Physikbasierte Modellierung und Simulation
Assignment 3

Present your solutions for this sheet in the exercise on Wednesday, November 20, 2013.
In this exercise, you will connect the particles in a particle system with springs to obtain a mass-spring system. Such systems are used to simulate cloth and solids, for example.

3.1 Springs (20 points)
The force exerted on a particle at position $\vec{x}_1$ by a particle at position $\vec{x}_2$, where both particles are connected by a spring, is given by Hooke’s law:

$$\vec{F}_s = k_s (||\vec{x}_2 - \vec{x}_1|| - l) \frac{\vec{x}_2 - \vec{x}_1}{||\vec{x}_2 - \vec{x}_1||}$$

In this formula, $k_s$ is the stiffness of the spring, while $l$ denotes its rest length.
In a realistic spring, there will also be damping due to energy loss during the deformation of the spring. The damping can be modeled as another force which is defined by the components of the particle velocities $\vec{v}_1$ and $\vec{v}_2$ in direction of the spring:

$$\vec{F}_d = k_d (\vec{v}_2 - \vec{v}_1) \cdot \frac{\vec{x}_2 - \vec{x}_1}{||\vec{x}_2 - \vec{x}_1||} \frac{\vec{x}_2 - \vec{x}_1}{||\vec{x}_2 - \vec{x}_1||}$$

Here, $k_d$ is the damping coefficient.
In the code framework you will find a new class named Spring. In Spring.cpp, implement the function getForce() which returns the sum of Hooke’s force and the damping force.

3.2 Mass-spring system (20 points)
In main.cpp, a mass-spring system is created that simulates a flag on a pole. The mass-spring system, like every system, contains a function computeAccelerations() that will compute the acceleration of each particle. In MassSpringSystem.cpp, implement computeAccelerations() so that the forces of the springs are correctly applied to each particle. You can run the simulation to see if the particles start moving appropriately.

3.3 Per-particle damping (10 points)
You have already included a damping term for springs, which is motivated by the fact that the deformation of a spring leads to a loss of energy and therefore slows down the movement of the particles. Another way to define a damping is to directly apply a damping force to each particle. This damping force $\vec{F}_p$ is proportional to the velocity $\vec{v}$ of the particle:

$$\vec{F}_p = -k_p \vec{v}$$
This damping can be explained by the friction of the particles against the surrounding medium. It is also often more numerically stable than per-spring damping. Modify \texttt{computeAccelerations()} to account for per-particle damping.

### 3.4 Gravity (10 points)

Up to now, not much is happening in the simulation because there is no external force. Add an external gravitational acceleration of 9.81 m s$^{-2}$ in the negative $y$ direction to \texttt{computeAccelerations()}. 

### 3.5 Constraints (10 points)

You will notice that gravity makes the flag fall down. However, some particles should be fixed to the flag pole. These particles have their \texttt{fixed} flag set to \texttt{true}. At the end of \texttt{computeAccelerations()}, set the acceleration of these particles to zero to keep them fixed.

### 3.6 Wind (20 points)

To make your flag move, add another contribution for “wind”. We will not attempt to accurately simulate the fluid dynamics of wind, but just add a certain force to each particle to pretend the particle is affected by wind:

\[
\vec{F}_w = \begin{pmatrix} 1.5 + \cos(t/T) \\ 0 \\ \sin(t/T) \end{pmatrix} \vec{F}_0
\]

Here, $T$ is the \textit{period} of the oscillating wind direction, and $\vec{F}_0$ is the basic wind \textit{force}. $t$ is the simulation time, stored in the \texttt{time} variable of the system. Include the contribution of the wind in \texttt{computeAccelerations()}.

### 3.7 Experiments (10 points)

Try at least one of the following experiments:

- What happens when you change amount of per-spring and per-particle damping? How does this affect the stability of the simulation?

- How does the behavior of the flag change if you change the configuration of the springs (e.g. by removing the diagonal springs)?

- You have so far used the mass-spring system to simulate cloth; try to use the same system to simulate, for example, a steel bridge. You will probably have to use a much higher stiffness; how does this affect the stability of the simulation?

\begin{verbatim}
http://graphics.tu-bs.de/teaching/lectures/ws1314/pbm/
\end{verbatim}