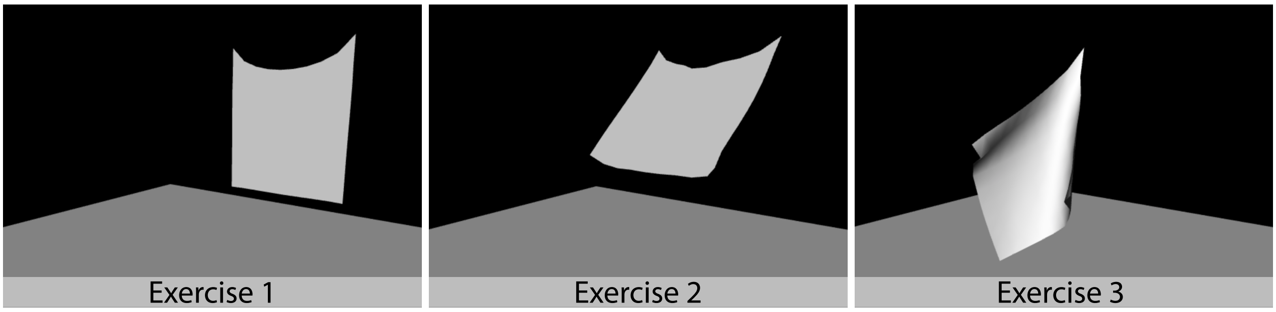
Università della Svizzera italiana

Faculty of Informatics

Computer Graphics (Fall 2022)

Assignment 11: Mass-spring System

December 2 2022



In this assignment, you will implement a simple mass-spring system to animate a piece of cloth. The template already provides you with the rendering environment and cloth definition (mass-spring topology). Your tasks include implementing the simulation step using semi-implicit Euler integration (Excercise 1), adding external forces to get an interesting animation (Excercise 2), as well as recomputing the normal vectors of the cloth to get a nicer rendering (Excercise 3).

Template

The new template implements for you the rendering of the cloth and the cloth geometry definition. The rendering does not update at this point the normal vectors, which is the goal of the Exercise 3. Important information regarding the template:

- `geometry.js` file includes now of the definition cloth geometry, i.e., vertices and triangles. You are free to analyze the code, but what is important for the assignment is the order and the number of vertices. The cloth is defined as a grid of vertices. One row and one column consists of `cloth_size` vertices. This gives the total number of vertices (also particles for the simulation) `cloth_size`×`cloth_size`. The vertices of the cloth are ordered row by row starting from the top-left corner. Hint: the first and the last vertices of the top row have indices 0 and `cloth_size-1`.
- The whole rendering pipeline is defined for you. In this assignment, you do not have to code anything related to WebGL. In particular, we have already defined for you the GPU buffers, which store positions and normals, that can be updated every frame. We also added the code which does the update based on the content of `cloth_vertices` and `cloth_normals` arrays.
- The template contains a well-documented skeleton of the simulation. Please check the main files starting at line 180. The functions you should modify in the assignments are: `updateCloth`, `updateNormals`. Of course, you are encouraged to play around with the other parts, especially parameters of the springs and time step.

Exercise 1 [9 points]

To solve this exercise, you need to modify the `updateCloth` function to implement one step of the simulation. Follow the comments and the skeleton, which should make your task easier. Remember about including both the forces coming from Hooke's spring model and damping. Remember to constrain some of the particles, so they do not move. Otherwise, your cloth will not hang but fall under gravity.

Exercise 2 [3 points]

Use the simulation to animate the cloth in an exciting way. To this end, add time-depended external forces acting on each particle. You can use for this purpose `time` variable, which is already defined in the template and contains the time elapsed from the beginning of the simulation counted in milliseconds. Like that, you can, for example, simulate the cloth moving on a wind. Additionally, you can release at some point the cloth such that it flies away.

Exercise 3 [3 points]

If you solved exercise 2, you probably realized that something is wrong with the illumination of your cloth. In particular, you do not see it nicely folding. This is because the normal vectors of the cloth are not recomputed during the simulation. To fix this, implement the function `updateNormals`, which should updated the `cloth_normals` array based on `cloth_vertices` array.

Submitting the assignment

Submit an archive file (ZIP) with all the files required for running your application. The additional readme file should contain information about the authors of the solution as well as explanation of any encountered problems.

Solutions must be returned on December 30, 2022 via iCorsi3