

# CMIS Hand-in 5: Finite Element Method 2

Nikolai Plambeck Nielsen

lpk331@alumni.ku.dk

Niels Bohr Institute, University of Copenhagen

## 1 INTRODUCTION

In this hand-in we focus on solving a linear, elastic deformation for a system, using the finite element method. The governing equation for the system is the Cauchy momentum equation:

$$\rho \ddot{\mathbf{x}} = \mathbf{b} + \nabla \cdot \sigma, \quad (1)$$

where  $\mathbf{x}$  are the deformed (or spatial) coordinates of the material,  $\rho$  is the mass density of the body,  $\mathbf{b}$  is the different body forces acting on the system, and  $\sigma$  is the Cauchy stress tensor. On the boundary we have  $\sigma \mathbf{n} = \mathbf{t}$ , where  $\mathbf{t}$  is the surface traction.

The deformed coordinates can also be expressed as a function of the deformation field  $\Phi$ , with the undeformed (or material) coordinates being the value of the field at  $t = 0$ :

$$\mathbf{x} = \Phi(\mathbf{X}, t), \quad \mathbf{X} = \Phi(\mathbf{x}, 0). \quad (2)$$

In this case we will focus on a homogeneous rectangular bar in two dimensions, whose left side is adhered to a wall, and whose right side experiences a constant traction over its area (or length, rather). We will also consider the quasistatic problem, instead of the dynamic problem. As such we set out to solve for the value  $\Phi(\mathbf{X}, \infty)$ , where  $\ddot{\mathbf{x}} = 0$ . As such, the spatial coordinates can just be expressed as the sum between the material coordinates and the displacement  $\mathbf{u}$  of the material (which we solve for) due to the traction:

$$\mathbf{x} = \mathbf{X} + \mathbf{u}. \quad (3)$$

Further we neglect all body forces, such as gravity. With this, the governing equation becomes:

$$\nabla \cdot \sigma = 0 \quad (4)$$

Now we perform the regular steps of the finite element method: We multiply the equation by some appropriate trial function  $\mathbf{v}$  and then integrate over the volume of the system:

$$\int_{\Omega} (\nabla \cdot \sigma) \cdot \mathbf{v} \, d\Omega = 0 \quad (5)$$

Next we use the product rule for divergence of tensors to split the integral in two:

$$\int_{\Omega} (\nabla \cdot \sigma) \cdot \mathbf{v} \, d\Omega = \int_{\Omega} \nabla \cdot (\sigma \mathbf{v}) \, d\Omega - \int_{\Omega} \sigma : \nabla \mathbf{v}^T \, d\Omega = 0$$

Using Gauss' theorem for divergence on the first integral gives us:

$$\begin{aligned} \int_{\Omega} \nabla \cdot (\sigma \mathbf{v}) \, d\Omega &= \int_{\partial\Omega} (\sigma \mathbf{v}) \cdot \mathbf{n} \, dS \\ &= \int_{\partial\Omega} \mathbf{v} \cdot (\sigma \mathbf{n}) \, dS = \int_{\partial\Omega} \mathbf{v} \cdot \mathbf{t} \, dS \end{aligned} \quad (6)$$

In the second integral we leverage the fact that  $\sigma$  is symmetric to write:

$$\begin{aligned} \int_{\Omega} \sigma : \nabla \mathbf{v}^T \, d\Omega &= \int_{\Omega} \sigma : \nabla \mathbf{v} \, d\Omega \\ &= \int_{\Omega} \sigma : \frac{1}{2} (\nabla \mathbf{v} + \nabla \mathbf{v}^T) \, d\Omega \end{aligned} \quad (7)$$

Now we choose our trial function to be a virtual displacement  $\delta \mathbf{u}$  of the system. This can be written, as in last week, as the product of our trusty barycentric coordinates  $N^e$  for the triangular elements, and a virtual displacement  $\delta \mathbf{u}^e$  for each element:

$$\mathbf{v} = \delta \mathbf{u} = N^e \delta \mathbf{u}^e$$

where the barycentric coordinates are written as a  $2 \times 6$  matrix and the virtual displacement is a 6-component vector:

$$N^e = [N_i^e I_2 \, N_j^e I_2 \, N_k^e I_2], \quad \delta \mathbf{u}^e = [\delta u_{i,x}^e \, \delta u_{j,x}^e \, \dots \, \delta u_{k,y}^e]$$

With this choice of trial function we recognise the right hand side tensor in eq 7 as the virtual form of the Euler strain tensor  $\delta \varepsilon$ , giving us:

$$\int_{\Omega} \sigma : \frac{1}{2} (\nabla \mathbf{v} + \nabla \mathbf{v}^T) \, d\Omega = \int_{\Omega} \sigma : \delta \varepsilon \, d\Omega$$

Now, since both  $\sigma$  and  $\delta \varepsilon$  are symmetric rank 2 tensors in two dimensions, we might as well write the double contraction between them, as the dot product of two 3-component vectors (where the first two components are their diagonal elements, and the last component is the off diagonal element).

Next we use the fact that strain can be written as a product of a differential operator and a displacement:

$$\varepsilon = S \mathbf{u} = S N^e \mathbf{u}^e = B^e \mathbf{u}^e$$

where  $\mathbf{u}$  is the displacement and  $S$  is the  $3 \times 2$  differential operator

$$S = \begin{pmatrix} \partial_x & 0 \\ 0 & \partial_y \\ \partial_y & \partial_x \end{pmatrix}$$

which makes  $B^e$  a  $3 \times 6$  matrix:

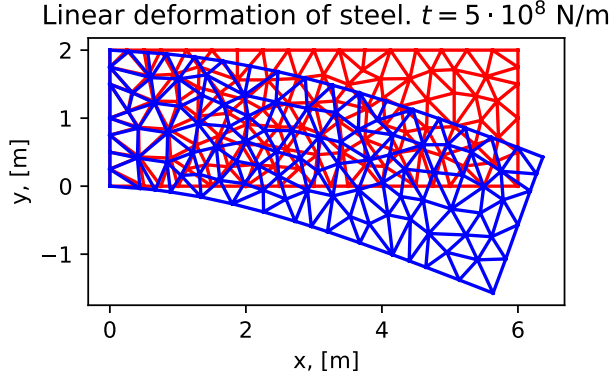
$$B^e = \begin{pmatrix} \partial_x N_i^e & 0 & \partial_x N_j^e & 0 & \partial_x N_k^e & 0 \\ 0 & \partial_y N_i^e & 0 & \partial_y N_j^e & 0 & \partial_y N_k^e \\ \partial_y N_i^e & \partial_x N_i^e & \partial_y N_j^e & \partial_x N_j^e & \partial_y N_k^e & \partial_x N_k^e \end{pmatrix}$$

This of course also applies to the virtual strain:  $\delta \varepsilon = B^e \delta \mathbf{u}^e$ . Likewise, the stress tensor can (in linear elastic deformation) be written as the product between the elasticity matrix and the strain  $\sigma = D \varepsilon$ , with the elasticity matrix for a 2D system being

$$D = \frac{E}{1 - \nu^2} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix}$$

where  $E$  is Youngs modulus and  $\nu$  is the Poisson ratio. Putting everything together we have:

$$\begin{aligned} \int_{\Omega} \sigma : \delta \varepsilon \, d\Omega &= \int_{\Omega} \sigma^T \delta \varepsilon \, d\Omega \\ &= \int_{\Omega} \delta \varepsilon^T \sigma \, d\Omega \\ &= \int_{\Omega} (\delta \mathbf{u}^e)^T (B^e)^T D B^e \mathbf{u}^e \, d\Omega. \end{aligned}$$



**Figure 1: Deformation of a steel bar. Max triangle area is  $0.1 \text{ m}^2$ . Traction  $t = 5 \cdot 10^2$ . The material coordinates is shown in red, whilst the spatial coordinates are shown in blue.**

For a final equation of the form:

$$(\delta \mathbf{u}^e)^T \left( \int_{\Omega} (\mathbf{B}^e)^T \mathbf{D} \mathbf{B}^e \mathbf{u}^e \, d\Omega - \int_{\partial\Omega} (\mathbf{N}^e)^T \mathbf{t} \, dS \right) = 0 \quad (8)$$

which as usual must hold for all virtual displacements. Giving us the familiar FEM equation  $\mathbf{K} \mathbf{u} = \mathbf{f}$ .

The boundary integral  $f$  can be computed for our case of a constant traction  $\mathbf{t}$ . The elements on the boundary of the system are just line segments, so the shape functions are just the hat functions from last week, and the integral over these are just one half times the segment length.

$$\begin{aligned} f &= \int_{\partial\Omega} (\mathbf{N}^e)^T \mathbf{t} \, dS = \sum_{e \in \partial\Omega} f^e, \\ f^e &= \int_{\partial\Omega^e} (\mathbf{N}^e)^T \mathbf{t} \, dS = \frac{1}{2} \ell^e \mathbf{t}. \end{aligned} \quad (9)$$

Likewise we get  $\mathbf{K} = \sum_{e \in \Omega} \mathbf{K}^e$  with

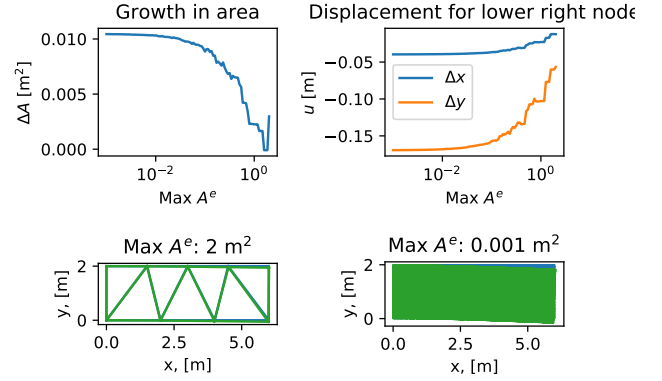
$$\mathbf{K}^e = \int_{\Omega^e} (\mathbf{B}^e)^T \mathbf{D} \mathbf{B}^e \, d\Omega.$$

A thing to note is that this is entirely analogous to the principle of virtual work, which states that the virtual work of a system is equal to 0. The virtual work is in this case the work done by a real force on a virtual displacement.

The real force is just the divergence of the Cauchy stress tensor (or more generally, the force density expressed in the Cauchy momentum equation). Taking the dot product between this and the virtual displacement, and then integrating over the whole system must then equal zero for the principle of virtual work to hold.

## 2 EXPERIMENTS

All experiments for this week will be performed on a 6 by 2 metre rectangular steel bar. Steel has a Young modulus of  $E = 69 \cdot 10^9 \text{ Pa}$  and a Poisson ratio of  $\nu = 0.3$ . The solution for a sample grid with traction  $t = 5 \cdot 10^8 \text{ N/m}$  is seen in figure 1, and sure enough, the



**Figure 2: Growth in area and displacement of the bottom right corner node, both plotted as a function of the maximum triangle area. Note the logarithmic scale on the first axis. Also shown is the deformation of the first and last mesh. Minimum angle argument set to 0, and a constant traction of  $5 \cdot 10^7 \text{ N/m}$  was used for all simulations.**

bar is deformed downwards on the right side, whilst the left stays put - just as expected.

### 2.1 Examining the effects of mesh resolution on the deformation

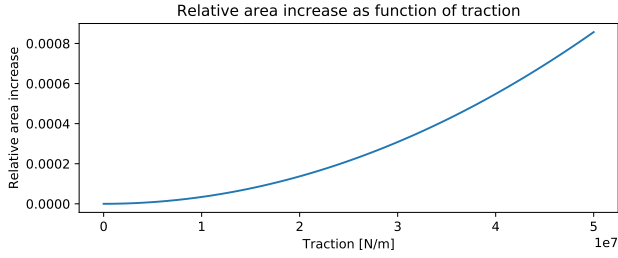
For the first experiment we test the effects of mesh resolution on the deformation of the steel bar. We set a constant traction of  $t = 5 \cdot 10^7 \text{ N/m}$  for a common point of reference between simulations. To vary the mesh resolution we pass a .poly file to Triangle with just the four corners of the bar, and lines between them, but vary the max area parameter. We perform the simulation for 50 logarithmically spaced numbers (base 2) between  $2 \text{ m}^2$  and  $0.001 \text{ m}^2$ . For each simulation we record the difference in area between the initial and final state, along with the displacement for the lower right corner node.

We have chosen these measures for a couple of reasons: We want to evaluate whether an increase in mesh resolution will lead to a convergence upon the “true” value of the deformation. As such, if we want to measure the deformation, we either have to perform take some average over displacements, or just look at a single node, shared between systems. The latter is of course much easier to implement, and well behaved no matter the mesh resolution. Only four nodes are shared amongst the different meshes, namely the corners, two of which are clamped to the wall. We could of course just as well have chosen the top right node, as this would also deform, but the bottom right felt like the “natural” choice.

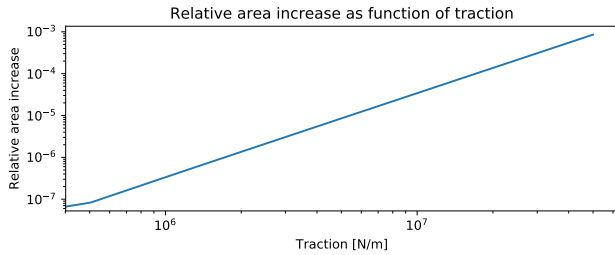
As for the area: It was an easy measure to implement, and sure enough, produces interesting results.

We expect to see convergence for both measures as the max triangle area decreases, as this increases the number of nodes and elements, and will therefore more closely model the real world (this is of course assuming the meshes generated are of sufficient quality)

The results of the experiment can be seen in figure 2. Sure enough, the displacement of the lower right node settles towards some value



**Figure 3: Relative area increase as a function of the traction. A mesh with max triangle area of  $0.01 \text{ m}^2$  is used.**



**Figure 4: Figure 3 but for a log-log scale.**

as the max triangle area decreases. The same thing applies for the area of the bar, though we see a peculiar thing: one might expect the growth in area to end at a low value, not increase! It does however seem to converge upon a value, so it might just be, that this increase in area is precisely what the linear elastic deformation model predicts for a steel bar.

In the figure we have also included plots of the spatial (green) and material (blue) coordinates for the first and last mesh. The increase in displacement for the lower right node is indeed apparent from these plots as well.

## 2.2 Examining the incompressibility of steel

Normally, as stated before, we would not expect steel to increase in volume/area as it experiences stresses, but the model seems to predict this. Next we perform the simulation for a range of different loads: 100 linearly spaced values between 0 and  $5 \times 10^7$ . We also choose a mesh with max triangle area  $0.01 \text{ m}^2$ . We record the area of the bar after each of the simulations, and calculate the relative area increase of the  $i$ 'th simulation as  $(A_i - A_0)/A_0$ . The results are seen in figure 3. We see an increase in area as indicated in the simulation before. It seems to scale like a parabola, and a look at the plot on a log-log scale reveals a strikingly linear relation between the load and relative area increase (see figure 4) Calculating the slope is done by computing the rise over run of the values in the log-log plot:

$$\text{slope} = \frac{\log A_i - \log A_j}{\log t_i - \log t_j} \quad (10)$$

for  $i > j$ . We choose  $i = 100, j = 20$ , giving a slope of the log-log plot of 2.0012. So it is to a good approximation a parabola. Let us see, if we can make sense of it:

All non-zero values in  $f$  are the ones corresponding to nodes that experience traction, and since these values, per eq 9, scale linearly with the traction, we expect the displacement to also scale linearly since

$$Ax = b \Rightarrow x = A^{-1}b, \quad Ax' = 2b \Rightarrow x' = 2A^{-1}b = 2x$$

Now, area scales as the square of length, so a linear increase in the displacement might well induce a quadratic increase in the relative area (which of course is directly proportional to the absolute area). Of course, this is all post-hoc hypotheses based on hand-wavy arguments. To ascertain this one might more reasonably do a mathematical analysis on the increase in area as function of displacement, but I am a physicist, so I prefer doing this experimentally.

To do this experimentally, we need to vary as many different parameters as possible (one at a time, of course) and see if the effect still occurs. If it does, it is a strong hint in the direction of a quadratic relationship between the displacement and final area of the bar.

In this model we have essentially three different parameters to change: The Young modulus of the material, the Poisson ratio, and the resolution of the mesh. The Young modulus appears as a scale factor on the elasticity matrix  $D$ , and as such, the displacement and Young modulus are inversely proportional:

$$Ax = b \Rightarrow x = A^{-1}b, \quad 2Ax' = b \Rightarrow x' = A^{-1}b/2 = x/2$$

This leaves the Poisson ratio and the mesh resolution. We choose 3 different values of the Poisson ratio (0.1, 0.3, 0.5) and 3 different values of the max triangle area (1, 0.1, 0.01), and compute the relative area increase for the same range of tractions, for each set of parameter values. We then plot these on a log-log plot, which is seen in figure 5.

As seen in the figure, the relative area increase does not seem to depend on the Poisson ratio (at least not in this range of Poissons ratio), there is some deviation from the linearity of the log-log plot when varying the max triangle area. But still, for meshes of a decent resolution and up, the relationship between relative area increase and traction seems to be approximately quadratic (for large enough tractions).

Of course, this experiment only samples a small, small portion of the available parameter space, so a finer analysis of the phenomenon is warranted - a larger selection of Poisson ratios and mesh resolutions, and perhaps a proper fitting to a polynomial of some order, to get a better estimate of the relationship.

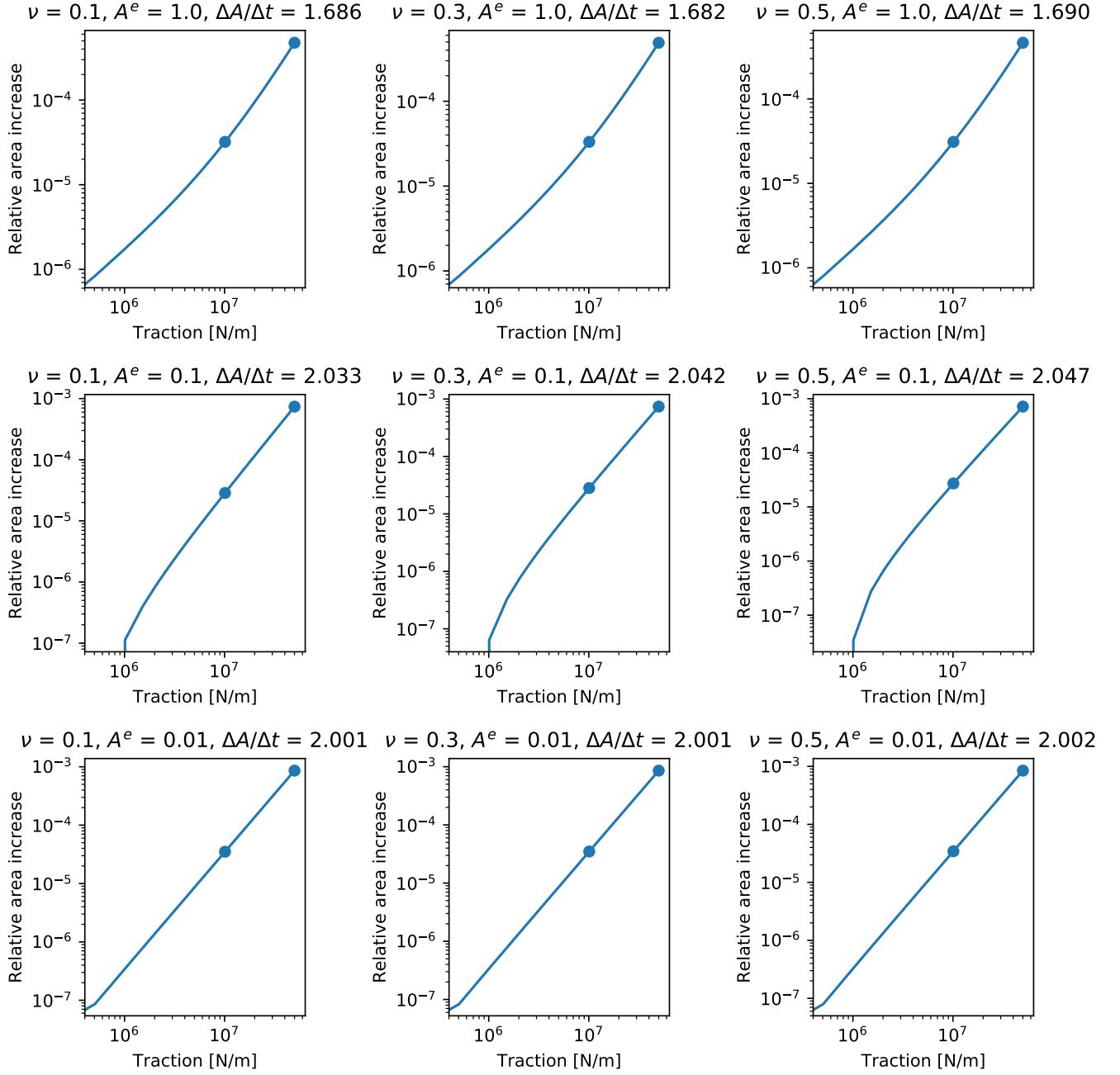


Figure 5: log-log plots of traction and relative area increase, for different values of the Poisson ratio  $\nu$  and max triangle size  $A^e$ . The two points shown are the ones for which the slope is calculated ( $i=100, j=20$  in eq 10).