Nejc
Novak

Matej
Borovinšek

Matej
Vesenjak

Zoran
Ren

Introduction to the Computer Simulations

Script

Introduction to the Computer Simulations

Script

Authors

Nejc Novak
Matej Borovinšek
Matej Vesenjak
Zoran Ren

March 2024

Table of Contents

University of Maribor Press

1 Introduction and short history overview

The history of finite element software can be traced back to the early 1960s when the first computer programs were developed to solve complex problems in engineering and mechanics [1]. At that time, analytical methods were used primarily in mechanics, which were unsuitable for solving complex problems. Analytical methods are based on the assumption that the problem domain is simple and can be represented by a set of simplified equations. However, many real-world problems are complex and cannot be represented as such. The Finite Element Method (FEM) overcomes this limitation by dividing the problem domain into smaller subdomains with simple geometry called finite elements. Each finite element is first computed locally, and then combined with other finite elements to solve the overall problem. This approach allows the FEM to solve problems with very complex geometries and underlying fundamental relationships. The early FEM programs were elementary and required a lot of manual input, but they laid the foundation for the sophisticated software systems that are available today. Even today, simulation programs rely on model data input via text files, reminiscent of the card-based input technique of the past [2].

One of the first significant advances in finite element software came in 1964 with the development of the NASTRAN program (NASA STRucture ANalysis) by NASA [3]. NASTRAN is a general-purpose finite element program that can be used to solve a wide variety of problems, including linear and non-linear static and dynamic analysis. NASTRAN was adopted quickly by industry and academia, and it remains one of the most popular finite element programs today.

Another significant milestone in the history of finite element software was the development of the ANSYS program in 1970 by John Swanson [4]. ANSYS was one of the first commercial finite element programs, adopted quickly by engineers and scientists due to its versatility and ease of use. ANSYS has continued to evolve over the years, and it is now one of the world's most influential and widely used finite element programs.

Other notable finite element software programs include ABAQUS, LS-DYNA, Marc, COMSOL Multiphysics, and many more. Engineers and scientists use these programs to analyse problems in various industries, including aerospace, automotive, civil engineering, manufacturing and biomedical engineering. Finite element software solves many problems, including structural analysis, heat transfer analysis, fluid flow analysis, electromagnetic analysis, and multiphysics problems.

Computational simulations have revolutionised how engineers and scientists design and analyse products and systems in all stages of the design process. It has enabled engineers to solve previously intractable problems and improved engineering design's efficiency and quality significantly.

Computational mechanics is a constantly changing discipline, with breakthroughs occurring regularly. The following are some of the essential trends in the finite element software development:

- The development of new and more accurate finite element formulations;
- The integration of finite element software with other engineering software tools, such as (Computer Aided Design – CAD) and (Computational Fluid Dynamics – CFD) software;
- The development of cloud-based finite element software;
- The development of Artificial Intelligence and Machine Learning techniques to improve the efficiency and accuracy of finite element analysis.

2   Theoretical foundations

The Finite Element Method is a numerical technique for approximating solutions of boundary value problems described with governing partial differential equations in 3D, 2D or 1D spaces [5]. It involves dividing a complex problem domain into smaller, geometrically simpler regions known as finite elements (FE) (Figure 1). The finite elements can be of different types, depending on the underlying one-dimensional (1D), two-dimensional (2D), or three-dimensional (3D) theory describing the finite element's ability to simulate specific behaviour. These elements are characterised by nodes at their vertices, edges, or even volume, representing specific points in the domain where the problem state variables are computed. Each node has several degrees of freedom (DOF), depending on its formulation. The type of finite element used always predetermines the number of DOFs associated with a node, see Section 2.2. The variation of state variables between nodes within finite elements is approximated by interpolation schemes, ranging from simple to complex, depending on the analysed problem. The finite elements are usually assembled into a mesh so that they are interconnected at the nodes.

Since the problem domain is divided into many finite elements and then uses interpolation functions to approximate the solution within each element, and, consequently, over a while domain, the final result is always an approximation rather than an exact solution. The accuracy of FEM solutions depends on several factors, including the number/size of finite elements, the choice of interpolation functions, the mesh quality and the applied fundamental theory. More finite elements, higher-order interpolation functions and

detailed governing equations will generally lead to more accurate solutions. However, this also increases the computational time, and, consequently, the cost of the analysis.
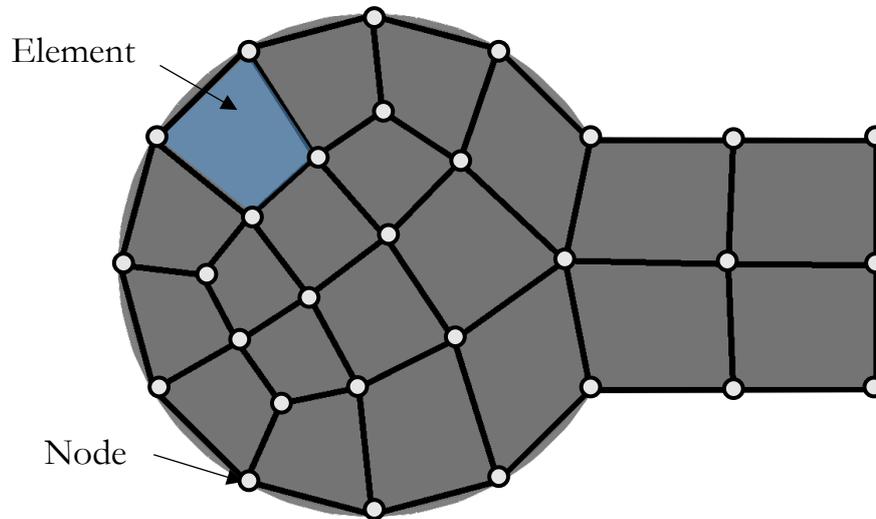
Figure 1: Finite element mesh

The governing equations typically follow the principles of equilibrium and compatibility, which are fundamental concepts in mechanics. The primary goal in solid mechanics is to establish a relationship between the applied loads, the material properties, and the resulting displacements in a given structure [6]. In structural mechanics, the basic governing equation is derived from the quasi-static equilibrium equation, which states that the sum of forces acting on a structure must be zero. This equilibrium equation is, in the FEM, typically expressed as:

$$[\mathbf{K}]\{\boldsymbol{u}\} = \{\boldsymbol{F}\} \implies \qquad \{\boldsymbol{u}\} = [\mathbf{K}]^{-1}\{\boldsymbol{F}\}$$

properties · function = action   unknowns

Figure 2: Fundamental relationships in the FEA

where $\mathbf{K}$ is the stiffness matrix, representing the material and geometric properties of the structure, $\boldsymbol{u}$ is the vector of nodal displacements for each degree of freedom, and $\boldsymbol{F}$ is the vector of all external forces.

The equation states that, when a structure is in equilibrium, the internal forces (represented by $\mathbf{K}\cdot\boldsymbol{u}$) must balance the external forces ($\boldsymbol{F}$). This system of equations forms the core of the FEM approach in structural analysis. The boundary conditions have to be prescribed to solve the system. The stiffness matrix $\mathbf{K}$ and the external force vector $\boldsymbol{F}$ are derived

based on the properties of the finite elements and the selected material model. It's important to note that the governing equations can vary, depending on the problem being solved (e.g., structural, thermal, fluid dynamics) and the specific characteristics of the material and geometry involved. The process involves discretising the domain into finite elements, computing the contributions of each element locally, and assembling the global system of equations that govern the entire structure.

2.1   Finite Element Analysis
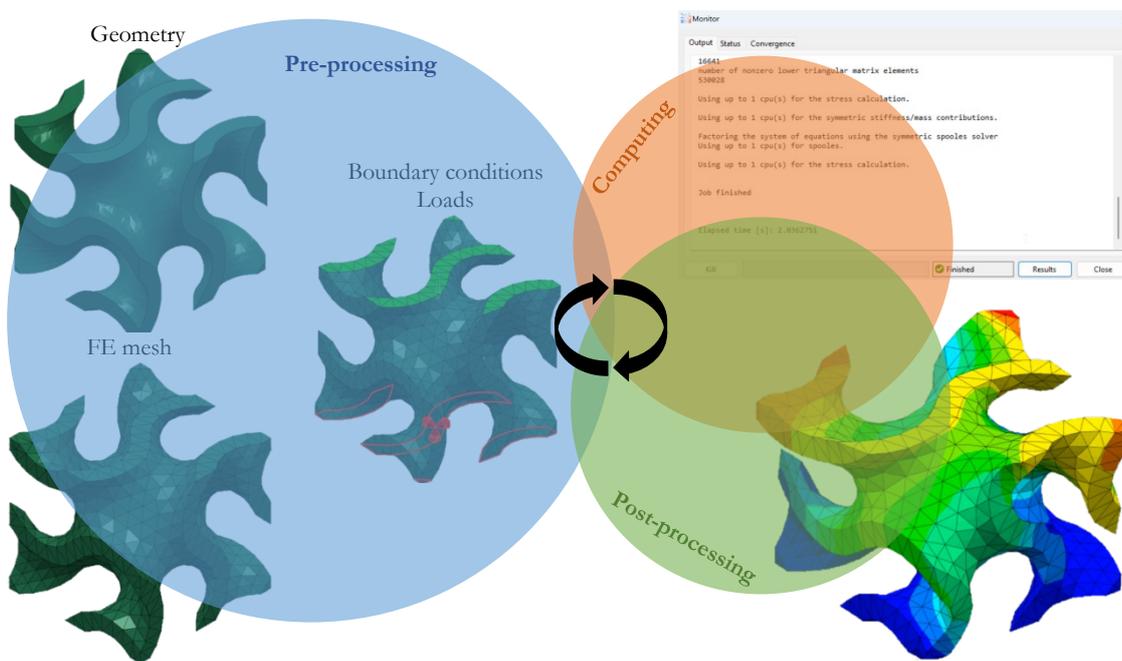
Figure 3: Typical FEA steps

The Finite Element Analysis (FEA) is divided into three basic steps (Figure 3):

1. The pre-processing step, where the computational model is prepared and checked with all the necessary computational data (FE mesh, boundary conditions (supports and loads), material properties, etc.). This step is usually done with integral or separate pre-processing programs, to create/import geometry, generate the FE mesh and define the boundary conditions, together with some specific problem data, like material or other domain data.

2. The computing (or analysis) step, is where the computational model prepared in the previous step is used as input data for the solver, which can be integral or separated again. The solver program contains all the necessary physics, computing theory and algorithms, to compute the underlying problem described by the input computational model efficiently and generate the desired results as output data.

These programs are sometimes called the "black box" programs, since they usually do not allow user interaction during the computing, i.e. solution phase.

3. The post-processing step, where the output data generated in the previous step are evaluated and visualised. This step can also be done with integral or separate pre-processing programs.

A single FEA is usually not enough to achieve accurate results, so the above three steps are repeated in a loop by improving the computational model until convergence is achieved of the output results.

2.2   Types of finite elements

Finite elements in the FEM are mathematical entities used to discretise and represent physical structures or domains. The choice of finite element type depends on the geometry, material properties, and the behaviour of the physical system under consideration. The most common types of finite elements are solid (3D), shell (2D), and line (1D) finite elements (Figure 4).
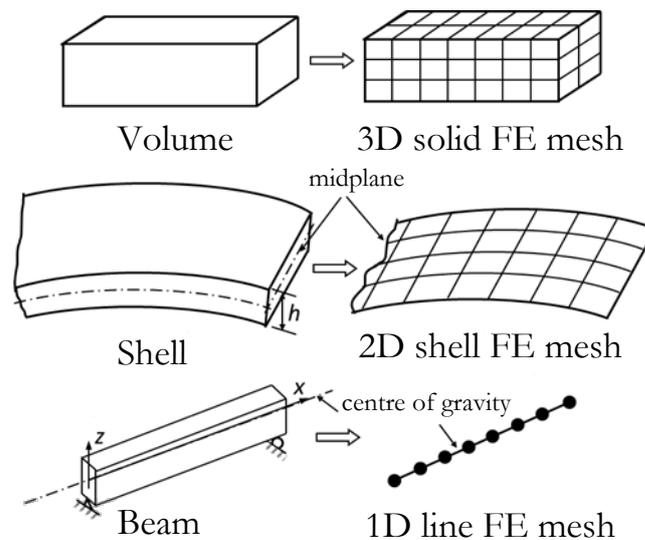
Figure 4: Use of different types of finite elements [2]

In mechanical systems, we typically consider six basic degrees of freedom associated with translational and rotational movements in a three-dimensional coordinate system. These six degrees of freedom correspond to translation motion along the x, y and z axes and rotations about the x, y and z axes, as shown in Figure 5.

Certain specific simulation requirements may require additional degrees of freedom. These can include non-mechanical degrees of freedom (e.g. temperature). It's important to consult the documentation or user guide of the specific software package to understand the additional degrees of freedom for proper simulation implementation.
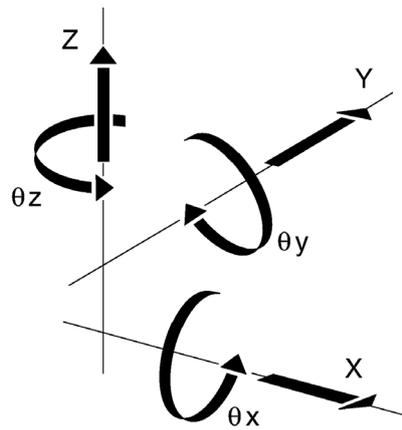
Figure 5: Degrees of freedom

The number of degrees of freedom in the FEA depends on the analysis type, which is often associated with the finite element type being used. In general 3D structural analysis, using 3D volume elements, only three translational DOFs (3 displacements) are enough to describe the problem fully. However, some specialised fundamental theories (like the beam or shell theory, separating the problem to the sum of in-plane and out-of-plane behaviour) also involve rotational degrees of freedom, where the nodes of associated beam and shell finite elements have the full set of DOF (3 displacements + 3 rotations) in 3D.

2.2.1 Solid finite elements

Solid (volumetric) finite elements are used for spatial discretization when the mechanical part or the problem domain displays typically dissimilar behaviour in all three dimensions. Solid finite elements are usually prismatic in shape, with tetrahedra and hexahedra being the most widely used elements in practical applications (Figure 6). However, due to the complexity of their shape functions, solid elements are computationally expensive and sometimes prone to errors. A common problem with solid elements is a negative volume that occurs when one face of the element passes through its opposite face under compression [7].

Solid finite elements, typically used for three-dimensional analysis of structures, have three translational degrees of freedom (x-displacement, y-displacement and z-displacement) at each node (Figure 7), and do not consider rotational displacements. The element shown in Figure 7 has a total of 24 degrees of freedom (8 nodes multiplied by 3 degrees of freedom per node).

Tetrahedron

Prism

are used for

Thick plates and flanges

Hexahedron

Thick wall shells and reservoirs

Castings
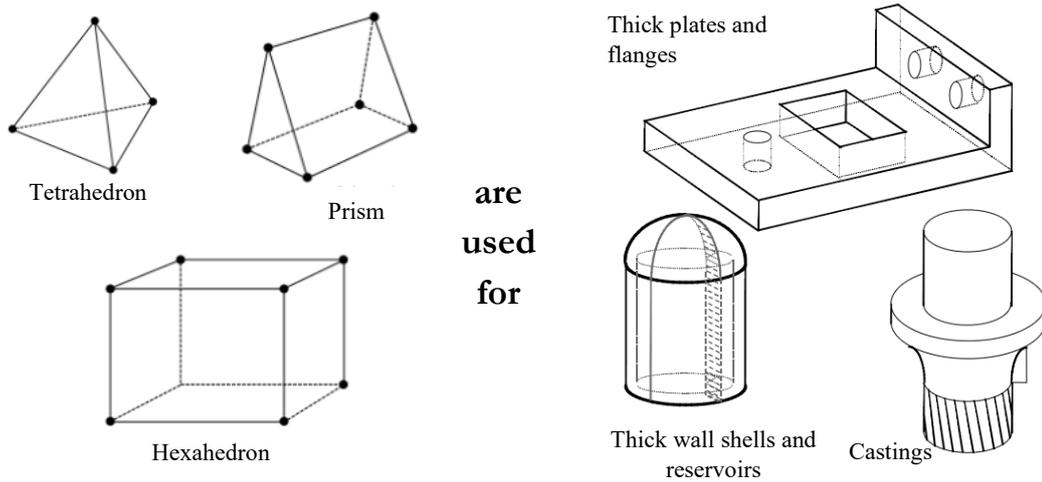
Figure 6: Volume finite elements [1]

u_z

u_x

Element

u_y

Node

z

x

y
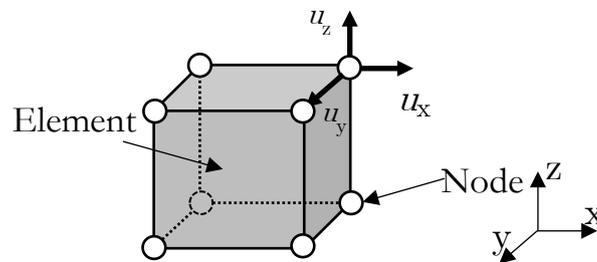
Figure 7: Volume FE

2.2.2 Surface finite elements

Surface finite elements are used in cases where two-dimensional effects, such as bending and membrane loads, dominate the structural or mechanical behaviour. In such cases, the structure has a relatively thin cross-section compared to its other dimensions (e.g., plates, walls, shells). Shell finite elements commonly have triangular or quadrilateral shapes, and are positioned along the mid-plane of the volumes under consideration (Figure 8). In shell finite elements, it is often assumed that the thickness of the components does not change due to the applied load. They balance computational efficiency and accuracy for problems with two-dimensional effects dominating structural behaviour. Their application is particularly advantageous when modelling a structure's three-dimensional volume is unnecessary or computationally expensive.

The basic variables are computed in finite element nodes on the mid-plane, while some theoretical relationships assume their variation through the thickness. In solid mechanics, such assumptions can be about specific stress-strain states, such as plane strain, plane

stress, axisymmetric symmetry, and separate membrane and bending stresses. These theoretical assumptions simplify the treatment of the problem by introducing some simple mathematical relationships to determine basic variables at nodes considering only two planar DOFs of the mid-plane. The stress-strain states outside the mid-plane, such as bending stress on the outer surface of a shell or circumferential stress in axisymmetric problems, are computed in the post-processing phase of the FEA, using simple analytical relationships based on established theories and principles. While these assumptions may not capture the full complexity of the problem, they provide reasonable approximations, and facilitate easier and faster analysis of the underlaying problem.
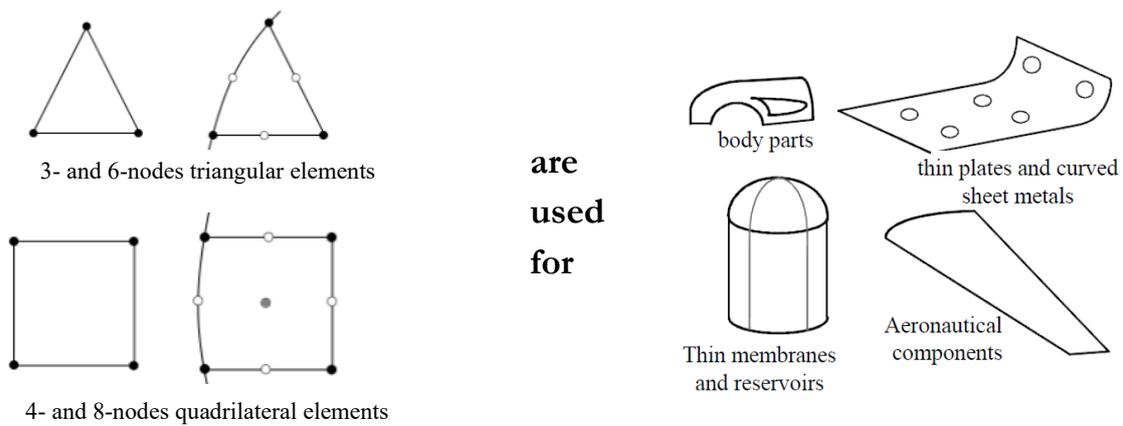
Figure 8: Surface finite elements [1]

The 3D surface finite elements are generally divided into two groups. The first group comprises membranes and plates, while the other consists of shells. All of them represent a thin, flat structure. The shell elements have all six mechanical DOF (3 displacements + 3 rotations) defined at each node, as shown in Figure 9. Three displacements (x-displacement, y-displacement and z-displacement) and three rotations (around the x- , y- and z-axes) are defined in each node. This allows for the shell element's full translational and rotational motion at each node. The shell finite element in Figure 9 has a total of 24 degrees of freedom (4 nodes multiplied by 6 DOF per node).
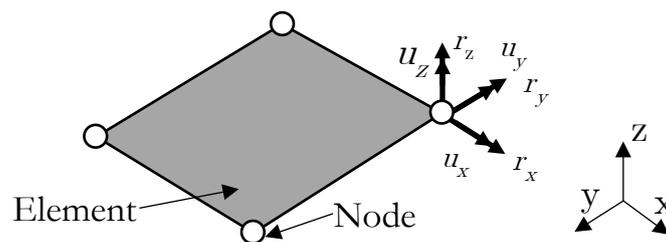
Figure 9: Shell finite elements

The membrane and plate finite elements don't account for the in-plane bending, so each node of the plates and membranes has five degrees of freedom, as shown in Figure 10. They account for all three possible displacements (x-displacement, y-displacement and z-displacement) and two out-of-plane rotations (around the x- and y-axes). The plate and membrane finite element in Figure 10 has a total of 20 degrees of freedom (4 nodes multiplied by 5 DOF per node).
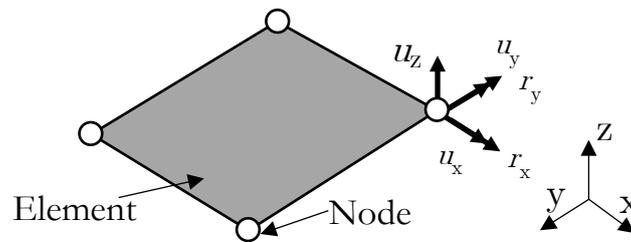
Figure 10: Plates and membranes

Planar finite elements are limited to two-dimensional analysis, and do not consider out-of-plane displacements or rotations as fundamental variables (Figure 11). Each node of a planar finite element has only two DOFs (x-displacement and y-displacement). The plate finite element in Figure 11 has a total of 8 degrees of freedom (4 nodes multiplied by 2 DOF per node).
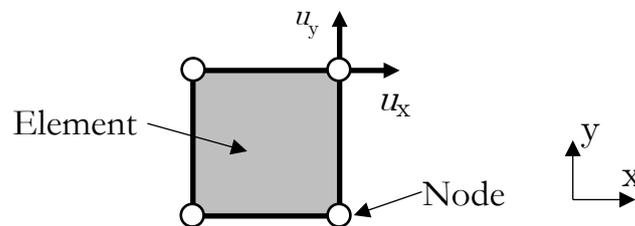
Figure 11: 2D planar FE

2.2.3 Line finite elements

Line finite elements are employed in the FEM when analysing structures that are long and slender, experiencing predominantly one-dimensional effects, such as axial and bending deformations. These elements are arranged along the mid-line or centroidal axis of the components (Figure 12). Line finite elements are mathematically the simplest, and involve theoretical assumptions about the basic variable variation over the cross-section of line elements, typically described by analytical functions. With line finite elements, it is often assumed that their cross-sections do not deform under the applied load. Thus, constant cross-sectional properties are considered (moments of inertia).
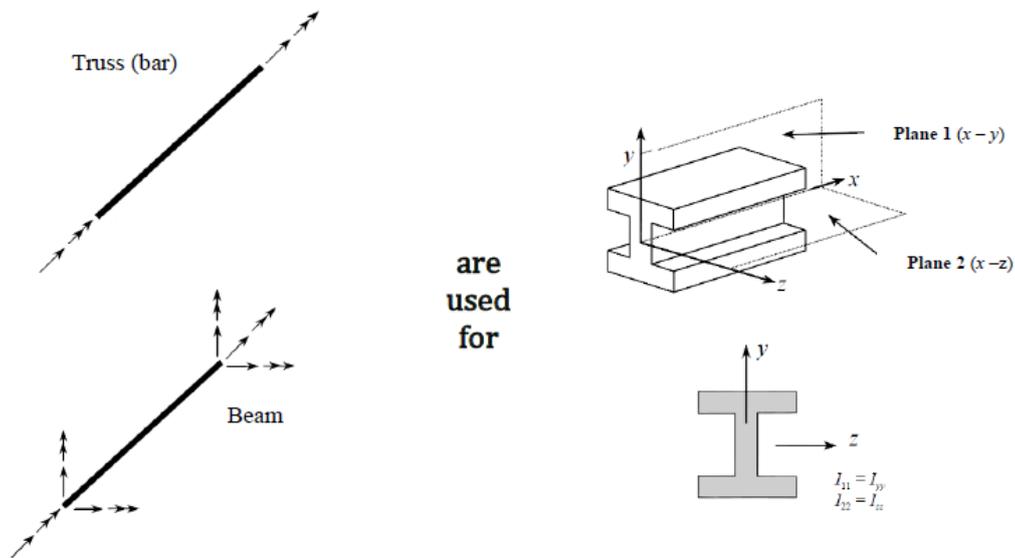
Figure 12: Beam and truss finite elements [2]

Line finite elements are structural elements that model slender structures like trusses and beams. Beam elements are used to model structures that experience axial, bending and shear deformations. These elements are more complex than truss elements, and can capture structures' bending behaviour. Therefore, they have all six mechanical degrees of freedom (x-displacement, y-displacement, z-displacement and rotations around the x-axis, y-axis and z-axis) defined at each node (Figure 13). They allow for modelling the full range of structural deformations, including bending and torsion. For beams with an open profile, an additional DOF, known as the seventh DOF, is included, to control torsion in the open profile beams. This additional DOF allows for a more accurate simulation of the behaviour of beams with open cross-sections. The beam finite element in Figure 13 has a total of 12 degrees of freedom (2 nodes multiplied by 6 DOF per node).
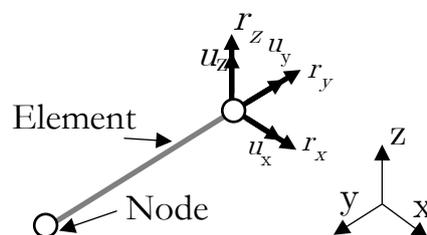
Figure 13: Beam finite elements

Truss elements are the simplest finite elements for structural analysis by assuming that the deformation behaviour is primarily axial, neglecting bending effects. Truss elements (Figure 14) are used to model structures that experience primarily forces and deformations

along the axis of the element. Therefore, they have only three displacements (x-displacement, y-displacement and z-displacement) defined at each node. Rotations are not defined for this element, as it is used primarily to model axial loads and does not account for bending or torsion effects. The truss finite element in Figure 14 has a total of 6 degrees of freedom (2 nodes multiplied by 3 DOF per node).
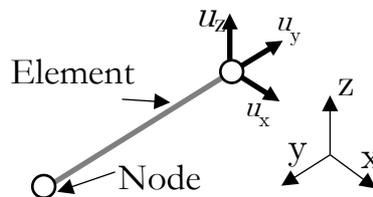
Figure 14: Truss (bar) finite elements

2.3   Symmetry and asymmetry

Symmetry, or asymmetry, is often employed in FEA as a modelling technique, to reduce computational costs and simplify the analysis of structures whose boundary and loading conditions are (a)symmetrical. Symmetry conditions are applied to only one symmetric/asymmetric part, and replace the other with symmetric/asymmetric boundary conditions on the symmetry/asymmetry plane, where spatial degrees of freedom must be constrained appropriately. Figure 15 illustrates symmetric and asymmetric boundary conditions for a geometrically symmetric plate bending case exposed to symmetric (left) and asymmetric (right) bending load cases.
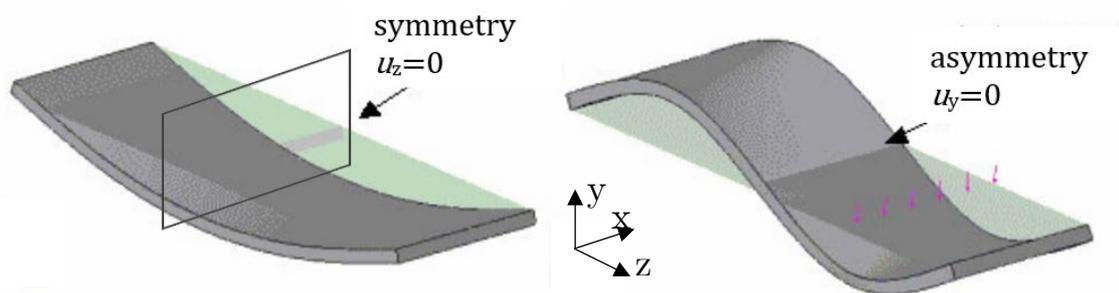
Figure 15: Symmetry and asymmetry [17]

Constraining only one displacement for both symmetric and asymmetric volume elements is adequate. However, when rotations are permitted in the chosen finite element (e.g., beams, shells), it is also essential to restrict rotations. Table 1 shows the boundary

conditions for symmetry and asymmetry, specifying the appropriate fixed displacements and rotations in all three coordinate directions.

Table 1: Symmetry definition

Type of (a)symmetry	Degrees of freedom
Symmetry in the x-direction	$u_x = ur_y = ur_z = 0$
Symmetry in the y-direction	$u_y = ur_x = ur_z = 0$
Symmetry in the z-direction	$u_z = ur_x = ur_y = 0$
Asymmetry in the x-direction	$u_y = u_z = ur_x = 0$
Asymmetry in the y-direction	$u_x = u_z = ur_y = 0$
Asymmetry in the z-direction	$u_x = u_y = ur_z = 0$

2.4   Unit systems

Consistent units are mandatory for successful finite element analysis [8]. Some modern FEA systems have no built-in unit systems, making the unit consistency the analyst's responsibility. The fundamental units in structural analysis are mass, length, time and temperature. All other units are derived from these fundamental units. The International System of Units (SI from French *Système International*) defines a consistent set of fundamental and derived units used commonly in engineering, as shown in Table 2. While SI uses 1 m as a fundamental unit for length, mechanical engineers prefer the 1 mm unit for length and 1 t for weight. Since 1 mm and 1 t are not SI-consistent units, scaling of dependent units is necessary, as in Table 2.

Table 2: Consistent units

Quantity	SI (m, kg, s)	mm-t-s
Length	m	mm
Mass	kg	t
Time	s	s
Temperature	K	K
Work/Energy	J	mJ
Properties for mild steel		
Density	7850 kg/m^3	$7.85{\cdot}10^{-9} \text{ ton/mm}^3$
Young's modulus	$2.1{\cdot}10^{11} \text{ N/m}^2 \text{ (Pa)}$	$2.1{\cdot}10^5 \text{ N/mm}^2 \text{ (MPa)}$
Yield stress	$2.35{\cdot}10^8 \text{ N/m}^2 \text{ (Pa)}$	$2.35{\cdot}10^2 \text{ N/mm}^2 \text{ (MPa)}$
Thermal conductivity	$5.2{\cdot}10^1 \text{ W/mK}$	$5.2{\cdot}10^1 \text{ mW/mmK}$
Heat capacity	$4.8{\cdot}10^2 \text{ J/kgK}$	$4.8{\cdot}10^8 \text{ mJ/tonK}$
Heat flux	1.0 W/m^2	$1{\cdot}10^{-3} \text{ mW/mm}^2$
Acceleration of gravity	9.81 m/s^2	$9.81{\cdot}10^3 \text{ mm/s}^2$
Equivalent to 1 km/h	$2.78{\cdot}10^{-1} \text{ m/s}$	$2.78{\cdot}10^2 \text{ mm/s}$

Basic equations are used to test whether a set of units is consistent. For example, for a force using the second Newton's law: $F = m \cdot a$, or elastic stress $\sigma = E \cdot \varepsilon$.

3   Material definiton in FEA

The material properties of the components being analysed by the FEM are crucial for accurate simulations. Material properties characterise how a material responds to various loads and deformations. Typically, the material properties are associated with the CAD model, and are transferred to the computational model automatically if the FEA is integrated with the CAD software. However, most FEA software is not integrated with CAD software; therefore, the material properties must be defined following the consistent units described in Section 2.4.

In linear elastic solid mechanics, two material parameters - the elastic modulus (E) and Poisson's ratio (v), are sufficient to describe the linear elastic behaviour of solid isotropic material [9]–[11]. Additional material parameters are required in the case of more advanced material behaviour (nonlinear), loading (dynamic effects), or anisotropy. This necessitates a more refined and challenging determination of material parameters.

3.1   Young's modulus and Poisson's ratio

Young's modulus, also known as the modulus of elasticity (symbol E), is a measure of a material's stiffness in the elastic deformation region [9]. Hooke's law defines it as the ratio of stress to strain for a material in the elastic proportional deformation region [11]. The most straightforward method to measure the modulus of elasticity is through a uniaxial tensile test, as illustrated in Figure 16. A test specimen is subjected to an axial tensile force during a uniaxial tensile test, resulting in its deformation. The applied stress is determined

by dividing the applied force by the cross-sectional area of the specimen, which generally changes during deformation. The strain is determined by dividing the specimen length change (extension) by the specimen length (initial measuring length + extension), which generally also changes during deformation. By plotting the true or engineering stress-strain relationship and identifying the linear elastic region, the modulus of elasticity can be determined from the slope of the linear portion of the curve by equation (1). It is important to note that this equation assumes that the material exhibits linear elastic behaviour, meaning that both stress and strain maintain a linear relationship within the material's elastic proportionality range. In practice, the elastic modulus is often determined by averaging multiple measurements taken within the linear portion of the stress-strain curve.

$$E = \frac{\frac{F}{A}}{\frac{\Delta L}{L_0}} = \frac{\sigma}{\varepsilon} \tag{1}$$

The modulus of elasticity is constant in all directions for isotropic materials, but may vary in orthotropic and anisotropic materials. Different testing methods or models are often used to assess the material's elastic behaviour accurately in the latter materials.
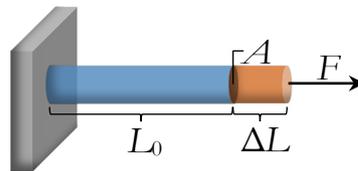
Figure 16: Tensile loading

This simple equation does not account for any imbalances or nonlinearity within the material, nor does it consider the influence of other factors such as temperature changes, moisture, loading rate, etc. In such cases, more advanced methods and models are required to determine the material's elastic properties accurately.

Table 2 provides typical values of the elastic modulus for materials used commonly in engineering. The standard unit for the elastic modulus is Pascal (Pa) or Newton per square metre (N/m^2). However, when performing simulations with different units for length or load, it is necessary to adjust the unit of the elastic modulus accordingly.

In mechanical engineering, it is common to use millimetres (mm) for length units and Newtons (N) for loads. The elastic modulus must be expressed in megapascals (MPa) or Newtons per square millimetre (N/mm²) to retain unit consistency.

Table 3: Young's modulus of different materials [9]

Material	Young's modulus [N/m²]	Young's modulus [N/mm²]
tungsten	$3,6 \cdot 10^{11}$	$3,6 \cdot 10^5$
steel	$2,1 \cdot 10^{11}$	$2,1 \cdot 10^5$
copper	$1,1 \cdot 10^{11}$	$1,1 \cdot 10^5$
aluminium	$6,9 \cdot 10^{10}$	$6,9 \cdot 10^4$
Pyrex glass	$6,2 \cdot 10^{10}$	$6,2 \cdot 10^4$
nylon	$3,7 \cdot 10^9$	$3,7 \cdot 10^3$
teflon	$3,7 \cdot 10^8$	$3,7 \cdot 10^2$

Shear modulus, often denoted as *G*, is a material property that quantifies the material's resistance to deformation under shear stress. It is one of the elastic moduli, along with the previously described Young's Modulus and compressibility (bulk) modulus *K* (which characterises the material's response to volume-changing stress, i.e. hydrostatic stress). Shear modulus is particularly relevant in materials subjected to shear forces, such as those involved in torsional or shearing deformations.

$$G = \frac{E}{2(1 + v)} \tag{2}$$

$$K = \frac{\sigma}{\frac{\Delta V}{V}} = \frac{E}{3(1 - 2v)} \tag{3}$$

Poisson's ratio *v* is a material property that equals the negative ratio of lateral strain to longitudinal strain in a material subjected to uniaxial loading, and is defined as [10]:

$$v = -\frac{\varepsilon_y}{\varepsilon_x} \tag{4}$$

where ε_x is the longitudinal specific strain and ε_y is the transverse specific strain. Both deformations are also determined based on measurements of changes in length and transverse dimension during a uniaxial tensile test of the material at engineering strains:

$$\varepsilon_x = \frac{\Delta L}{L_0} \qquad \text{and} \qquad \varepsilon_y = \frac{\Delta d}{d_0} \tag{5}$$

According to the second law of thermodynamics, all three moduli (elastic modulus, shear modulus and compressibility modulus) must be greater than 0. Therefore, the Poisson's ratio can only have values greater than -1 and less than 0.5. The theoretical upper limit for Poisson's ratio in isotropic materials is 0.5, corresponding to uncompressible material experiencing the maximum possible lateral contraction at a given axial strain. Materials with lower Poisson ratios exhibit smaller lateral deformation under axial loading.

Most common materials have positive Poisson's ratios from 0 to 0.5. This means that the material contracts laterally when stretched longitudinally. Metals, polymers, and many other materials fall into this category. The higher the value, the greater the lateral contraction relative to axial elongation.

A Poisson's ratio of 0 indicates that the material does not change in a lateral direction when stretched. Natural materials with such Poisson's ratio are very rare. Cork may exhibit values close to zero.

A negative Poisson ratio implies that the lateral deformation of the material is qualitatively the same as the longitudinal. i.e. the material expands laterally when stretched longitudinally. Natural materials with such behaviour are ever rarer, and are known as auxetic materials [12]. Materials with negative Poisson's ratios experience large volumetric changes under deformation, which enhances some mechanical properties. The most well-known examples of such materials can be found in sports equipment (tennis rackets)[13] and clothes (Gore-Tex) [14].

Table 4 provides typical values of Poisson's ratio for various materials. It's important to note that the specific Poisson`s ratio of a material depends on factors such as its molecular structure, composition and mechanical properties. Poisson's ratio can also vary with temperature, pressure, and other environmental conditions. Therefore, the range of values of Poisson's ratio is usually given. However, a slight change of Poisson's ratio generally doesn't affect the simulation results.

Table 4: Poisson's ratio of different materials [10], [15]

Material	Poisson's ratio
rubber	~ 0,5
gold	0,42
clay	0,3–0,45
magnesium	0,35
titanium	0,34
copper	0,33
aluminium	0,33
stainless steel	0,3–0,31
steel	0,27–0,3
cast iron	0,21–0,26
sand	0,2–0,45
concrete	0,2
glass	0,18–0,3
foam	0,1–0,4
cork	~0
auxetic metamaterials	-1 < ν < 0

3.2   Material data in advanced simulations

Additional material data are required for more realistic simulations of certain natural phenomena, or considering material behaviour beyond the elastic range. When materials undergo permanent (plastic) deformation, only Hooke's law (the linear relationship between stress and strain) can no longer describe their behaviour accurately.
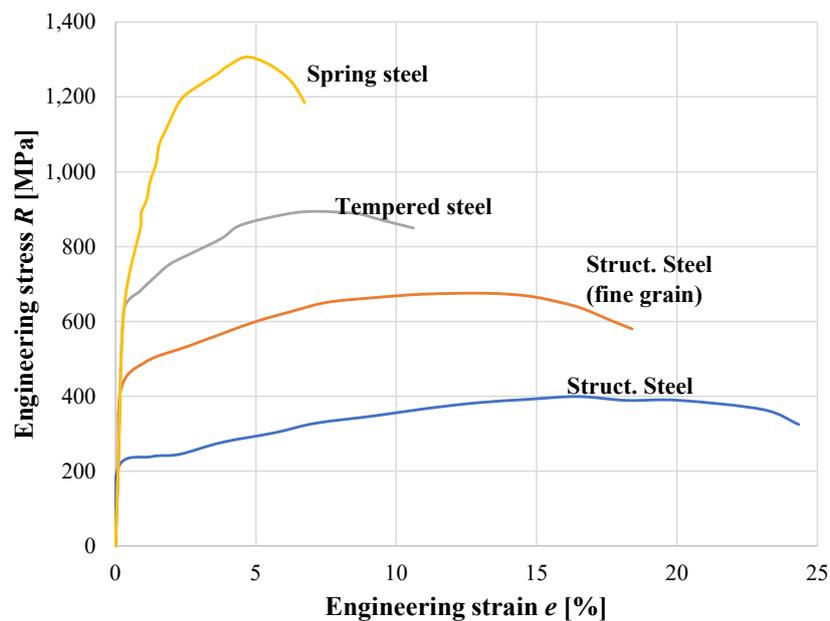

Figure 17: Engineering stress-strain relationships for different materials

To capture the nonlinear behaviour of materials beyond the elastic range, it is necessary to determine the stress-strain relationship under increasing load until failure [16]. As mentioned in the previous section, uniaxial tensile testing is performed commonly, to

obtain engineering stress-strain relationships for different materials. During this test the material is subjected to increasing tensile loads until it fails, and the corresponding engineering stress R (the load divided by the original cross-sectional area) and engineering strain e (the change in measuring length divided by the original length) values are recorded, as can be seen in Figure 17.

The resulting stress-strain relationship typically exhibits several distinct regions, including the elastic deformation region, the yield point and the plastic deformation region. The linear relationship between stress σ and strain ε in the elastic region is described by Hooke's law ($\sigma = E \cdot \varepsilon$), where E is the elastic modulus. However, as the stress exceeds the yield point, the relationship becomes nonlinear ($\sigma = \mathrm{f}(\varepsilon)$), indicating plastic deformation and permanent changes in the material's shape.

The change of the specimen's cross-section during deformation is difficult to measure, so the engineers have simplified the stress determination by neglecting this change and defined the engineering stress instead. The engineering stress is determined by dividing the applied force by the specimen's initial (undeformed) cross-sectional area. Similarly, the engineering strain is calculated by dividing the specimen length change by the initial measuring length. The true and engineering stresses and strains are the same in the elastic region, but differ in the plastic or elastoplastic deformation regions. The true stresses are always higher than the engineering stresses, while the opposite is true for the true and engineering strains.

The engineering stress-strain relationship provides valuable information about the material's behaviour under various load levels, including its strength, ductility and ability to withstand deformation. These data are essential for simulating the nonlinear behaviour of materials accurately, and predicting their response under different loading conditions in engineering simulations.

However, it is important to note that the engineering stress-strain relationship, which assumes a constant cross-sectional area, does capture the material's behaviour accurately beyond the yield point, especially for ductile materials that undergo significant necking or localised deformations during testing.

In the case of tensile testing of ductile materials, once the stresses exceed the yield point, a contraction of the cross-sectional area occurs, leading to a local increase of the stresses and a decrease of strains in this post-yield region. Since traditional uniaxial tests can not capture localised stresses and strains in the necking region, so-called true stresses (σ) and

true strains (ε) can be calculated from the engineering stresses (R) and engineering strains (e) using the following equations:

$$\sigma = R \cdot (1 + e) \qquad \text{and} \qquad \varepsilon = \ln(1 + e) \qquad (6)$$

Where the engineering stresses and strains are defined as:

$$R = F/A_0 \qquad \text{and} \qquad e = \Delta L/L_0 \qquad (7)$$

with A_0 and L_0 being the initial cross-section and initial measuring length of the specimen, respectively.

The true stress considers the specimen's cross-sectional area at each point, reducing as the material undergoes necking or localised deformation. This accounts for the reduction in the effective area, and results in higher true stress values than engineering stress (Figure 18).

Similarly, the true strain represents the actual deformation of the material within the contracted section, accounting for the change in length (ΔL) relative to the total current length (L). The logarithmic term in the true strain equation accounts for the strain measured relative to the current length of the specimen. The true strains in the plastic deformation range of a material are generally smaller than their corresponding engineering values.

It is important to note that true stresses and strains are equal to engineering stresses and strains in the elastic deformation region of the material, and that the initial true yield stress and engineering yield stresses are the same ($R_y = \sigma_y$).

A more accurate representation of the material's behaviour beyond the yield point is obtained using the true stress and strain values. These true stress-strain values are crucial for capturing the material's response under high loads and plastic deformation accurately.

It's important to note that the transition from engineering stress-strain to true stress-strain assumes a uniform deformation and a Poisson's ratio of 0,5. In reality, localised variations in deformation and Poisson's ratio may influence the behaviour. Nevertheless, true stress and true strain approximate the material's behaviour better during plastic deformation.
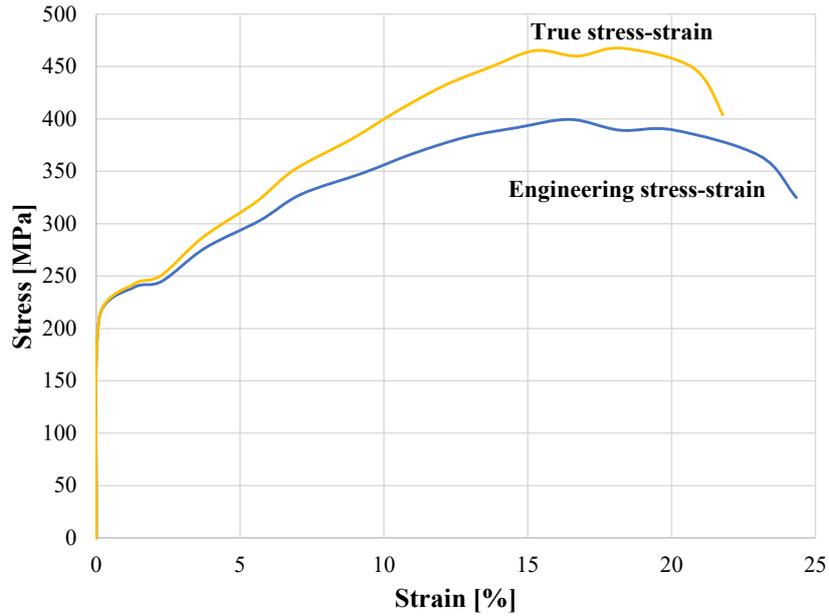
Figure 18: Comparison of engineering and true stress-strain relationships

In computational simulations, when simulating the nonlinear (plastic) behaviour of a material, it is necessary to consider the relationship of true stresses (σ) and true strains (ε) instead of using the approximate engineering values. The true stress-strain relationship provides a more accurate representation of the material's response under plastic deformation.
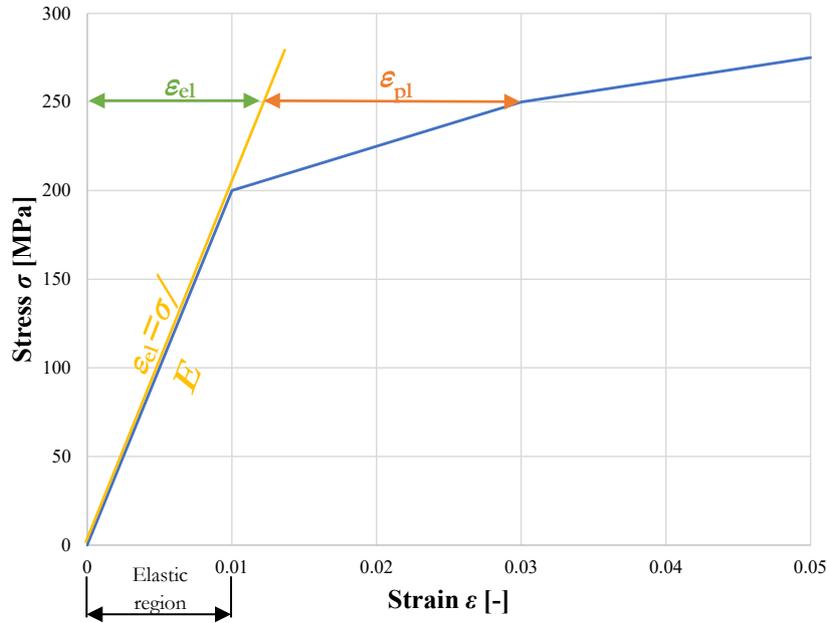
Figure 19: Piecewise linear true stress-strain relationship

The true stress-strain relationships are often represented using simplified, piecewise linear approximate curves in computational models, to capture the nonlinear behaviour. These are typically derived from experimental data, such as the stress-strain relationships from tensile tests. Figure 19 shows an example of the measured stress-strain relationship divided into linear segments with varying slopes. These segments represent different stages of plastic deformation, such as yielding, strain hardening and failure. The piecewise linear representation allows for a simplified approximation of the material's nonlinear behaviour in computational simulations.

Advanced constitutive models, such as multiparametric plasticity models, can capture the highly nonlinear behaviour of specific materials more accurately, and simulate complex material responses in engineering simulations [16].

4 Boundary conditions and loads

The boundary conditions in FEA are conditions applied to the model to simulate the constraints and loading conditions that the actual structure or component would experience in the real world. They are essential for solving the system of equations that govern the behaviour of the structure. They ensure the structure`s equilibrium, and help determine the system's response to applied loads and constraints. Properly defined boundary conditions are critical for obtaining accurate and realistic simulation results in FEA. They define how the structure interacts with its surroundings, and are classified into two main types: displacement boundary conditions and force (load) boundary conditions. In the case of displacement boundary conditions, the degrees of freedom of finite elements are usually constrained, while, in the case of force boundary conditions, the load is applied to the FE nodes.

4.1 Definition of boundary conditions

Defining boundary conditions in FEA is essential for ensuring accurate and meaningful results. In FEA, each known displacement of a structure corresponds to a nodal degree of freedom displacement value, serving as input data for computational analysis. A DOF is assigned a zero value commonly, to indicate that the node is fixed or stationary in that direction. Unassigned DOFs are considered free, and their displacement is determined during the analysis.

In static analysis, the structure must be in equilibrium, necessitating the specification of displacements and rotations in global coordinate axes. Unconstrained DOFs result in an underdefined model, preventing the unique determination of displacements and reactions. Degrees of freedom are typically defined in Cartesian coordinates, but alternative coordinate systems can be employed, such as cylindrical or spherical.

The connectivity of degrees of freedom is crucial, especially in models consisting of multiple parts that must be analysed as a whole. Connections can be established by merging parts, creating common nodes, or retaining separate bodies and adding connections between their respective nodes. These connections represent the interdependencies of degrees of freedom in the computational model, and are usually called constraints. Degree of freedom connections can range from simple ties, where the displacements of connected DOFs are equal, to more complex relationships involving constraints on specific DOFs, or equations describing the interdependence. Attention is required when connecting nodes with different numbers of degrees of freedom, especially in cases where additional rotational degrees of freedom are present.

Errors in boundary conditions can lead to inaccuracies in FEA results. While some errors are detectable through computational analysis, others, such as incorrect stresses or displacements, may be more challenging to identify, and require experimental validation. Special consideration should be given to potential errors near transitions between different mesh densities, and it is advisable to seek boundary conditions that reflect real-world physical conditions better.

Defining boundary conditions on geometric entities such as surfaces, edges, or reference points in CAD models is preferable when prescribing boundary conditions. Using reference points allows for independence from model geometry changes, ensuring boundary conditions persist, even when the model undergoes modifications. The user specifies boundary conditions for a geometric domain, and the software converts them into appropriate conditions for corresponding finite element nodes. The process involves defining the region for assignment, creating a set based on user selection, and altering the specified conditions into suitable boundary conditions for analysis.

4.2   Loads

In FEA, the loads can vary in location and shape. In the Finite Element Method, the location of a load is always associated with a degree of freedom at a node. Modern software always links the location of loads to geometric entities, and pre-processing tools convert

the prescribed loads into equivalent nodal loads according to their DOFs. Loads are not always required for every type of computational analysis. For example, the system's natural frequencies can be obtained in modal analysis without prescribed loads. The user should determine the type of analysis before specifying the loads for a given problem.

The force is the usual type of load, but other types of loads, such as moments, pressures, displacements, velocities, accelerations, temperatures, etc., are also possible. Point loads are localised forces or moments, often applied at joints or critical connection points. Distributed loads, on the other hand, represent forces spread over lines, areas, or volumes. A line load might be used to apply a distributed load across a truss or beam finite element, while a pressure load acts on a surface, and captures the impact of fluids or gases on a structure. A gravity load acts on the whole structure's volume and accounts for the structure's weight. Force and moment loads, whether concentrated or distributed, simulate external dynamic forces and rotational effects essential for assessing structural responses.

Other loads are prescribed displacements, where a selected DOF is set to a value different from 0, or remote loads, which allow for the simulation of forces acting at a distance from the structure. Inertia loads, integral to dynamic analyses, account for mass and acceleration effects. The versatility of these load types enables FEA practitioners to tailor simulations to diverse engineering scenarios, ensuring a thorough investigation of structural performance under various conditions. In general, the most commonly used loads in FEA are:

- **Point Loads**: Concentrated forces or moments applied to specific nodes. Since the load is prescribed to the node and translated to the DOF of the node, the singular solution appears in the node, which is a consequence of the singular solution of the Boussinesq-Cerruti problem in the theory of elasticity [18]. Nevertheless, the FEM operates as an approximation method, relying on piecewise polynomial interpolation functions that struggle inherently to capture the singular solution in the proximity of a node subjected to a point load. However, saint-Venant's principle [19] plays a crucial role by diminishing the influence of the singularity associated with the point load as the distance from the loaded node increases.
- **Distributed Loads**: Real-world loads are generally distributed over specific areas. Consequently, in simulations, these loads are distributed virtually across a defined region within the computational model, transforming them into nodal loads upon discretization. The efficacy of this process depends upon factors such as the number of DOFs associated with the finite element nodes, and the degree of

polynomial interpolation functions applied. Modern pre-processors for FEA typically allow users to associate loads with the geometric model independently of the finite element mesh. The computational model might lack a finite element mesh during the initial stages when users introduce loads. Subsequently, the program correlates the loads with the finite element nodes automatically upon mesh generation.

Understanding the nature of the physical problem and selecting the appropriate type of loads is crucial for obtaining accurate and meaningful results in FEA simulations. Different load combinations can be applied to study complex scenarios and assess the structural response under various conditions. Load units must be consistent with the units of geometry and material properties to obtain results in the desired units, which the user must ensure in most FEA programs.

5   Meshing

5.1   Importing of geometry

Mesh generation, or pre-processing software, often allows for modelling the problem's geometry, usually used for simple geometries. However, often an object that has already been modelled geometrically in 3D modelling software is the subject of computational analysis. This geometric model can be imported easily into an appropriate pre-processing program to prepare the computational model.

The Standard STEP (Standard for the Exchange of Product Data - ISO 10303) is the most suitable format for importing data, which provides a neutral format for representing 3D CAD models and exchanging data between different CAD software systems [20]. It allows for transferring geometric and non-geometric information, making it a versatile choice for interoperability among various CAD applications. A STEP file typically has the .stp or .step extension, but the file extension alone does not provide information about its content. Geometric models can often be imported in formats supported by modelling libraries, such as ACIS (with the .sat extension) and Parasolid (with the .x_t or .x_b extension). Geometric models can also be imported as the IGES Standard, which provides a good description of lines and surfaces, but has limitations in representing volumetric geometry.

Often, errors may occur during geometry import, or the imported model may not suit our needs. Issues may arise when a seemingly appropriate model is insufficient for computational analysis. Therefore, different tools are available in the pre-processing software to prepare the geometry for meshing and avoid errors or low-quality mesh,

5.2   Density of the FE mesh

The definition of the FE type and dimensions follows the successful import of the geometry. The meshing process starts with the meshing of the geometry edges, then propagates to the meshing of the geometry surfaces, and ends with meshing the volume between the surfaces. The simplest method to define the size of an FE is edge-based meshing, where the user defines how many line elements (trusses) the edges will be divided into. The basic options are the definition of the size of the elements, or the definition of the number of elements per edge. Additionally, the size of an FE can be refined around the area of specific interest using the local mesh refinement (bias). Two methods of determining the size of finite elements in PrePoMax software (global mesh size and mesh refinement) can be seen in Figure 20. Edge-based meshing allows easy control over elements' sizes and distribution along a geometric model's edges. The choice between global element size, number of elements, or local mesh refinement, depends on the problem's specific requirements, the object's geometry, and the desired accuracy of the results. The same methods can be used for solid, surface and line finite elements. Each method has its advantages and limitations, and the appropriate choice should be made based on the specific needs of the analysis.
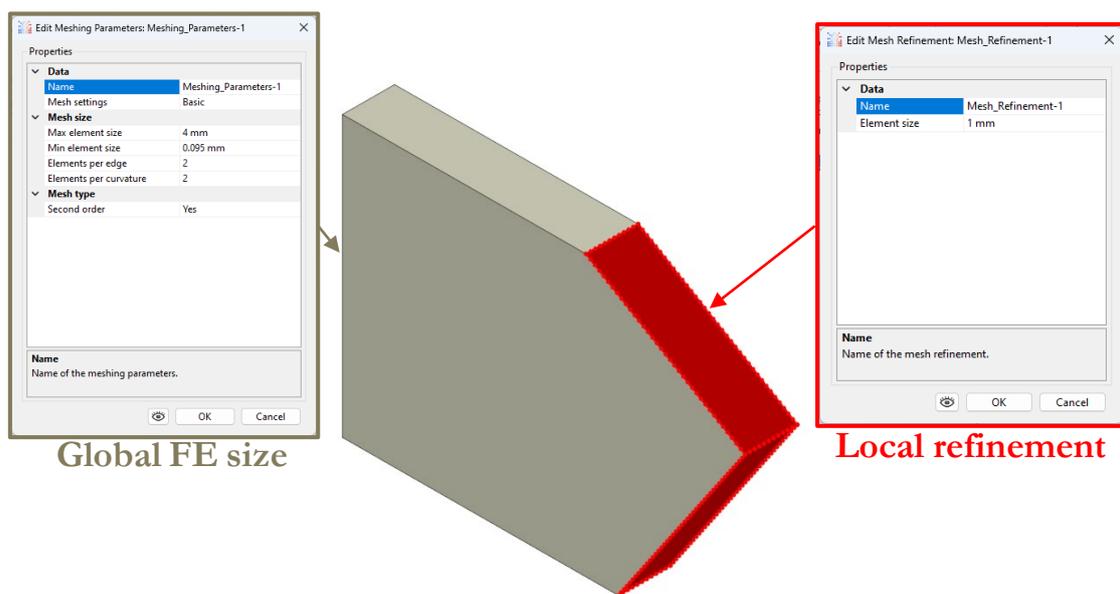

Figure 20: Finite element mesh definition

After defining the global FE size and local refinement (Figure 20), a discretization with precisely defined elements is achieved by specifying a global element size and refinement on two surfaces. Based on the geometry, the program generates approximately the prescribed sized elements, and the generated mesh is shown in Figure 21.
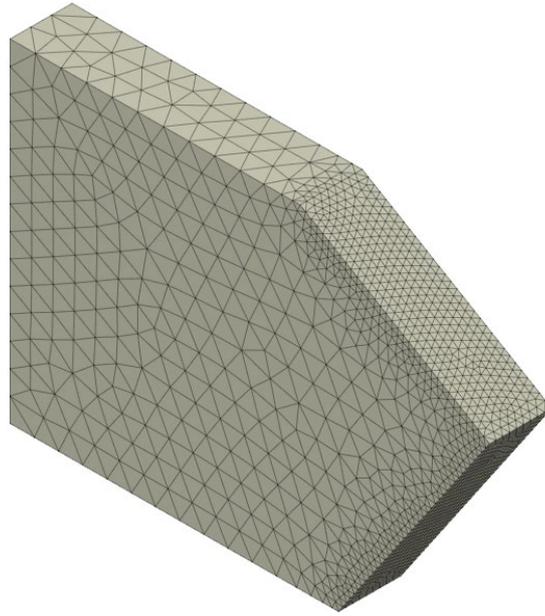

Figure 21: Volume finite elements mesh

An FE mesh size is a critical consideration in computational simulations, and is influenced by many factors. The complexity of the geometry, ranging from simple to complex shapes, dictates the necessary mesh density to capture details accurately. The type of analysis being conducted, whether structural, dynamic, thermal, or fluid flow, guides mesh refinement further. Material properties, boundary conditions, and the desired accuracy of results also play essential roles. The capabilities of the solver and available computational resources impose practical constraints on mesh size. A reasonable balance between computational efficiency and result accuracy is necessary, and it's up to the user to choose the appropriate mesh size. During initial analyses, finite elements are typically large (a small number of elements). As the analyst improves the FE model, the element size is reduced in areas with higher result gradients, increasing the number of elements until reaching the desired convergence of results, which is usually determined with the convergence study. Figure 22 illustrates a convergence analysis using multiple finite element meshes of varying densities for the same object. As the number of finite elements increases, the results converge progressively towards theoretical solutions. Coarse meshes with a limited number of elements exhibit a relatively large error, emphasising the substantial improvement achieved through mesh refinement. Notably, for less dense meshes, displacement values approach correct results faster than stress values. Computational models with more elements exhibit

minimal accuracy differences, showcasing near-identical results as the element count rises. However, caution is warranted, as further mesh refinement may lead to adverse effects, due to computational errors associated with finite computer arithmetic.
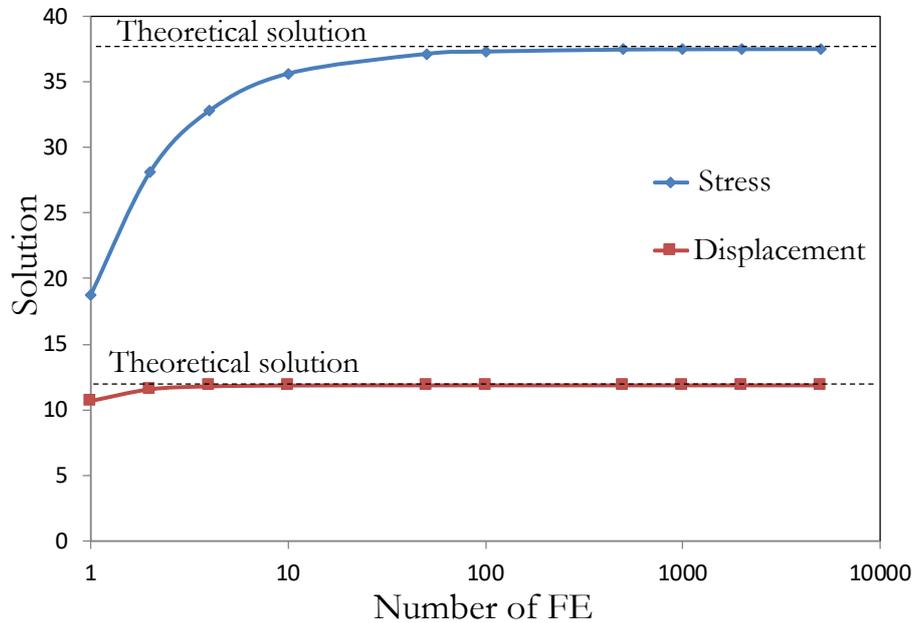

Figure 22: Typical convergence analysis results [2]

When building a Finite Element (FE) model for the first time, using a coarse mesh can be beneficial for several reasons:

1. Simplicity: A coarse mesh consists of fewer elements and nodes than a fine mesh. This simplifies the model, and reduces the computational complexity, making it easier to set up and solve.
2. Computational efficiency: With a coarse mesh, the number of equations to solve is low, resulting in faster computation times. This is particularly advantageous when dealing with large, complex models requiring significant computational resources.
3. Convergence testing: Starting with a coarse mesh allows you to assess the model's behaviour and perform convergence testing quickly. Convergence testing helps determine if the solution is approaching a stable and accurate result as the mesh is refined. By identifying potential issues early on, you can adjust the model's parameters and refine the mesh incrementally.
4. Insight into solution characteristics: Coarse meshes can provide valuable insights into the overall behaviour and trends of the solution. Observing the preliminary results allows you to identify areas of interest, potential errors, or unexpected behaviour that may require further investigation or refinement.

5. Mesh independence study: Using a coarse mesh initially allows for a mesh independence study. By refining the mesh gradually, you can analyse how the solution changes with increasing mesh density. This study helps to determine the mesh size required to achieve accurate and reliable results without excessive computational costs.

However, it's important to note that using a coarse mesh also has limitations. It may not capture small-scale details, or represent localised phenomena in the model accurately. Therefore, once the initial analysis with a coarse mesh is complete, refining the mesh and increasing its density is typically necessary to improve accuracy and capture finer details in the solution.

5.3   Stress singularities

In finite element method analysis, a stress singularity refers to an infinite or highly concentrated stress value at specific points or regions in a structure. These singularities typically arise due to point loads, geometric features, or boundary conditions.

It is important to note that stress singularities appear due to mathematical idealisations and do not exist in reality. In practical terms, they represent regions where the assumptions of the finite element model may break down or fail to capture the true behaviour of the structure accurately. Stress singularities are often regarded as artefacts of the analysis, and their magnitude may be mitigated or eliminated by adopting more advanced modelling techniques.

When analysing structures with stress singularities, it is crucial to be aware of stress singularities and their potential impact on the overall accuracy of the results. Physical factors such as material properties, imperfections, and other phenomena can influence the stress distribution near sharp corners. Therefore, they should be interpreted cautiously, and validated against other analytical or experimental methods to understand the stress distribution and its influence on the structural response properly.

Various techniques can be employed to mitigate the issue of stress singularities, including:

1. Mesh refinement: By refining the mesh near the sharp corners or geometric discontinuities, the stress gradients can be captured better, reducing the effect of stress singularities. This approach involves increasing the number of elements near the singularity to represent the stress distribution accurately.

2. Higher-order elements: Using higher-order elements with more complex interpolation functions can help alleviate stress singularities somewhat. Higher-order elements provide a more flexible solution interpolation, allowing a better representation of stress gradients near corners.

3. Stress singularity removal techniques: Some specialised techniques, such as using the crack tip elements or enrichment functions, can address stress singularities directly. These methods modify the finite element formulation near the singularity, to capture the local stress behaviour more accurately.

5.3.1 Stress singularities at geometric features

The reason for stress singularity at sharp corners is due to the approximation of the solution using finite elements, which assumes a continuous and smooth variation of stress and strain throughout the domain inherently. However, at sharp corners or geometrical discontinuities, such as re-entrant corners or cracks, the stress and strain gradients become infinitely large, leading to stress singularities.

Stress singularities occur because the shape functions used in FEM are typically chosen to be smooth and continuous over each element, but may not capture the abrupt changes or high gradients in stress that occur at sharp corners. As a result, the finite element approximation tends to underestimate the stress concentrations near these singularities.

5.3.2 Stress singularities at boundary conditions

At a boundary condition, especially at a fixed boundary condition which is a fully constrained or immovable boundary condition, stress singularities can occur (they always occur when using 3D finite elements). This is because the fixed support restricts the displacement of the structure, and imposes infinite stiffness at that location. As a result, stress concentrations can develop near the fixed support.

The stress singularity at the support is characterised by an extremely high-stress value, approaching infinity. It manifests typically as a sharp spike or peak in the stress distribution near the boundary, or at specific points adjacent to the fixed support. The singularity is a consequence of the idealised boundary condition (infinite stiffness), and the simplification assumptions in preparing the finite element model.

5.4   Automatic meshing in FE pre/post-processors

Most pre-processing programs can generate finite element meshes automatically, which can be unstructured, structured, or hybrid. Discretising any geometric model using an unstructured finite element mesh is always achievable. Unstructured meshes, often comprised of triangles or tetrahedra, find their basis in Jim Rupert's algorithm [21], pioneered in the early 1990s. This marked a significant departure from the predominantly manual or partially automated mesh generation approaches employed before the advent of finite element mesh modelling. Figure 23 demonstrates an unstructured mesh visually, showcasing its composition with tetrahedral finite elements.
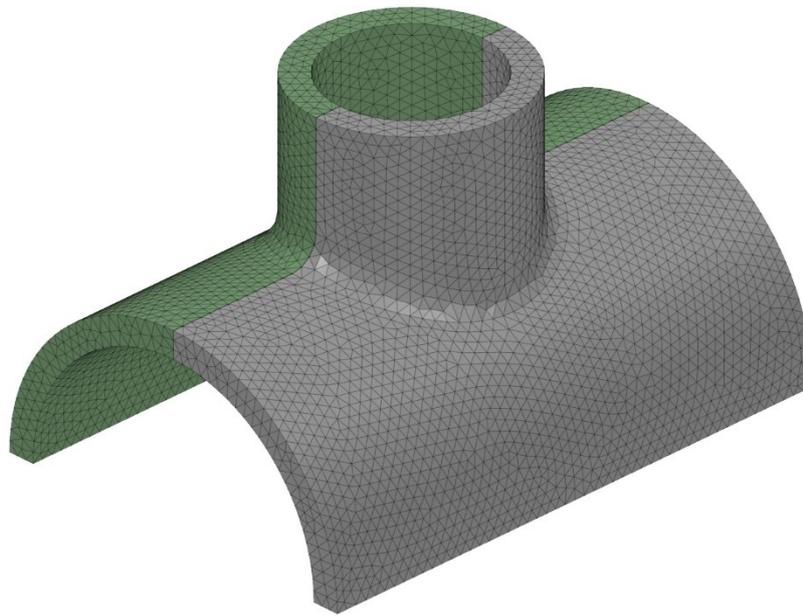

Figure 23: Unstructured FE mesh

Using an unstructured mesh provides flexibility in discretising geometries, adapting well to irregular shapes and complex structures. Comprising tetrahedral or a combination of tetrahedral and hexahedral FE, unstructured meshes are versatile, catering to specific analysis requirements and preprocessor capabilities. However, triangular or tetrahedral finite elements, common in unstructured meshes, may yield less accurate results than quadrilateral or hexahedral elements. This discrepancy often arises from formulating the triangular element as a collapsed quadrilateral with merged vertices. The situation is worsened with quadratic or cubic finite elements, where even more vertices are merged.

Unstructured meshes are suitable for initial analyses, or when time constraints hinder the creation of a higher-quality mesh. On the contrary, the most accurate results, especially in three dimensions, are achieved with structured meshes composed of quadrilateral or

hexahedral finite elements. Figure 24 depicts a structured mesh of hexahedral finite elements, offering better control over element shape and size for increased analysis accuracy in the targeted region. Ideally, a mesh with uniform element sizes produces optimal results, as deviations from this ideal shape introduce errors. While structured meshes generally outperform unstructured ones, the challenge lies in their automatic generation, which is feasible primarily for relatively simple geometric shapes. For more complex models, the geometry must be subdivided into simpler shapes before meshing, to create a structured finite element mesh.
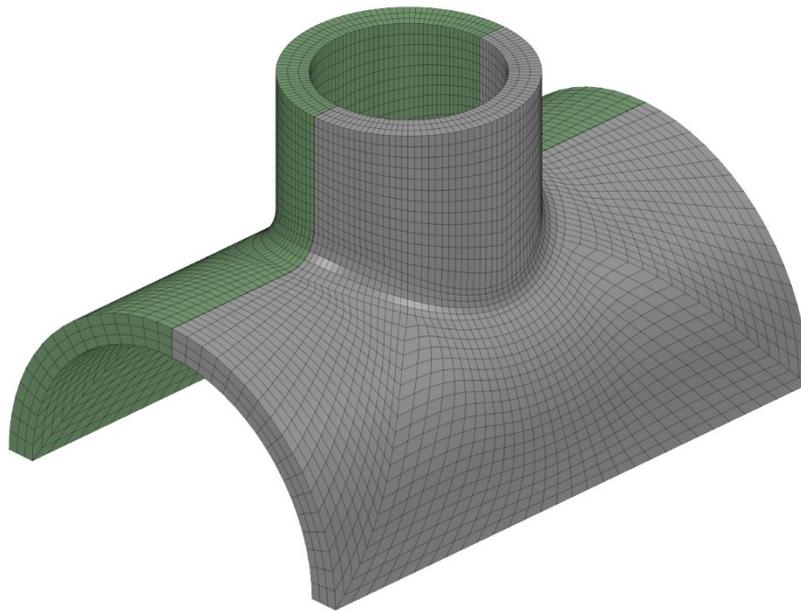

Figure 24: Structured FE mesh

Dividing complex geometric models is a highly challenging task, which is why meshing, despite automated generation, remains the most time-consuming process in the pre-processing phase. Therefore, computational models often consist of hybrid meshes, which combine structured and unstructured meshes, allowing for better adaptation to complex geometries. Figure 25 illustrates an example of a hybrid mesh, where structured elements (hexahedrons) are used in the fillet between the pipes, which can be divided into regular shapes, such as cubes or rectangular prisms easily, and generate a mesh with structured hexahedra. The unstructured mesh (tetrahedrons) is used in areas with higher complexity, where the division is less regular. Three-sided prism elements are needed to connect hexahedral and tetrahedral elements in hybrid meshes. Using hybrid meshes enables a better balance between result accuracy and meshing efficiency, as structured elements are employed where possible for more accurate analysis. In contrast, unstructured elements are utilised for more complex geometry parts. This facilitates faster mesh generation and better adaptation to analysis requirements.
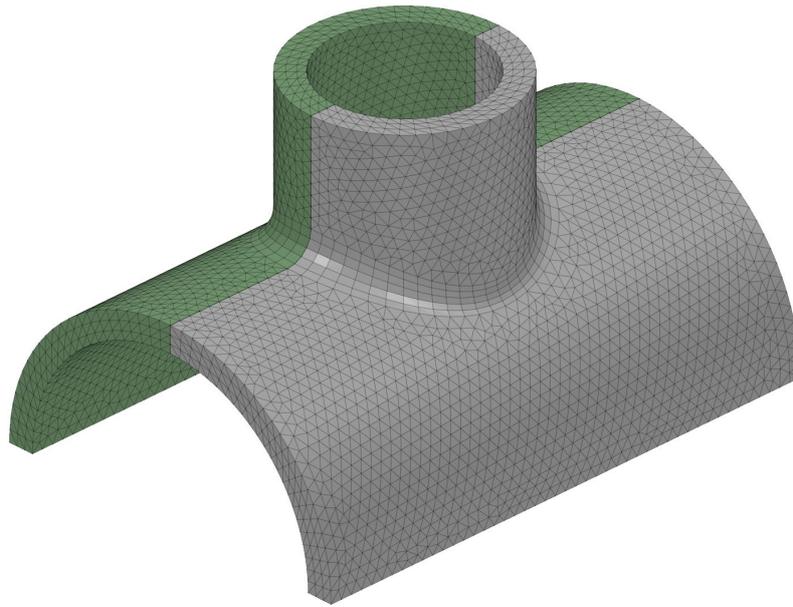
Figure 25: Hybrid FE mesh

The choice between different meshing strategies, including the combination of structured and unstructured meshes, depends on factors such as computational resources, analysis accuracy requirements, and the complexity of the modelled geometry. It is a trade-off between computational efficiency and accuracy, aiming to achieve the best possible balance in the given constraints.

5.4   Quality of finite elements

Evaluating finite element mesh quality is a critical step in ensuring the accuracy and reliability of computational simulations. Various metrics assess the geometric and numerical characteristics of individual elements within the mesh. Key considerations include the aspect ratio, which measures the elongation of elements, skewness deviation from ideal shapes, and the Jacobian ratio, indicating potential element distortion. Perfect finite elements are equilateral triangles, squares and cubes. However, we cannot mesh all models with ideally shaped elements. Due to the adaptation to the actual geometric shape of the domain, an individual finite element can deviate significantly from the ideal shape, leading to less accurate results. Accurate results can be expected only in the case of an ideal-shaped finite element. In extreme cases, the loss of accuracy can be so significant that the results become unusable. In such cases, mesh generation programs usually prevent using highly distorted finite elements. For this reason, mesh generation programs often provide options to set specific thresholds or acceptable ranges for these geometric properties, to ensure that the resulting mesh meets the desired criteria.

By controlling these properties, the mesh generation programs can optimise the mesh, to improve the accuracy and stability of the subsequent analysis. For example, excessive taper or highly distorted elements can lead to computational instabilities or inaccurate results. The program can avoid generating problematic elements and produce a higher-quality mesh by setting appropriate limits on these properties.

It's important to note that the acceptable ranges for these geometric properties may vary, depending on the specific analysis requirements and the characteristics of the problem being solved. Therefore, users have the flexibility to adjust these settings based on their needs and the expected behaviour of the finite element model.

6   Practical examples

6.1   Basics of using PrePoMax

In the following chapters, practical examples are described in which the procedure of preparing a finite element model is demonstrated, together with running the analysis and analysing the results. The practical examples are prepared and analysed using the open-source program PrePoMax. PrePoMax is a pre- and post-processor for the Finite Element Method, which uses the open-source finite element solver CalculiX.

PrePoMax does not support modelling or preparation of the geometry, so the geometry for the finite element models must be prepared using other CAD systems. The geometry can be transferred to PrePoMax using the geometry file formats STEP and IGES, or the STL file format usually applied with 3D printing.

6.1.1   Downloading PrePoMax

To use PrePoMax, download the zip compressed container from the PrePoMax home page: https://prepomax.fs.um.si/downloads. Move the downloaded file into a non-system folder (a folder where the user has all the security permissions) and extract it. The extracted folder PrePoMax v1.4.0 contains five subfolders; the lib folder contains the program dynamic library files, the Models folder contains some general geometry models, the NetGen folder contains the finite element mesher, the Solver folder contains the CalculiX finite element solver, and the Temp folder, where temporary files and analysis

results will be stored. No other installation process is necessary. Run the executable file PrePoMax.exe to start the program and open the main control window (Figure 27).

6.1.2 Structure of the main window

The main window is divided into five active areas (Figure 26), which are:

1. Main menu,
2. Toolbar menu,
3. Feature tree,
4. 3D view and
5. Data output control.

The Main menu and its submenus contain all the features and commands implemented in the user interface. The most common commands have their shortcuts depicted as icons in the Toolbar menu. A graphical representation of the model setup is shown in the Feature tree, while the 3D view is used to visualise the 3D model, mesh and results. During feature and command executions, some information is printed in the Data output control.
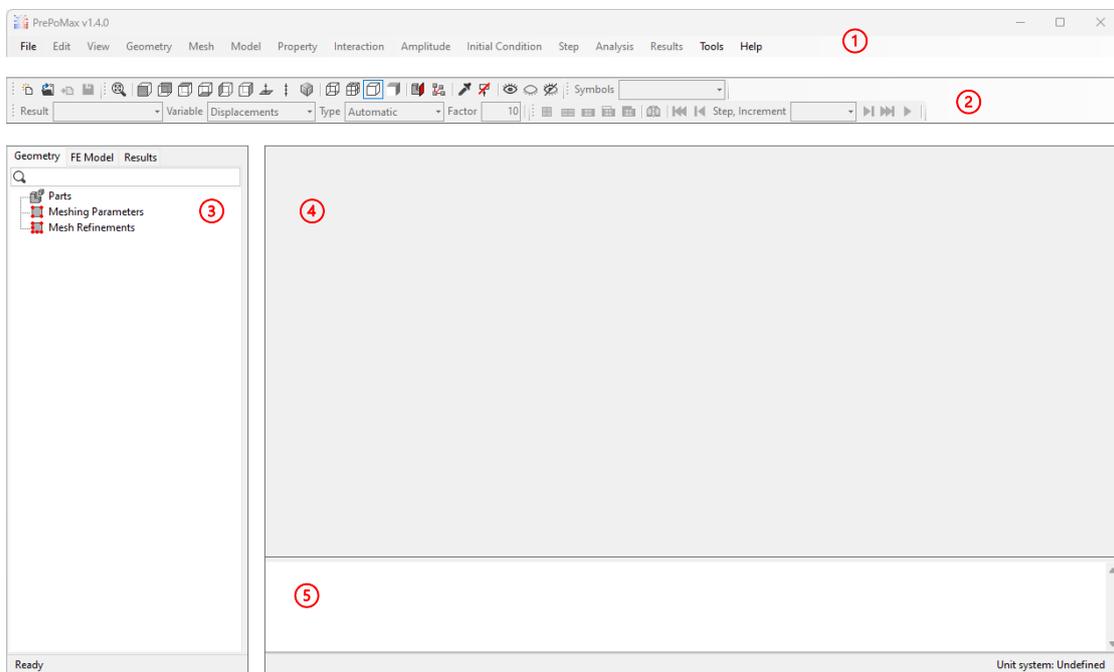

Figure 26: Five active areas of the PrePoMax main window

6.1.3   PrePoMax modules

The finite element model preparation is divided into three major steps representing the three main modules in PrePoMax. These modules are (Figure 27):

1. Geometry
2. FE Model
3. Results

The Geometry module is used for importing the geometry, analysing the geometry, preparing the local and global mesh sizes and starting the meshing procedure. After meshing, the module is changed automatically to the FE Model module. In this module, all features of the finite element model are added to the mesh, such as materials, sections, constraints, contacts, analysis steps with boundary conditions and loads. After the model is completely prepared, the analysis can be started. PrePoMax writes the model into the inp file (CalculiX input file), starts the solver CalculiX, which reads the inp file, solves the analysis and writes the results into the frd file (CalculiX output file). The results file is then read into PrePoMax, and visualised in the Results module. To change between modules, click on the tab that has the module name written in it.
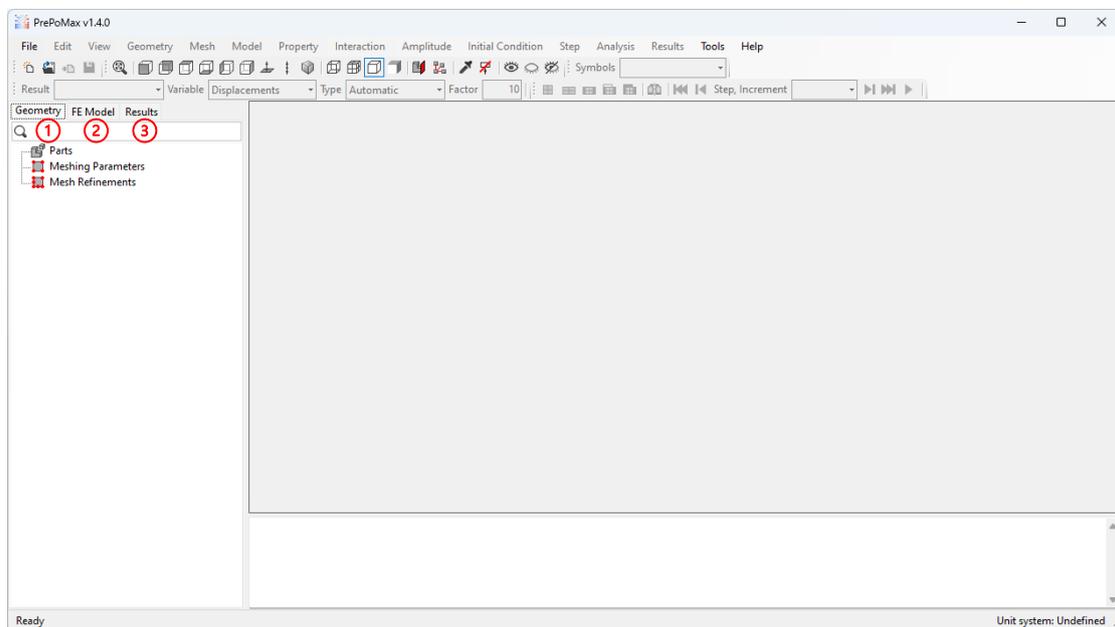

Figure 27: PrePoMax main window

6.1.4 View manipulation

Once a model is opened in PrePoMax it will be visualised in a 3D view. The view manipulation is performed using the mouse in the following ways:

1. Rotation: hold down the middle mouse button while moving the mouse.
2. Set rotation centre: click the middle mouse button once over the model geometry.
3. Reset rotation centre: click the middle mouse button once over an empty region of the 3D view.
4. Zoom: scroll the mouse wheel.
5. Pan: hold the Shift button on the keyboard and the middle mouse button while moving the mouse.

The view manipulation toolbar (Figure 28) contains shortcuts to the following commands:

1. Zoom to fit: scale and centre the model to the 3D view.
2. Front view: rotate the view to show the model from the front.
3. Back view: rotate the view to show the model from the back.
4. Top view: rotate the view to show the model from the top.
5. Bottom view: rotate the view to show the model from the bottom.
6. Left view: rotate the view to show the model from the left.
7. Right view: rotate the view to show the model from the right.
8. Normal view: rotate the view to point one axis perpendicular to the monitor.
9. Vertical view: rotate the view to point one axis in the vertical direction.
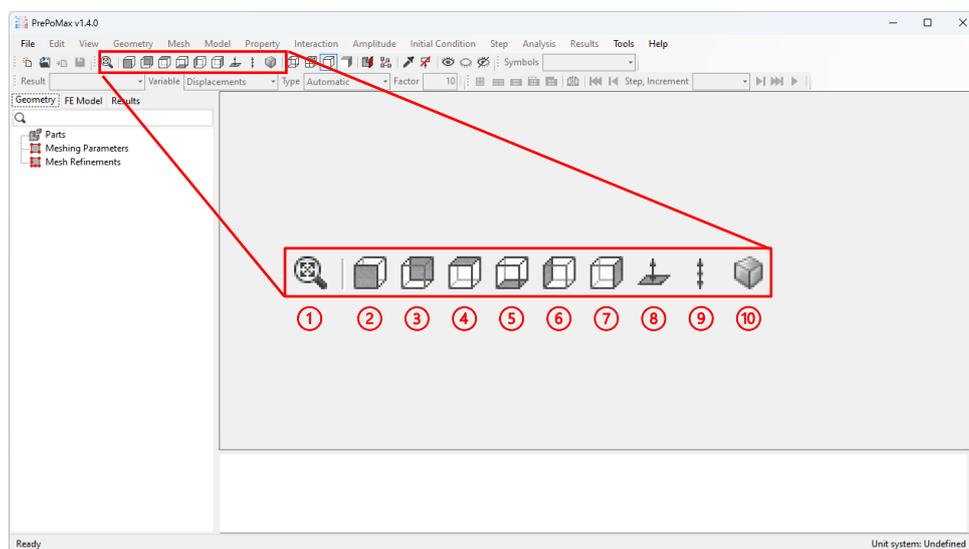10. Isometric view: rotate the view to show the model in the isometric view.

Figure 28: View manipulation toolbar

6.1.5   Visualisation representation

The model can be visualised in four different ways, depending on the requirements of the specific module and user actions (Figure 29):

1. Wireframe: only the model edges are shown (model edges split the model into surfaces).
2. Show Element Edges: the model is shown using shaded surfaces with element edges drawn over them (in the Geometry module the element edges represent the triangulation of the geometry for the s, while in other modules the element edges represent the edges of the finite elements in the mesh).
3. Show Model Edges: the model is shown using shaded surfaces with model edges drawn over them.
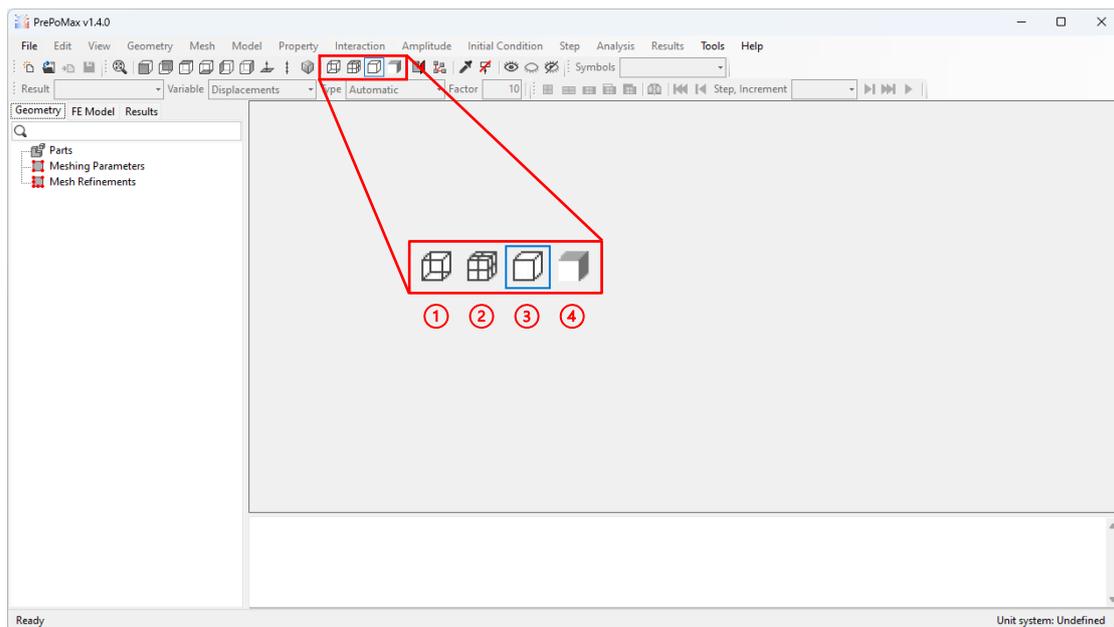4. No Edges: the model is shown only using shaded surfaces without any edges.

Figure 29: Visualisation representation

6.1.6   Results` representation

The results can be visualised only in the Results module in five different ways, depending on the requirements of the user actions (Figure 30):

1. Undeformed: the undeformed model is shown without results.
2. Deformed: the deformed model is shown without results.

3. Deformed with colour contours: the deformed model is shown with the colour contours of the selected field output.

4. Show undeformed wireframe model: the deformed model is shown with the colour contours of the selected field output, and an undeformed model in a wireframe representation is superimposed on it.
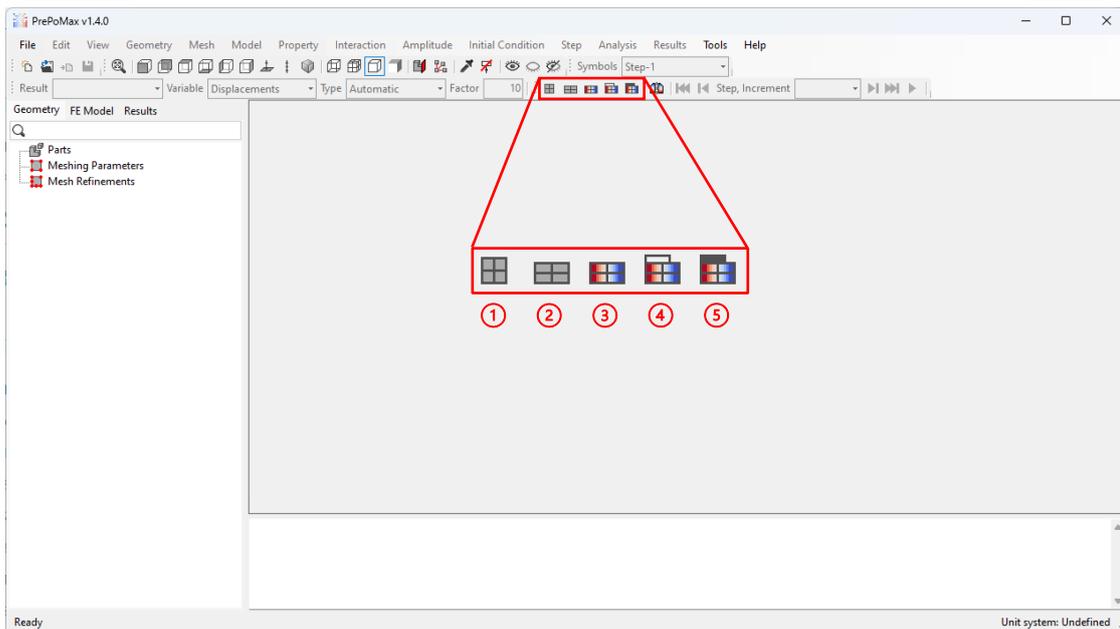
5. Show undeformed solid model: the deformed model is shown with the colour contours of the selected field output, and an undeformed model in a solid representation is superimposed on it.

Figure 30: Results` representation

6.1.7 Troubleshooting

If the main window does not open, or there are any problems while using the software, moving the program folder to a folder without any special characters like C:\Fem\PrePoMax is recommended.

6.2   Beam in Tension

The first practical example is a beam in tension, as shown in Figure 31. The beam is fixed to the wall on one side and loaded in an axial direction with a tensile force of $F = 10$ kN. The beam is made of S235 carbon steel, with the following elastic material parameters: Young's modulus $E = 210$ GPa and Poisson's ratio $\nu = 0.3$. The yield stress of the S235 material is $\sigma_y = 235$ MPa. It is necessary to determine the maximum stresses and displacement in the beam.
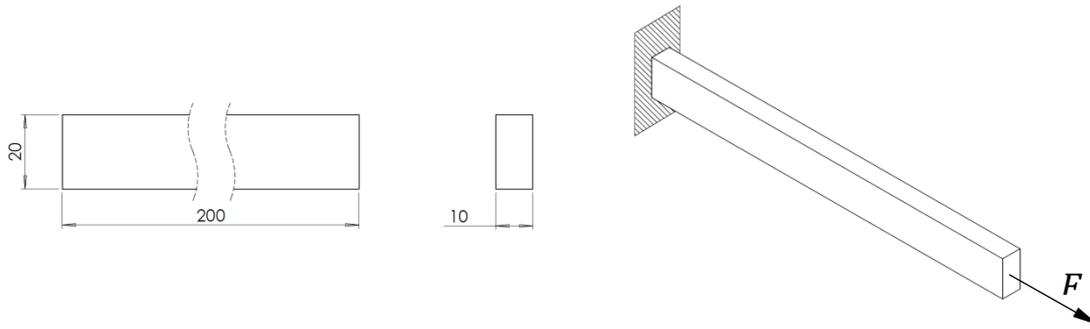
Figure 31: Beam in tension

6.2.1 Analytical solution

The beam example is formulated so that an analytical solution can be obtained. The 3D beam model is first simplified into a 1D beam model, shown in Figure 32. For such a model the internal normal tensile stress σ_n can be computed using Equation 8, while the beam elongation Δl is determined by Equation 9.

The resulting normal tensile stress of 50 MPa, which is constant through the whole length of the beam, is lower than the material yield stress, which shows that the beam can withstand the applied load elastically.
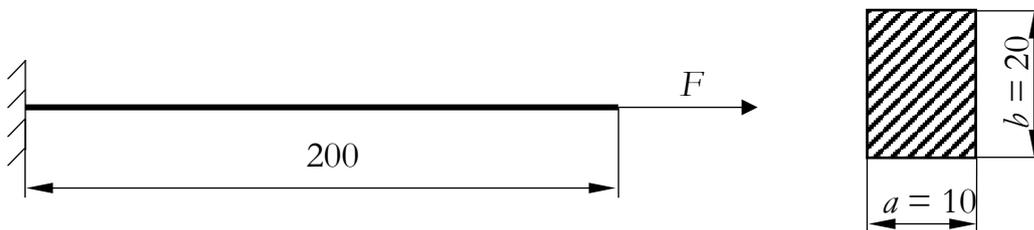

Figure 32: 1D model of the beam

$$\sigma_n = \frac{F}{A} = \frac{10000 \text{ N}}{200 \text{ mm}^2} = 50 \text{ MPa}$$

$$A = a \cdot b = 10 \text{ mm} \cdot 20 \text{ mm} = 200 \text{ mm}^2$$

(8)

$$\varepsilon = \frac{\Delta l}{l_0} \quad \sigma_n = E \cdot \varepsilon$$

(9)

$$\Delta l = \frac{\sigma_n \cdot l_0}{E} = \frac{50 \text{ MPa} \cdot 200 \text{ mm}}{210000 \text{ MPa}} = 0.0476 \text{ mm}$$

6.2.2 Finite element solution

Preparing a new model

A model space and the system of units must be selected to start working with a new finite element model. The beam model will be analysed as a solid 3D model, and the unit system will be used using mm for the length and N for the force. Apply the following steps to create a new model in PrePoMax (Figure 33):

1. After starting PrePoMax select File → New in the main menu to open the Model Properties dialog.
2. Select 3D as the Model Space.
3. Select mm, t, s, and °C as a Unit System Type.
4. Click the OK button to confirm the new model preparation.

Importing geometry

The geometry will be imported in the next step. The geometry for this example was prepared in advance, and is saved in the Models subfolder of the PrePoMax folder. The file is named "Beam in tension.STEP". The geometry import is done by following the next steps:

1. Select File → Import in the main menu to open the Open file dialog.
2. Browse to the location of the Models subfolder ../PrePoMax v1.4.0/Models.
3. Select the file "Beam in tension.STEP".
4. Click the OK button to confirm the import of the geometry file.
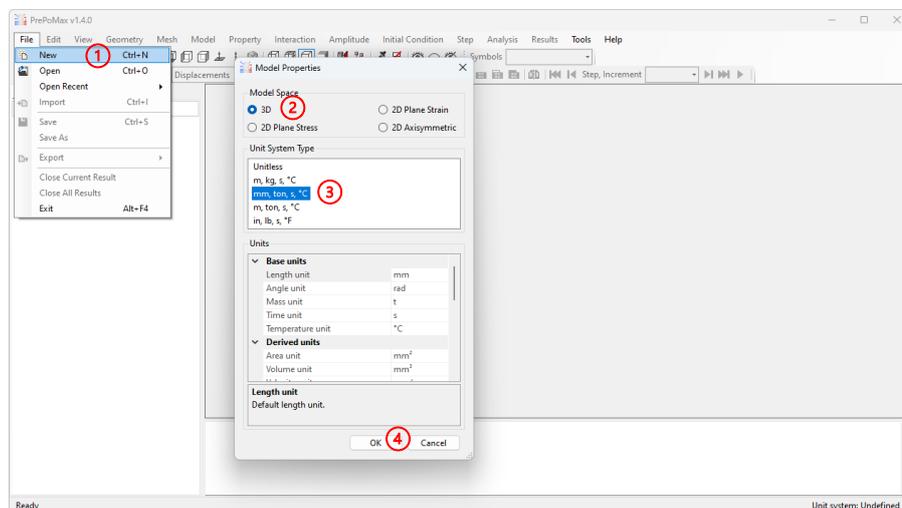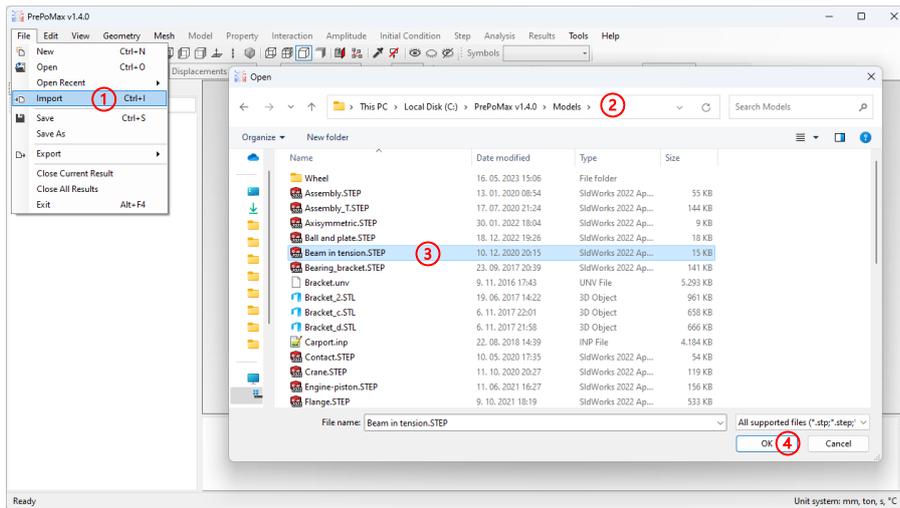

Figure 33: Preparing a new model

Figure 34: Importing geometry

Changing the part properties

By default, the imported parts are given a generic name, such as "Solid_part-1", "Solid_part-2", etc. Renaming the parts is a good practice, especially when analysing assemblies with multiple parts. The parts' names and colours can be changed by editing the parts` properties. For this example, change the name of the imported part to "Beam" using the following steps (Figure 35):

1. Select the imported part in the Geometry feature tree or in the 3D view by right-clicking it to open the context menu.
2. Select Edit from the context menu to open the Edit Part window.
3. Select the Name data field and enter a new name for the part "Beam" (all names inside PrePoMax can only use letter characters, number characters, the underscore character _ and the hyphen character -).
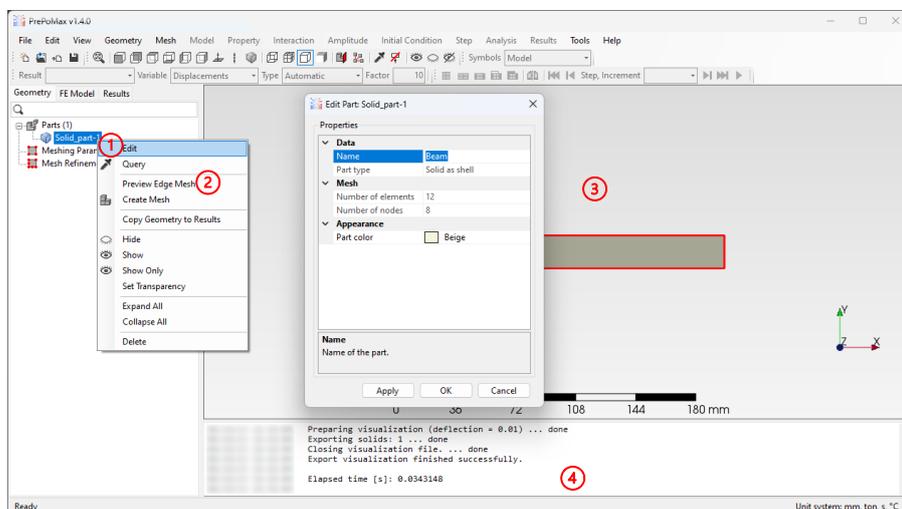4. Click the OK button to confirm the newly set part properties.

Figure 35: Changing the part properties

Definition of the meshing parameters

The parameters of the required finite elements for all parts must be defined before generating the finite element mesh. If the user creates no mesh parameters, default mesh parameters will be used. Mesh parameters can be set for each part separately, or they can be set for multiple parts at the same time. Mesh parameters represent the global mesh properties, while the mesh size can also be set locally to refine the mesh in the areas of interest.

First, a coarse mesh (Density of the FE mesh) will be used for this example, defined by the global meshing parameters. Set the global maximum mesh size to 10 mm and minimum global mesh size to 0.2 mm, and set the element interpolation functions to second order using the following steps (Figure 36):

1. Select Meshing Parameters in the Geometry feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Meshing Parameters window.
3. To set the meshing properties for the part named "Beam" select the part in the 3D view by clicking on it.
4. Rename the meshing parameters in the Name data field to "Beam".
5. Set the Max element size data field to 10 mm.
6. Set the Min element size data field to 0.2 mm.
7. Set the Second order data field to Yes.
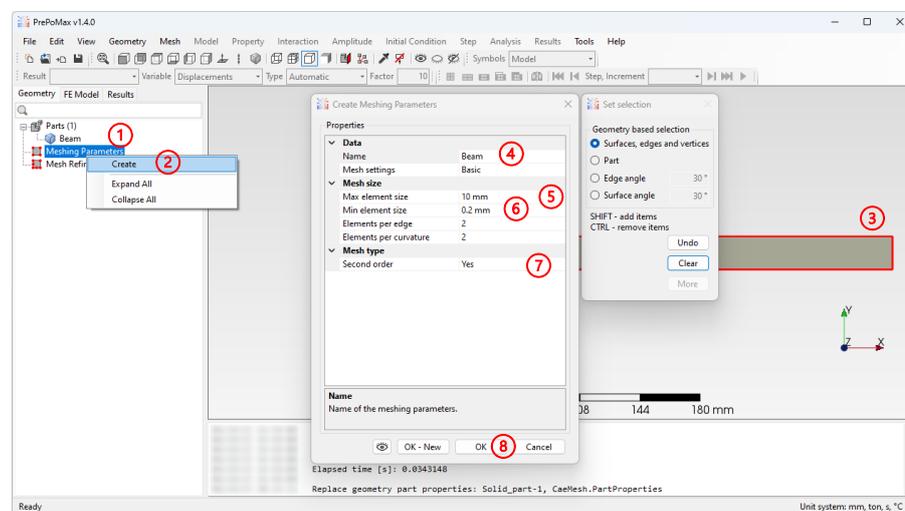8. Click the OK button to create the meshing parameters.

Figure 36: Setting the meshing parameters

Creating the finite element mesh

Once the meshing parameters are set, the part mesh can be generated by (Figure 37):

1. Select the Beam part in the Geometry feature tree, or in the 3D view by right-clicking it to open the context menu.
2. Select the Create Mesh from the context menu to generate the mesh (mesh generation for complex models can be time-consuming, so using the Preview Edge Mesh command can inspect how the model edges will be divided into finite elements quickly).
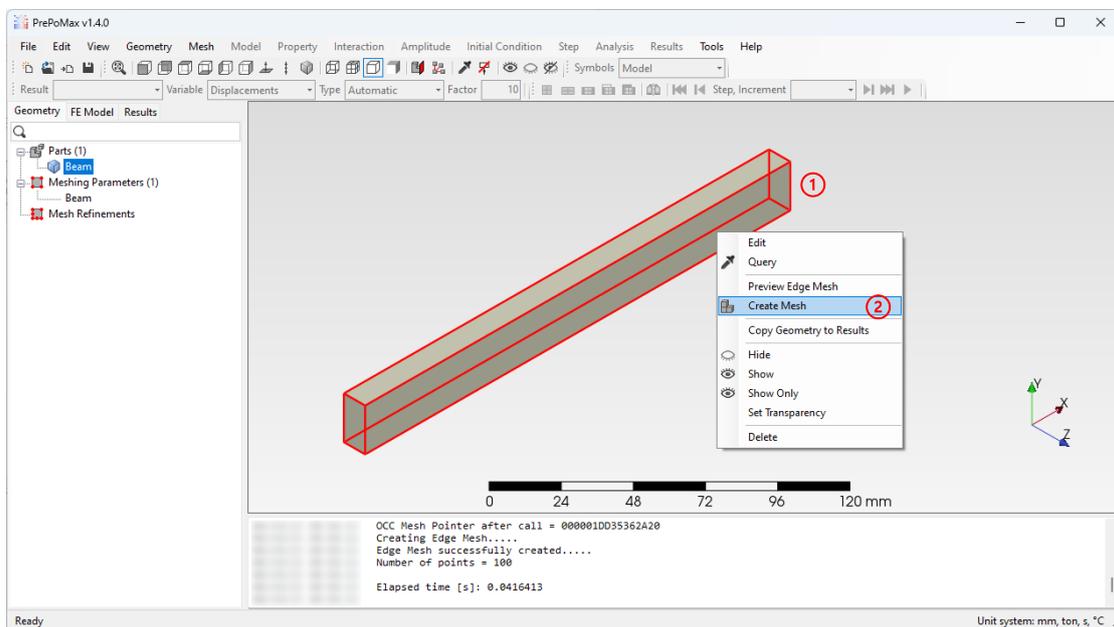

Figure 37: Creating the finite element mesh

Querying the part mesh properties

After creating the mesh, the program changes the Geometry module automatically to the FE Model module and displays the generated mesh. The properties of the part mesh can be inspected by editing the mesh part properties (Figure 38):

1. Select the "Beam" part in the FE Model feature tree or in the 3D view by right-clicking it to open the context menu.
2. Select Edit from the context menu to open the Edit Part window.
3. Check the Number of elements data field to get the number of elements in the selected part: 344.

4. Check the Number of nodes in the data field to get the number of nodes in the selected part: 820.
5. Check the Parabolic tetra type data field to get the finite element type: C3D10 (C – continuum, 3D – three-dimensional, 10 – the element has 10 nodes).
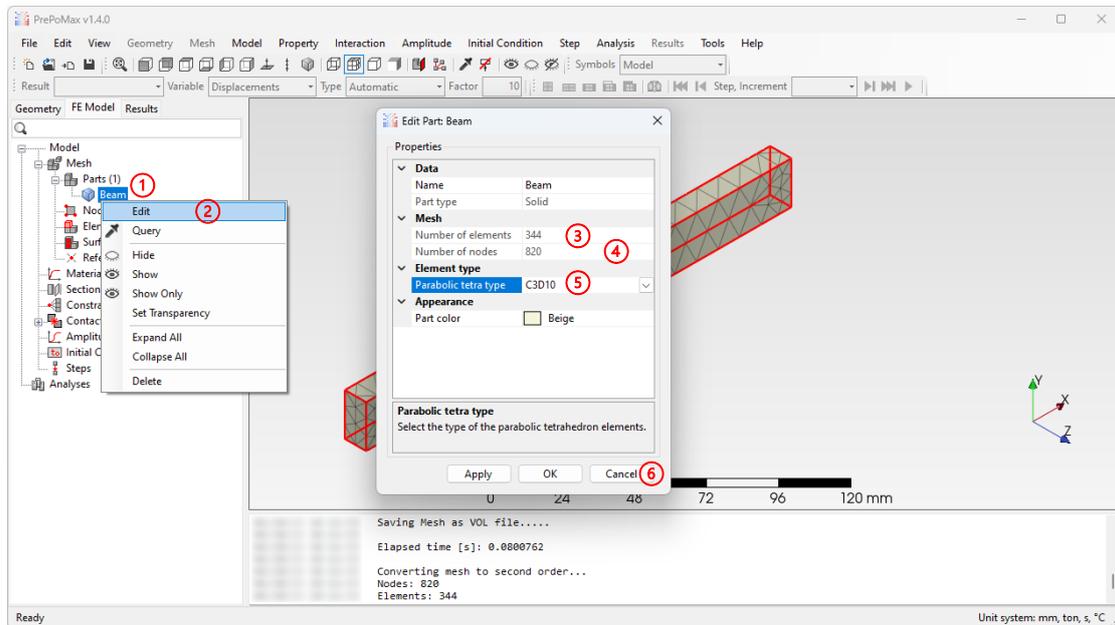6. Click the Cancel button to close the Edit Part window.

Figure 38: Querying the part mesh properties

Adding a material model

For most static engineering problems the material can be modelled using the linear-elastic material model. This model is defined only by two material parameters, namely, Young's modulus and Poisson's ratio. Such a model describes the material behaviour correctly up to the limit of the elastic behaviour denoted by the yield stress. Yield stress is usually not entered as a material parameter in the linear elastic FEA. It is used in the post-processing step, to check if the computed stresses are lower or higher than the yield stress, thus indicating if linear elastic analysis is feasible. The linear elastic model can be added to the FEM model in the following way (Figure 39):

1. Select Materials in the FE Model feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Material window.
3. In the Material name field, enter the name S235.
4. Click Elastic from the Available material models to select it.

5. Click the right arrow button to transfer the available model to the selected models.
6. Select the Young's modulus data field and enter 210 GPa (data fields with units support unit conversions 210 GPa = 210000 MPa; any unit entered is converted to the selected unit system type shown in the bottom right corner of the main window).
7. Select the Poisson's ratio data field and enter 0.3.
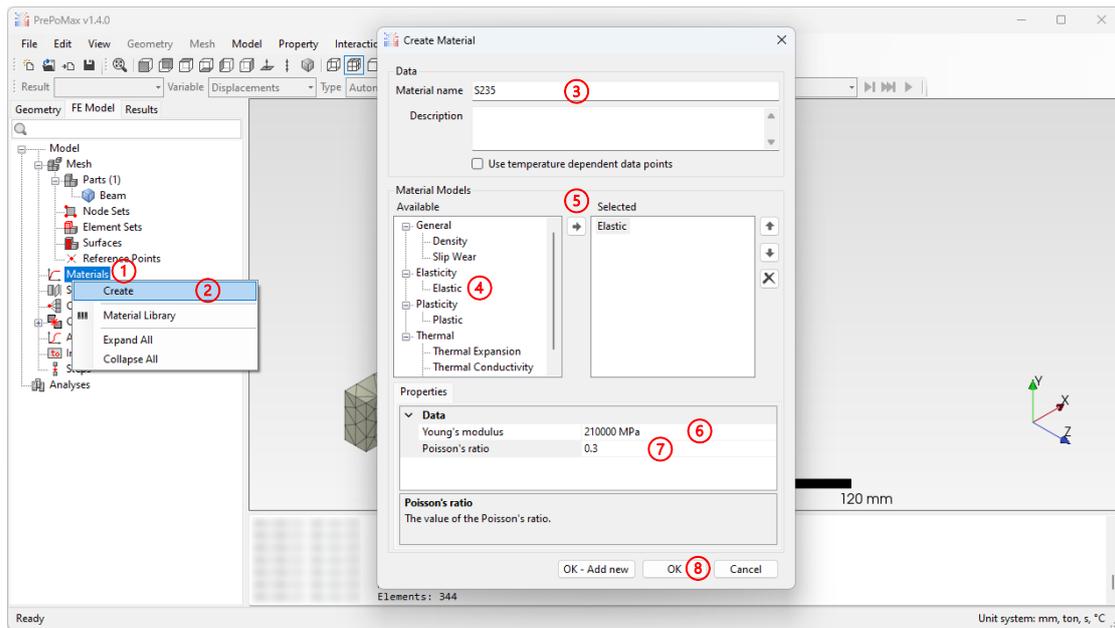8. Click the OK button to confirm the material creation.

Figure 39: Adding a linear-elastic material model

Creating a section assignment

In one finite element model multiple materials can be defined, since different parts of an assembly can be made from different materials. To connect a material property to the part of an assembly a section assignment must be prepared, even if the assembly is comprised of only one part. A section assignment is generally also used to prescribe additional physical properties to parts, such as thickness in the case of shell finite elements or section properties, and section orientation in the case of beam finite elements. The section assignment is created by the following steps (Figure 40):

1. Select Sections in the FE Model feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Section window.
3. In the Name data field, enter the section name Solid_S235.

4. In the Material data field, select material S235 from the drop-down menu.
5. In the 3D view, select the geometry of the Beam part to which this section will be assigned.
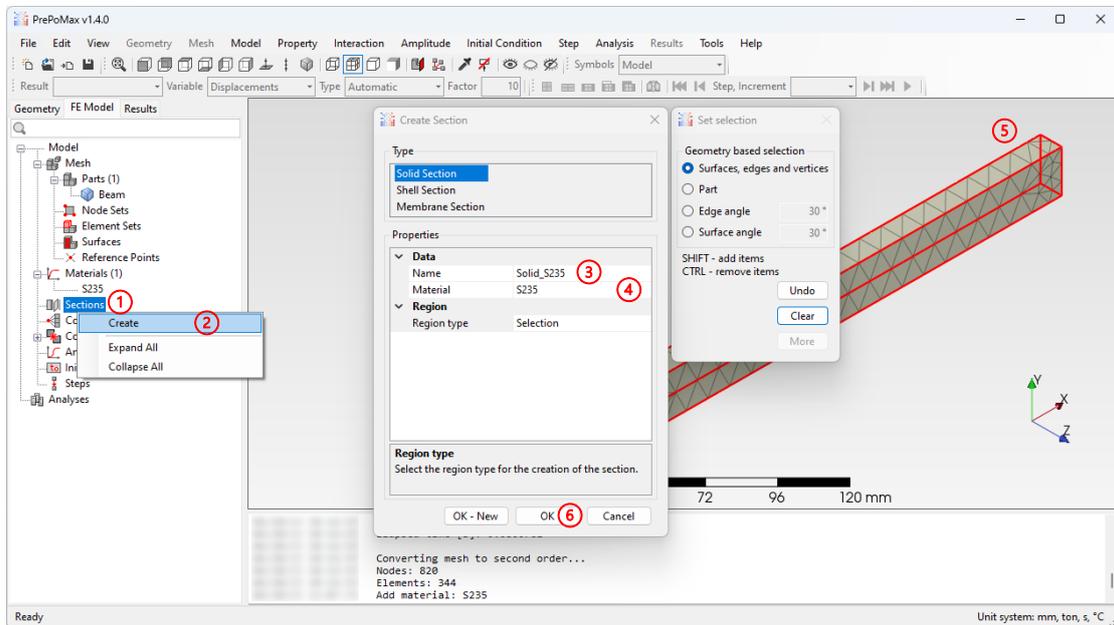6. Click the OK button to confirm the section creation.

Figure 40: Creating a section assignment

Adding a simulation step

Actual structures are exposed to many loading conditions through the time of their use, and each loading condition can be divided into multiple loading phases. Multiple loading phases can be modelled as time-dependent loads, or using multiple analysis steps that follow one after the other. These analysis steps can represent different solution procedures, like a static, linear, or nonlinear dynamic procedure. For the Beam model, a default static analysis step will be created by the following steps (Figure 41):

1. Select Steps in the FE Model feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Step window.
3. In the Name data field, enter the step name: Static.
4. Click the OK button to confirm the step creation.

Depending on the type of the analysis step, a new entry is added to the FE model feature tree under Steps. A static analysis step contains the following feature collections: History Outputs, Field Outputs, Boundary Conditions (BCs), Loads and Defined Fields, into

which new items can be added. The History Output collection defines which results should be stored for individual nodes or elements. The Field Output collection is used to define which results should be stored for all nodes and elements of the model, the BCs collection is used to define the model's boundary conditions, the Loads collection is used to define the loads acting on a model, and the Defined Fields collection is used to define scalar fields for selected nodes.

A default step configuration contains two Field Outputs. A nodal field output, named NF-Ouptut-1, where outputs of the displacements and reaction forces in all nodes is requested, and an element field output named EF-Output-1, where outputs are requested of the stresses and strains in all the elements.
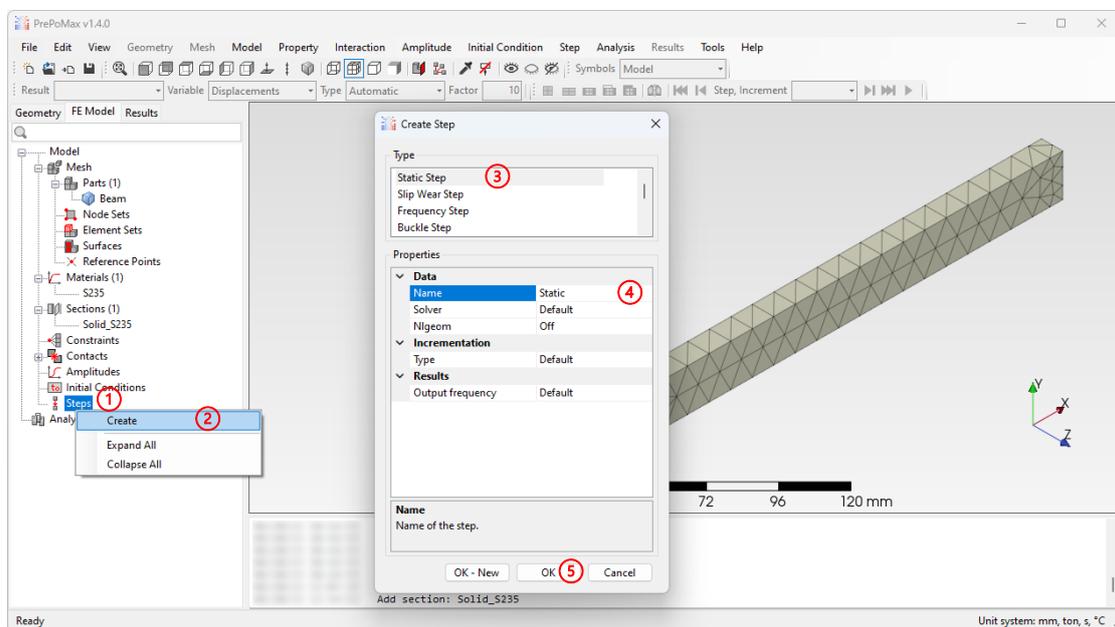

Figure 41: Creating a static analysis step

Creating a displacement/rotation boundary condition

In a 3D space there are 6 independent degrees of freedom (DOF), 3 translations and 3 rotations. In a static analysis, a static equilibrium problem can only be solved if all parts of the assembly are supported in all directions to prevent free body motion – all DOF must be defined as zero or non-zero values in at least one node on each part of the assembly, thus generating the reaction forces at these nodes to assure static equilibrium.

In case of the beam in tension, one side surface of the beam is fixed to the wall. In the analysis, a fixed support will replace this connection to the wall. For this reason, a fixed boundary condition exists in PrePoMax, but a displacement/rotation boundary condition

will be used to demonstrate a more general approach. Using a displacement/rotation boundary condition, each DOF of the selected nodes can be prescribed individually. Nodes can be unconstrained (free to move), and their position can be fixed to 0 mm (a homogeneous boundary condition), or equal to a prescribed nonzero displacement (a non-homogeneous boundary condition). The translational DOFs of all nodes on the beam surface connected to the can be set to 0 mm using the following steps (Figure 42):

1. Select BCs in the FE Model feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Boundary Condition window.
3. Select Displacement/Rotation as the boundary condition type.
4. In the Name data field, enter the boundary condition name: Wall.
5. In the U1 data field (U – displacement, 1 – the first axis of the coordinate system) enter the value of the prescribed displacement: 0 mm.
6. In the U2 data field, enter the value of the prescribed displacement: 0 mm.
7. In the U3 data field, enter the value of the prescribed displacement: 0 mm.
8. In the 3D view, select the side surface of the beam oriented towards the negative x-axis direction.
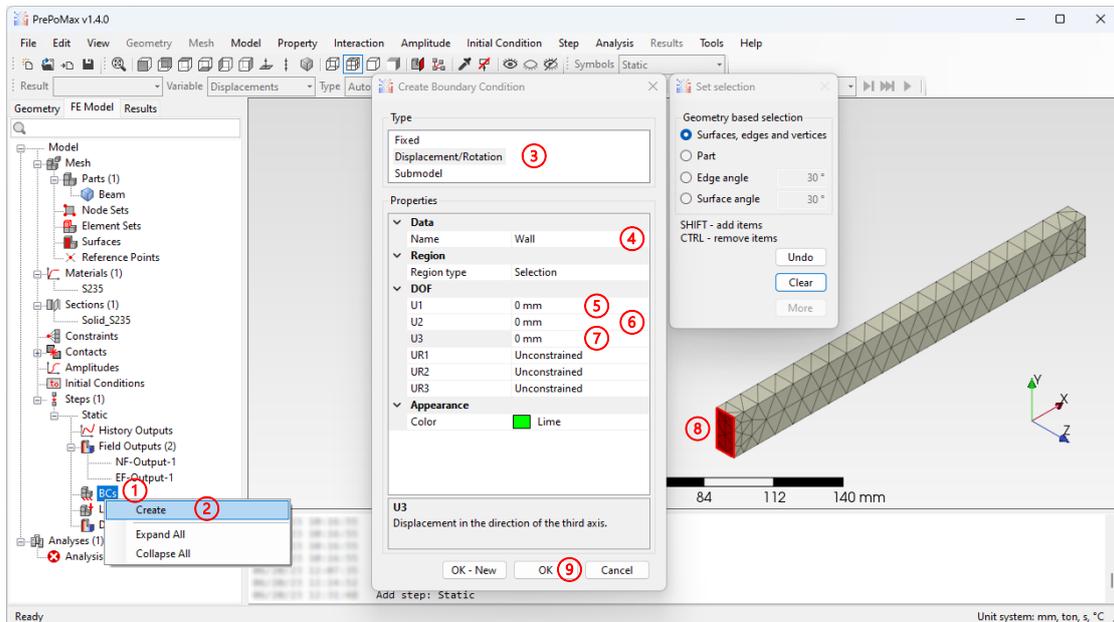9. Click the OK button to confirm the boundary condition creation.
10.

Figure 42: Creating a displacement/rotation boundary condition

Defining a surface traction load

The beam in tension is loaded by an axial tension load acting on one side surface of the beam. Since the whole surface is loaded, a surface load type must be applied to it. Two types of surface loads can be prescribed in PrePoMax. One type is a pressure load, where the magnitude of the load must be defined while the load direction is defined by the loaded surface normal, and the second type is a surface traction load, where both the magnitude and direction must be prescribed. Here, both load types could be used (to apply the pressure load first, the pressure magnitude would have to be computed), but the surface traction load will be added to the analysis step using the following steps (Figure 43):

1. Select Loads in the FE Model feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Load window.
3. Select Surface Traction as the load type.
4. In the Name data field, enter the load name: Force.
5. In the F1 data field (F – force, 1 – the coordinate system's first axis), enter the axial load value: 10 kN.
6. In the 3D view, select the side surface of the beam oriented towards the positive x-axis direction.
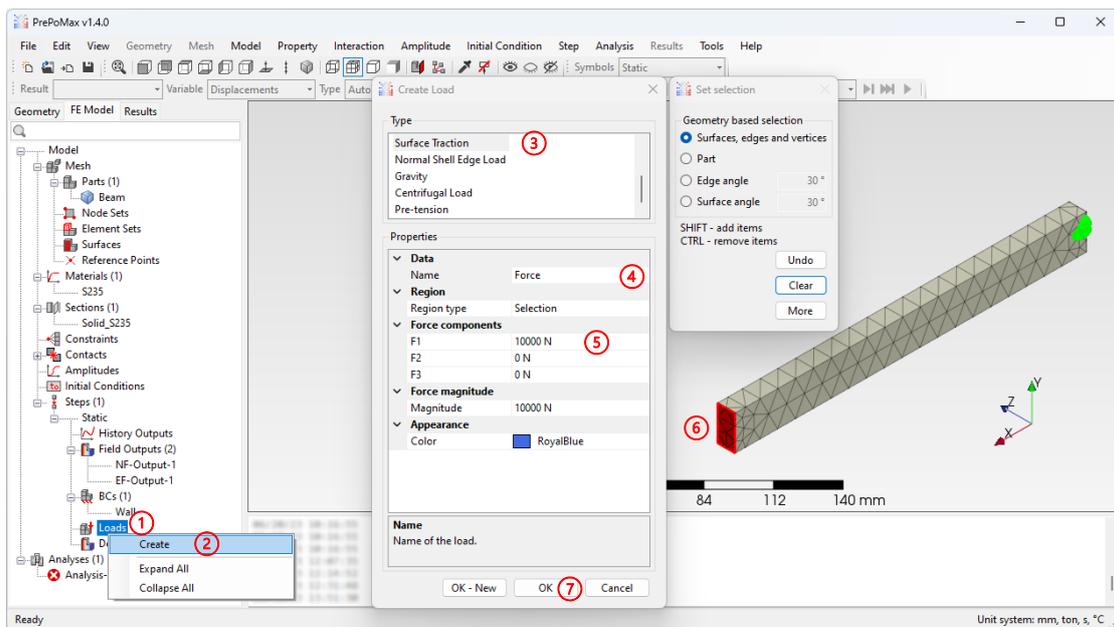7. Click the OK button to confirm the load creation.

Figure 43: Creating a load

Analysis setup

Once the setup of the finite element model is finished, the analysis must be prepared before it is started. An analysis named Analysis-1 is created by default when the first step is added to the steps` collection. To rename the analysis and determine the solver and the work folder that will be used for the analysis, follow the steps:

1. Select Analysis-1in the Analyses collection of the FE Model feature tree by right-clicking it to open the context menu.
2. Select Edit from the context menu to open the Edit Analysis window.
3. In the Name data field, enter the analysis name: Beam-Static.
4. The solver and its path that will be used to run the analysis is displayed under the Executable data field.
5. The work folder that will be used to store temporary and result files is displayed under the Work directory data field.
6. Click the OK button to confirm the changes in the analysis properties.

Running the analysis and loading the results

To submit the analysis, which will solve the prepared finite element model and load the results into PrePoMax, use the following steps (Figure 45):

1. Select Beam-Static in the Analyses collection of the FE Model feature tree by right-clicking it to open the context menu.
2. Select Run from the context menu to submit the analysis to the solver. This will open the Monitor window, where the solver progress in shown in the Output tab (if the analysis is running for a long time and needs to be stopped, use the Kill button).
3. Once the analysis is finished, the status will change to Finished (a Job finished prompt and the elapsed time will also be shown in the Output tab).
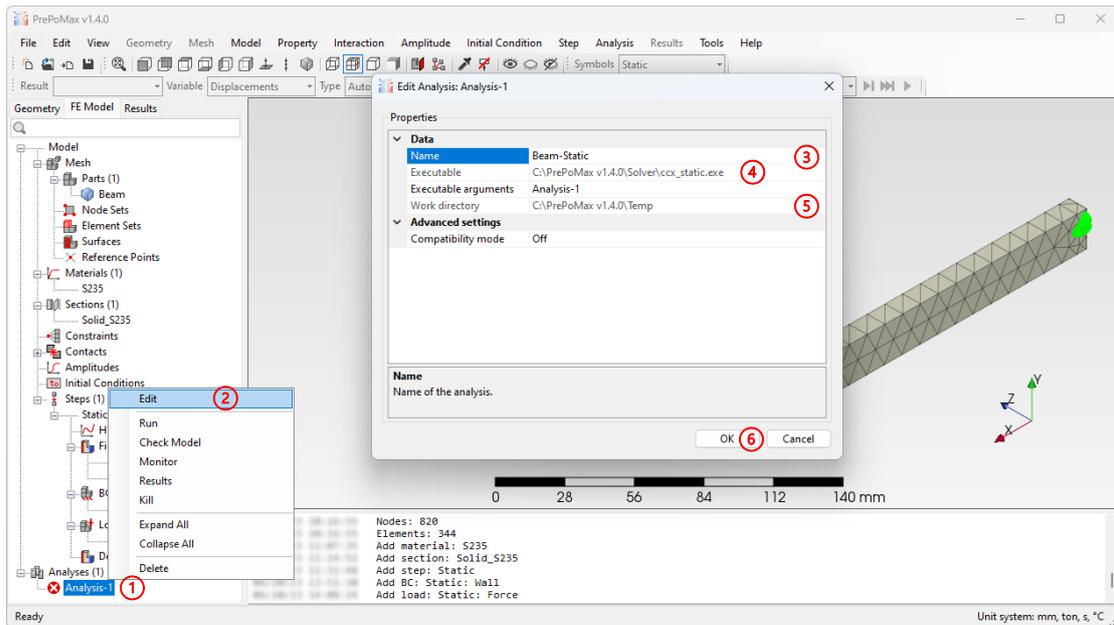4. Click the Results button to load the results into PrePoMax.

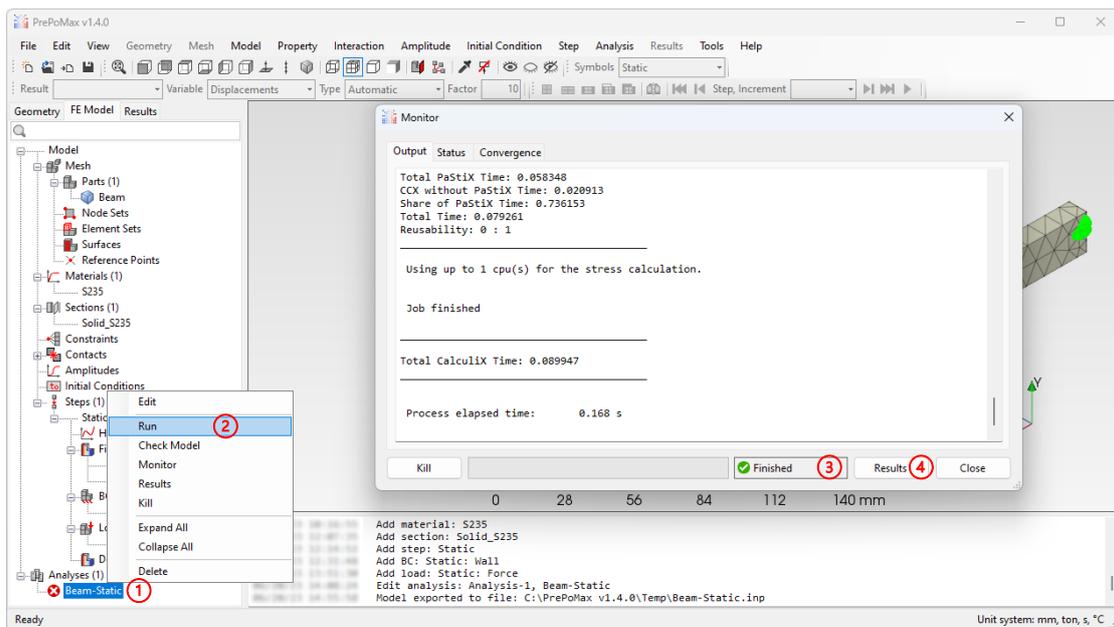**Figure 44: Editing the properties of the default analysis**

Figure 45: Running the analysis and loading the results

Analysing the results

To visualise the results (Figure 46), the active module is switched automatically to the Results Module in which the results can be analysed. The results of the computed scalar fields in all nodes and elements can be visualised on the model's mesh using colours. The following information about the results is displayed by default:

1. The path and name of the results frd file.
2. The Field Outputs collection shows the available results stored in the frd file, and enables the selection of the visualised field output. The DISP field represents displacements, the STRESS field represents stresses, the TOSTRAIN field represents total strains, the FORC field represents nodal forces, and the ERROR field represents an approximate accuracy of the results.
3. The Colour legend shows which output field is currently visualised and it's unit, and gives interval values for each colour.
4. The Status block shows the name of the results file and the date and time when the analysis was completed. It shows the numbers of the selected result step, increment, and displays the increment time. Lastly, it displays the name of the deformation variable and the deformation scale factor. The deformation scale factor defines the factor used to scale the deformations of the model, to adjust them for better visibility to the user. If the deformations of the results are relatively small compared to the model size, a scale factor larger than 1 is computed automatically to improve the quality of the visualised results.
5. The maximum value box shows the location, value and name of the node in which the maximum value of the selected output field is determined.

The displacement results of the beam in tension analysis are shown in Figure 46. The results show the maximum displacement of 0.0475 mm on the loaded side surface of the beam, while the displacement at the support is equal to 0 mm. The displacements increase linearly from the support towards the loaded surface. The comparison of the results with the analytical result from chapter 6.2.1, which equals 0.0476 mm, shows a relative difference of about 0.2 %, which can be regarded as negligible. In this case, the results of both approaches gave the same result. Based on this fact, a conclusion can be made that, for computing displacements, the selected mesh is fine enough.

Selecting DISP→U2 (1 in Figure 47) shows the displacements only in the y-axis direction. It can be seen from the results (DISP→U2 and DISP→U3) that, while the beam is extended in the longitudinal direction, its cross-section decreases in the transverse direction. This results from the applied linear elastic material model with a positive Poisson's ratio (0.3 for the steel S235). The cross-section decreases equally almost at the whole beam length except at the support. There, the prescribed boundary condition with displacements of 0 mm prevents the deformation of the beam in the transversal direction.
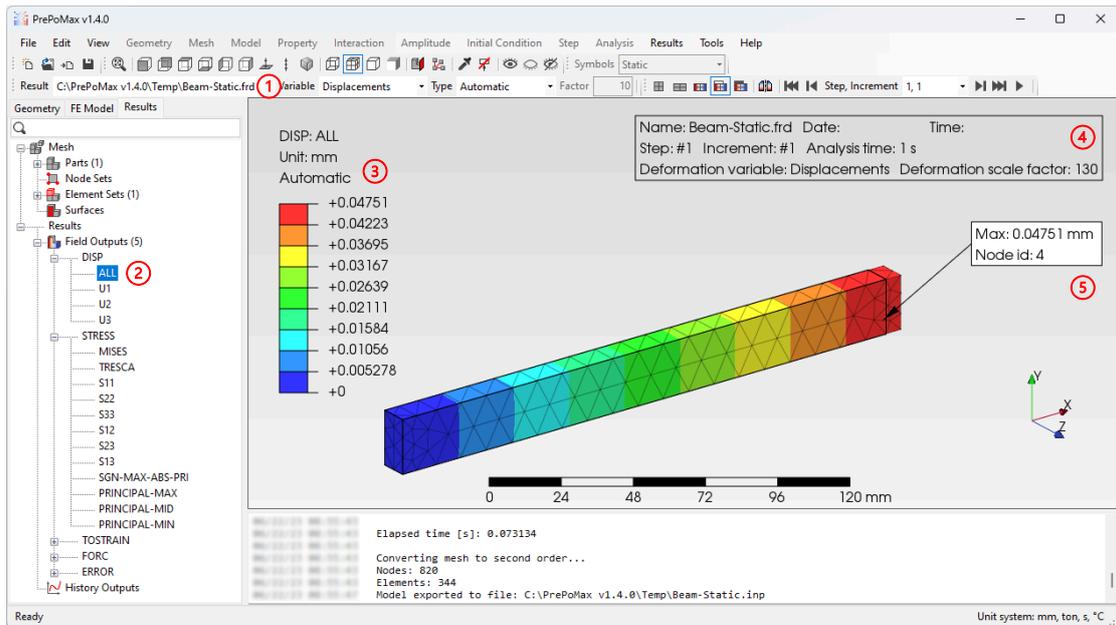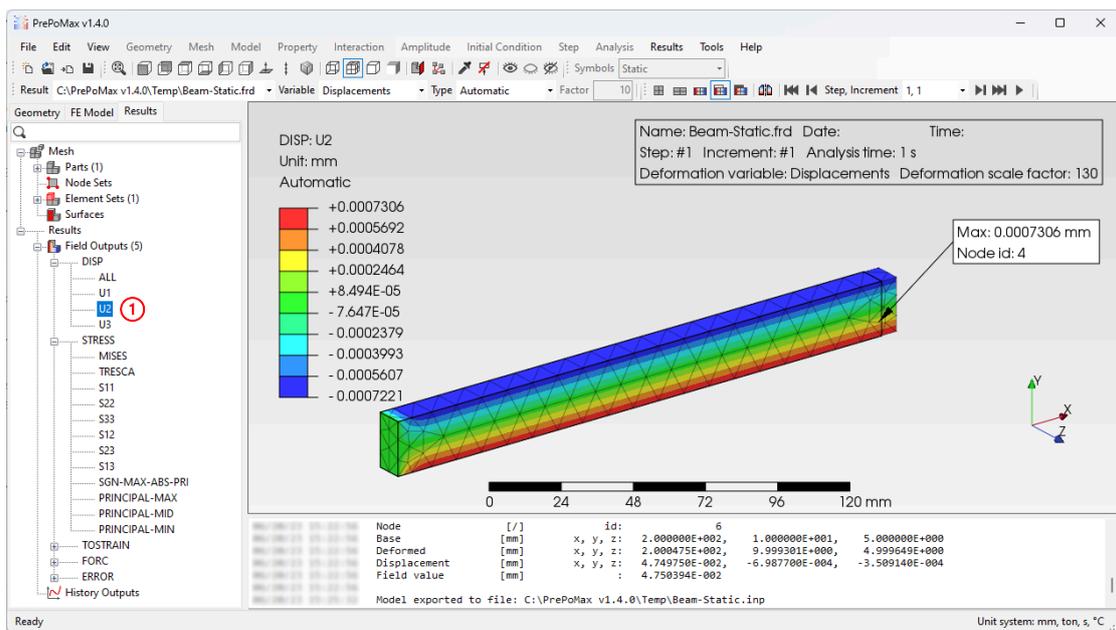
Figure 46: Visualising the results

Figure 47: Resulting displacements in the y-axis direction

Selecting STRESS→MISES (1 in Figure 48) shows the von Mises equivalent stress, a good scalar stress measure for steel, since it is a ductile isotropic material with the same behaviour in the tensile and compressive directions. Other stress measures must be considered for other material types (brittle material).

The results show a constant von Mises stress through the beam length, with a discrepancy in the stress field at the support, where the values (colours) of the stress field change rapidly irregularly. This indicates that, locally, the mesh in this region is comprised of finite

elements that are too large, and to improve the accuracy of the result, a finer mesh is needed. A mesh convergence study based on the value of stress can be used to determine the correct mesh size in this region.

To measure the field values at any mesh node a query tool can be used (2 in Figure 48). With the query tool set to Vertex/Node (3 in Figure 47) and moving the mouse over the model, the node id coordinates, and field values are displayed in a moving query box (4 in Figure 48). Using the query box, a constant stress of 50 MPa can be measured through the beam length away from the support.

The stress field changes at the support, and reaches the minimum and maximum values of 26 MPa and 55.35 MPa (< 235 MPa), respectively. Comparing this result with the analytical result from Chapter 6.2.1, which equals 50 MPa, shows a relative difference of about 7 % for the maximum stress value, and about 50 % for the minimum stress value, which cannot be regarded as negligible.

The reason for this relative difference in stress values is the support preventing the beam's lateral deformation. The analytical model is a 1D model, where the support only acts in the axial beam direction, while, in the transversal directions, there are no supports that would prevent the lateral deformation of the beam. At the same time, the analytical solution neglects the effect of the transversal deformation altogether, assuming a Poisson's ratio of 0.
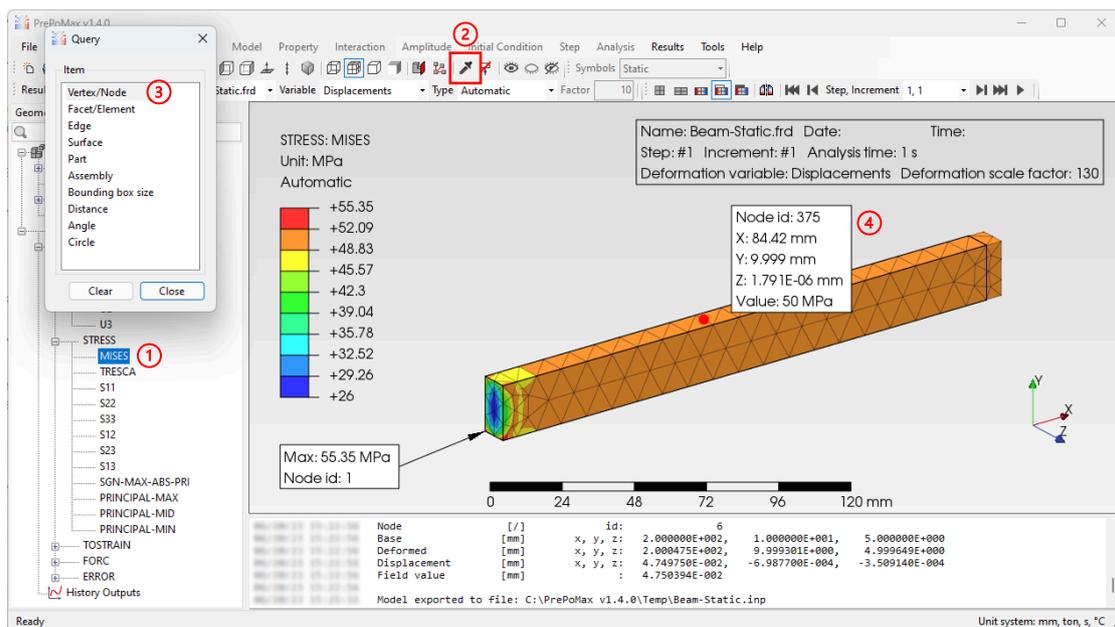

Figure 48: Resulting von Mises stress

A mesh convergence study should usually be carried out to determine the appropriate mesh size. From the analysis of the displacements in this case, it was determined that the global mesh size was good enough to determine the displacements' value accurately. However, the analysis of the stress results showed that stress at the support was not computed accurately. This suggests that a finer mesh should only be used around the support. It is known from the theory (Chapter 5.3) that a stress singularity will appear at the fixed support when using 3D finite elements. In this case, the mesh convergence will result in higher stress as the size of the finite elements decreases.

A mesh convergence study will be carried out to demonstrate this behaviour using the stress value at node 1 (maximum value box in Figure 48) as the convergence measure.

Defining the local mesh size

The mesh size can be defined globally using the Meshing Parameters feature, or locally using the Mesh Refinements feature in the Geometry module (change the module if necessary by clicking on the tab named Geometry above the feature tree). For this example, the mesh size will be adjusted locally on the beam surface connected to the wall, where the size of the elements will be set to 4 mm. Use the following steps to define a local mesh size based on geometry (Figure 49):

1. Select Mesh Refinements in the Geometry feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Mesh Refinement window.
3. Rename the mesh refinement in the Name data field to Wall.
4. Set the Element size data field to 4 mm.
5. In the 3D view, select the side surface of the beam connected to the wall.
6. Click the OK button to create the mesh refinement.

Updating the mesh, running the analysis and loading new results
Once a global mesh size changes or a new local mesh size is defined, the mesh must be generated again. Follow the instructions from the chapter "Creating the finite element mesh" to create an updated mesh of the model. Then, run the analysis and load its results following the instructions in the chapter "Running the analysis and loading the results" again.
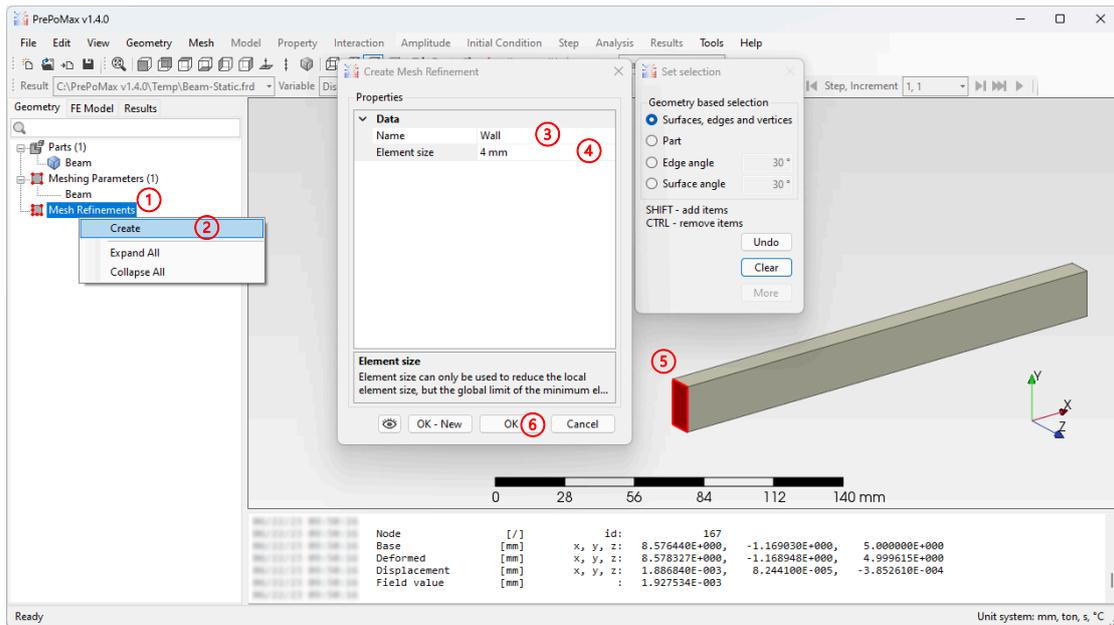
Figure 49: Defining the local mesh size

Results with improved mesh on the wall

Figure 50 shows the resulting von Mises stress field on the side surface of the beam connected to the wall. The value of the stress at node 1 can be determined using the Query tool (2 in Figure 48), and to improve the visibility of colours, turn off the element edges by setting the visualisation representation to Show Model Edges (3 in Figure 29). As predicted in the previous chapter, the stress value in node 1 increased and is equal to 62.68 MPa. The mesh convergence study will be repeated using the finite element size of 3 mm.
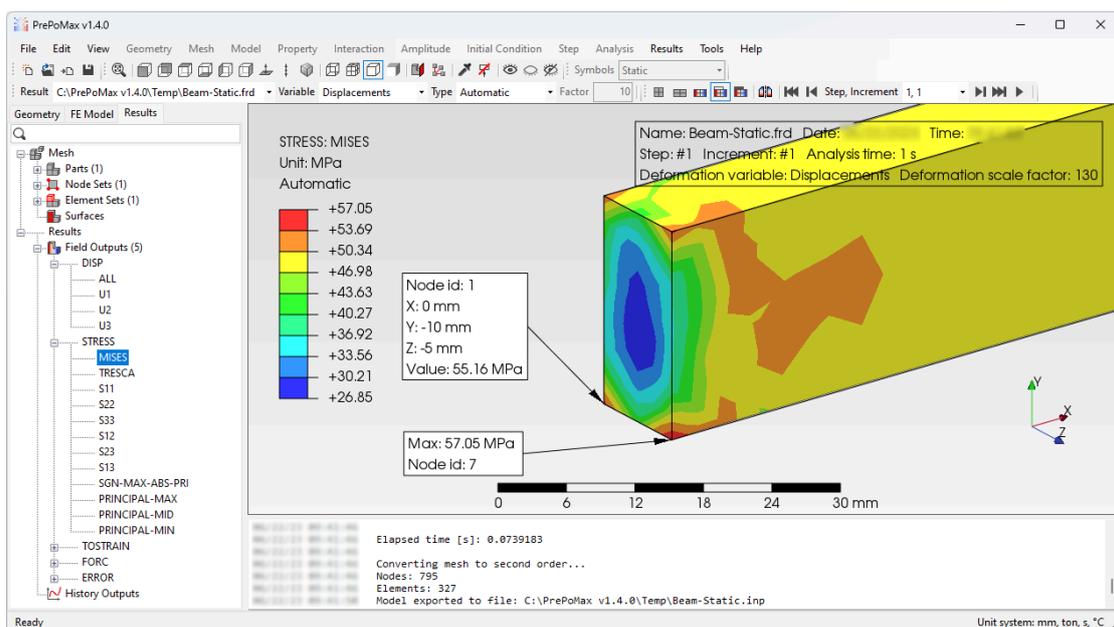

Figure 50: Resulting von Mises stress

Editing the mesh refinement

Once a mesh refinement is created, its properties can be changed by editing in the Geometry module (change the module if necessary by clicking on the tab named Geometry above the feature tree). Use the following steps to change the local mesh size of the mesh refinement named Wall from 4 mm to 3 mm (Figure 51):

1. Select the Wall mesh refinement in the Mesh Refinements collection in the Geometry feature tree by right-clicking it to open the context menu.
2. Select Edit from the context menu to open the Edit Mesh Refinement window.
3. Set the Element size data field to 3 mm.
4. Click the OK button to confirm the changes in the mesh refinement.
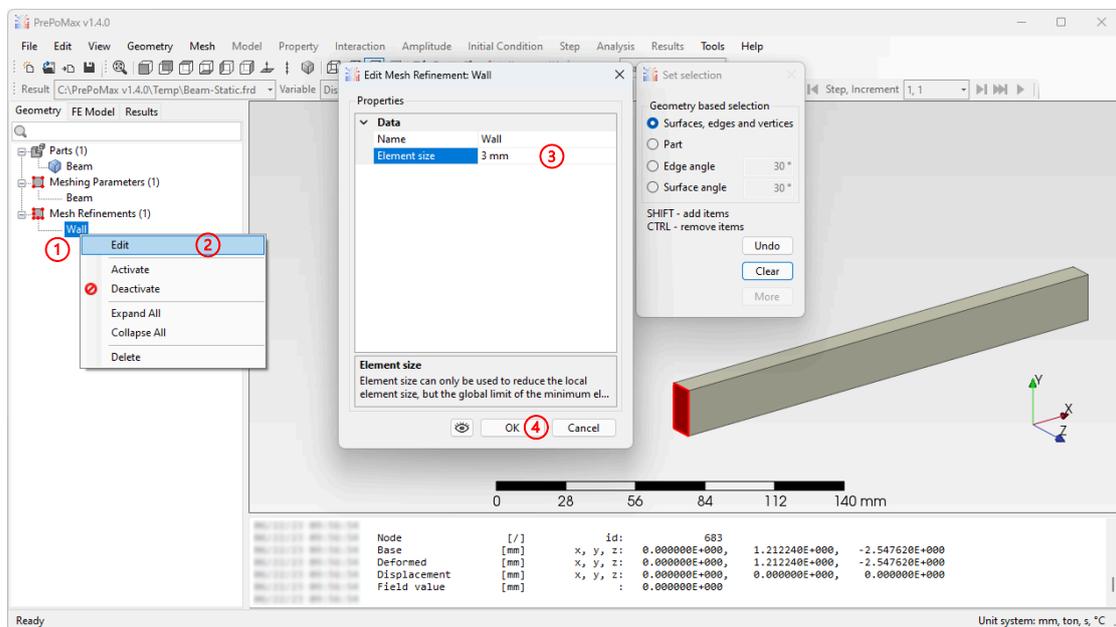

Figure 51: Editing the local mesh size

Updating the mesh, running the analysis and loading new results

Once a global mesh size changes or a new local mesh size is defined, the mesh must be generated again. Follow the instructions from the chapter "Creating the finite element mesh" to create an updated mesh of the model. Then, run the analysis and load its results following the instructions in the chapter "Running the analysis and loading the results" again.

Continuing the mesh convergence study

Looking at the results of the von Mises stress, the value of the stress in node 1 increased again, and is now equal to 67.52 MPa. The process of the mesh convergence study will be continued using meshes with the following sizes of the finite elements at the wall: 2 mm, 1 mm and 0.5 mm. This can be achieved by repeating the chapters "Editing the mesh refinement" and "Updating the mesh, running the analysis and loading new results".

The results of the complete mesh convergence study are shown in Figure 52. The Figure shows the change of von Mises stress in node 1 concerning the the finite element size at the wall. The results show that stress increases exponentially as the size of the finite element decreases. This confirms the existence of the stress singularity at the support. Despite the existence of the stress singularity at the support, the stresses away from the wall are computed accurately, and represent the correct solution to the problem.
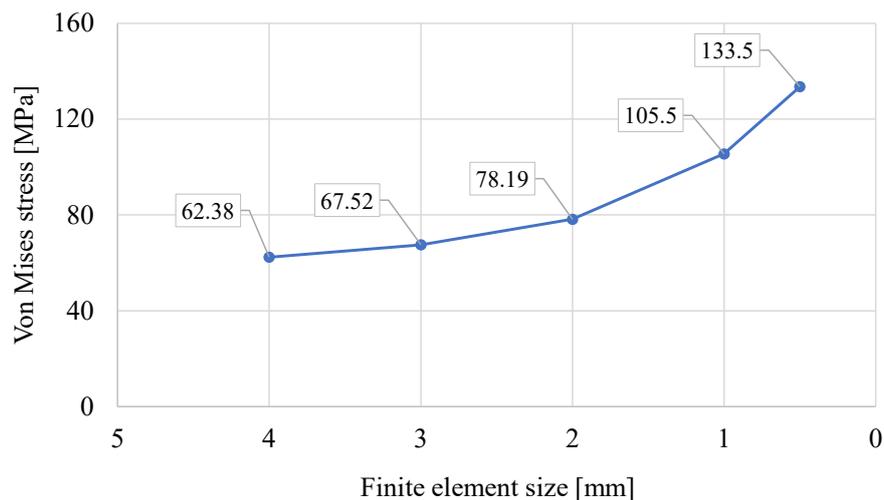

Figure 52: Mesh convergence results

Improving the modelling of the support

The stress singularity at the fixed support results from model simplification when preparing the boundary condition. In reality, the beam is connected to a deformable wall, while the fixed boundary condition is rigid, thus making the wall nondeformable.

Improving the support modelling is the only way to get around the stress singularity. There are different approaches to improving the models, where moving the support away from the important parts of the assembly is the most common. In this example, we could move

the support away from the side surface of the beam if we included the wall in the model and applied the support to the wall.

Another working approach in this case is to use an elastic support on the wall instead of the fixed support. First, the fixed support will be disabled, and then a new elastic spring constraint will be added on the wall.

Deactivating mesh refinements

Some features in the feature tree can be deactivated or activated to study their effect on the results. The mesh refinements are one of these features, and thus can be deactivated in the Geometry feature tree (change the module if necessary by clicking on the tab named Geometry above the feature tree). To deactivate the mesh refinement named Wall, follow the steps (Figure 53):

1. Select the Wall mesh refinement in the Mesh Refinements collection in the Geometry feature tree by right-clicking it to open the context menu.
2. Select Deactivate from the context menu to deactivate the feature.

To deactivate/activate other features, search for them by name in the appropriate feature tree and then repeat steps 1 and 2.
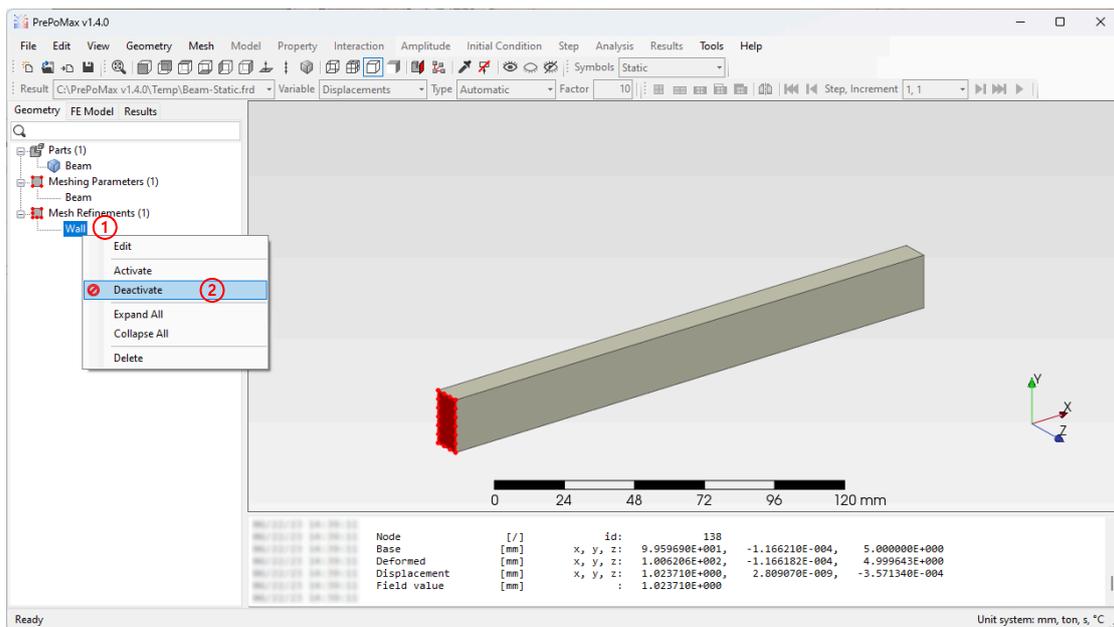

Figure 53: Deactivating mesh refinements

Updating the mesh

Follow the instructions from the chapter "Creating the finite element mesh" to create an updated mesh of the model.

Deactivating the support

Change the module to the FE model (by clicking on the tab named FE model above the feature tree) and search for the support named Wall in the BSc collection. To deactivate the support, follow the steps from the chapter "Deactivating mesh refinements".

Creating a surface spring constraint

Constraints of different kinds can be added to the finite element models. For this example, a surface spring constraint with linear elastic properties will be added to the model.

In general, the stiffness of spring constraints must be selected carefully, to reflect the actual behaviour of the structure. A surface spring constraint with a stiffness of 10,000 N/mm in the axial and 100 N/mm in the transversal direction was chosen for this example. The axial stiffness was selected so that the constraint springs would elongate by exactly 1 mm when the load was applied to the model. In contrast, a much smaller transversal stiffness was selected, only to prevent the rigid body motion of the beam. These stiffness values were selected to get the analysis stress results close to the analytical stress results.

A new constraint can be added to the Constraints collection in the FE model module (change the module if necessary by clicking on the tab named FE model above the feature tree). Use the following steps to create a new surface spring constraint on the side surface of the beam connected to the wall (Figure 54):

1. Select Constraints in the FE model feature tree by right-clicking it to open the context menu.
2. Select Create from the context menu to open the Create Constraint window.
3. Select the Surface Spring as the constraint type.
4. Rename the constraint in the Name data field to: Spring_Wall.
5. Set the K1 data field (K – stiffness constant, 1 – the first axis of the coordinate system) to 10,000 N/mm.
6. Set the K2 data field to 100 N/mm.
7. Set the K3 data field to 100 N/mm.

8. In the 3D view, select the side surface of the beam connected to the wall.
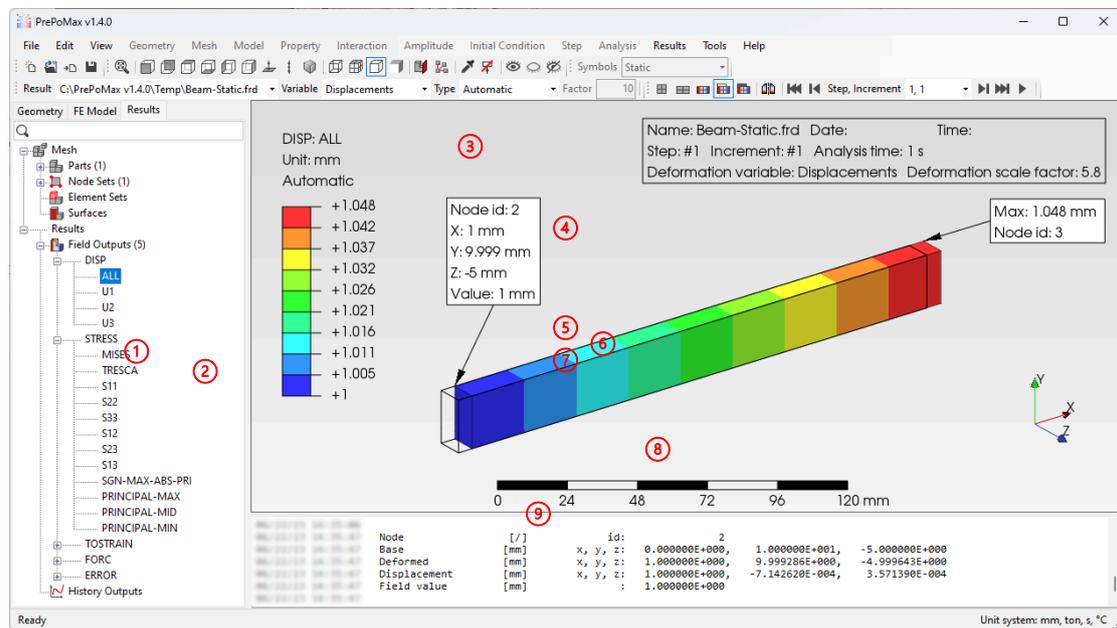9. Click the OK button to create the surface spring constraint.

Figure 54: Creating a surface spring constraint

Running the analysis and loading new results for the spring constraint

Changing the model requires the analysis to be submitted again. Follow the steps from the chapter "Running the analysis and loading the results" to load new results from the model with the surface spring.

Results with the surface spring

The results of beam displacements (Figure 54) show an increase in the displacement values. The deformed results show that the beam moved exactly 1 mm at the surface supported by the springs. The rest of the beam deformed similarly as previously, but to get the actual beam deformations, a length of 1 mm must be subtracted from the displacement value.

The results of the von Mises stress are shown in Figure 56. The colours at the beam surface attached to the surface spring constraint show that the stress field in this area was not constant. On the other hand, checking the values in the Colour legend indicates that the difference between the maximum and minimum value is less than the currently used number precision, so all colours represent a value of 50 MPa.
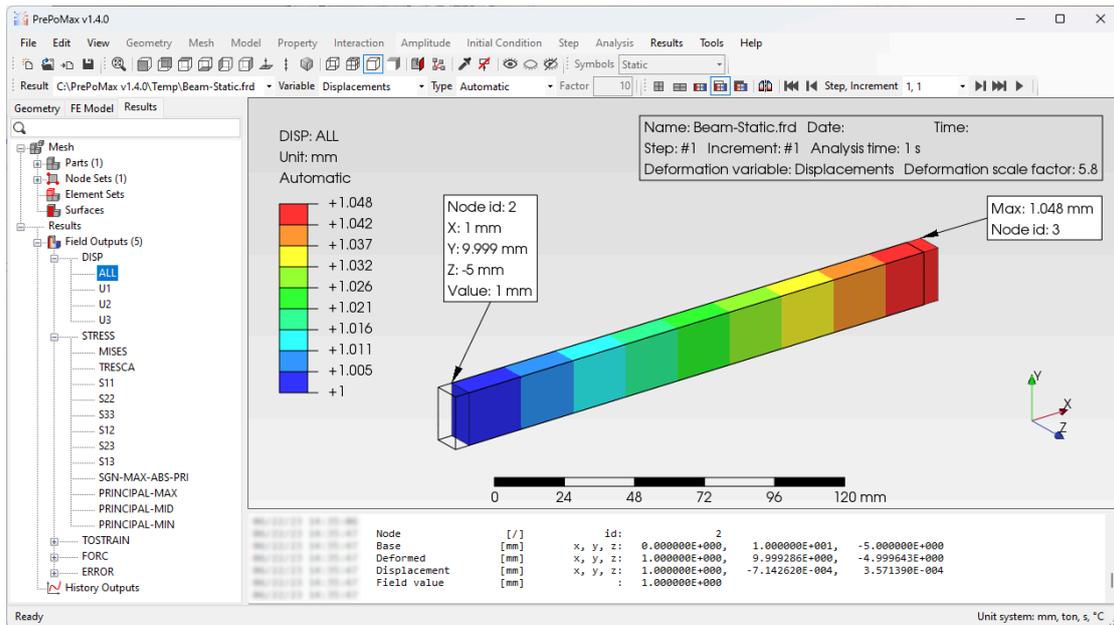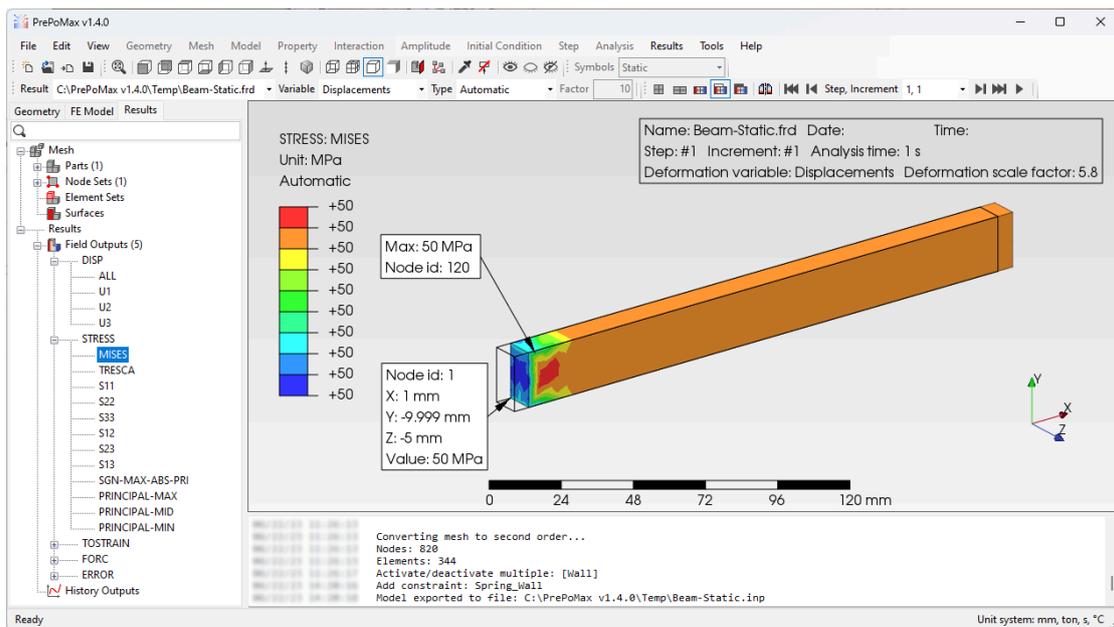
Figure 55: Resulting displacements

Figure 56: Resulting von Mises stress

Conclusions

The beam in tension example first shows how to use the PrePoMax user interface to prepare a basic linear static finite element model and then deal with stress singularities. In the example, the stress singularity was avoided from the results by using an elastic instead of a fixed support. The disadvantage of this approach is that the absolute displacements are always increased. Combining both approaches can be used to evaluate the displacements and stresses correctly.

6.3 Bending of an L bracket

The second practical example is the L-shaped bracket in Figure 57. The bracket is fixed on its top surface (green colour). An $F = 200$ N load acts on the frontal bracket surface (blue colour) in the tranversal direction. The bracket is made of S235 steel, with the following material properties $E = 210$ GPa in $\nu = 0.3$ and $\sigma_y = 235$ MPa. Determine the maximum stresses and displacements in the bracket.
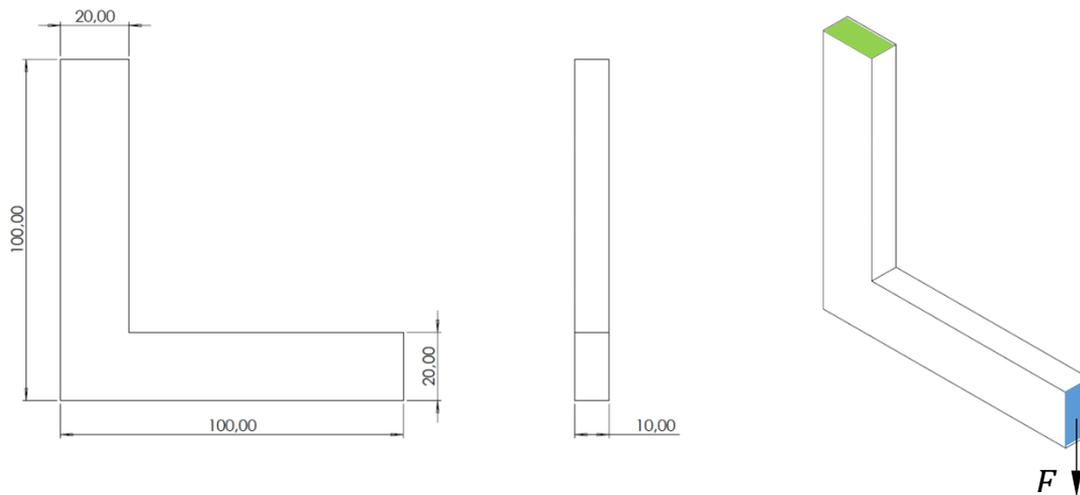

Figure 57: The L bracket example

6.3.1 Analytical solution

Solving the L bracket example analytically, the 3D beam model is simplified into a 2D beam model (a in Figure 58), and then divided into two 1D beam models (b and c in Figure 58).

The analytical solution first starts by solving model b using Equation (10). Model b is loaded by a bending stress (neglecting the transversal shear stress), where the largest bending stress appears at point B, and equals $\sigma_b = 27$ MPa. In the next step the reaction forces from point B are transferred to model c into point C, where they are applied as loads (their orientation is reversed). Model c is loaded with a constant bending stress and a constant axial stress that give a combined maximum stress of $\sigma_c = 28$ MPa.

$$F_T = F = 200 \text{ N}$$
$$M_z = F \cdot L = 200 \text{ N} \cdot 90 \text{ mm} = 18000 \text{ Nmm}$$
$$\sigma_b = \frac{M_b}{W} = \frac{M_z}{W} = \frac{18000 \text{ Nmm}}{666{,}6 \text{ mm}^3} = 27 \text{ MPa} \qquad (10)$$
$$W = \frac{b \cdot h^2}{6} = \frac{10 \text{ mm} \cdot 20^2 \text{mm}^2}{6} = 666{,}6 \text{ mm}^3$$

$$\sigma_t = \frac{F_T}{A} = \frac{200 \text{ } N}{200 \text{ mm}^2} = 1 \text{ MPa}$$
$$A = a \cdot b = 10 \text{ mm} \cdot 20 \text{ mm} = 200 \text{ mm}^2 \qquad (11)$$
$$\sigma_b = \frac{M_b}{W} = \frac{M_z}{W} = \frac{18000 \text{ Nmm}}{666{,}6 \text{ mm}^3} = 27 \text{ MPa}$$
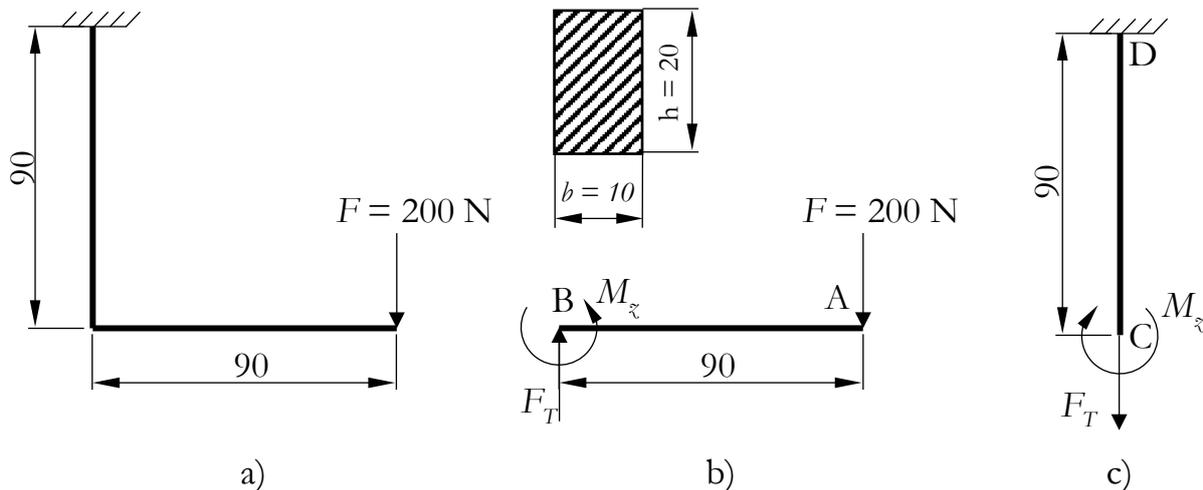$$\sigma_c = \sigma_b + \sigma_t = 27 \text{ MPa} + 1 \text{ MPa} = 28 \text{ MPa}$$

Figure 58: Dimensions of the L bracket

6.3.2 Finite element solution

The workflow of preparing the finite element model of the L bracket follows the same steps as the workflow for preparing the beam in the tension model:

1. Create a new model inside PrePoMax by using the 3D model space and the unit system type of mm, ton, s, °C (Preparing a new model).
2. Import the L bracket geometry form the file "L Bracket.STEP" contained in the Models subfolder of the PrePoMax folder (Importing geometry).
3. Change the name of the imported part from Solid_part-1 to L_Bracket (Changing the part properties).

4. Create new global meshing parameters for part L_Bracket. Rename the new meshing parameters to L_Bracket, with a maximum element size of 10 mm and a minimum element size of 0.1 mm (Definition of the meshing parameters). A coarse mesh will be used for the first analysis of the model. A minimum size of 0.1 mm is needed for a later mesh convergence study.

5. Create the finite element mesh (Creating the finite element mesh).

6. Add a linear elastic material model named S235 and the following elastic properties: Young's modulus of 210 GPa and Poisson's ratio of 0.3 (Adding a material model).

7. Create a section assignment called Solid_S235 from material S235 connected to the geometry of the L_Bracket part (Creating a section assignment).

8. Add a static analysis step called Step-1 with the default parameters (Adding a simulation step).

In the next step, a fixed boundary condition must be applied to the bracket's surface, which is green in Figure 59. This could be done the same way as for the beam in the tension model, but here a simpler approach will be used using a boundary condition called Fixed.

Creating a fixed boundary condition

A fixed boundary condition applies a value of 0 automatically to all DOFs, fixing them in place, and can be added by the following steps (Figure 59):

1. Select BCs in the FE Model feature tree by right-clicking it to open the context menu.

2. Select Create from the context menu to open the Create Boundary Condition window.

3. Select Fixed as the boundary condition type.

4. In the Name data field enter the boundary condition name: Wall.

5. In the 3D view select the smaller bracket surface oriented towards the positive y-axis direction.

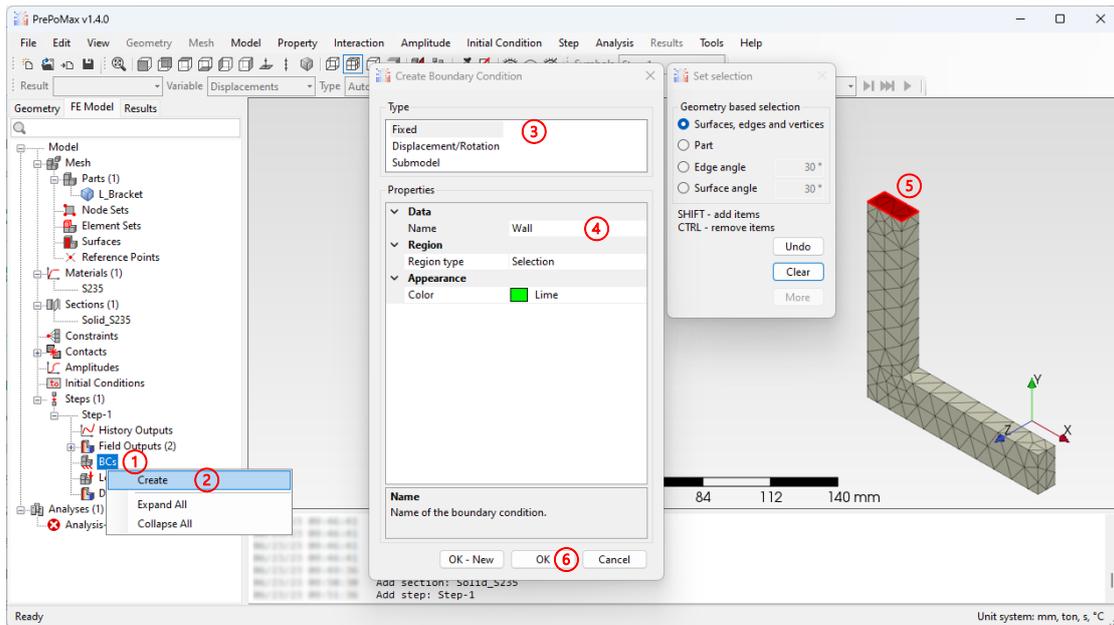6. Click the OK button to confirm the boundary condition creation.

Figure 59: Creating a fixed boundary condition

Continuation of the L bracket model preparation

The preparation of the L bracket model continues with the following steps:

7. Add a surface traction load to the smaller bracket surface oriented towards the positive x-axis direction (the blue surface in Figure 58) called Force, with the force component F2 equal to -200 N (Defining a surface traction load).
8. Rename the prepared Analysis-1 to Static_L_Bracket (Analysis setup). The finished L bracket model is shown in Figure 60.
9. Run the analysis and load the results (Running the analysis and loading the results).
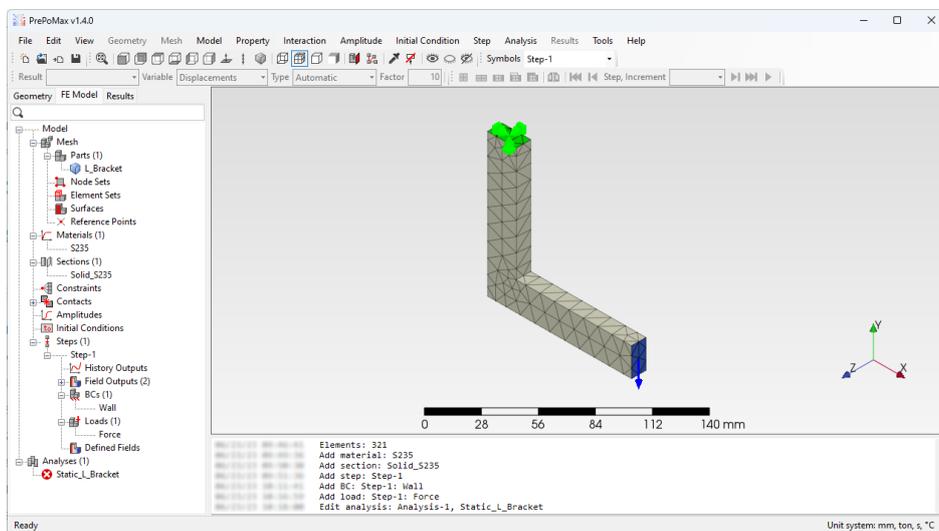

Figure 60: Finished L bracket finite element model

Analysing the results

The displacement results of the L bracket analysis are shown in Figure 61. The results show the maximum displacement of 0.1488 mm on the loaded side surface of the bracket, while the displacements at the support are equal to 0 mm. The black wire frame lines represent the bracket's undeformed shape, and the bracket deformation is increased by a factor of 55 (deformation scale factor). The undeformed shape of the bracket can be hidden by using a result visualisation called Deformed with colour contours (Results` representation).
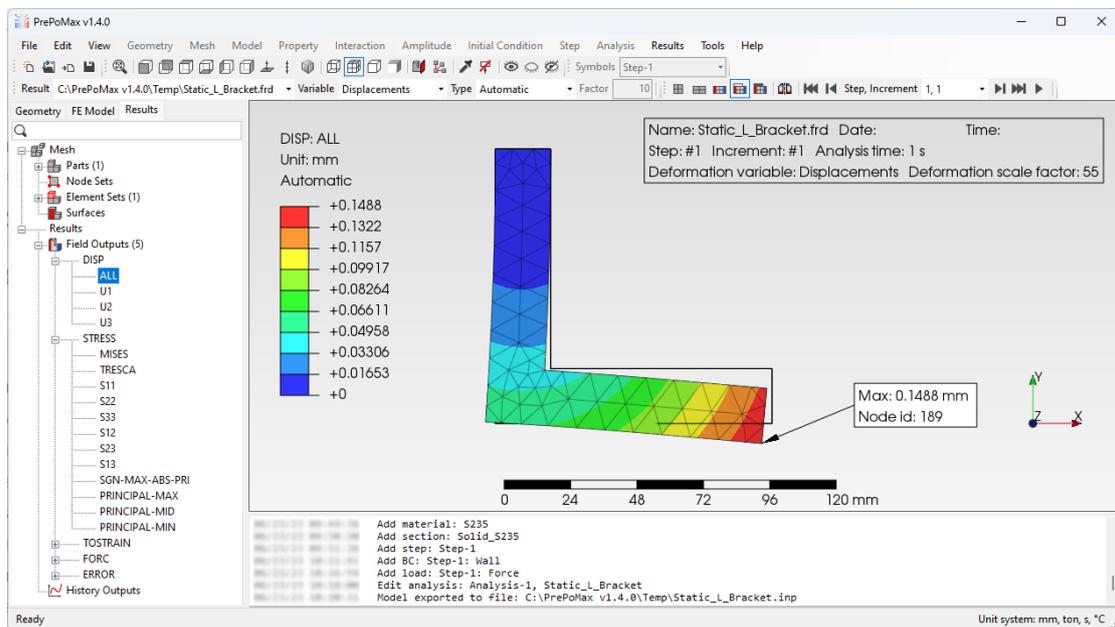

Figure 61: Resulting displacements for the L bracket

Figure 62 shows the von Mises stress distribution in the L bracket. The highest von Mises stress appears in the corner of the bracket on its internal side and equals 34.82 MPa. In this region it is hard to compare the numerical result with the analytical result, since the shape of the bracket is changing from a horizontal to a vertical direction. That is why an additional measure point was selected on the vertical part of the bracket (model c in Figure 58), where the stress was determined analytically as constant through the vertical part of the bracket, with the combined stress of σ_c = 28 MPa. At the same time, von Mises stress of 28 MPa was measured at a point in the middle of the vertical part of the bracket using the query tool (Node 281 in Figure 62) in the numerical model. Since analytically and numerically determined stress values are the same, the numerical model was prepared correctly, and the mesh density in the measured point is adequate.

The stress field around the position of the maximum von Mises stress is not smooth, suggesting that the result can be improved using a finer mesh. On the other hand, it is well known from the theory (Paragraph 5.3) that stress singularities may occur in sharp corners. To confirm this fact, a mesh convergence was carried out using the stress value at node 8 (Figure 62) as the convergence measure.
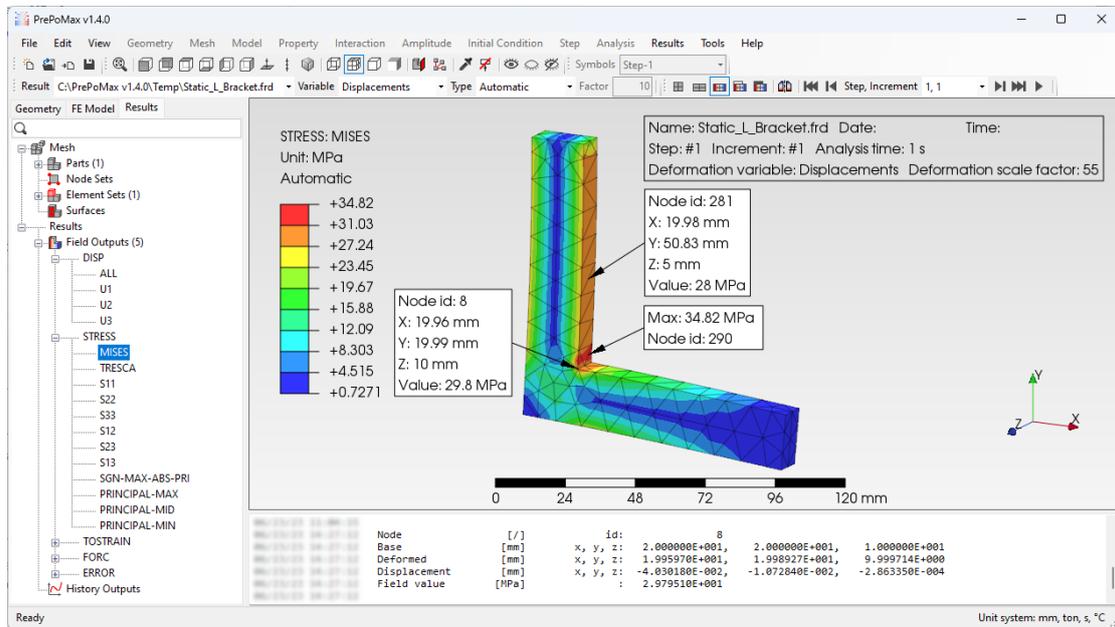

Figure 62: Resulting von Mises stress for the L bracket

Mesh convergence study of the L bracket

For the mesh convergence study, different local mesh sizes were used on the internal edge of the L bracket part (Figure 63). The local mesh sizes were 3 mm, 2 mm, 1 mm, 0.5 mm and 0.3 mm. The user is encouraged to do the mesh convergence study by repeating the following steps for each local mesh size used:

1. Create a mesh refinement named Edge on the edge of the L bracket corner with the local mesh size of 3 mm (Defining the local mesh size). The next time this process is repeated, edit the existing mesh refinement and change the finite element size (Editing the mesh refinement).
2. Update the finite element mesh (Creating the finite element mesh).
3. Run the analysis and load the results (Running the analysis and loading the results).
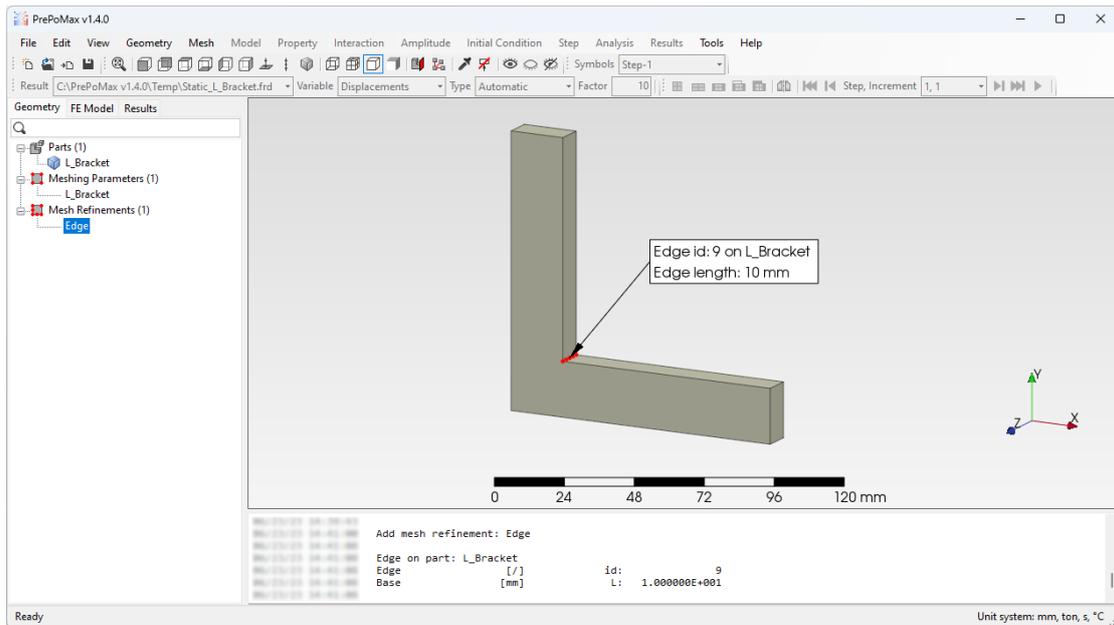
Figure 63: Mesh refinement on the edge of the L bracket corner

The results of the mesh convergence study for the L bracket model are shown in Figure 64. The Figure shows the von Mises stress in node 8 in dependence on the local finite element size at the edge. The result shows that stress increases exponentially as the size of the finite element decreases. This confirms the existence of the stress singularity at the edge of the corner. Despite the existence of the stress singularity at the support, the stresses away from the sharp corner are computed accurately and represent the correct solution to the problem.
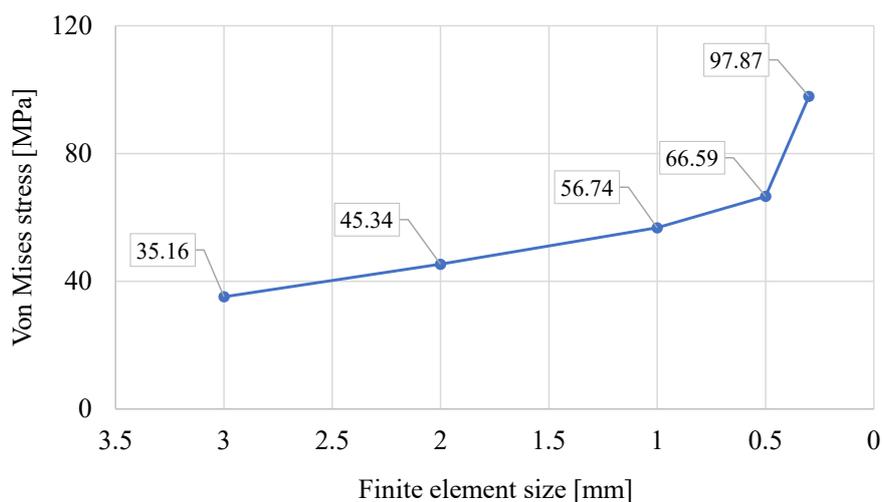

Figure 64: Mesh convergence results

Improving the modelling of the sharp corner

Stress concentration at the sharp corner is the consequence of simplification in the L bracket geometrical model. Namely, the L bracket part cannot be produced with a corner of 90°, since no matter which manufacturing process is used, a fillet will be in the corner. Additionally, a good engineer should predict a stress concentration in this region and design the bracket with a fillet in the first place.

An analysis of the updated geometry is carried out to show the stress distribution in the L bracket with a fillet. Generally, a completely new model would have to be prepared for that, but PrePoMax records all user steps while preparing the finite element model. Additionally, this recording can be repeated, and other geometry files can be used while importing the geometry.

Regenerating the model

The L bracket geometry containing a fillet in the sharp corner is stored in the file "L Bracket R.STEP" in the Models subfolder of the PrePoMax folder. To recreate the L bracket model using the modified geometry, follow the steps (Figure 65):

1. Select Edit in the Main menu to show the menu items.
2. Select the menu item Regenerate Using Other Files to open the File Open dialog window.
3. Browse to the location of the Models subfolder ../PrePoMax v1.4.0/Models.
4. Select the file "L Bracket R.STEP".
5. Click the OK button to confirm the model regeneration.
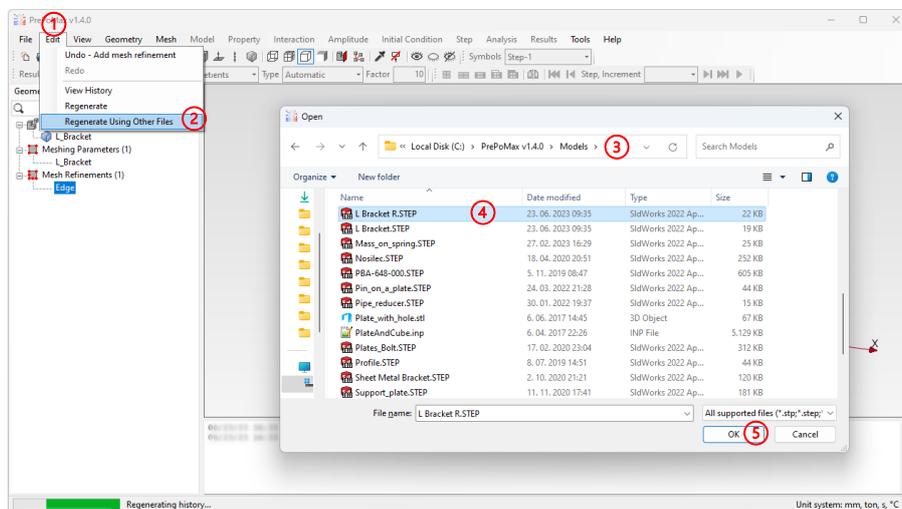

Figure 65: Regenerating the model

After the file is selected, the regeneration of the model will begin. Rebuilding the model might take some time, depending on the workflow of the model preparation, since all steps of meshing the model will be repeated.

Fixing the mesh refinement

While preparing the model with the sharp corner, a mesh refinement was created on the edge inside the sharp corner. The new bracket geometry misses this edge, so that the nearest geometry item will be selected automatically. For the new geometry, the mesh refinement should be created on the surface of the fillet, but the chosen automatically generated geometry item could not be the right one. To fix the automatic selection of the mesh refinement, edit the mesh refinement called Edge, change the element size to 0.5 mm, and select the fillet surface as the selected region (Figure 66).
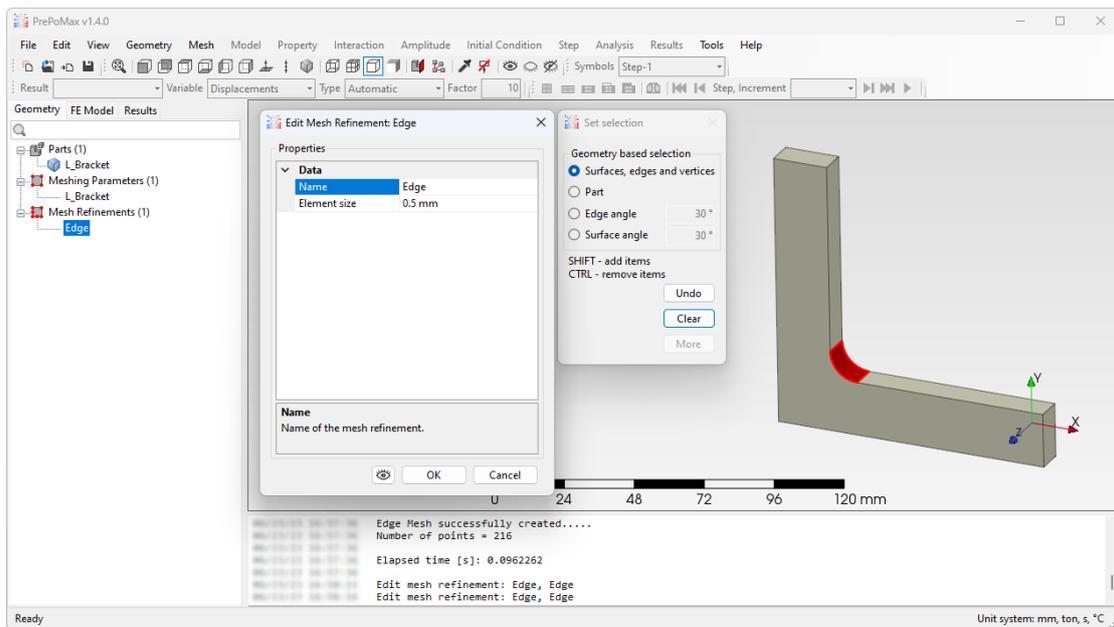

Figure 66: Fixing the mesh refinement

Updating the mesh, running the analysis and loading new results

Once the mesh refinement is fixed, the mesh needs to be generated again. Follow the instructions from the chapter "Creating the finite element mesh" to create an updated mesh of the model. Then, run the analysis and load its results (Running the analysis and loading the results).

Results with the fillet

The displacements of the L bracket with fillet are shown in Figure 67. The visualisation representation was changed to Show model edges (Visualisation representation) to hide the mesh. The maximum displacement of 0.1402 mm is shown on the loaded side surface of the bracket, which is less than the maximum displacement of the L bracket without a fillet (0.1488 mm). This is because, due to fillet creation, added material reinforces the model, making it stiffer and harder to deform.
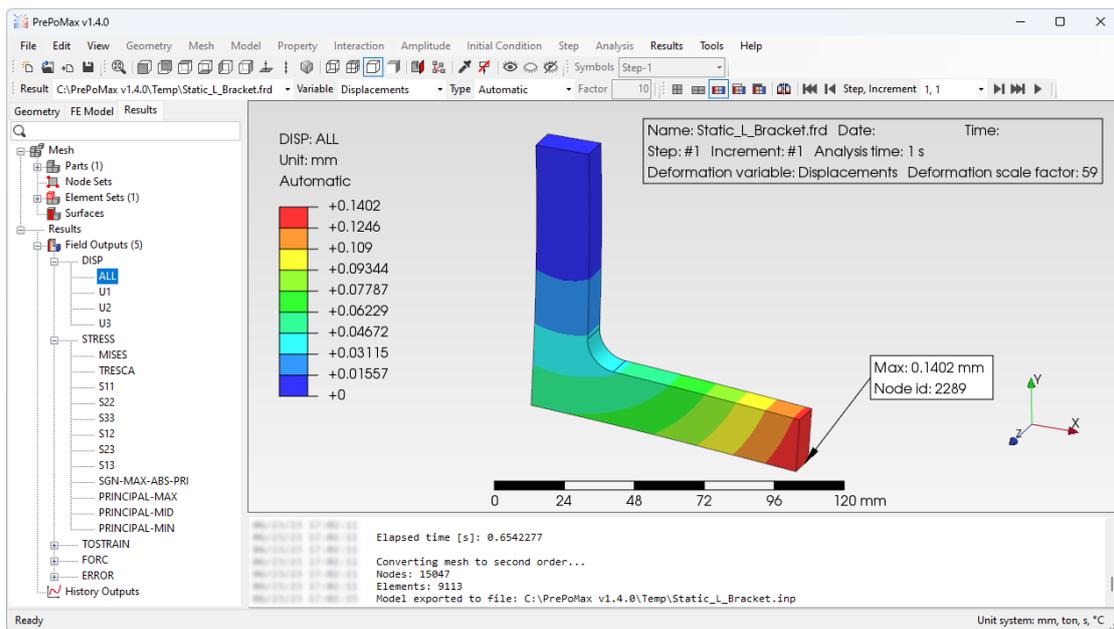

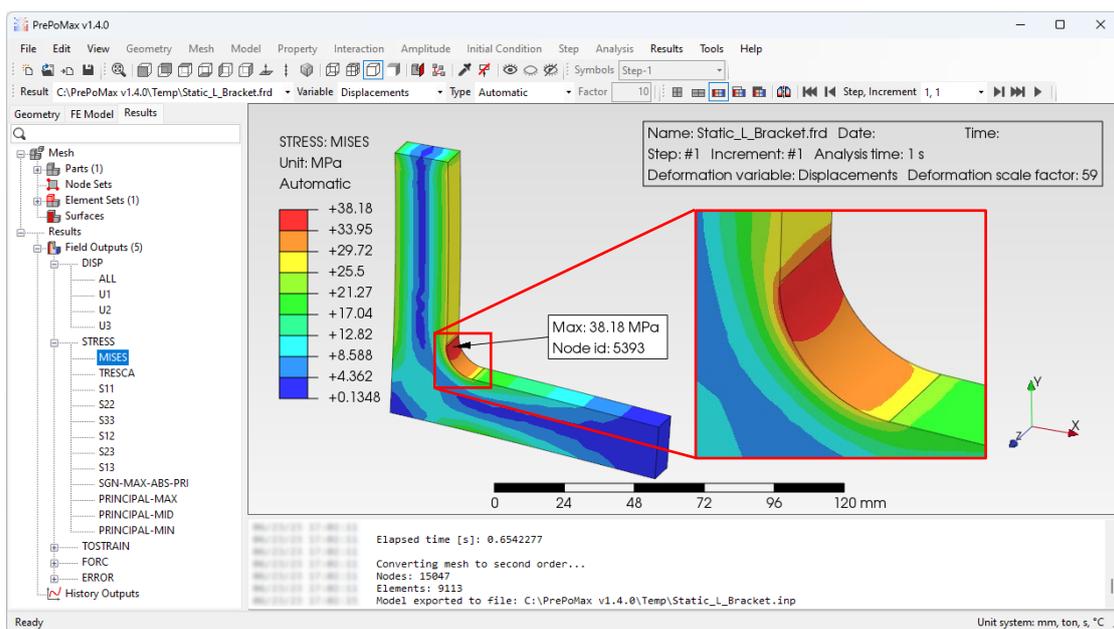**Figure 67: Resulting displacements for the L bracket with fillet**

Figure 68: Resulting von Mises stress for the L bracket with fillet

Figure 68 shows the von Mises stress of the updated L bracket geometry. The highest stress equals 38.18 MPa, and appears in the fillet of the bracket corner. The magnified view of the stress concentration shows a smooth stress field without any rapid changes of stress, and a stress concentration that stretches over multiple layers of finite elements (mesh must be turned on for this to be visible), which suggests that a convergent stress result was found. To prove this assumption, a mesh convergence study using the stress on the fillet surface would have to be carried out.

Conclusions

The L bracket example demonstrates the existence of stress singularity in sharp corners and how to deal with them. In the example, the stress singularity was removed from the model by adding a fillet to the critically loaded sharp corner. This approach's disadvantage is that fillets used on the production models are generally very small. To mesh a fillet accurately, at least a couple of elements must be used over their arc length, making the finite elements inside the fillets extremely small, increasing the complexity of the model and its computational time. Thus, the finite element models include only important fillets for predicting the stresses accurately.

Literature

[1] "The Finite Element Method (FEM)," *Accessed on 17.11.2023*. https://www.comsol.com/multiphysics/finite-element-method

[2] Z. Ren, M. Ulbin, and M. Vesenjak, *Inženirske računalniške simulacije v konstrukterstvu*. Maribor: Univerzitetna založba Univerze v Mariboru, 2018.

[3] "Nastran," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/Nastran

[4] "Ansys," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/Ansys

[5] "Finite element method," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/Finite_element_method

[6] G. Zienkiewicz and R. Taylor, *Finite element method. Volume 1. The basis*. Oxford: Butterworth-Heinemann, 2000.

[7] *Abaqus analysis user manual*. Dassault Systémes, 2007.

[8] J. Hallquist, *LS-DYNA theory manual*. Livermore, California: Livermore Software Technology Corporation, 2006. [Online]. Available: http://www.dynasupport.com/manuals/additional/ls-dyna-theory-manual-2005-beta/at_download/file

[9] "Young's modulus," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/Young%27s_modulus

[10] "Poisson's ratio," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/Poisson%27s_ratio

[11] "Hooke's law," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/Hooke%27s_law

[12] N. Novak, M. Vesenjak, and Z. Ren, "Auxetic cellular materials - a Review," *Strojniški Vestn. - J. Mech. Eng.*, vol. 62, no. 9, pp. 485–493, 2016, doi: 10.5545/sv-jme.2016.3656.

[13] "AUXETIC - THE SCIENCE BEHIND THE SENSATIONAL FEEL," *Accessed on 17.11.2023*. https://www.head.com/de_CH/rs/stories/auxetic-the-science-behind-the-sensational-feel

[14] X. Ren, R. Das, P. Tran, T. D. Ngo, and Y. M. Xie, "Auxetic metamaterials and structures: A review," *Smart Mater. Struct.*, 2018.

[15] D. François, A. Pineau, and A. Zaoui, *Mechanical Behaviour of Materials*, vol. 180. Dordrecht: Springer Netherlands, 2012. doi: 10.1007/978-94-007-2546-1.

[16] S. Marzi, O. Hesebeck, M. Brede, and F. Kleiner, "A Rate-Dependent , Elasto-Plastic Cohesive Zone Mixed-Mode Model for Crash Analysis of Adhesively Bonded Joints," 2009.

[17] T. T. SolidWorks Express, "How to use symmetry and anti-symmetry boundary conditions." https://www.clear.rice.edu/mech403/HelpFiles/CW_sym_anti-symmetry_BC.pdf

[18] "Linear Elasticity: Singular Solutions for the Half-Space," *Accessed on 17.11.2023*.https://www.brown.edu/Departments/Engineering/Courses/EN224/halfspace/halfspace.html

[19] A. E. H. Love, *A Treatise on the Mathematical Theory of Elasticity, Volume 1*. Cambridge University Press, 1892. [Online]. Available: https://books.google.com/books?id=JFTbrz0Fs5UC&pgis=1

[20] "ISO 10303," *Accessed on 17.11.2023*. https://en.wikipedia.org/wiki/ISO_10303

[21] S.-W. Cheng, K. T. Dey, J. Shewchuk, and Richard, *Delaunay Mesh Generation*. Boca Raton: CRC Press, 2013.

INTRODUCTION TO THE COMPUTER SIMULATIONS: SCRIPT

NEJC NOVAK, MATEJ BOROVINŠEK, MATEJ VESENJAK, ZORAN REN

University of Maribor, Faculty of Mechanical Engineering, Maribor, Slovenia
n.novak@um.si, matej.borovinsek@um.si, matej.vesenjak@um.si, zoran.ren@um.si

The script entitled „Introduction to the computational simulations" in the field of engineering computer simulations is intended as a study aid in the lectures of the courses Engineering Computer Simulations for foreign students at the University of Maribor and for students at Kumamoto University, Japan. It contains an explanation of the entire material that students must master in these subjects and is consistent with the curriculum of the aforementioned subject. The basics of computational simulations based on the finite element method are given from the theoretical basics to step-by-step preparation of simple computational models in PrePoMax software.