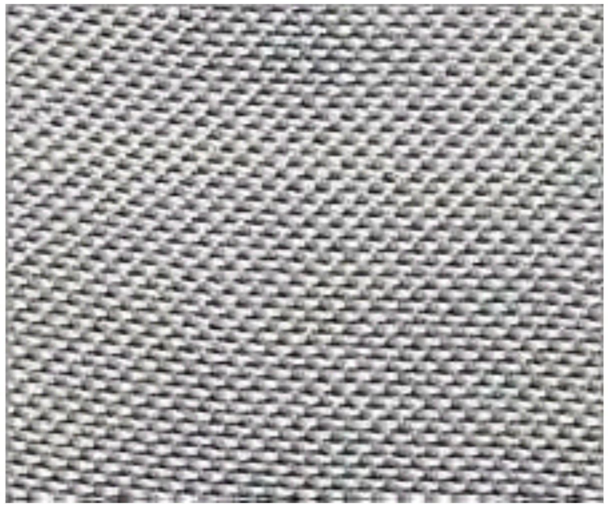
Cloth Simulation

What make cloth hard to simulate?

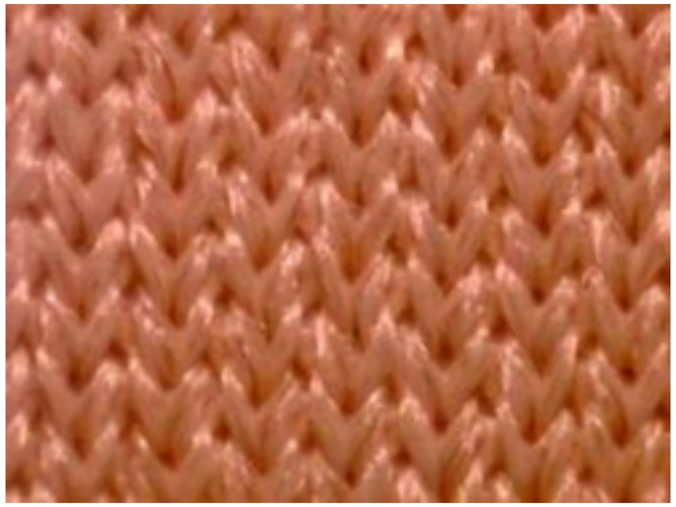
- Due to the thin and flexible nature of cloth, it produces detailed folds and wrinkles, which in turn can lead to complicated self-collisions.
- Cloth is characterized by strong resistance to stretch and weak resistance to bending, which leads to a stiff set of equations and thus prohibits the use of large time steps.

Woven versus knit cloth

- The yarns in woven fabric are nearly immobile with very limited deformations in the yarn structure.
- Knit materials consists of interlocked loops which deform and slide readily with dramatic changes in small-scale structures.
- Most research on cloth mechanics has focused on woven cloth.



woven fabric



knit fabric

Woven fabric

Knit fabric

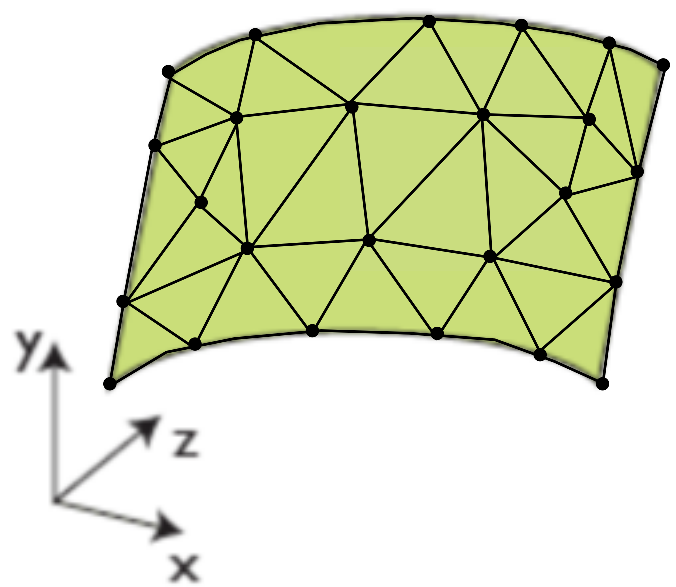
Mass-spring versus FEM

- Mass-spring system is easy to implementation and cheap to compute, but is not as accurate.
- FEM spatially discretizes a set of PDEs which govern the deformation in continuum mechanics.
- FEM provides a more accurate way to measure complex material behaviors beyond a simple elastic model.

Simulating different materials

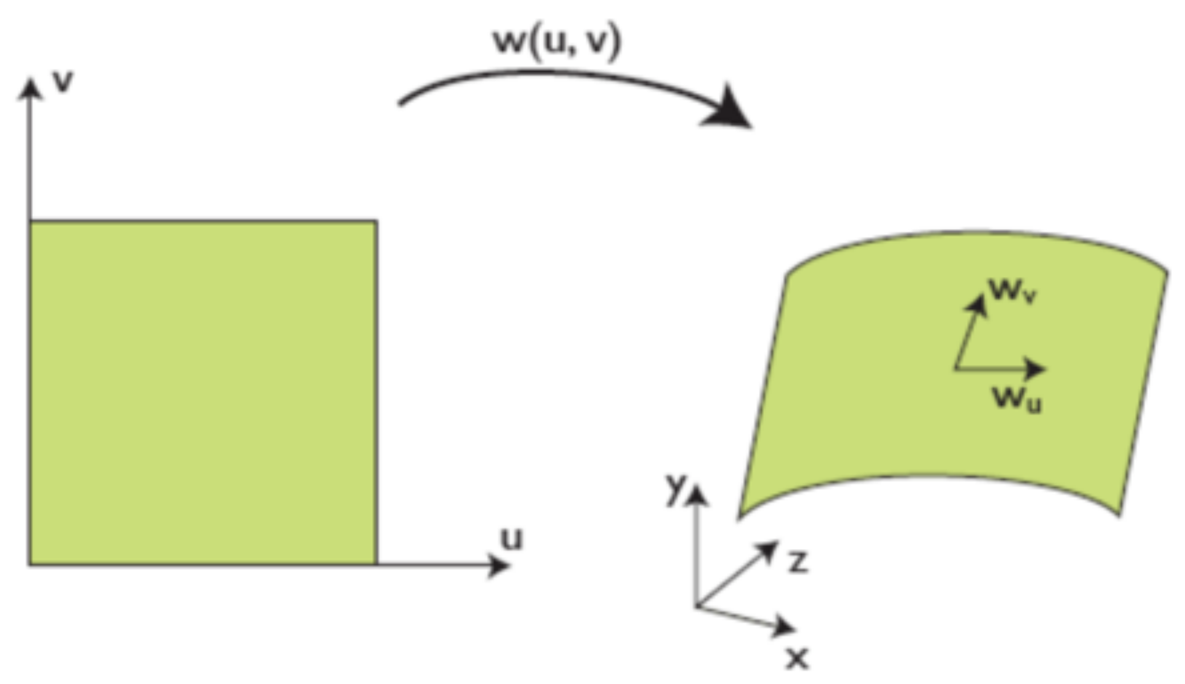
World space

- Model clothes as triangular mesh of particles in $\mathbf{R}^3$
- The geometry state of clothes is $\mathbf{x} \in \mathbf{R}^{3n}$



Material space

- Treat cloth as a 2-dimensional manifold embedded in $\mathbf{R}^3$.
- Capture the rest state of cloth by assigning each particle an unchanging coordinate (u , v) in the material space.
- The mapping between the material space and the world space is defined by $\mathbf{w}$(u, v).



Equations of motion

- Equation of motion: $\mathbf{M}\ddot{\mathbf{x}} = \mathbf{f}_{int} + \mathbf{f}_{ext}$

$\mathbf{M}$	mass matrix , $\mathbf{R}^{3nX3n}$
$\ddot{\mathbf{x}}$	acceleration of particles, $\mathbf{R}^{3n}$
$\mathbf{f}_{ext}$	gravity and contact force, $\mathbf{R}^{3n}$
$\mathbf{f}_{int}$	cloth internal forces, $\mathbf{R}^{3n}$

- Internal forces are derived from potential energy function $E(\mathbf{x})$.

Potential energy

- The negative gradient of each potential function defines a type of internal force:

$$-\frac{\partial E}{\partial \mathbf{x}}$$

- General form of E(x):

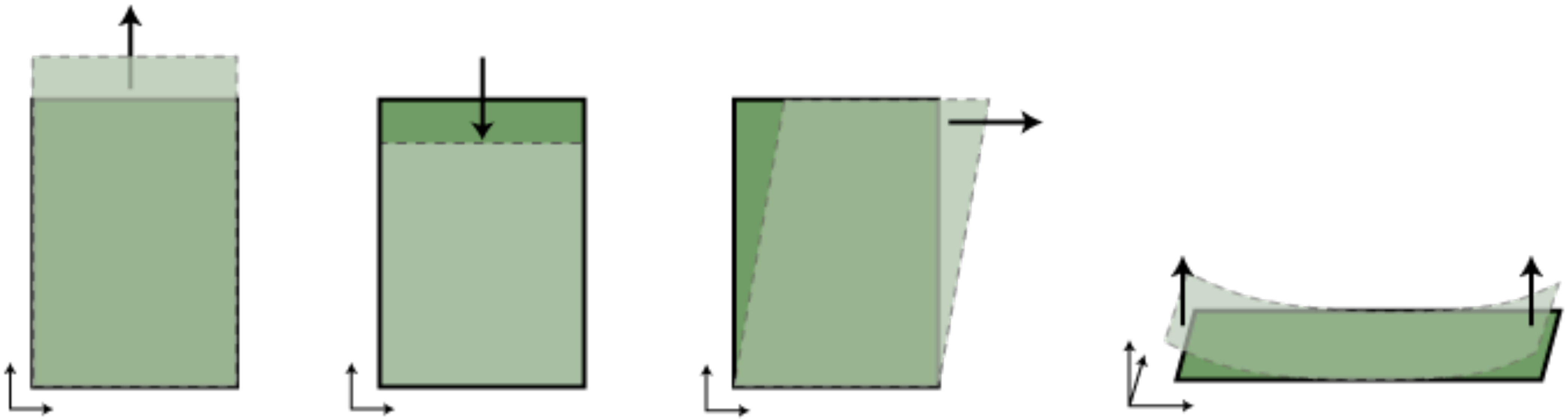
$$E(\mathbf{x}) = \frac{k}{2}\mathbf{C}(\mathbf{x})^T\mathbf{C}(\mathbf{x})$$

- The internal force can be computed by

$$\mathbf{f} = -\frac{\partial E}{\partial \mathbf{x}}^T = -k\frac{\partial \mathbf{C}(\mathbf{x})}{\partial \mathbf{x}}^T\mathbf{C}(\mathbf{x})$$

Internal forces

- In general, clothes resists motion in four directions



stretch compress shear

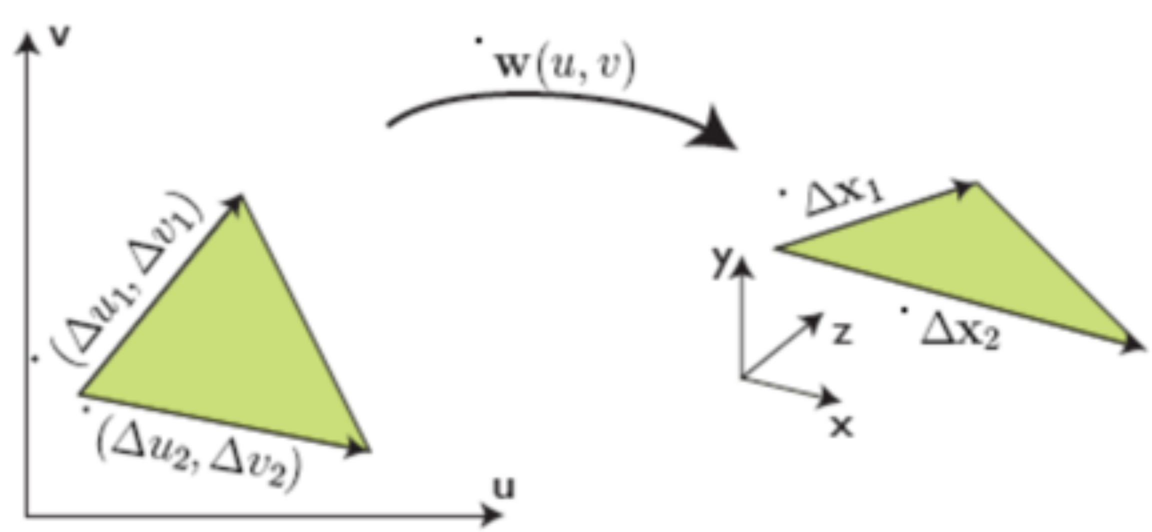
in-plane motion

bend

out-of-plane motion

Stretch force

- Stretch/compress force can be measured by deformation gradient.
- Assuming **w** is a linear function over each triangle, the gradient of **w** is constant within each triangle.
- Define energy function using

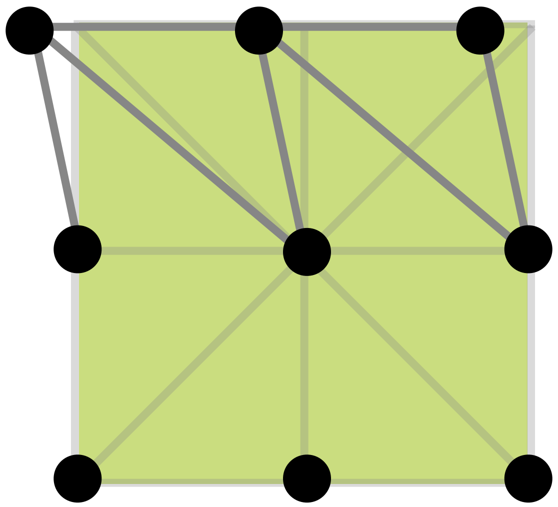


$$(\mathbf{w}_u \quad \mathbf{w}_v) = (\Delta\mathbf{x}_1 \quad \Delta\mathbf{x}_2)\begin{pmatrix} \Delta u_1 & \Delta u_2 \\ \Delta v_1 & \Delta v_2 \end{pmatrix}^{-1}$$

$$\mathbf{C}(\mathbf{x}) = a\begin{pmatrix} \|\mathbf{w}_u(\mathbf{x})\| - b_u \\ \|\mathbf{w}_v(\mathbf{x})\| - b_v \end{pmatrix}$$

Shear force

- Shear force can be measured by $\mathbf{w}_u^T \mathbf{w}_v$
- When the $\mathbf{w}_u$ and $\mathbf{w}_v$ are orthogonal, shear force is zero.
- Define energy function for shear force using $C(\mathbf{x}) = a\mathbf{w}_u(\mathbf{x})^T \mathbf{w}_v(\mathbf{x})$



Bend force

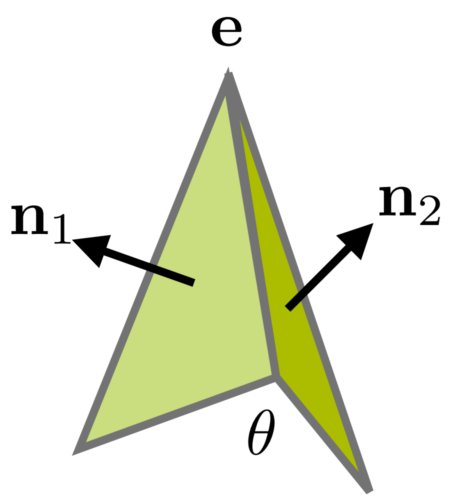
- Bend force is measured between a pair of adjacent triangles.
- Using two geometry relations, we can solve for the bending angle.

$$\sin\theta = (\mathbf{n}_1 \times \mathbf{n}_2) \cdot \mathbf{e}$$

$$\cos\theta = \mathbf{n}_1 \cdot \mathbf{n}_2$$

- Define energy function using

$$C(\mathbf{x}) = \theta$$



Damping force

- Damping force turns out to be important both for realism and numerical stability.
- Damping forces should
  - act in direction of corresponding elastic force, and
  - be proportional to the velocity in that direction.

$$\mathbf{d} = -k_d \dot{\mathbf{C}}(\mathbf{x}) \frac{\partial \mathbf{C}(\mathbf{x})}{\partial \mathbf{x}}$$

Buckling effect

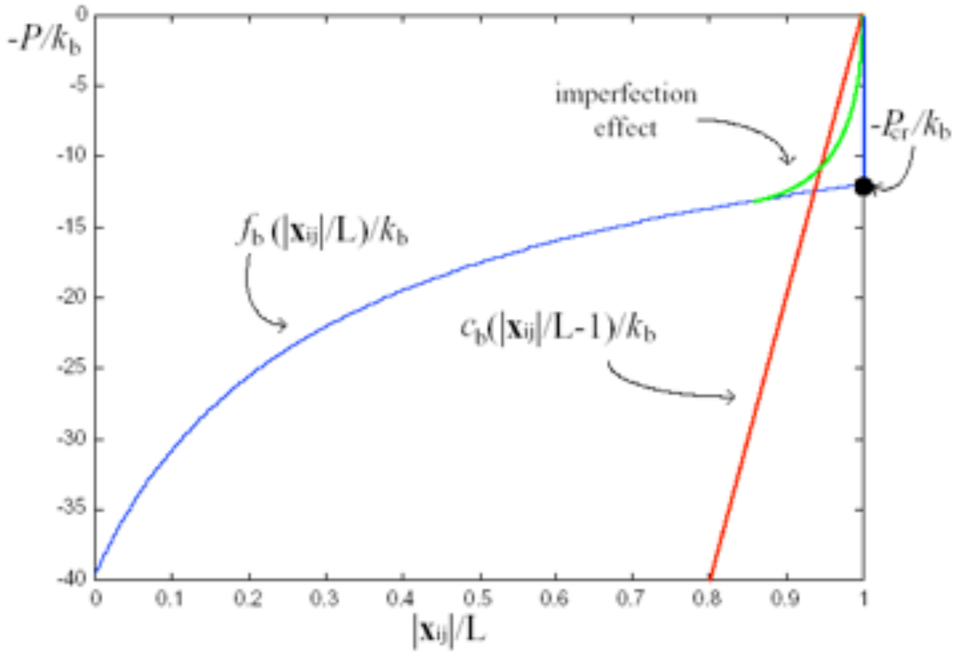
When we push cloth like this,

we expect to see this.

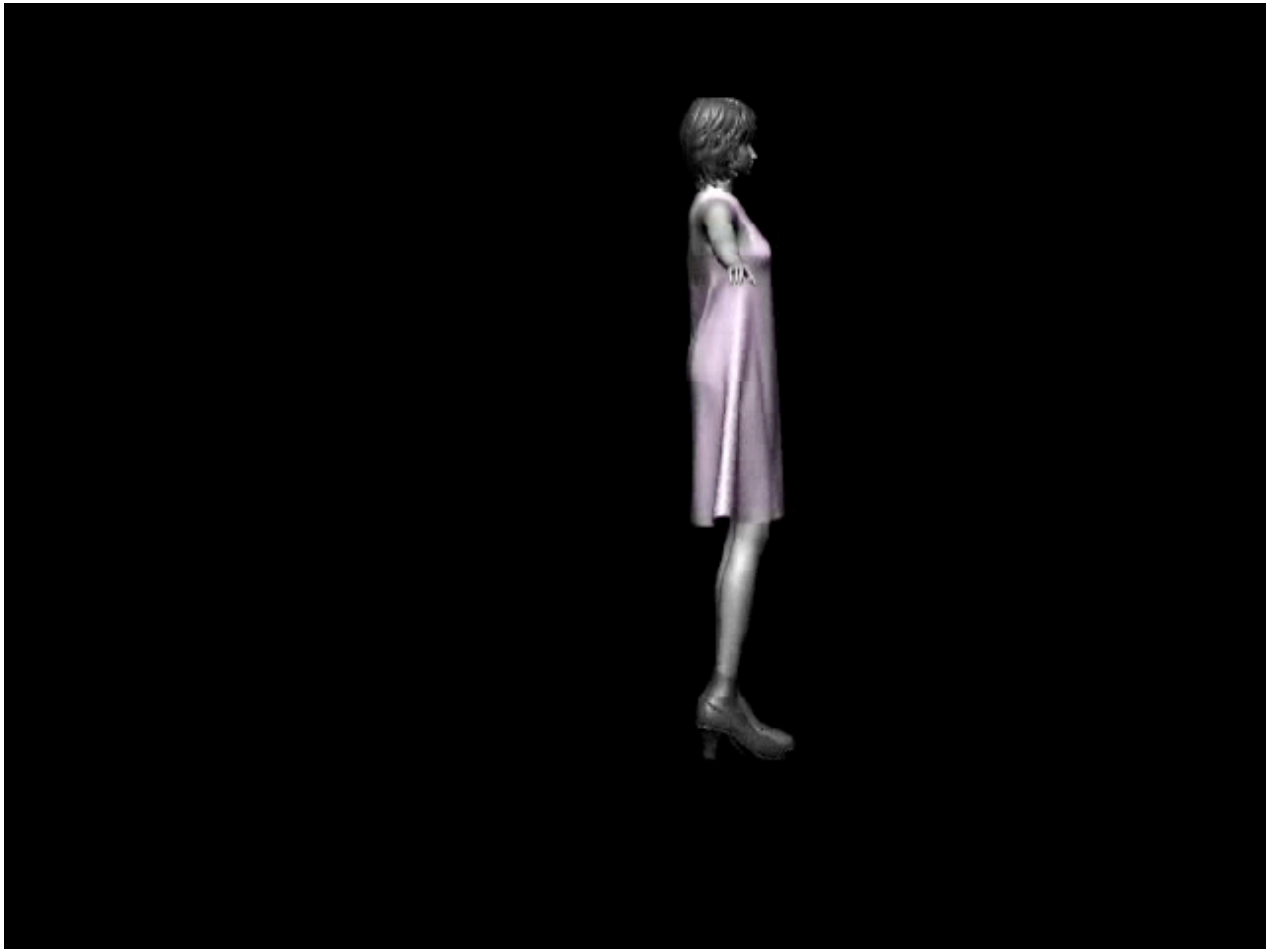
But in the mass-spring system, the compression force has to be very stiff to get out-of-plane motion.

Buckling model

- Replace bend and compression forces with a single nonlinear model.
- Details in Ko and Choi SIGGRAPH 2002.



Stable but responsive cloth



Integration issues

- In general, cloth stretches little if at all in the plane.
- To counter this, we generally have large in-plane stretch forces, which requires a high stiffness coefficient.
- Since explicit integrators suffer from this stiff system, an implicit integrator is used to achieve larger time steps.

Implicit integration

- Transfer a second-order ODE to the first-order form.

$$\frac{d}{dt}\begin{pmatrix}\mathbf{x}\\ \mathbf{v}\end{pmatrix} = \begin{pmatrix}\mathbf{v}\\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x},\mathbf{v})\end{pmatrix}$$

- Compute the next state based on the derivative evaluated at the *next* state.

$$\begin{pmatrix}\mathbf{x}\\ \mathbf{v}\end{pmatrix} = \begin{pmatrix}\mathbf{x}_0\\ \mathbf{v}_0\end{pmatrix} + h\begin{pmatrix}\boxed{\begin{matrix}\mathbf{v}\\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x},\mathbf{v})\end{matrix}}\end{pmatrix}$$

derivative at
next state

Linearize derivative function

- Linearize derivative function about the current state.

$$\begin{pmatrix} \mathbf{v} \\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x},\mathbf{v}) \end{pmatrix} = \begin{pmatrix} \mathbf{v}_0 \\ \mathbf{M}^{-1}\mathbf{f}_0 \end{pmatrix} + \frac{\partial \begin{pmatrix} \mathbf{v} \\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x},\mathbf{v}) \end{pmatrix}}{\partial \begin{pmatrix} \mathbf{x} \\ \mathbf{v} \end{pmatrix}} \begin{pmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{v} \end{pmatrix}$$

- Compute next state by implicit integration

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{v} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_0 \\ \mathbf{v}_0 \end{pmatrix} + h \begin{pmatrix} \mathbf{v}_0 + \Delta\mathbf{v} \\ \mathbf{M}^{-1}(\mathbf{f}_0 + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Delta\mathbf{x} + \frac{\partial \mathbf{f}}{\partial \mathbf{v}}\Delta\mathbf{v} \end{pmatrix}$$

- Solve for linear system

$$\left(\mathbf{I} - h\mathbf{M}^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{v}} - h^2\mathbf{M}^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right)\Delta\mathbf{v} = h\mathbf{M}^{-1}\left(\mathbf{f}_0 + h\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{v}_0\right)$$

- Update state

$$\mathbf{x} = \mathbf{x}_0 + h(\mathbf{v}_0 + \Delta\mathbf{v})$$

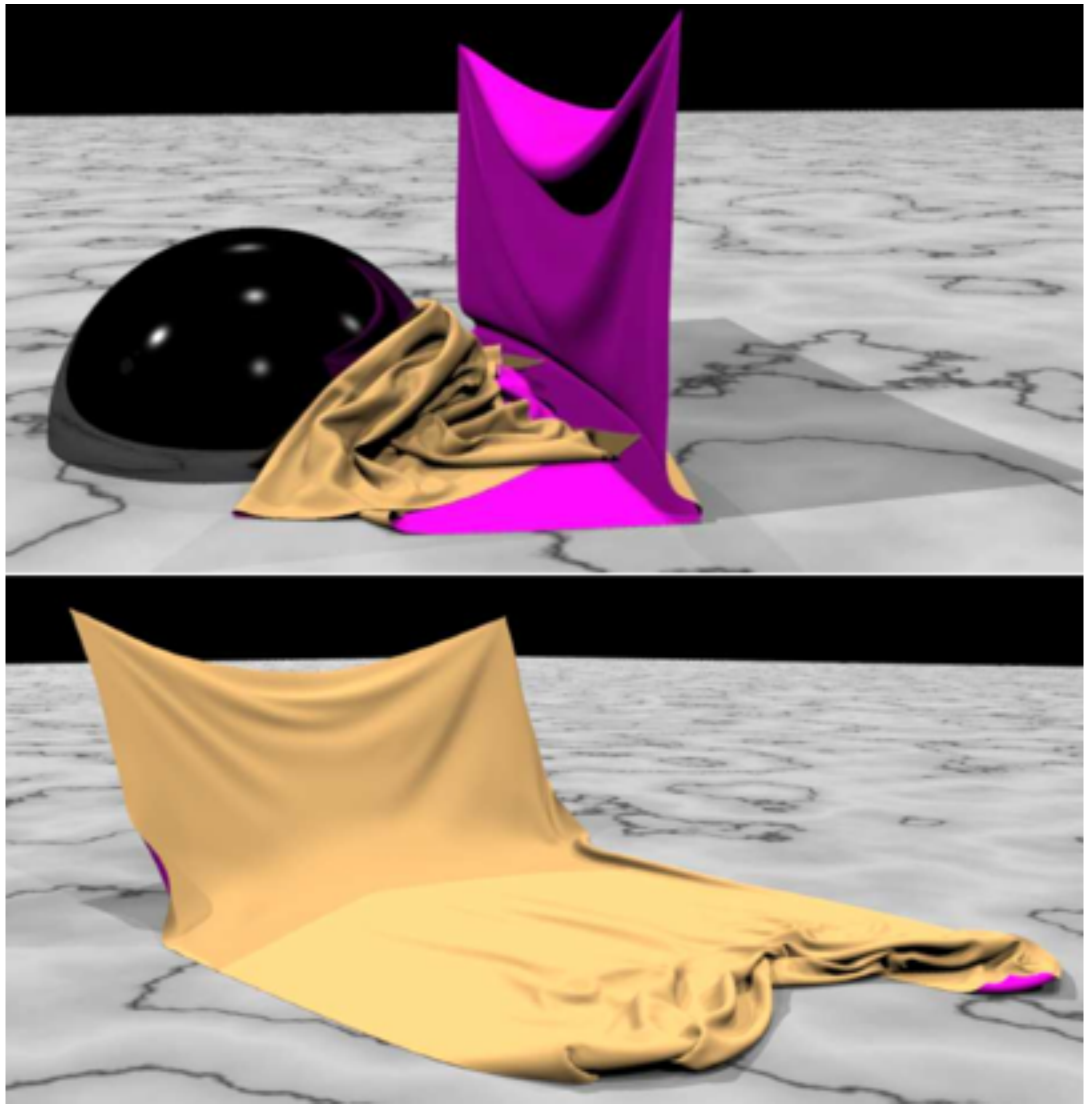
$$\mathbf{v} = \mathbf{v}_0 + \Delta\mathbf{v}$$

Collision and constraints

- Collision is the bottleneck of simulation due to a large number of collision points.
- Interpenetrating is very obvious and difficult to correct after the fact.
- Use repulsion forces and and impulses to handle collision.

Practical collision handling

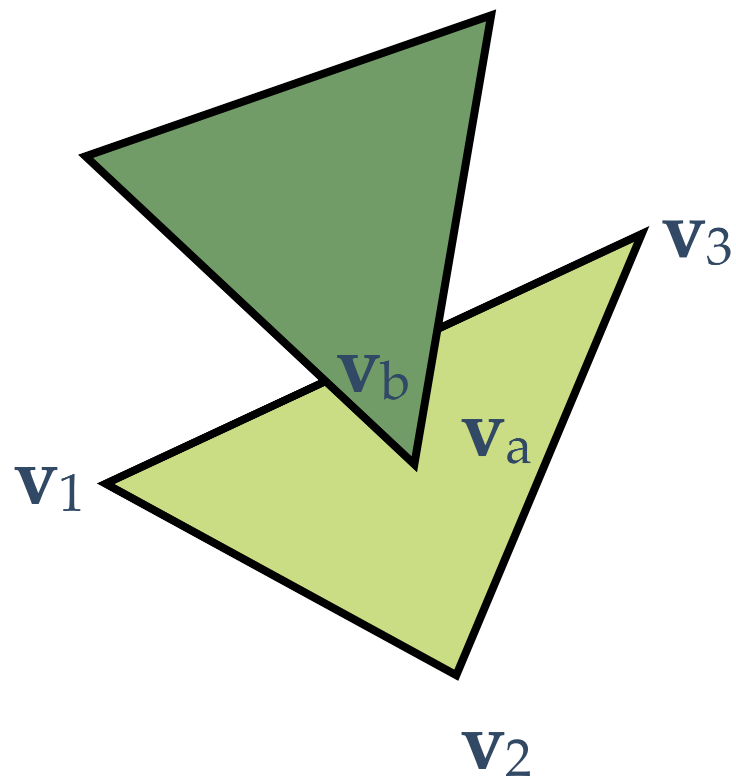
- Bridson et al proposed an efficient and robust method to handle collisions.
- Use repulsion forces to deal with this vast majority of collisions in a simple and efficient manner.
- Use a more expensive but completely robust method to stop the few that remain.



Hybrid collision handling

- Beginning of time step: $\mathbf{x}^0,\ \mathbf{v}^0$
- Integrate cloth dynamics to advance to intermediate state $\bar{\mathbf{x}}^1,\ \bar{\mathbf{v}}^1$
- Compute average velocity: $\bar{\mathbf{v}}^{1/2} = (\bar{\mathbf{x}}^1 - \mathbf{x}^n)/h$
- Apply repulsion and friction to average velocity to get $\tilde{\mathbf{v}}^{1/2}$
- Resolve collision by modifying $\tilde{\mathbf{v}}^{1/2}$ to the final midstep velocity $\mathbf{v}^{1/2}$
- Update the final position: $\mathbf{x}^1 = \mathbf{x}^0 + h\mathbf{v}^{1/2}$
- Update velocity: if collision $\mathbf{v}^1 = \mathbf{v}^{1/2}$; otherwise $\mathbf{v}^1 = \bar{\mathbf{v}}^1$

Impulse on discrete representation



$$\tilde{I} = \frac{2I}{1 + w_1^2 + w_2^2 + w_3^2}$$

$$\mathbf{v}_i^{new} = \mathbf{v}_i + w_i(\tilde{I}/m)\mathbf{n}$$
$$i = 1, 2, 3$$

$$\mathbf{v}_b^{new} = \mathbf{v}_b - (\tilde{I}/m)\mathbf{n}$$

Repulsion

- Repulsion forces dramatically reduce the number of collisions by pushing away vertices that are in close proximity.
- Ensure that pieces of the cloth are well separated at a distance on the order of this cloth thickness.
- The repulsion force is proportional to the overlap, d, beyond the cloth thickness h (e.g. 1mm)

$$d = h - (\mathbf{x}_b - w_1\mathbf{x}_1 - w_2\mathbf{x}_2 - w_3\mathbf{x}_3)$$

- The spring based repulsion force is modeled with a spring of stiffness k in the normal direction, $\mathbf{n}$

$$\text{repulsion impulse} = kd\Delta t\mathbf{n}$$

Repulsion

- Spring repulsion force is limited to a maximum when the objects touch to avoid stiffness system.

$$I_r = -\min(\Delta t k d, m(\frac{0.1d}{\Delta t} - v_r))$$

Friction

- Use Coulomb's model for friction, both static and kinetic, with a single friction parameter μ.
- The normal force is defined as the negative of the repulsion force, so the friction impulse is $\mu F_N \Delta t$
- The tangential velocity after friction is applied:

$$\mathbf{v}_T = \max(1 - \mu \frac{F_N \Delta t / m}{|\mathbf{v}_T|}, 0)\mathbf{v}_T$$

Collision resolution

- Collision processing algorithm is activated when a collision actually occurs.
- If the geometry is approaching, apply a completely inelastic repulsion impulse. Otherwise, apply a spring based repulsion force.

Demo