# CMIS18 - Final Project

Isabela Blucher

June 15, 2018

## Introduction

This week, we are dealing with the modeling of elastic solids. In the physical world, elasticity is the ability of a deformed body to return to its original shape and size when the forces causing the deformation are removed. The elasticity problem is related to the concepts of stress and strain. Stress is a measure of the force put on the object over the area. Strain is the change in length divided by the original length of the object after a force has been applied to it. The relationship between stress and strain that a particular material displays is known as the stress-strain curve and it is unique for that particular material. Each material has its elasticity described by that relationship. Different materials and objects also have different parameter values, such as elastic limits, and if the strain on a given object is bigger than its elastic limit, it will permanently deform or even fracture.

For this project we want to discretize the modeling of forces applied to an elastic solid and observe the consequences such as its deformation over time. For this we make use of the Finite Volume Method (FVM).

## Discretization of the problem

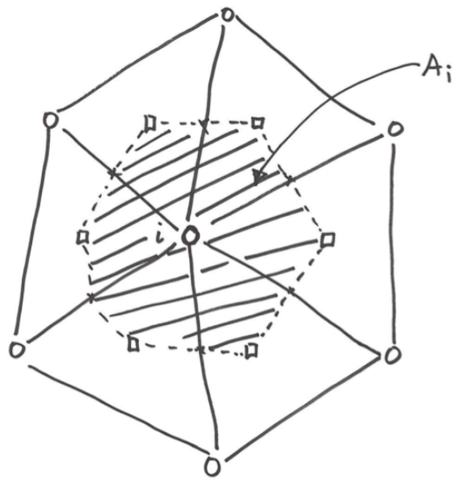
To derive the method that will be implemented in the simulator we have to explain how to approximate a solution for this week's governing equation, Cauchy's equation of motion:

$$\rho_0 \ddot{\mathbf{x}} = \mathbf{b}_0 + \nabla_{\mathbf{x}} \cdot \mathbf{P} \tag{1}$$

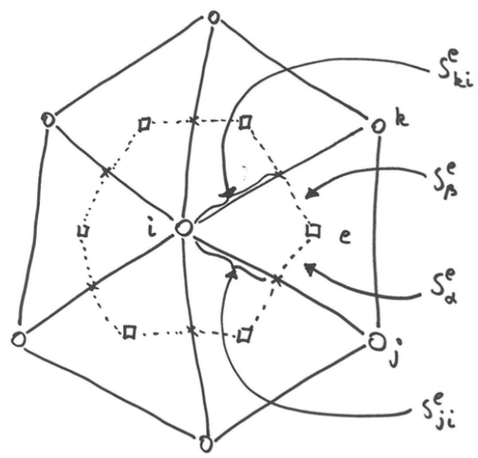
To begin explaining the terms in (1), it is necessary to differentiate between two sets of coordinates. Let's call $\mathbf{X}$ the undeformed or material coordinates, and $\mathbf{x}$ the deformed or spatial coordinates. The deformation between the sets is given by a mapping from the material to the spatial coordinates. It is important to keep in mind that the same global reference coordinate system is used for both sets.

With that said, we can start going through the terms in the equation. Terms with subscript 0 mean that we're referring to functions of material coordinates. So $\rho_0$ is the material density field, $\ddot{\mathbf{x}}$ is the second derivative of $\mathbf{x}$ with respect to time, $\mathbf{b}_0$ is the body force density field and $\mathbf{P}$ is the 1st Piola-Kirchhoff stress tensor.

Given the problem, the discretization of it will be done with the FVM approach on a 2D traingular mesh, with median dual vertex centered control volumes. We will consider that the deformation gradients are constant over triangular elements and we will use a simple first order finite difference approximation in time.




(a) Highlited control volume




(b) Surfaces of control volume and triangle element vertices

Figures a and b illustrate the median dual vertex centered control volumes in the triangular mesh.

Now that the structure of the implementation was introduced it is possible to write up the volume integrals and derive the final governing equation for the elasticity problem.

$$\int_{A_i} \rho_0 \ddot{\mathbf{x}}_i \, dA = \int_{A_i} \mathbf{b}_0 \, dA + \int_{A_i} \nabla_{\mathbf{x}} \cdot \mathbf{P} \, dA \tag{2}$$

where $A_i$ denotes the control volume of the $i^{th}$ vertex of the domain.

Since $\rho_0 \ddot{\mathbf{x}}_i$ and $\mathbf{b}_0$ are not dependent on the control volume integral, we can remove them and integrate on the control volume, obtaining the area of the control volume.

$$\rho_0 \ddot{\mathbf{x}}_i \int_{A_i} dA = \mathbf{b}_0 \int_{A_i} dA + \int_{A_i} \nabla_{\mathbf{x}} \cdot \mathbf{P} \, dA \tag{3}$$

$$\rho_0 \ddot{\mathbf{x}}_i A_i = \mathbf{b}_0 A_i + \int_{A_i} \nabla_{\mathbf{x}} \cdot \mathbf{P} \, dA \tag{4}$$

Applying the Gauss-Divergence theorem and the Leibniz Rule we can transform the last integral into a sum of integrals for all the boundary surfaces $\gamma$:

$$m_i \ddot{\mathbf{x}}_i = A_i \mathbf{b}_0 + \sum_{\gamma} \int_{S_{\gamma}^e} \mathbf{P}\mathbf{N} \, dS \tag{5}$$

where $m_i = A_i \rho_0$ is the nodal mass. Equation (5) is very close to the final governing equation, as the second term represents the external forces applied to every node. But the integral can still be separated into traction and elastic forces. We can do that due to the boundary conditions $\mathbf{t} = \mathbf{P}\mathbf{N}$, where $\mathbf{t}$ is a known traction field, applied on $S_{\gamma}^e$. On the free boundary, the traction is null, so $\int_{S_{\gamma}^e} \mathbf{P}\mathbf{N} \, dS = 0$, and on the non-free boundary we have a prescribed constant traction, so $\int_{S_{\gamma}^e} \mathbf{P}\mathbf{N} \, dS = \mathbf{t} l_{\gamma}^e$, where $l_{\gamma}^e$ is the length of the piecewise linear boundary $S_{\gamma}^e$. All of this means we can restrict the boundary summation to internal boundaries only. This separation of the forces yields:

$$m_i \ddot{\mathbf{x}}_i = f_i^{ext} + f_i^t + \sum_{\gamma} \int_{S_{\gamma}^e} \mathbf{P}\mathbf{N} \, dS \tag{6}$$

where $f_i^t = \sum_{\gamma} \mathbf{t} l_{\gamma}^e$ is the traction force applied per node.

Although equation (6) is a lot more discrete than the original problem, we still have to deal with the computation of the elastic forces represented by the last summation of the integral terms in the inner boundaries. To solve it, a clever approach to the integral is used, where instead of plainly applying the midpoint rule to $\int_{S_{\gamma}^e} \mathbf{P}\mathbf{N} \, dS$, we first note that a closed surface integral over a constant tensor field is always zero, so for the $e^{th}$ triangle, we can create an imaginary closed surface integral of the form $\int_{S_{ij}^e} \mathbf{P}^e\mathbf{N} + \int_{S_{ik}^e} \mathbf{P}^e\mathbf{N} + \int_{S_{\alpha}^e} \mathbf{P}^e\mathbf{N} + \int_{S_{\beta}^e} \mathbf{P}^e\mathbf{N} = 0$. Since we don't have the boundaries $\alpha, \beta$ stored on the control volume data structure, but we do have the triangle indices, we replace the sum of the surface integrals $S_{\alpha}^e$ and $S_{\beta}^e$ with $\int_{S_{\alpha}^e} \mathbf{P}^e\mathbf{N} + \int_{S_{\beta}^e} \mathbf{P}^e\mathbf{N} = -\int_{S_{ij}^e} \mathbf{P}^e\mathbf{N} - \int_{S_{ik}^e} \mathbf{P}^e\mathbf{N}$. We can finally apply the midpoint approximation rule on the substitution above. Combining everything, the final governing equation for the elasticity problem is:

$$m_i \ddot{\mathbf{x}}_i = f_i^{ext} + f_i^t + \sum_{e} f_i^e = f_i^{total} \tag{7}$$

where $f_i^e = -\frac{1}{2}\mathbf{P}^e\mathbf{N}_{ji}^e \mathbf{l}_{ji} - \frac{1}{2}\mathbf{P}^e\mathbf{N}_{ki}^e \mathbf{l}_{ki}$, $\mathbf{N}_{ki}^e$ is the normal vector for the triangle side from vertex i to k and $\mathbf{l}_{ki}$ is the length of the same side of triangle $e$.

Now that the governing equation is well defined, it is necessary to establish how the updating of the coordinates and velocity of the nodes is going to be.

Since we know that $\mathbf{v}_i = \dot{\mathbf{x}}_i$, we can define two ordinary differential equations $\dot{\mathbf{v}}_i = \frac{1}{m_i} f_i^{total}$ and $\dot{\mathbf{x}}_i = \mathbf{v}_i$, we can apply a first order finite difference approximation to both and obtain the velocity and position updates in (8) and (9).

$$\mathbf{v}_i^{t+\Delta t} = \mathbf{v}_i^t + \frac{\Delta t}{m+i} f_i^{total} \tag{8}$$

$$\mathbf{x}_i^{t+\Delta t} = \mathbf{x}_i^t + \Delta t \mathbf{v}_i^{t+\Delta t} \tag{9}$$

Now that all the theory has been explained, we finally have a step by step for the implementation of the numerical method. For one simulation step we do the following:

1. Compute deformation gradients $\mathbf{F}^e$ for all triangle elements e

2. Compute Green strain tensors $\mathbf{E}^e$ for all triangle elements e
3. Compute 2nd Piola-Kirchhoff stress tensors $\mathbf{S}^e$ for all triangle elements e
4. Compute 1st Piola-Kirchhoff stress tensors $\mathbf{P}^e$ for all triangle elements
5. Compute elastic forces $\sum_e f_i^e$ for all control volumes i
6. Compute total forces $f_i^{total}$ for all control volumes i
7. Compute velocity update $\mathbf{v}_i^{t+\Delta t}$ for all control volumes i
8. Compute position update $\mathbf{x}_i^{t+\Delta t}$ for all control volumes i
9. Increment time $t = t + \Delta t$

With the implementation ready, it is possible to experiment on our 2D hyper elastic materials simulator.

# Experiments

Before moving into experiments it is important to define the parameters we'll be setting for all of them. The following list shows all inputs to the simulator and what they represent for the problem.

- The triangle mesh: either given or generated with DistMesh;
- The surface traction field $\mathbf{t}$
- The material density field $\rho_0$
- The body force density field $\mathbf{b}_0$, defined by the gravity times the material density field;
- The Lamé parameters $\lambda$ and $\mu$, which define the elasticity of the material being worked on. These parameters are defined by formulas based on two input values: the Young's modulus ($E$) and the Poisson's ratio ($\nu$);
- The time step size $\Delta t$, which defines how fast a simulation iteration is.

## Different materials

The first experiment that we'll do is using the Lamé parameters for rubber and applying traction forces on the right hand side, and the Dirichlet conditions fixing the left hand side of the bar, so we can see the oscillation of the elastic movement. The parameters used for this experiment were: the given mesh from `data.mat`, $\mathbf{t} = [0, -10^5]$, $\rho_0 = 1050$, $\mathbf{b}_0 = [0, -9.82 * rho_0]$, $E = 0.01e9$, $\nu = 0.48$ and $\Delta t = 10^4/E = 1.0000e - 03$.

Since rubber is a very elastic material, we expect that this traction force will be sufficient to pull the bar down and that the elastic forces will counteract the traction and pull the bar up again. Figures 1 to 4 show some stills from the simulation. The original solid bar is shown in red and the deformation is shown in blue.

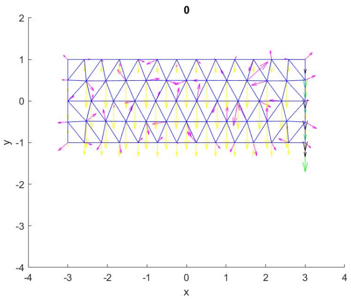



Figure 1: State of rubber bar at t = 0s

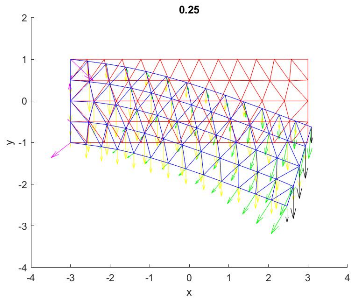



Figure 2: State of rubber bar at t = 0.25s

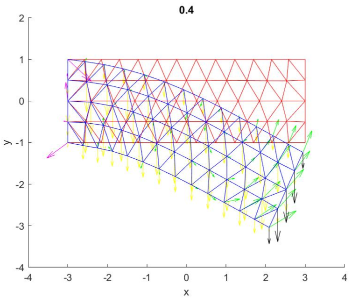



Figure 3: State of rubber bar at t = 0.4s

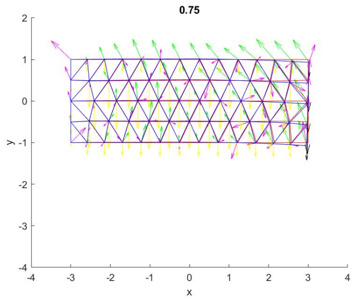



Figure 4: State of rubber bar at t = 0.75s

As expected, we see the movement of the bar bending down due to the traction, and then going back up again, due to the action of the elastic forces on the nodes. The arrows here represent the different forces being applied. The yellow arrows are the external (gravitational) forces, the pink are the elastic forces, the black are the traction forces and the green are the velocity vectors for each node.

Now we change our parameter values to represent steel, a much less elastic material. If we use the same time step and traction as its being used for the rubber, we won't see any deformation in the solid, which is why, to get the same elastic behavior as we did for the previous experiments we use a much faster simulation with a much faster traction force to try and observe the same elastic properties displayed by rubber.

The parameters used for this experiment were: the given mesh from `data.mat`, $\mathbf{t} = [0, -10^9]$, $\rho_0 = 7800$, $\mathbf{b}_0 = [0, -9.82 * rho_0]$, $E = 210e9$, $\nu = 0.31$ and $\Delta t = 10^7/E = 4.7619e-05$.

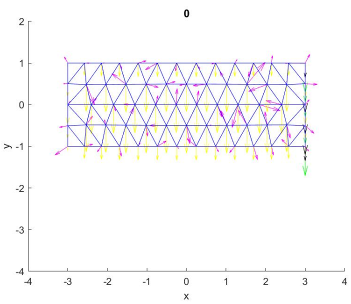



Figure 5: State of steel bar at t = 0s

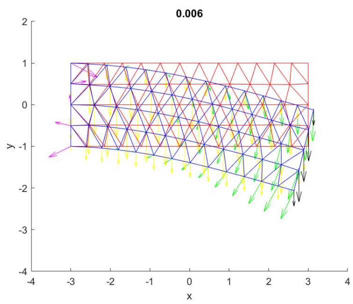



Figure 6: State of steel bar at t = 0.006s

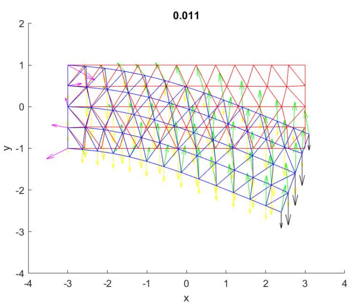



Figure 7: State of steel bar at t = 0.011s

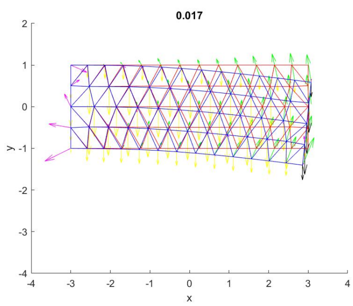



Figure 8: State of steel bar at t = 0.017s

We have a smaller time step but a much bigger force load being applied on this bar, which is why the elastic behavior of the oscillation observed can be reproduced with steel.

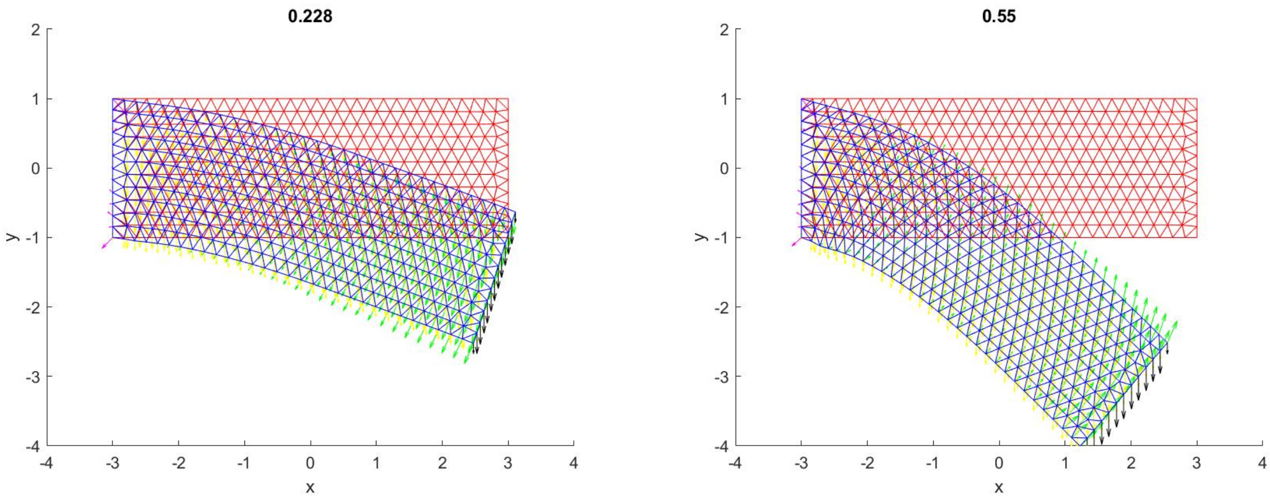
Besides rubber and steel, the Lamé parameters can be modeled to reproduce a variety of different materials. Figure 9, found in a website from a Stanford [1] course, illustrates various elasticity parameters for different rocks.

| Mineral | Density | Young's Modulus | Bulk Modulus | Shear Modulus | Vp | Vs | Poisson's Ratio |
|---|---|---|---|---|---|---|---|
| Quartz | 2.6500 | 95.756 | 36.600 | 45.000 | 6.0376 | 4.1208 | 0.063953 |
| Calcite | 2.7100 | 84.293 | 76.800 | 32.000 | 6.6395 | 3.4363 | 0.31707 |
| Dolomite | 2.8700 | 116.57 | 94.900 | 45.000 | 7.3465 | 3.9597 | 0.29527 |
| Clay (kaolinite) | 1.5800 | 3.2034 | 1.5000 | 1.4000 | 1.4597 | 0.94132 | 0.14407 |
| Muscovite | 2.7900 | 100.84 | 61.500 | 41.100 | 6.4563 | 3.8381 | 0.22673 |
| Feldspar (Albite) | 2.6300 | 69.010 | 75.600 | 25.600 | 6.4594 | 3.1199 | 0.34786 |
| Halite | 2.1600 | 37.242 | 24.800 | 14.900 | 4.5474 | 2.6264 | 0.24972 |
| Anhydrite | 2.9800 | 74.431 | 56.100 | 29.100 | 5.6432 | 3.1249 | 0.27888 |
| Pyrite | 4.9300 | 305.85 | 147.40 | 132.50 | 8.1076 | 5.1842 | 0.15417 |
| Siderite | 3.9600 | 134.51 | 123.70 | 51.000 | 6.9576 | 3.5887 | 0.31876 |
| gas | 0.00065000 | 0.0000 | 0.00013000 | 0.0000 | 0.44721 | 0.0000 | 0.50000 |
| water | 1.0000 | 0.0000 | 2.2500 | 0.0000 | 1.5000 | 0.0000 | 0.50000 |
| oil | 0.80000 | 0.0000 | 1.0200 | 0.0000 | 1.1292 | 0.0000 | 0.50000 |

Figure 9: Elasticity parameters for different materials

## Different meshes

Another experience is to try and use finer or larger meshes, and seeing what parameter values, such as traction force and time step, have to be changed in order to observe the elastic behavior seen above. For a DistMesh mesh of size 0.2, the initial parameters of the rubber break the simulator, so we try to make the time step smaller: $\Delta t = 4000/E$ (this was the closest value to the original time step that did not break the simulator). The elastic behavior is reproduced, but the oscillation takes a lot longer to be simulated for this finer mesh.

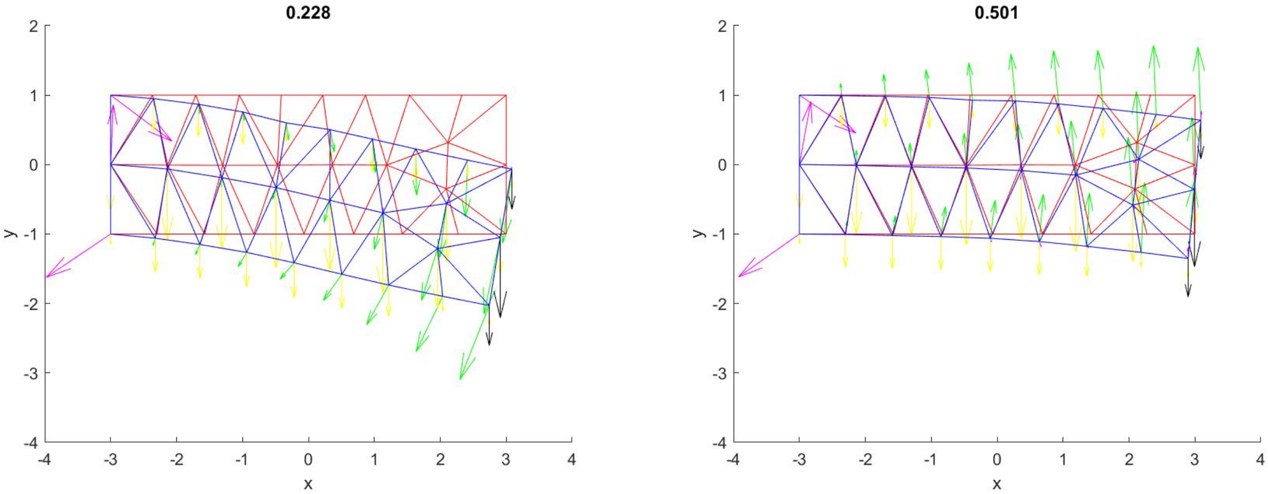



(a) Rubber bar on finer mesh going down at t = 0.228s

(b) Rubber bar on finer mesh going up at t = 0.55s

Figure 10: Finer mesh (0.2) with rubber properties

Repeating the experiment above, we will try to use the initial rubber parameters used for the first experiment and see if it breaks the simulator. Trying the parameters for a DistMesh of size 0.7 we see that the simulator does not break, and the oscillation seems to be much faster than for the finer meshes. Figure 11 shows two stills of the oscillation.




(a) Rubber bar on rougher mesh going down at t = 0.228s

(b) Rubber bar on finer mesh going up at t = 0.501s

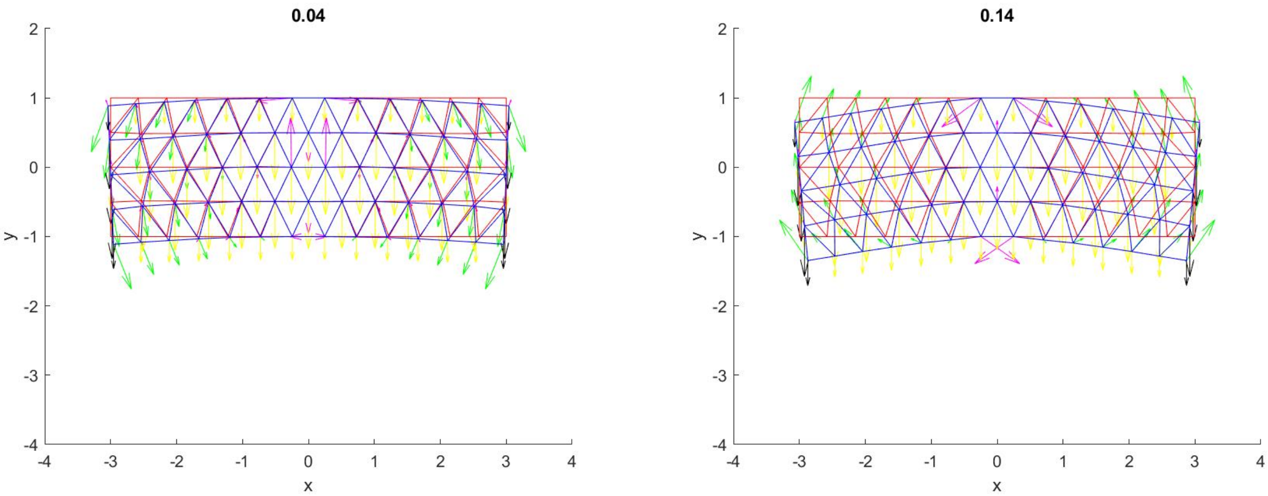
Figure 11: Rougher mesh (0.7) with rubber properties

From these experiments it is possible to affirm that finer meshes take longer to complete one oscillation of the elastic movement, even when all the parameters are very similar or equal between simulations. In similar time intervals, the finer mesh is still very bent down, while the rougher mesh has almost no deformation left in the solid. It is possible to assume that this behavior would be able to be reproduced for other materials besides rubber.

## Dirichlet Conditions

For all the experiments mentioned above, the Dirichlet conditions have only been applied to the left border of our solid. Another experiment that is possible to simulate is applying the conditions elsewhere and seeing if the same oscillating elastic behavior keeps happening for different materials. For the next batch of experiments we apply the Dirichlet conditions in the center of the bar (fixed from -0.5 to 0.5), and apply traction forces to both corners.

For the rubber parameters, we keep the traction force of $\mathbf{t} = [0, -10^5]$, but apply it on both sides of the bar. We also keep the time step of $\Delta t = 1e4/E$. The deformation is illustrated in Figure 12.



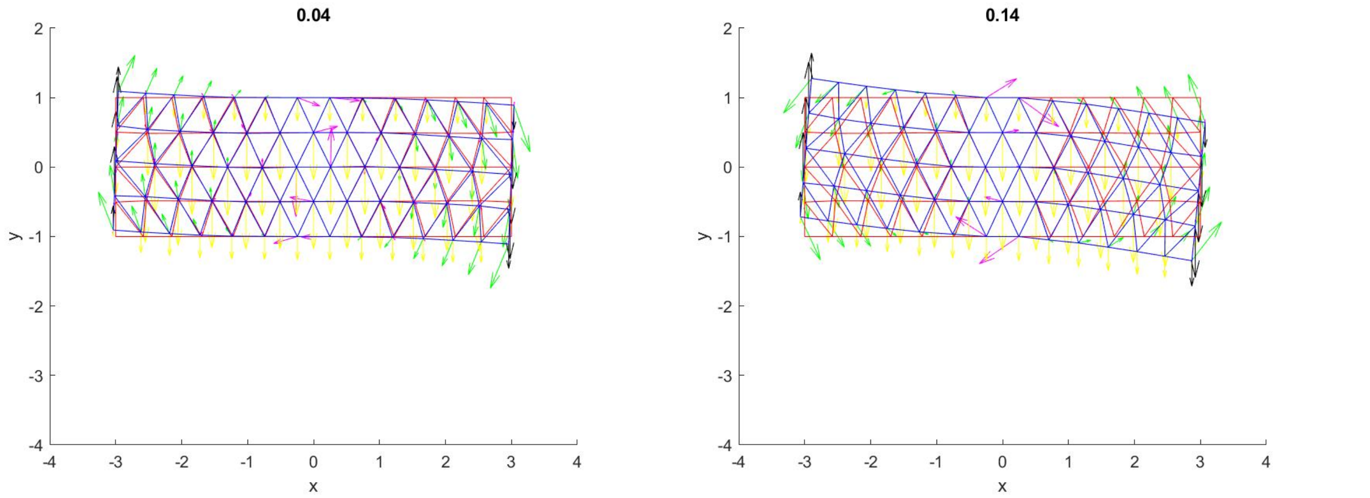

(a) Rubber bar on original mesh with traction applied to both sides at t = 0.04s

(b) Rubber bar on original mesh with traction applied to both sides at t = 0.14s

Figure 12: Original rubber mesh with traction on both ends and same directions

The solid is very quick to deform, but witch traction being applied on both sides, it definitely bends less than if force was applied on one side only, which is in accordance with real-world physics of the momentum of a force around a fixed axis.

Another possible simulation is continuing to apply traction on both sides, but make them applied in different directions. This way we can see how forces of the same intensity, but different orientations affect the elastic behavior of the material. We repeat the parameters for the experiment above, but just change one of the traction forces to be $\mathbf{t} = [0, 10^5]$.



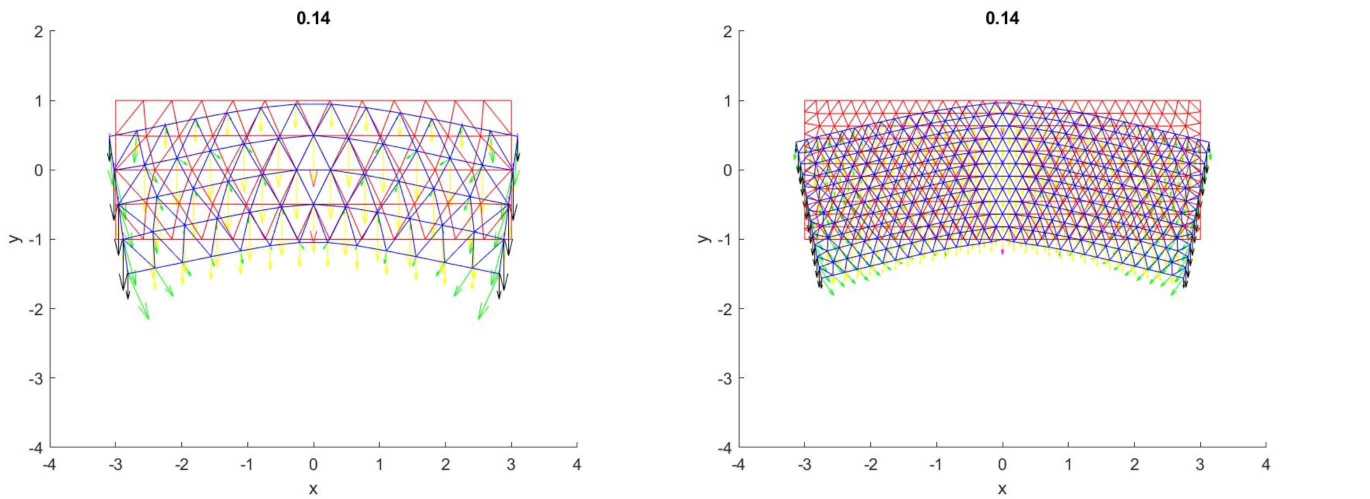

(a) Rubber bar on original mesh with traction applied to both sides in different directions at t = 0.04s

(b) Rubber bar on original mesh with traction applied to both sides in different directions at t = 0.14s

Figure 13: Original rubber mesh with traction on both ends and different directions

For the same stills in time, the deformation looks to be the same for Figures 12 and 13. This means that the orientation of the force does not change the behavior of the deformation if the fixed part is big enough. With that being said, the last experiment is going to be changing the Dirichlet conditions in the center to make the fixed part between -0.05 and 0.05, and seeing if this affects how much the bar bends. Hypothetically, since we would have more of a length to apply force on, the expectation is that we would see more bending.

Since with previous experiments we have seen that a finer mesh deforms more, the new Dirichlet conditions have been applied to the original mesh in `data.mat`, and to a finer mesh generated with DistMesh with size 0.2. Figure 14 shows the moment where both simulations are close to the maximum deformation.




(a) Rubber bar on original mesh with traction applied to both sides in same at t = 0.14s

(b) Rubber bar on fine DistMesh mesh (0.2) with traction applied to both sides at t = 0.14s

Figure 14: Deformation comparison between meshes

As it is illustrated in Figure 14, the original mesh does seem to bend a bit more than it did for a bigger area fixed in the center. But between the mesh resolutions the difference in deformation is not that significant as it was when the traction was applied to the right border of the solid.

## Conclusion

The 2D hyper elastic simulator is very good for conducting a multitude of experiments, as it has a lot of parameters that cam be altered to represent different materials, forces, and even solid shapes. The implementation seems to work reasonably well when tested with various materials, meshed and forces, which can be seen in the experiments in the sections above.

## References

[1] *Stanford Rock Physics Laboratory*
`https://pangea.stanford.edu/courses/gp262/Notes/5.Elasticity.pdf`