AM227: Computational Methods for Physical Sciences
7nd Assignment
Research Project: Mass-Spring System for Soft Body Simulation

Yukun Xia

December 13, 2018

1. Project Background

From continuum assumption, elasticity theory has been perfectly developed for a long time. According to the results from elasticity theory, computational scientists implemented discretization and weak form of PDE to build the Finite Element Method (FEM). FEM has already been successfully and broadly used in structure analysis as well as multiphysical simulation. However, FEM essentially needs to solve a matrix problem "Ax=b", while here matrix A is usually pretty large, and thus it's a computationally intensive task to solve vector x. For a 3D structure, if there are n nodes in each edge, then there are n^3 nodes totally, and solving $A^{-1}b$ has the complexity $O(n^9)$. Another latent problem for FEM is the difficulties to keep programs converging when local deformation is large, and nonlinear effect dominates there.

Switching from area or voxel view of material, we can also regard them as a combination of particles. This perspective belongs to mesh-free method. The first mesh-free method is Smooth Particle Method. Liquids are divided into particles, with particles moving and interacting. If we know the position of every particle, we can then calculate the surface of liquid by multiplying particle positions with the kernel function. A smooth kernel can always lead to smooth surface.

For solid material, the relationship between particles should have much more limits. One way is to add springs to connect neighborhood particles. Since this method only count local interaction, it's quite computationally cheap, and can work for real time simulation [1] [2]. In this project, I would firstly introduce how to build such a spring-mass system, and then demonstrate some simple cases to verify the feasibility, and finally, electrostatic forces are incorporated for Dielectric Elastomer Actuator study.

2. Project Analysis

A heuristic way to solve deformation problem is to regard a block of solid as balls connected by springs, and the linkage only exists between adjacent balls. Discretizing solids into balls, and setting boundary condition are similar to FEM. The next step is to apply force to certain balls, so these balls will have acceleration and start to move, like the spreading of strain wave. For a dynamical system like this, the motion will never end. To stabilize the system, artificial damping is added to every nodes, so every ball suffers a negative force in its direction of moving. Finally, there will only be one stable position of every node, and the deformation has been simulated.

If this system is further compared with neural network, the balls are nodes in network, force or displacement implemented to certain nodes are data input, and when the velocity spreads to the fixed boundary nodes, the wave will stop and turn to propagate backwards.

In the optimal situation, the complexity of this method is only O(n), where n is the number of nodes. The reason is that the time for the dynamical system to stabilize can be limited and constant, and in each time step, there are no more than 10 forces implemented on each node in 1D case. 10 forces include 8 spring forces, 1 external forces, and 1 artificial viscosity force.

3. Forces in Dynamical system

Here's a node, whose index is (i,j). Node(i,j) connects to 8 neighbors with spring, and 8 spring forces are defined accordingly.

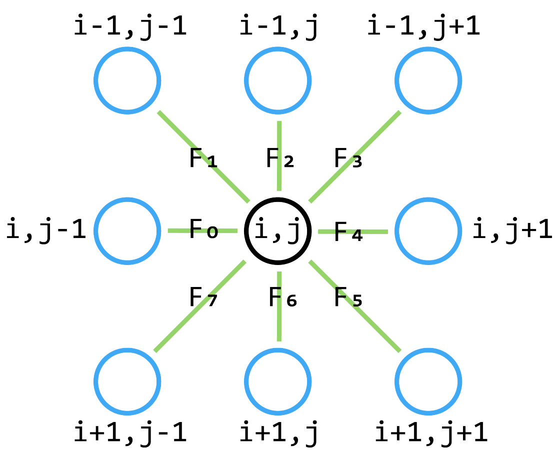

Figure 1. Schematic diagram of nodes and springs in 2D. Black circle, blue circles and green lines are respectively the node being studied now, neighborhood nodes, and springs connected to central node.

4. Pseudo Code

Algorithm 1 Spring-mass system for deformation simulation

```
Discretization;
Initialize acceleration, velocity, and position matrix;
Initialize distance matrix; # distance between every pair of adjacent nodes
Set boundary condition;
Apply F_external to certain nodes;
while Max(velocity) > ϵ do
    for every non-fixed node do
        a[node] ← (F_external + F_spring + F_viscosity)/m
        v[node] ← v[node] + a[node] × δ_t
        x[node] ← x[node] + v[node] × δ_t
    end for
    Update distance matrix
    Update F_spring
end while
```

5. Parameters and Integration Methods

5.1 Spring Stiffness

In Fig.1, there's 4 springs in x and y axises, and 4 in diagonal lines. The first 4 springs generate compression and stretching stress, while the other 4 represent the shearing effect. To tune the Poisson ratio, we should derive the its relationship to the stiffnesses. After solving the force balancing equations in a single square element including 4 particles, in a simple stretching case, we have

$$\frac{\Delta y}{\Delta x} = \frac{k_2}{2k_1 + k_2} = Poisson\ Ratio$$

, where, k_1 and k_2 are respectively the stiffness of the x-y axises springs and the diagonal springs, and Δx and Δy are respectively the displacement along and vertical to the stretching force. For incompressible material (Poisson Ratio = 0.5), the stiffnesses should satisfy this equation: $k_2 = 2k_1$.

5.2 Lattice Structure

We can always find more structure to assemble the spring-mass system in one material with a certain shape. One problem with my D2Q9-like structure is the difficulties to simulate isotropic materials. I then tried to build material with particles randomly distributed, and use Voronoi method to get connection between neighborhoods. Voronoi occupied me a long time to get it work in my spring-mass system. However, results are not like my expectation. There's nearly no Δy here, so Poisson Ratio is always 0. I also checked that for D2Q9 structure, Voronoi graph will not incorporate the diagonal springs. What's worse is that the surface is not smooth due to the limitation of how many particles I could have.

Some recent work also used other structures for spring-mass simulation, here's one example: [3] Chapter 3.3 - Types of Lattice.

5.3 Virtual Viscosity

To stabilize the mass-spring network from external stimulation, a virtual viscosity term is intentionally added: $F_{visocsity} = -v \cdot damping$. Like all dynamical system, different damping coefficient may correspond to overdamping or underdamping situations. In our spring-mass system, the best viscosity coefficient is the critical value between over and under damping, otherwise, the time for convergence will be longer. In the following figure, two values are picked to test this effect. Usually, underdamped system with over-shots will cause further unstable effect, for example, one particle pass another particle. Consequently, we would rather have an over-damped system, if the critical value is unknown.

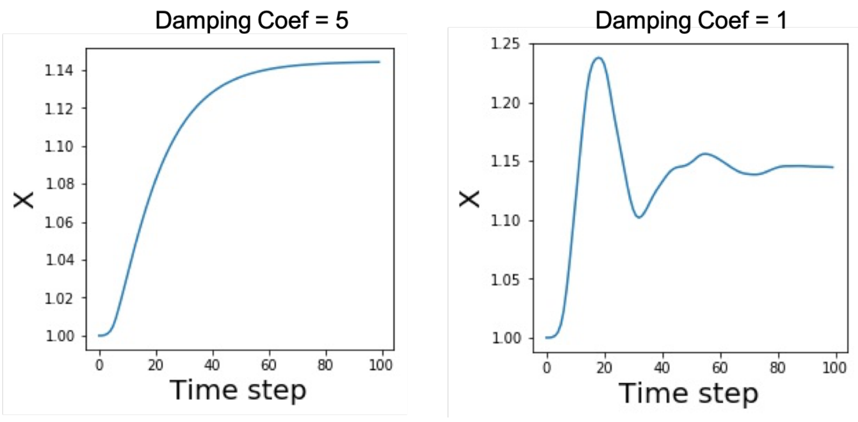

Figure 2. Using two damping coefficients for the same spring-mass system. The right one with less damping is underdamped, having over-shots at $\sim$ 20th and 55th time steps.

5.4 Dynamical Integration Forms

Here we choose four integration methods: Explicit Euler, Euler-Cromer, Leap Frog, Velocity Verlet.

(1) Explicit Euler:

$$x_{t+h} = x_t + hv_t$$

$$v_{t+h} = v_t + h\frac{1}{m}F_t$$

(2) Euler-Cromer:

$$v_{t+h} = v_t + h\frac{1}{m}F_t$$

$$x_{t+h} = x_t + hv_t$$

(3) Leap Frog:

$$v_{t+h/2} = v_t t - h/2 + h\frac{1}{m}F_t$$

$$x_{t+h} = x_t + hv_{t+h/2}$$

(4) Velocity Verlet:

$$v_{t+h/2} = v_t + \frac{h}{2}\frac{F_t}{m}$$

$$x_{t+h} = x_t + hv_{t+h/2}$$

$$v_{t+h} = v_t + \frac{h}{2}\frac{F_t}{m}$$

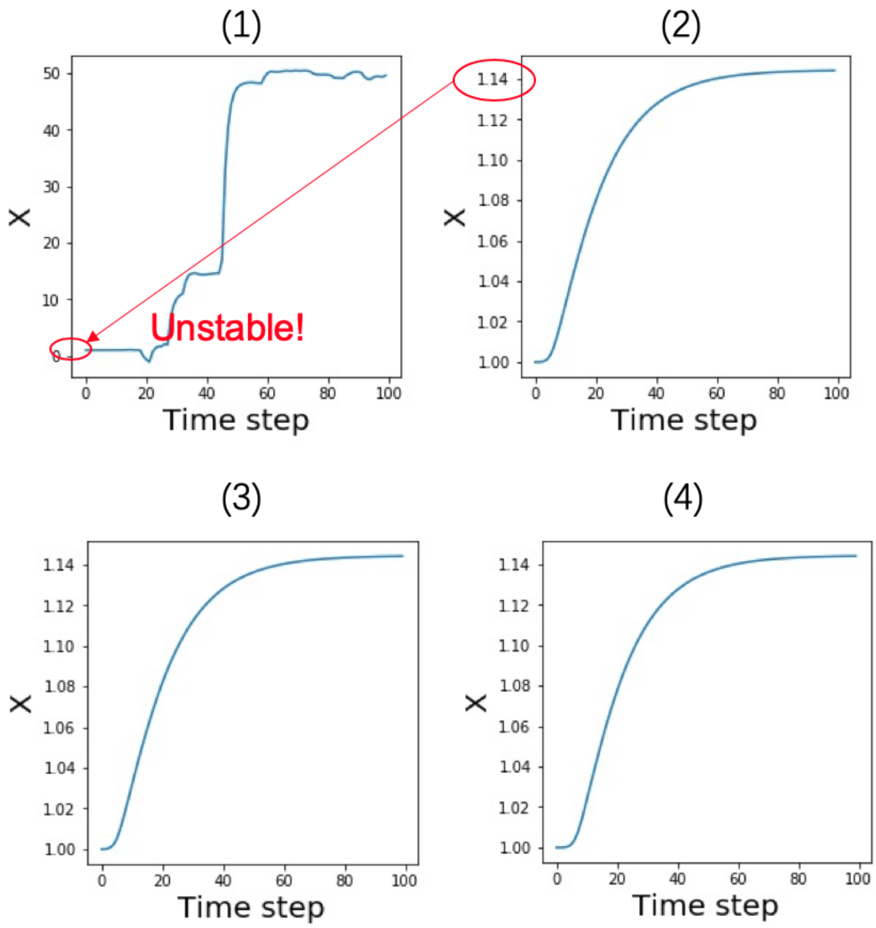

Figure 3. In the testing spring-mass system, Explicit Euler is the most likely one to be unstable. Others methods have nearly the same smooth x-t curve.

6. Simple Cases

In this section, a few basic types of forces are applied to a block of 6x6 particles.

6.1 Pressure

Both the effects of one side pressure (asymmetric) and double sides pressure (symmetric) are calculated, showed in Fig. 4 and 5. As we can see, the force is in horizontal direction, but also cause the block to inflate along the vertical direction, due to the Poisson effect.

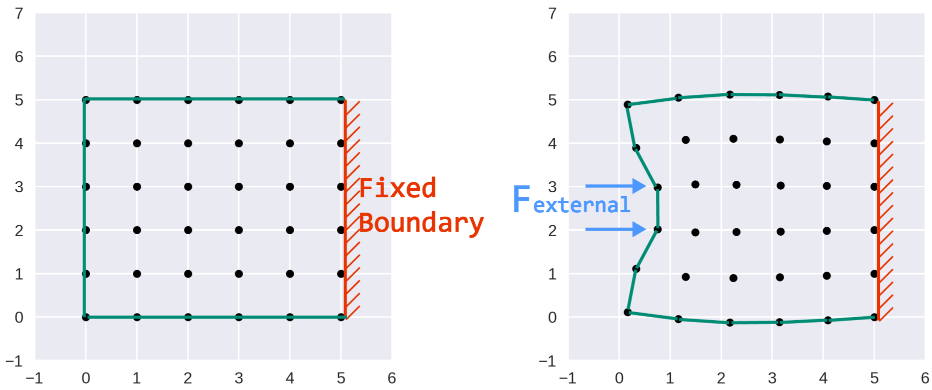

Figure 4. Pressure force are applied from left side to right, on two central nodes. On the y-axis, material slightly swells, while on x-axis, it shrinks.

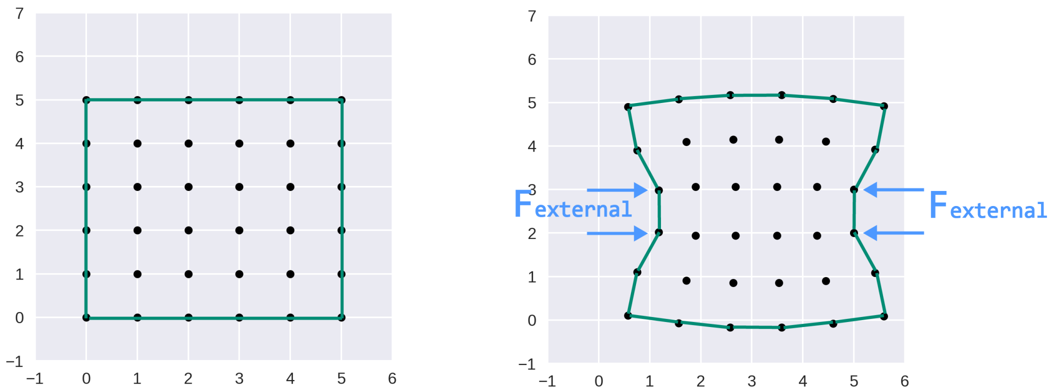

Figure 5. Pressure forces are from both left and right sides. On the y-axis, material slightly swells, while on x-axis, it shrinks.

6.2 Shearing

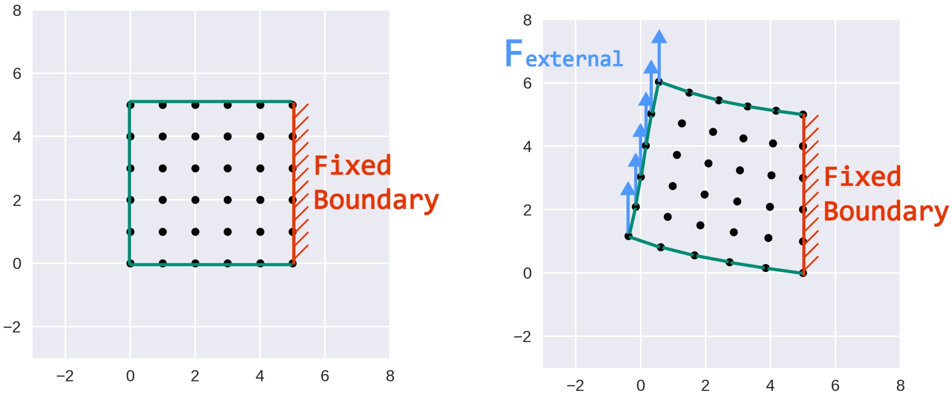

Figure 6. Shearing force is initially applied onto the left edge, and the direction is to the positive y-axis. As we can see, the upper edge is compressed, while the bottom edge elongates.

6.3 Three-Point Bending

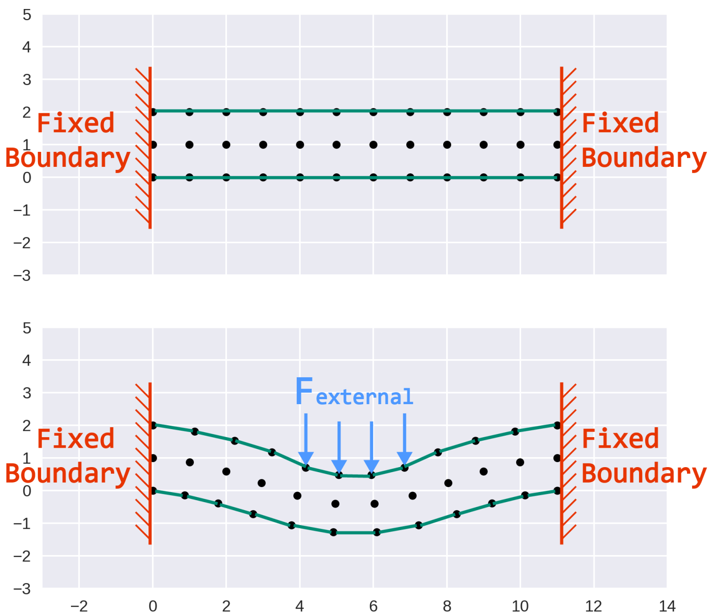

Figure 7. A rectangular material is fixed on its left and right edges. A downward force is applied to the central part of the upper edge. Finally, there is a sine-like curve here.

7. Spring-mass system for dielectric elastomer simulation

To give others a sense of what dielectric elastomer is, here's a brief introduction from Wikipedia (https://en.wikipedia.org/wiki/Dielectric_elastomers):

Dielectric elastomers (DEs) are smart material systems that produce large strains. They belong to the group of electroactive polymers (EAP). DE actuators (DEA) transform electric energy into mechanical work. They are lightweight and have a high elastic energy density. They have been investigated since the late 1990s. Many prototype applications exist.

DEA has a sandwich structure. Two compliant electrodes are on the two sides of a rubber membrane, and after voltage is applied to electrodes, the membrane will deform due to Maxwell Stress. Since rubber is typically incompressible, the membrane can finally get both compressed at thickness direction and elongated at other directions. This structure is showed in Fig. 8(1).

The first step to simulate DEA is to calculate the charge distribution in electrodes. The method I used was Method of Moments (a good tutorial from MIT OCW https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-635-advanced-electroma lecture-notes/Mar10.pdf). The top and bottom springs are set with different fixed voltages. Each charged spring contributes a certain value of potential to any other charged springs.

$$V = \int\int \frac{1}{4\pi\epsilon} \frac{\sigma(x', y')}{\sqrt{(x-x')^2 + (y-y')^2 + (z-z')^2}} dx'dy'$$

, where V is the potential of one spring, ϵ is the permittivity, and x' and y' are integrated in the conductor's surface. The summation of all the contributions should lead to all the given voltages on springs. After knowing the charge density on each electrodes, we can easily use Coulomb's Law to calculate the total electrostatic force acted on each charged spring. The forces on springs are divided into two equal parts to the particles connected to them. The other parts of work is simply like the basic cases.

For DEA with pre-stretching along area directions has nearly uniform force distribution, result should be like a simple compression case. So here, some more complicated cases are tested, where the bottom of the rubber film is fixed, and on the top, different area of electrodes are deployed. Results are showed in Fig. 8 (2)(3).

For the DEA film whose top is fully covered by an electrode (showed in (2)), the membrane has a strong marginal effect, and the effect grows with voltage increasing. For DEA whose top is partially covered in the central region, when voltage is low, the main effect is a slight compression in the middle, while when voltage has a higher value, the top surface becomes wavy and start to get unstable. One interesting detail here is the unstable part is not at the margin of top electrode like the former case. This may be due to the indentation at the first step, and charges concentrate more on the central part, which should be checked later.

By the way, be cautious that the figure has different length scale in x and y axises, and the rubber in (2) has 2 times larger thickness than (3) is only for better visualization.

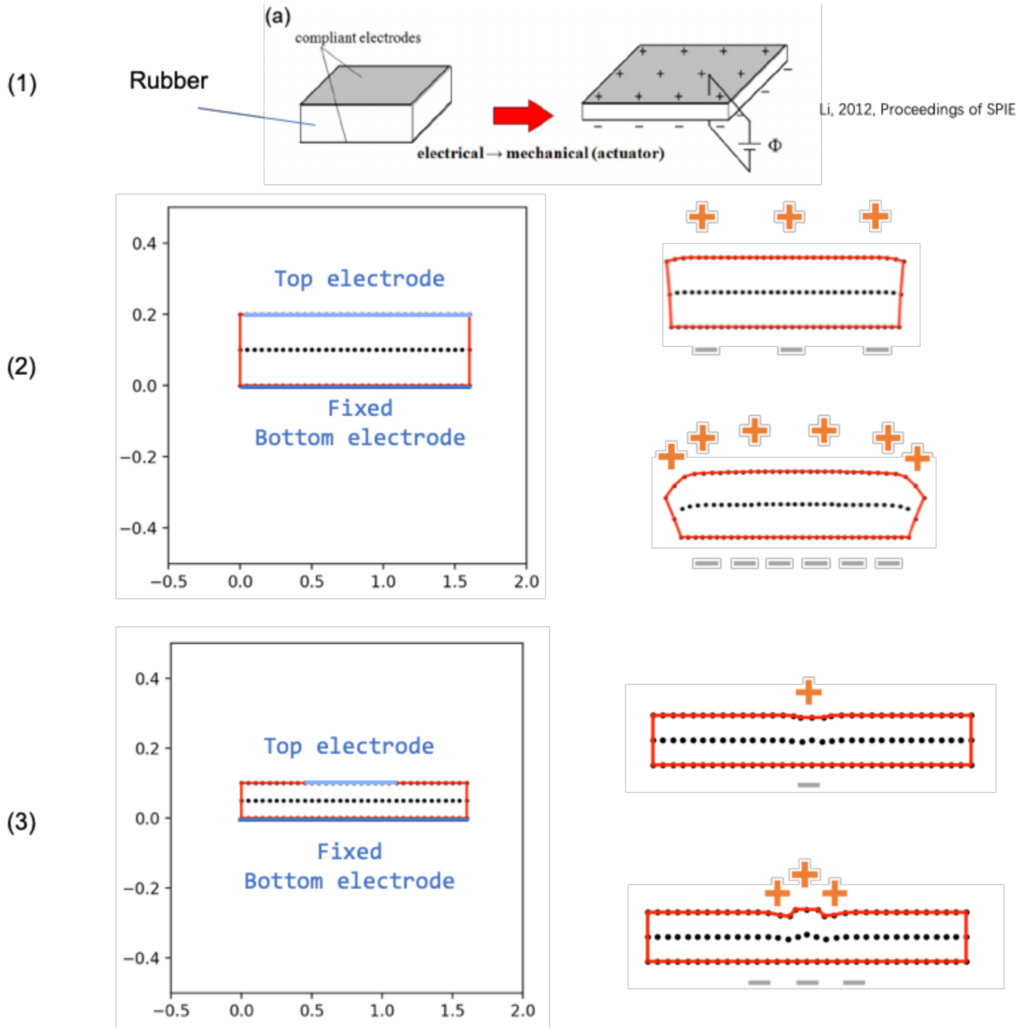

Figure 8. (1) Schematic diagram of Dielectric Elastomer Actuator. (2) (3) DEA with fixed bottom electrode and free top electrode.

8. Discussion and Conclusion

As we can see, using spring-mass dynamical system, we can predict deformation of solids easily, even if the maximum strain here is not a small one ($strain_{max} > 5\%$).

By adding dampen to nodes, the velocity can decrease exponentially, considering $\vec{F}_{viscosity} = -Coef \times \vec{v}$, where $\vec{v}$ is the velocity. It can be a little bit tricky to tune the value of $Coef$, if $Coef$ is too large, then the system will be over-dampened or nodes will move to the false direction and may cause instability, while if $Coef$ is too small, it will be pretty difficult for nodes to get slow down, and increases the computational cost.

Another important parameter is Poisson Ratio. For rubber-like material, one assumption is incompressibility, and the Poisson Ratio should be equal to 0.5. We can tune the ratio of different springs to get a certain Poisson Ratio easily. But problems still remain in how to get an isotropic rubber by some spring-mass assembly, and how to incorporate the nonlinear stress-string relationship of rubber.

In the demonstration sections, firstly, several simple deformation modes for soft solid materials were tested: compression, shearing, and bending. These modes worked well under the spring-mass assumption. Then I shifted to Dielectric Elastomer, a more complex and real-world problem in soft actuator simulation. Method of Moments was used to get the nonuniform surface charge distribution. In this electrostatic problem, the charge density were defined on springs, rather than on nodes, for the definition consistency among both marginal and internal particles. With this result, we could easily incorporate Coulomb's Law for the force calculation, followed by the normal spring-mass method. The results should be compared with FEM results later.

Finally, spring-mass method is also supposed to be examined the validation for other complex situations, such as contact, fracture, and other multi-physics field problem.

References

[1] Peter E Hammer, Michael S Sacks, J Pedro, and Robert D Howe. Mass-spring model for simulation of heart valve tissue mechanical behavior. *Annals of biomedical engineering*, 39(6):1668–1679, 2011.

[2] Ullrich Meier, Oscar López, Carlos Monserrat, Mari C Juan, and M Alcaniz. Real-time deformable models for surgery simulation: a survey. *Computer methods and programs in biomedicine*, 77(3):183–197, 2005.

[3] Karolina Golec. *Hybrid 3D Mass Spring System for Soft Tissue Simulation*. PhD thesis, Université de Lyon, 2018.