### Last Week?
- Spline Curves & Surfaces
- Subdivision Surfaces
  - Catmull Clark, Loop
  - Creases
  - Texture
  - Interpolation vs. Approximation
- Surface Reconstruction from Points

### Today
- **Particle Systems**
  - Equations of Motion (Physics)
  - Numerical Integration (Euler, Midpoint, etc.)
  - Forces: Gravity, Spatial, Damping
- Mass Spring System Examples
  - String, Hair, Cloth
- Stiffness
- Discretization

### Types of Dynamics
- **Point**
- **Rigid body**
- **Deformable body**
  (include clothes, fluids, smoke, etc.)

### 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.

### Particle Motion
- mass $m$, position $x$, velocity $v$
- equations of motion:
  \[
  \frac{dx}{dt} = v(t) \\
  \frac{dv}{dt} = \frac{1}{m} F(x, v, t)
  \]
- 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*
Path Through a Field

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

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 $v(t)$:
  \[
  \frac{d}{dt} X(t) = v(t) \\
  \frac{d}{dt} v(t) = \frac{1}{m} F(x,v,t)
  \]
- $X = \begin{pmatrix} x \\ v \end{pmatrix}$, $f(X,t) = \begin{pmatrix} v \\ \frac{1}{m} F(x,v,t) \end{pmatrix}$

For a Collection of 3D particles...

\[
X = \begin{pmatrix} p^{(1)}_x \\ p^{(1)}_y \\ p^{(1)}_z \\ p^{(2)}_x \\ p^{(2)}_y \\ p^{(2)}_z \\ p^{(3)}_x \\ p^{(3)}_y \\ p^{(3)}_z \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}
= \begin{pmatrix} v^{(1)}_x \\ v^{(1)}_y \\ v^{(1)}_z \\ v^{(2)}_x \\ v^{(2)}_y \\ v^{(2)}_z \\ v^{(3)}_x \\ v^{(3)}_y \\ v^{(3)}_z \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}
= f(X,t) = \begin{pmatrix} \pm F^{(1)}(X,t) \\ \pm F^{(2)}(X,t) \\ \pm F^{(3)}(X,t) \end{pmatrix}

Today

- Particle Systems
  - Equations of Motion (Physics)
  - Numerical Integration (Euler, Midpoint, etc.)
  - Forces: Gravity, Spatial, Damping
- Mass Spring System Examples
  - String, Hair, Cloth
- Stiffness
- Discretization

Euler’s Method

- Examine $f(X,t)$ at (or near) current state
- Take a step of size $h$ to new value of $X$:
  \[
  t_1 = t_0 + h \\
  X_1 = X_0 + h f(X_0,t_0)
  \]
- 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
Euler’s Method: Inaccurate

- Moves along tangent & can leave curve, e.g.:
  \[ f(X, t) = \left( -\gamma \right) \]
- Exact solution is circle:
  \[ X(t) = \left( \begin{array}{c} \cos(\gamma t) \\ \sin(\gamma t) \end{array} \right) \]
- Euler’s spirals 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:
  \[ h \leq \frac{1}{k} \quad \text{OK} \]
  \[ h > \frac{1}{k} \quad \text{oscillates} \]
  \[ h > \frac{2}{k} \quad \text{explodes} \]
- If \( k \) is big, \( h \) must be small

Analysis using Taylor Series

- Expand exact solution \( X(t) \)
  \[ X(t_0 + h) = X(t_0) + h \left( \frac{dX}{dt}(t_0) \right) + h^2 \left( \frac{d^2X}{dt^2}(t_0) \right) + \frac{h^3}{6} \left( \frac{d^3X}{dt^3}(t_0) \right) + \cdots \]
- Euler’s method:
  \[ X(t_0 + h) = X_0 + h f(X_0, t_0) \quad \text{\( O(h^2) \) error} \]
  
  \[ h \rightarrow h/2 \Rightarrow \text{error} / 4 \text{ per step} \times \text{twice as many steps} \]

  \[ \Rightarrow \text{error} / 2 \]

- 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

Comparison: Euler, Midpoint, Runge-Kutta

- 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

2nd-Order Methods

- Midpoint:
  - \( \frac{1}{2} \) Euler step
  - evaluate \( f_a \)
  - full step using \( f_a \)
- Trapezoid:
  - Euler step (a)
  - evaluate \( f_b \)
  - full step using \( f_a \) (b)
  - average (a) and (b)
- Not exactly same result
- Same order of accuracy

Euler will always diverge (even with small \( dt \))
Comparison: Euler, Midpoint, Runge-Kutta

- 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

Today

- Particle Systems
  - Equations of Motion (Physics)
  - Numerical Integration (Euler, Midpoint, etc.)
  - Forces: Gravity, Spatial, Damping
- Mass Spring System Examples
  - String, Hair, Cloth
- Stiffness
- Discretization

Forces: Gravity

- Simple gravity: depends only on particle mass
- N-body problem: depends on all other particles
  - Magnitude inversely proportional to square distance
  - $F_{ij} = \frac{G m_i m_j}{r^2}$

Forces: Spatial Fields

- Force on particle $i$ depends only on position of $i$
  - wind
  - attractors
  - repulsers
  - vortices
- Can depend on time
- Note: these add energy, may need damping too

Forces: Damping

$$f^{(i)} = -d\dot{y}^{(i)}$$

- Force on particle $i$ depends only on velocity of $i$
- Force opposes motion
- Removes energy, so system can settle
- Small amount of damping can stabilize solver
- Too much damping makes motion too glue-like

Questions?

Image by Baraff, Witkin, Kass
Today

- Particle Systems
  - Equations of Motion (Physics)
  - Numerical Integration (Euler, Midpoint, etc.)
  - Forces: Gravity, Spatial, Damping
- Mass Spring System Examples
  - String, Hair, Cloth
- Stiffness
- Discretization

How would you simulate a string?

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

How would you simulate hair?

- Similar to string…
- Deformation forces proportional to the angle between segments

Spring Forces

- Force in the direction of the spring and proportional to difference with rest length $L_0$
  $$F(P_i, P_j) = K \left( L_0 - ||P_i P_j|| \right) \frac{P_i P_j}{||P_i P_j||}$$
- $K$ 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

Reading for Today (2/6)

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 \((i, j)\) \&(\(i, j+2\))

From Lander

The Stiffness Issue

- What relative stiffness do we want for the different springs in the network?
- Cloth is barely elastic, shouldn’t stretch so much!
- Inverse relationship between stiffness & \(\Delta t\)
- We really want a constraints (not springs)
- Many numerical solutions
  - reduce \(\Delta t\)
  - use constraints
  - implicit integration
  - ...

The Discretization Problem

- What happens if we discretize our cloth more finely, or with a different mesh structure?
- Do we get the same behavior?
- Usually not! It takes a lot of effort to design a scheme that does not depend on the discretization.

Questions?

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

Reading for Tuesday 2/13:

- “Realistic Animation of Liquids”, Foster & Metaxas, 1996