Notes

- Homework 3 is now due Friday
  - (really Saturday when I check my email)
- Make a mesh of a sphere by projecting mesh of a cube to the sphere
- Rendering bug - malloc doesn't necessarily initialize normals to 0
- Start this class with review of the details of rigid body collisions (for the assignment)

Rigid Body Collisions

- Advance rigid bodies over a collision time step (get candidate new X, q, V)
- Process the elastic collisions first, using velocities from start of collision time step
  - Check each body against all others and ground
  - For each pair of objects
    - Apply collision impulse to deepest non-separating point
    - This impulse changes candidate new X, q, V
    - Iterate a few (3-5) times
- Repeat whole elastic collision loop if you want
- Then process inelastic contacts the same way, but using average velocities

Average velocities

- Define the average velocities the same way that you update the candidate new X, q
  - For example:
  - $X^{new}=X(t)+\Delta t V^{after}$ means use $V^{avg}=(X(t+\Delta t)-X(t)) / \Delta t$
  - $q^{new}=q(t)+\Delta t/2\ \omega^{after}$ (before normalization) means use $\omega^{avg}=2(q(t+\Delta t)-q(t)) / \Delta t$
  - Or $q^{new}=\text{rotate}(\Delta t\omega)q(t)$ means use $\omega^{avg}=\text{unrotate}(q^{new}\ q^{-1})/\Delta t$
    (see jan20 slides for what I mean by rotate)

Setting up collision

- Point from object 1 is intersecting object 2
- World space position is x
- Object space position in object 1 is p_1
- Object space position in object 2 is p_2
- Normal n comes from object 2's level set (but remember that this is the world space normal, not the object space normal: have to rotate object space normal with q_2)
- Check if relative velocity v has negative normal component: the separating condition (if separating, no collision)

Velocities

- $v_1=\omega\times(x-X_1)+V_1$ and $v_2=\omega\times(x-X_2)+V_2$
  - Note for ground, $v_2=0$
- Relative velocity $v=v_1-v_2$
- We will apply an impulse J to object 1 and -J to object 2
- $V_1^{after}=V_1+J/M_1$ and $V_2^{after}=V_2-J/M_2$
- $L_1^{after}=L_1+(x-X_1)\times J$ and $L_2^{after}=L_2-(x-X_2)\times J$
- $\omega_1^{after}=\omega_1+I_1^{-1}[(x-X_1)\times J]$ and $\omega_2^{after}=\omega_2-I_2^{-1}[(x-X_2)\times J]$

More velocities

$$\begin{aligned}
v_1^{after} &= v_1 + \frac{J}{M_1} + \left[I_1^{-1}\left[(x-X_1)\times J\right]\right]\times(x-X_1) \\
&= v_1 + \tfrac{1}{M_1}J - (x-X_1)^* I_1^{-1}(x-X_1)^* J \\
&= v_1 + \left(\tfrac{1}{M_1}\delta + (x-X_1)^{*T} I_1^{-1}(x-X_1)^*\right)J \\
&= v_1 + K_1 J \\
v_2^{after} &= v_2 - K_2 J
\end{aligned}$$

- So post-impulse relative velocity is $v^{after}=v+KJ$ where $K=K_1+K_2$
  - Take $K_2=0$ for the ground (infinite mass, infinite inertia tensor)

Finding the impulse

- Now we just need to figure out J of a particular form that gives us a particular post-impulse relative velocity
- For frictionless collision:
  - Find $J=jn$ such that $v_n^{after}=-\varepsilon v_n$
- For Coulomb friction:
  - Find static friction J (such that $v_n^{after}=-\varepsilon v_n$ and $v_T^{after}=0$)
  - Check if that J is in friction cone
  - If not, find sliding friction J instead: $J=j(n-\mu T)$ which has to be on friction cone. Solve for j so that $v_n^{after}=-\varepsilon v_n$

Frictionless case

- Use impulse in normal direction: $J=jn$
- Want $n^T v^{after}=-\varepsilon n^T v$
- Plug in v^{after} formula and our choice of J: $n^T(v+Knj)=-\varepsilon n^T v$
- Simplify: $n^T K n\, j = -(1+\varepsilon)n^T v$
- Solve:

$$j = \frac{-(1+\varepsilon)n^T v}{n^T K n}$$

Static friction

- Use general impulse J
- Want $n^T v^{after} = -\varepsilon n^T v$ (=0 for inelastic) AND $v_T^{after} = 0$ (no sliding)
- Since $v_T = v - nn^T v$, can write this as $v^{after} = -\varepsilon nn^T v$
- Plug it in: $v + KJ = -\varepsilon nn^T v$
- Solve: $$J = -K^{-1}\left(v + \varepsilon nn^T v\right) = -K^{-1}\left(v + \varepsilon(v \cdot n)n\right)$$

Static friction test

- Check if J from last slide satisfies

$$\left|J_T\right| = \left|J - \left(J \cdot n\right)n\right| \le \mu\left|J \cdot n\right|$$

- This is the friction cone test
- Remember Coulomb friction is defined by the friction force always satisfying this inequality
- If J doesn't satisfy inequality, throw it away: look for a sliding friction impulse **on** the friction cone $$\left|J_T\right| = \mu\left|J \cdot n\right|$$

Sliding friction

- Find tangential relative velocity: $v_T = v - (v \cdot n)n$
- Normalize to get sliding direction: $T = v_T / |v_T|$
  - If $v_T = 0$ then no friction force…
- Take $J = j(n - \mu T)$ (has to be on friction cone)
- Want $n^T v^{after} = -\varepsilon n^T v$
- Plug in: $n^T(v + K(n - \mu T)j) = -\varepsilon n^T v$
- Solve: $$j = -\frac{(1+\varepsilon)n^T v}{n^T K(n - \mu T)}$$

Other elements for FEM

- Not so obvious ones:
  - Isoparametric elements (meshes with curved edges)
  - Radial-basis functions (mesh-free methods)
  - Mixed element meshes (triangles and quads together)
  - Embedded elements
  - Special-purpose elements (e.g. for cracks)

Elastic Surfaces

- We've covered basic 2D elasticity
  - Actually, 3D isn't much different
- This class: stick with 2D objects, but embed in 3D
  - E.g. cloth
- Somewhat more complicated
  - Object space is 2D, world space is 3D
  - Deformation gradient A is 3x2, not square
  - Green strain G is 2x2, but we want 3x3 stress!
- (springs often work fine still)

First steps

- We want to rotate surface element into xy plane, forget (constant) z coordinate
- Do the usual 2D stuff in xy plane
- Rotate tractions back
- This is fairly messy, but the way to go for completely general constitutive model
- But, do we need more physics?
  - [line/arc]

Hyper-elasticity

- Want a framework that can handle all this stuff easily
- Instead define an elastic potential energy
  - Strain energy density W=W(A)
  - W=0 for no deformation, W>0 for deformation
  - Total potential energy is integral of W over object
- This is called hyper-elasticity or Green elasticity
- For most (the ones that make sense) stress-strain relationships can define W
  - E.g. linear relationship: $W=\sigma:\varepsilon$

Variational Derivatives

- Force is the negative gradient of potential
  - Just like gravity
- What does this mean for a continuum?
  - $W=W(\partial X/\partial p)$, how do you do $-d/dX$?
- Variational derivative: $W_{total}\left[X+\varepsilon Y\right]=\int W\left(\frac{\partial X}{\partial p}+\varepsilon\frac{\partial Y}{\partial p}\right)$
  - So variational derivative is $-\nabla\cdot\partial W/\partial A$
  - And $f=\nabla\cdot\partial W/\partial A$
  - Then stress is $\partial W/\partial A$

$$\approx\int W\left(\frac{\partial X}{\partial p}\right)+\varepsilon\frac{\partial W}{\partial A}\frac{\partial Y}{\partial p}$$

$$=W_{total}+\varepsilon\int\frac{\partial W}{\partial A}\frac{\partial Y}{\partial p}$$

$$=W_{total}-\varepsilon\int Y\nabla\cdot\frac{\partial W}{\partial A}$$

Numerics

- Simpler approach: find discrete W_{total} as a sum of W's for each element
  - Evaluate just like FEM, or any way you want
- Take gradient w.r.t. positions $\{x_i\}$
  - Ends up being a Galerkin method
- We've actually done this before: soft constraints
  - Total energy was $1/2\ C^TC$
  - And we know how to do Rayleigh damping for this
  - See Jan 27 lecture
  - Here each element's W(A) corresponds to an entry in C

Curve / Springs

- Take $W(A)=1/2\ E(|A|-1)^2\ L$ for each segment
  - Note factor of L: this is approximation to an integral over segment in object space of length L
- $A=(x_{i+1}-x_i)/L$ is the deformation gradient for piecewise linear elements
- Then take derivative w.r.t. x_i to get this element's contribution to force on i
- Lo and behold [exercise] get exactly the original spring force from first week

Surface elasticity

- For linear stress-strain, can use $W(A)=\sigma:G=\sigma_{ij}G_{ij}$
- The simplest model from before gives $W=\lambda G_{kk}^2 + \mu G_{ij}G_{ij}$
- Remember $G=1/2(A^TA-I)$
- Tedious to differentiate, but doable
  - Tensors and chain rule over and over
- Let's leave it that
  - In practice, springs with speed-of-sound heuristic are good enough most of the time

Cloth modeling

- Cloth behaves in a fairly nonlinear way
- In extension, biphasic
  - For small stretching, only weak resistance: the threads are simply straightening out
  - For large stretching, strong resistance: the threads are being pulled apart
- If we model with springs, need to introduce nonlinearity
- Simplest approach to getting strong resistance: inequality constraints (springs may not stretch more than, say, 10%)

Strain limiting

- Solving inequality constraints is difficult
- We're happy with approximation
- Loop through mesh:
  - Whenever a spring has strain beyond some limit, apply impulse to return it to legal strain
  - (constraint impulse as before: find impulse parallel to spring that causes updated positions to be exactly the right distance apart)
- Iterate if you want, just like collisions etc. (but usually once per time step is enough)

Qualitative behaviour

- Strain limiting with weak spring constants means small wrinkles, creases, etc. can form easily
  - Stiff materials can't easily wrinkle in non-metric-preserving ways
  - Stiff springs or FEM induce additional unwanted numerical stiffness resisting bending
- But large sagging, rubbery stretching, etc. are eliminated

Compression and buckling

- Cloth also behaves oddly under compression
  - Almost never compresses, like 2D materials in 2D or 3D materials in 3D
  - Instead **buckles** out of plane
- Two (good) ways to go:
  - Assume mesh can't resolve buckling, but let it happen anyways (subgrid modeling)
    - Good for coarse meshes
  - Force mesh to resolve buckling
    - Good for fine meshes

Subgrid modeling

- From Choi & Ko (SIGGRAPH'02)
- Make the springs much weaker in compression
  - Can actually derive formula based on model of a buckled beam
  - Simpler approach: k=0 or much smaller when compressed
- [draw model]

Enforced buckling

- Require that the mesh resolves the out-of-plane buckling:
  - Do not allow springs to compress
- Another inequality constraint
- Again, take simple route:
- Loop over springs
  - If spring in compression (strain < -0.001), apply corrective impulse to get it back to rest length
- Can repeat if wanted
- Naturally goes together with strain limiting

Simple bending

- Can fake bending resistance by adding extra springs between second neighbours
- When mesh bends, these extra springs compress and push it back to planar
- Not so obvious what to do for unstructured meshes
  - Or how to scale the bending springs

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 κ 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?