

# Assignment 3: Particle System and Cloth Simulation

Release Date: Thursday, October 1, 2009  
Due Date: Tuesday, October 20, 2009, 11:59pm  
Grading Value: 15%

## Overview:

Cloth simulation has been an important topic in computer graphics since 1980's. In this assignment, you are required to implement a particle system and use this system to simulate cloth based on the mass-spring model. You will also have the opportunity to explore Maya's ncloth. You must at least implement the required features. Anything beyond the basic requirements could be considered as extra credit. Some ideas are provided in the Extra Credit section.

## Requirement:

If you are using our starter code, a particle class and a generalized force structure is already provided, you must implement two subclass forces:

**GravityForce** acts as gravity.

**SpringForce**. Implement it to connect two particles as a damped spring.

**Arrange your particles properly** and connect them into a piece of cloth by adding:

**stretch spring and damping forces,**

**shear spring and damping forces,** and

**bend spring and damping forces.**

Implement **Forward Euler's method** as your integration schemes.

At last, create a piece of cloth in Maya using ncloth, render a short animation.

Describe in your presentation material:

How to change parameters to create different cloth properties?

What kinds of cloth are easy to model?

Which are more difficult?

Compare the cloth you simulated with the one you created in Maya.

## Deliverables:

You are required to submit your code and documentation of how to run it, the features you have implemented and the ones you have not. Please also include the features you implemented for extra credit and explain how you implemented them.

If you are using mac or linux, please make sure your code could run on one of the 5336 machines.

If you are using windows, please submit all the necessary files.

You need to submit one video of your animation to show off your cloth simulation.

You need to submit an animation of your cloth you created and rendered from Maya and also submit your .mb file.

You also need to prepare a 10 minutes presentation and submit your slides or webpage to describe your goals, techniques you used, your results, as well as to answer the questions in the requirement section.

Your handin directory is at [/afs/cs.cmu.edu/academic/class/15464-f09-users/andrewid/Assign3](http://afs/cs.cmu.edu/academic/class/15464-f09-users/andrewid/Assign3).

Note that you will be randomly selected to demo/discuss at least 1 of your 3 projects in class

during the semester.

### **Starter Code:**

You have been provided with starter code you may use for this assignment. The starter code integrates particles to the viewer code for you. You can get the whole package from [/afs.cs.cmu.edu/academic/class/15464-f09/asst3/](https://afs.cs.cmu.edu/academic/class/15464-f09/asst3/)

It should compile without any modification, so you can jump directly into your cloth simulation coding using it. (You can also think about some interesting interactions between the cloth and the character using this starter code for extra credit.)

The files we added to the basic viewer are located at the “Character” directory, including:

Particle.hpp / Particle.cpp

SpringForce.hpp / SpringForce.cpp

Force.hpp / Force.cpp

Cloth.hpp / Cloth.cpp

We also modified BrowserMode.hpp and BrowserMode.cpp to load the particles and springs in the viewer.

There are “TODO”s in the starter code to help you.

Note: You can make any change to the code as necessary, or you can build everything from scratch if you want.

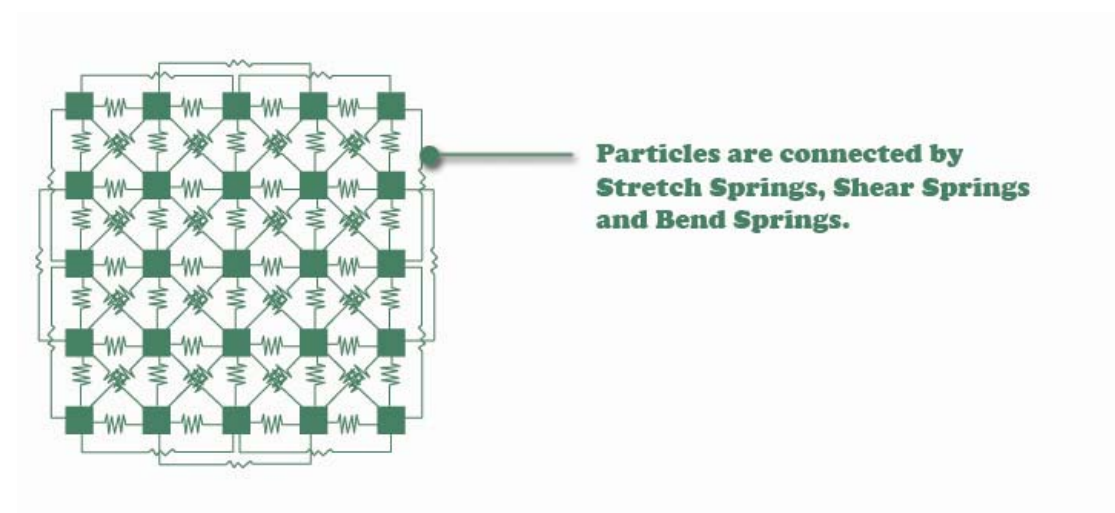
### **Implementation:**

#### **1. Code based cloth simulation**

##### **Mass-Spring Model**

The Mass-Spring Model is a very basic method to simulate cloth. Since it is relatively easy to implement and could achieve good results, implementing this model could be a very good exercise before exploring some more advanced cloth simulation methods.

In this model, cloth is simulated by a grid of particles which are interconnected with spring-dampers. Each spring-damper connects two particles and generates a force based on the particles’ positions and velocities. The structure of this model could be illustrated using the image below:



Each particle is also influenced by gravity.  
With these basic forces, we can form a cloth system.

## Particle

The starter code already provides you with a particle class. Each particle has several properties at a specific time, including its position, velocity, mass and a force accumulator which stores all the forces influence the particle at that time. From these properties, we can calculate how particles would move at next time step.

## Computing Forces

### 1. Gravity:

Computing gravity is simple, we can use this equation:

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

### 2. Spring-Dampers:

A spring-damper connects two particles, it has three constants defining its behavior:

Spring constant  $K_s$ , Damping factor  $K_d$  and Rest length  $l_0$

The equation below gives you the forces the spring-damper exerts on the two particles:

$$\mathbf{f}_1 = - \left[ k_s(|\mathbf{p}_1 - \mathbf{p}_2| - l_0) + k_d \left( \frac{(\mathbf{v}_1 - \mathbf{v}_2) \cdot (\mathbf{p}_1 - \mathbf{p}_2)}{|\mathbf{p}_1 - \mathbf{p}_2|} \right) \right] \cdot \frac{(\mathbf{p}_1 - \mathbf{p}_2)}{|\mathbf{p}_1 - \mathbf{p}_2|}$$
$$\mathbf{f}_1 = -\mathbf{f}_2$$

## Euler Integration Method

Once we've computed all of the forces in the system, we can use Newton's Second Law to compute the acceleration for each particle:

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

In this equation,  $\mathbf{a}_n$  is the acceleration of particle  $n$ ,  $\mathbf{f}_n$  is the sum of all the forces which influence this particle, and  $m$  is the particle's mass.

Then we can use the acceleration to advance the simulation forward by some time step  $\Delta t$ :

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

We can simply go through all the particles in system, and apply those two equations to each of them. The whole system will be updated.

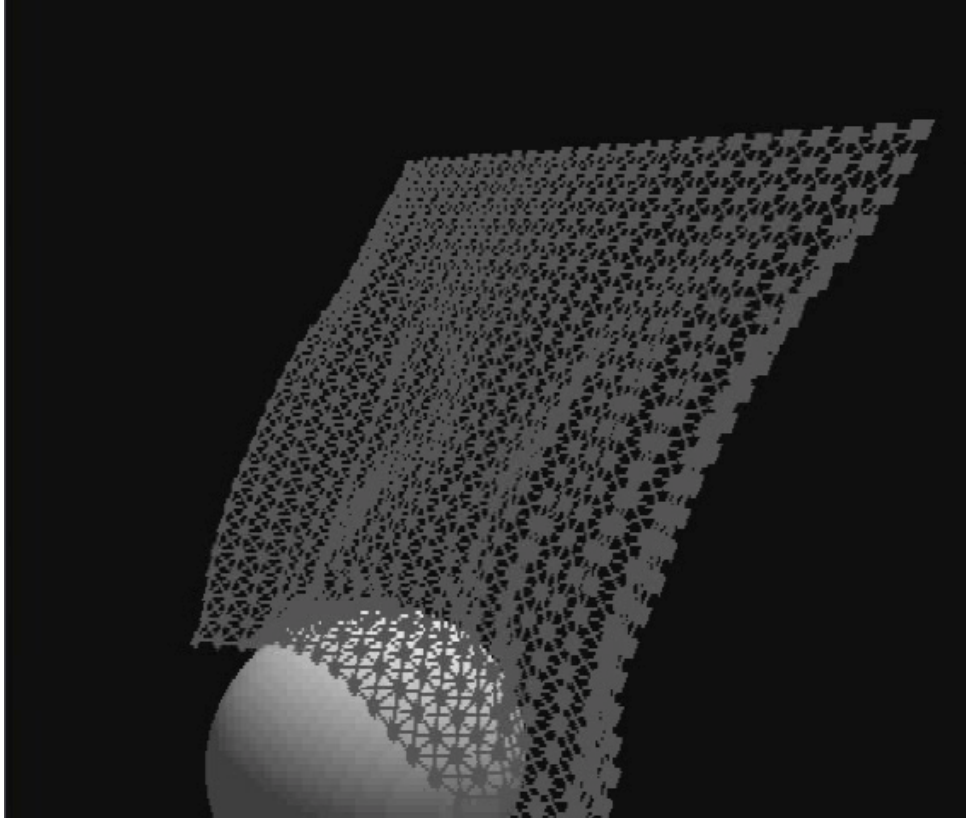
## Cloth Simulation

The algorithm of simulating cloth could be summarized as

1. Compute forces for each particle
  2. Integrate motion: apply Forward Euler Integration for each particle
- Repeat

## Reference Image

(Note: for the basic requirements, you don't need to implement collision.)

**One Last Thing:**

Real cloth simulation rarely uses springs. However, the mass-spring model can serve as a good introduction to other advanced method such as Continuum models.

Also, Forward Euler's Method is not stable and it requires very small time steps.

**2. ncloth in Maya**

ncloth is Maya's built-in cloth solution, it is composed of particles and links connect particles. Links maintain the distance between particles, cross links maintain the angles between links.

Your task is to create a short animation using Maya's ncloth. It could be a piece of cloth falling on a table or sphere; alternatively you can make a shirt for a character. The animation could be just a few seconds.

Maya's Help documentation gives a thorough introduction of ncloth, please refer to it to get started.

**Grading:**

This assignment will be graded on your implementation. Your animation from Maya, your video, report and presentation will also be a part of your grade.

**Tips:**

Start early. This assignment should be more straightforward than our 2<sup>nd</sup> assignment, but it will still take time and effort to make a good simulation.

**Extra Credit:**

1. Other Integration Schemes. Here are a list you can consider to implement:
  - Runge-Kutta 2 and Runge-Kutta 4.
  - Verlet Integrator.
  - Leapfrog Integrator.
  - Implicit Integration. (This one might require some effort, you can think about it for your final project.)Compare your observations on different integration schemes in your report.
2. Implement texture mapping on the cloth. Or anything which can make your cloth look better.
3. Collisions. You'll have to first detect collisions between a particle and an object, and then perform collision response. We recommend using penalty functions.
4. Implement friction, air drag, self collision or any other interesting aerodynamic effects. (like wind, etc.)
5. Interaction between your cloth and the character. You might attach your cloth to the character's shoulder to make it a cape. You can also creative any other interesting interactions. This one would require solving particle-cylinder collision since our character is a "Cylinder Man".
6. Interactive interface. Implement click and drag interaction on the cloth, or any interface you can come up with.
7. Soft Constraints.
8. Hard Constraints. For example, a distance constraint could keep two particles to be a fixed distance apart. You can simulate a pair of strings using distance constraint and then tie your cloth to somewhere using those strings.
9. Simulate Cloth use other method rather than the mass-spring model.
10. Invent your own!