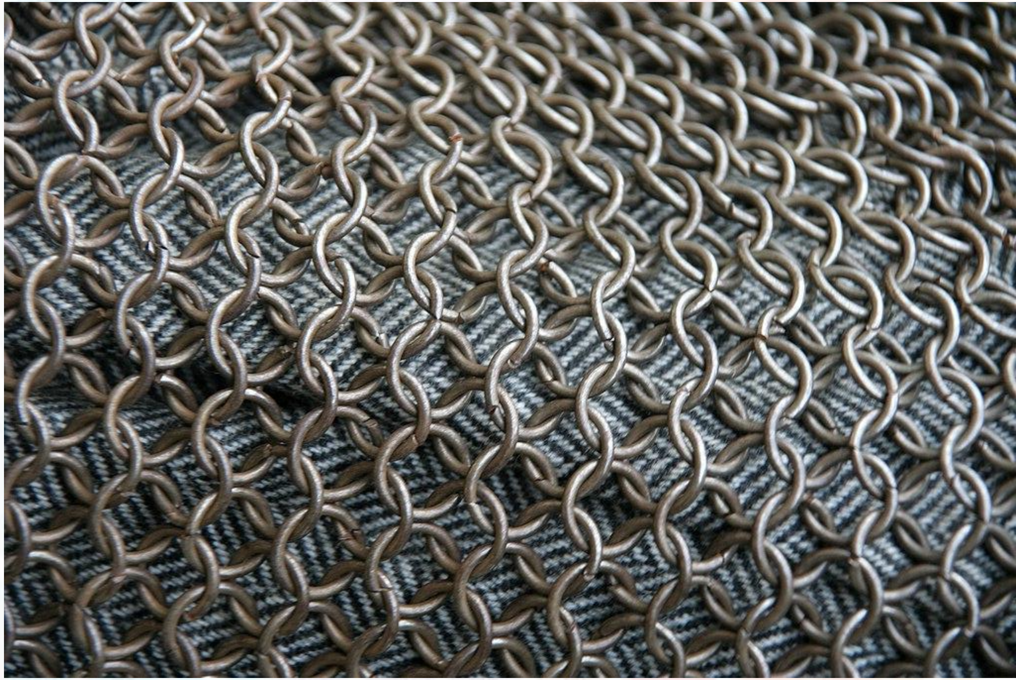
CSCI 4530/6530 Advanced Computer Graphics

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

Lecture 6: Mass-Spring Systems



https://simple.wikipedia.org/wiki/File:Roman_chainmail_detail.jpg

High Fashion in Equations

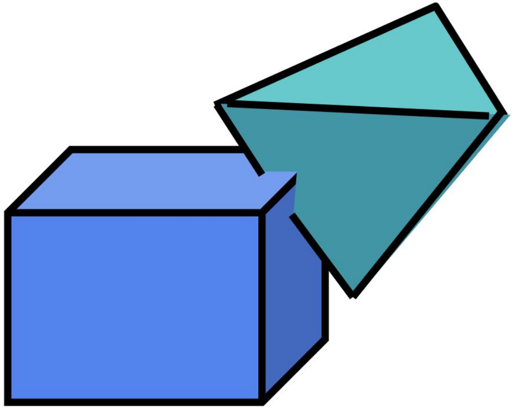
MIRALab,
University of Geneva,
SIGGRAPH 2007

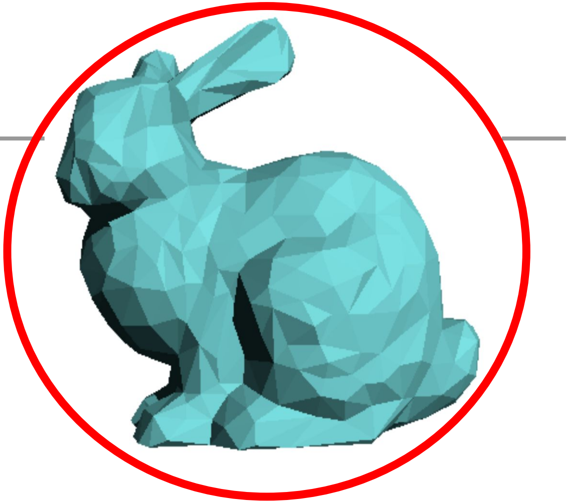
Simulating Knitted Cloth at the Yarn Level

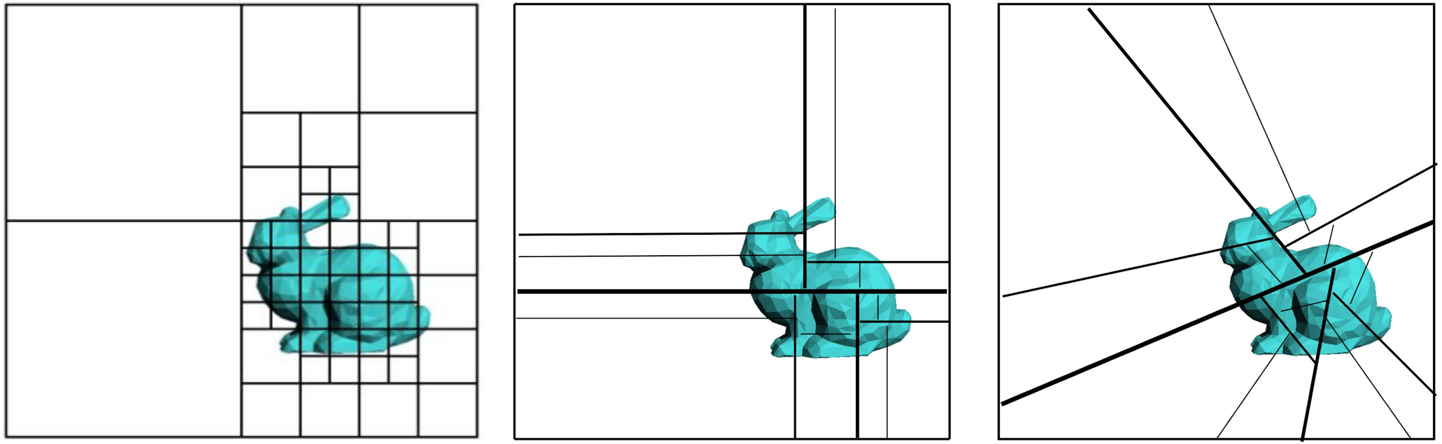
Kaldor, James, & Marshner, SIGGRAPH 2008

Last Time?

- Collision Detection
- Conservative Bounding Regions
- Spatial Acceleration Data Structures
    - Octree
    - k-d Tree
    - BSP



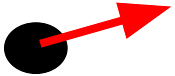
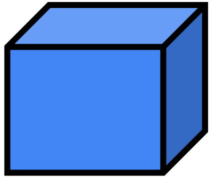


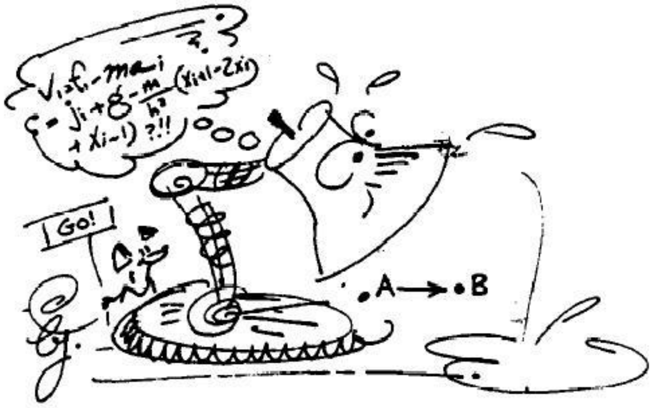


Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

Types of Dynamics

- Point 
- Rigid body 
- Deformable body
  (include clothes, fluids, smoke, etc.)

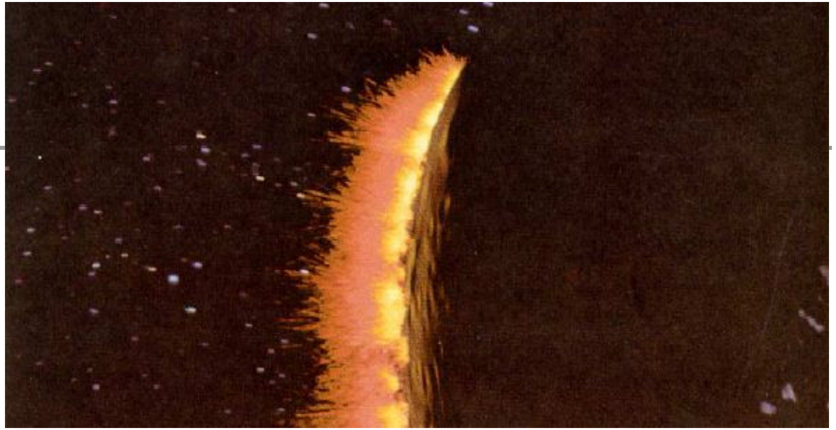


Witkin & Kass, "Spacetime Constraints", 1988

Carlson, Mucha, Van Horn, & Turk 2002

What is a Particle System?

- Collection of many small simple particles that maintain *state* (position, velocity, color, etc.)
- Particle motion influenced by external *force fields*
- *Integrate* the laws of mechanics (ODE Solvers)
- To model: sand, dust, smoke, sparks, flame, water, etc.



Star Trek, The Wrath of Kahn, 1982

Sateesh Malla, 2008, http://www.sateeshmalla.com/blog/2008/05/water-fountain-simulation/

Particle Motion

- mass *m*, position *x*, velocity *v*
- equations of motion:

$$\frac{d}{dt}x(t) = v(t)$$

$$\frac{d}{dt}v(t) = \frac{1}{m}F(x, v, t)$$

F = *ma*

- Analytic solutions can be found for some classes of differential equations, but most can't be solved analytically
- Instead, we will numerically approximate a solution to our *initial value problem*

Higher Order ODEs

- Basic mechanics is a 2nd order ODE:

$$\frac{d^2}{dt^2}x = \frac{1}{m}F$$

- Express as 1st order ODE by defining *v*(*t*):

$$\frac{d}{dt}x(t) = v(t)$$

$$\frac{d}{dt}v(t) = \frac{1}{m}F(x, v, t)$$

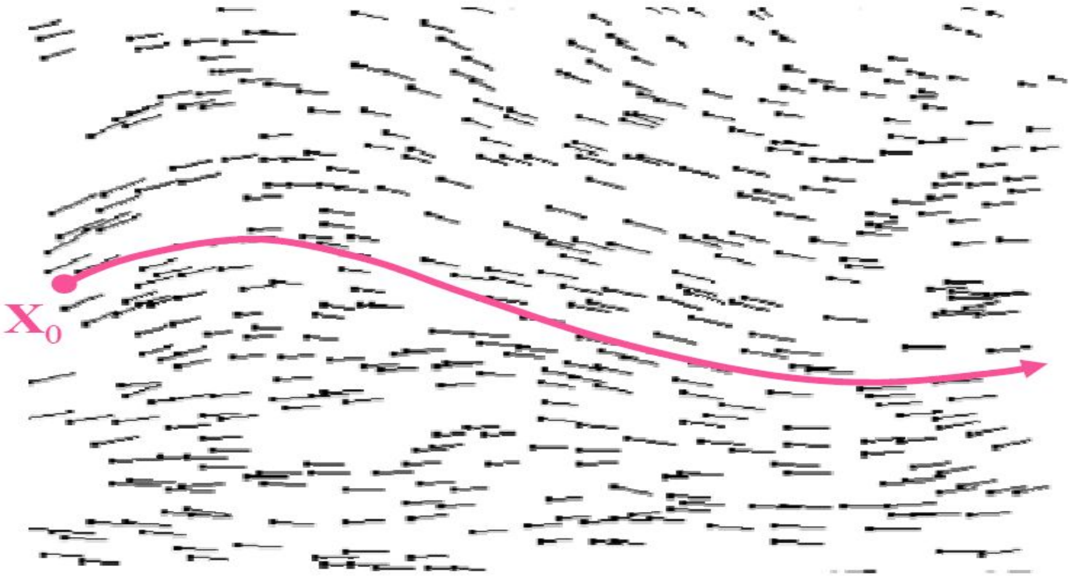
$$\mathbf{X} = \begin{pmatrix} x \\ v \end{pmatrix} \qquad f(X, t) = \begin{pmatrix} v \\ \frac{1}{m}F(x, v, t) \end{pmatrix}$$

X is a vector storing the *current state* of the particle

f (***X***,*t*) describes how to *update* the state of the particle

Path Through a Field

- *f* (**X**,*t*) is a vector field defined everywhere
  - E.g. a velocity field which may change over time



Note: In the simplest particle systems, the particles do not interact with each other, only with external force fields

- ***X***(*t*) is a path through the field

For a Collection of 3D particles…

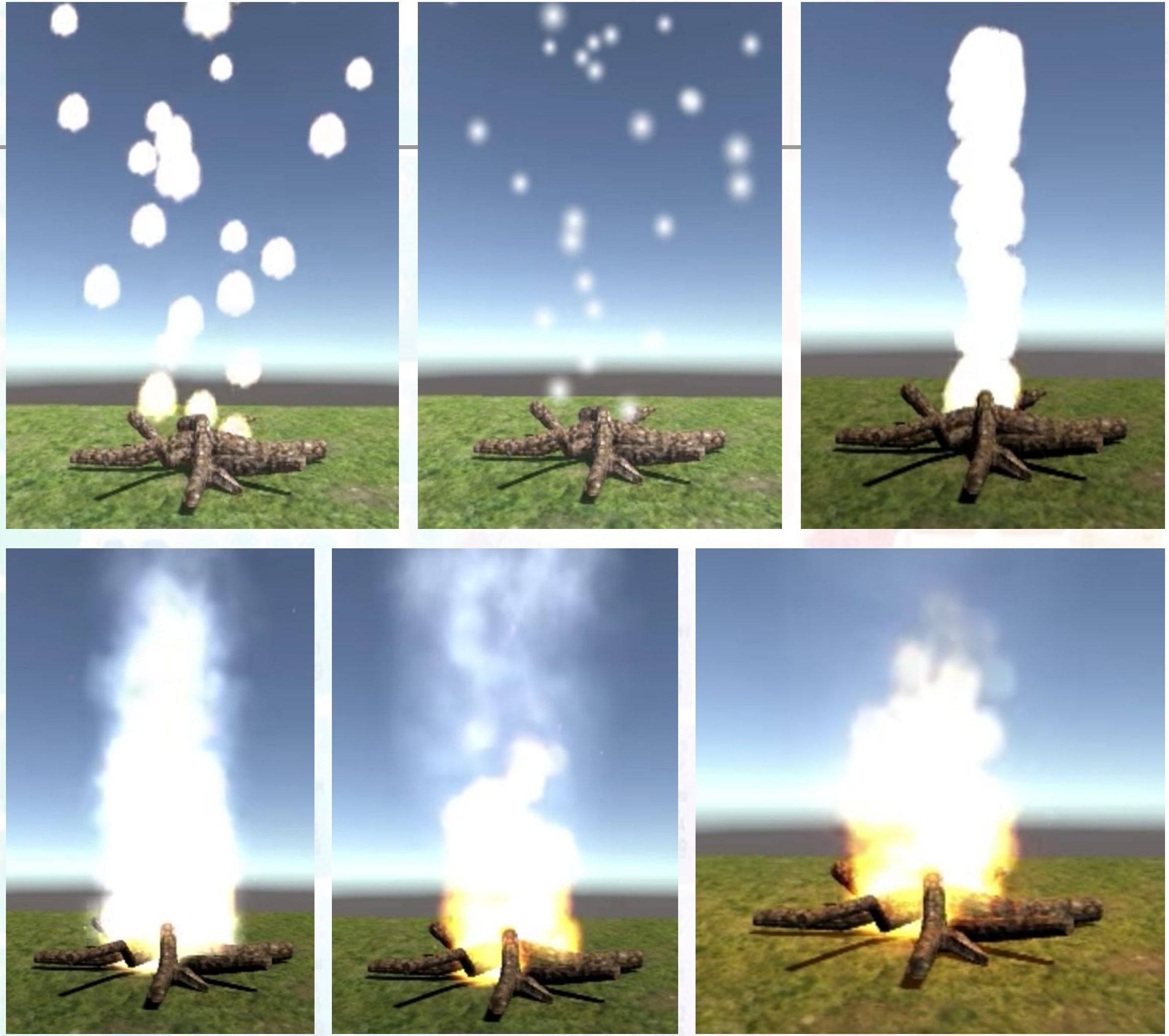
$$
\mathbf{X} = \begin{pmatrix} p_x^{(1)} \\ p_y^{(1)} \\ p_z^{(1)} \\ v_x^{(1)} \\ v_y^{(1)} \\ v_z^{(1)} \\ p_x^{(2)} \\ p_y^{(2)} \\ p_z^{(2)} \\ v_x^{(2)} \\ v_y^{(2)} \\ v_z^{(2)} \\ \vdots \end{pmatrix} \qquad f(\mathbf{X},t) = \begin{pmatrix} v_x^{(1)} \\ v_y^{(1)} \\ v_z^{(1)} \\ \frac{1}{m_1} F_x^{(1)}(\mathbf{X},t) \\ \frac{1}{m_1} F_y^{(1)}(\mathbf{X},t) \\ \frac{1}{m_1} F_z^{(1)}(\mathbf{X},t) \\ v_x^{(2)} \\ v_y^{(2)} \\ v_z^{(2)} \\ \frac{1}{m_2} F_x^{(2)}(\mathbf{X},t) \\ \frac{1}{m_2} F_y^{(2)}(\mathbf{X},t) \\ \frac{1}{m_2} F_z^{(2)}(\mathbf{X},t) \\ \vdots \end{pmatrix}
$$

more generally, we can define ***X*** *as a huge vector storing the current state of all particles in a system*

Questions?

- Current state ***X*** can also include color & transparency
- *f(**X**,t)* can animate changes in these values over time

MixPixVisuals
Mikael Bellander
https://www.youtube.com/watch?v=M-Hz9Za5mCE

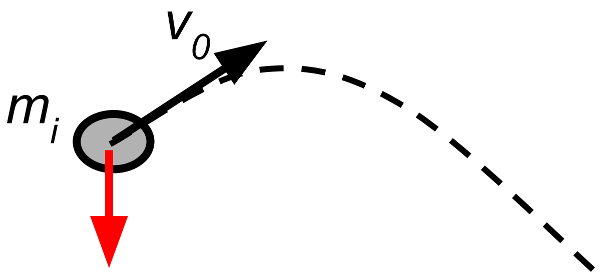


Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

Forces: Gravity

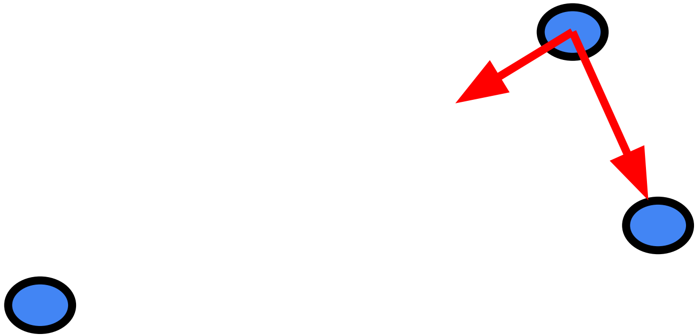
- Simple gravity: depends only on particle mass



For smoke, flame: make "gravity" point up!

$$\text{Gravity: } f^{(i)} = \begin{pmatrix} 0 \\ 0 \\ -m_i G \end{pmatrix}$$

- N-body problem: depends on all other particles
  - Magnitude inversely proportional to square distance
  - $F_{ij} = G\, m_i m_j / r^2$



Quickly gets impractical to compute analytically. Expensive to numerically approximate too!

Forces: Spatial Fields

- Force on particle *i* depends only on position of *i*
  - wind
  - attractors
  - repulsors
  - vortices
- Can depend on time (e.g., wind gusts)

- *Note: these forces will generally add energy to the system, and thus may need damping…*

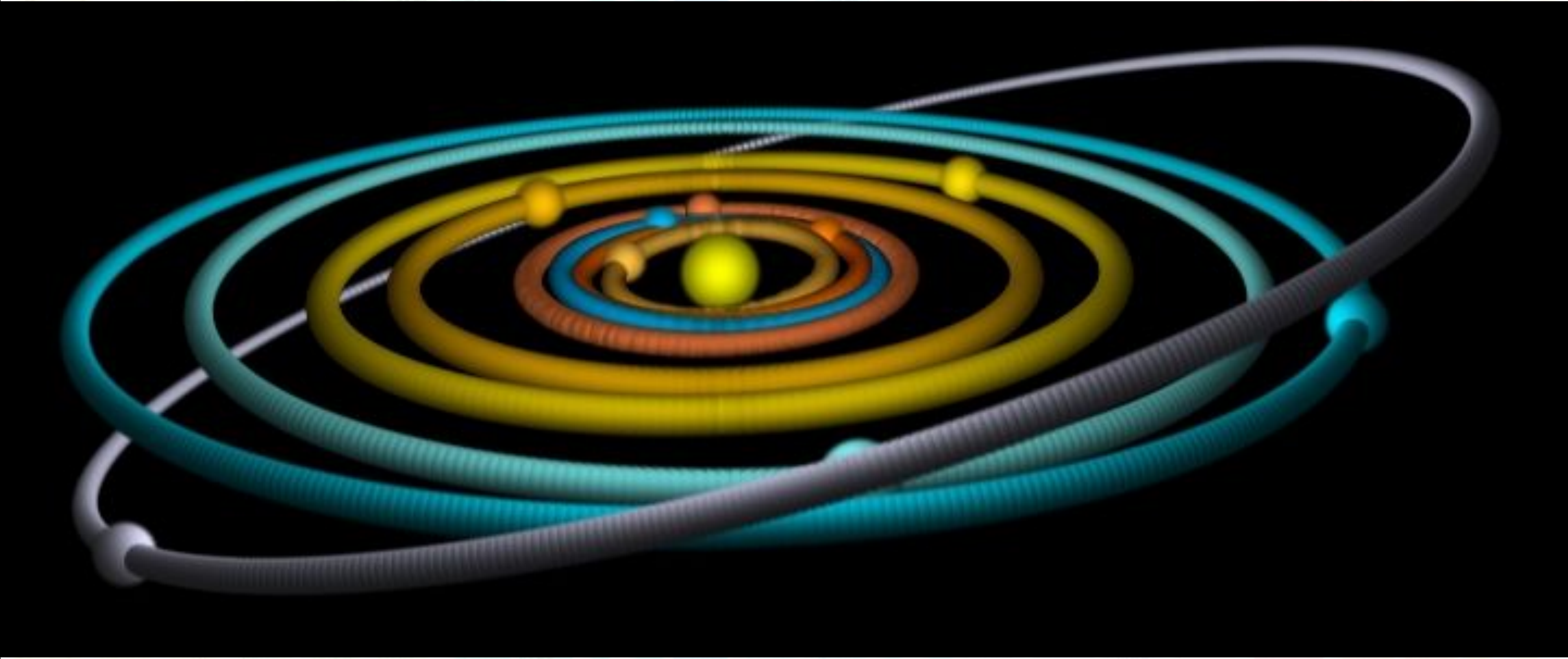
Forces: Damping

- Force on particle *i* depends only on velocity of *i*

$$f^{(i)} = -dv^{(i)}$$

- Force opposes motion
  - *A hack mimicking real-world friction/drag*
  - *Friction is complicated…*
- Damping removes energy, so system can settle
- Small amount of damping can stabilize solver
- Too much damping makes motion too glue-like and unrealistic

Questions?



http://www.lactamme.polytechnique.fr/Mosaic/images/NCOR.U1.2048.D/display.html

Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

Euler's Method

- Examine *f* (**X**,*t*) at (or near) current state
- Take a step of size *h* to new value of **X**:

$$t_1 = t_0 + h$$

$$\mathbf{X}_1 = \mathbf{X}_0 + h\, f(\mathbf{X}_0, t_0)$$

$$\mathbf{X} = \begin{pmatrix} x \\ v \end{pmatrix} \quad f(X,t) = \begin{pmatrix} v \\ \frac{1}{m} F(x,v,t) \end{pmatrix}$$

update the position by adding a little bit of the current velocity

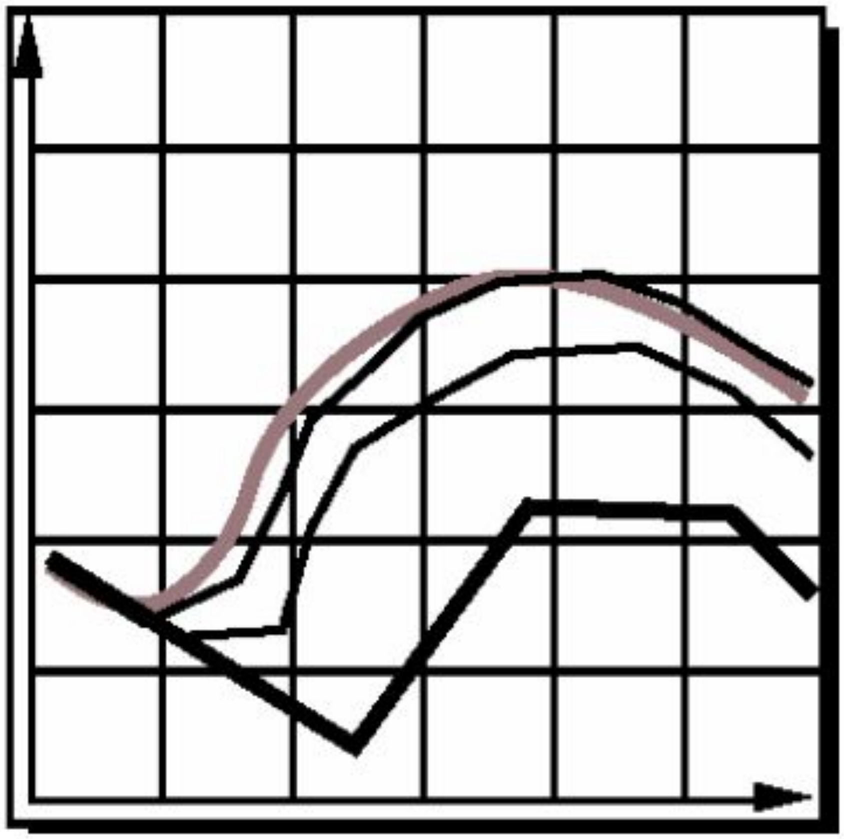
&

update the velocity by adding a little bit of the current acceleration

- Piecewise-linear approximation to the curve

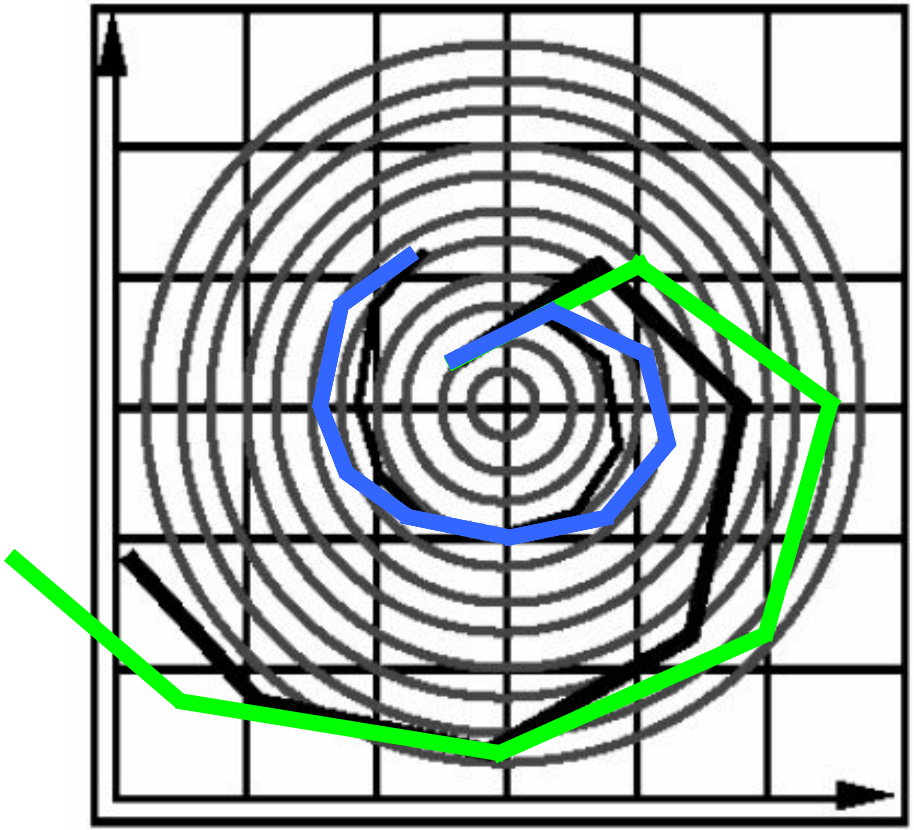
Effect of Step Size

- Step size controls accuracy
- Smaller steps more closely follow curve
- For animation, we may want to take *many* small steps per frame
  - How many frames per second for animation?
  - How many steps per frame?



Euler’s Method: Inaccurate

- Simple example: particle in stable circular orbit around planet (origin)
- Current velocity is always tangent to circle
- Force is perpendicular to circle
- Euler method will spiral outward no matter how small *h* is



Euler’s Method: Unstable

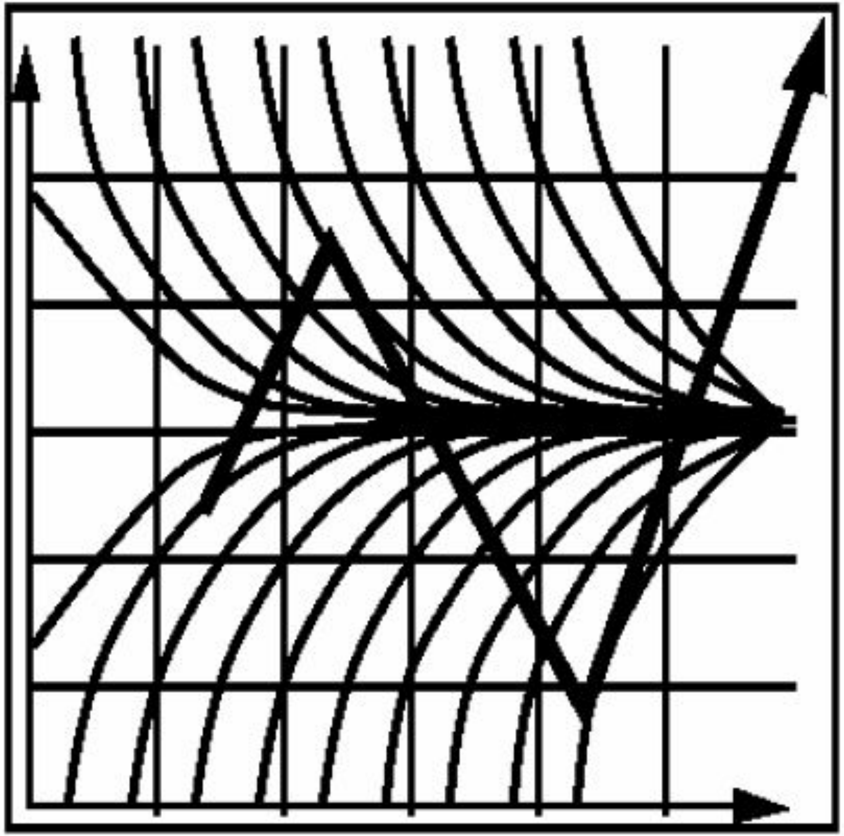
- Problem: $f(x,t) = -kx$
- Solution: $x(t) = x_0 e^{-kt}$

- Limited step size:

$$x_1 = x_0(1 - hk)$$

$$\begin{cases} h \le 1/k & \text{ok} \\ h > 1/k & \text{oscillates} \pm \\ h > 2/k & \text{explodes} \end{cases}$$

- If *k* is big, *h* must be small



Analysis using Taylor Series

- Expand exact solution **X***(t)*

$$\mathbf{X}(t_0+h)=\mathbf{X}(t_0)+h\left(\tfrac{d}{dt}\mathbf{X}(t)\right)\Big|_{t_0}+\tfrac{h^2}{2!}\left(\tfrac{d^2}{dt^2}\mathbf{X}(t)\right)\Big|_{t_0}+\tfrac{h^3}{3!}\left(\cdots\right)+\cdots$$

- Euler's method:

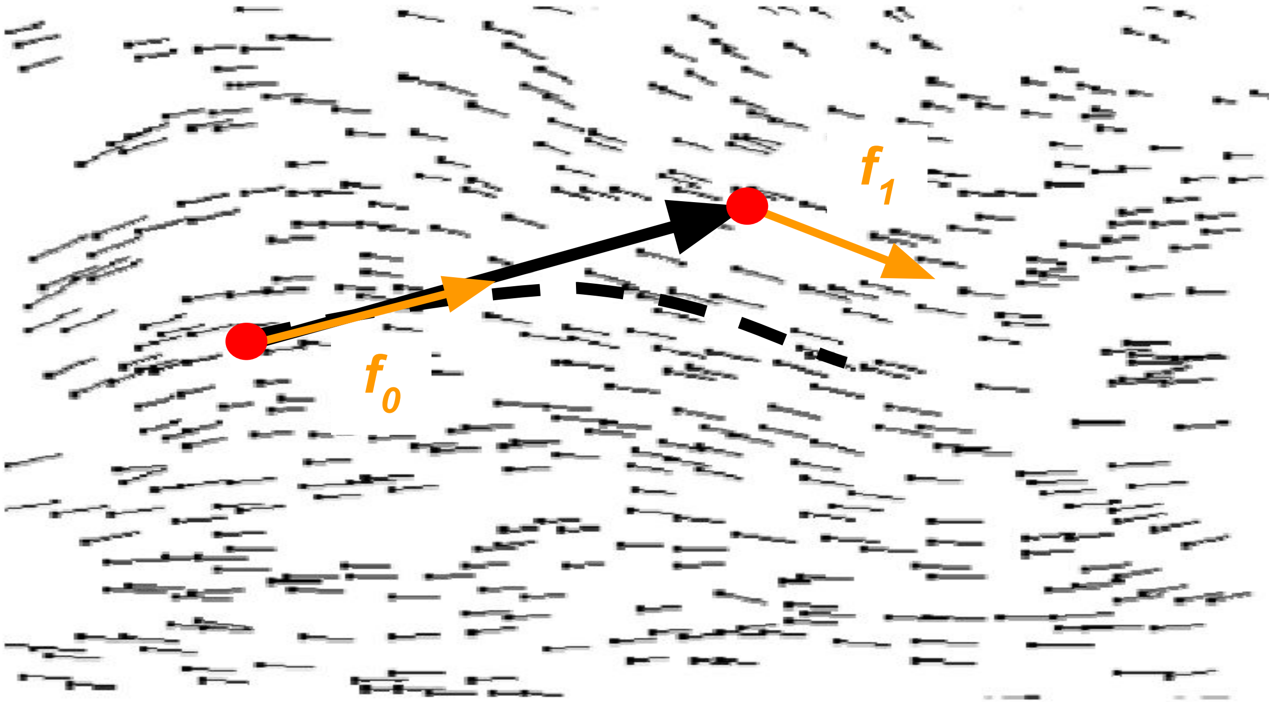
$$\mathbf{X}(t_0+h)=\mathbf{X}_0+h\,f(\mathbf{X}_0,t_0)\qquad \ldots+O(h^2)\,\text{error}$$

$$h\rightarrow h/2\ \Rightarrow\ error\rightarrow error/4\,\text{per step}\times\text{twice as many steps}$$
$$\rightarrow error/2$$

- First-order method: Accuracy varies with *h*
  - To get 100x better accuracy need 100x more steps

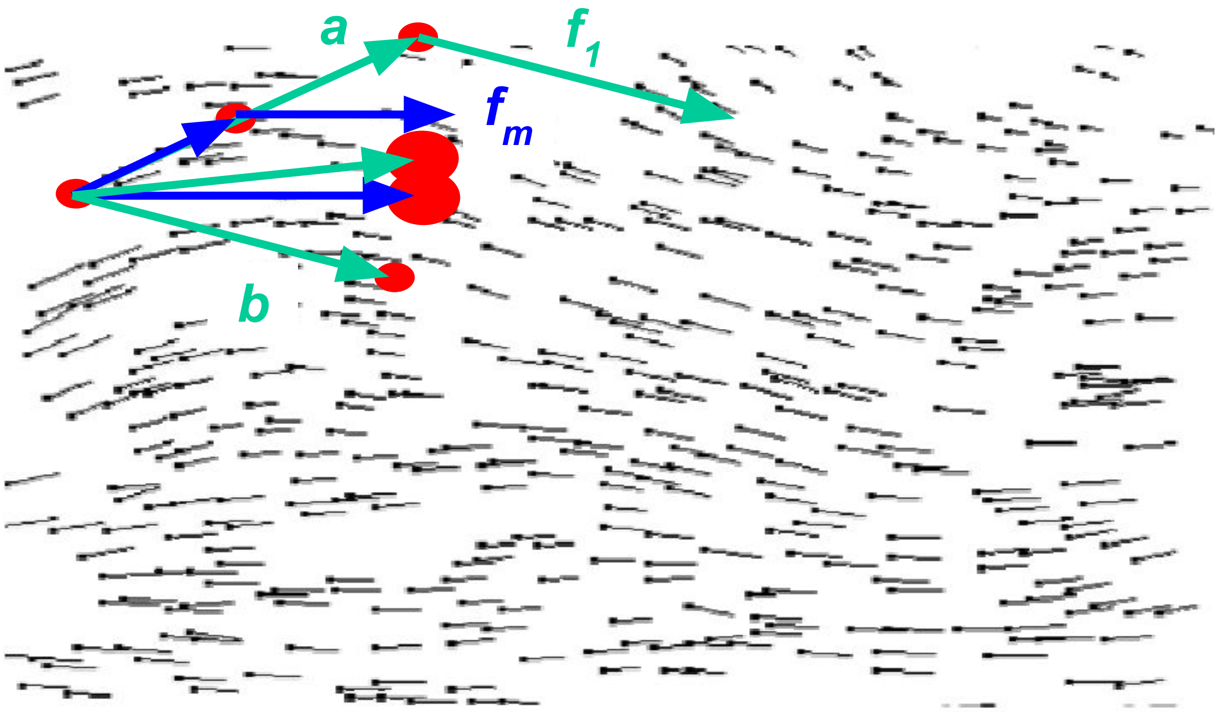
Can we do better than Euler’s Method?

- Problem: f has varied along the step
- Idea: look at f at the arrival of the step and compensate for variation



2nd-Order Methods

- Midpoint:
  - ½ Euler step
  - evaluate f_m
  - full step using f_m
- Trapezoid:
  - Euler step (a)
  - evaluate f_1
  - full step using f_1 (b)
  - average (a) and (b)



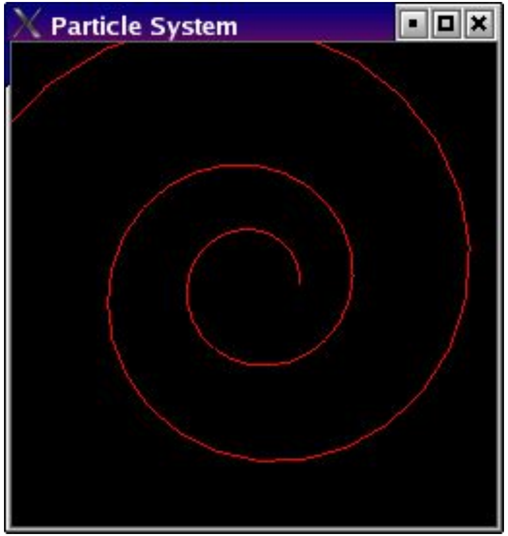
Midpoint & trapezoid do not yield exactly the same result, but they have same order of accuracy

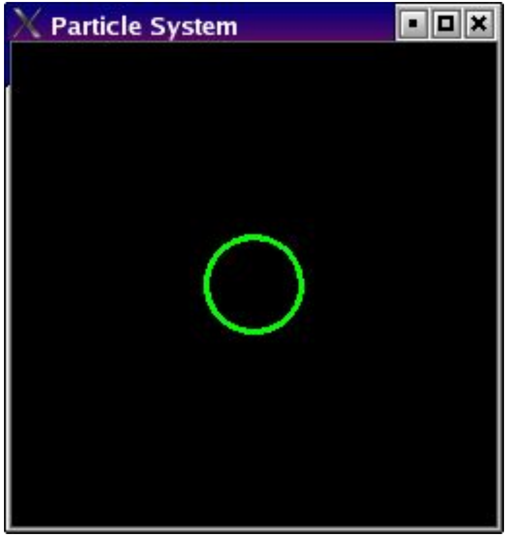
Comparison: Euler, Midpoint, Runge-Kutta

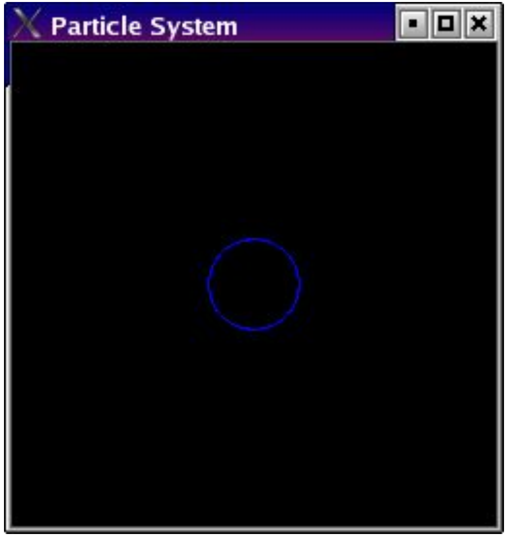
A 4th order method!

- *initial position* : (1,0,0)
- *initial velocity* : (0,5,0)
- *force field* : pulls particles to origin with magnitude proportional to distance from origin
- *correct answer* : circle

Euler will always diverge (even with small dt)





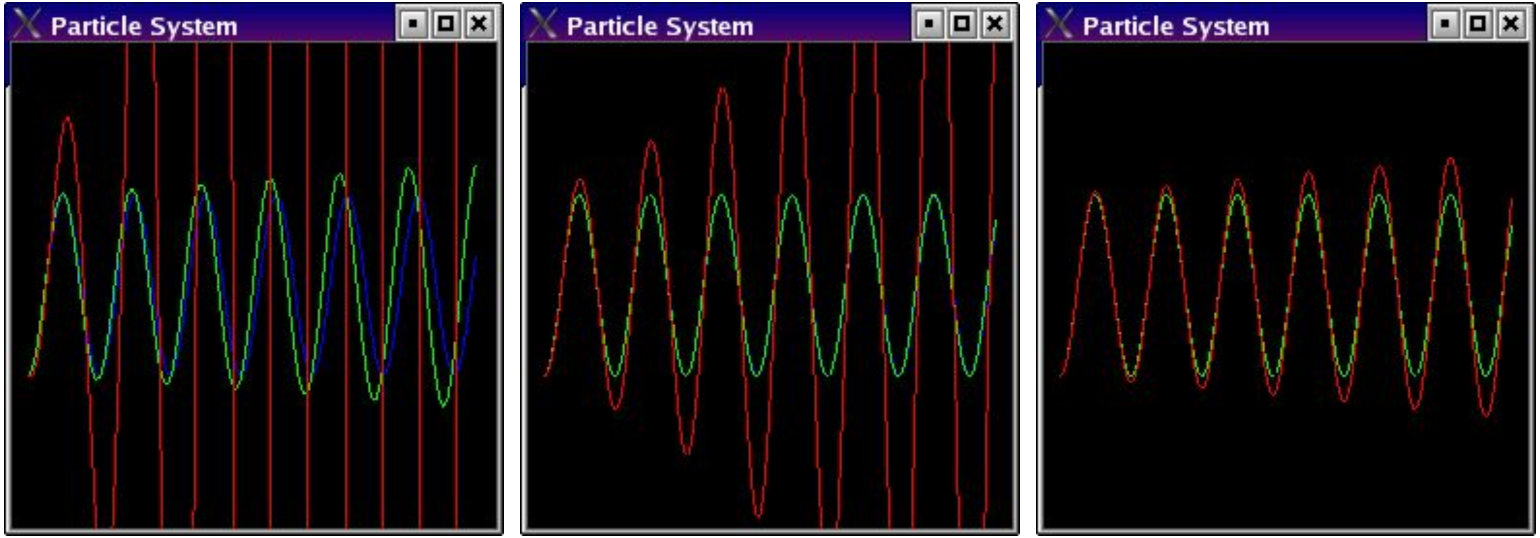


Comparison: Euler, Midpoint, Runge-Kutta

A 4th order method!

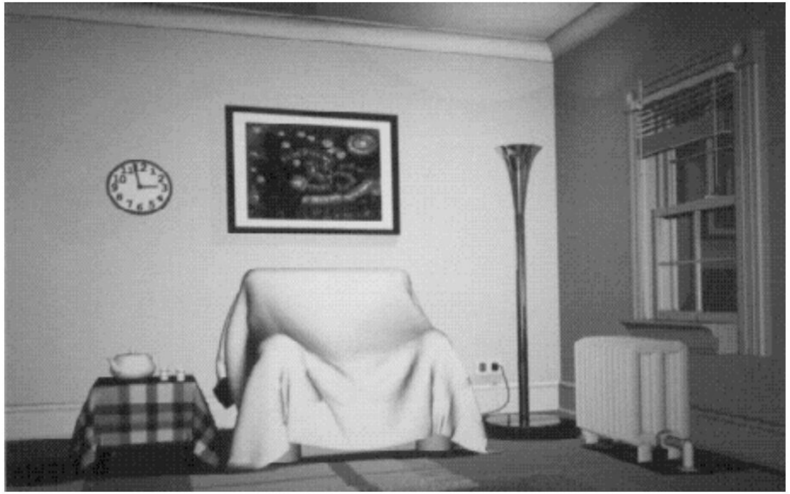
- *initial position* : (0,-2,0)
- *initial velocity* : (1,0,0)
- *force field* : pulls particles to line y=0 with magnitude proportional to distance from line
- *correct answer* : sine wave

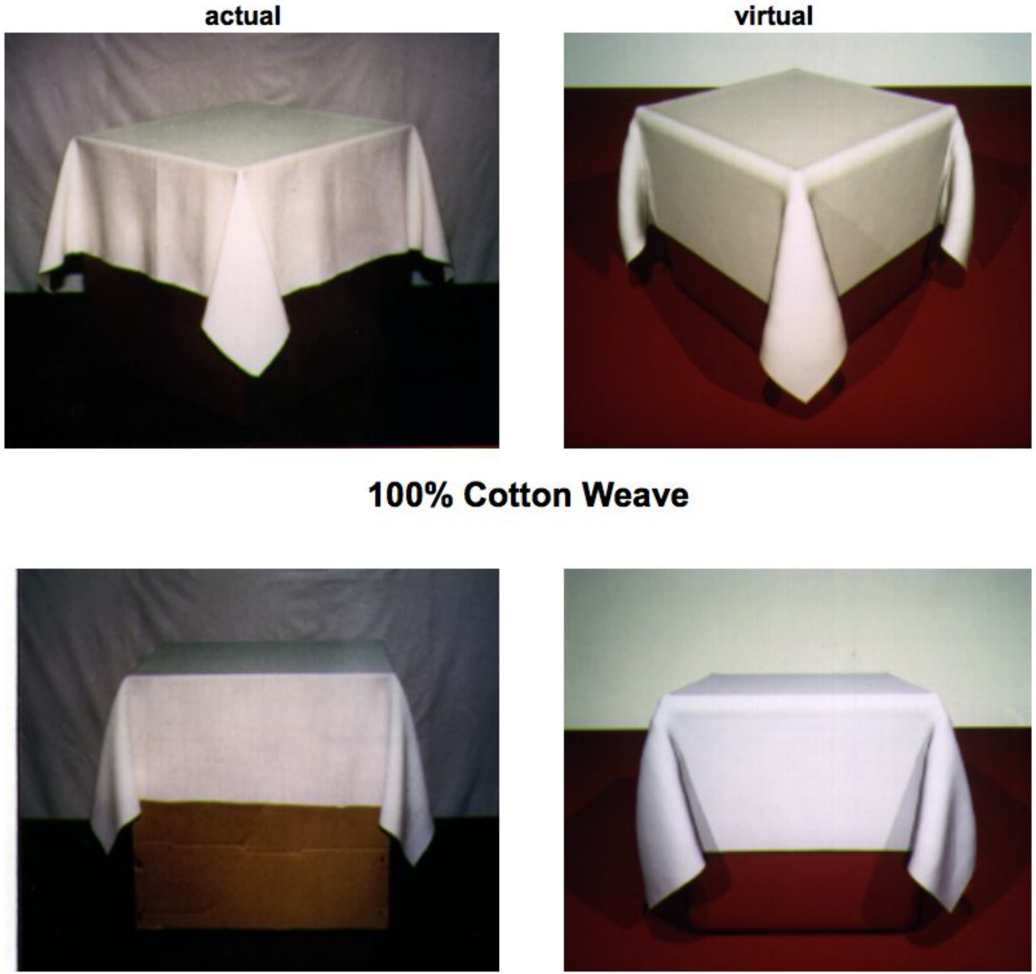
Decreasing the timestep (dt) improves the accuracy



Questions?

"Predicting the Drape of Woven Cloth Using Interacting Particles"
Breen, House, and Wozny, SIGGRAPH 1994



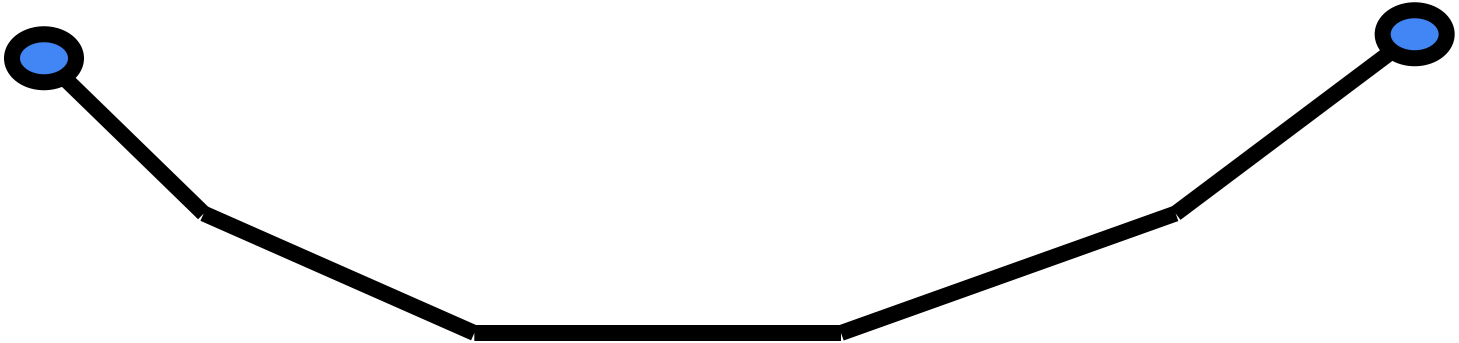


Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

How would you simulate a string?

- Each particle is linked to two particles
- Forces try to keep the distance between particles constant
- What force?

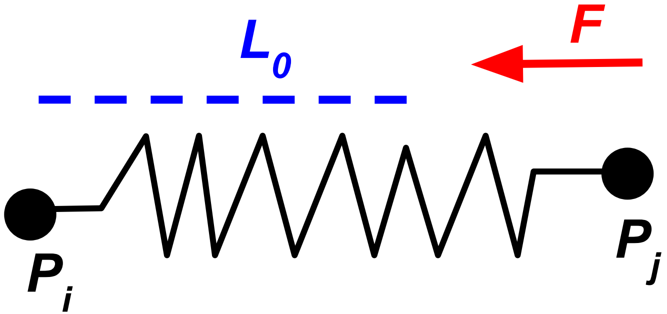


Spring Forces

- *Force* in the direction of the spring and proportional to difference with rest length L_0

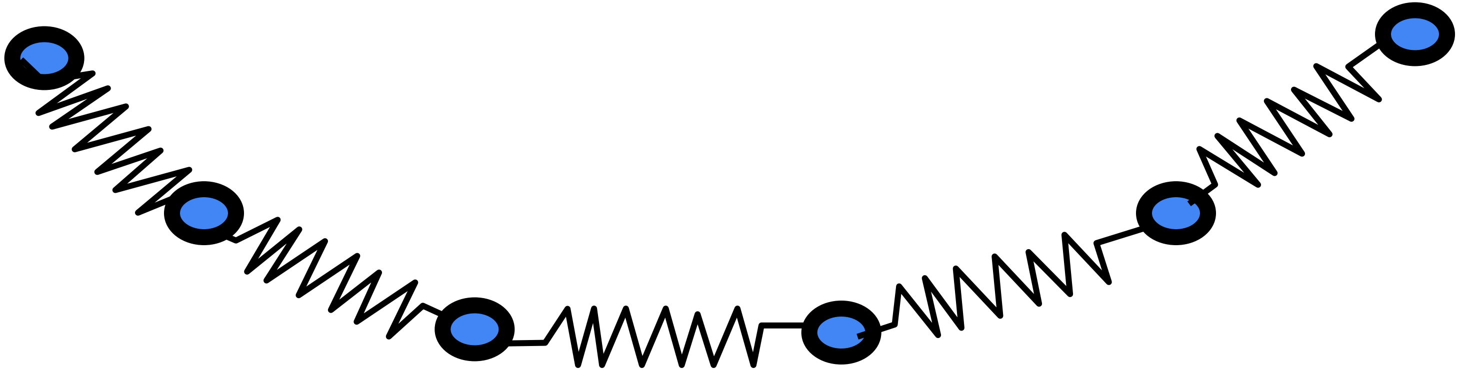
$$F(P_i, P_j) = K(L_0 - ||\vec{P_iP_j}||)\frac{\vec{P_iP_j}}{||\vec{P_iP_j}||}$$

- K is the stiffness of the spring
  - *When K gets bigger, the spring really wants to keep its rest length*



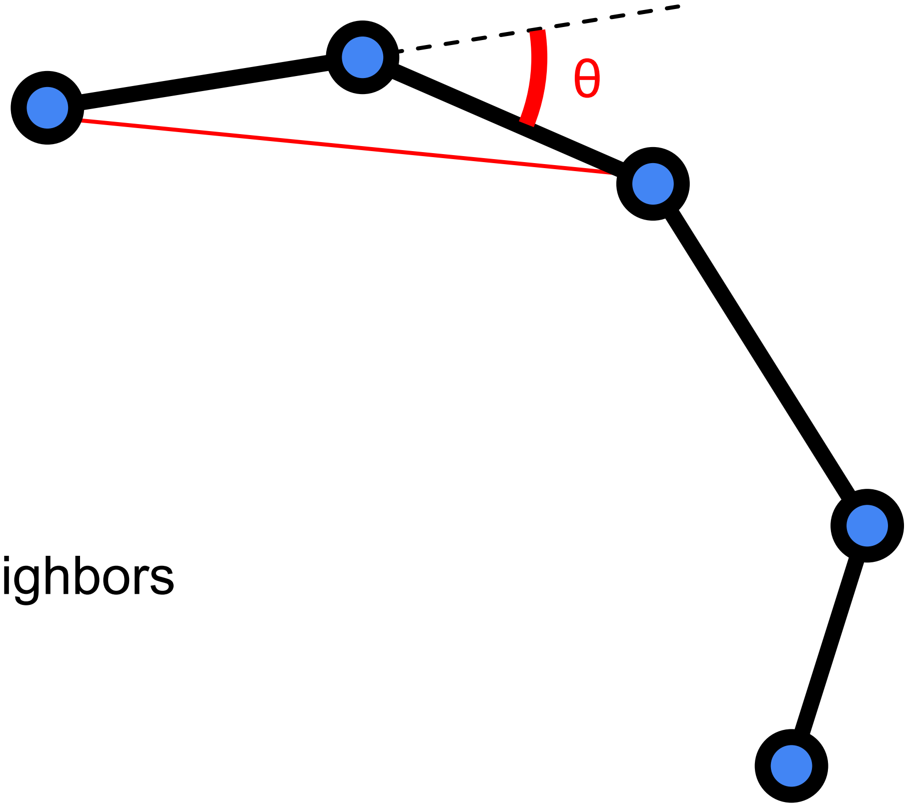
How would you simulate a string?

- Springs link the particles
- Springs try to keep their rest lengths and preserve the length of the string
- Problems?
  - Stretch, actual length will be greater than rest length
  - Numerical oscillation



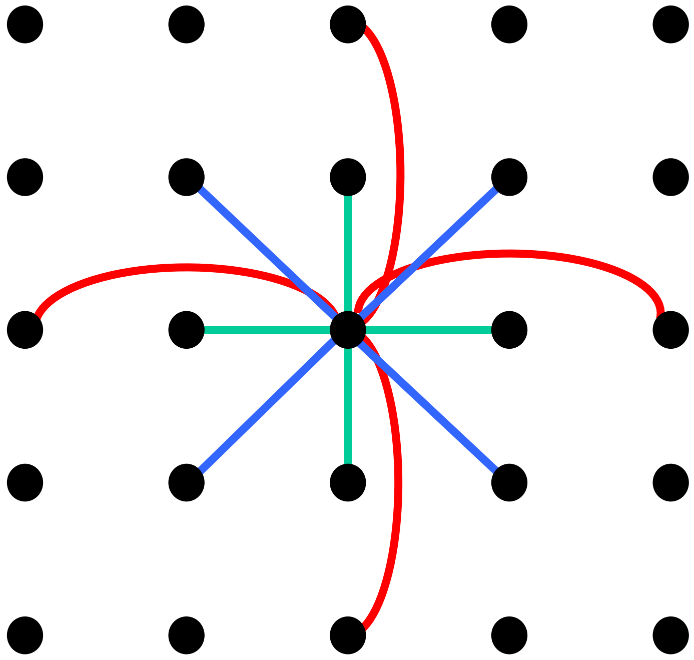
How would you simulate hair?

- Similar to string
- Also, to specify hair shape (e.g., straight or curly)
  - Add forces based on the angle between segments
  - Add additional springs/constraints stretching between the non-immediate neighbors



Cloth Modeled with Mass-Spring

- Network of masses and springs
- structural springs :
  - link (i, j) & (i+1, j) and (i, j) & (i, j+1)
- shear springs :
  - link (i, j) & (i+1, j+1) and (i+1, j) & (i, j+1)
- flexion (bend) springs :
  - link (i, j) & (i+2, j) and (i, j) & (i, j+2)
- *Be careful not to index out of bounds on the cloth edges!*



Questions?

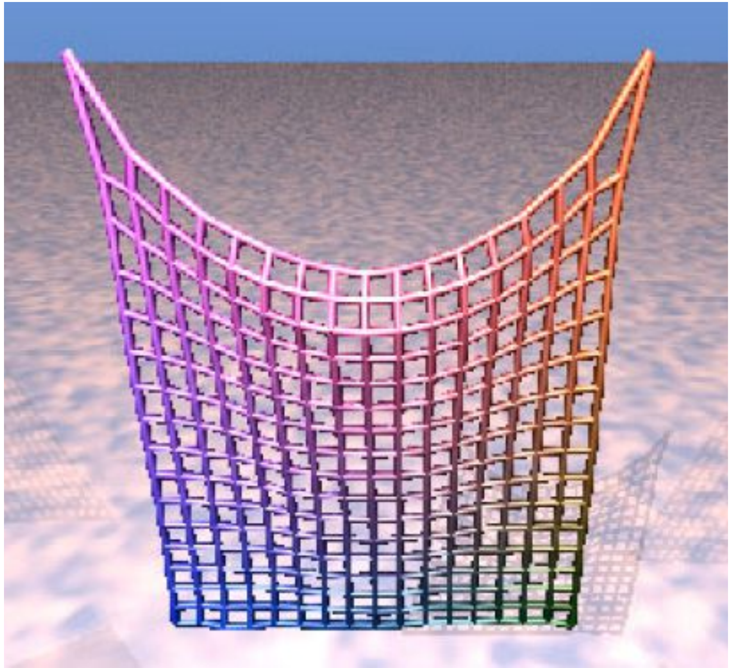
Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

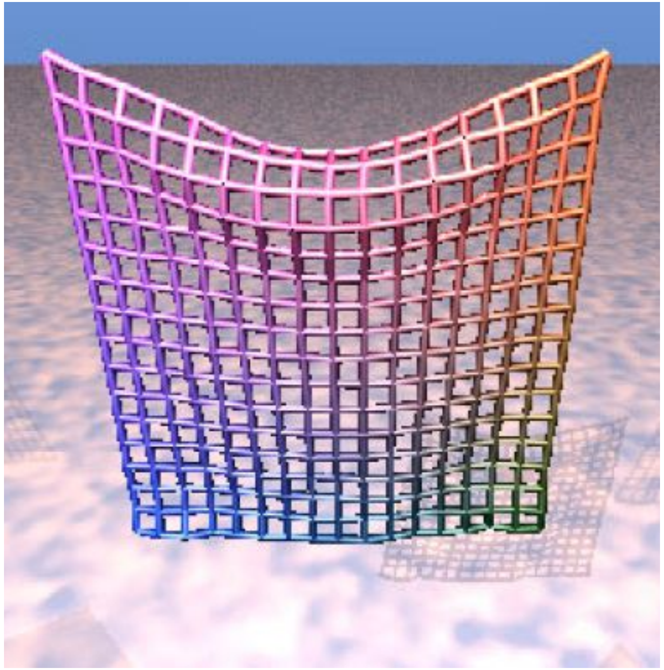
Reading for Next Time:

Everyone should read this paper!
(simple cloth model used in HW2)

- "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior", Provot, 1995.



Simple mass-spring system



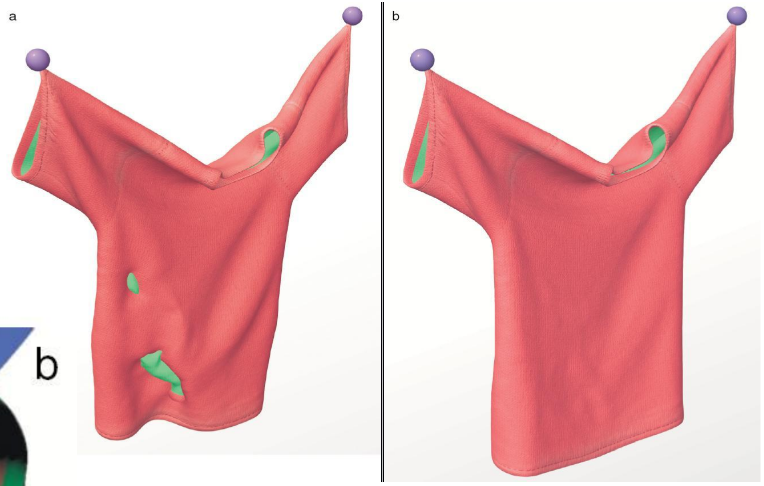
Improved solution

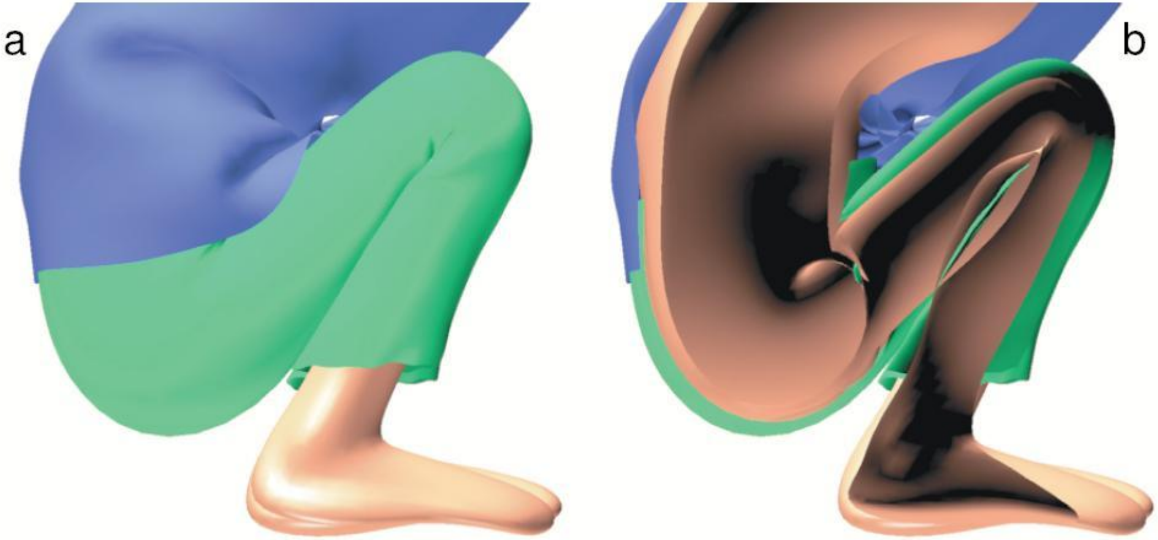
- In order to be more realistic, we increase the stiffness of the cloth?
- Adjust parameters to simulate different materials
- Implementation details are intuitive, well presented in logical order
- (mostly) Effective use of images in paper, labels could be improved
- Knit vs. woven, dry vs. wet - has impacts on drape of cloth
- Limitation/weakness: cannot handle large forces in small area of cloth
- What about collisions with objects & cloth self-collisions? & friction?
- Does order of update/adjustment matter? why/why not?
- Surprising that this simple ‘hack’ visually works well enough.
- Only simulated flat topologies of cloth - no fitted, structured clothing
- Sadly, we only notice cloth in an animation if it looks wrong.

Cloth in Practice (w/ Animation)

Optional Reading

- Baraff, Witkin & Kass, *Untangling Cloth*, SIGGRAPH 2003





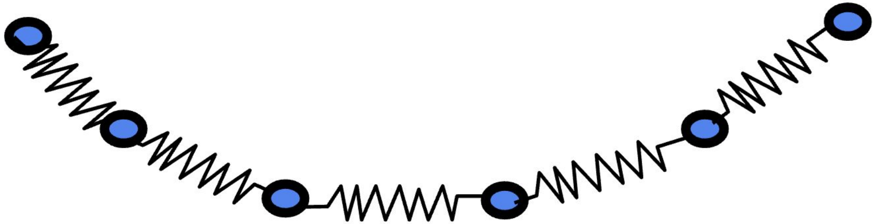
- Overall algorithm seem too simple - surprised that it works well
- Observation: Self-intersection is often hidden by other parts of the scene
- Glad paper discussed other attempts to solve the problem that did not work

Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

The Stiffness Issue

- What relative stiffness ***K*** do we want for the different springs in the network?

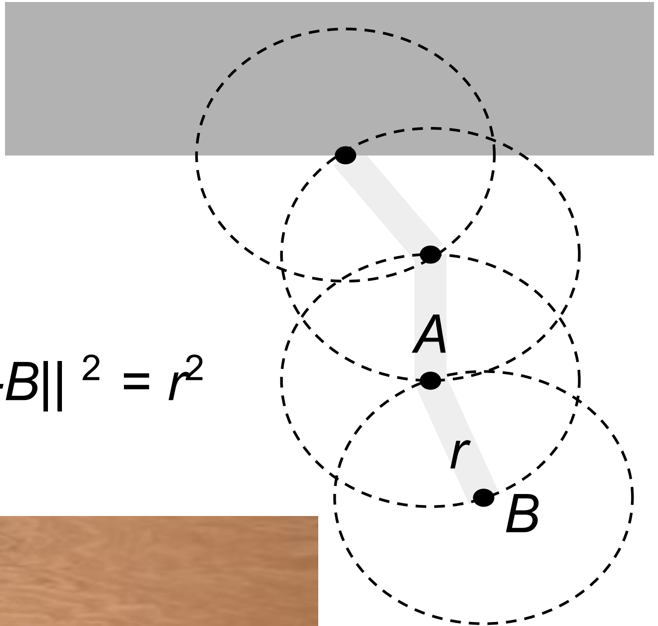


- Simple (non-spandex) cloth is barely elastic - *it shouldn't stretch much!*
- The actual spring length will always be greater than rest length
- Challenge/Unpleasant Compromise: Inverse relationship between stiffness & Δt necessary for stable simulation
    - Numerical oscillation and instability if Δt is too big
    - Simulation is costly & slow if Δt is small

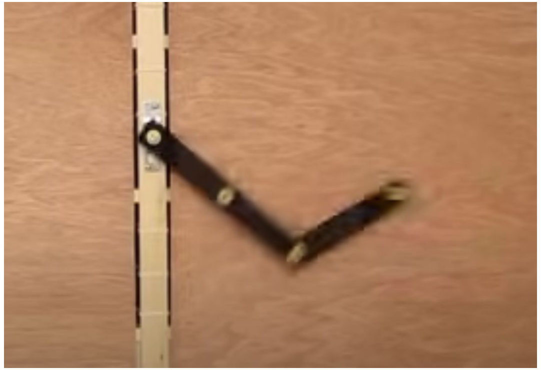
What about Rigid Constraints?

- What we really want is *no stretch or maximum stretch constraints* (not springs!)
- *E.g., rigid, fixed-length bars that link the particles*
  - Dynamics + constraints must be solved simultaneously
  - non-trivial, even for tiny systems

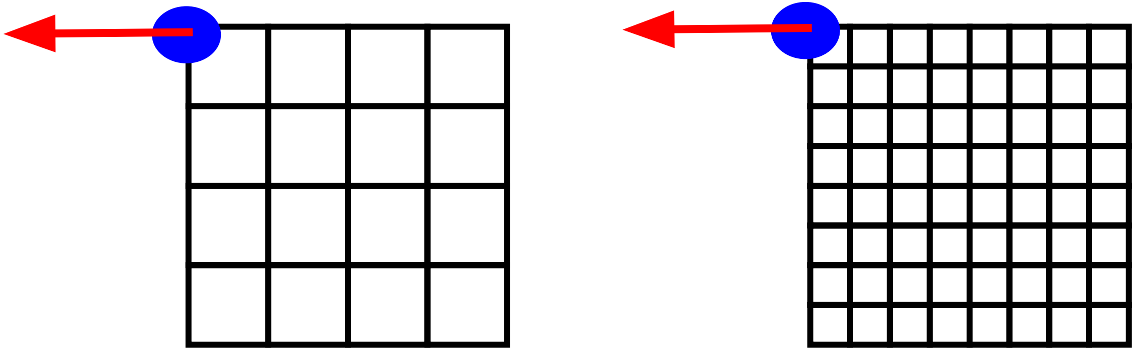
$$||A-B||^2 = r^2$$



even 2 rigid links = ***Double Pendulum*** *is chaotic!*
https://www.youtube.com/watch?v=AwT0k09w-jw



The Discretization Problem



- What is the impact of the grid resolution?
- What happens if we double the resolution of our mesh?
- Do we get the same simulation behavior for the two meshes?
  - *Usually not! It takes a lot of effort to design a scheme that does not depend on the discretization.*
- Using (explicit) Euler simulation, how many timesteps before a force propagates across the mesh in the two meshes?
  - *It will take twice as many timesteps in the higher resolution mesh!*

A Better Solution: Explicit vs. Implicit Integration

- Explicit/forward integration :

$$\mathbf{y}_{k+1} = \mathbf{y}_k + h\,\mathbf{g}(y_k)$$

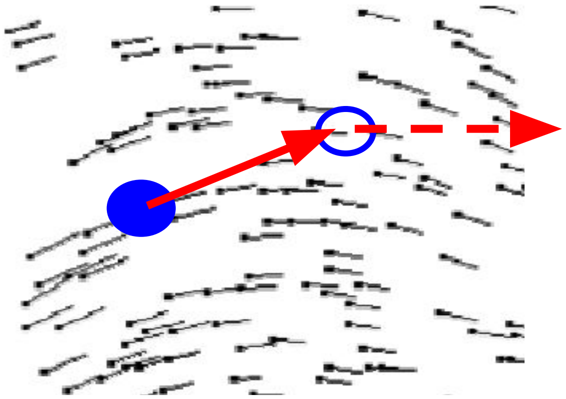
The future state (position & velocity) of this particle is a function of the current state of the particle.

- Implicit/backwards integration :

$$\mathbf{y}_{k+1} = \mathbf{y}_k + h\,\mathbf{g}(y_{k+1})$$

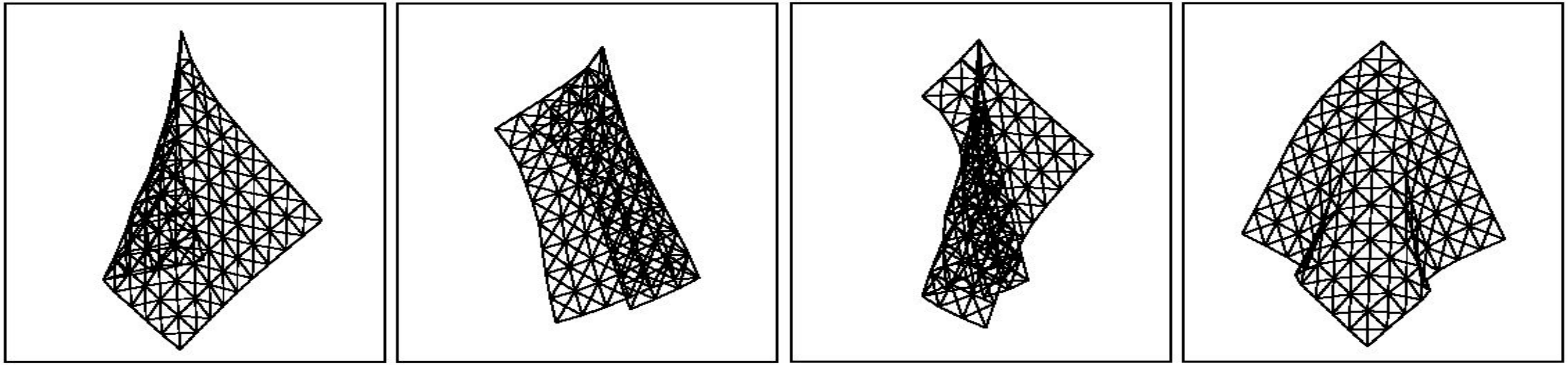
$$\mathbf{y}_{k+1} - h\,\mathbf{g}(y_{k+1}) = \mathbf{y}_k$$

The future state of this particle depends on the current state AND the future state.



  - Because particles are interconnected, must solve global system (not just local)
  - Solving each each step is more expensive (Newton’s Method, Conjugate Gradients, …)
  - *Larger timesteps are possible with implicit methods!*
  - Thus it can be overall faster than the equivalent explicit simulation

Questions?



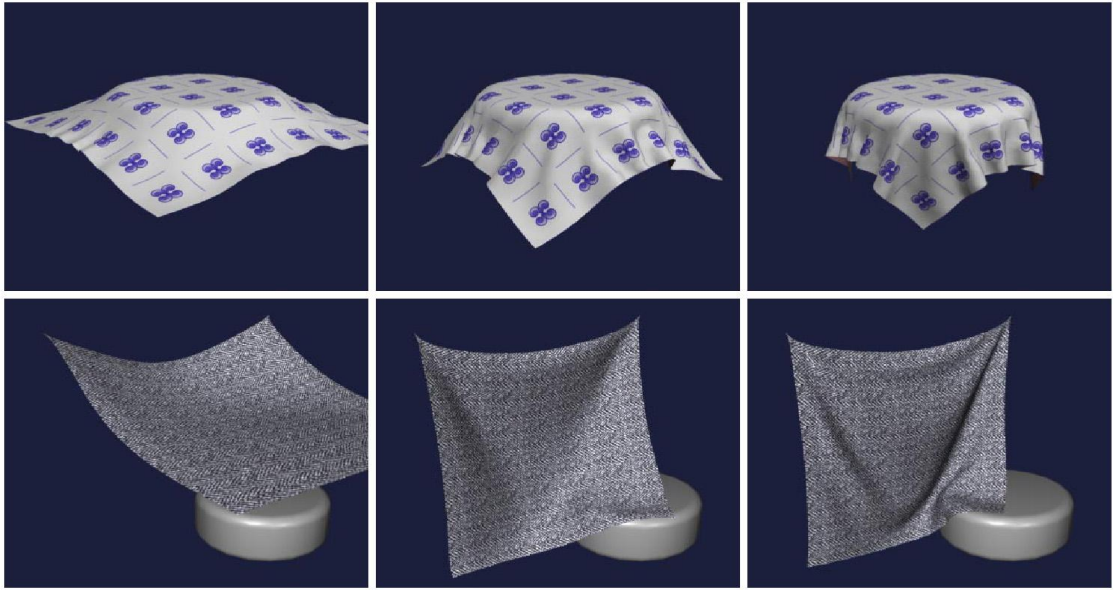
Interactive Animation of
Structured Deformable Objects
Desbrun, Schröder, & Barr 1999

Today

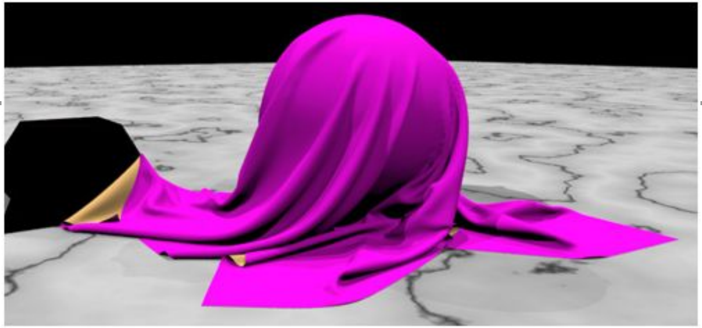
- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

Papers for Tuesday *(pick one)*

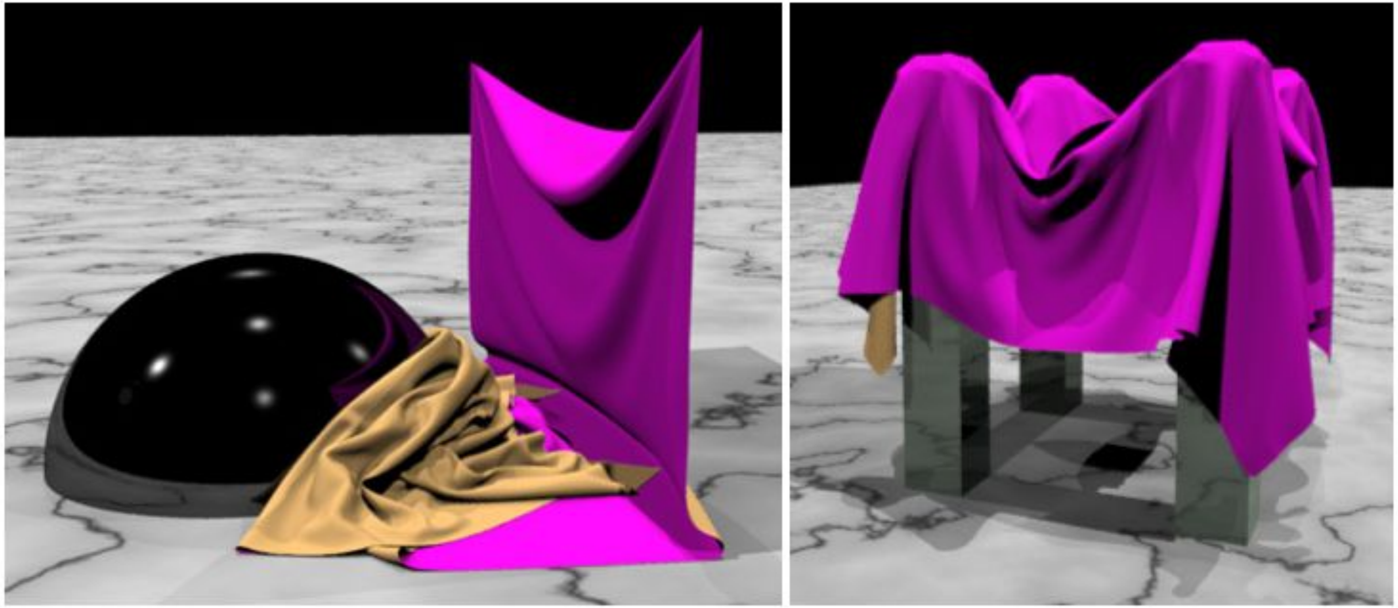
- “Large Steps in Cloth Simulation”,
  Baraff & Witkin,
  SIGGRAPH 1998



Papers for Tuesday *(pick one)*



- "Robust Treatment of Collisions, Contact and Friction for Cloth Animation", Bridson, Fedkiw & Anderson, SIGGRAPH 2002



Papers for Tuesday *(pick one)*

“Artistic Simulation of Curly Hair”, Iben, Meyer, Petrovic, Soares, Anderson, and Witkin, Symposium on Computer Animation 2013

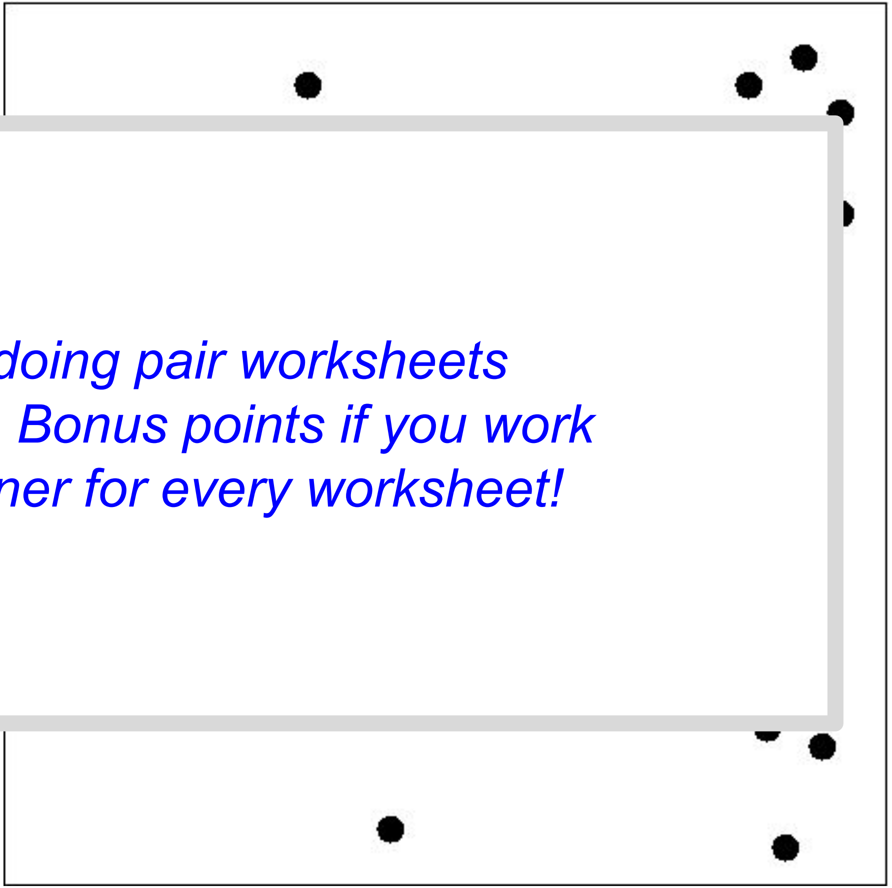
Today

- Particle Systems
  - Equations of Motion (Physics)
  - Forces: Gravity, Spatial, Damping
  - Numerical Integration (Euler, Midpoint, etc.)
- Mass Spring System Examples
  - String, Hair, Cloth
- Papers for Today
- Stiffness & Discretization
- Papers for Tuesday
- Worksheet on Volumetric Structures (if time allows)

Worksheet: Spatial Data Structures

For each adaptive grid method
(quad
partitio
if we s
and all
of 5 (m

Wher
minim
and n
from each point to the split



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