Particle Systems

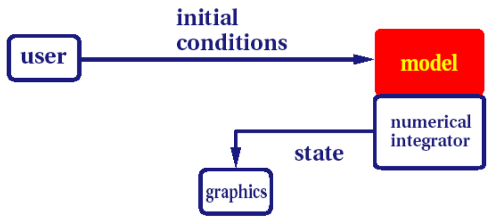
Thomas Funkhouser
Princeton University
C0S 526, Fall 2002

Kinematics and Dynamics

- Kinematics
  - Considers only motion
  - Determined by positions, velocities, accelerations
- Dynamics
  - Considers underlying forces
  - Compute motion from initial conditions and physics

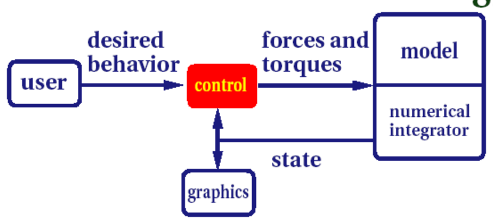
Passive vs. Active Dynamics

Passive––no muscles or motors



particle systems
leaves
water spray
clothing

Active––internal source of energy

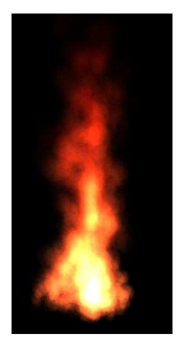


running human
trotting dog
swimming fish

Hodgins

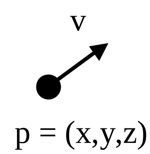
Passive Dynamics

- No muscles or motors
  - Smoke
  - Water
  - Cloth
  - Fire
  - Fireworks



Particle Systems

- A particle is a point mass
  - Mass
  - Position
  - Velocity
  - Acceleration
  - Color
  - Lifetime



- Use lots of particles to model complex phenomena
  - Keep array of particles

Particle Systems

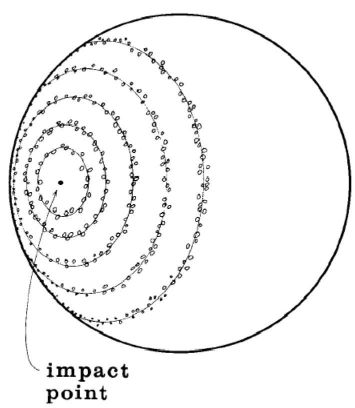
- For each frame:
  - Create new particles and assign attributes
  - Delete any expired particles
  - Update particles based on attributes and physics
  - Render particles

Creating/Deleting Particles

- Where to create particles?
  - Around some center
  - Along some path
  - Surface of shape
  - Where particle density is low
- When to delete particles?
  - Where particle density is high
  - Life span
  - Random

This is where user controls animation

Example: Wrath of Khan



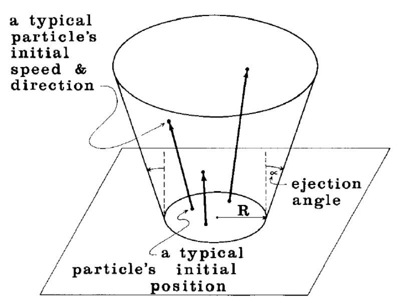
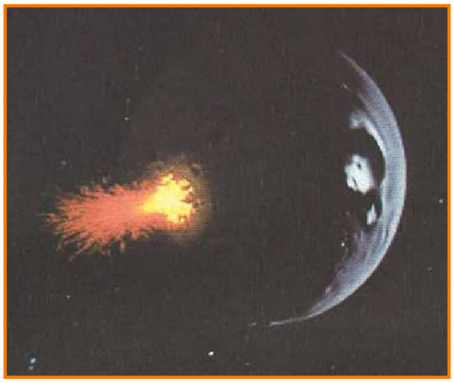


Fig. 2. Distribution of particle systems on the planet's surface.

Reeves

Example: Wrath of Khan



Reeves

Example: Wrath of Khan

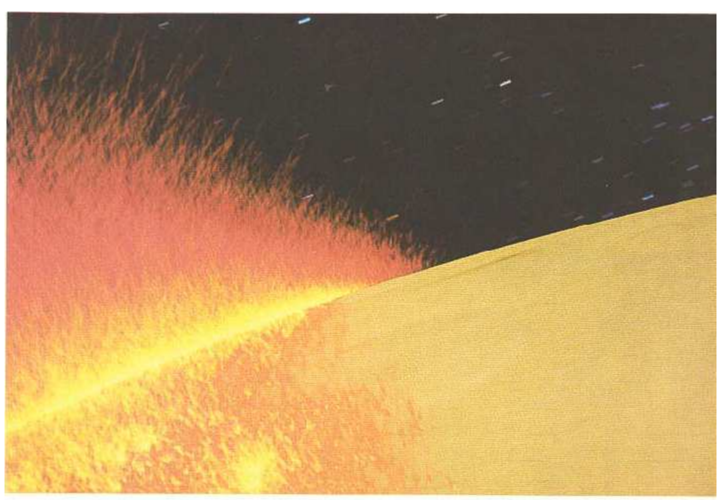


Fig. 7. Wall of fire about to engulf camera.

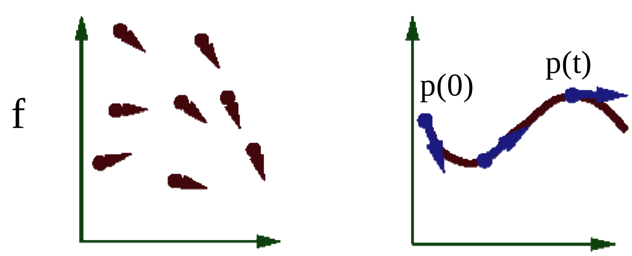
Reeves

Equations of Motion

- Newton's Law for a point mass
  - f = ma
- Update every particle for each time step
  - $a(t+\Delta t) = g$
  - $v(t+\Delta t) = v(t) + a(t)*\Delta t$
  - $p(t+\Delta t) = p(t) + v(t)*\Delta t + a(t)^2*\Delta t/2$

Solving the Equations of Motion

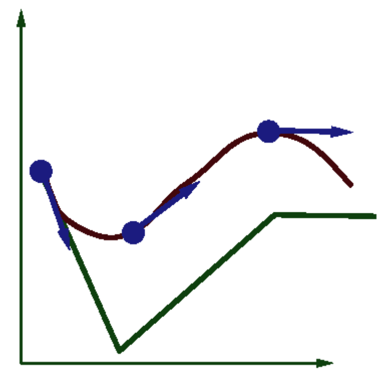
- Initial value problem
  - Know p(0), v(0), a(0)
  - Can compute force at any time and position
  - Compute p(t) by forward integration



Hodgins

Solving the Equations of Motion

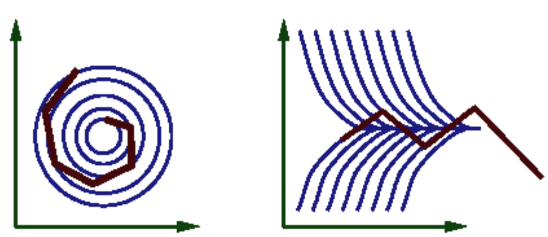
- Euler integration
  - p(t+Δt)=p(t) + Δt f(x,t)



Hodgins

Solving the Equations of Motion

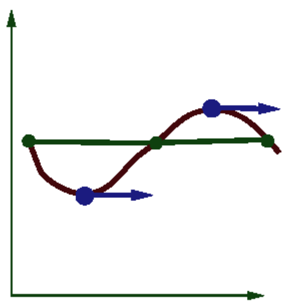
- Euler integration
  - p(t+Δt)=p(t) + Δt f(x,t)
- Problem:
  - Accuracy decreases as Δt gets bigger



Hodgins

Solving the Equations of Motion

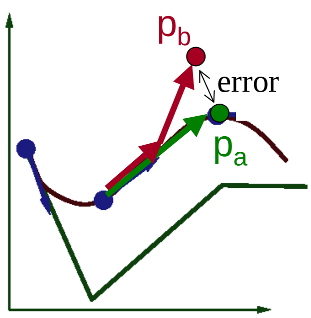
- Midpoint method (2nd order Runge-Kutta)
  - Compute an Euler step
  - Evalute f at the midpoint
  - Take an Euler step using midpoint force
    - » $p(t+\Delta t)=p(t) + \Delta t\ f(\ p(t) + 0.5^*\Delta t\ f(t),t)$



Hodgins

Solving the Equations of Motion

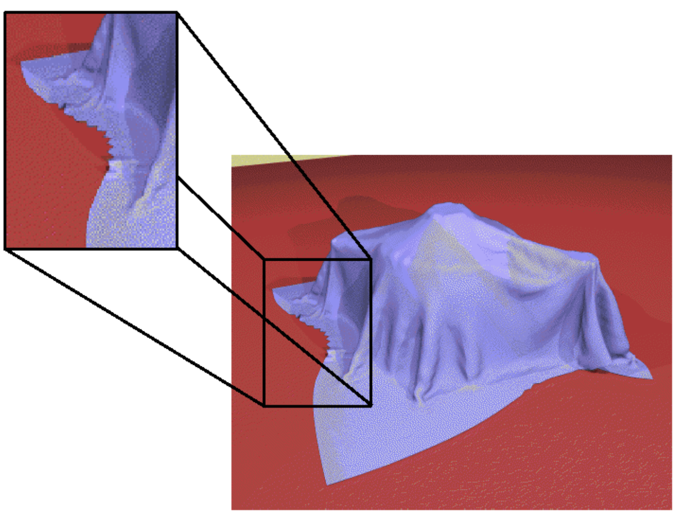
- Adapting step size
  - Compute p_a by taking one step of size h
  - Compute p_b by taking 2 steps of size h/2
  - Error = $|\ p_a - p_b\ |$
  - Adjust step size by factor $(\text{epsilon}/\text{error})^{1/f}$



Particle System Forces

- Force fields
  - Gravity, wind, pressure
- Viscosity/damping
  - Liquids, drag
- Collisions
  - Environment
  - Other particles
- Other particles
  - Springs between neighboring particles (mesh)
  - Useful for cloth

Example: Cloth



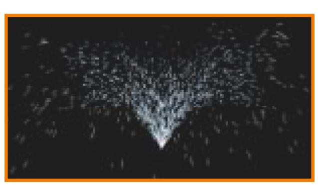
Breen

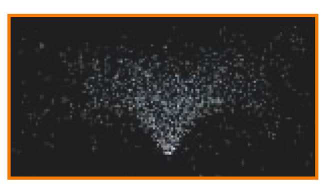
Example: Grass

Fig. 12. *white.sand.*

Rendering Particles

- Volumes
  - Ray casting, etc.
- Points
  - Render as individual points
- Line segments
  - Motion blur over time





Summary

- Particle systems
  - Lots of particles
  - Simple physics
- Interesting behaviors
  - Waterfalls
  - Smoke
  - Fire
  - Cloth
- Solving motion equations
  - Don’t use Euler integration
  - Do use adaptive step sizes