

Laboratory exercise no. 6: Simulation of stress and strain distribution using finite element method

Piotr Gawryś < *pgawrys2@gmail.com* >

May 16, 2018

1 Introduction

The goal of this laboratory is to implement simulation of stress and strain distribution using finite element method. In contrary to earlier classes the simulation is not written entirely in code but it also takes advantage of MATLAB's Partial Differential Equation Toolbox.

1.1 Mathematical Model

Although the main goal is to conduct simulation without implementing equation manually we still want to calculate it and compare results with theoretical model. If the simulation object is cantilevered beam under a load then it will be deformed by a value h which can be calculated using formula below:

$$h = \frac{F \cdot L^3}{3E \cdot J} \quad (1)$$

Where:

- $F = 1000$ N – loading force.
- $L = 1.5$ m – beam length.
- $E = 1.8 \cdot 10^8$ Pa – Young's modulus.
- J – moment of inertia of plane area.

$$J = \frac{g \cdot d^3}{12} \quad (2)$$

Where:

- $g = 1 \text{ m}$ – beam thickness.
- $d = 0.2 \text{ m}$ – beam width.

2 Simulation

The modelled object is cantilevered beam supported from the left side. In this simulation we will explore situation where some force is applied at the end (right side) going downwards.

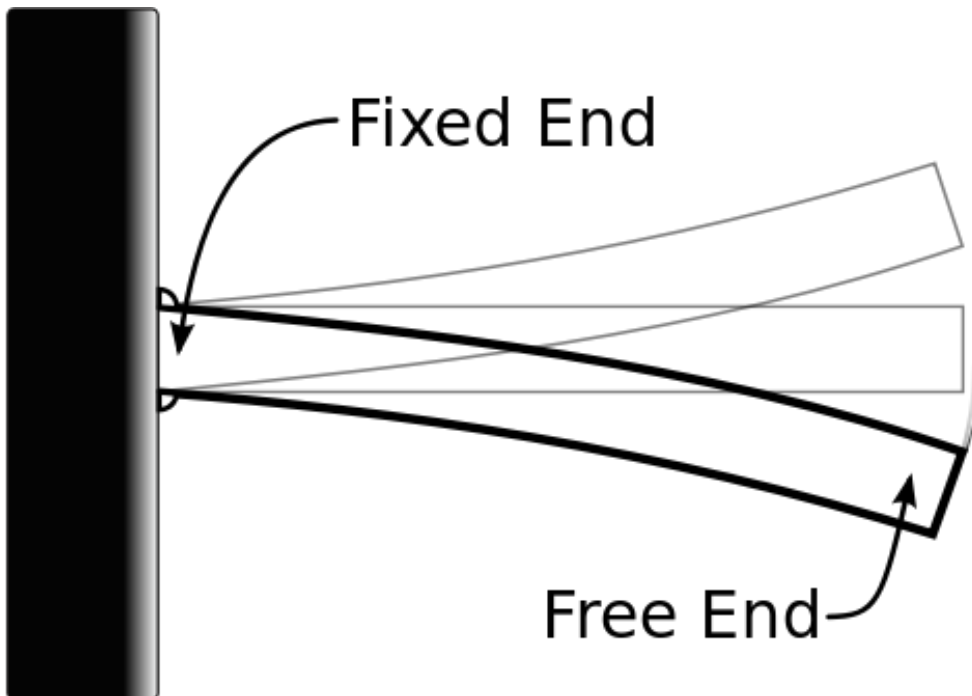


Figure 1: Cantilever Beam.

Source: https://commons.wikimedia.org/wiki/File:Cantilever_Beam.svg

2.1 PDE Toolbox

To implement simulation we will use **pdetool** (included in Partial Differential Equation Toolbox) which provides an interactive for solving 2-D geometry problems. Running it is as simple as writing *pdetool* in MATLAB's Command Window.

2.2 Drawing the object

After turning on Options \rightarrow Grid and selecting Draw \rightarrow Rectangle/square we can draw simulated object using parameters described in the previous section which are 0.2 m for width and 1.5 m for length. This is reflected in the rectangle's dimensions.

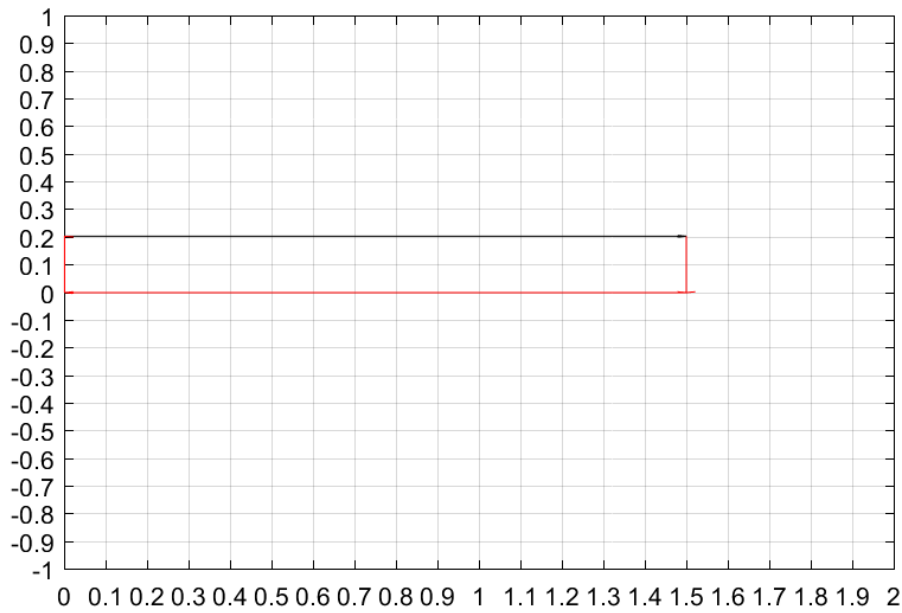


Figure 2: Cantilever Beam drew using MATLAB's PDE Toolbox in boundary mode.

2.3 Setting boundary conditions

To correctly define model we need to include boundary conditions taking into account beam support (left side) and force load (top right side going downwards).

First of all we need to choose correct type of PDE application in the drop down at the top – for our uses *Structural Mech*, *Plane Strain* works fine. Next, entering boundary mode is done choosing Boundary → Boundary Mode or clicking Ctrl + B with default setup. Clicking any side twice causes Boundary Condition window to pop up. We are using following parameters:

1. Left – $h11 = 1$, $h12 = 0$, $h21 = 0$, $h22 = 1$
2. Top – $h11 = 0$, $h12 = 0$, $h21 = 0$, $h22 = 0$
3. Right – $g1 = 0$, $g2 = -1000$ (with Neumann condition)
4. Bottom – $h11 = 0$, $h12 = 0$, $h21 = 0$, $h22 = 0$

Note that values for **Left** indicate that this side won't move under pressure (since it is supported).

$g2 = -1000$ indicates 1000 N applied downwards.

2.4 Coefficients in the equation

After setting boundary conditions we should set coefficients in PDE → PDE Specification. According to <https://www.engineeringtoolbox.com>, Young's modulus for Stainless Steel is about $1.8E8$ Pa, Poisson's ratio is 0.305 and density is between $7480 - 8000 \frac{kg}{m^3}$. For the purpose of the simulation middle value was chosen ($7740 \frac{kg}{m^3}$).

The image shows a software window titled "PDE Specification". At the top, it says "Equation: Structural mechanics, plane strain". Below this is a table with four columns: "Type of PDE:", "Coefficient", "Value", and "Description".

Type of PDE:	Coefficient	Value	Description
<input checked="" type="radio"/> Elliptic	E	1.8E8	Young's modulus
<input type="radio"/> Parabolic	nu	0.305	Poisson ratio
<input type="radio"/> Hyperbolic	Kx	0.0	Volume force, x-direction
<input type="radio"/> Eigenmodes	Ky	0.0	Volume force, y-direction
	rho	7740	Density

At the bottom of the window are two buttons: "OK" and "Cancel".

Figure 3: PDE specification with entered variables.

2.5 Visualization

Final steps are to calculate and visualize simulation. Mesh → Mesh Mode and then Mesh → Refine Mesh followed by Solve → Solve PDE is all it takes to prepare model for visualization.

In Plot → Parameters... there are various options for plotting the result. Most interesting for this use case is changing **Property** to **y displacement (v)**. The plot is shown in the figure 4.

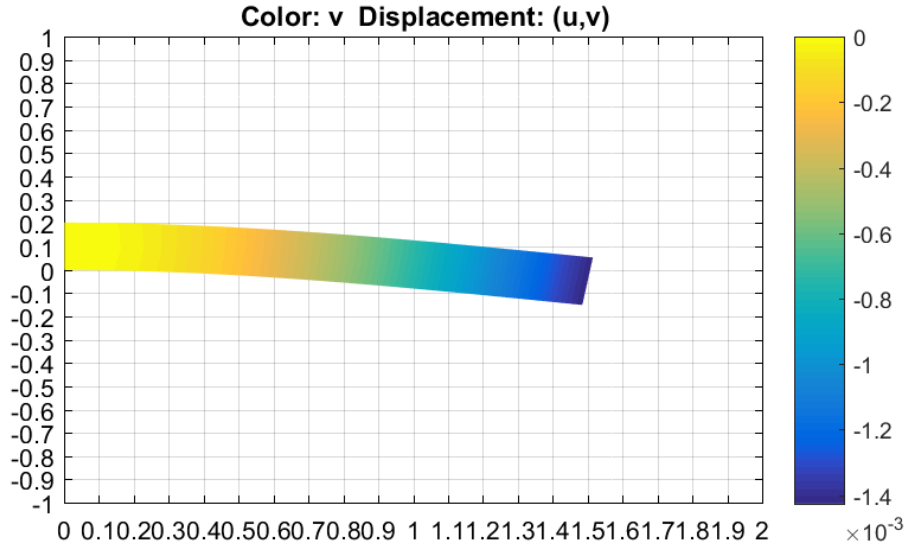


Figure 4: Cantilever Beam after 1000 N pressure from the right side visualized using PDE Toolbox.

We don't observe any deformation on the left side because beam is attached but there is a significant deformation closer to the point of applied force.

It is possible to know the exact values of deformation by exporting solution (Solve → Export Solution) to MATLAB. It returns an array with -0.0014 as lowest value which is consistent with the plot (dark blue color is -0.0014). That's a good point to compare with manual approach.

2.6 Comparison to theoretical model

Relevant equations were introduced in section 1.1. We could either calculate it by hand or using MATLAB, implementation is straightforward:

```
F = -1000; % kiadubg force
```

```

L = 1.5; % beam length
E = 1.8 * 10^8; % Young's modulus for stainless steel
g = 1; % beam thickness
d = 0.2; % beam width

J = (g * d^3) / 12;
h = (F * L^3) / (3 * E * J)

```

The value **h** at the very end (right side) is **-0.0094** which means almost 7 times higher deformation than result obtained from using PDE Toolbox (**-0.0014**). It stays within the same order of magnitude but it's noticeable difference nonetheless. Perhaps it is caused by more sophisticated machinery behind PDE Modeler (due to possibility of taking into account additional parameters about stainless steel such as density which were not present in used equation) leading to more accurate result or just human error.

2.7 More complicated geometry

Cantilever beam is just a rectangle which might not be convincing use case for PDE Toolbox since it can be calculated by hand without too much effort. Fortunately, nothing is stopping us from drawing much more complicated shapes.

Figures 5 present schematics and solved plot for a bit more complicated shape attached in two places, less regular width and different forces.

Implementing this in code would be more educating process but also much slower – entire process of this simulation took minutes.

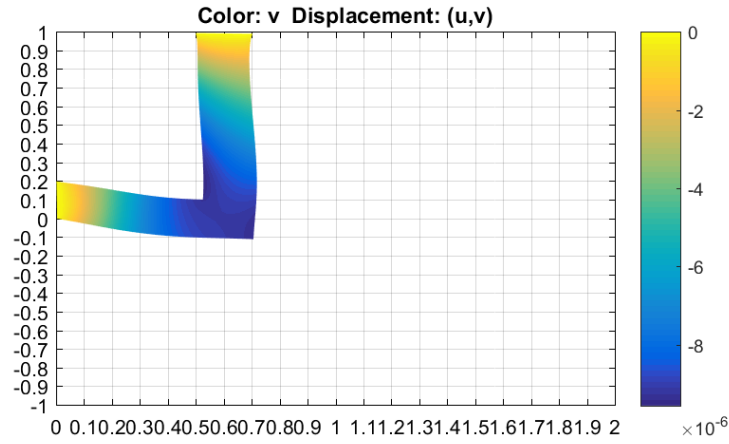
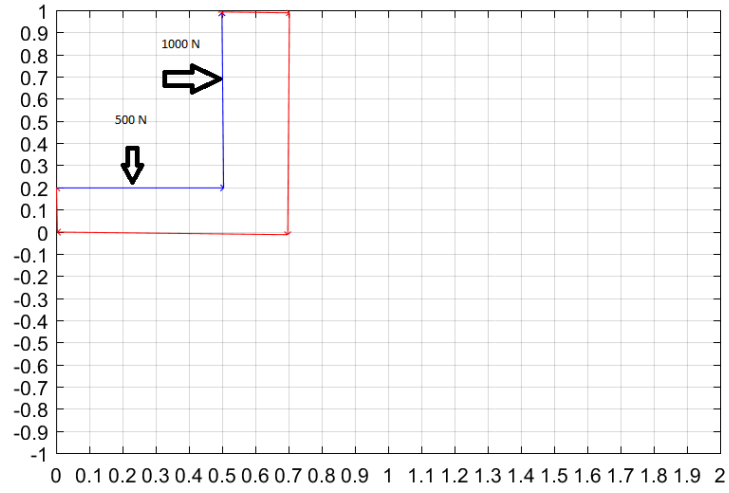


Figure 5: L-shaped stainless steel Beam attached from the left and top with 1000 N applied at left side of the top arm and 500 N on top side of the bottom arm.

3 Conclusion

Over the course of this classes we learned how to simulate stress and strain simulation using PDE Toolbox. Although manual calculations can give us better understanding and confidence about given model the PDE Toolbox is very easy to use and allows for fast prototyping with nice visualization.

I can easily imagine very complicated shapes which are insanely hard to implement and maintain using pure code so I would say using such toolbox scales better for more complicated problems and whenever we have doubts about precision of modelling simulation we can work with generated code to search for issues.