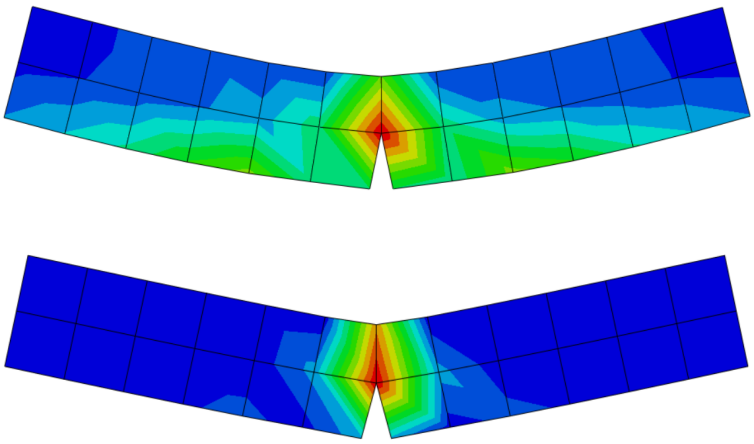
GCE Civil and Environmental Engineering

UCLouvain
École polytechnique de Louvain

ASSIGNMENT

LGCIV2041
Numerical Analysis of Civil Engineering Structures

Professors : João Pacheco de Almeida & Hadrien Rattez
Teaching assistant : Luc Simonin



Group
Foret Félix
Herphelin Louis

Year 2024 -2025

Contents

1 Problem 1

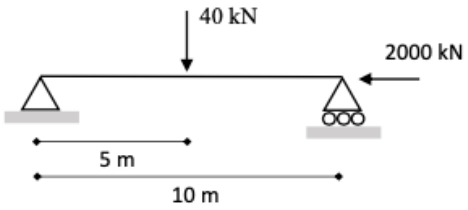


Figure 1.1: Problem analyzed

1.1 Displacement and rotational field

First, we need to determine the transverse displacement $u_{y0}(x)$ and the rotational field $\theta(x)$. To obtain these analytically, we use the constitutive equation and the compatibility equation:

$$M(x) = EI\frac{d\theta}{dx}, \quad \frac{du_{y0}(x)}{dx} = \theta(x).$$

By substituting the compatibility equation into the constitutive equation, we obtain:

$$M(x) = EI\frac{d^2u_{y0}(x)}{dx^2}.$$

Our problem gives us the moment equation:

$$M(x) = \begin{cases} \frac{F}{2}x, & 0 \leq x \leq 5 \\ \frac{F}{2}(L-x), & 5 \leq x \leq 10 \end{cases}$$

where $F = 40$ kN is the applied load at $x = 5\,\text{m}$. Using this moment function, we proceed with the integration to obtain $u_{y0}(x)$.

For $0 \leq x \leq 5$:

$$u_{y0}^{left}(x) = \frac{F}{12EI}x^3 + C_1x + C_2$$

For $5 \leq x \leq 10$:

$$u_{y0}^{right}(x) = \frac{F}{2EI}\left(\frac{Lx^2}{2} - \frac{x^3}{6}\right) + C_3x + C_4$$

Using the four boundary conditions, we can obtain the coefficients:

- Blocked displacement: $u_{y0}^{\text{right}}(L) = 0$ and $u_{y0}^{\text{left}}(0) = 0$,
- Transverse displacement continuity: $u_{y0}^{\text{right}}(L/2) = u_{y0}^{\text{left}}(L/2)$,
- Rotational continuity: $\frac{du_{y0}^{\text{right}}}{dx}(L/2) = \frac{du_{y0}^{\text{left}}}{dx}(L/2)$.

We obtain the folowing system

1. $u_{y0}^{\text{left}}(0) = 0 \quad \Rightarrow \quad C_2 = 0.$

2. $u_{y0}^{\text{right}}(L) = 0 \quad \Rightarrow \quad \dfrac{F}{2EI}\left(5L^2 - \dfrac{L^3}{6}\right) + C_3 L + C_4 = 0.$

3. $u_{y0}^{\text{left}}(L/2) = u_{y0}^{\text{right}}(L/2) \quad \Rightarrow \quad \dfrac{F}{12EI}\left(\dfrac{L^3}{8}\right) + \dfrac{C_1 L}{2} = \dfrac{F}{2EI}\left(5\dfrac{L^2}{4} - \dfrac{(L/2)^3}{6}\right) + \dfrac{C_3 L}{2} + C_4.$

4. $\dfrac{du_{y0}^{\text{left}}}{dx}(L/2) = \dfrac{du_{y0}^{\text{right}}}{dx}(L/2) \quad \Rightarrow \quad \dfrac{3F}{12EI}\left(\dfrac{L^2}{4}\right) + C_1 = \dfrac{F}{2EI}\left(10\dfrac{L}{2} - \dfrac{3(L/2)^2}{6}\right) + C_3.$

The coefficients are as follows:

$$C_1 = \frac{-FL^2}{16EI}, \quad C_2 = 0, \quad C_3 = \frac{-3FL^2}{16EI}, \quad C_4 = \frac{FL^3}{48EI}.$$

We obtain the following displacement field and rotational field :

$$u_{y0}(x) = \begin{cases} \frac{F}{48EI}(4x^3 - 3L^2 x), & 0 \le x \le 5 \\ \frac{F}{48EI}(-4x^3 + 12x^2 L - 9L^2 x + L^3), & 5 \le x \le 10 \end{cases}$$

$$\theta(x) = \begin{cases} \frac{F}{48EI}(12x^2 - 3L^2), & 0 \le x \le 5 \\ \frac{F}{48EI}(-12x^2 + 24xL - 9L^2), & 5 \le x \le 10 \end{cases}$$

We also perform the calculation using the Python code for the Euler-Bernoulli finite element method. We conduct the calculation for two different meshes: one with 2 elements and another with 20 elements. Here are the results for the transverse displacement and the rotational field. We plot on the same graph three results: for 2 elements, for 20 elements, and the exact solution found earlier.

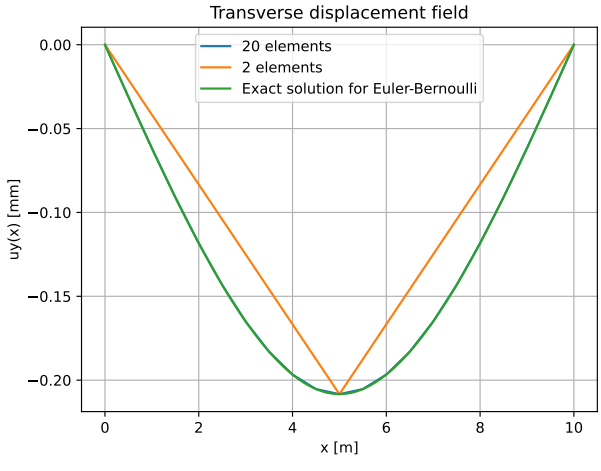


Figure 1.2: Transverse displacement field (FE solutions and exact solution).

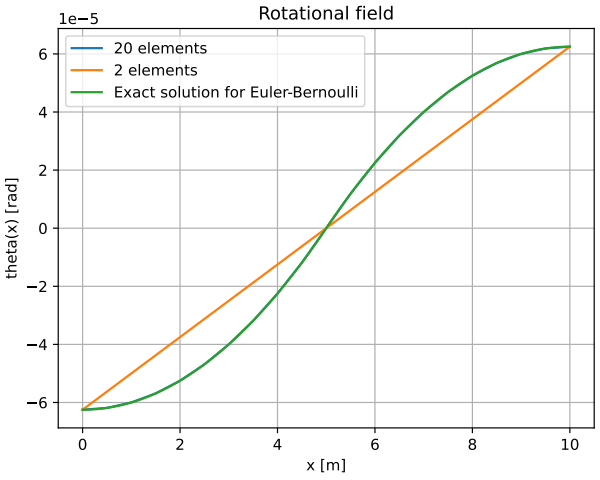


Figure 1.3: Rotational field (FE solutions and exact solution).

We can observe that the more we refine our mesh, the closer we get to the exact solution. In both graphs, we see that the mesh with 2 elements has a significant error between the nodes, whereas with 20 elements, this error decreases. If we zoom in, we can locally observe some small differences even with 20 elements, but they are very minor compared to the case with 2 elements. A solution to further reduce this error is to discretize our beam with more elements. However, we must be careful because increasing the number of elements also increases the computation time. In our exercise, we measured the computation time:

- For 2 elements: 0.0013 seconds
- For 20 elements: 0.0050 seconds

In our case, the computation times are very small because we only analyze a single beam. However, if we have to model a larger structure, such as a building, this time difference would become much more significant.

1.2 Timoshenko finite element

If we now consider the Timoshenko beam theory, which takes into account both shear deformation and flexure, the shear distortion $\gamma(x)$ is nonzero. In other words, the cross-sections do not necessarily remain perpendicular to the beam axis after deformation. Therefore, the constitutive and compatibility equations change and become:

$$M(x) = EI\frac{d\theta}{dx}, \quad V(x) = \kappa GA\gamma(x), \quad \gamma(x) = \frac{du_{y0}(x)}{dx} - \theta(x).$$

Using the moment equation $M(x)$, we can first find $\theta(x)$ by integrating $M(x)$:

$$\theta(x) = \begin{cases} \frac{F}{4EI}x^2 + C_1, & 0 \le x \le 5 \\ \frac{FL}{2EI}x - \frac{F}{4EI}x^2 + C_2, & 5 \le x \le 10 \end{cases}$$

Using $V(x)$, we can determine the shear distortion $\gamma(x)$:

$$V(x) = \begin{cases} -\frac{F}{2}, & 0 \le x \le 5 \\ \frac{F}{2}, & 5 \le x \le 10 \end{cases} \quad \Rightarrow \quad \gamma(x) = \begin{cases} -\frac{F}{2\kappa GA}, & 0 \le x \le 5 \\ \frac{F}{2\kappa GA}, & 5 \le x \le 10 \end{cases}$$

We can now find the transverse displacement $u_{y0}(x)$:

$$u_{y0}(x) = \begin{cases} -\frac{F}{2\kappa GA}x + \frac{F}{12EI}x^3 - C_1x + C_3, & 0 \le x \le 5 \\ \frac{F}{2\kappa GA}x + \frac{FL}{4EI}x^2 - \frac{F}{12EI}x^3 - C_2x + C_4, & 5 \le x \le 10 \end{cases}$$

Using the same boundary conditions as for the Euler-Bernoulli beam, we find the four coefficients:

$$C_1 = \frac{-FL^2}{16EI}, \quad C_2 = \frac{-3FL^2}{16EI}, \quad C_3 = 0, \quad C_4 = -\frac{FL}{2\kappa GA} + \frac{FL^3}{48EI}.$$

Thus, we obtain the following displacement and rotational fields:

$$u_{y0}(x) = \begin{cases} \frac{F}{48EI}(4x^3 - 3L^2x) - \frac{F}{2\kappa GA}x, & 0 \le x \le 5 \\ \frac{F}{48EI}(-4x^3 + 12x^2L - 9L^2x + L^3) + \frac{F}{2\kappa GA}x - \frac{FL}{2\kappa GA}, & 5 \le x \le 10 \end{cases}$$

$$\theta(x) = \begin{cases} \frac{F}{48EI}(12x^2 - 3L^2), & 0 \le x \le 5 \\ \frac{F}{48EI}(-12x^2 + 24xL - 9L^2), & 5 \le x \le 10 \end{cases}$$

As we can see in the figures below, the exact (analytical) solution changes when we include shear deformation. Once again, we run the simulation using the Timoshenko finite element in our Python code. We had to change a few things in the code. During the course, we saw that including shear deformation changes the stiffness matrix of each element (k_{loc}), which also changes the global stiffness matrix of the structure.

Here are the results for the problem:

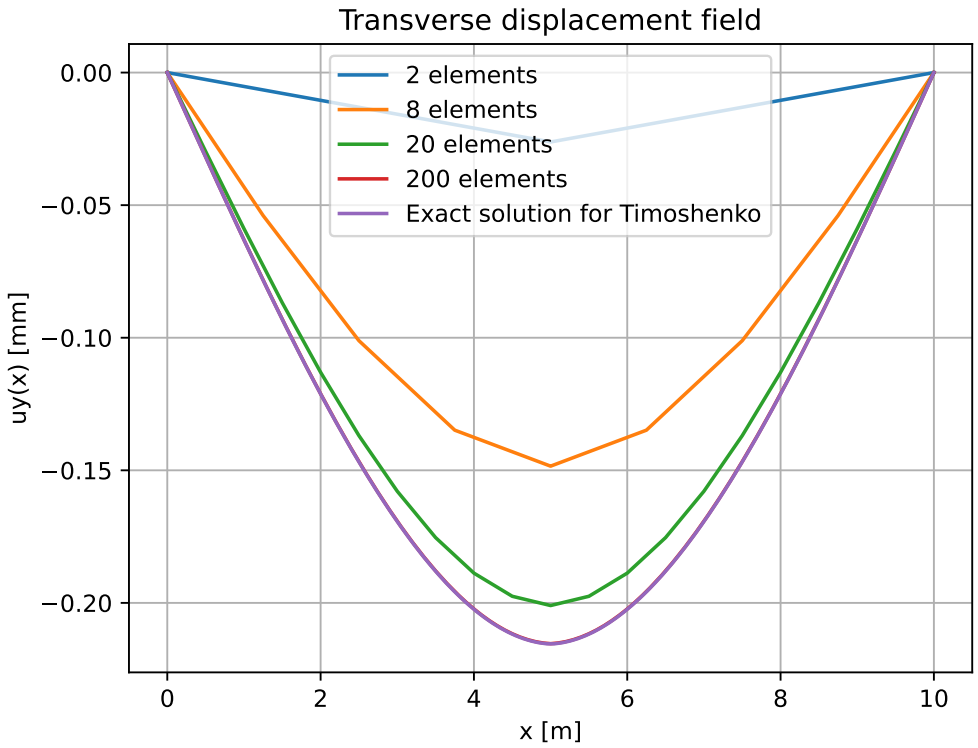


Figure 1.4: Displacement field from the Timoshenko finite element method with 2, 8, 20, and 200 elements, compared to the exact solution.

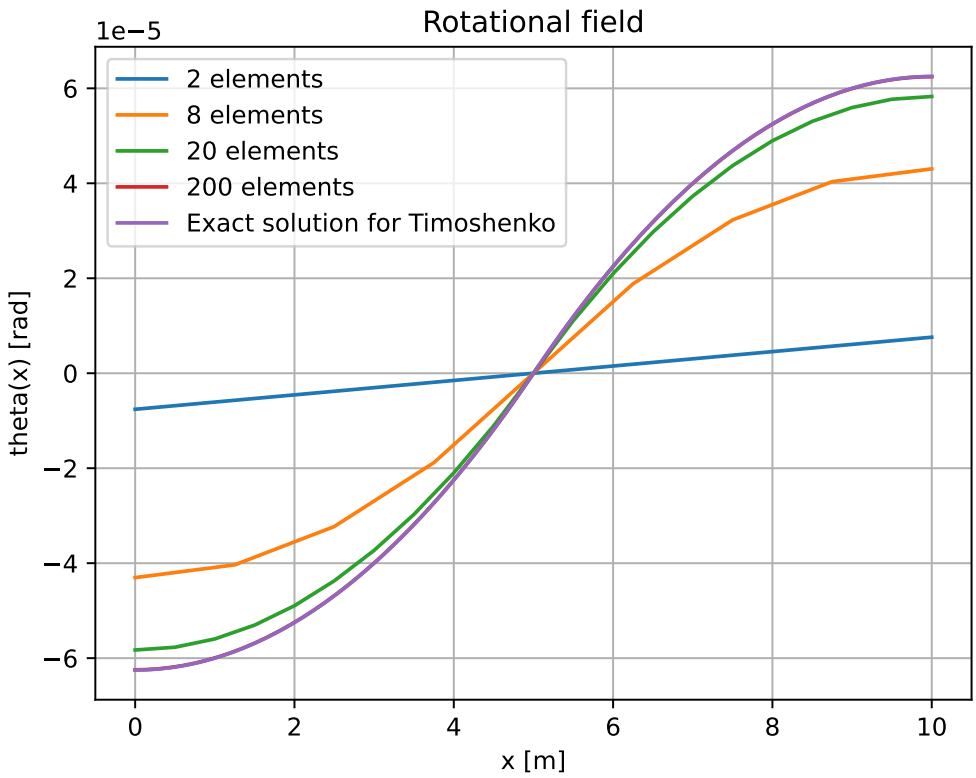


Figure 1.5: Rotation field from the Timoshenko finite element method with 2, 8, 20, and 200 elements, compared to the exact solution.

We notice the same thing as with the Euler-Bernoulli beam: the more we refine the mesh, the closer the solution gets to the exact one. But refining the mesh also makes the code take more time to run.

Looking at the boundary conditions, we can see they are always respected, no matter how many elements we use. The displacement field is also continuous inside each element and between elements (no sudden jumps). This is because, in the weak form, the essential boundary conditions (displacement and slope) are not automatically satisfied — we have to apply them ourselves in the code. That's why they are respected in all the cases we tested. We also observe that more elements are required compared to the Euler-Bernoulli FEs for the solution to converge to the exact one.

1.3 Bending moment and shear force

We now analyze the bending moment and shear force along the beam using a mesh of 8 elements. These internal forces are shown in Figures 1.6 and 1.7.

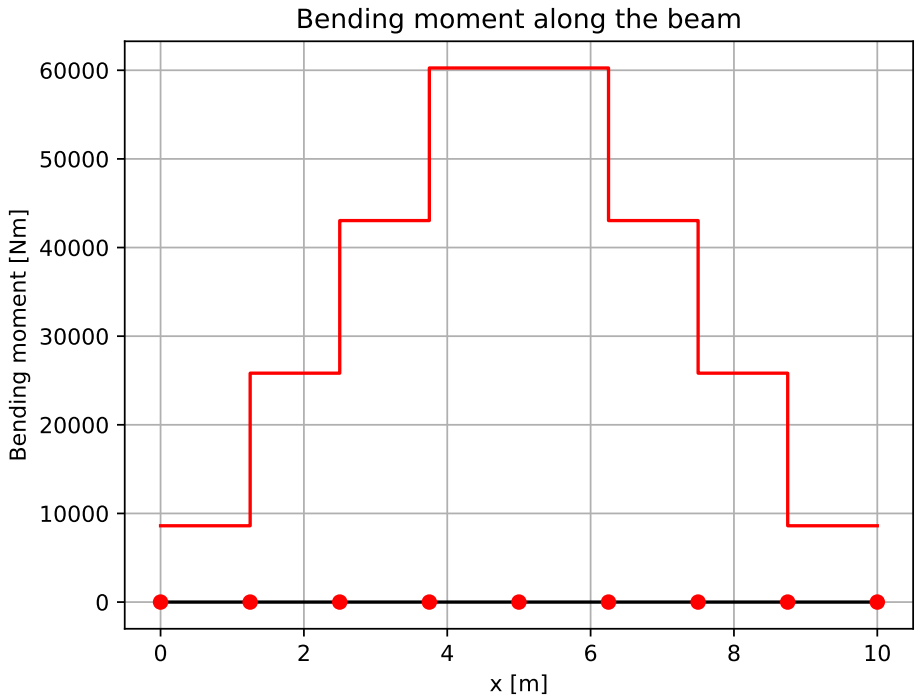


Figure 1.6: Evolution of the bending moment along the beam, for a mesh of 8 Timoshenko finite elements.

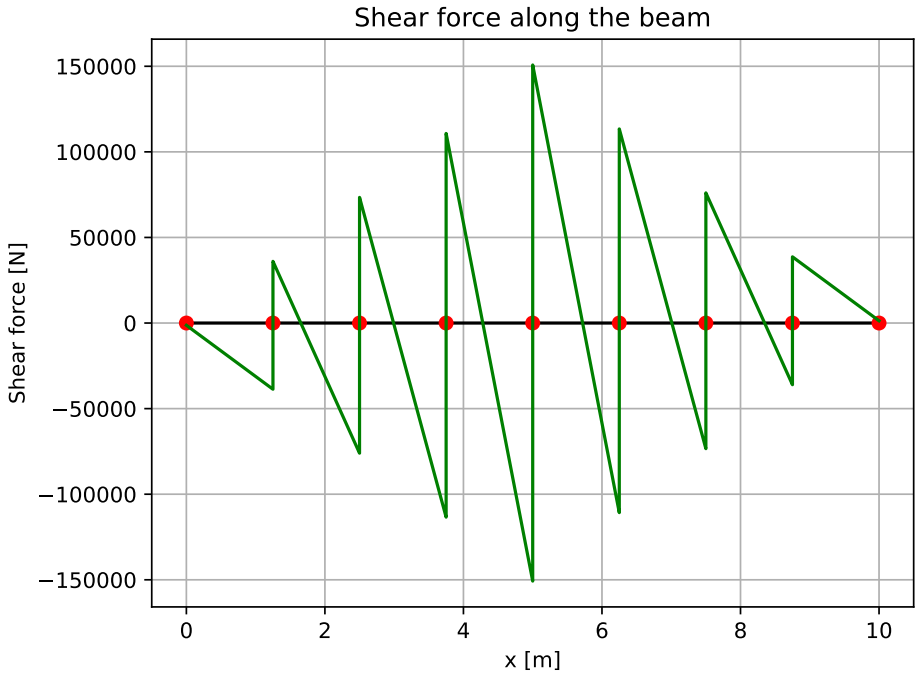


Figure 1.7: Evolution of the shear force along the beam, for a mesh of 8 Timoshenko finite elements.

As we can see, the internal forces are not exactly the same as those calculated analytically in Section 1.1. This is because in the weak form, natural boundary conditions (such as bending moments and shear forces)

are already included. They are part of the integrals and are therefore only satisfied in an average sense, not exactly. So, the equilibrium conditions are satisfied locally on average, but not necessarily at the element interfaces nor at the natural boundaries. This is why we see jumps in the bending moment at each node in Figure 1.6, and also why the bending moment is not zero at the beam ends, even though it should be (at a pinned support). For the shear force (Figure 1.7), the situation is similar. There are also visible jumps between elements, and the shear force at the boundaries does not match the expected theoretical values (should not be zero). This confirms that the shear force is also only satisfied in an average sense, not pointwise at the element interfaces or at the beam ends.

1.4 Beam length variation

We now analyze the vertical displacements by varying the length L of the beam. For this study, we use a mesh of 200 elements to ensure high accuracy.

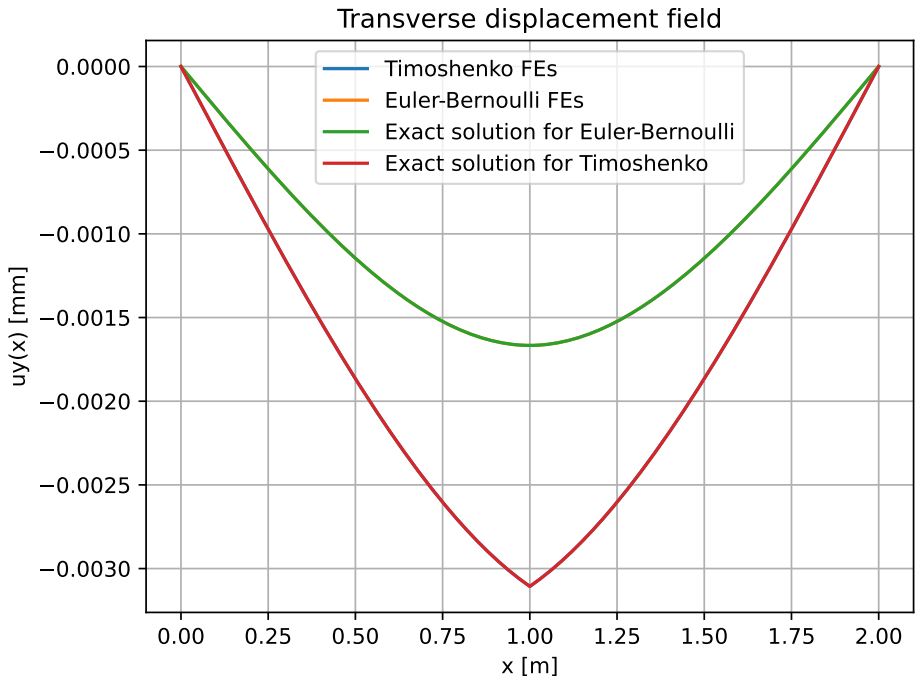


Figure 1.8: Vertical displacement for a beam length of 2 m.

As shown in Figure 1.8, since we use a fine mesh (200 elements), both finite element solutions (Euler-Bernoulli and Timoshenko) are very close to their corresponding analytical solutions. However, we can clearly observe the effect of shear deformation. For short beams, the Euler-Bernoulli model underestimates the displacement because it does not account for shear effects. The Timoshenko model gives a displacement that is about twice as large, which is more accurate in this case.

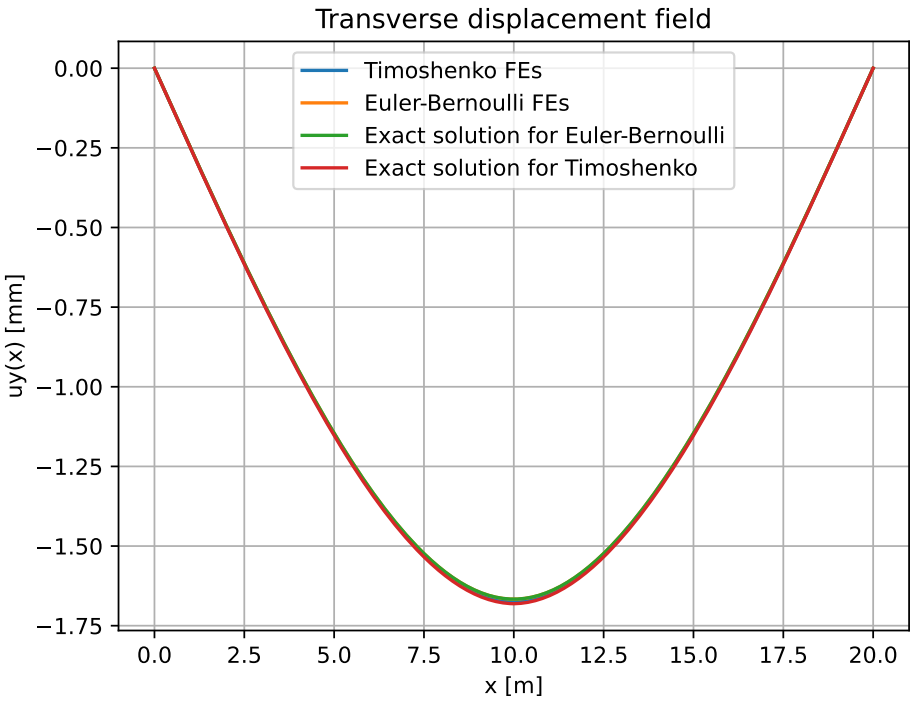


Figure 1.9: Vertical displacement for a beam length of 20 m.

In the case of a 20 m beam (Figure 1.9), the influence of shear deformation becomes less significant compared to the total deflection. The Euler-Bernoulli and Timoshenko results are very close. This shows that as the beam gets longer, shear deformation contributes less to the overall displacement. The small difference between the two models is mainly due to shear effects.

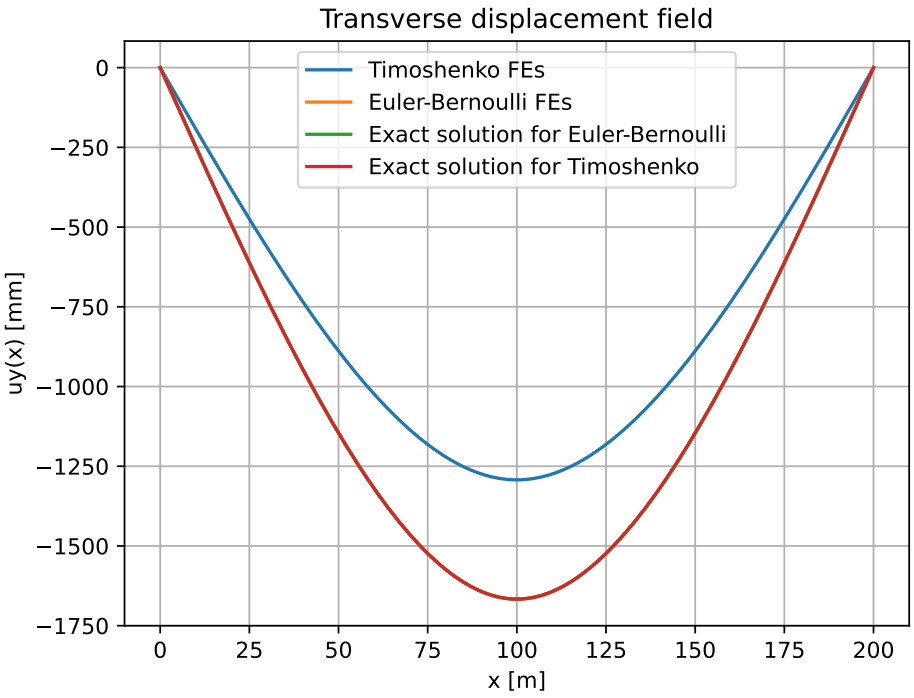


Figure 1.10: Vertical displacement for a beam length of 200 m.

In theory, for a very long beam like in Figure 1.10, the displacement predicted by the Timoshenko and Euler-Bernoulli models should be almost the same, because the influence of shear deformation becomes negligible compared to the total bending deformation.

However, here we observe that the Timoshenko displacement is actually smaller than expected, which is not physically correct. This result is caused by a numerical issue known as **shear locking**. Shear locking happens when independent linear approximations are used for both the transverse displacement u_{y0} and the rotation θ. This causes the element to become too stiff in shear, leading to an underestimated displacement for very slender beams.

1.5 BONUS – Beam Length Variation with Abaqus

We repeated the analysis of the beam length variation using Abaqus. The results are more or less the same as those obtained with our own finite element implementation. However, we do not observe any shear locking in the Abaqus results (Figure 1.13). This means that Abaqus uses some internal technique to automatically avoid this numerical issue, possibly through reduced integration. We also addressed this in the next section using selective reduced integration.

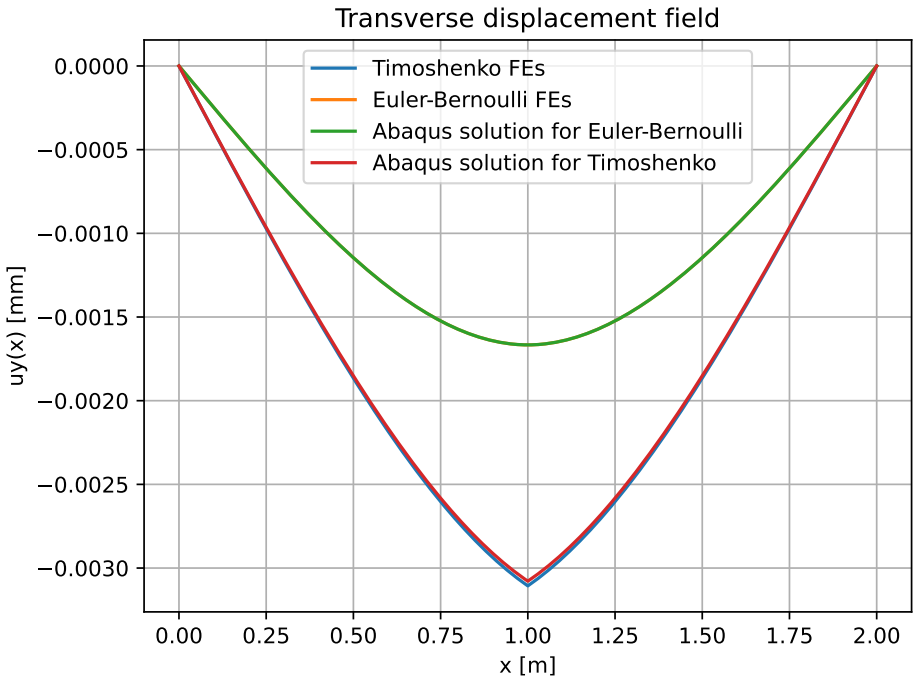


Figure 1.11: Vertical displacement for a beam of length 2 m using Abaqus.

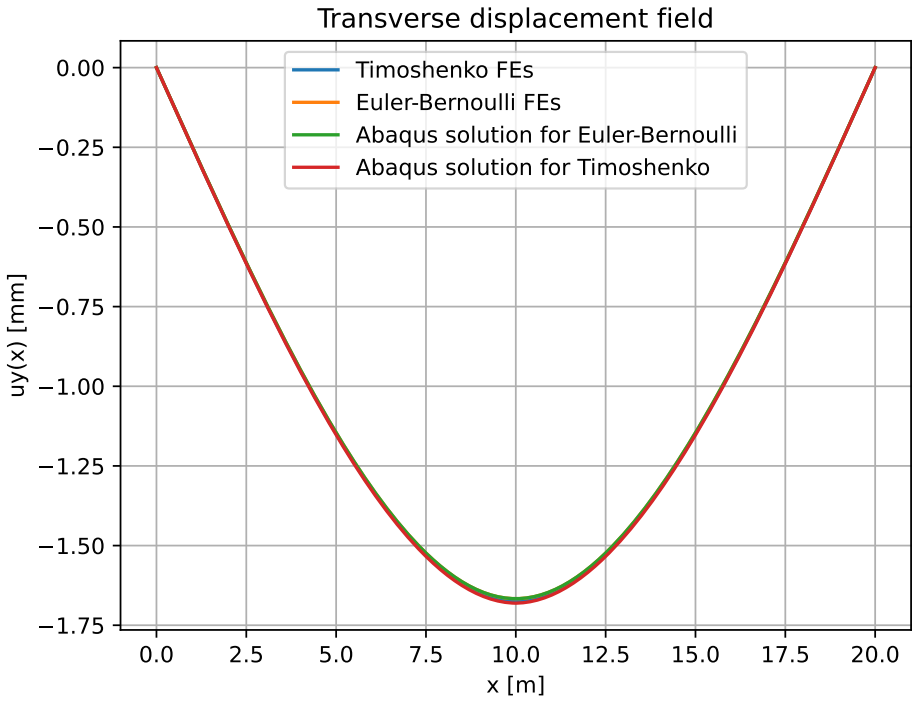


Figure 1.12: Vertical displacement for a beam of length 20 m using Abaqus.

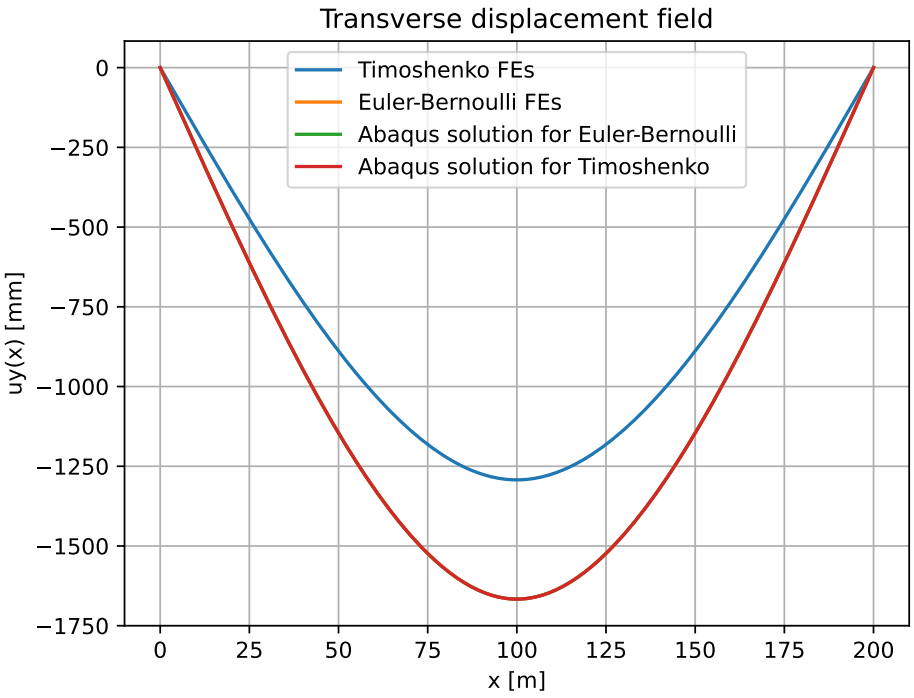


Figure 1.13: Vertical displacement for a beam of length 200 m using Abaqus.

1.6 Timoshenko stiffness matrix - selective reduced integration

To avoid the shear locking phenomenon, we can use a technique called selective reduced integration. This requires computing the stiffness matrix using numerical integration. The idea is to intentionally use less Gauss points than required for exact integration of the shear term. For the shear part (which involves quadratic functions, degree 2), exact integration would require 2 Gauss points. Instead, we use only 1 Gauss point. For the bending part (involving constant terms, degree 0), we still use 1 Gauss point, which is sufficient for exact integration. This approach leads to a modified stiffness matrix. The full derivation is presented in an Excel spreadsheet `NACES_SelRedInt.xslx`[1], as it is more visual.

The stiffness matrix for the Timoshenko beam with selective reduced integration is:

$$\begin{bmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{GA_c}{L} & \frac{GA_c}{2} & 0 & -\frac{GA_c}{L} & \frac{GA_c}{2} \\ 0 & \frac{GA_c}{2} & \frac{EI}{L}+\frac{GA_cL}{4} & 0 & -\frac{GA_c}{2} & -\frac{EI}{L}+\frac{GA_cL}{4} \\ -\frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{GA_c}{L} & -\frac{GA_c}{2} & 0 & \frac{GA_c}{L} & -\frac{GA_c}{2} \\ 0 & \frac{GA_c}{2} & -\frac{EI}{L}+\frac{GA_cL}{4} & 0 & -\frac{GA_c}{2} & \frac{EI}{L}+\frac{GA_cL}{4} \end{bmatrix}$$

The only changes compared to the full integration matrix are in terms k_{33}, k_{36}, k_{63}, and k_{66}. These correspond to integrals involving quadratic terms, which are not integrated exactly using just one Gauss point. We now show the vertical displacements computed with selective reduced integration for three beam lengths:

[1] `https://uclouvain-my.sharepoint.com/:x:/g/personal/felix_foret_student_uclouvain_be/EXTR_kQQc6dKl9ArTGIna_EBpq_lBgW_J_fhgpHesP0TMQ?e=PK9YGO` - Also on the GitHub

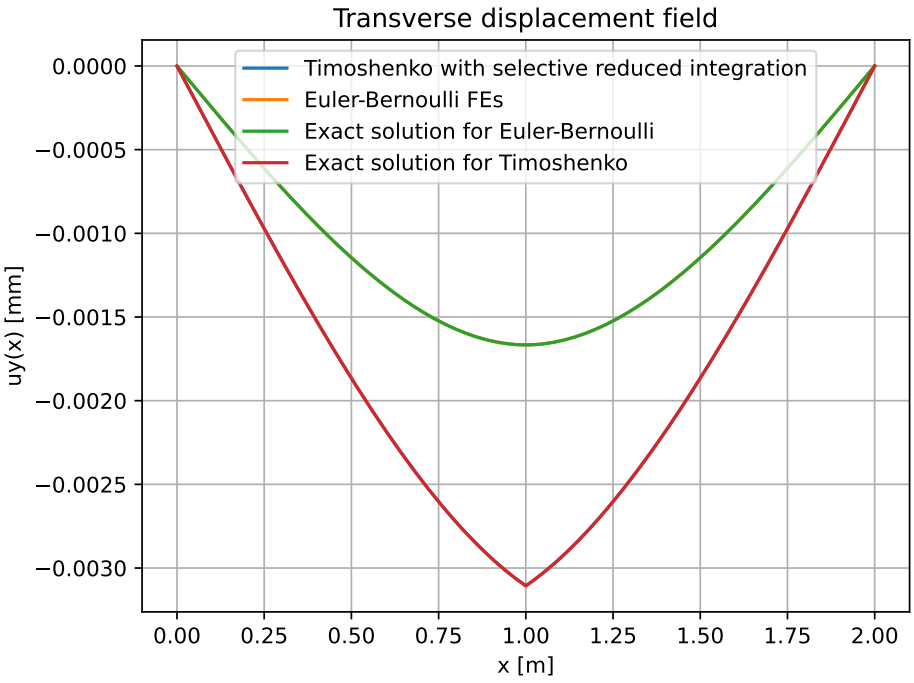


Figure 1.14: Vertical displacement for a beam length of 2 m.

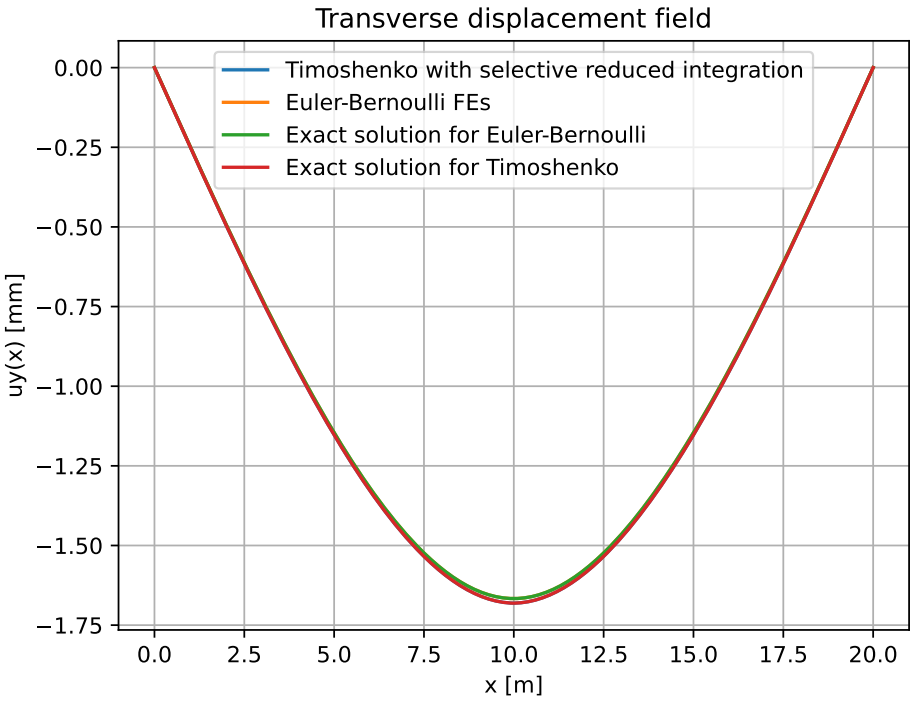


Figure 1.15: Vertical displacement for a beam length of 20 m.

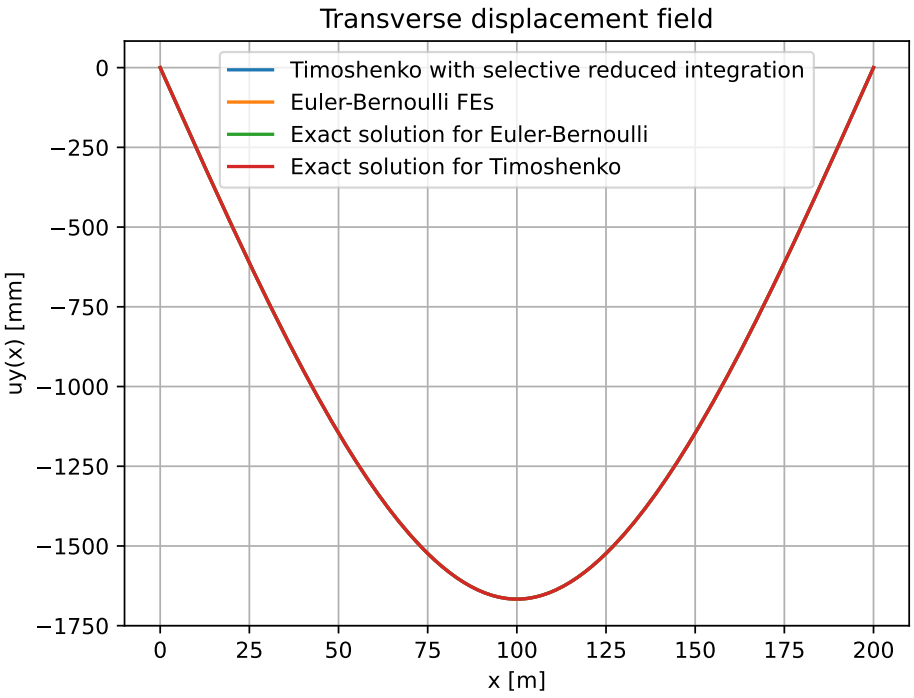


Figure 1.16: Vertical displacement for a beam length of 200 m.

Once again, since we use a fine mesh (200 elements), the FEM results match the exact solution very well. In Figure 1.16, corresponding to a long and slender beam, we observe that the Timoshenko solution with selective reduced integration behaves similarly to the Euler-Bernoulli solution. This confirms that shear locking is successfully avoided. We can also see that the displacement is slightly different from the exact Timoshenko solution (not very visible in our graph). This small discrepancy is due to the approximation introduced by reduced integration.

1.7 Geometrically Nonlinear Response of the Beam

In the previous sections, we performed a geometrically linear analysis of the beam. This means that displacements and forces were evaluated based on the undeformed configuration. A geometrically nonlinear approach, on the other hand, takes into account the deformation of the structure when computing final displacements and internal forces. This is often called a P-Δ analysis. In simple terms, when a load P is applied, the structure deforms, and this deformation can increase the effect of the applied load. For example, the bending moment can increase as $M = P \cdot \Delta$. To implement this in our code, we increment the load gradually until reaching the total applied load. We use the Newton-Raphson algorithm to solve the resulting nonlinear system. Our results are shown below:

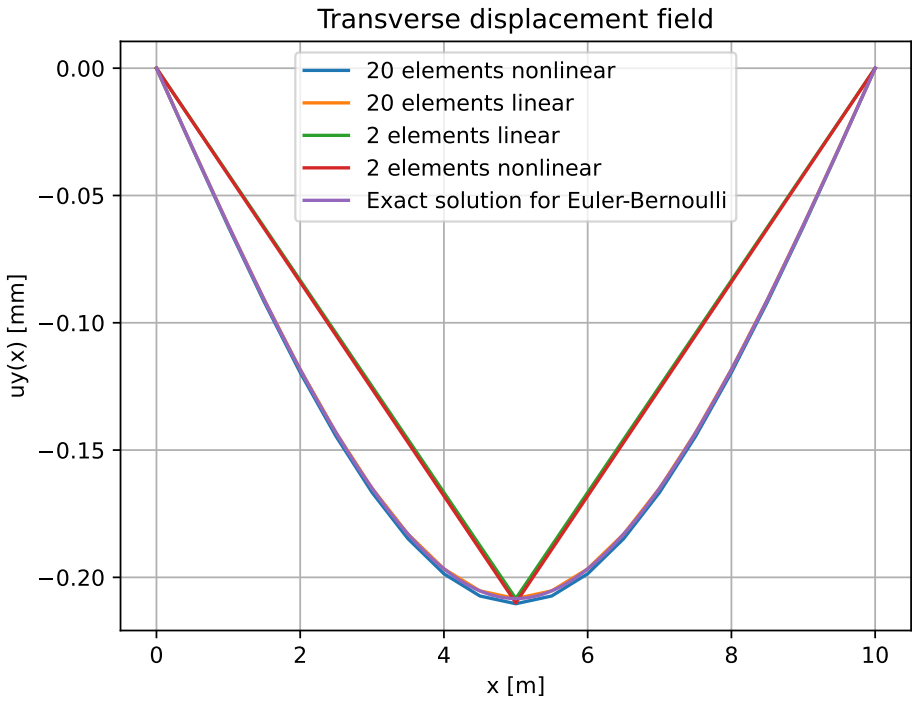


Figure 1.17: Vertical displacement for a beam of 10 m using a nonlinear Euler-Bernoulli FEs.

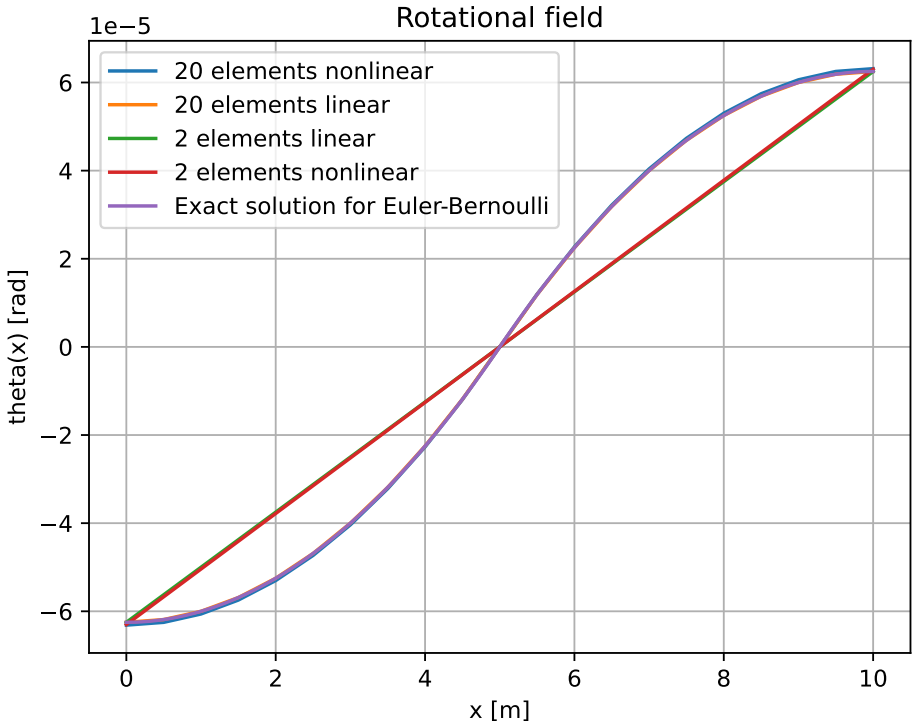


Figure 1.18: Rotation field for a beam of 10 m using a nonlinear Euler-Bernoulli FEs.

We observe that accounting for the nonlinear behavior increases the overall displacement. This is because

the horizontal force of 2000 kN now has a vertical effect due to the deformation of the beam. The more the structure deforms, the greater the additional bending caused by the horizontal load. We provide a zoomed view of the displacement curve since the effect is not clearly visible in the previous graphs:

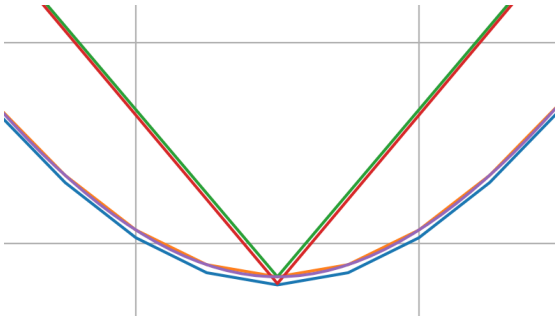


Figure 1.19: Zoom on the displacement curve to highlight nonlinear effects.

Analysis Type	**Max Vertical Displacement (mm)**
Linear, 2 elements	-0.2083
Linear, 20 elements	-0.2083
Nonlinear, 2 elements	-0.2100
Nonlinear, 20 elements	-0.2104

Table 1.1: Comparison of maximum vertical displacements (Euler-Bernoulli).

2 Problem 2

2.1 One dimensional behavior

2.1.1 Choice of the element type and beam theory

As the ratio $L/h = 6.5 < 10$ for this beam, the shear deformations can not be considered as negligible. The risk of shear-locking is low.
We therefore decided to use Timoshenko beam elements without selective reduced integration to take shear deformations into account and avoid hourglass instability.

2.1.2 Four point bending test - vertical displacement and von Mises stress, mesh convergence

Simulations were performed using successively 3, 6 and 60 beam elements. It has allowed to compare the results between 3 and 6 elements, and to ensure that 6 elements gave a result sufficiently close to the one with a much more refined beam. Those numbers of elements were also chosen to give them the same length along the beam.
The vertical displacement is represented on Figure 2.1, and the von Mises stress on Figure 2.2. We can see that the vertical displacement is under-evaluated for the 3 elements mesh, because there is no node in the middle. Between 6 and 60 elements, the displacements at each node are very close. For instance, at the middle node, the displacement is respectively 1.028 and 1.032 mm.

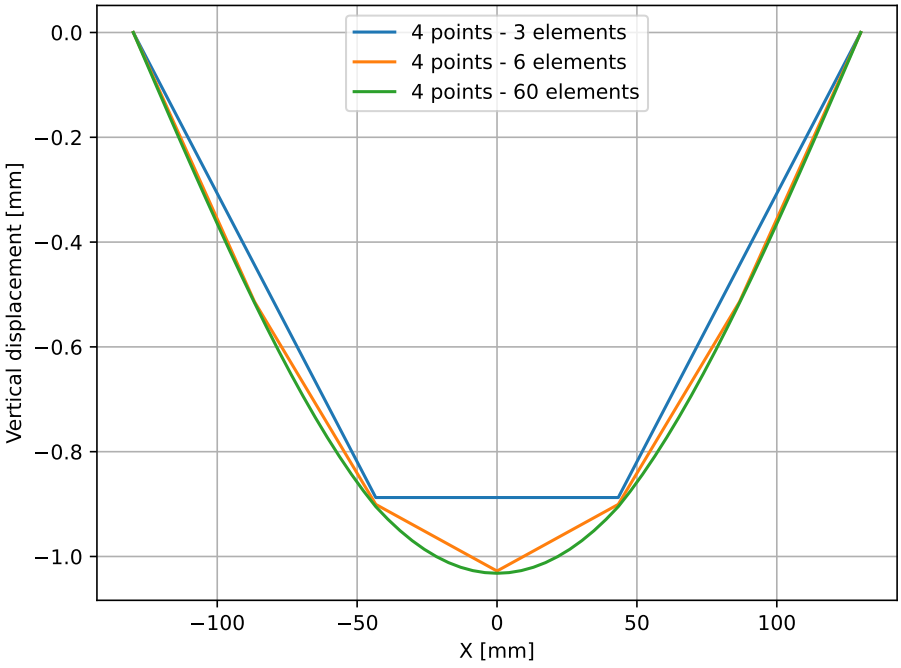


Figure 2.1: Vertical displacement for a four-point bending test.

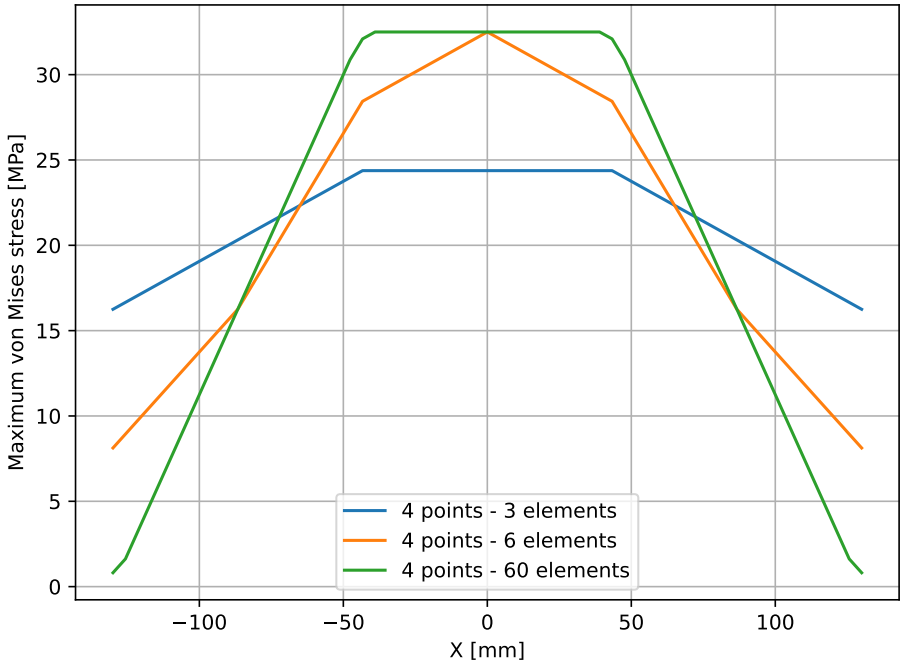
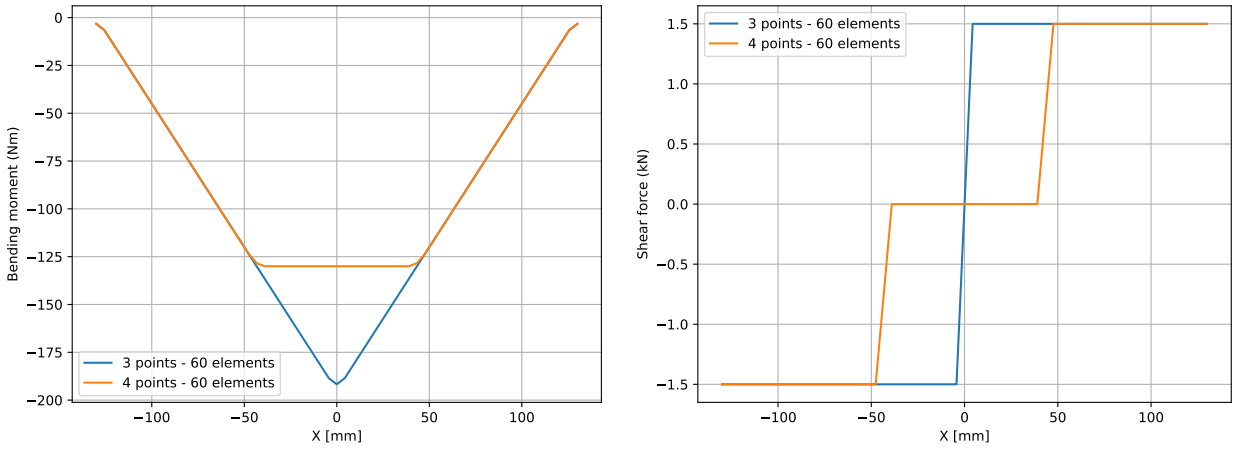


Figure 2.2: von Mises stress for a four-point bending test.

Looking at von Mises stresses, we can see that the maximum stress appears at the middle at the center. It is equal for 6 and 60 elements (32.5 MPa) but lower for 3 elements, again certainly because of the absence of node at the middle. We can also see that the stress is relatively different at the other nodes of the beam. We can conclude that the mesh with 6 elements is accurate for the displacement at the nodes and for von Mises stress at the middle of the beam, but inaccurate for von Mises stresses at the other places.

2.1.3 Three- and four-point bending test - Bending moments and shear forces

The graphs of bending moments and shear stresses are represented on Figure 2.3.



(a) Bending moment.

(b) Shear force.

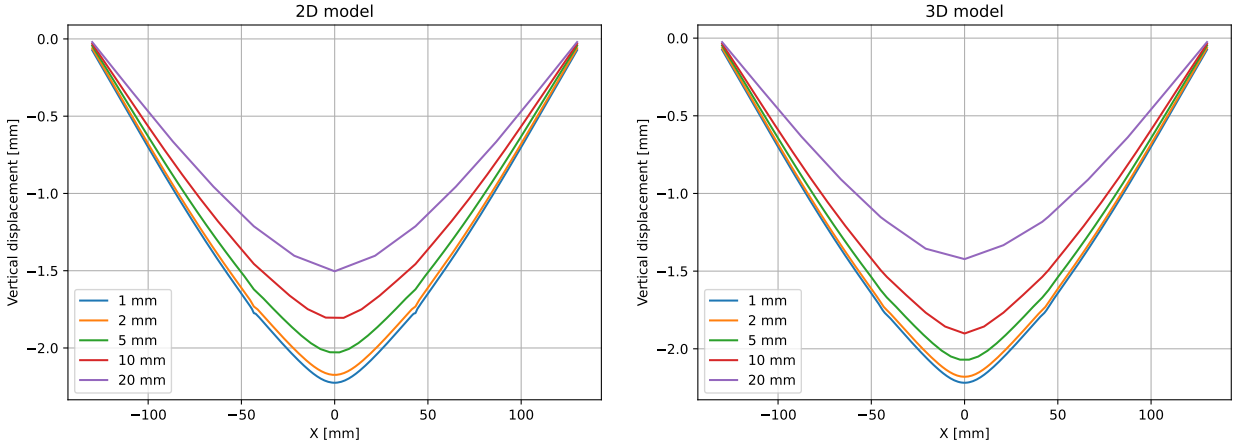
Figure 2.3: Comparison between internal efforts in three- and four-point bending tests.

We can see that the moment is constant on the middle third of the beam for a four-point test unlike the tree-point test where the bending moment diagram is purely triangular. The shear force diagram is composed of constant sections separated by steps where the forces are applied. The shear force is equal to zero un the middle third for the four-point test.
Four-point test is prefered because it allows to be sure that the break or the plastification happens in a cross-section where we kow exactely the values of the bending moment and shear force. This allows to calculate precisely the stress at this point, facilitating interpretation of results.

2.2 Mesh convergence - elasticity in 2D and 3D

We performed the mesh convergence with elements of the same dimensions according to three axes. The biggest element size is 20mm and the smallest one is 1mm. The vertical displacements for different meshing sizes are represented on Figure 2.4. The displacements in the middle both for 2D and 3D are shown in Table 2.1
To ensure consistency of the represented fiber, we chose to represent the displacement of the upper fiber. Due to the irregular mesh, we could not extract the data for an initially straight fiber other than this one. This causes some small particularities on the graph:

- The displacements at the extremities are not equal to zero because the upper fiber moves compared to the lower fiber where the roller are placed
- We see a small punching at the places where the forces are applied. To mitigate this effect, we applied a 30 MPa pressure on a surface of 15×3.33 mm^2 instead of a linear force. A residual punching still present but very small compared to the global displacement.



(a) Vertical displacement for a 2D model.

(b) Vertical displacement for a 3D model.

Figure 2.4: Vertical displacements along the beam.

Mesh size [mm]	Displacement [mm]	
	2D	3D
20	1.50	1.42
10	1.80	1.90
5	2.03	2.07
2	2.17	2.18
1	2.22	2.22

Table 2.1: Vertical displacement in the middle of the beam.

Both in 2D and in 3D, the displacements seem converging but the accuracy is not that good. We tried to refine the mesh even more but the computer of Candix room had memory lacks. It seems that the displacement will converge somewhere between 2.22 and 2.3 mm.

This value of displacement is way greater than the one obtained using beam elements in (a). This is probably due to the fact that the notch was ignored for the first case. Te notch reduces a lot the rigidity of the whole beam.

Looking at the differences between the models in 2D and in 3D, we see that the 2D model is closer to reality for a (very) coarse mesh, but converges more slowly as we refine the mesh. The opposite applies to the 3D model.

2.3 Elasto-plasticity in 3D

From this part, even the 1 and 2 mm mesh sizes leaded to memory lacks from the computer. All the following analysis was performed on 5, 10 and 20 mm mesh size.

We observe on Abaqus that the plastification occurs first at the top on the notch, on all the depth of the beam. The Table 2.2 shows the force and displacement leading to plastification for different mesh sizes.

Mesh size [mm]	Displacement [mm]	Force P [N]
20	1.26	2610
10	1.38	2190
5	0.85	1200

Table 2.2: Displacement and corresponding force leading to plastification.

We see that refining the mesh makes the plastification occur for a lower force. This is probably because the stress concentration next to the notch, and bigger element are more resistant to that because they kind of average the peak of stress over their whole volume.

We also observe here and in the next sections that the elastic rigidity decreases with the refinement of the mesh. This phenomenon is also highlighted on previous sections, where a more refined mesh leads to higher displacements for the same force.

Figure 2.5 shows the "total force"-"vertical displacement" relation for different mesh sizes. The point where the first plastifiaction occurs is highlighted. We distinguish clearly the linear elastic part before this point and the non linear part after.

We also see that the force increments could not go up to 3 kN for a 5mm mesh. Indeed, at 2670 N the cross-section in the middle is fully plastified and can deform infinitely without increment of force.

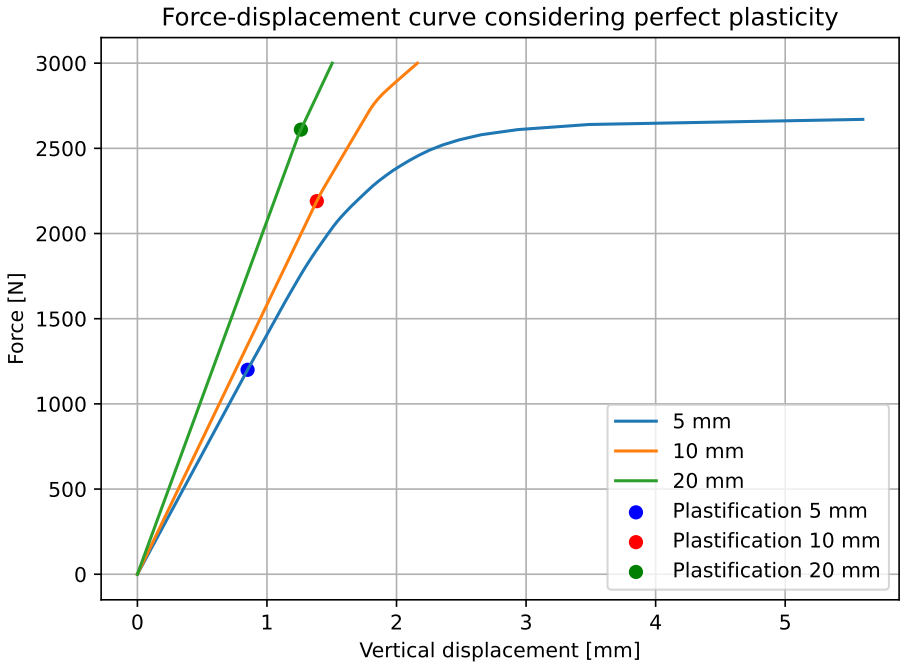


Figure 2.5: Force-displacement curve for different mesh sizes using perfect plasticity.

2.4 Hardening in 3D

The force-displacement curve is shown on Figure 2.6. We can see that the elastic part is exactly the same as in perfect plasticity. The difference is that the plastic part of the curve is more vertical. Indeed, the plastified elements still increase their rigidity, leading to a globally more rigid beam. This allows the increments of force to reach 3 kN for the 5mm mesh.

We finally see that the deformations increase with the refinement of the mesh. The 20mm mesh is nearly fully linear while the 5mm one shows an important curvature. This leads to a final deformation more than three times greater. This highlights the importance of mesh convergence to assess the deformations of a structure.

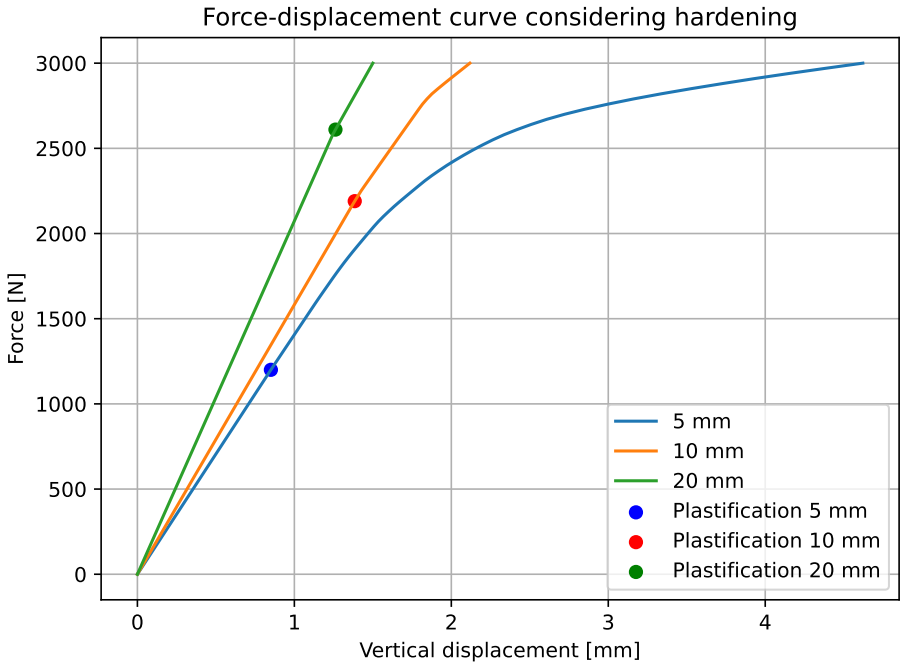


Figure 2.6: Force-displacement curve for different mesh sizes using a hardening rule.

2.5 Softening in 3D

Simulate the beam using a softening rule and calculating by increments of force prevent the program from converging. This is caused by the plastified sections reducing their rigidity. As the geometry of the beam only allows a little bit of plastic redistribution of stresses, the plastified sections need less and less stress to deform. The displacement thus tends to infinity after the start of plastification.

To counteract this effect, we applied increments of displacement instead of force at the two upper roller positions.
Figure 2.7 shows the force-displacement curve for this part. We can see that the beam is less stiff after plastification in this case. We also see on 10 and 5mm meshes that the force reaches a peak after which it decrease for bigger displacements.

When refining the mesh, we see that the peak of force is significantly lower and appears for a smaller displacement. When too coarsely meshed, it seems that there is a kind of bigger plastic redistribution of the stresses.

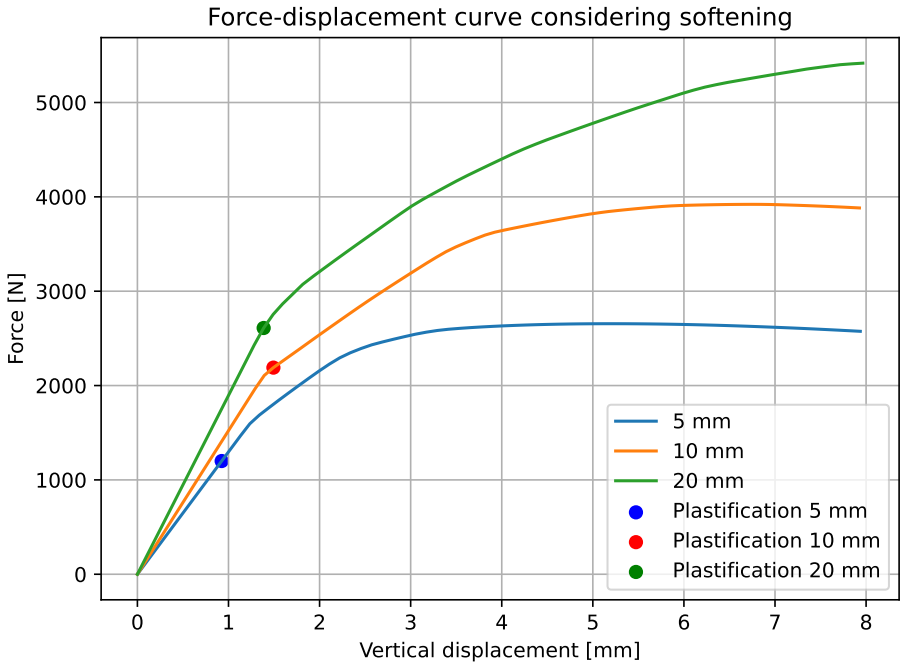


Figure 2.7: Force-displacement curve for different mesh sizes using a softening rule.

Unfortunately, we couldn't capture the deformed shape of the beam on the Candix computers. What we saw is that the deformations were more concentrated in the middle of the beam with softening than with hardening. There was also a dilatancy of the material on the upper fiber on the middle cross-section.

3 Code

All our python scripts, figures, inputs and outputs from Abaqus are available in a GitHub repository, which can be accessed through the following link: `https://github.com/feforet/LGCIV2041-NACES.git`.