

Fall 2015

CSCI 420 Computer Graphics

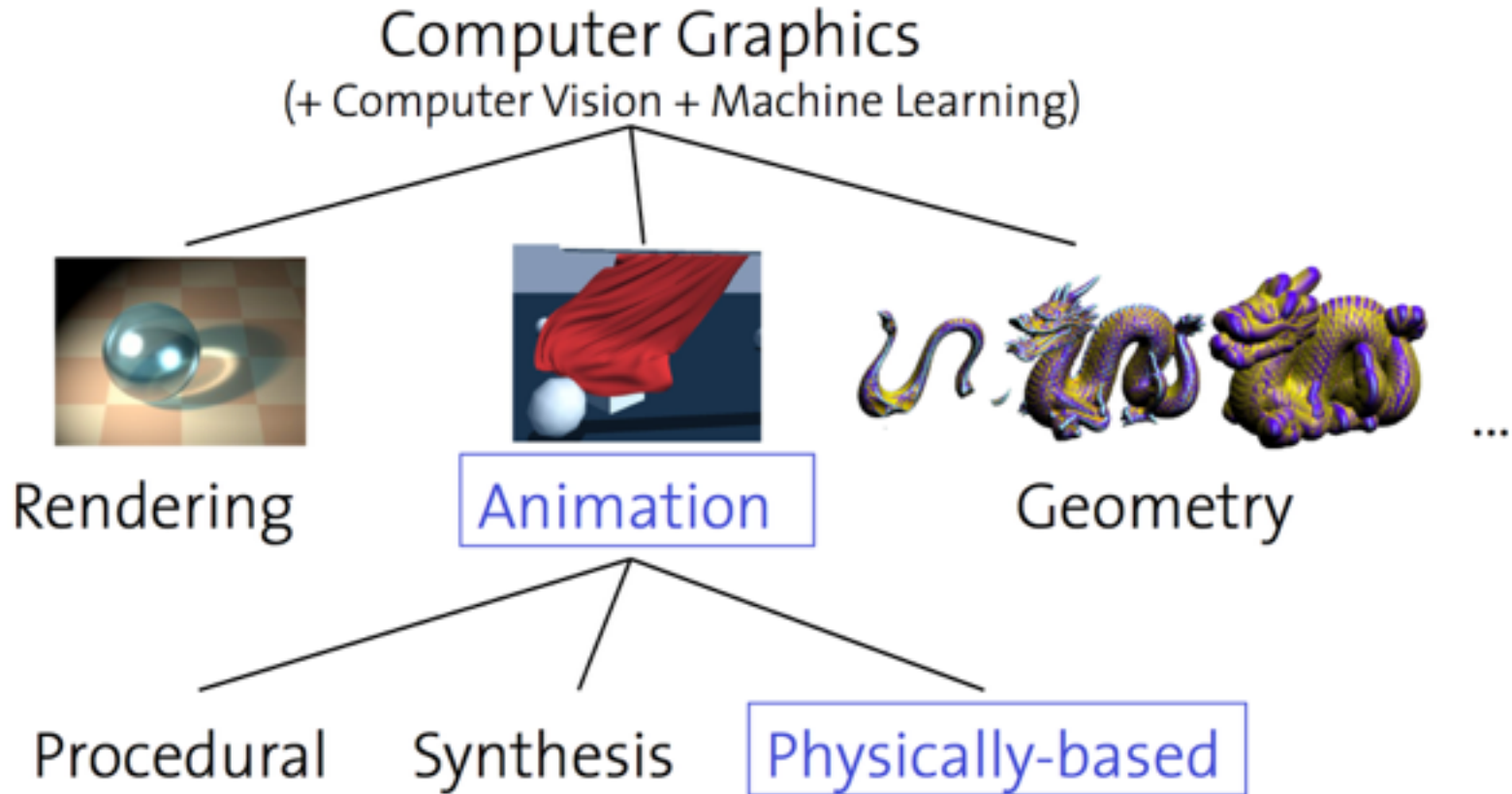
13.2 Physically Based Simulation I



Hao Li

<http://cs420.hao-li.com>

Visual Computing



Animation

- Animation from *anima* (lat.)
= *soul, spirit, breath of life*
- Bring images to life!
- Examples
 - Character animation
(humans, animals)
 - Secondary motion (hair, cloth)
 - Physical world (rigid bodies, water, fire)

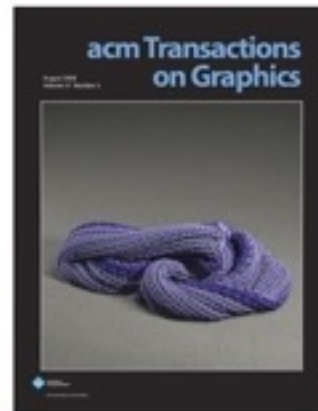
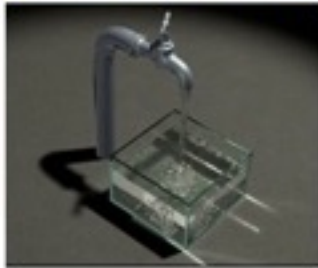
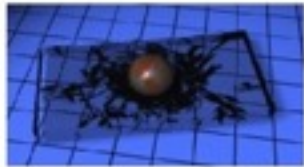


Animation Techniques

- For character animation
 - Keyframing
 - Motion capturing / motion synthesis
- For secondary motion, physical effects
 - Procedural
 - Simulation (physically based animation)

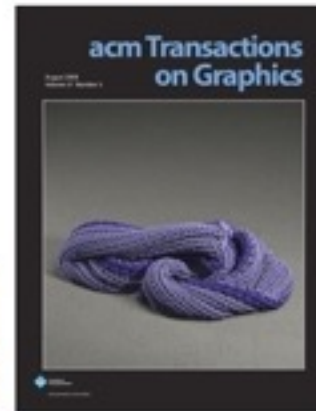
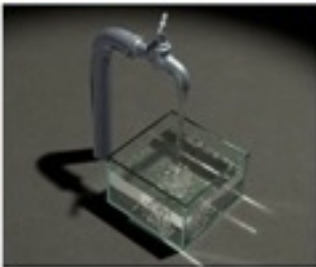
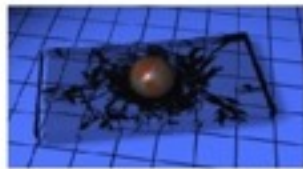
Physics in Computer Graphics

- Very common
- Computer Animation, Modeling (computational mechanics)
- Rendering (computational optics)



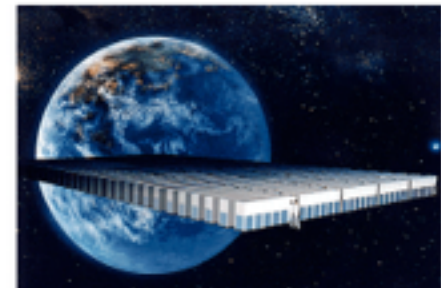
Physics in Computer Animation

- Fluids
- Smoke
- Deformable strands (rods)
- Cloth
- Solid 3D deformable objects and many more!



Physical Simulation

- Equations known for a long time
 - Motion (Newton, 1660) $d/dt(m\mathbf{v}) = \mathbf{f}$
 - Elasticity (Hooke, 1670) $\boldsymbol{\sigma} = \mathbf{E}\boldsymbol{\varepsilon}$
 - Fluids (Navier, Stokes, 1822) $\rho\left(\frac{\partial\mathbf{v}}{\partial t} + \mathbf{v}\cdot\nabla\mathbf{v}\right) = -k\nabla\rho + \rho\mathbf{g} + \mu\nabla^2\mathbf{v}$
- Simulation made possible by computers
 - 1938: Zuse 1, 0.2 flops,
 - 2008: Roadrunner, 122k cores, 1026 teraflops



Scientific Goals and Challenges

- Goal of scientific computations
 - **Reproduction** of physical phenomena
 - **Substitute** for real experiments
 - Goal of physically-based animation
 - **Imitation** of physical phenomena
 - **Visually plausible** behavior
 - As much realism as possible within **performance** and **stability constraints**
- Different goals require different methods/
representations...

Offline Physics

- Special effects (film, commercials)
- Large models:
millions of particles / tetrahedra / triangles
- Use computationally expensive rendering
(global illumination)
- Impressive results
- Many seconds of computation time per frame

Real-time Physics

- Interactive systems:
computer games
virtual medicine (surgical simulation)
- Must be fast (30 fps, preferably 60 fps for games)
Only a small fraction of CPU time devoted to physics!
- Has to be stable, regardless of user input



Flight/car Simulators



Surgery trainers

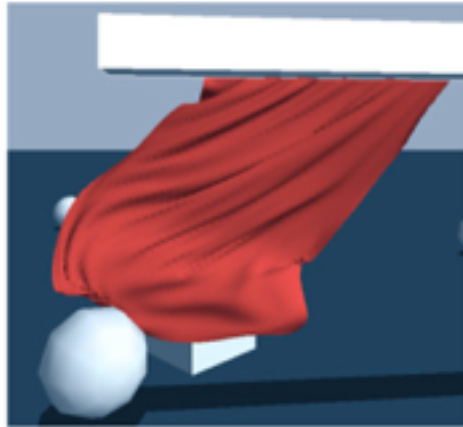


3D Games

Examples



Rigid

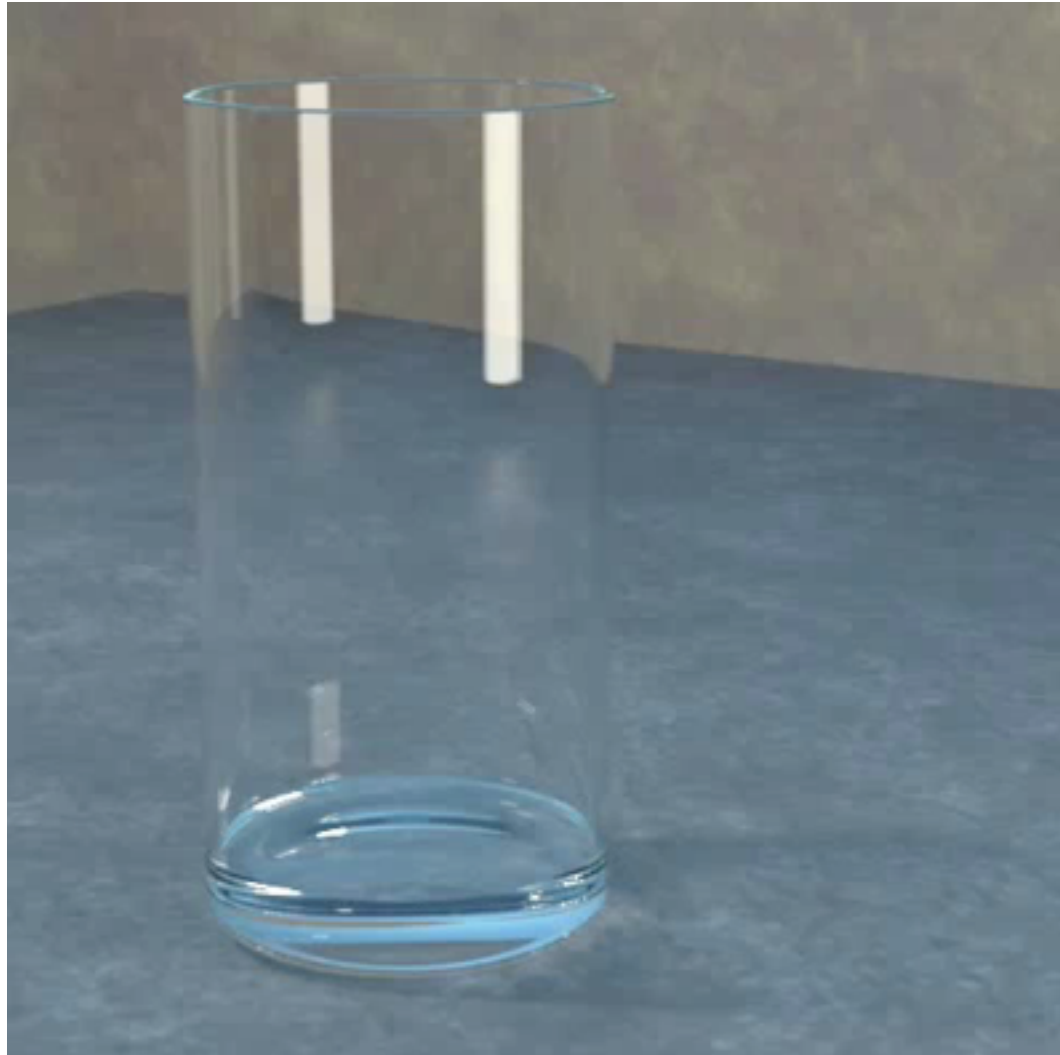


Deformable



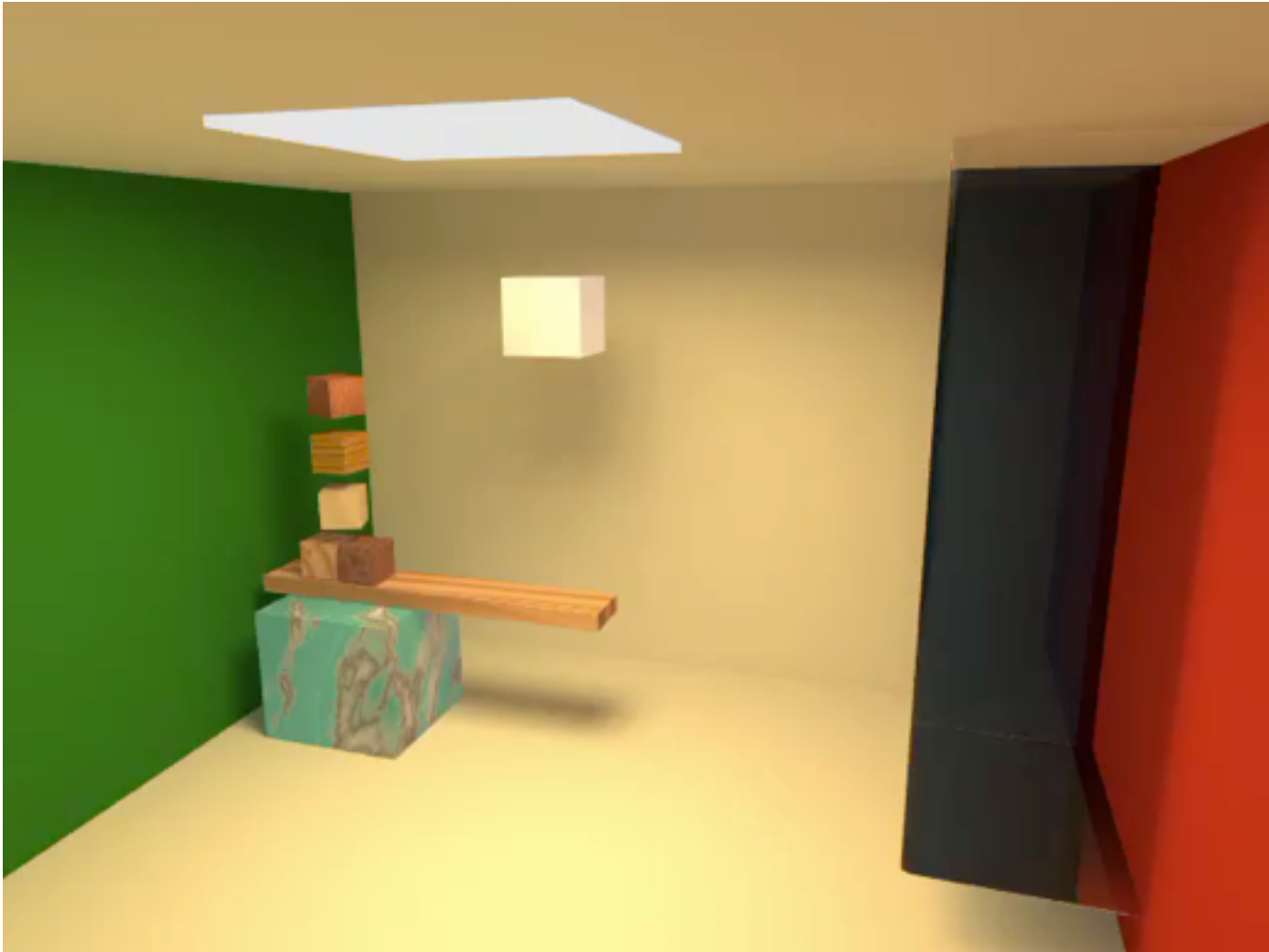
Liquid

Fluids



Enright, Marschner,
Fedkiw,
SIGGRAPH 2002

Fluids and Rigid Bodies



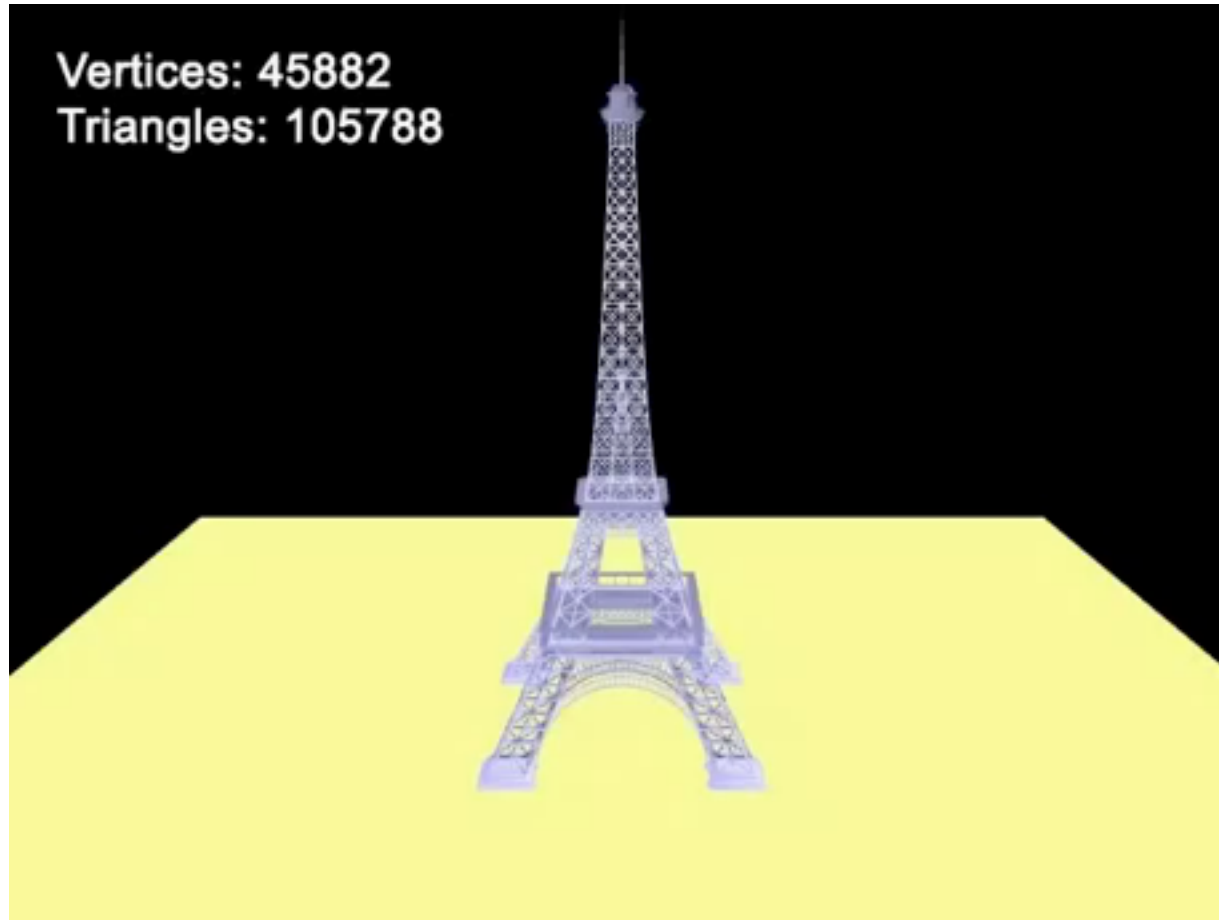
Fluids with Deformable Solid

[Robinson-Mosher,
Shinar,
Gretarsson,
Su, Fedkiw,
SIGGRAPH 2008]

Two-way Coupling of Fluids to Rigid and Deformable Solids and Shells

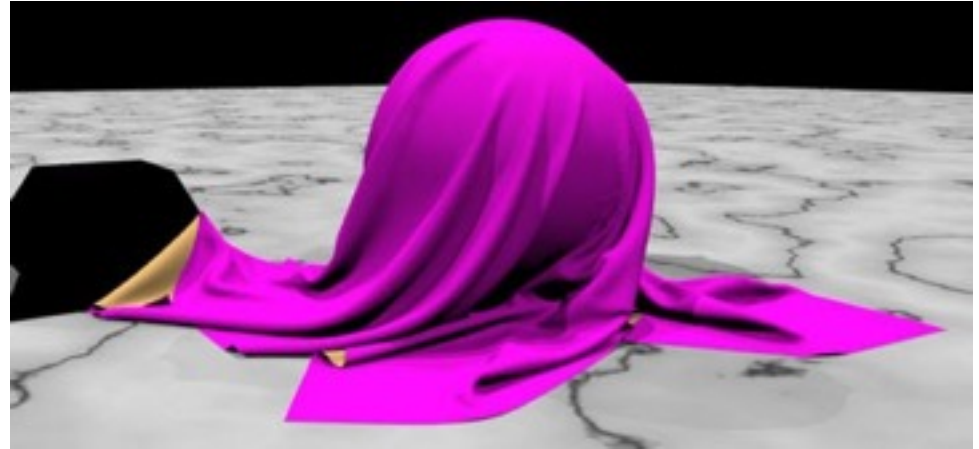
**Avi Robinson-Mosher
Tamar Shinar
Jon Gretarsson
Jonathan Su
Ronald Fedkiw**

Deformations



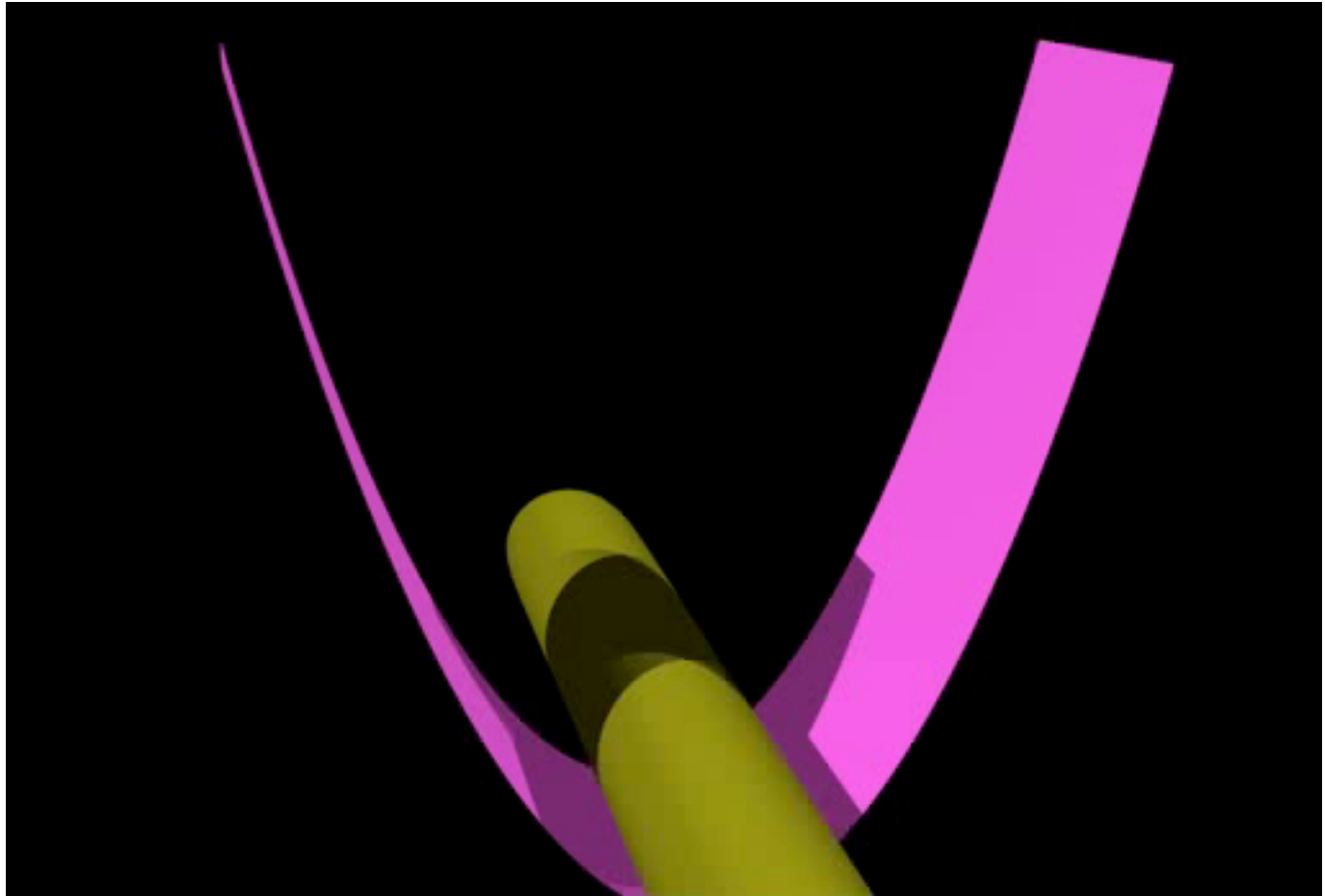
[Barbic and James,
SIGGRAPH 2005]

Cloth



Source:
ACM SIGGRAPH

Cloth (Robustness)

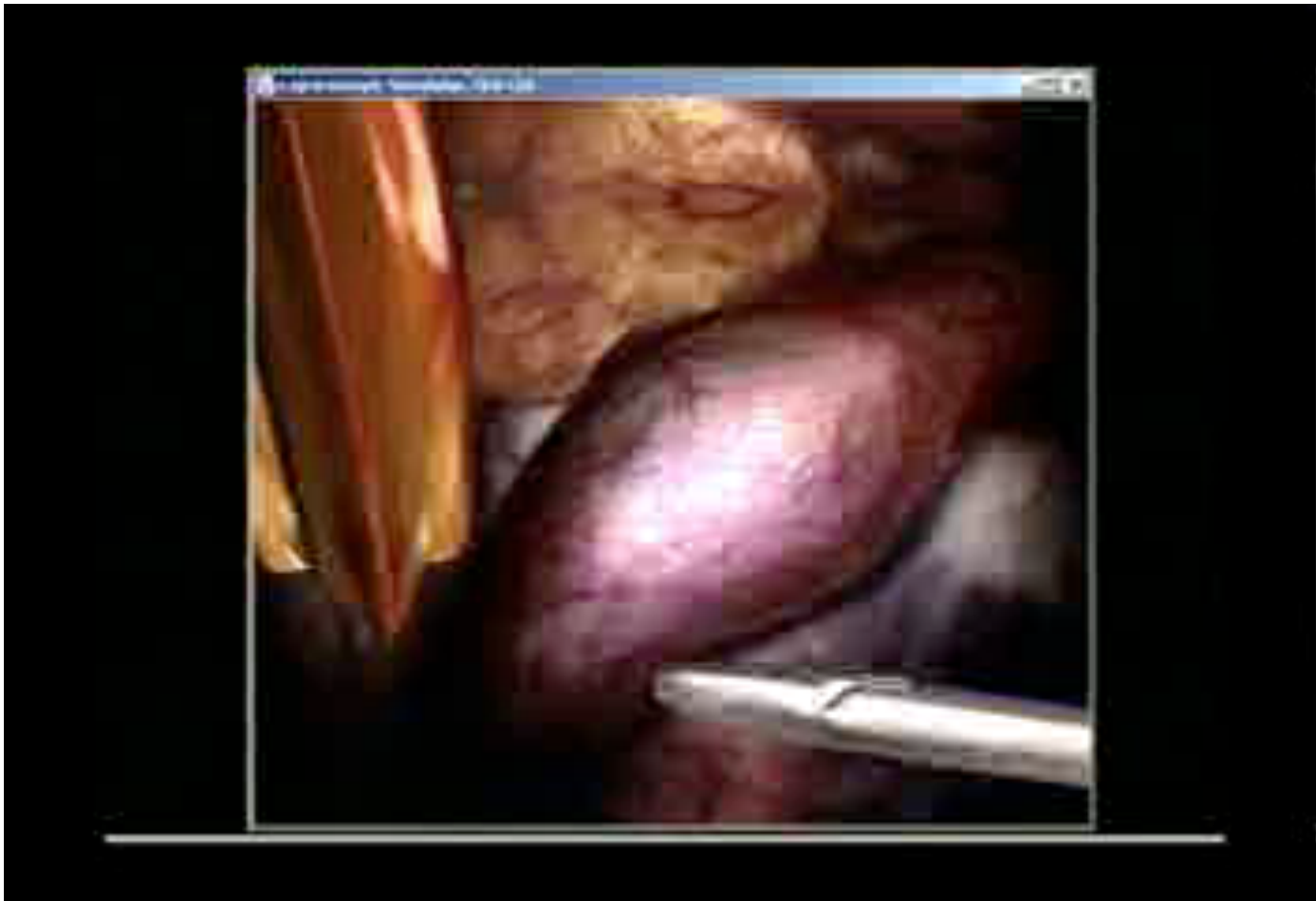


[Bridson, Fedkiw,
Anderson, ACM
SIGGRAPH 2002

Multibody Dynamics + Self-collision Detection



Surgical Simulation



Multibody Dynamics

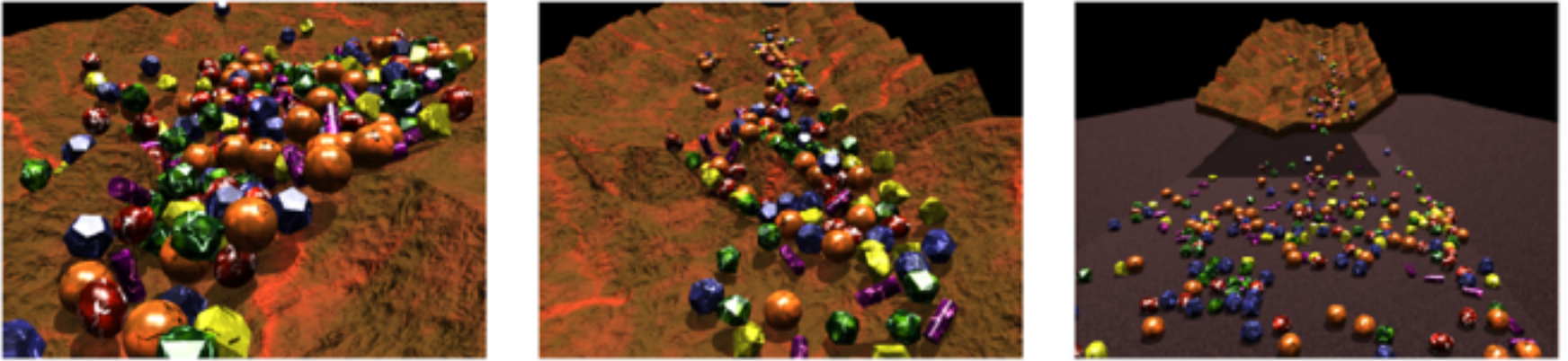


Figure 1: *Avalanche*: 300 rocks tumble down a mountainside.

Physics in Games

Real-Time Deformation and Fracture
in a Game Environment

Eric Parker
Pixelux Entertainment

James O'Brien
U.C. Berkeley

Video Edited by Sebastian Burke

From the proceedings of SCA 2009, New Orleans

Sound Simulation (Acoustics)



[James, Barbic, Pai,
SIGGRAPH 2006]

Techniques

- Particle systems
 - Fire, smoke, water ...
- Mass-spring systems
 - Deformable objects, cloth ...
- Rigid body simulation
 - Cars, airplanes, furniture ...
- Grid based methods
 - Water, smoke, airflow ...
- Finite Elements
 - Accurate deformable objects ...

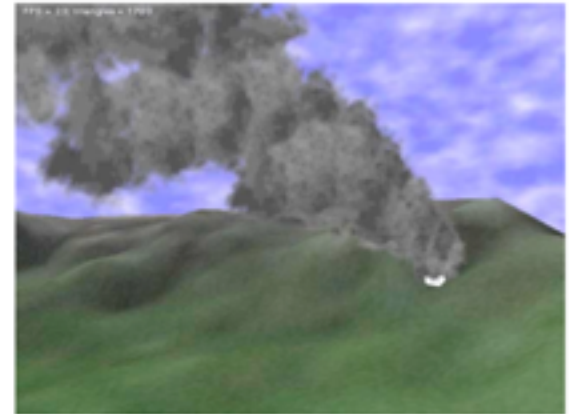
Particle System



Snow, dust, sand

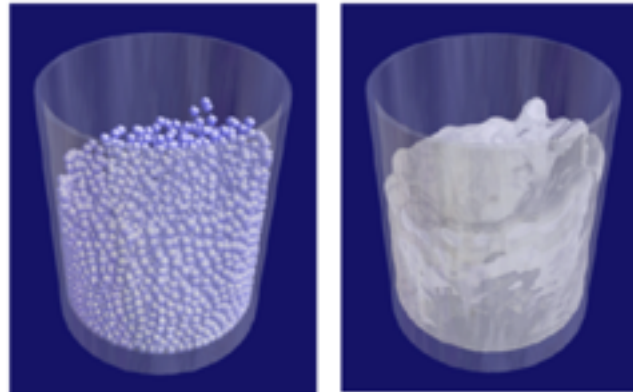


Fire



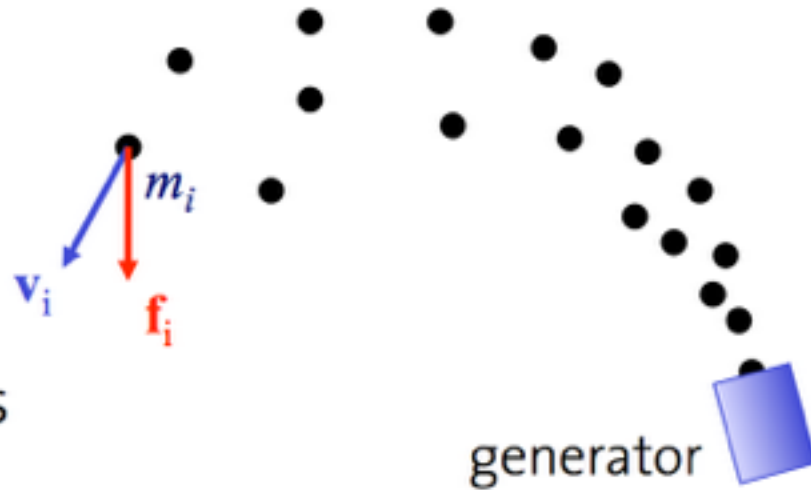
Smoke

Water



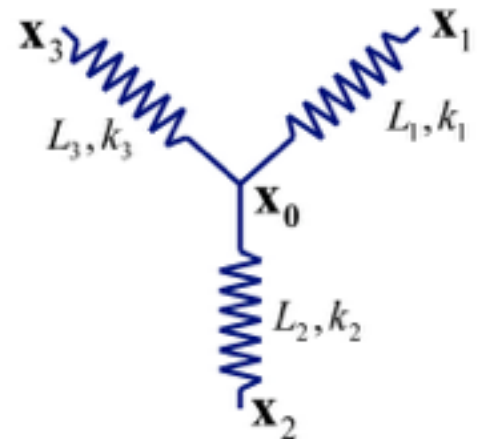
Particle System

- Collection of many small simple particles
- Particle motion influenced by forces
- Generated by emitters
- Deleted when lifetime reached or out of scene



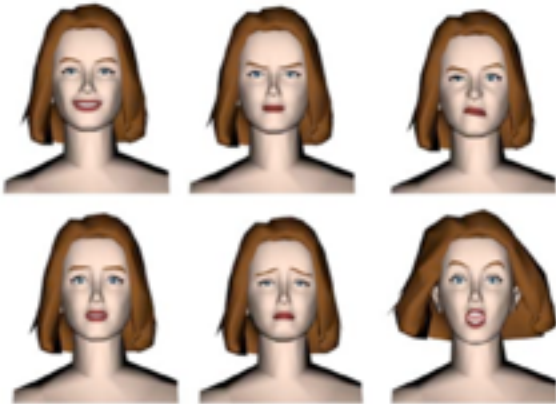
Mass-Spring Systems

- Particle system + springs
- Special interaction force
- Issues:
 - Where to put springs
 - Choice of stiffnesses
 - Collision detection
 - Collision response
 - Stability (time step or stiffness too high)



Applications

Facial animation



Thalmann

Cloth simulation



Strasser

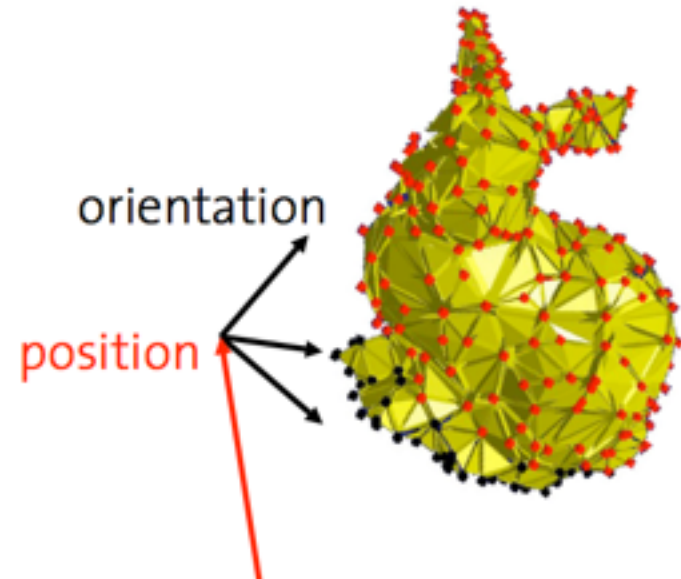
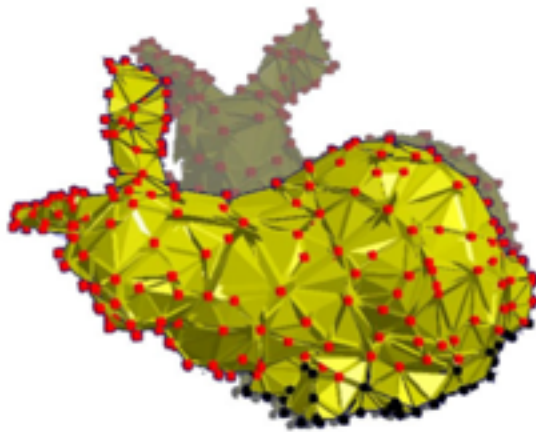
Surgery simulation



Kuehnappel

Rigid Body Simulation

- Deformable objects have many degrees of freedom
- Each vertex is simulated separately
- A rigid body only has 6 degrees of freedom
- Faster simulation possible



Challenges

- Collision detection
- Collision response for complex configurations
- Constraints (joints)

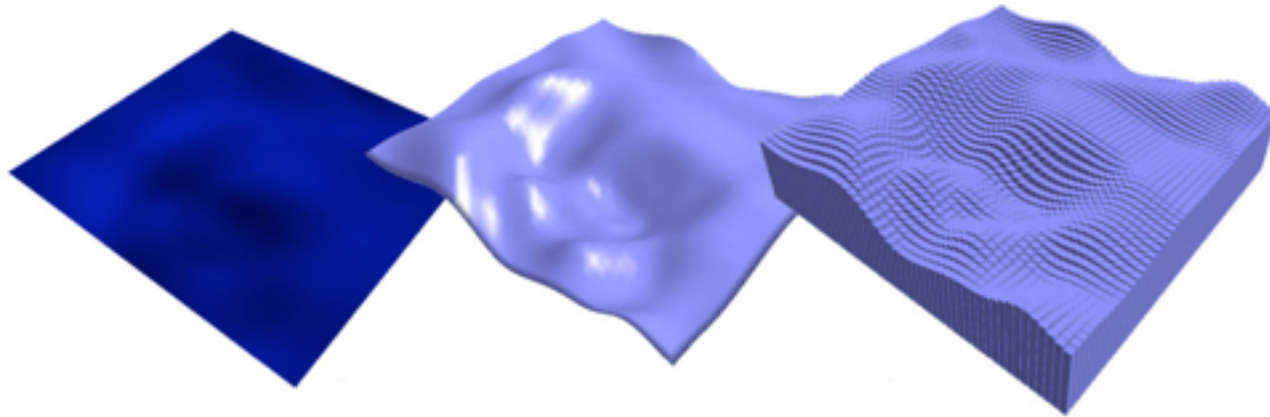


Applications

- Robotic simulations
- 3D computer games



Grid-Based Methods

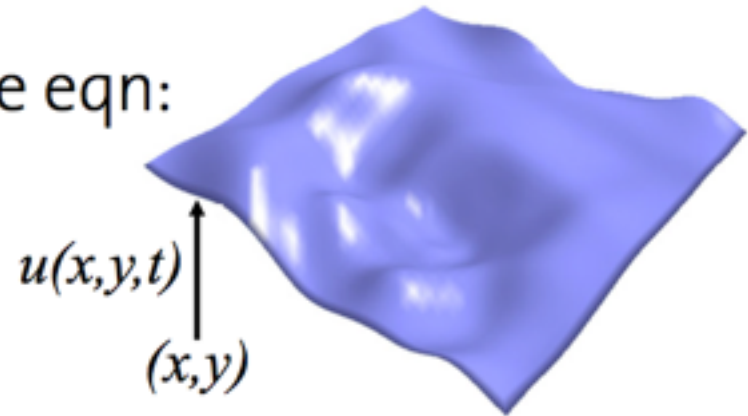


- Basic idea:
 - Solve partial differential equation on (regular) grid
 - Replace differentials by finite differences

Example: Fluid Surface

- Water surface defined as height $u(x,y,t)$ at location x,y at time t
- Dynamics given by 2D wave eqn:

$$\frac{\partial^2}{\partial t^2} u = c^2 \left(\frac{\partial^2}{\partial x^2} u + \frac{\partial^2}{\partial y^2} u \right)$$



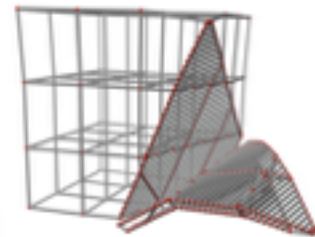
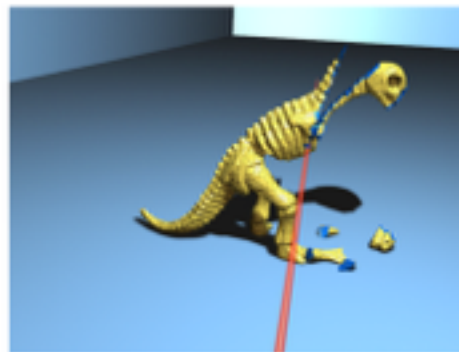
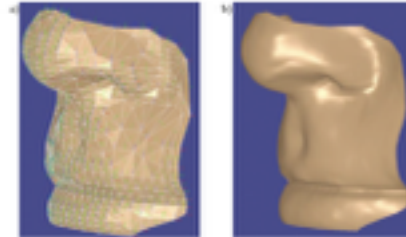
- Discretization:

$$v^{t+1}[i, j] = v^t[i, j] + \Delta t c^2 \frac{u^t[i+1, j] + u^t[i-1, j] + u^t[i, j+1] + u^t[i, j-1] - 4u^t[i, j]}{h^2}$$

$$u^{t+1}[i, j] = u^t[i, j] + \Delta t v^{t+1}[i, j]$$

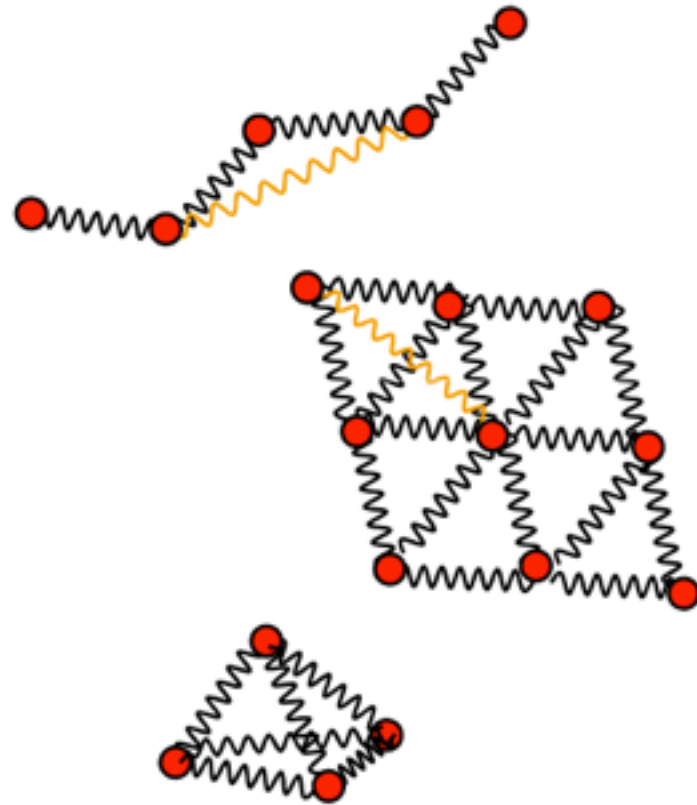
FEM Simulation

- Discretize equations from continuum mechanics
- Solve (more) accurately
- Independent of tessellation
- Volumetric meshes



Case Study: Mass-spring

- Mass particles



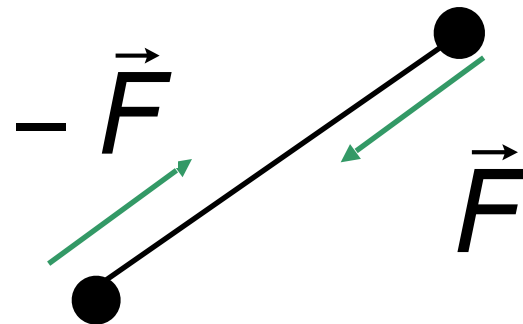
Source:Matthias Mueller, SIGGRAPH

Newton's Laws

- Newton's 2nd law:

$$\vec{F} = m\vec{a}$$

- Gives acceleration, given the force and mass
- Newton's 3rd law: If object A exerts a force F on object B, then object B is at the same time exerting force $-F$ on A

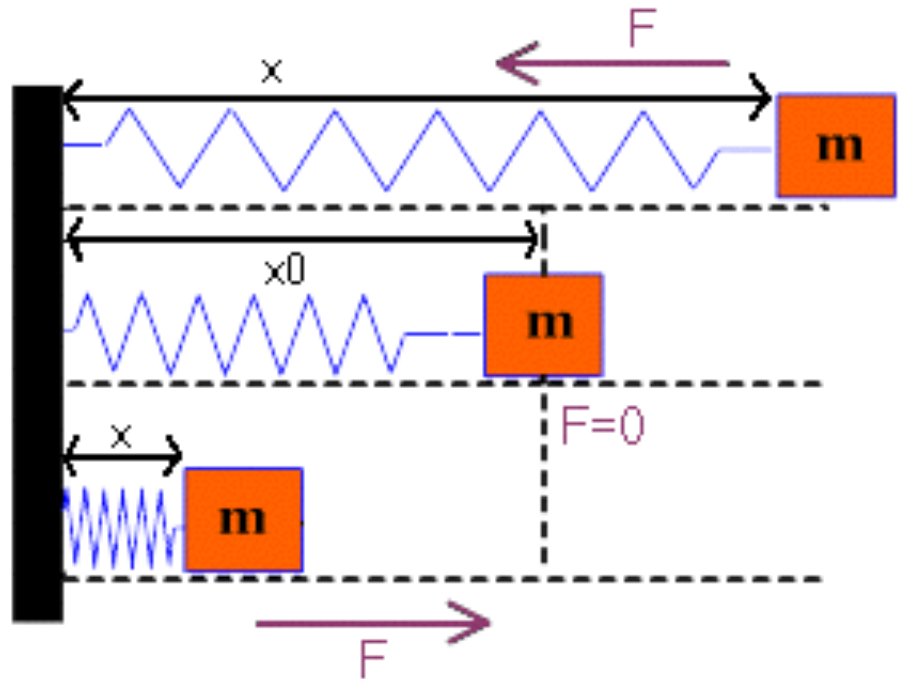


Single spring

- Obeys the *Hook's law*:

$$F = k (x - x_0)$$

- x_0 = rest length
- k = spring elasticity (*stiffness*)
- For $x < x_0$, spring wants to extend
- For $x > x_0$, spring wants to contract



Hook's law in 3D

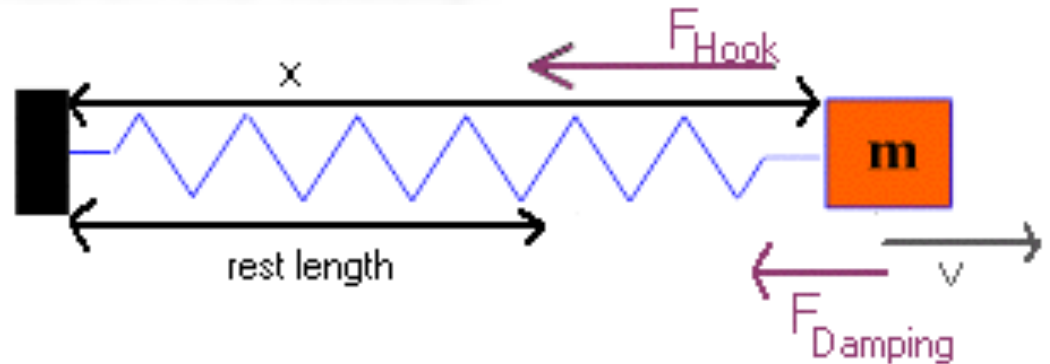
- Assume A and B two mass points connected with a spring.
- Let \vec{L} be the vector pointing from B to A
- Let R be the spring rest length
- Then, the elastic force exerted on A is:

$$\vec{F} = -k_{Hook} (|\vec{L}| - R) \frac{\vec{L}}{|\vec{L}|}$$

Damping

- Springs are not completely elastic
- They absorb some of the energy and tend to decrease the velocity of the mass points attached to them
- Damping force depends on the velocity:

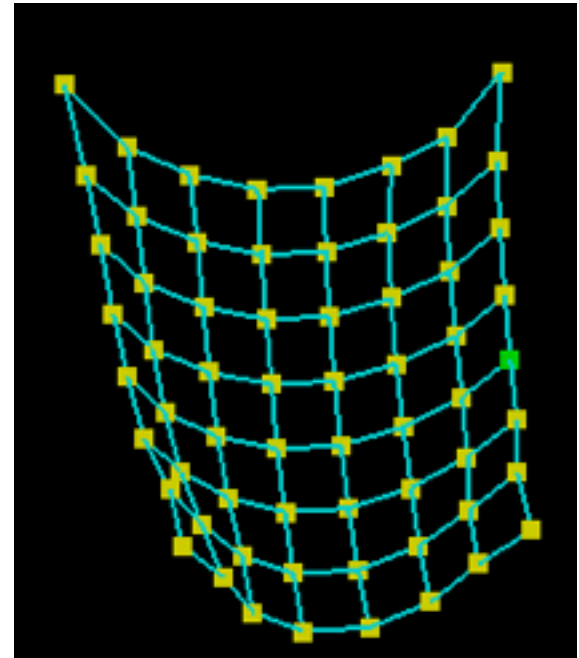
$$\vec{F} = -k_d \vec{v}$$



- k_d = damping coefficient
- k_d different than k_{Hook} !!

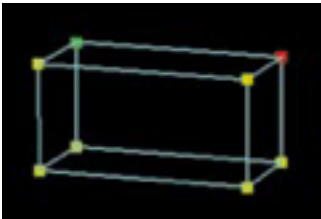
A network of springs

- Every mass point connected to some other points by springs
- Springs exert forces on mass points
 - Hook's force
 - Damping force
- Other forces
 - External force field
 - Gravity
 - Electrical or magnetic force field
 - Collision force

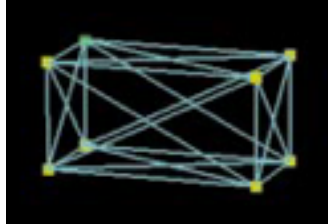


Network organization is

- For stability, must organize the network of springs in some clever way



Basic network

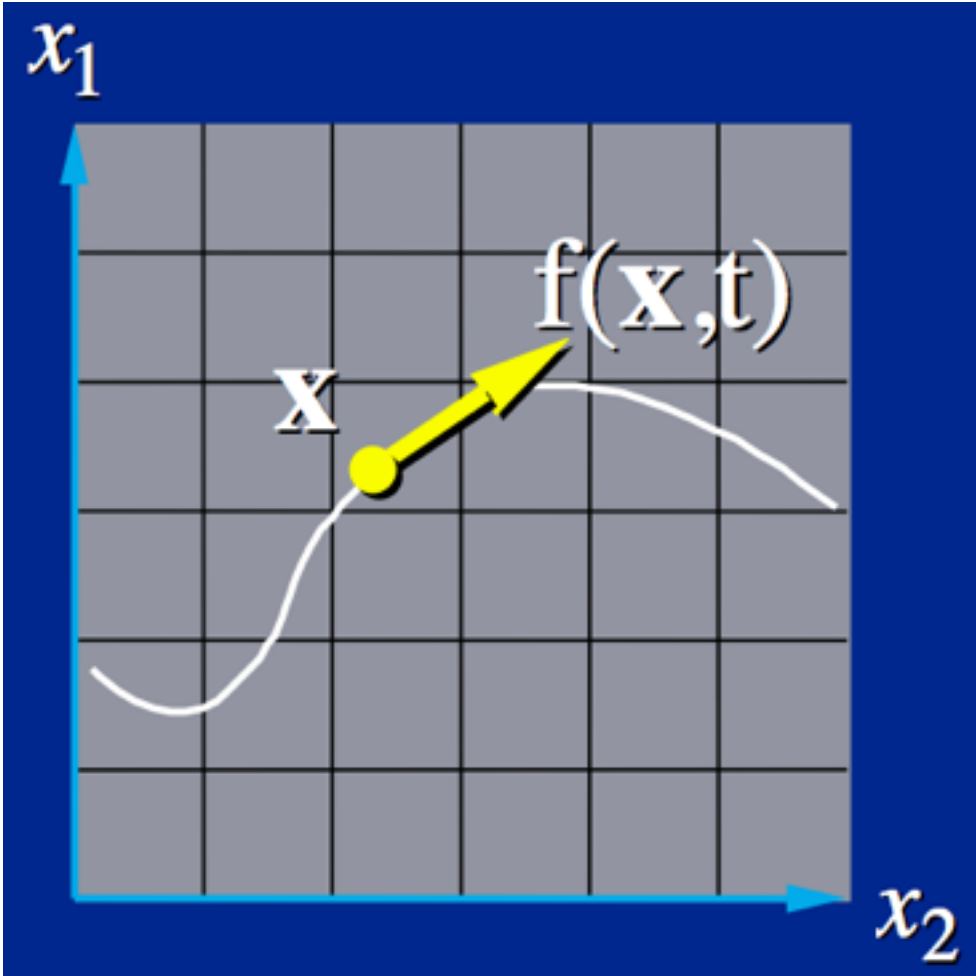


Stable network



Network out
of control

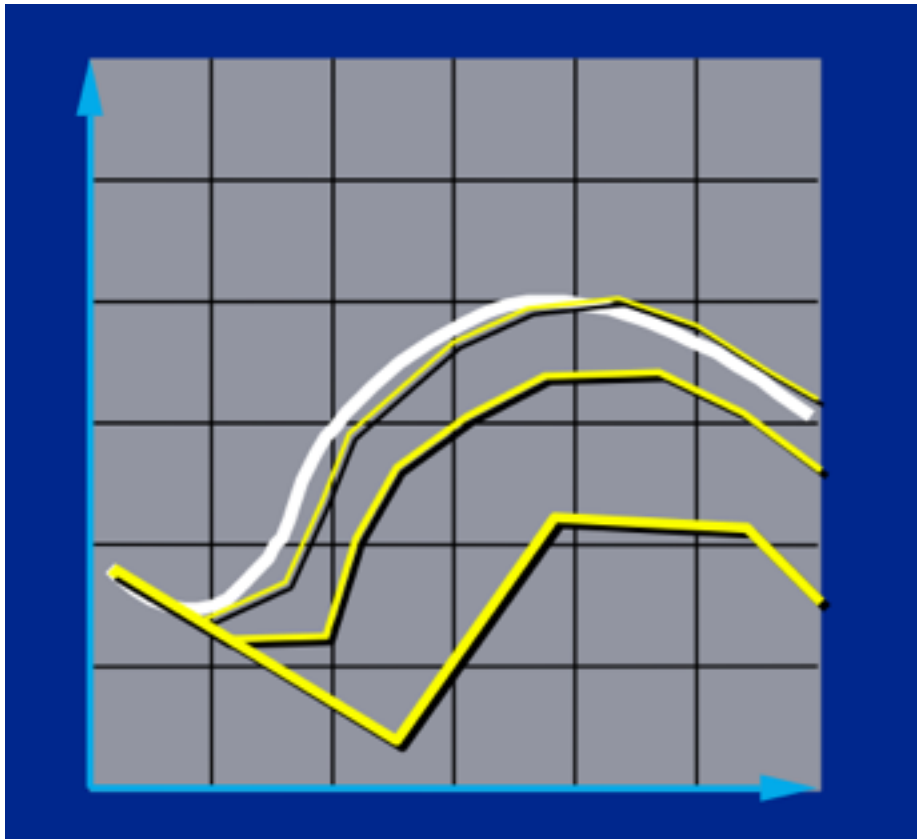
Time Integration



Physics equation:
 $\mathbf{x}' = f(\mathbf{x},t)$

$\mathbf{x}=\mathbf{x}(t)$ is particle trajectory

Euler Integration



Simple,
but inaccurate.

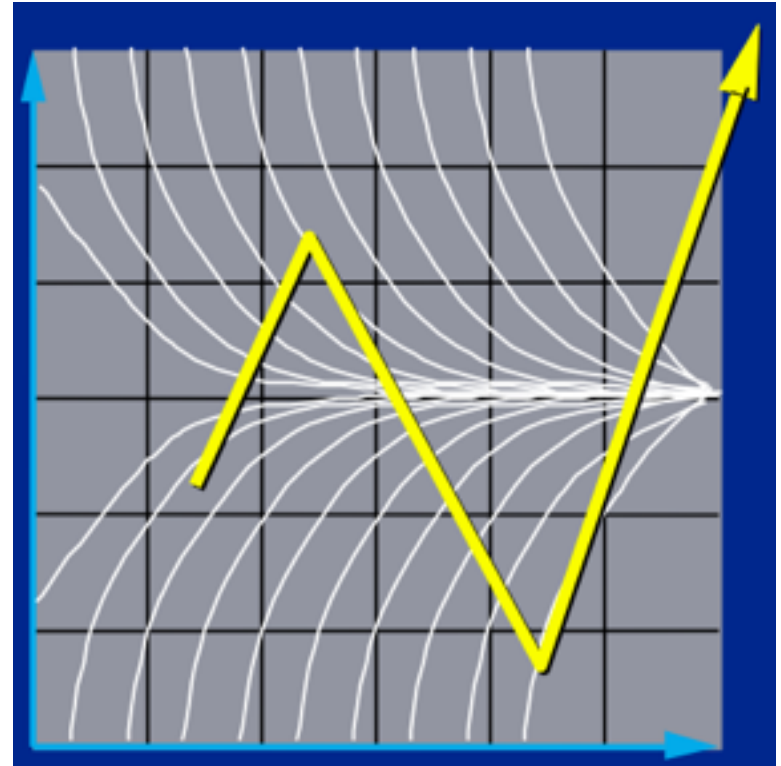
Unstable with
large timesteps.

Source: Andy Witkin, SIGGRAPH

Inaccuracies with explicit

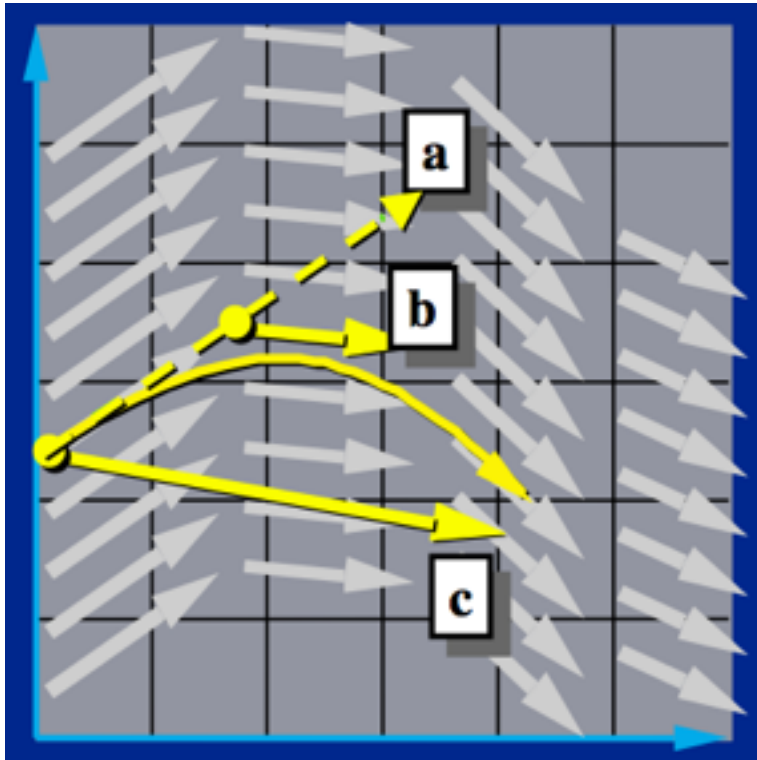


Gain energy



“Blow-up”

Midpoint Method



Source: Andy Witkin, SIGGRAPH

Improves stability

1. Compute Euler step
 $\Delta x = \Delta t f(x, t)$
2. Evaluate f at the midpoint
 $f_{\text{mid}} = f((x+\Delta x)/2, (t+\Delta t)/2)$
3. Take a step using the midpoint value
 $x(t + \Delta t) = x(t) + \Delta t f_{\text{mid}}$

Many more methods

- Runge-Kutta (4th order and higher orders)
- Implicit methods
 - sometimes unconditionally stable
 - very popular (e.g., cloth simulations)
 - a lot of damping with large timesteps
- Symplectic methods
 - exactly preserve energy, angular momentum and/or other physical quantities
 - Symplectic Euler

Cloth Simulation

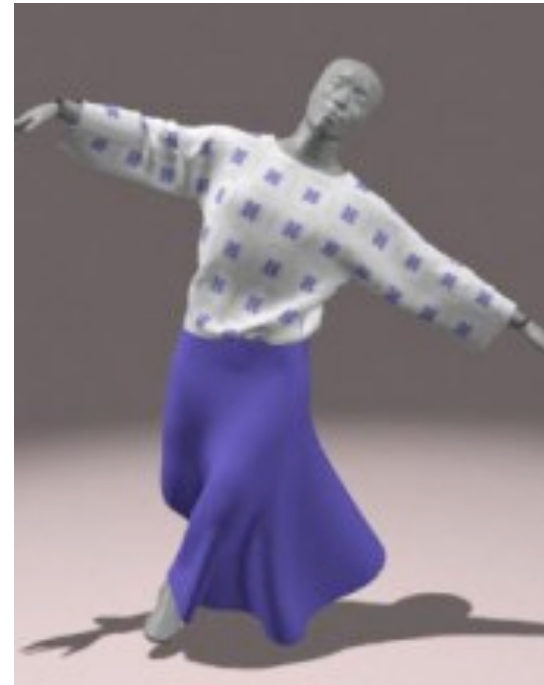
- Stretch



- Shear



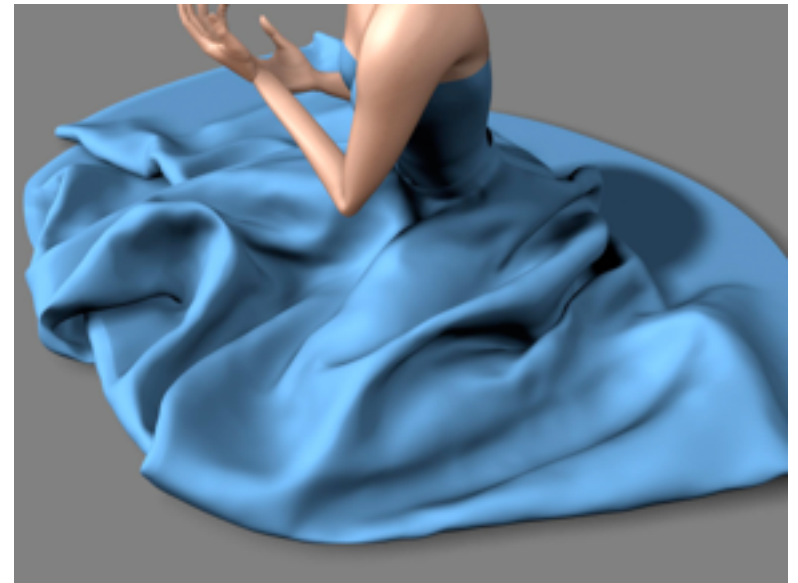
- Bend



[Baraff and Witkin,
SIGGRAPH 1998]

Challenges

- Complex Formulas
- Large Matrices
- Stability
- Collapsing triangles
- Self-collision detection

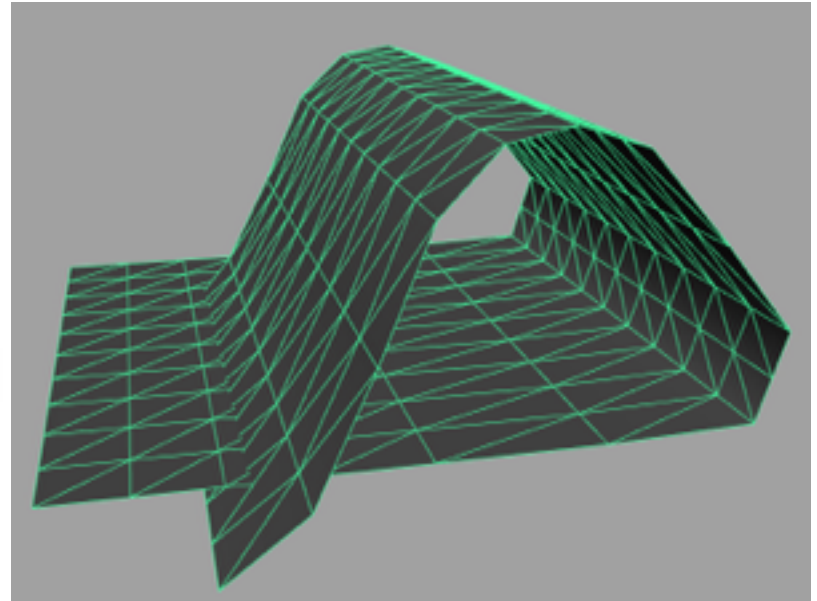


[Govindaraju et al. 2005]

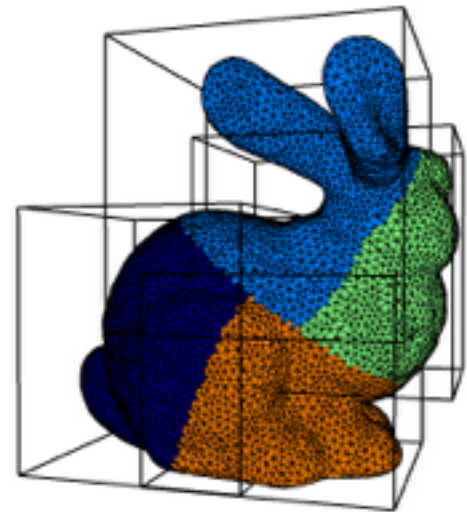
Self-collisions: definition

Deformable model is
self-colliding iff

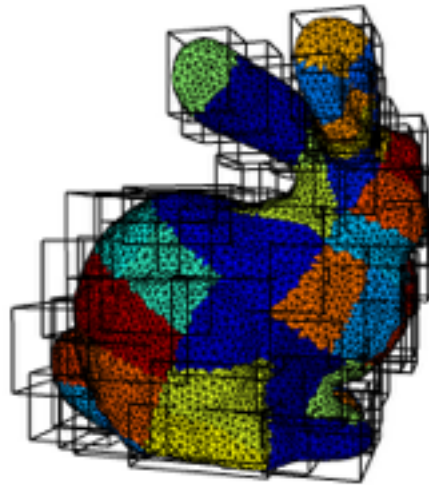
there exist non-neighboring
intersecting triangles.



Bounding volume hierarchies



AABBs
Level 1



AABBs
Level 3

[Hubbard 1995]

[Gottschalk et al. 1996]

[van den Bergen 1997]

[Bridson et al. 2002]

[Teschner et al. 2002]

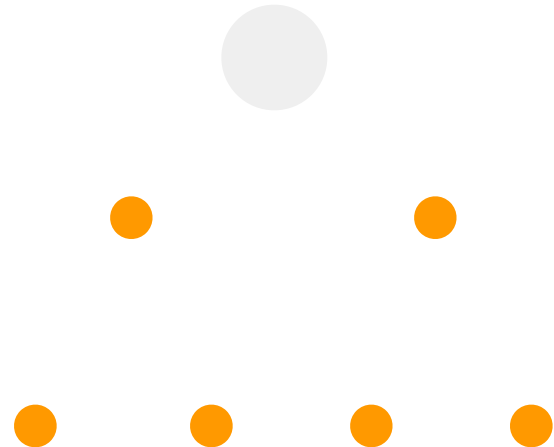
[Govindaraju et al. 2005]

Bounding volume hierarchy

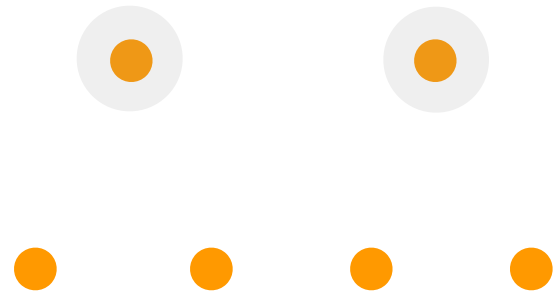
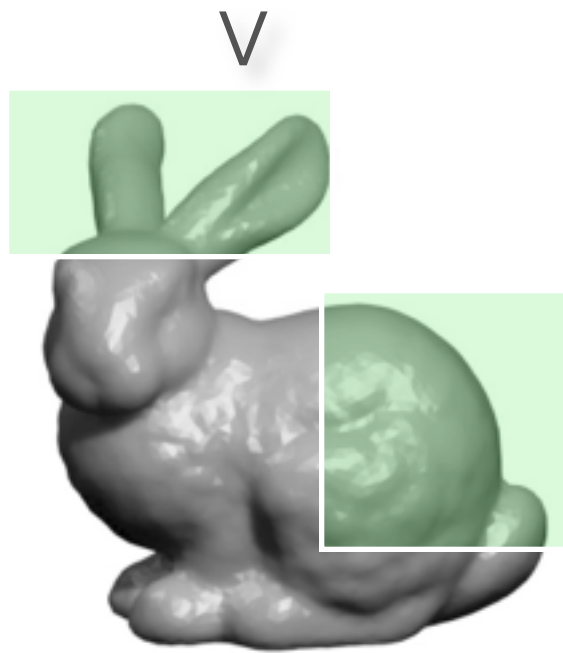
root



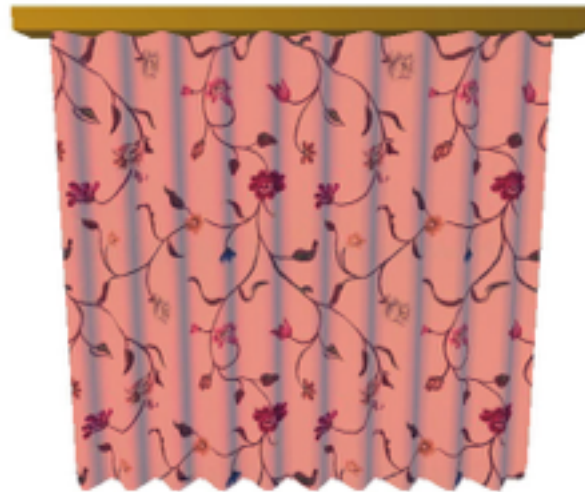
root



Bounding volume hierarchy



Real-time cloth simulation



Source:
Andy Pierce

Model	Triangles	FPS	% Forces + Stiffness Matrix	% Solver
Curtain	2400	25	67	33

<http://cs420.hao-li.com>

Thanks!

