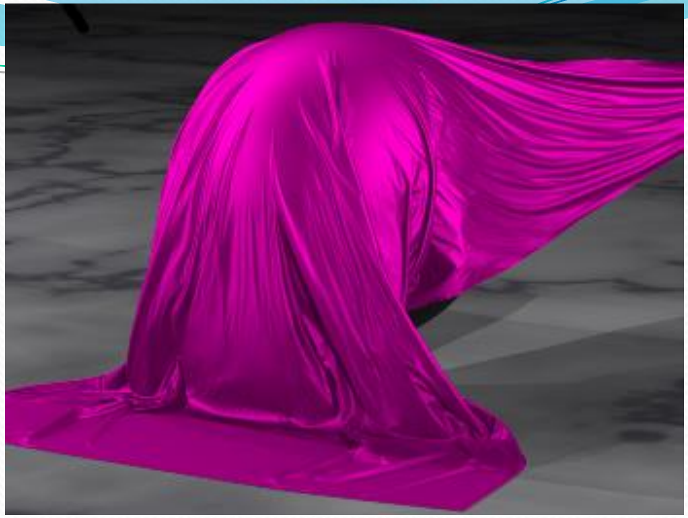
Cloth Simulation

PPGCC – Linha de Pesquisa SIV
Disciplina: Animação Computadorizada

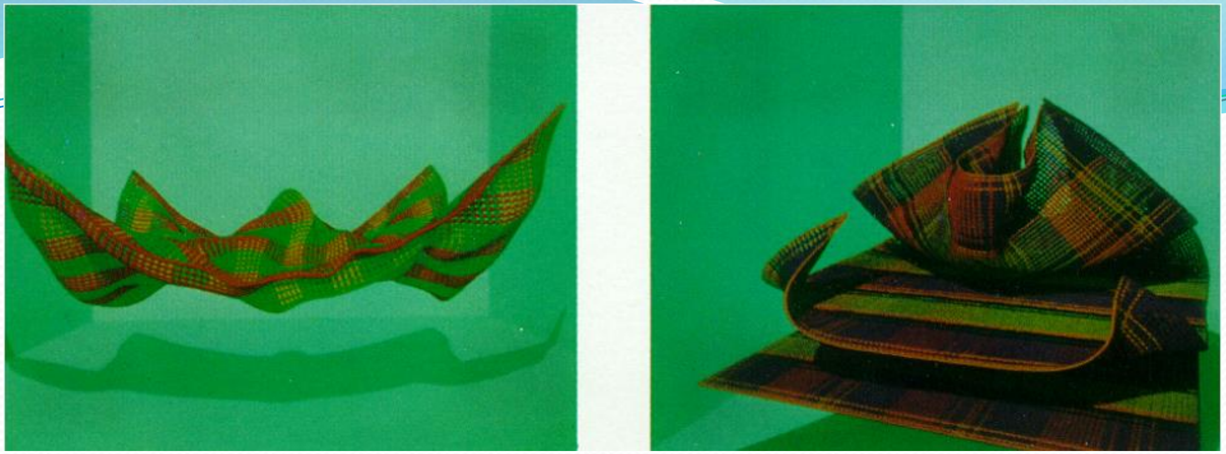
Profª. Drª. Soraia Raupp Musse
Pós-doc Dr Leandro Dihl

12/05/2015

Cloth Simulation



- Cloth simulation has been an important topic in computer animation since the early 1980's
- It has been extensively researched, and has reached a point where it is *basically* a solved problem [2]
- While a lot of existing researches have been done for physical simulation of cloth, areas concerning with cloth appearances are still largely unexplored.
- Two main focuses: engineering and cloth modeling and animation



- The synthesis of cloth objects - SIGGRAPH 1986
- Jerry Weil - AT&T Bell Laboratories, New Jersey
  - Describes a method for modelling cloth material hanging in three dimensions when supported by any number of constraint points.
  - Describes a method for rendering the cloth.

Applicability

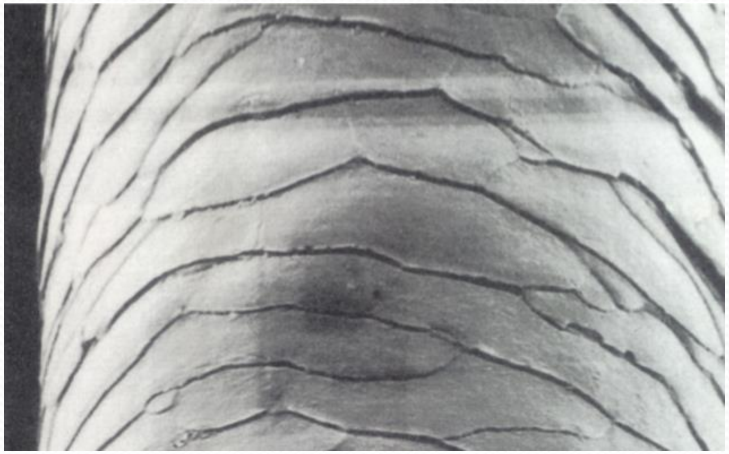
- Textiles are an essential component of most virtual scenes:
  - The appearance of human avatars relies on realistic virtual clothing; other textiles such as carpets and curtains are common indoor elements.
  - Apart from rendering plausible images for movies, virtual prototyping and design are common applications for cloth rendering.
  - Design of seat cushions for cars.
  - In several iterations, different variations of fabrics are produced until a final design decision is found. [1]

Background

- The three inherent scales of cloth can be represented explicitly:
  - fibers,
  - yarns, and
  - compositions;

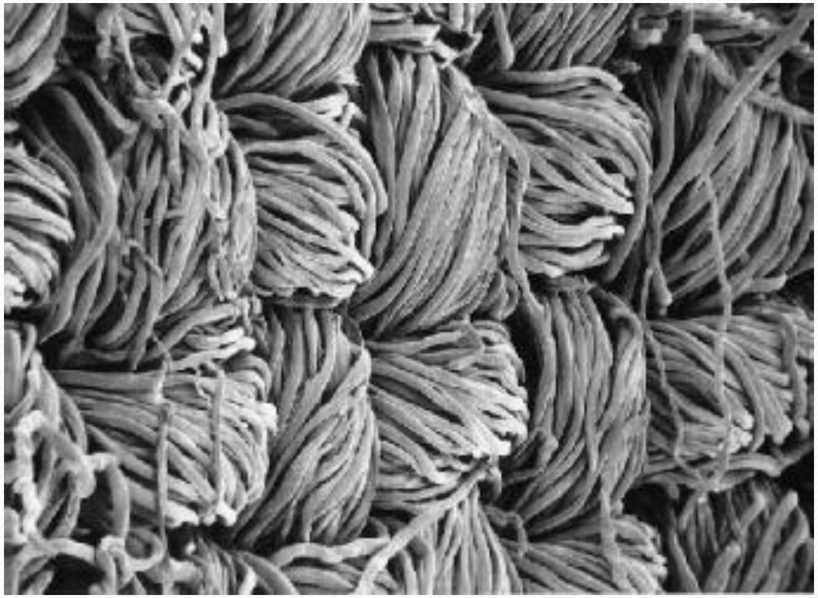
Fibers

- The micro-structure of fibers is described by the geometric and optical properties of the small dielectric fibers that comprise yarns – they are mainly determined by absorption, refractive index (e.g. index of refraction for wool 1.576, silk 1.35, polyester 1.53) and cross sectional shape.



Yarns

- Filament yarns are produced by grouping or twisting a few hundred, long, continuous fibers (e.g. fibers that are taken from cocoons made by the larvae of the silkworm which can be hundreds of meters long).



Composition

- Many techniques have been developed to create pieces of cloth from yarns and fibers.
- Three of the most common ones:
  - Woven cloth and
  - Two types of non-wovens:
    - knitwear and felt.



Approaches

- Geometric approach – start
  - derive the cloth motion and deformation from geometrical curves and functions that are parametrized by time.
- Physically based simulation
  - Particle-based model - Breen et al – 1992 [4]

Physically-based

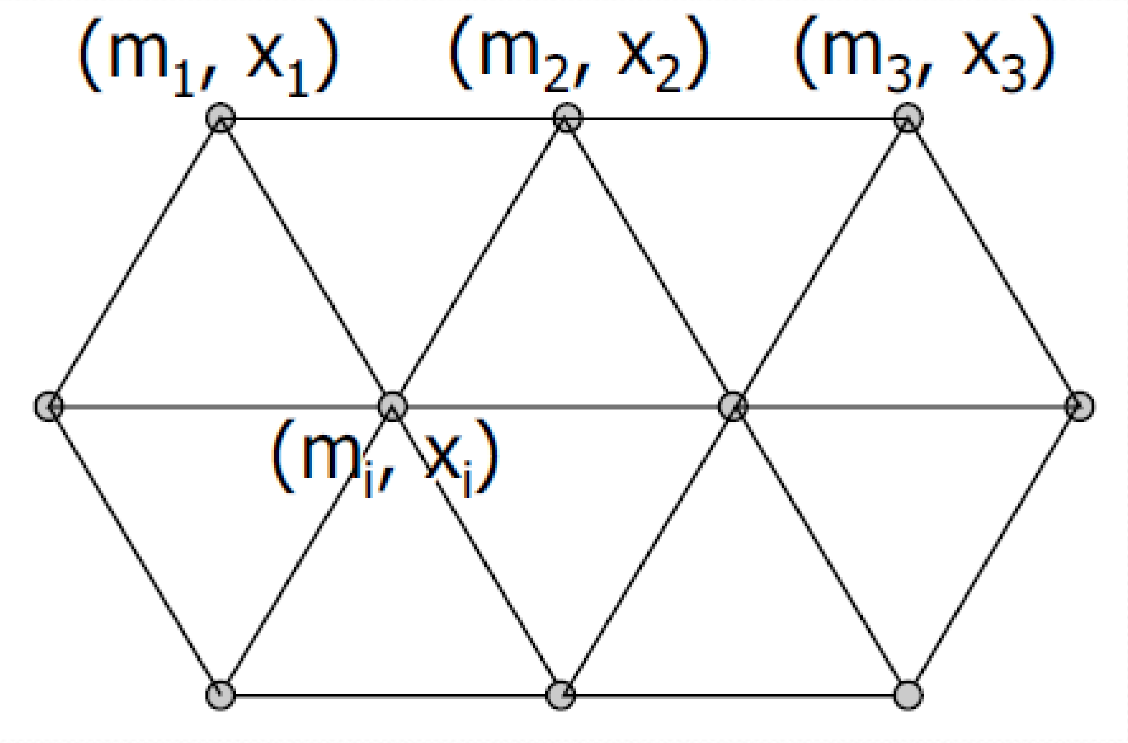
- Although specific details vary (underlying representations, numerical solution methods, collision detection and constraint methods, etc.), there is a deep commonality amongst all the approaches:
  - Physically-based cloth simulation is formulated as a time-varying partial differential equation which, after discretization, is numerically solved as an ordinary differential equation

$$\ddot{\mathbf{x}} = \mathbf{M}^{-1}\left(-\frac{\partial E}{\partial \mathbf{x}} + \mathbf{F}\right).$$

    - x : vector, the geometric state
    - M : diagonal matrix, mass distribution of the cloth
    - E : a scalar function of x, cloth's internal energy
    - F : a function of x and x', other forces acting on cloth

Notations

- Set particles with interconnecting springs



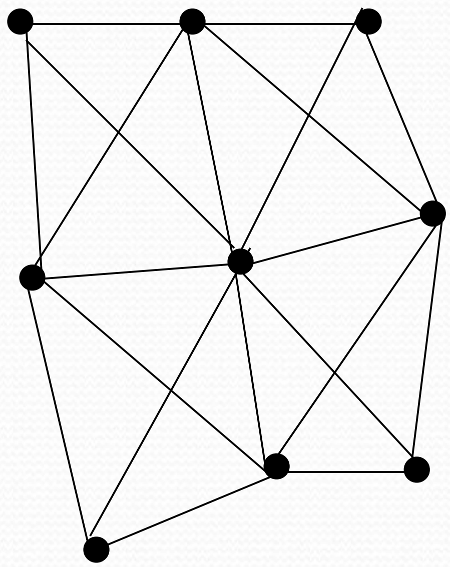
Cloth Simulation with Springs

- The cloths can be treated as a system of particles interconnected with spring-dampers
- Each spring-damper connects two particles, and generates a force based on their positions and velocities
- Each particle is also influenced by the force of gravity
- With those three simple forces (gravity, spring, & damping), the foundation of the cloth system is formed
- Then, we can add some fancier forces such as aerodynamics, bending resistance, and collisions, plus additional features such as plastic deformation and tearing

Cloth Simulation

- Particle

Spring-damper



Particle

$\mathbf{r}$: *position*

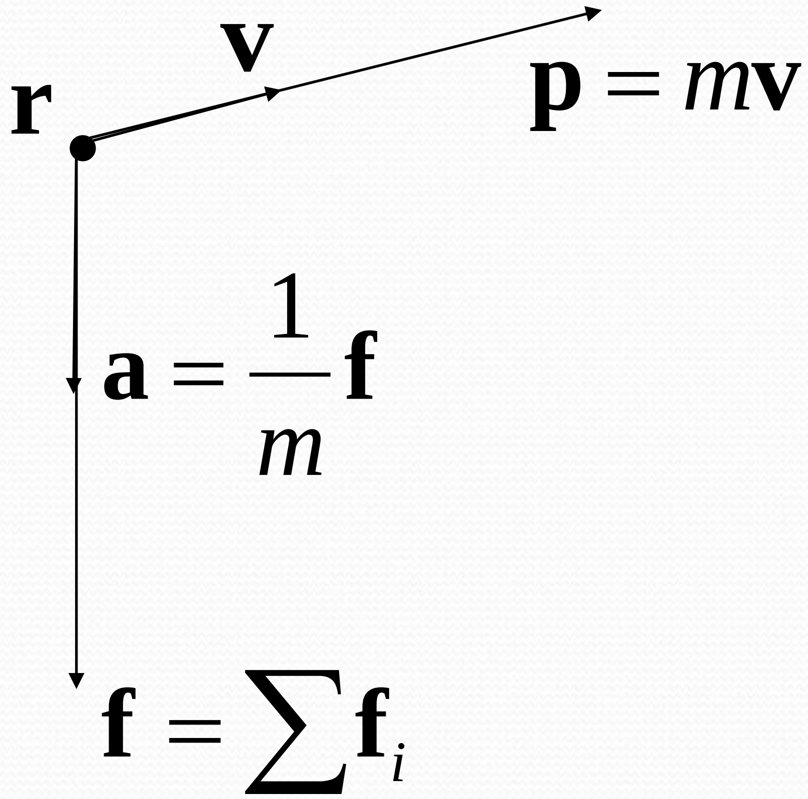
$\mathbf{v}$: *velocity*

$\mathbf{a}$: *acceleration*

m : *mass*

$\mathbf{p}$: *momentum*

$\mathbf{f}$: *force*



Euler Integration

- Once we've computed all of the forces in the system, we can use Newton's Second Law (f=ma) to compute the acceleration

$$\mathbf{a}_n = \frac{1}{m}\mathbf{f}_n$$

- Then, we use the acceleration to advance the simulation forward by some time step Δt, using the simple Euler integration scheme

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \mathbf{a}_n \Delta t$$

$$\mathbf{r}_{n+1} = \mathbf{r}_n + \mathbf{v}_{n+1} \Delta t$$

Physics Simulation

General Physics Simulation:

1. Compute forces
2. Integrate motion

- Repeat

Cloth Simulation

1. Compute Forces

    For each particle: Apply gravity

    For each spring-damper: Compute & apply forces

    For each triangle: Compute & apply aerodynamic forces

2. Integrate Motion

    For each particle: Apply forward Euler integration

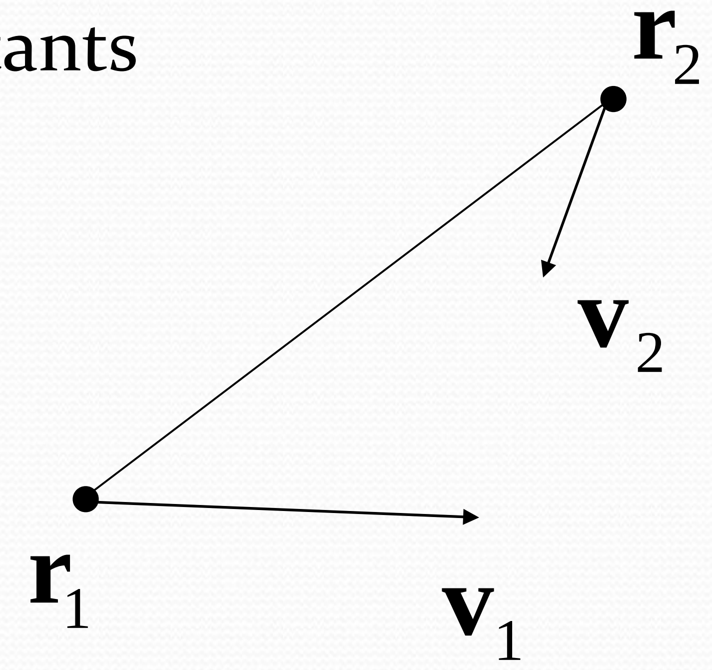
Uniform Gravity

$$\mathbf{f}_{gravity} = m\mathbf{g}_0$$

$$\mathbf{g}_0 = \begin{bmatrix} 0 & -9.8 & 0 \end{bmatrix} \quad \frac{m}{s^2}$$

Spring-Dampers

- The basic spring-damper connects two particles and has three constants defining its behavior
  - Spring constant: k_s
  - Damping factor: k_d
  - Rest length: l_o



Spring-Damper

- A simple spring-damper class might look like:

```
class SpringDamper {
   float SpringConstant,DampingFactor;
   float RestLength;
   Particle *P1,*P2;
public:
   void ComputeForce();
};
```

Spring-Dampers

- The basic linear spring force in one dimension is:

$$f_{spring} = -k_s x = -k_s \left(l_0 - l \right)$$

- The linear damping force is:

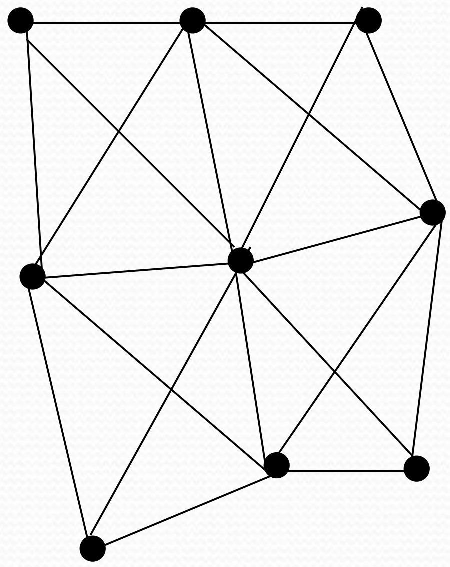
$$f_{damp} = -k_d v = -k_d \left(v_1 - v_2 \right)$$

- We can define a spring-damper by just adding the two:

$$f_{sd} = -k_s \left(l_0 - l \right) - k_d \left(v_1 - v_2 \right)$$

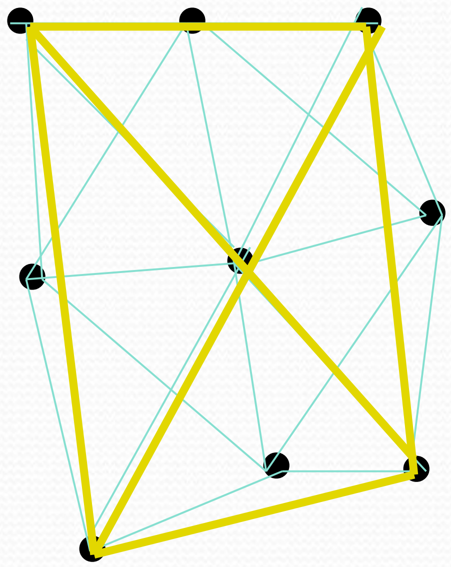
Bending Forces

- If we arrange our cloth springs as they are in the picture, there will be nothing preventing the cloth from bending
- This may be find for simulating softer cloth, but for stiffer materials, we may want some resistance to bending



Bending Forces

- A simple solution is to add more springs, arranged in various configurations, such as the one in the picture
- The spring constants and damping factors of this layer might need to be tuned differently…



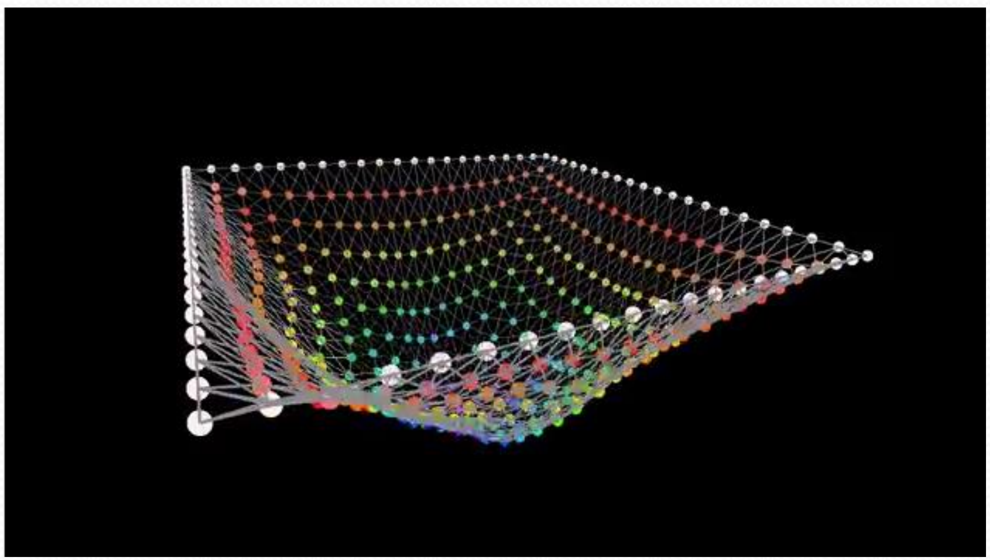
Fracture & Tearing

- We can also allow springs to break
- One way is to define a length (or percentage of rest length) that will cause the spring to break
- This can also be combined with the plastic deformation, so that fracture occurs at the plastic limit
- Another option is to base the breaking on the force of the spring (this will include damping effects)
- It's real easy to break individual springs, but it may require some real bookkeeping to update the cloth mesh connectivity properly…

Integration

- There are *many* methods of numerical integration. Some examples are:
  - Explicit Euler
  - Implicit Euler
  - Midpoint (Leapfrog)
  - Crank-Nicolson
  - Runge-Kutta
  - Adams-Bashforth, Adams-Moulton
  - etc…

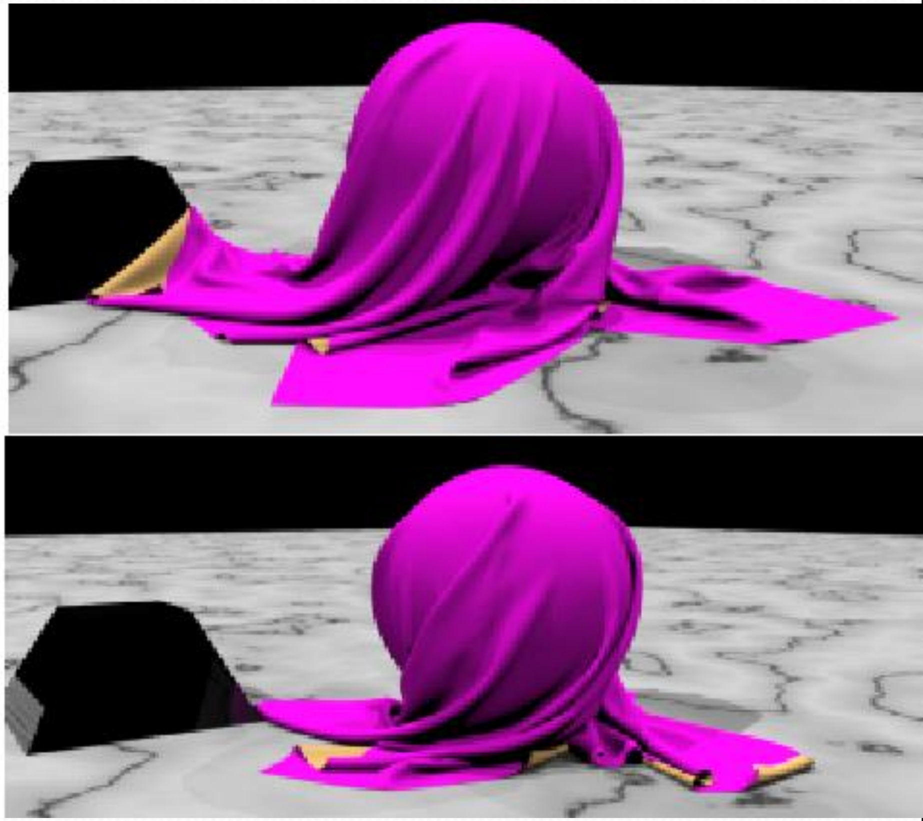
Mass Spring Damper Cloth Simulation



Advanced Cloth

Collision Detection & Response

- Cloth colliding with rigid objects is tricky
- Cloth colliding with itself is even trickier
- There have been several published papers on robust cloth collision detection and response methods



Integration

- Nobody uses forward Euler integration for cloth in the real world
- Modern systems use adaptive time steps, high order interpolation, and implicit integration schemes

Cloth rendering techniques

- Two main directions:
  - The first one uses heuristic methods of procedural modeling to create plausible images.
  - The second one is data-driven and directly based on optical measurements.

Rendering

- For general material rendering applications, much work has focused on statistical surface based models such as Bidirectional Reflectance Distribution Functions (BRDFs) and Bidirectional Texture Functions (BTFs).
- These offer a simplified representation of a material that is often sufficient for rendering from a certain distance.

Cloth Appearance Models BTF

- The most simple approach to modeling cloth is probably to use a combination of texture and normal mapping. However, this approach fails to model the anisotropic highlights and effects of multiple scattering and is nowadays only employed for very simple realtime applications.

References

[1] Kai Schröder, Shuang Zhao, and Arno Zinke. 2012. Recent advances in physically-based appearance modeling of cloth. In *SIGGRAPH Asia 2012 Courses* (SA '12). ACM, New York, NY, USA, , Article 12 , 52 pages. DOI=10.1145/2407783.2407795 http://doi.acm.org/10.1145/2407783.2407795

[2] **Cloth Simulation** - Lecture Notes - Instructor: Steve Rotenberg UCSD, Winter 2004 available: graphics.ucsd.edu/courses/cse169_w04/**CSE169_16.ppt**

[3] Jerry Weil. 1986. The synthesis of cloth objects. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques* (SIGGRAPH '86), David C. Evans and Russell J. Athay (Eds.). ACM, New York, NY, USA, 49-54. DOI=10.1145/15922.15891 http://doi.acm.org/10.1145/15922.15891

[4] Breen, D. E., House, D. H., and Getto, P. H. 1992. A physically-based particle model of woven cloth. The Visual Computer 8, 264–277.