Notes

- Please read:
  - Choi & Ko, “Stable but responsive cloth”, SIGGRAPH'02
  - Grinspun et al., “Discrete shells”, SCA'03
- Homework #4 (cloth simulation) goes out today
  - Due March 4, but you may want to look at it over the break

Bending energy

- Bending is very difficult to get a handle on without variational approach
- Bending strain energy density: $W=1/2 B \kappa^2$
  - Here $\kappa$ is mean curvature
    - Look at circles that fit surface
    - Maximum radius $R$ and minimum radius $r$
    - $\kappa=(1/R + 1/r)/2$
    - Can define directly from second derivatives of $X(p)$
    - Uh-oh - second derivatives? [FEM nastier]
- $W$ is 2nd order, stress is 3rd order, force is 4th order derivatives!

Interlude...

- Linearize and simplify drastically, look for steady-state solution ($F=0$): spline equations
  - Essentially 4th derivatives are zero
  - Solutions are (bi-)cubics
- Model (nonsteady) problem: $x_{tt}=-x_{pppp}$
  - Assume solution $x(p,t) = e^{\kappa t}x(p-ct)$
    - Wave of spatial frequency $k$, moving at speed $c$
      - [solve for wave parameters]
      - Dispersion relation: small waves move really fast
      - CFL limit (and stability): for fine grids, BAD
      - Thankfully, we rarely get that fine

Implicit/Explicit Methods

- Implicit bending is painful
- In graphics, usually unnecessary
  - Dominant forces on the grid resolution we use tend to be the 2nd order terms: stretching etc.
- But nice to go implicit to avoid time step restriction for stretching terms
- No problem: treat some terms (bending) explicitly, others (stretching) implicitly
  - $v_{n+1}=v_n+\Delta t/m(F_1(x_n,v_n)+F_2(x_{n+1},v_{n+1}))$
  - All bending is in $F_1$, half the elastic stretch in $F_1$, half the elastic stretch in $F_2$, all the damping in $F_2$
Discrete Mean Curvature

- \[ \kappa \sim \frac{\theta}{h_1 + h_2} \]
- \[ W = \sum (\frac{\theta^2}{(h_1 + h_2)} (|A_1| + |A_2|)) \]
- \[ \sim \sum \frac{\theta^2|e|^2}{|A_1| + |A_2|} \]
- Edge length, triangle areas, normals are all easy to calculate
- \( \theta \) needs inverse trig functions
- But \( \theta^2 \) behaves a lot like \( 1 - \cos(\theta/2) \) over interval \([-\pi, \pi]\)
- [draw picture]

Bending Force

- Force on \( x_i \) due to bending element involving \( i \) is then
  \[ F_i = -B \frac{\partial W}{\partial x_i} \sim -B \frac{|e|^2}{|A_1| + |A_2|} \sin(\frac{\theta}{2}) \frac{\partial \theta}{\partial x_i} \]
- Treat first terms as a constant (precompute in the rest configuration)
  \[ \sin \frac{\theta}{2} = \pm \sqrt{\frac{1}{2}(1 - n_1 \cdot n_2)} \]
- Sign should be the same as \( \sin \theta = n_1 \times n_2 \cdot \hat{e} \)
- Still need to compute \( \partial \theta / \partial x_i \)

Gradient of Theta

- Can use implicit differentiation on \( \cos(\theta) = n_1 \cdot n_2 \)
  - Not too fun
- Another approach: modal analysis
  - Gradient is orthogonal to isocontours of \( \theta \)
  - Find a basis for tangent plane to isocontours (perpendicular to grad(\( \theta \))):
  - [go through 11 modes: 6 rigid body modes, 5 planar deformations]
  - Solve for the mode that is orthogonal to all the rest
  - This gives directions, do a little diff. for mag
- [see Bridson et al., “Simulation of clothing…”, SCA’03]
  \[ u_i = \frac{\partial \theta}{\partial x_i} \]
  \[ \Delta_1 = (x_1 - x_3) \times (x_1 - x_4) \quad \Delta_2 = (x_2 - x_4) \times (x_2 - x_3) \]
  \[ u_1 = \frac{|\Delta_1|}{|A_1|^2} \quad u_2 = \frac{|\Delta_2|}{|A_2|^2} \]
  \[ u_3 = \frac{(x_1 - x_4) \cdot e \Delta_1}{|e| |A_1|^2} + \frac{(x_2 - x_4) \cdot e \Delta_2}{|e| |A_2|^2} \]
  \[ u_4 = -\frac{(x_1 - x_3) \cdot e \Delta_1}{|e| |A_1|^2} - \frac{(x_2 - x_3) \cdot e \Delta_2}{|e| |A_2|^2} \]
Modal Analysis and Soft Constraints

• Go back to $W = \frac{1}{2} k C(x)^T C(x)$
• Then elastic force is $F = -\frac{\partial W}{\partial x} = -k \frac{\partial C^T}{\partial x} C$
• Here $C=\emptyset$
• Think of $\frac{\partial C}{\partial x}$ as the mode which directly influences $C$ (steepest ascent)
  • Any other direction moves along isocontours of $C$: inefficient, interferes with other physics
• So we want force in this direction
• And the magnitude should be proportional to $-C$ to restore $C$ to 0

Damping force

• We also want damping force to be in the same mode
  • Any other direction would again interfere with other physics
• And should be proportional to the component of the velocities in this mode
  • The other components of velocity shouldn’t influence the damping
• So damping force is
  $$F = -D \frac{\partial C^T}{\partial x} \frac{\partial C}{\partial x} v = -D \frac{\partial C^T}{\partial x} \dot{C}$$
  • Note symmetric, negative semi-definite, linear in $v$

Damping bending

• Follow the same reasoning:
  $$F = -D|\dot{e}| u u^T v = -D|\dot{e}| u \dot{\theta}$$
• The $|e|$ factor is to get it to converge as the mesh is refined

Shells

• “Plates” in elasticity refer to surfaces that resist bending, and have a flat rest pose
• “Shells” are same but with a curved rest pose
• We can easily do shells by storing a nonzero rest angle
• Replace $\sin(\theta/2)$ in elastic force by $\sin(\theta/2)-\sin(\theta_0/2)$
  • Or if $\theta_0$ is far from zero, may need to go to $\theta-\theta_0$
• Note: using second neighbour springs you can’t do shells! (popping - also with inconsistent edge springs)
Other stuff in cloth mechanics

- Better model planar elasticity
  - Take into account anisotropy of weave and weft
- Creasing - allow rest angle to move when current angle too big
  - Plasticity coming up after the break…
- Subdivision for rendering
  - Geometric buckling, collision-aware
- Adaptive meshes
- Interaction with the wind
- Speed-ups

Cloth collisions

- Can use just inelastic collisions
  - Continuum takes care of rebound naturally
- Cloth colliding against non-sharp thick objects
  - Just treat as a particle system, like assignment 2
  - Can even allow particles to interpenetrate a bit
    (staying relaxed about this can let cloth look nicer)
  - For rendering need to push cloth -- the triangles as well as the vertices -- out of the objects
    - [wrinkle-preserving map]
- Cloth colliding against non-sharp thin objects
  - Need to worry about robustness -- solve for collision times, use repulsions.

Not quite…

- For large collision steps this can be bad
  - This is not just a particle system: particles are connected!
  - If a collision impulse discontinuously bounces one particle off, nearby cloth should feel it
  - Otherwise can have excessive strains (or strain rates) to deal with next time step
  - Want to naturally smooth out discontinuity in velocity field

Implicit Velocity Smoothing

- If we just evolve $v_t = F_{damp}/m$ we get smoothing
  - Damping forces seek to minimize relative velocities
  - Model problem is $v_t = v_{xx}$, the heat equation (also in multiple dimensions)
    - Solution is convolution with a Gaussian
  - Doing one implicit time step of this equation is a similar smoothing
    - Explicit time steps can only do very local smoothing…
  - We’ll use this to filter velocities after collision processing
**Collision time steps for cloth**

- Evolve \((x,v)\) to candidate new state
- Check for collisions
  - Use average velocity \(v_{\text{avg}} = (x_{\text{new}} - x) / \Delta t\) for collision formulas (only doing inelastic)
  - Resolve with a collision resolution pipeline
- Get \(x_{\text{new}}\) with collisions resolved
- Find updated average velocity \(v_{\text{avg}} = (x_{\text{new}} - x) / \Delta t\)
  - This “lives” at the midpoint: \(v(t + \Delta t/2)\)
- Use implicit velocity update to get end value:
  - Note it’s just linear:

\[
v_{n+1} = v_{n+1/2} + \frac{\Delta t}{2} M^{-1} F(x_{n+1}, v_{n+1})
\]

**Self-collision**

- Need a collision resolution pipeline like we had for rigid bodies
  - Simple, accurate, fast but local algorithms at the start
  - Robust, global algorithms at the end
- Robustness is useful: it’s not so easy to recover from a mistake without harming motion
  - Can do it though, and might need to (animators doing the impossible…)
- See Baraff et al, SIGGRAPH’03