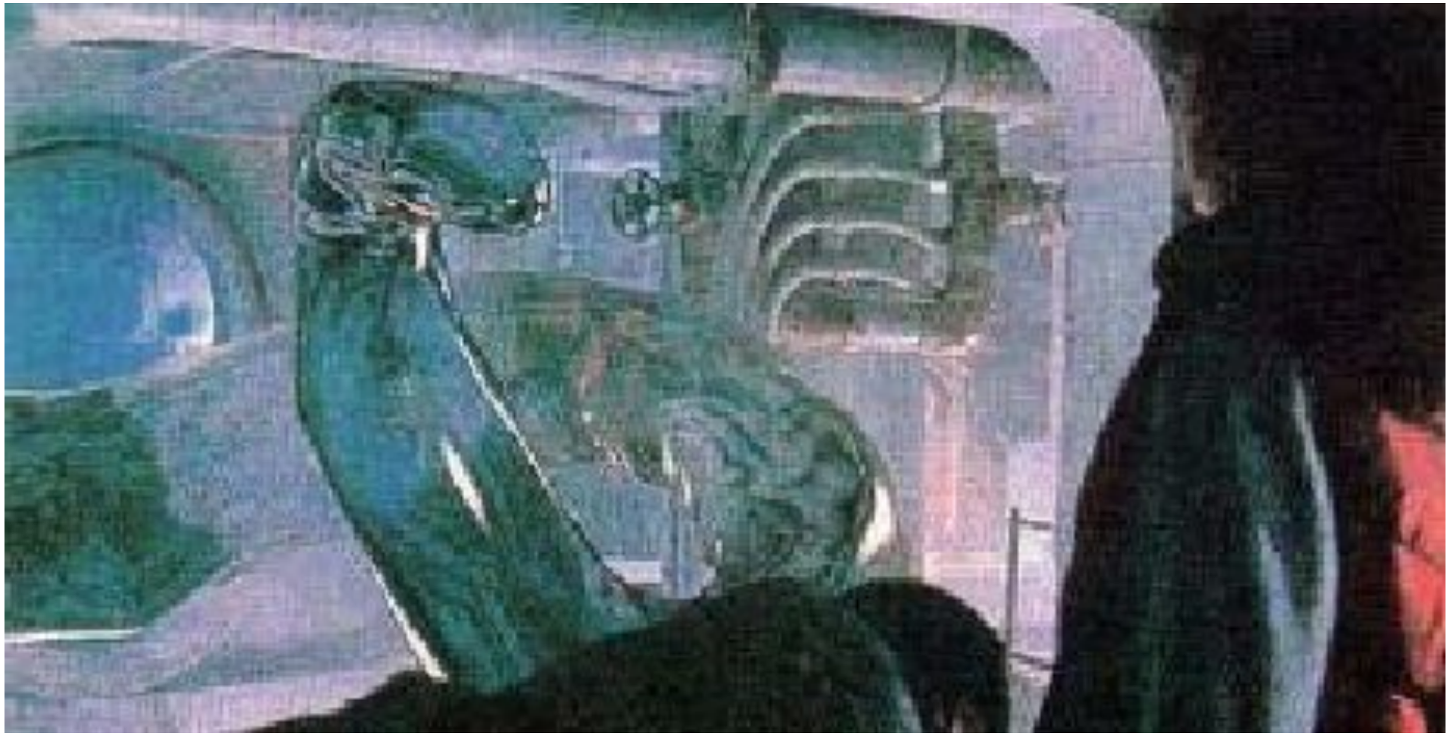
Virtual springs

- Induce forces that do not directly model physical elements
- For example, in the penalty method
- Sometimes one can use a proportional derivative controller which controls that a certain variable and speed is close to the desired value
- For example, this is used to keep the object close to the desired speed
- A virtual spring is added to keep things as desired

Energy minimization

- One can use energy to control the motion of the objects
- Energy constraints can be used to pin objects together, to restore the shape of an object, to minimize the curvature of a path or trajectory
- Energy constraints induce restoring forces on the system

End



+++ Ende - The end - Finis - Fin - Fine +++ Ende - The end - Finis - Fin - Fine +++