CSCI 4530/6530 Advanced Computer Graphics

https://www.cs.rpi.edu/~cutler/classes/advancedgraphics/S25/

Lecture 8: Rigid Body Dynamics, Collision Response, & Deformation

Worksheet: Mass-Spring Cloth Simulation

Sketch the first few frames of a 2D explicit Euler mass-spring simulation for a 2

netwo

masse

only s

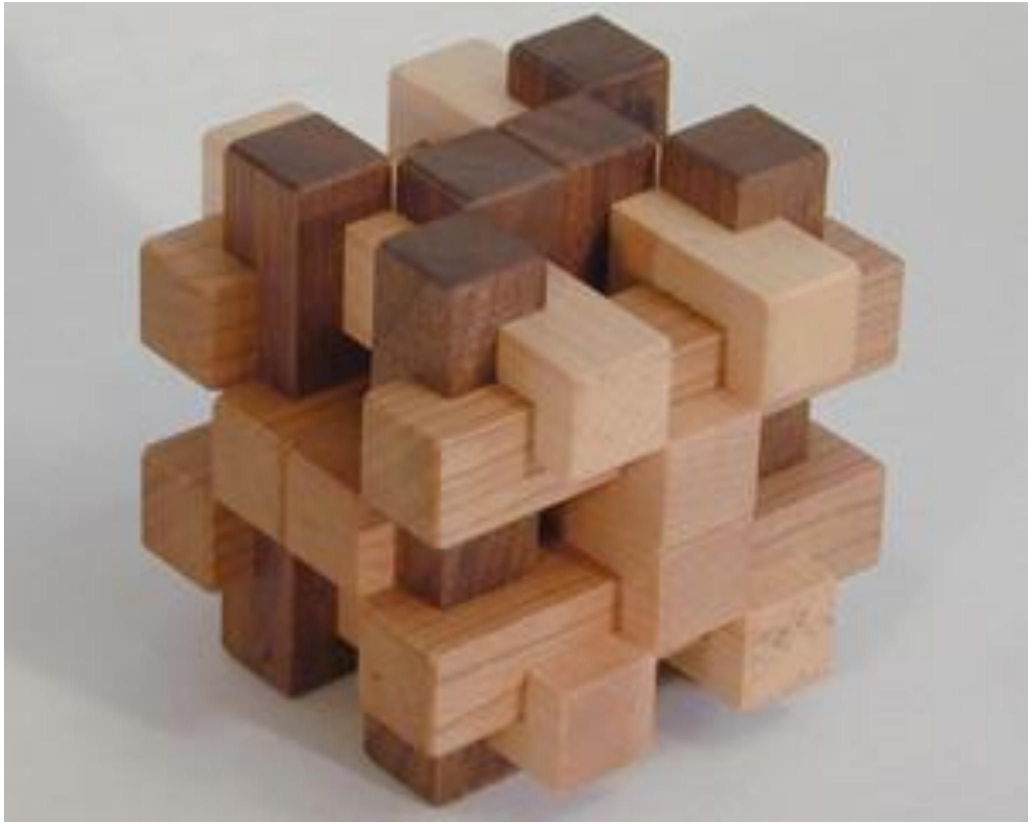
spring

unifor

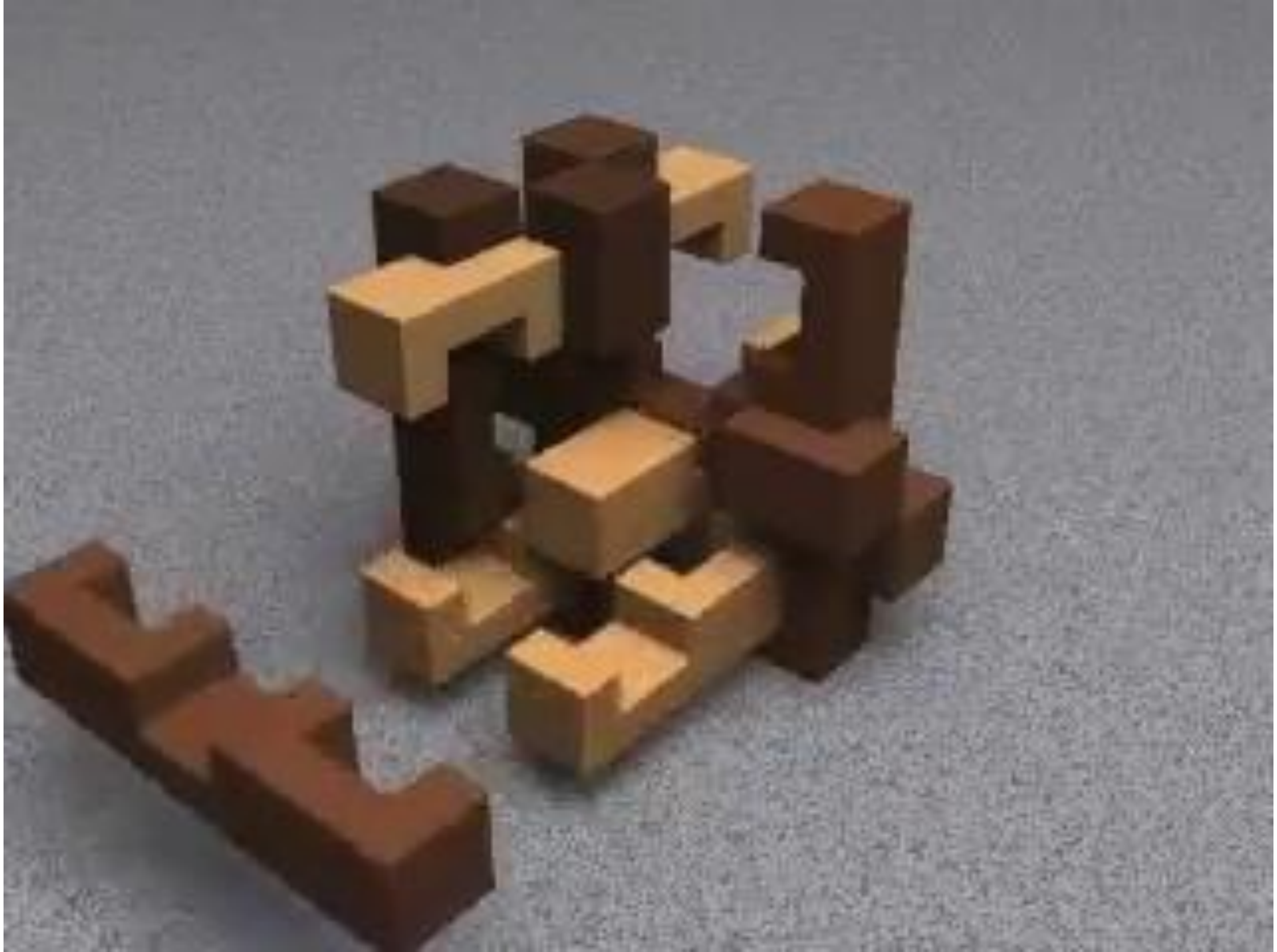
NOTE: We'll be doing pair worksheets throughout the term. Bonus points if you work with a different partner for every worksheet!

t = 0 t = 1 t = 2

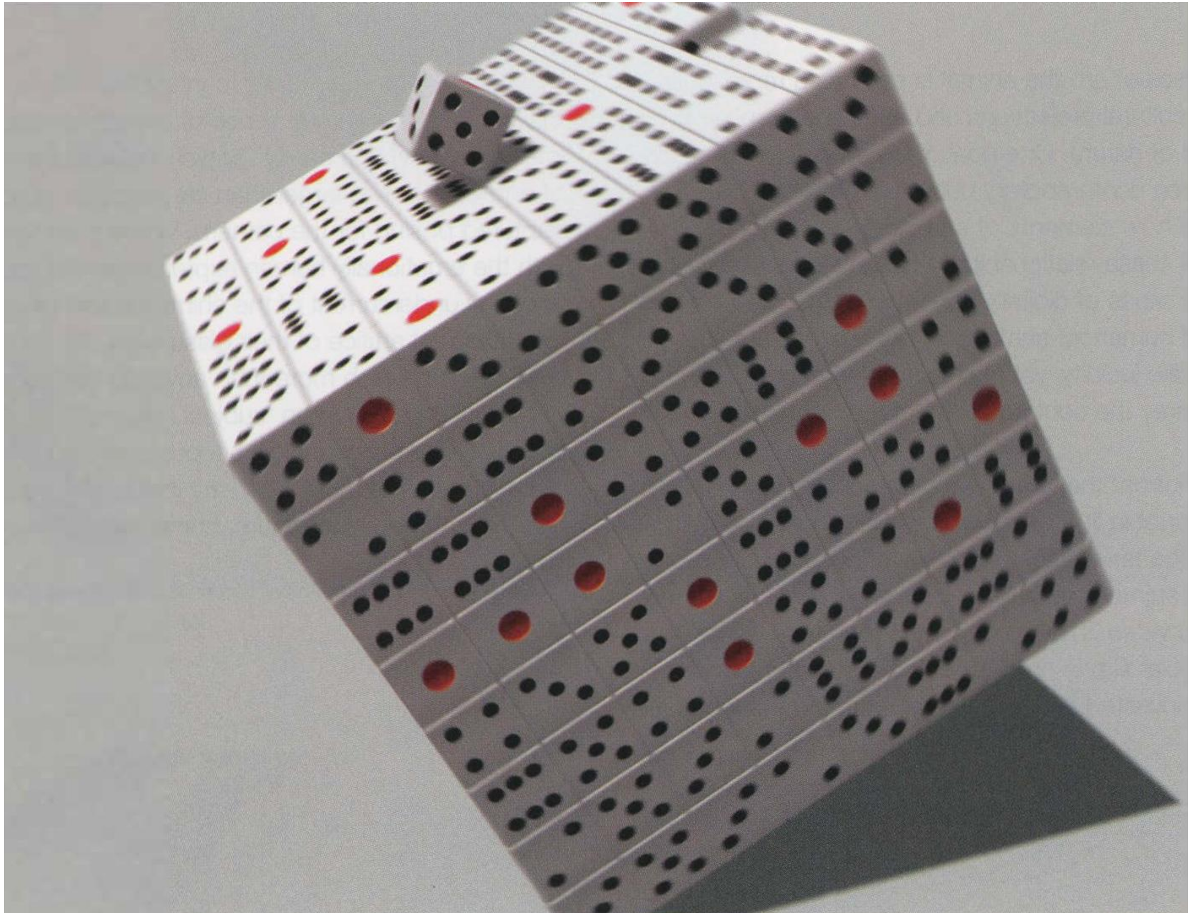
Burr, Justin Legakis ~1999

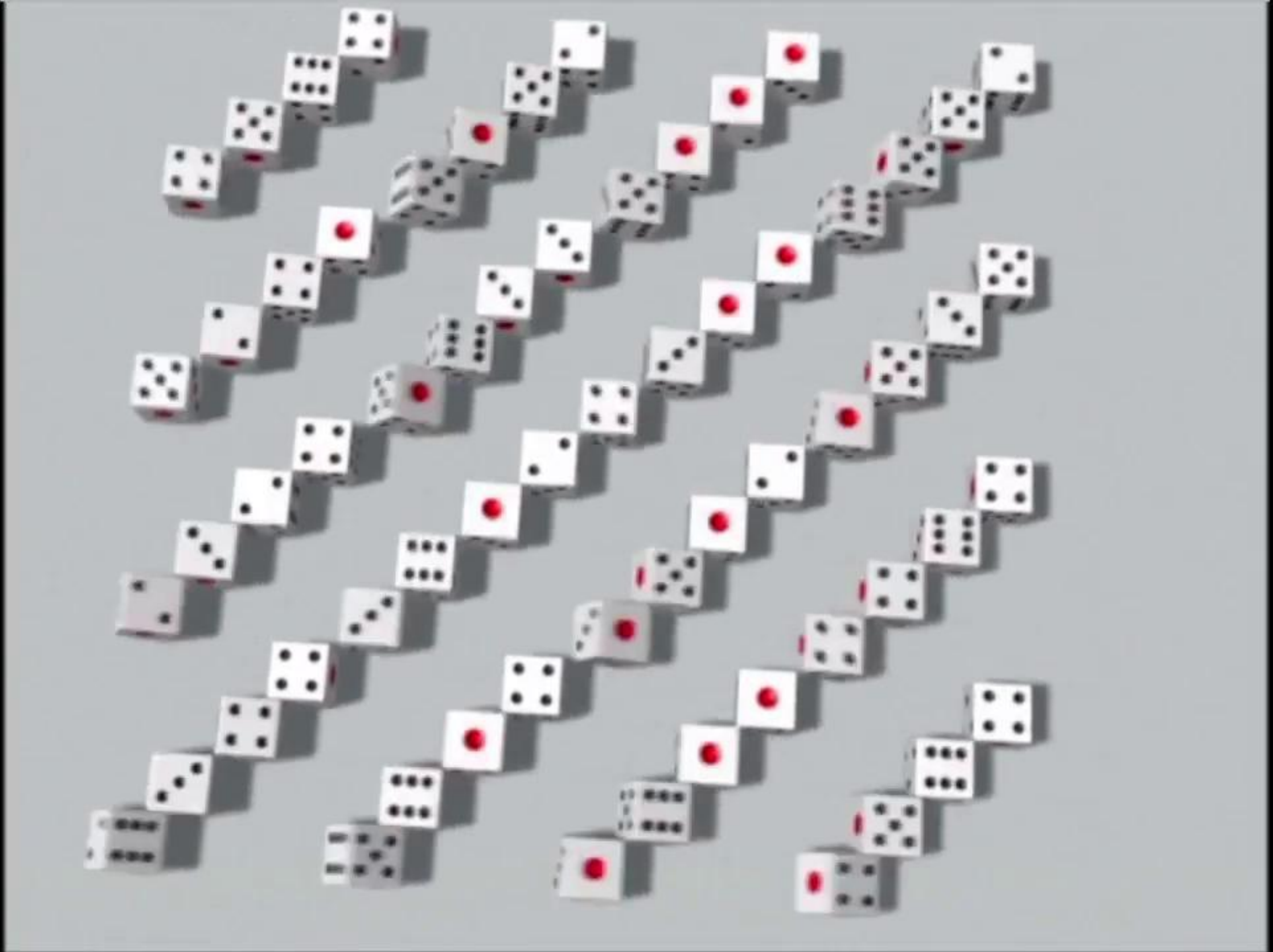


18 Piece Burr, Bill Cutler Puzzles



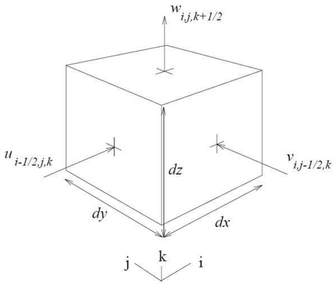
Dice, Hitoshi Akayama, SIGGRAPH 2005





Last Time?

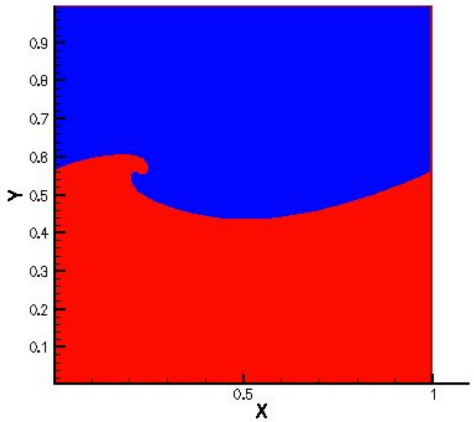
- Navier-Stokes Equations
- Conservation of Momentum & Mass
- Incompressible Flow

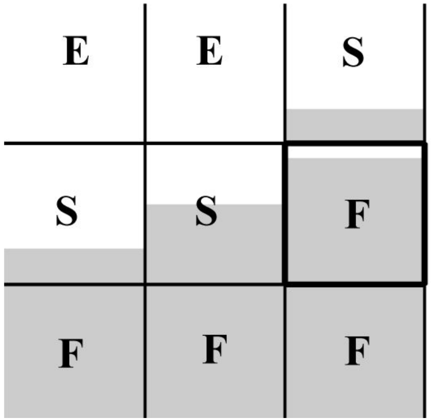


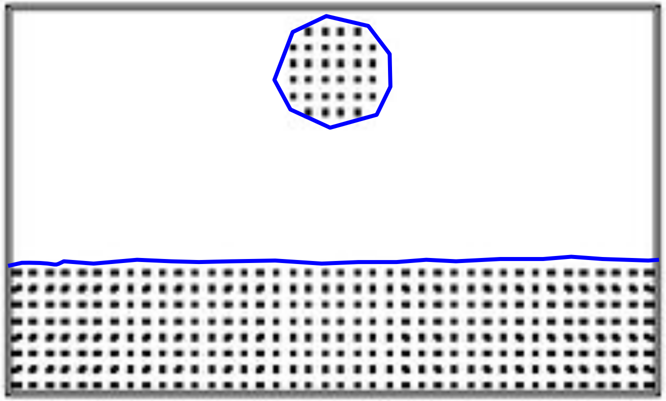
$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z} = -\frac{\partial p}{\partial x} + g_x + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)$$

$$\frac{\partial v}{\partial t} + \frac{\partial vu}{\partial x} + \frac{\partial v^2}{\partial y} + \frac{\partial vw}{\partial z} = -\frac{\partial p}{\partial y} + g_y + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right)$$

$$\frac{\partial w}{\partial t} + \frac{\partial wu}{\partial x} + \frac{\partial wv}{\partial y} + \frac{\partial w^2}{\partial z} = -\frac{\partial p}{\partial z} + g_z + \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right)$$

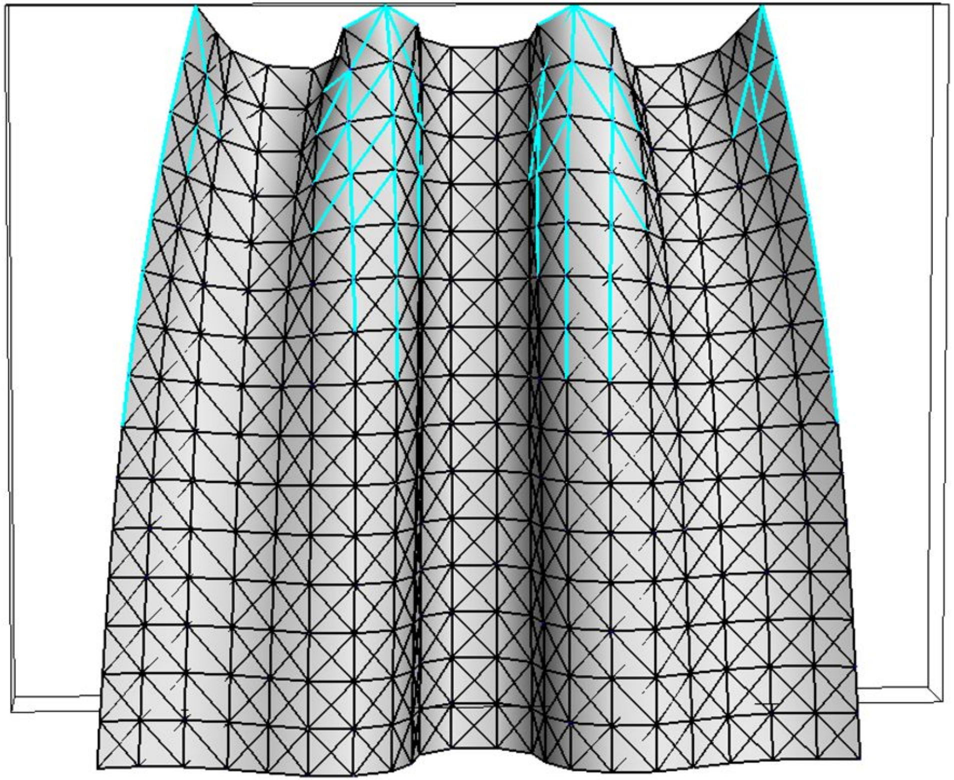


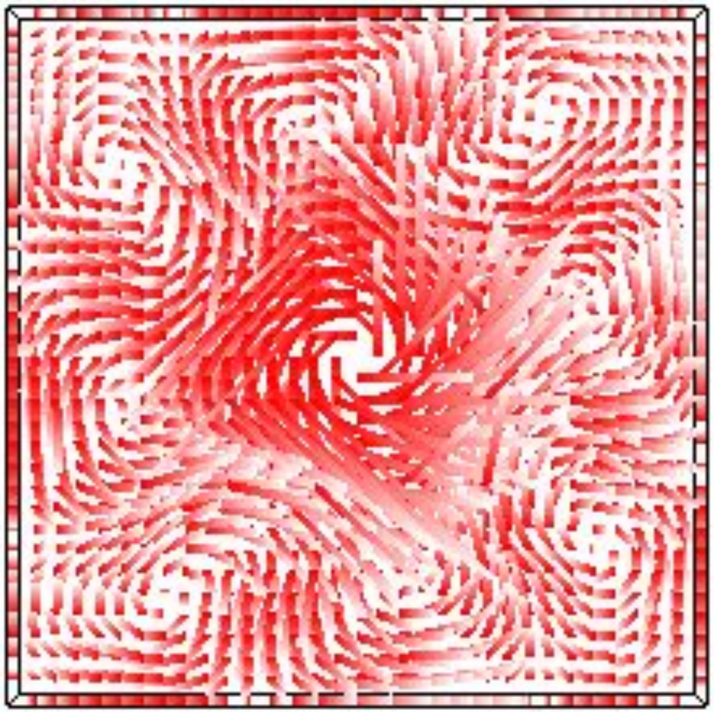




HW2: Cloth & Fluid Simulation

HW2 progress post deadline extended due Saturday 11:59pm





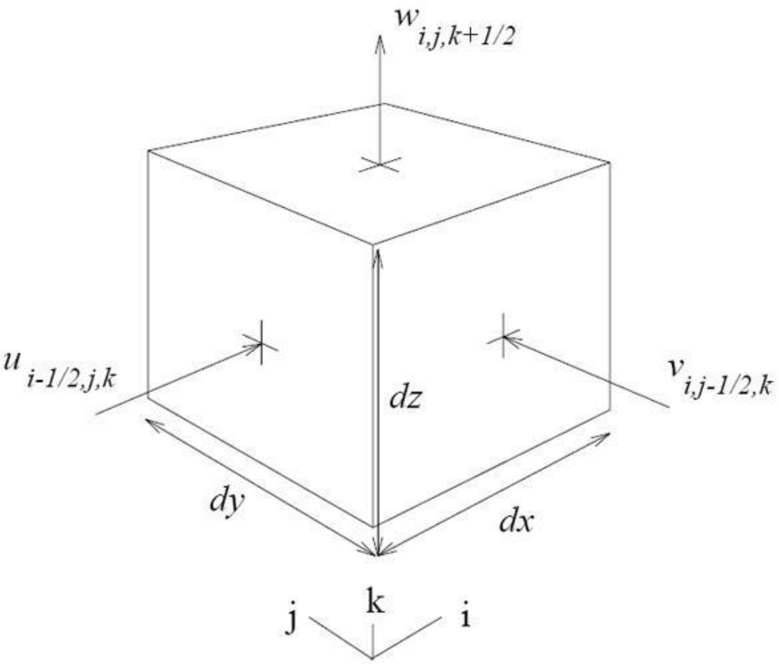
Today

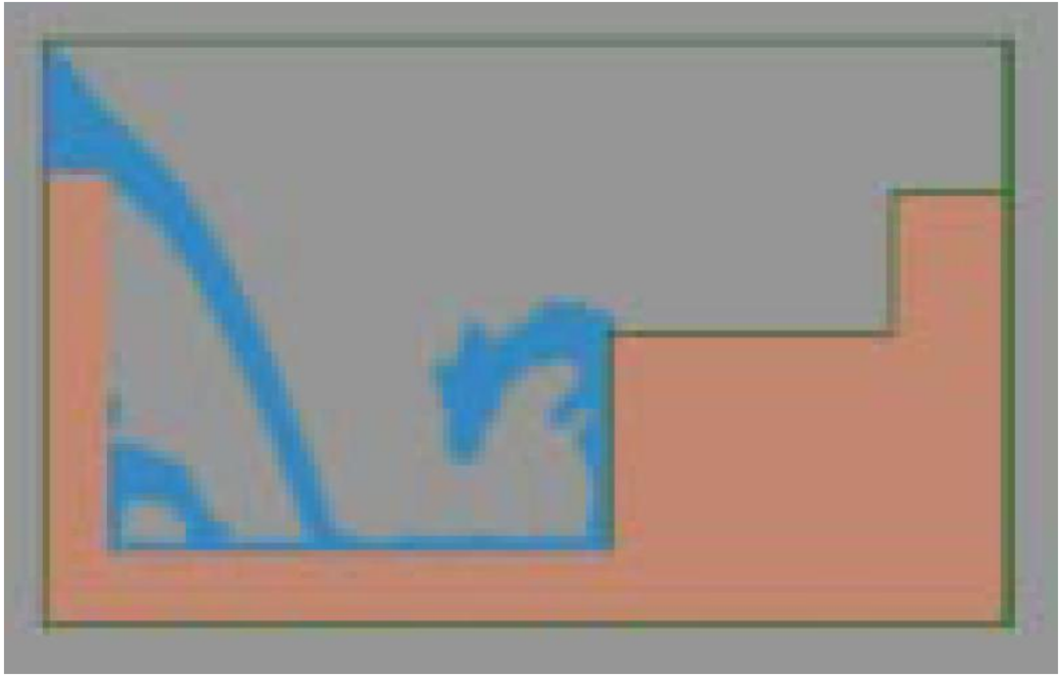
- Worksheet: Mass-Spring Cloth Simulation
- Readings for Today
- From Last Time: Data Structure & Algorithm for Fluid Simulation
- Rigid Body Dynamics
- Collision Response
- Non-Rigid, Deformable Objects
- Finite Element Method
- Papers for Tuesday

Reading for Next Time

Everyone should read this
(simple fluid model used in HW2)

- "Realistic Animation of Liquids", Foster & Metaxas, 1996

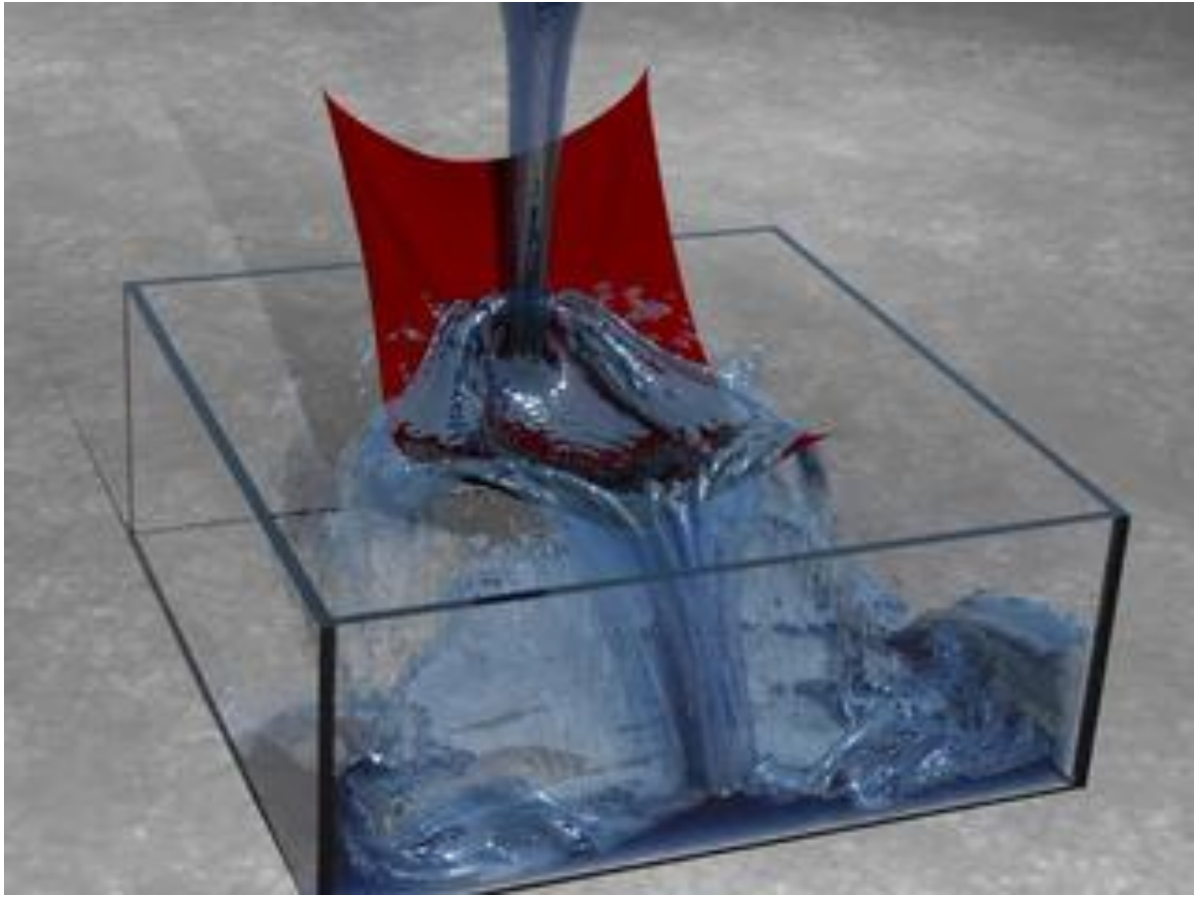




- “Standard fluid dynamics”
- Balances accuracy, efficiency, artistic control
- Low resolution navier stokes + higher resolution (height field?) surface
- Leverage symmetry when possible
- Sources & sinks, external forces, pressure adjustments, viscous fluids, buoyancy (floating objects)
- Boundary conditions: must be aligned with/approximated by grid
  - What artifacts will this introduce for smooth curves?
- Writing: Details not fully explained -or- mathematics explained fully -or- best paper we’re read so far?
- Includes implementation outline/pseudocode – helpful to reproduce results.
- Hard to understand/evaluate the quality/artifacts/limitations of this method without access to the video

Optional Reading for Next Time

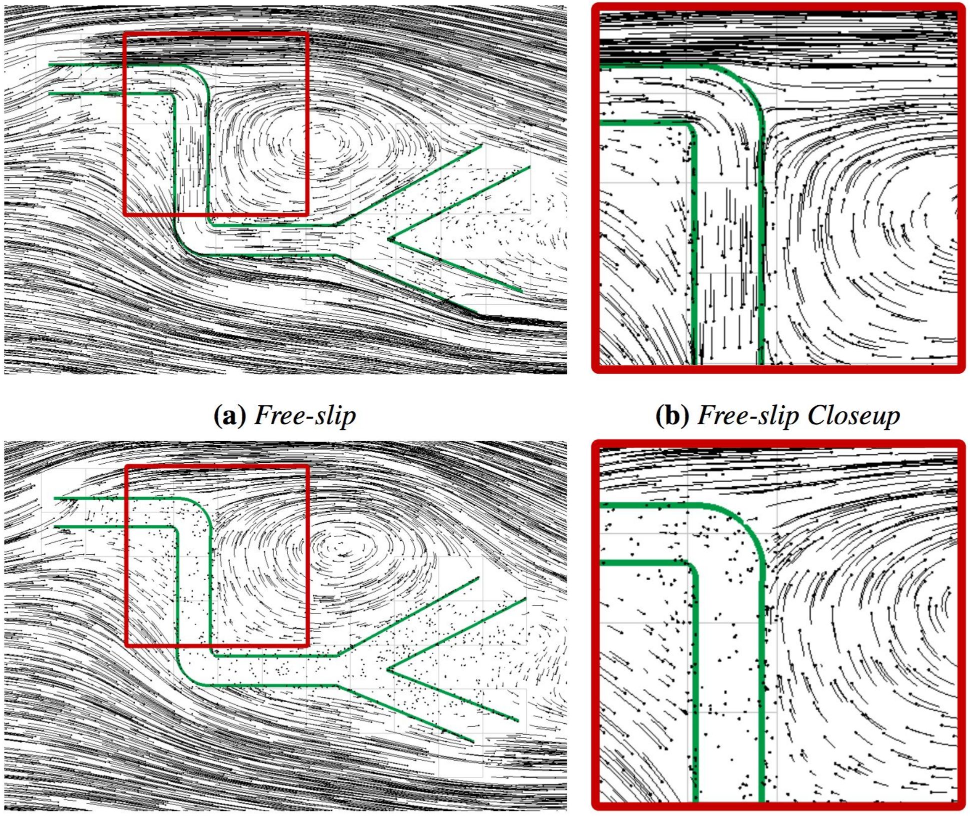
- "Coupling Water and Smoke to Thin Deformable and Rigid Shells", Guendelman, Selle, Losasso, & Fedkiw, SIGGRAPH 2005.



- Coupled cloth + fluid simulation
- Goal: eliminate leaking of fluid through the cloth (e.g. simulate a waterproof plastic sheet)
  - Detecting and preventing collisions with thin surfaces is very challenging - time step must be proportionally small to notice/‘catch’ the collision.
- But for many/most real-world cloth textiles, water will leak through

Optional Reading for Next Time

- "Preserving Geometry and Topology for Fluid Flows with Thin Obstacles and Narrow Gaps" Azevedo, Batty, & Oliveira, SIGGRAPH 2016



(a) *Free-slip* **(b)** *Free-slip Closeup*

(c) *No-slip* **(d)** *No-slip Closeup*

- Excellent diagrams
- Intuitive description of technique
- Expands upon previous concepts/work (doesn’t come up with an entirely new method)

How to read a research paper?

-

How to read a research paper?

(especially an advanced paper in a new area)

- Multiple readings are often necessary
- Don't necessarily read from front to back
- Lookup important terms
- Target application & claimed contributions
- Experimental procedure
- How well results & examples support the claims
- Scalability of the technique (Big O Notation)
- Limitations of technique, places for future research
- Possibilities for hybrid systems with other work

Components of a well-written research paper?

-

Components of a well-written research paper?

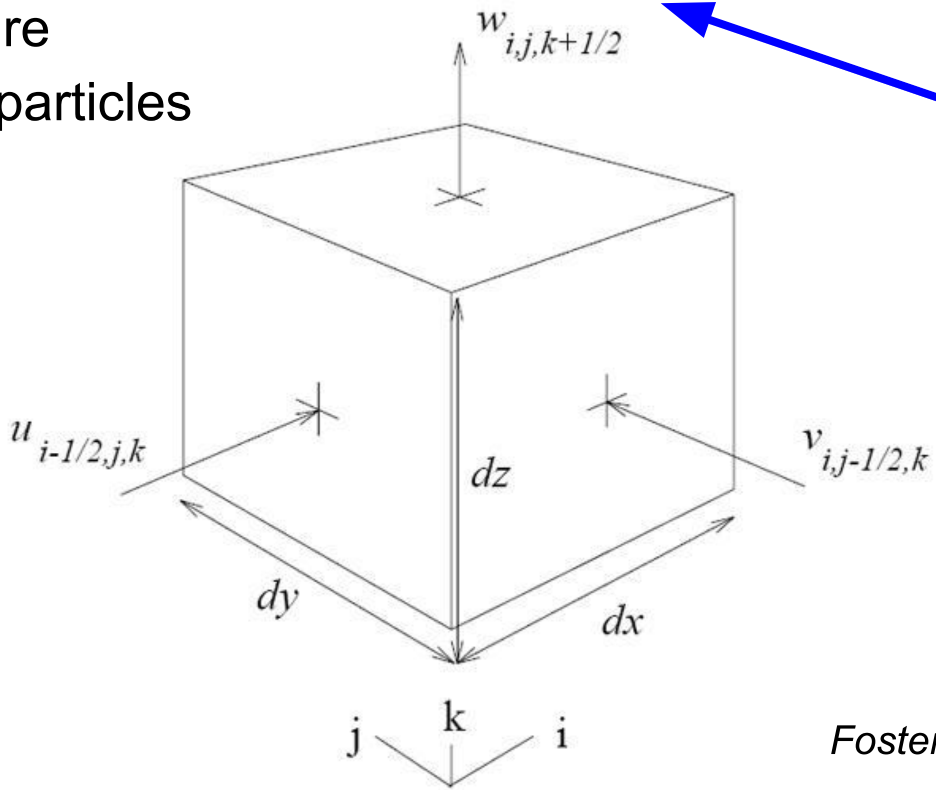
- Motivation/context/related work
- Contributions of this work
- Clear description of algorithm
  - Sufficiently-detailed to allow work to be reproduced
  - Work is theoretically sound
    (hacks/arbitrary constants discouraged)
- Results
  - well chosen examples
  - clear tables/illustrations/visualizations
- Conclusions
  - limitations of the method are clearly stated

Today

- Worksheet: Mass-Spring Cloth Simulation
- Readings for Today
- From Last Time: Data Structure & Algorithm for Fluid Simulation
- Rigid Body Dynamics
- Collision Response
- Non-Rigid, Deformable Objects
- Finite Element Method
- Papers for Tuesday

Each Grid Cell Stores:

- Velocity *at the cell faces* (offset grid)
- Pressure
- List of particles



This is a critically important detail! (and makes correct implementation rather annoying)

Image from Foster & Metaxas, 1996

Fluid Simulation Implementation

- Initialization:
  - Choose a voxel resolution
  - Choose a particle density
  - Create grid & place the particles
  - Initialize pressure & velocity of each cell
  - Set the viscosity & gravity
- Choose a timestep & go!

This piece needs more explanation!

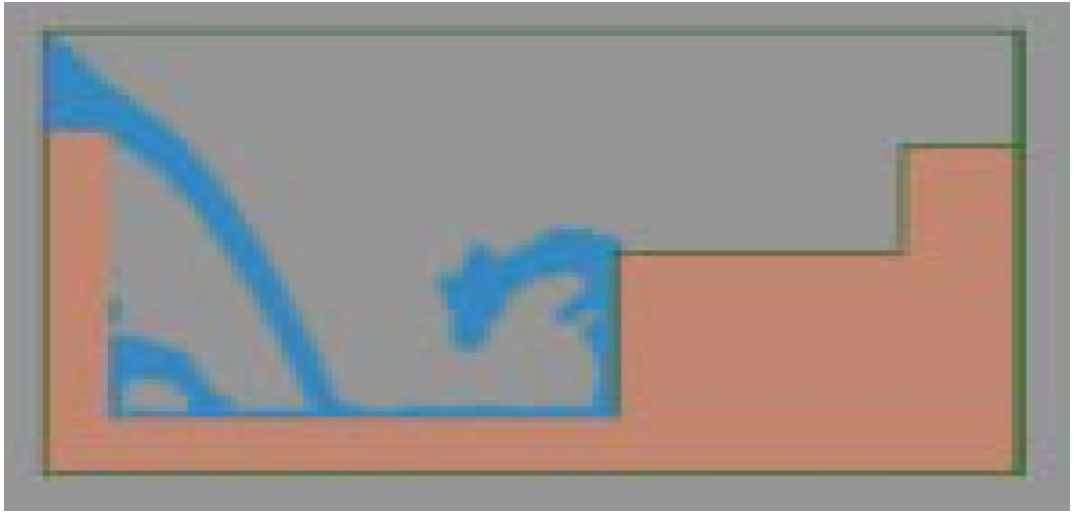
At each Timestep:

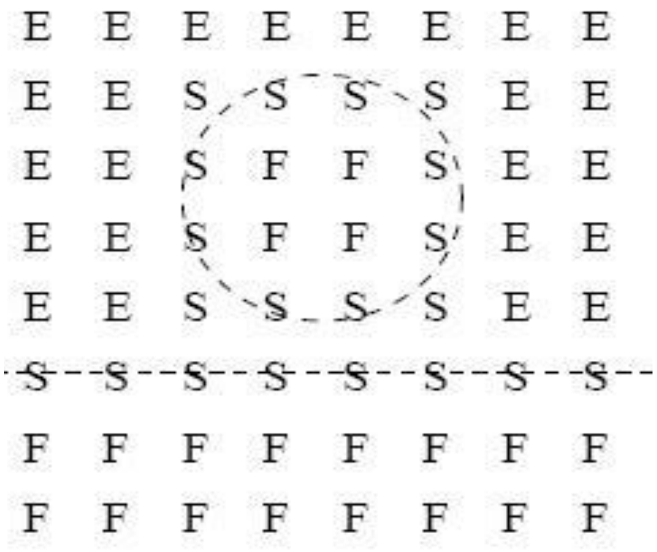
- Identify which cells are Empty, Full, or on the Surface
- Compute new velocities
- Adjust the velocities to maintain an incompressible flow
- Move the particles
    - Interpolate the velocities at the faces
- Render the geometry and repeat!

Empty, Surface, & Full Cells

This step is necessary for 2-phase simulations (e.g. air+water), where we enforce incompressibility for only one phase!

- Empty: no marker particles
- Surface: has an neighbor that is “Empty”
- Full: not “Empty” or “Surface”





Images from Foster & Metaxas, 1996

At each Timestep:

- Identify which cells are Empty, Full, or on the Surface
- Compute new velocities
- Adjust the velocities to maintain an incompressible flow
- Move the particles
    - Interpolate the velocities at the faces
- Render the geometry and repeat!

Compute New Velocities

$$\tilde{u}_{i+1/2,j,k} = u_{i+1/2,j,k} + \delta t\{(1/\delta x)[(u_{i,j,k})^2 - (u_{i+1,j,k})^2]$$
$$+(1/\delta y)[(uv)_{i+1/2,j-1/2,k} - (uv)_{i+1/2,j+1/2,k}]$$
$$+(1/\delta z)[(uw)_{i+1/2,j,k-1/2} - (uw)_{i+1/2,j,k+1/2}] + g_x$$
$$+(1/\delta x)(p_{i,j,k} - p_{i+1,j,k}) + (\nu/\delta x^2)(u_{i+3/2,j,k}$$
$$-2u_{i+1/2,j,k} + u_{i-1/2,j,k}) + (\nu/\delta y^2)(u_{i+1/2,j+1,k}$$
$$-2u_{i+1/2,j,k} + u_{i+1/2,j-1,k}) + (\nu/\delta z^2)(u_{i+1/2,j,k+1}$$
$$-2u_{i+1/2,j,k} + u_{i+1/2,j,k-1})\},$$

Note: some of these values are the average velocity within the cell rather than the velocity at a cell face

At each Timestep:

- Identify which cells are Empty, Full, or on the Surface
- Compute new velocities
- Adjust the velocities to maintain an incompressible flow
- Move the particles
    - Interpolate the velocities at the faces
- Render the geometry and repeat!

Adjusting the Velocities

- Calculate the *divergence* of the cell (the extra in/out flow)
- The divergence is used to update the *pressure* within the cell
- Adjust each face velocity uniformly to bring the divergence to zero
- Iterate across the entire grid until divergence is < ε

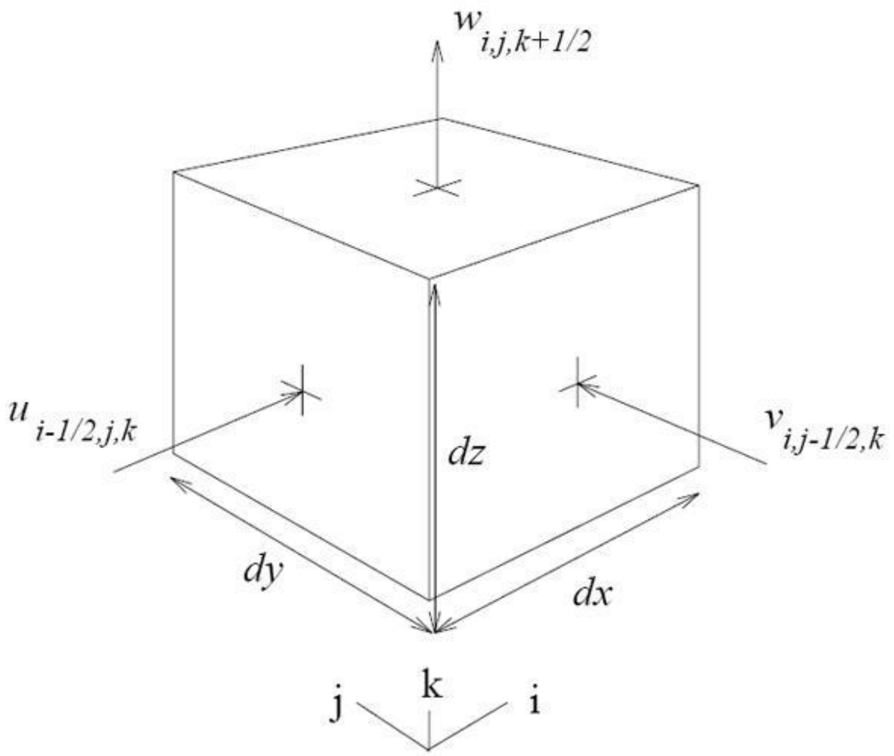
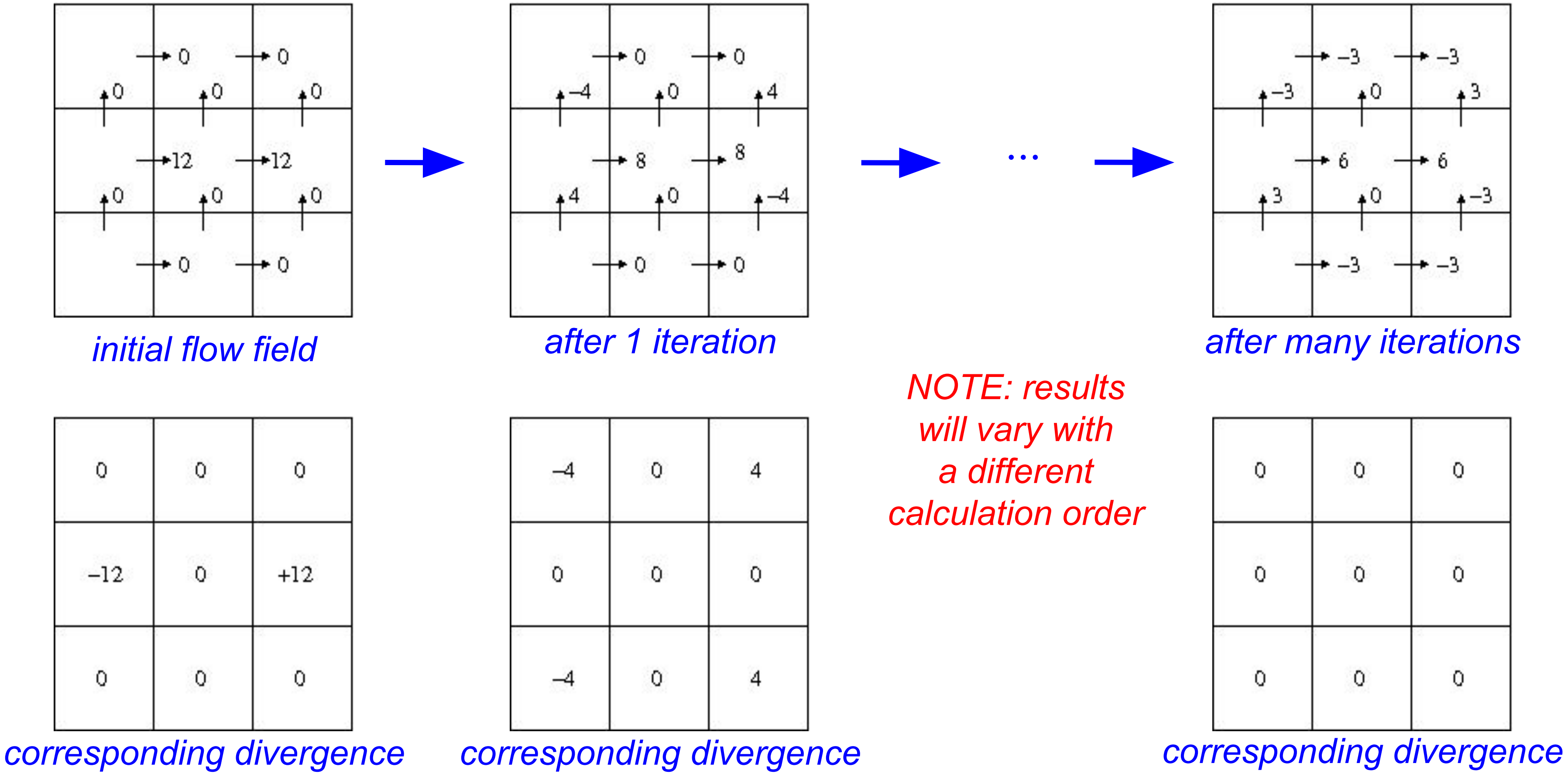


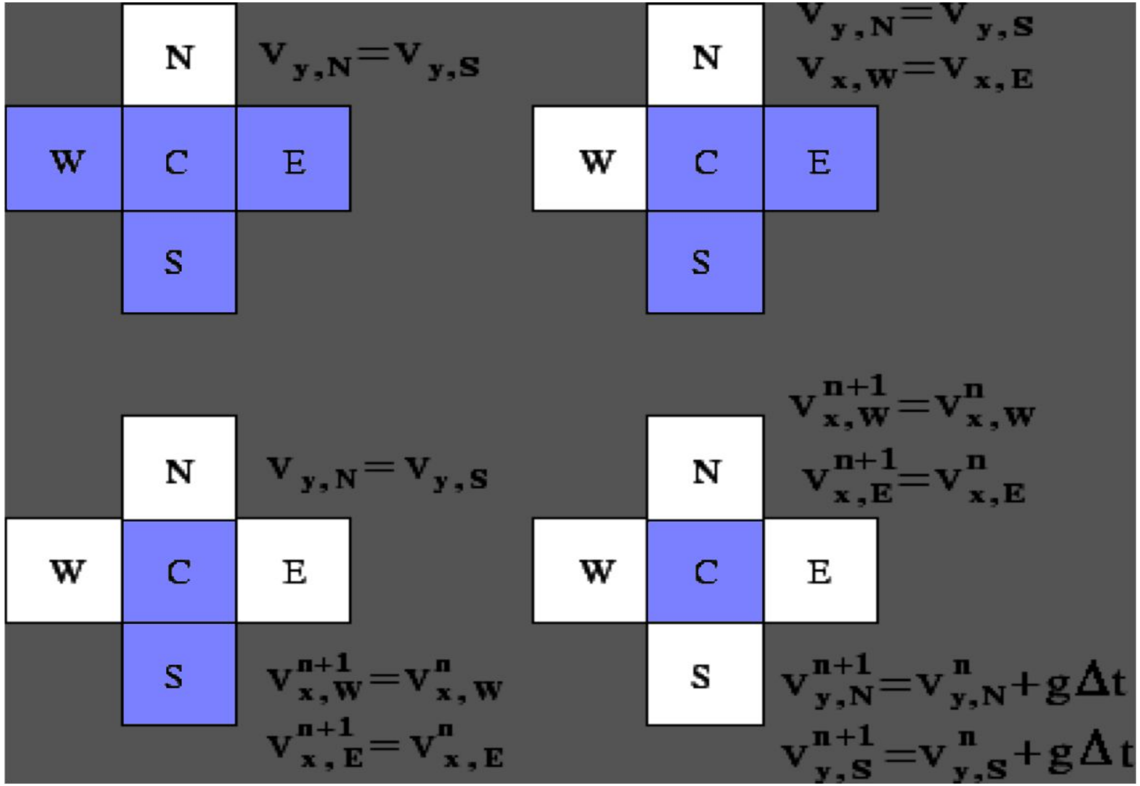
Image from Foster & Metaxas, 1996

Calculating/Eliminating Divergence



Handing Divergence with a Free Surface with MAC

- Divergence in **surface cells**:
  - Divide excess equally amongst neighboring **empty cells**
  - Handle the special cases (include gravity)
- Zero out the divergence & pressure in empty cells



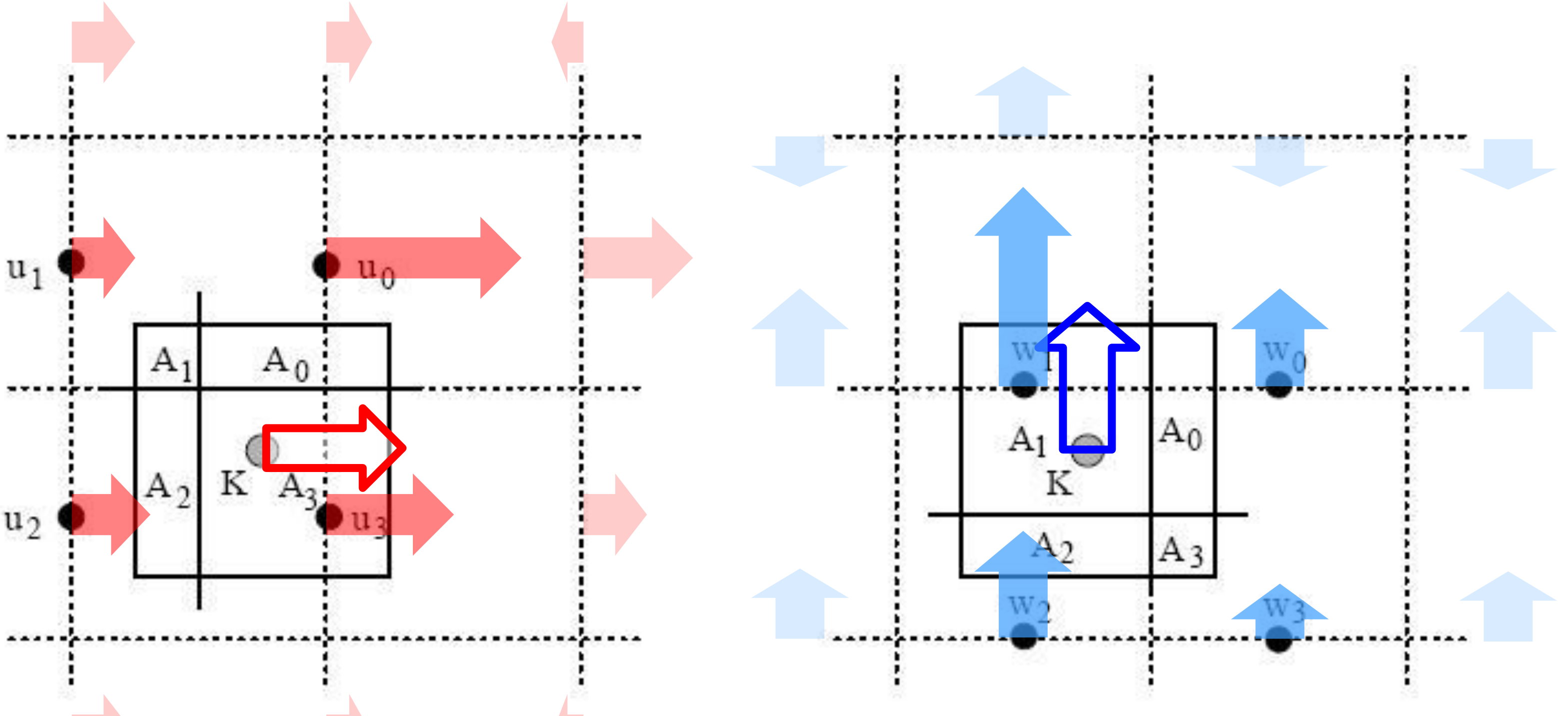
At each Timestep:

- Identify which cells are Empty, Full, or on the Surface
- Compute new velocities
- Adjust the velocities to maintain an incompressible flow
- Move the particles
    - Interpolate the velocities at the faces
- Render the geometry and repeat!

Velocity Interpolation

Original image from
Foster & Metaxas, 1996

- In 2D: For each dimension, find the 4 closest ***face velocity*** samples

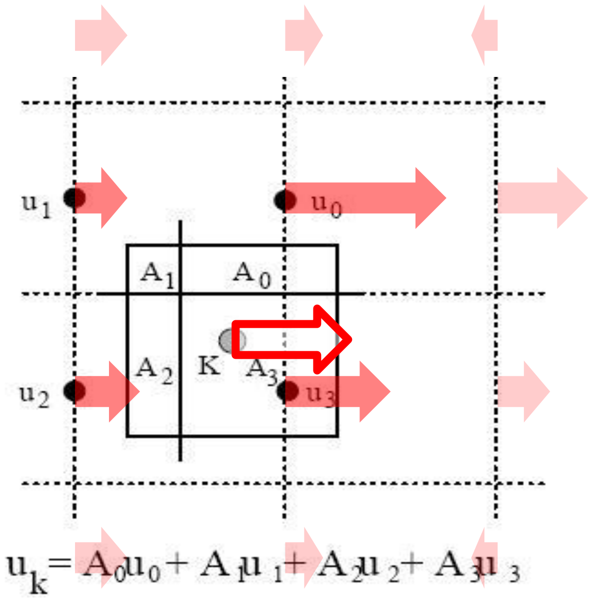


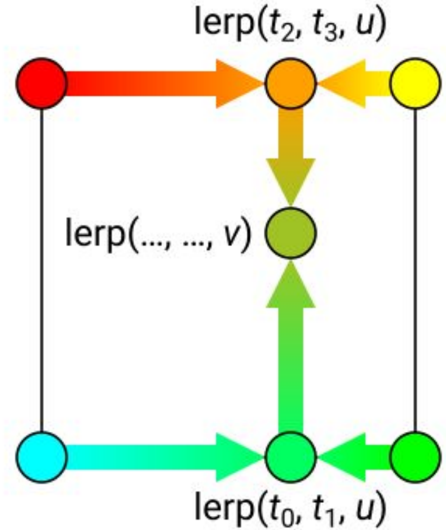
$u_k = A_0 u_0 + A_1 u_1 + A_2 u_2 + A_3 u_3$

$w_k = A_0 w_0 + A_1 w_1 + A_2 w_2 + A_3 u_3$

Velocity Interpolation = *Bilinear Interpolation*

- In 2D: For each dimension, find the 4 closest ***face velocity*** samples
- *In 3D: Find 8 closest face velocities in each dimension*
- Separately interpolate velocity for each axis, then combine





http://reedbeta.com/blog/quadrilateral-interpolation-part-2/

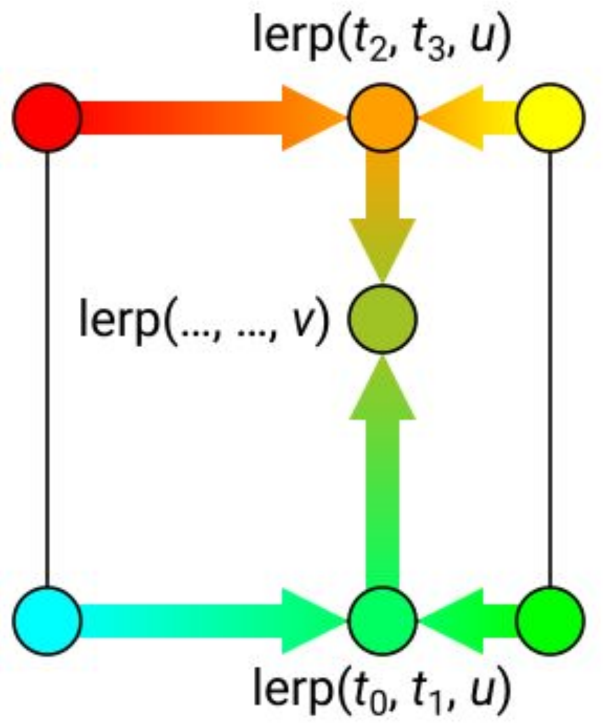
Bilinear Interpolation

- It might be simplest to think about interpolating one axis at a time
  It doesn't matter which axis you start with!
- Calculate *u*, the fraction of the distance along the horizontal axis, ***e.g., u=0.65***
- Then calculate the top & bottom averages:

  *orange = (1-u)*red + u*yellow*

  *bluegreen = (1-u)*cyan + u*green*
- Calculate *v*, the fraction of the distance along the vertical axis, ***e.g., v=0.6***
- Then calculate the final average:

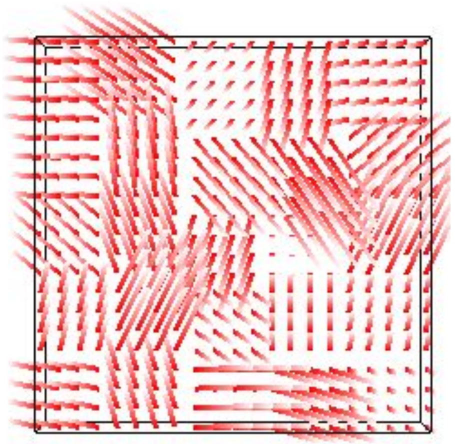
  *pukegreen = (1-v)*bluegreen + v*orange*



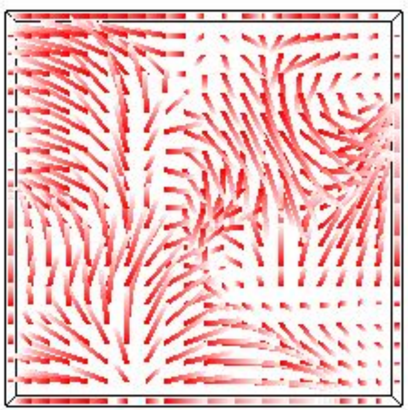
http://reedbeta.com/blog/quadrilateral-interpolation-part-2/

Correct Velocity Interpolation

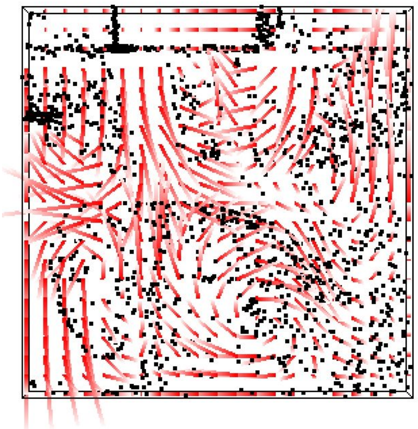
- WARNING: The finished code for 3D face velocity interpolation isn't particularly elegant… Storing velocities at face midpoints (req'd for conservation of mass) makes the index math messy!



No Interpolation (just use the left/bottom face velocity)
Note the discontinuities in velocity at cell boundaries



Correct Interpolation
Note that the velocity perpendicular to the outer box is zero



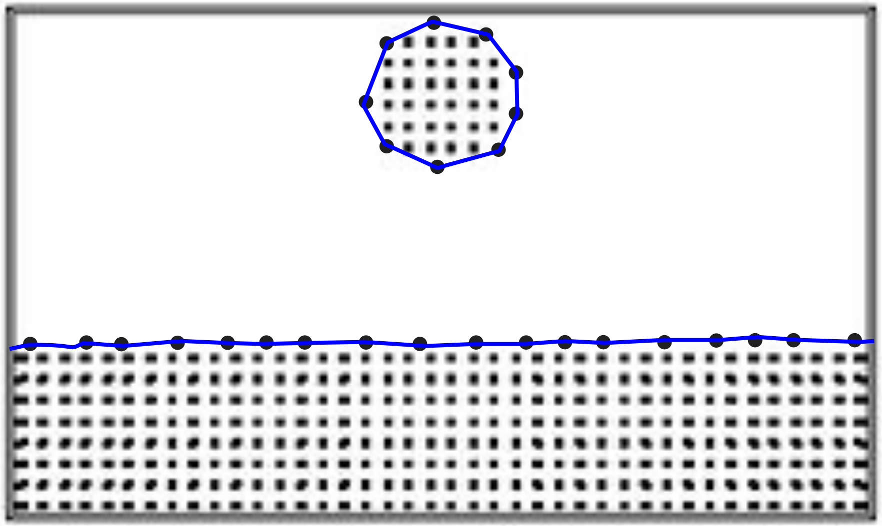
Buggy Interpolation
Note the clumping particles, and the discontinuities at some of the cell borders (& particles might escape the box!)

At each Timestep:

- Identify which cells are Empty, Full, or on the Surface
- Compute new velocities
- Adjust the velocities to maintain an incompressible flow
- Move the particles
    - Interpolate the velocities at the faces
- Render the geometry and repeat!

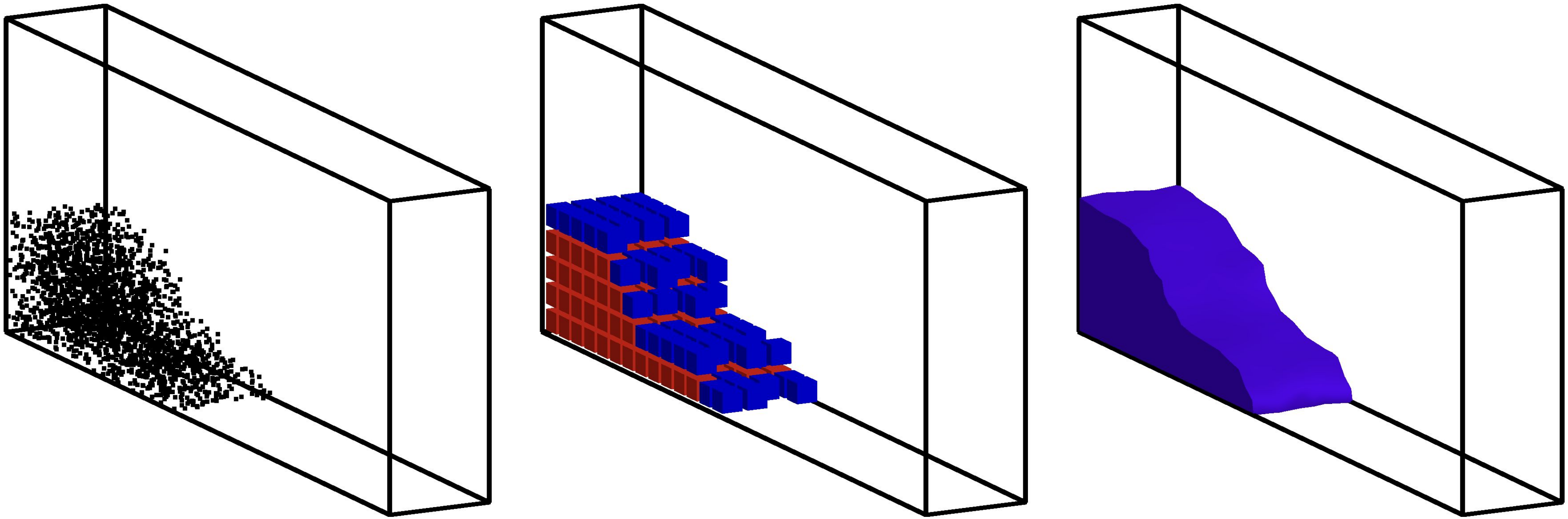
Rendering via Surface Marker Particles

- Volumetric marker particles (black) track the volume the fluid/water (which cells contain fluid)
- Additionally, surface marker particles (blue) track the boundary between water & air
- Nearby surface particles are connected with edges & triangles
- *Where the surface geometry changes significantly, surface marker particles must be added and/or removed. The surface is re-triangulated. The surface topology may change as regions of fluid merge or separate.*



Fluid Surface Rendering using Marching Cubes

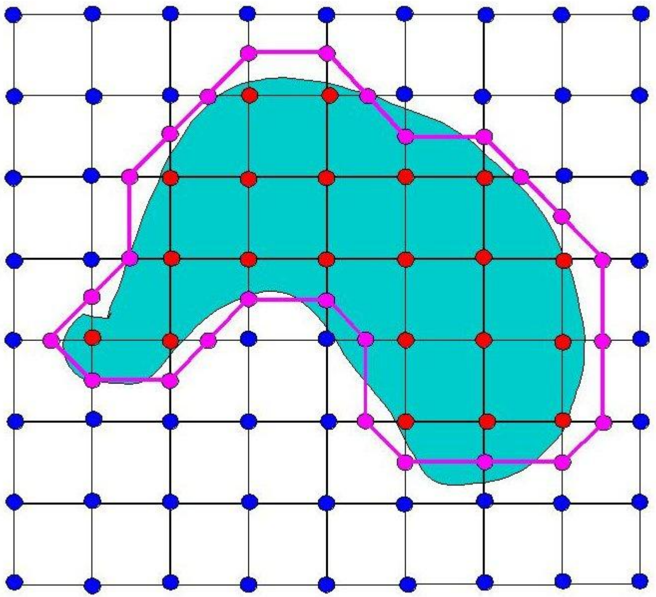
- Provided HW2 code approximates the surface using Marching Cubes (this is not the best/typical/recommended method)



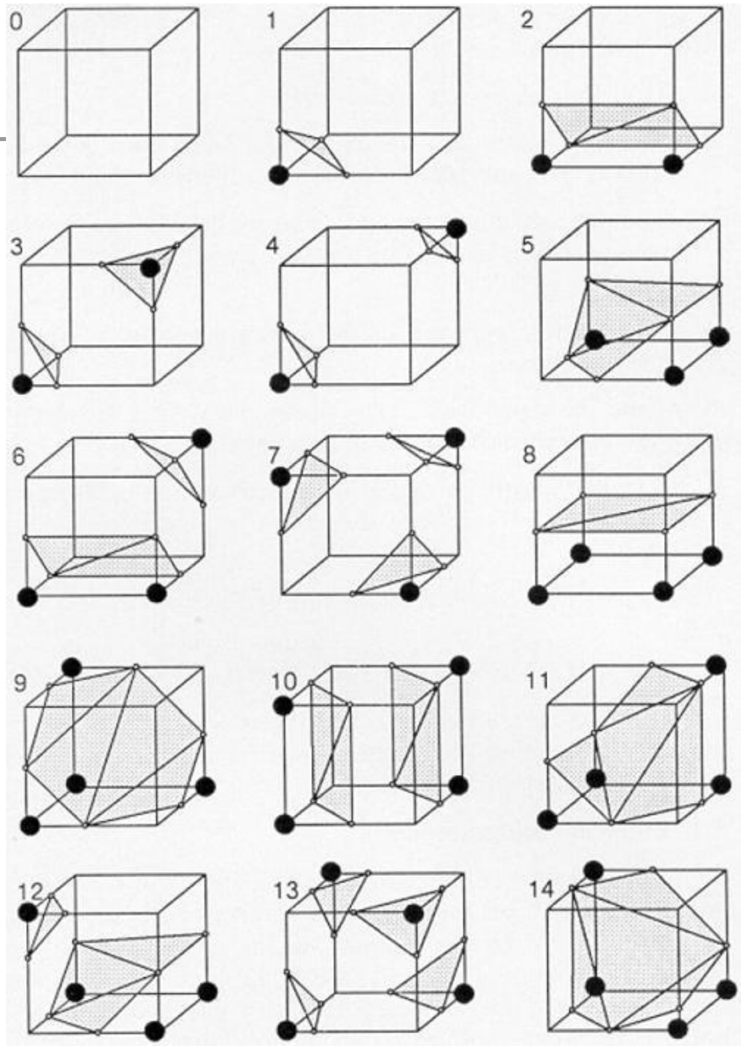
Marching Cubes

"Marching Cubes: A High Resolution 3D Surface Construction Algorithm", Lorensen and Cline, SIGGRAPH '87.

- Classic technique for extracting surface from scalar voxel data



http://www.cs.carleton.edu/cs_comps/0405/shape/marching_cubes.html

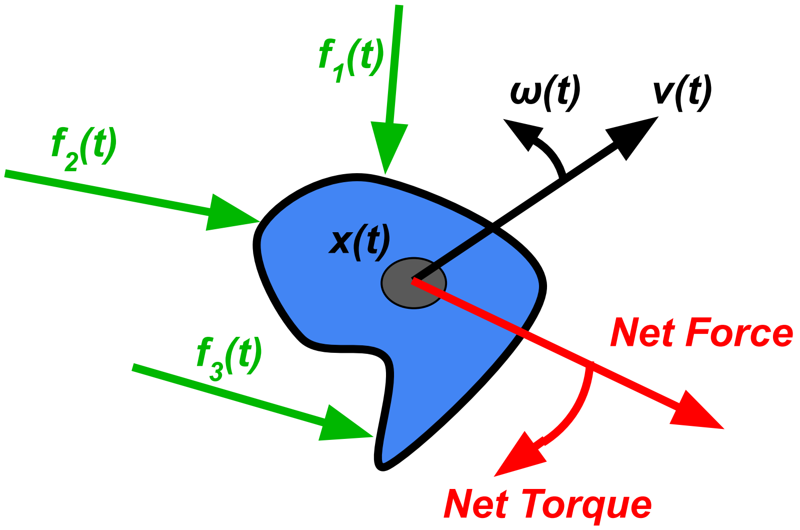


Today

- Worksheet: Mass-Spring Cloth Simulation
- Readings for Today
- From Last Time: Data Structure & Algorithm for Fluid Simulation
- Rigid Body Dynamics
- Collision Response
- Non-Rigid, Deformable Objects
- Finite Element Method
- Papers for Tuesday

Rigid Body Dynamics

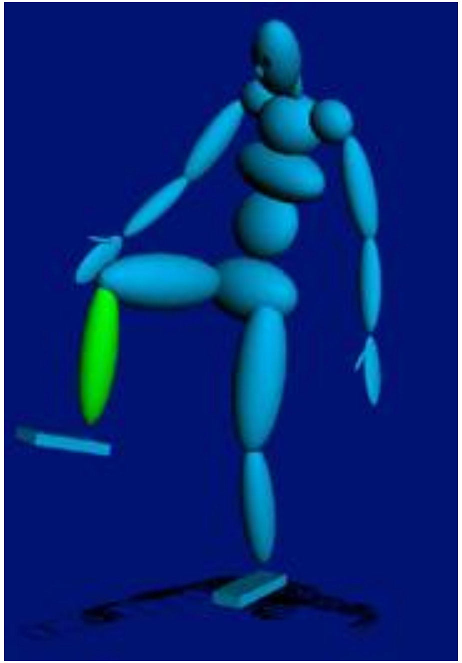
- How do we simulate this object's motion over time?
- We could discretize the object into many particles…
  - But a rigid body does *not* deform
  - Only a few *degrees of freedom*
- Instead, we use only one particle at the center of mass
  - Body has velocity ***v(t)*** *and* angular velocity ***ω(t)***
  - Compute net force & net torque



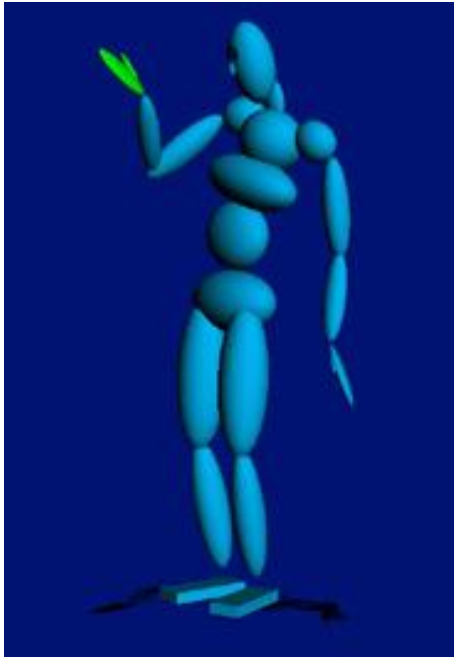
Nice Reference Material: http://www.pixar.com/companyinfo/research/pbm2001/

Degree of Freedom (DOF)

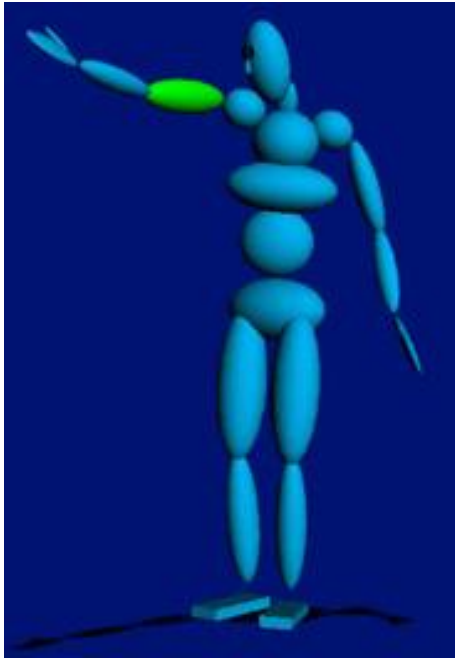
- Rotations:



1 DOF: knee



2 DOF: wrist

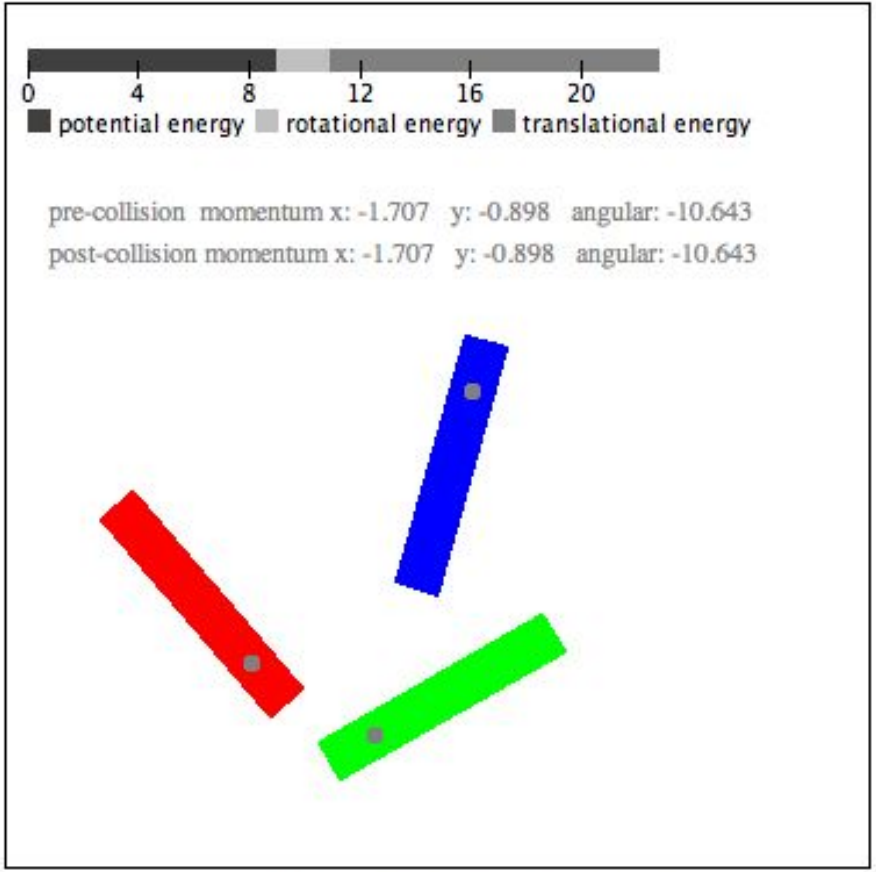


3 DOF: arm

- Translations count too… ➔ *6 Degrees of Freedom (DOF)*

Energy & Rigid Body Collisions

- Total Energy =
  Kinetic Energy +
  Potential Energy +
  Rotational Energy
- Total Energy stays constant if there is no damping and no friction
- Rotational Energy is constant between collisions



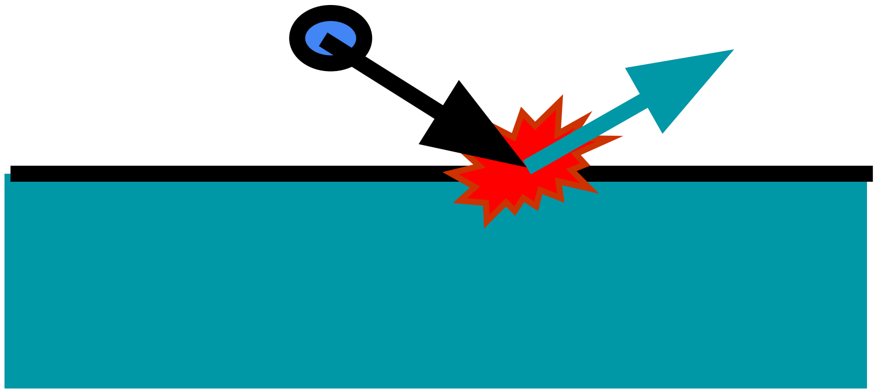
http://www.myphysicslab.com/collision.html

Today

- Worksheet: Mass-Spring Cloth Simulation
- Readings for Today
- From Last Time: Data Structure & Algorithm for Fluid Simulation
- Rigid Body Dynamics
- Collision Response
- Non-Rigid, Deformable Objects
- Finite Element Method
- Papers for Tuesday

Collisions

- Detection
- Response
- Overshooting problem
  (when we enter the solid)

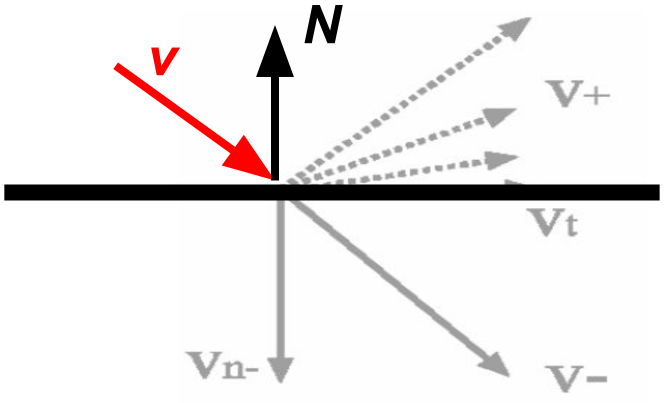


Collision Response

- tangential velocity $\boldsymbol{v_t}$ unchanged
- normal velocity $\boldsymbol{v_n}$ reflects:

$$v = v_t + v_n$$

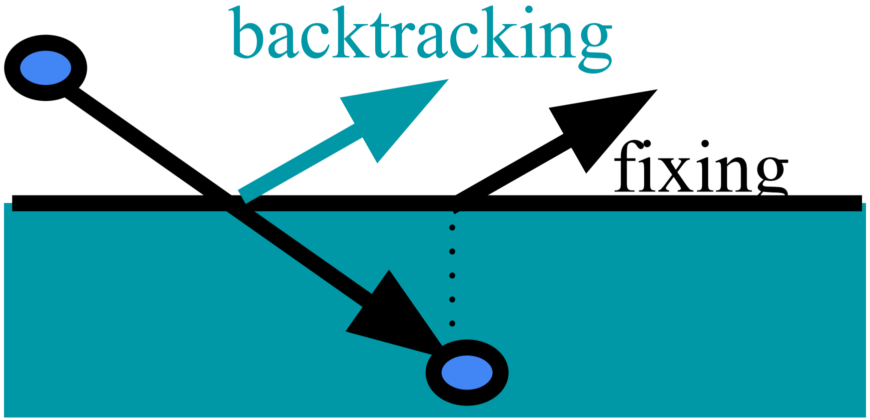
$$v \leftarrow v_t - \varepsilon v_n$$



- coefficient of restitution
  - 1 for elastic
  - 0 for plastic
- change of velocity = $-(1+\epsilon)v$
- change of momentum *Impulse* = $-m(1+\epsilon)v$

Collisions - Overshooting

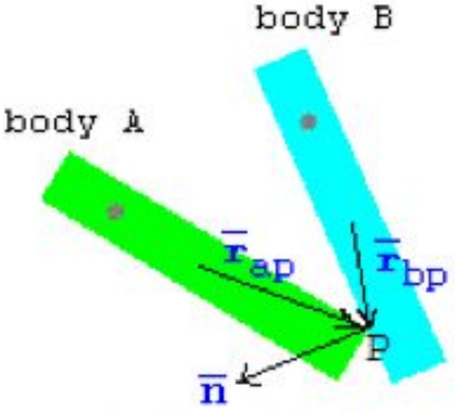
- Usually, we detect collision when it's too late: we're already inside
- Solutions: back up
  - Compute intersection point
  - Compute response there
  - Advance for remaining fractional time step
- Other solution: Quick and dirty fixup
  - Just project back to object closest point



Collision Between Two Objects

- Suppose a vertex on body A is colliding into an edge of body B at point P. Define the following variables:

m_a, m_b = mass of bodies A, B
$\bar{\mathbf{r}}_{ap}$ = distance vector from center of mass of body A to point P
$\bar{\mathbf{r}}_{bp}$ = distance vector from center of mass of body B to point P
ω_{a1}, ω_{b1} = initial pre-collision angular velocity of bodies A, B
ω_{a2}, ω_{b2} = final post-collision angular velocity of bodies A, B
$\bar{\mathbf{v}}_{a1}, \bar{\mathbf{v}}_{b1}$ = initial pre-collision velocities of center of mass bodies A, B
$\bar{\mathbf{v}}_{a2}, \bar{\mathbf{v}}_{b2}$ = final post-collision velocities of center of mass bodies A, B
$\bar{\mathbf{v}}_{ap1}$ = initial pre-collision velocity of impact point on body A
$\bar{\mathbf{v}}_{bp1}$ = initial pre-collision velocity of impact point on body B
$\bar{\mathbf{n}}$ = normal (perpendicular) vector to edge of body B
e = elasticity (0 = inelastic, 1 = perfectly elastic)

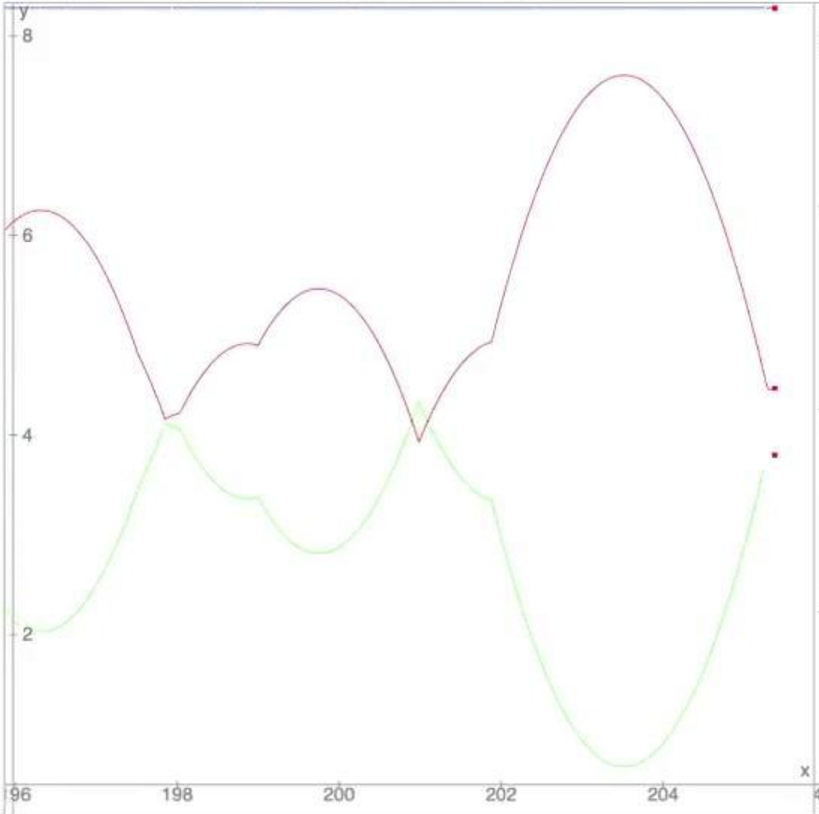


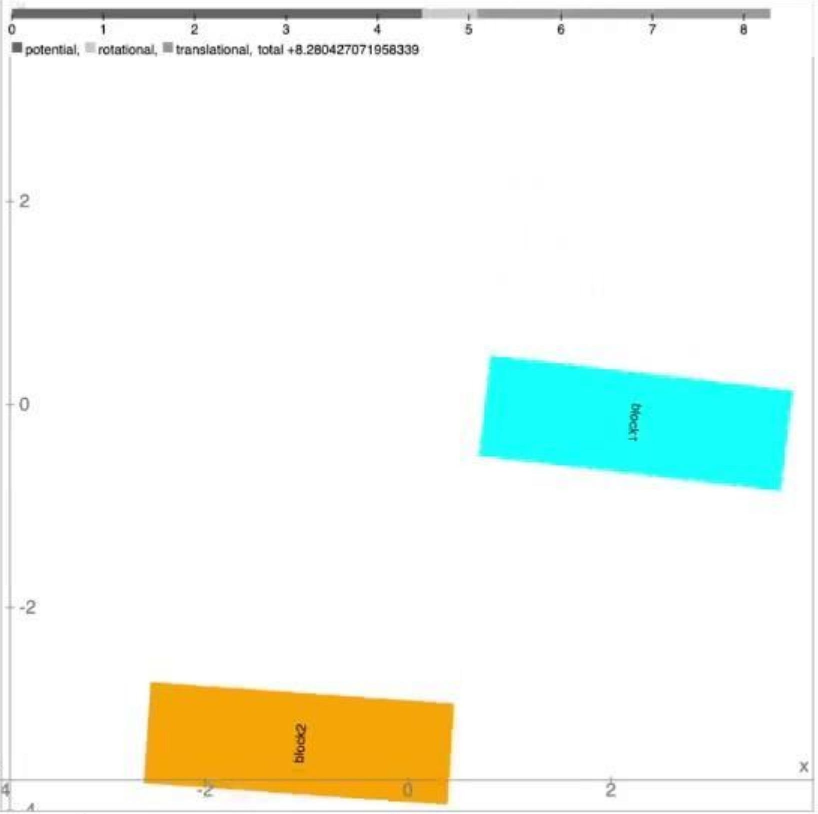
vectors involved in collision

http://www.myphysicslab.com/collision.html

Sim | Graph | Time Graph | Multi Graph

English | previous | next





green: kinetic energy

red: potential energy

blue: total energy

X: time

time window: 10.0

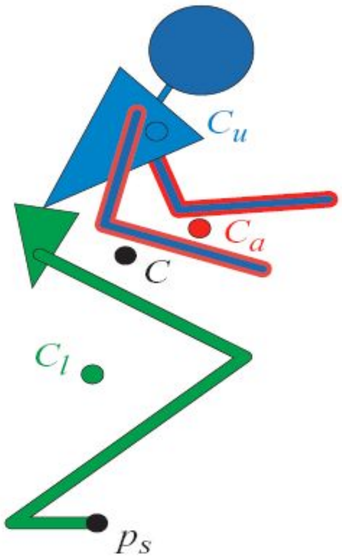
clear graph

pan-zoom

share

Center of Mass & Moment of Inertia

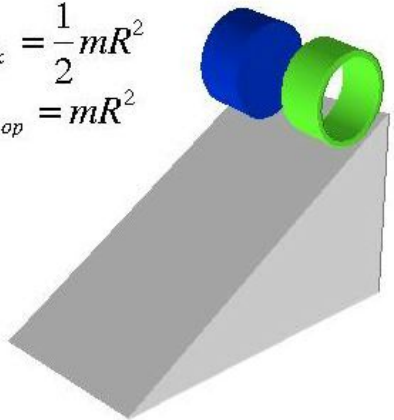
- Center of Mass: mean location of all mass in the system
- Moment of Inertia: a measure of an object's resistance to changes to its rotation
- If a solid cylinder & a hollow tube have the same radius & the same mass, which will reach the bottom of the ramp first?



Liu & Popović


http://en.wikipedia.org/wiki/Fosbury_Flop

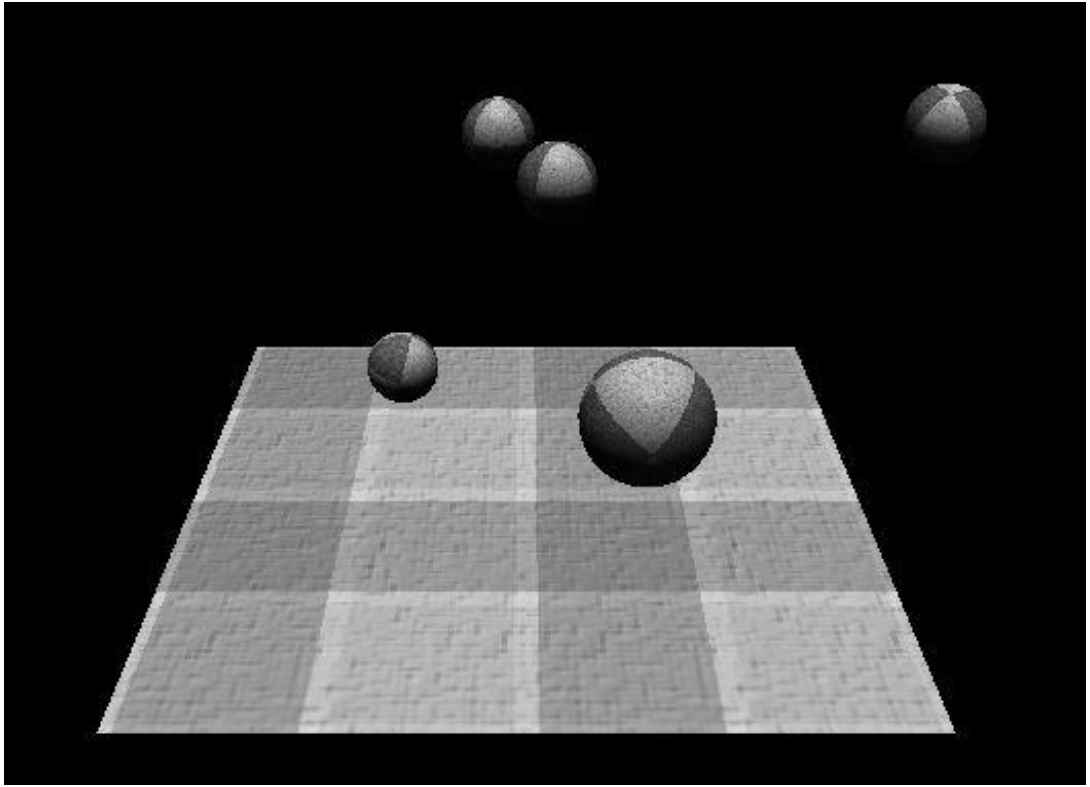
$$I_{disk} = \frac{1}{2} mR^2$$
$$I_{hoop} = mR^2$$



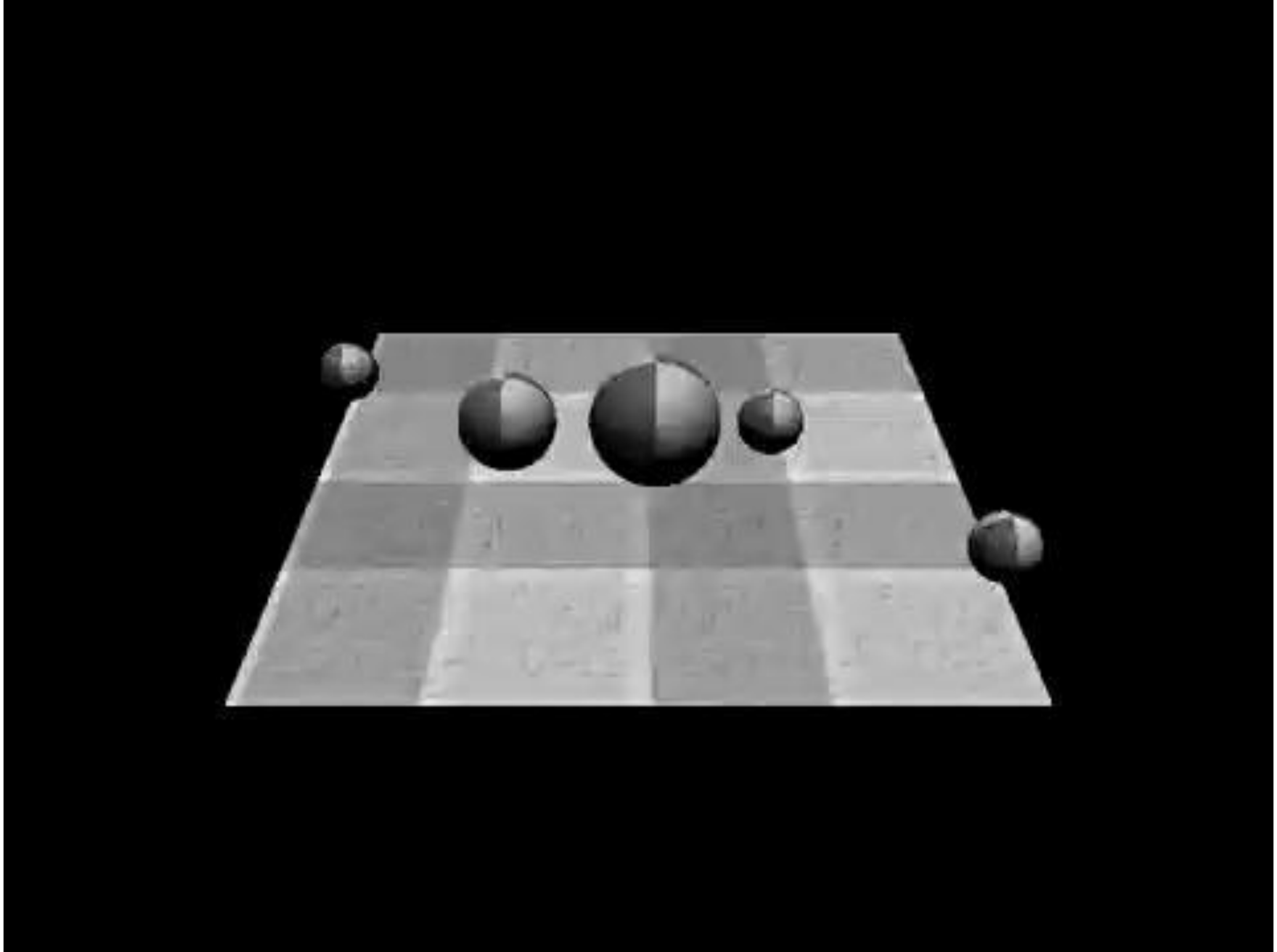
http://solomon.physics.sc.edu/~tedeschi/demo/demo12.html
http://hyperphysics.phy-astr.gsu.edu/hbase/hoocyl2.html

Rigid Body Dynamics

- Physics
  - Velocity
  - Acceleration
  - Angular Momentum
- Collisions
- Friction

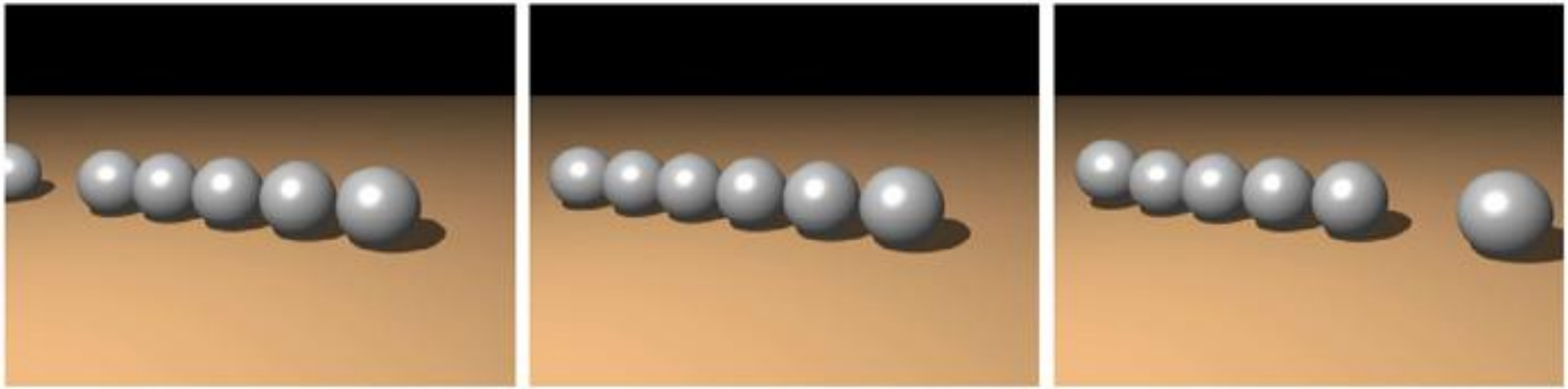


from: Darren Lewis
http://www-cs-students.stanford.edu/~dalewis/cs448a/rigidbody.html



Advanced Collisions

- What about friction?
- What if the contact between two objects is not a single point?
- What if more than two objects collide simultaneously?

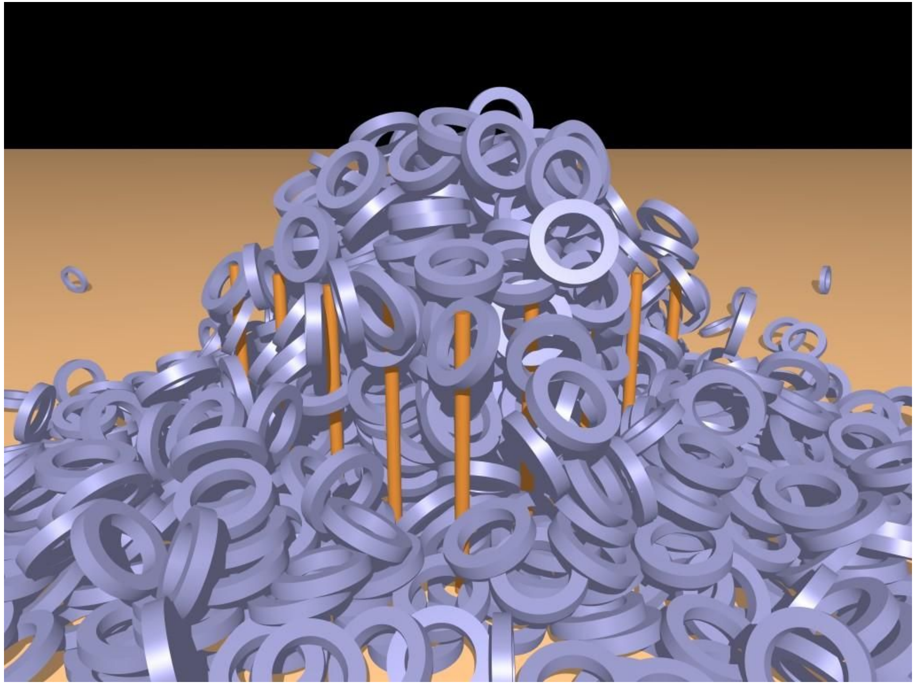


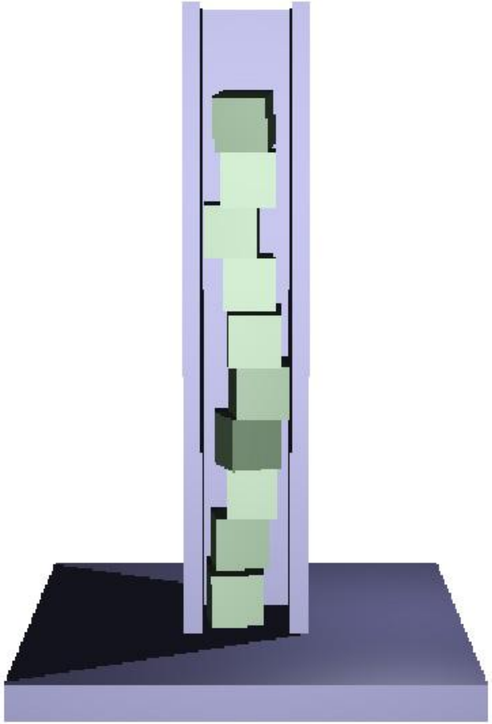
Guendelman, Bridson & Fedkiw
Nonconvex Rigid Bodies with Stacking
SIGGRAPH 2003

Resting Collisions

Victor J. Milenkovic & Harald Schmidl
Optimization-Based Animation
SIGGRAPH 2001

- We know how to simulate bouncing really well
- But resting collisions are harder to manage



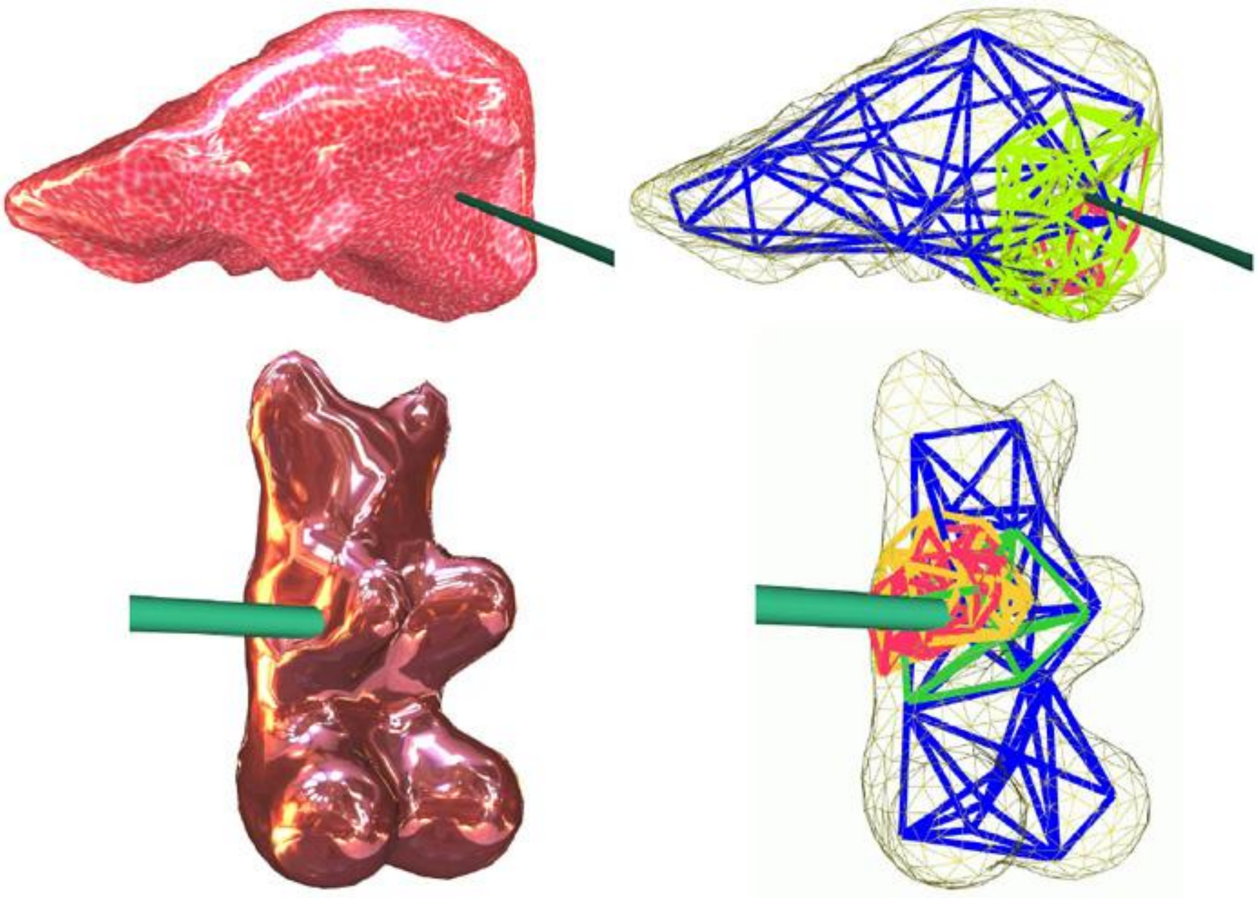


Guendelman, Bridson & Fedkiw
Nonconvex Rigid Bodies with Stacking,
SIGGRAPH 2003

Today

- Worksheet: Mass-Spring Cloth Simulation
- Readings for Today
- From Last Time: Data Structure & Algorithm for Fluid Simulation
- Rigid Body Dynamics
- Collision Response
- Non-Rigid, Deformable Objects
- Finite Element Method
- Papers for Tuesday

Deformation & Level of Detail



Gilles Debunne,
Mathieu Desbrun,
Marie-Paule Cani,
& Alan H. Barr
Dynamic Real-Time Deformations using Space & Time Adaptive Sampling
SIGGRAPH 2001

Simulation of Non-Rigid Objects

- We modeled string & cloth using mass-spring systems.
  Can we do the same?

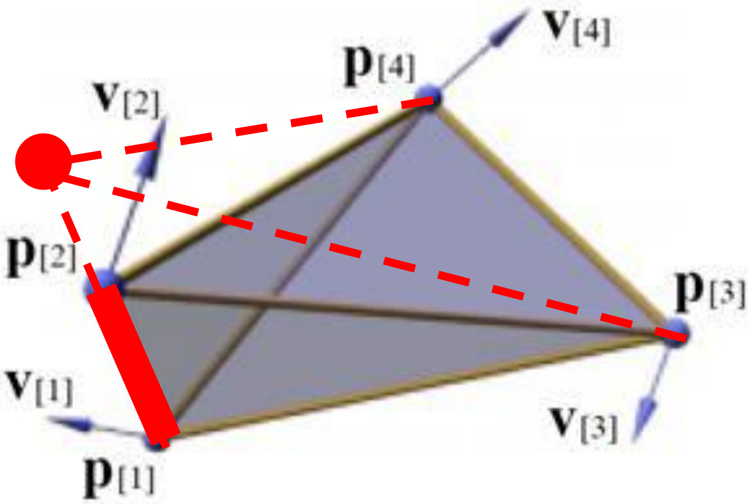
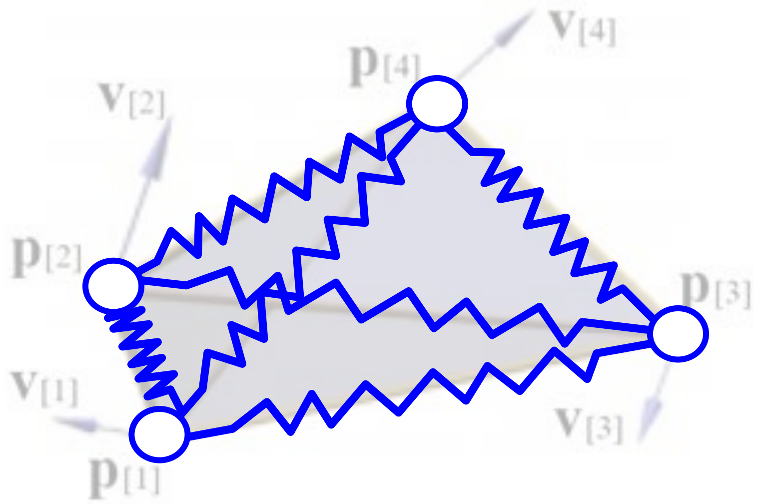


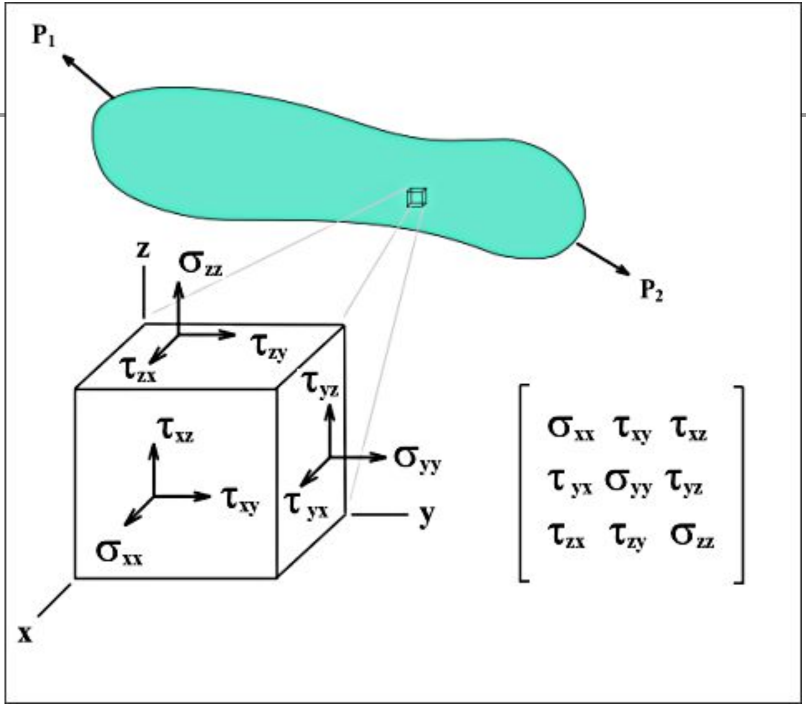
Image from O'Brien et al. 1999



- Yes… But a more physically accurate model uses *volumetric elements*

Strain & Stress

- Stress
  - the internal distribution of forces within a body that balance and react to the loads applied to it
  - *normal stress* & *shear stress*
- Strain
  - material deformation caused by stress.
  - measured by the change in length of a line or by the change in angle between two lines

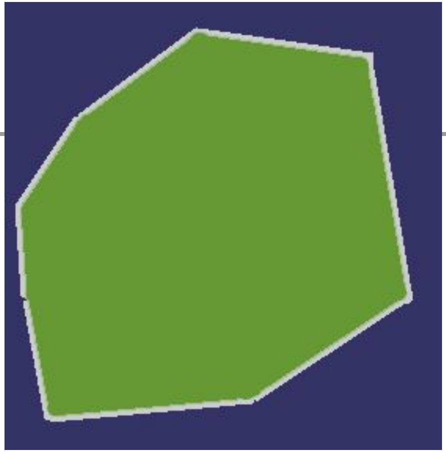


$$\varepsilon = \frac{\Delta_l}{l_0}$$

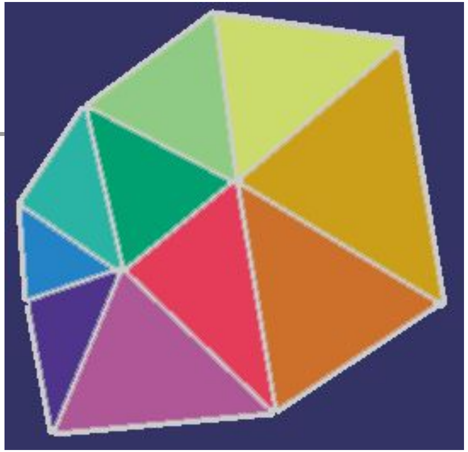
http://en.wikipedia.org/wiki/Image:Stress_tensor.png

Finite Element Method

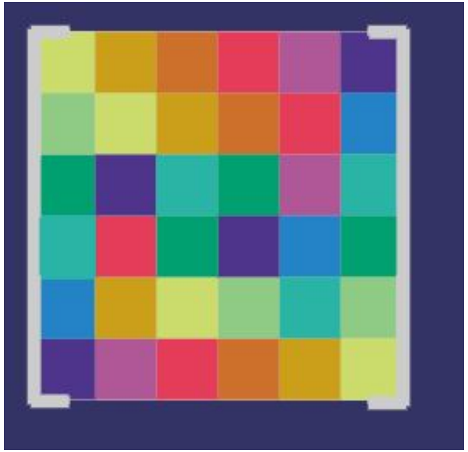
- To solve the continuous problem (deformation of all points of the object)
  - Discretize the problem
  - Express the interrelationship
  - Solve a big linear system
- More principled than Mass-Spring



object



finite elements



large matricial system

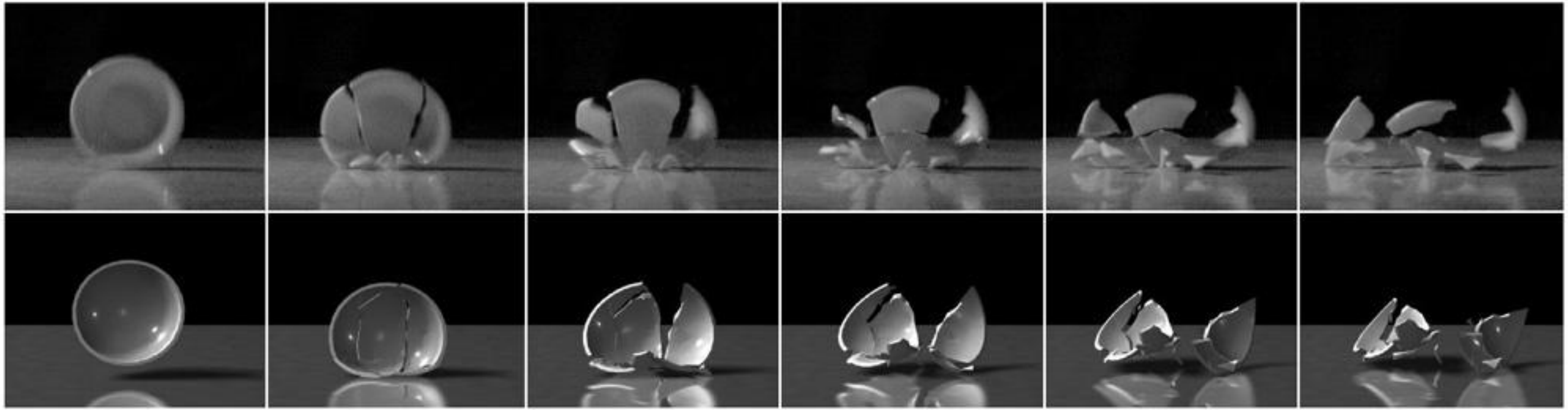
Diagram from Debunne et al. 2001

Today

- Worksheet: Mass-Spring Cloth Simulation
- Readings for Today
- From Last Time: Data Structure & Algorithm for Fluid Simulation
- Rigid Body Dynamics
- Collision Response
- Non-Rigid, Deformable Objects
- Finite Element Method
- Papers for Tuesday

Readings for Tuesday *(pick one)*

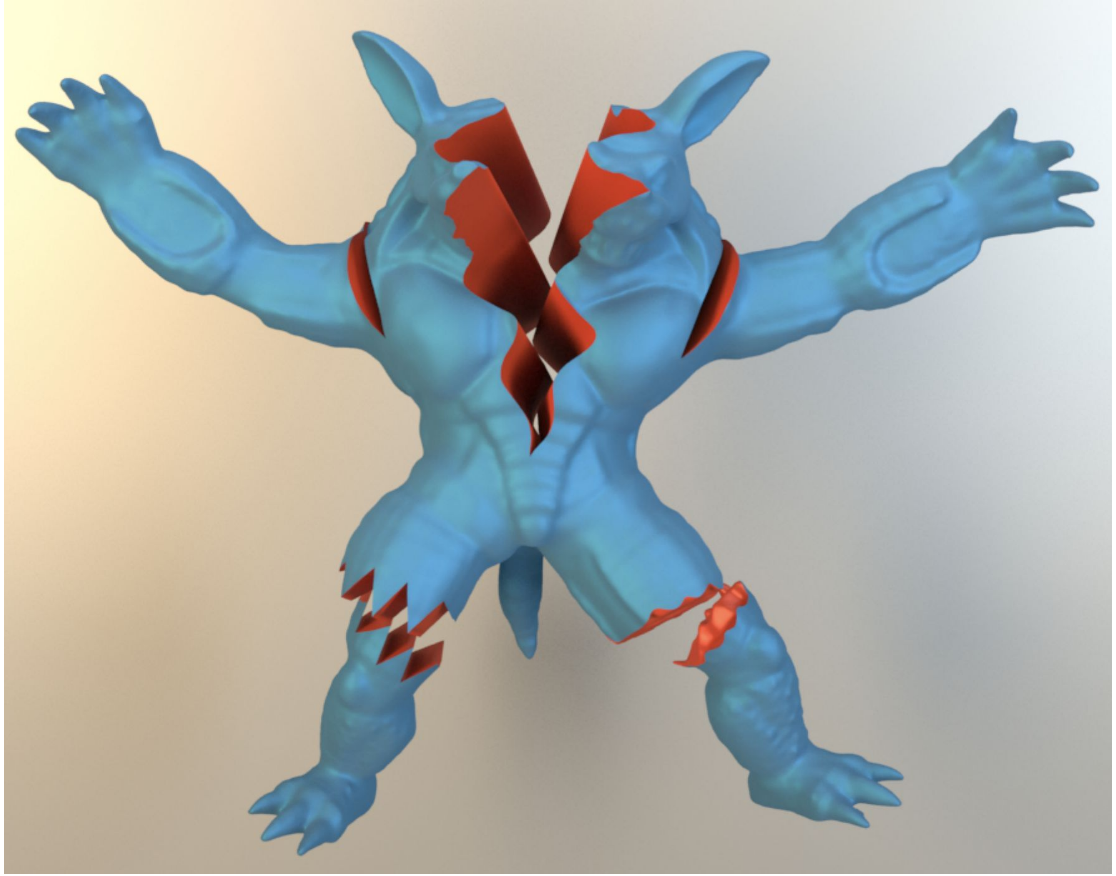
- James O'Brien & Jessica Hodgins "*Graphical Modeling and Animation of Brittle Fracture*" SIGGRAPH 1999.



- Fracture threshold
- Remeshing
  - need connectivity info!
- Material properties
- Parameter tuning

Readings for Tuesday *(pick one)*

- "Robust eXtended Finite Elements for Complex Cutting of Deformables", Koschier, Bender, & Thuerey, SIGGRAPH 2017



Readings for Tuesday *(pick one)*

- "Multi-species simulation of porous sand and water mixtures", Pradhana, Gast, Klar, Fu, Teran, Jiang, and Museth, SIGGRAPH 2017.