Department of Mechanical Engineering

MEC 5068Z

Topics in Computational and Applied Mechanics

Abdul Gafeeth Benjamin
BNJABD005

November 2024

Contents

Notations

The following is a list of some of the parameters and what they represent which are present in this report.

- ρ - density
- $\boldsymbol{v}$ - velocity vector
- $\boldsymbol{\sigma}$ - Cauchy stress tensor
- $\mathbf{b}$ - body force vector
- $\boldsymbol{\varepsilon}$ - infinitesimal strain tensor
- $\mathbf{u}$ - displacement vector
- $\mathbf{s}$ - deviatoric stress tensor
- $\mathbf{I}$ - second-order identity tensor
- f - yield function
- σ_y - yield stress
- $d\varepsilon^p$ - plastic strain increment
- ξ - plastic multiplier
- H - isotropic hardening parameter
- γ - accumulated plastic strain
- σ_0 - original yield stress
- L_0 - original length
- E - elastic modulus
- E_{tan} - tangent modulus
- $\bar{u}$ - applied displacement
- V - volume
- V_0 - original volume
- ν - Poisson's ratio
- V_e - volume during elastic loading

- V_p - volume during plastic loading
- u_x - displacement in the x-direction
- u_y - displacement in the y-direction
- u_z - displacement in the z-direction
- λ - Lamé's first parameter
- μ - Lamé's second parameter
- σ_1 - first principal stress
- σ_2 - second principal stress
- σ_3 - third principal stress
- σ_f - final stress
- t_x - traction component in the x-direction
- t_y - traction component in the y-direction
- t_z - traction component in the z-direction
- R_o - hollow-cylinder outer radius
- R_i - hollow-cylinder inner radius
- $\boldsymbol{\beta}$ - back-stress tensor
- $\bar{\beta}$ - kinematic hardening tensor
- $\boldsymbol{\eta}$ - relative stress tensor
- τ - shear stress

Abbreviations

The following is a list of some of the abbreviations and acronyms and what they represent which are present in this report.

- FEM - Finite Element Method
- VM - Von-Mises
- T - Tresca
- LIH - linear isotropic hardening
- PP - perfect plasticity
- RMSE - root mean square error

List of Figures

List of Tables

1 Introduction

1.1 Overview

This report forms part of the completion of the MEC 5068Z course. It covers the different aspects of the plasticity model; built in C++ using the `deal.ii` FEM library.

The model is capable of running the VM or T yield criteria with LIH or PP. The small strain assumption was made.

1.2 Document Outline

A summary of the theoretical background is presented in Section [2]. It covers the governing equations, Von-Mises and Tresca and yield-criteria and hardening.

Section [3] contains an explanation of `plasticity_model.cc` in the form of text descriptions and flow diagrams. Additionally, it provides a general explanation of how the code can be adjusted to incorporate mixed (isotropic and kinematic) hardening.

Comparisons between the analytical and numerical results, for various test cases, is presented in Section [4]. Section [5] provides an assessment of the objectives of the model.

2 Theoretical Background

2.1 Governing Equations

$$\text{Linear momentum balance} \qquad \rho\frac{\mathrm{D}\boldsymbol{v}}{\mathrm{D}t} - \operatorname{div}\boldsymbol{\sigma} - \rho\boldsymbol{b} = 0, \tag{1}$$

$$\text{Infinitesimal strain} \qquad \boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right) \tag{2}$$

2.2 Classical Plasticity

The conditions under which a material transitions from elastic to plastic behaviour are defined by yield criteria. VM and T are two yield criteria which are explained below.

2.2.1 Von-Mises

The VM yield criteria states that plastic loading begins when the second principal scalar invariant of the stress deviator (J_2) reaches a critical value.

$$\mathbf{s} = \boldsymbol{\sigma} - \frac{1}{3}(\operatorname{tr}\boldsymbol{\sigma})\mathbf{I} \tag{3}$$

$$J_2 = \frac{1}{2}\operatorname{tr}(\mathbf{s}^2) \tag{4}$$

The yield function for the VM yield criterion takes on the following form:

$$f(\boldsymbol{\sigma}) = J_2 - \frac{\sigma_y^2}{3}. \tag{5}$$

2.2.2 Tresca

The T yield criteria assumes that a material will start to behave plastically once the maximum shear stress reaches a critical value. The spectral decomposition of the stress tensor,

$$\boldsymbol{\sigma} = \sum_{i=1}^{3}\sigma_i\mathbf{e}_i\otimes\mathbf{e}_i \tag{6}$$

where σ_i are the principal values and $\boldsymbol{e_i}$ the associated eigenvectors (of the stress tensor). The maximum and minium principal stress needs to be identified:

$$\sigma_{\max} = \max(\sigma_1, \sigma_2, \sigma_3), \tag{7}$$

$$\sigma_{\min} = \min(\sigma_1, \sigma_2, \sigma_3). \tag{8}$$

The yield function for the T yield criterion is

$$f(\boldsymbol{\sigma}) = \frac{1}{2}(\sigma_{\max} - \sigma_{\min}) - \sigma_y. \tag{9}$$

2.2.3 Hardening

The increase in stress with an increase in strain, in the plastic range, is referred to as hardening behaviour. Perfect plastic behaviour occurs if there is no such increase. Linear hardening refers to the linear increase in stress with an increase in strain in the plastic range. The phenomena mentioned above can be seen in Figure [1].

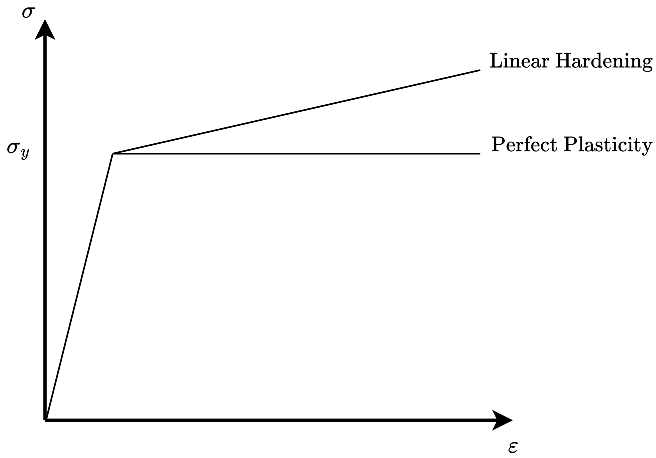


Figure 1: PP and linear hardening

Various hardening models exist however, only isotropic and kinematic hardening will be discussed further with a focus on linear hardening. The yield function is represented as $f(\boldsymbol{\sigma})$ where $f(\boldsymbol{\sigma}) \leq 0$. If $f(\boldsymbol{\sigma}) < 0$ then the loading is elastic and if $f(\boldsymbol{\sigma}) = 0$ then it is plastic.

The plastic strain increment can be computed with

$$d\boldsymbol{\varepsilon}^p = \xi \nabla f(\boldsymbol{\sigma}) = \xi \frac{\partial f}{\partial \boldsymbol{\sigma}}, \quad \xi \geq 0 \tag{10}$$

which implies the complementarity condition:

$$\xi \geq 0, \quad f(\boldsymbol{\sigma}) \leq 0, \quad \xi f(\boldsymbol{\sigma}) = 0.$$

The complementarity condition states that if the loading is elastic ($f(\boldsymbol{\sigma}) < 0$) then the plastic multiplier is 0 ($\xi = 0$). Additionlly, if the loading is plastic ($f(\boldsymbol{\sigma}) = 0$) then the plastic multiplier has to be greater than 0 ($\xi > 0$).

Perfect Plasticity

During PP the yield surface does not change. The yield function associated with PP (in one-dimension):

$$f(\sigma) = |\sigma| - \sigma_y \leq 0. \tag{11}$$

Isotropic Hardening

Isotropic hardening is characterised by a uniform expansion of the yield surface (in all directions) in stress space as plastic deformation occurs. The yield function which incorporates isotropic hardening is

$$f(\sigma, q) = |\sigma| - (\sigma_y + q) \leq 0. \tag{12}$$

For LIH

$$q = H\gamma. \tag{13}$$

The variable q, can be thought of as, how much the yield surface has expanded.

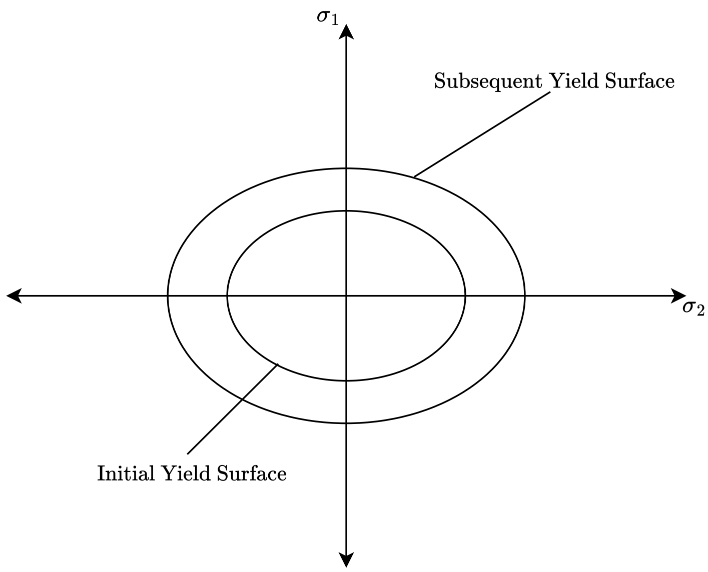


Figure 2: Isotropic hardening in two-dimensional stress space

Kinematic Hardening

Kinematic hardening describes how the yield surface translates in stress space as plastic deformation progresses. The yield surface does not change its size (as seen in isotropic hardening) or shape. The yield function which incorporates kinematic hardening is

$$f(\sigma, \bar{q}) = |\sigma - \bar{q}| - \sigma_y \leq 0. \tag{14}$$

$\bar{q}$ is the back-stress. The centre of the yield surface gets translated by $\bar{\boldsymbol{q}}$ which can be seen in the Figure [3].

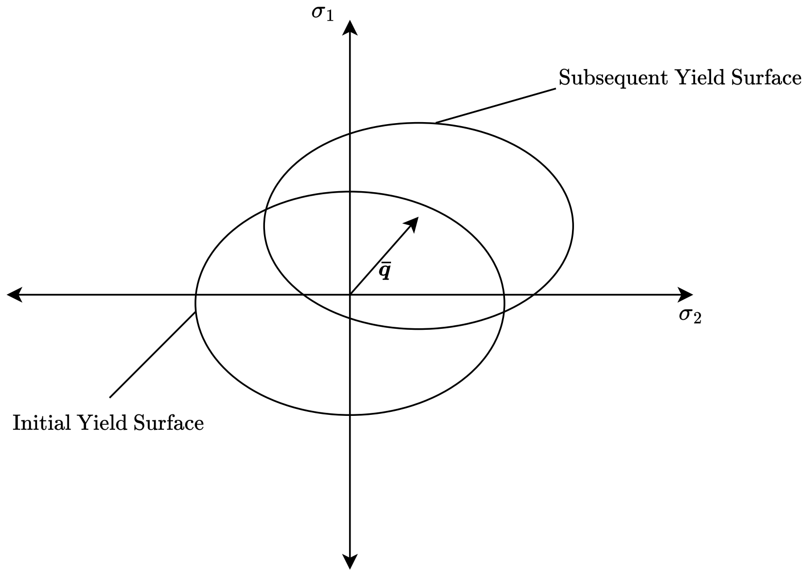


Figure 3: Kinematic hardening in two-dimensional stress space

3 Model Explanation

3.1 Classes and Functions

This chapter highlights the classes of `plasticity_model.cc` and provides an explanation of each function (accompanied by a flow diagram, where appropriate) in the every class.

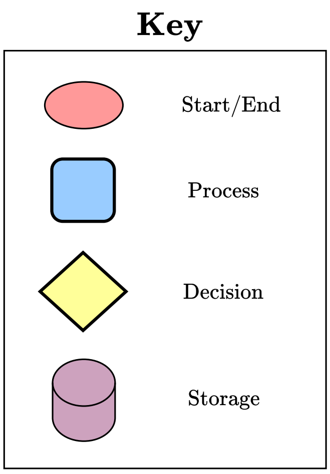
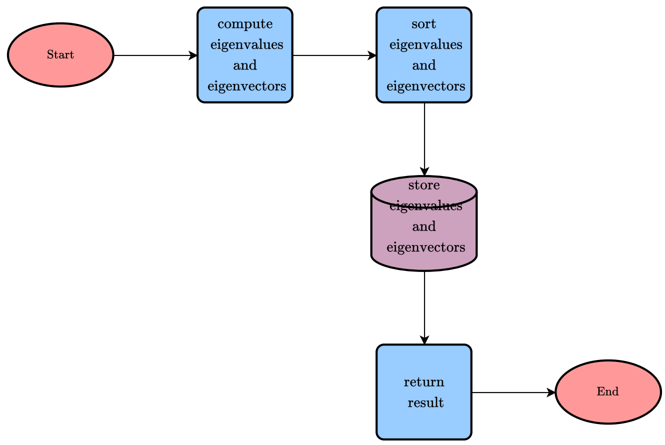


Figure 4: Code flow diagram key

3.1.1 Utility Functions

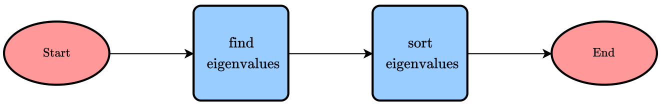
`compute_principal_values_vectors(const SymmetricTensor<2, dim> &A)`

The function computes the eigenvalues and eigenvectors of a second-order tensor (`A`). The eigenvalues are arranged in descending order, and the eigenvectors are placed in corresponding order.



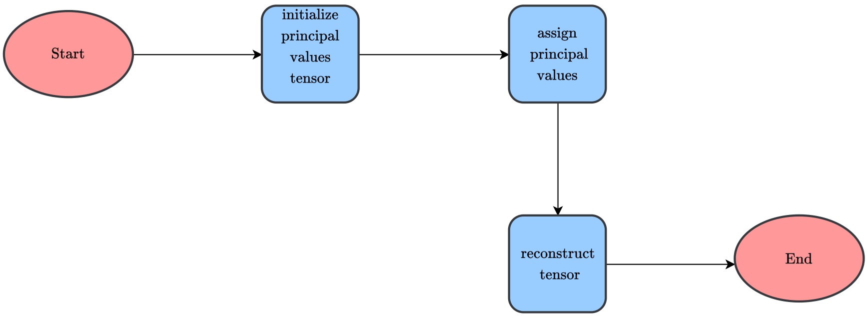
`compute_principal_values(const SymmetricTensor<2, dim> &A)`

The function computes the eigenvalues of a second-order tensor (`A`) and places them in descending order.



`reconstruct_tensor(const std::array<double, dim> &principal_values, const Tensor<2, dim> &eigenvectors_tensor)`

This function is used to transform a tensor from its principal directions back to the original coordinate system.



`yield_stress(double sigma_y_old, double H, double epsilon_p)`

The function computes the new yield stress as a function of the original yield stress, linear hardening parameter, and the accumulated plastic strain.

`formatValue(T value, int intDigits = 0, int mantissaPrecision = 2, int exponentDigits = 3)`

The function is used to format integers and floating-point numbers into strings with specific formatting requirements. It handles both integer and floating-point types differently to ensure the output is consistent and readable.

3.1.2 `PlasticityProblem` Class

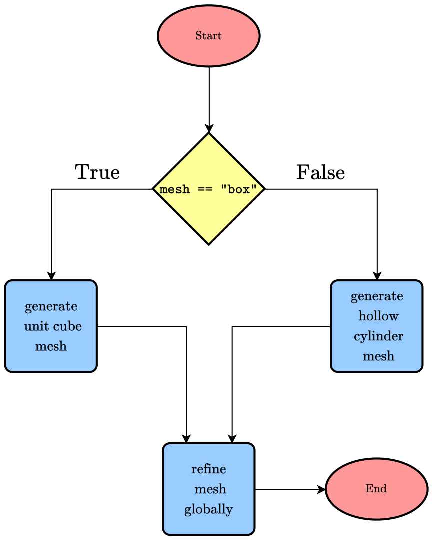
The `PlasticityProblem` class is responsible for setting up and solving the plasticity problem.

`declare_parameters(ParameterHandler &prm)`

The function is used to declare the parameters, which can be adjusted, in the input `.prm` file. These parameters include various settings and configurations required to set up and solve the plasticity problem, such as the polynomial degree of the finite element space, the number of initial refinements, the number of time-steps, the output directory, and other problem-specific parameters. This function ensures that all necessary parameters are defined and can be read from the input file.

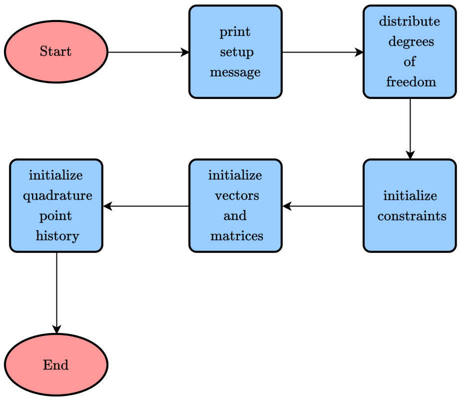
`make_grid(std::string mesh)`

This function is responsible for generating the computational grid (mesh) based on the specific mesh type. It creates either a unit cube mesh or a hollow cylinder mesh, depending on the input parameter. Additionally, it refines the mesh globally according to the initial refinement level specified.



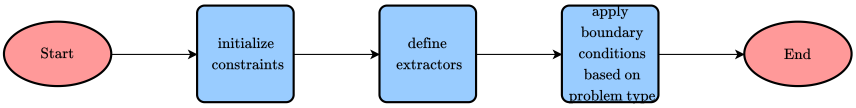
`setup_system()`

The function is responsible for setting up the finite element system. This includes initializing the degrees of freedom, applying constraints, and preparing data structures for the solution process.



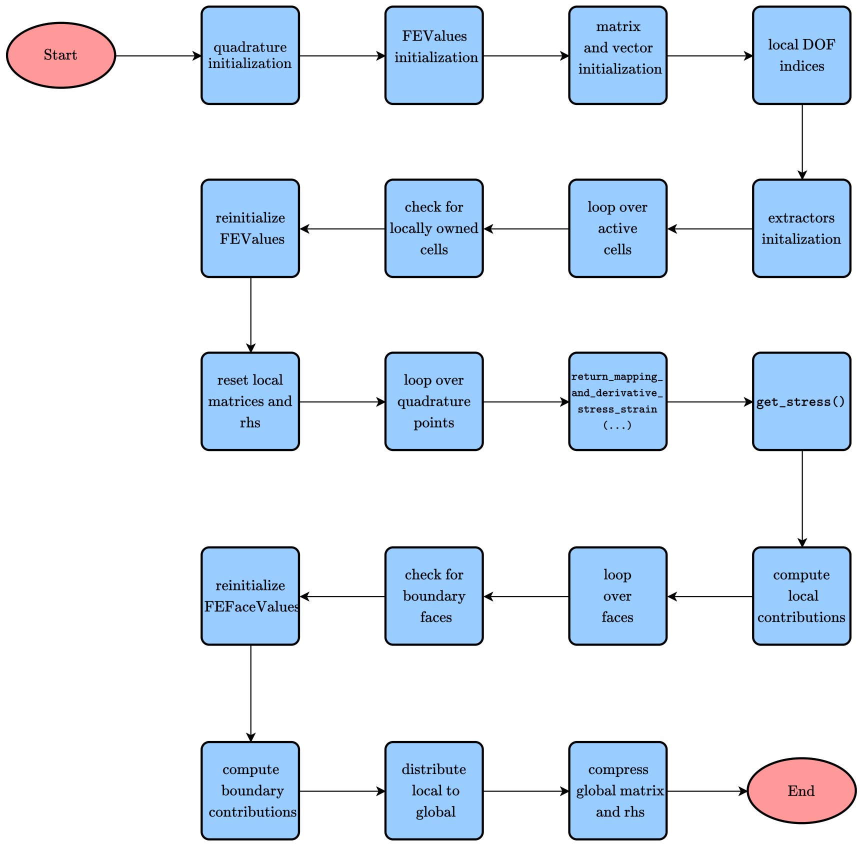
`compute_dirichlet_constraints(const double displacement, std::string problem_type)`

The function is responsible for computing the Dirichlet boundary conditions based on the problem type and applied displacement. It sets up the constraints for the degrees of freedom on the boundaries where Dirichlet conditions are applied. The constraints are problem-specific.



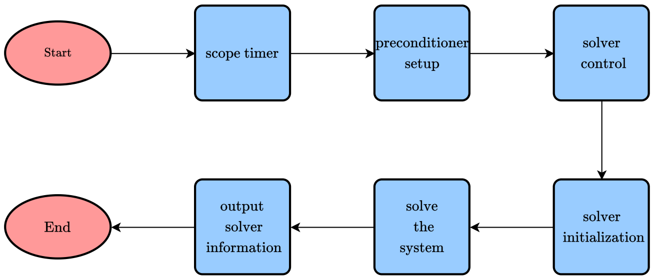
```
assemble_newton_system(const TrilinosWrappers::MPI::Vector &solution,
const TrilinosWrappers::MPI::Vector &old_solution, const bool rhs_only = false,
unsigned int n_t_steps = 1, unsigned int t_step = 0)
```

This function is responsible for assembling the system matrix and right-hand side vector for the Newton-Raphson method used to solve the nonlinear system of equations. This function computes the contributions from each cell in the mesh and assembles them into the global system matrix and right-hand side vector.



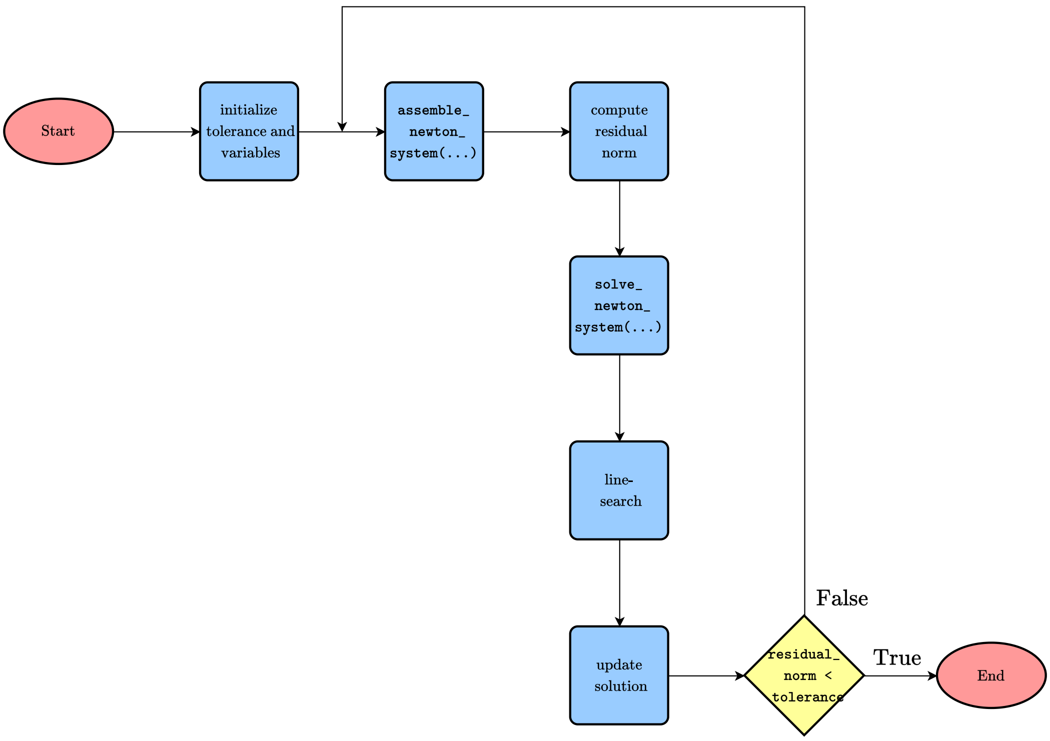
`solve_newton_system(TrilinosWrappers::MPI::Vector &newton_increment)`

This function is responsible for solving the linear system of equations that arises in each iteration of the Newton-Raphson method. This function uses an iterative solver to find the increment in the solution vector that will be used to update the current solution.



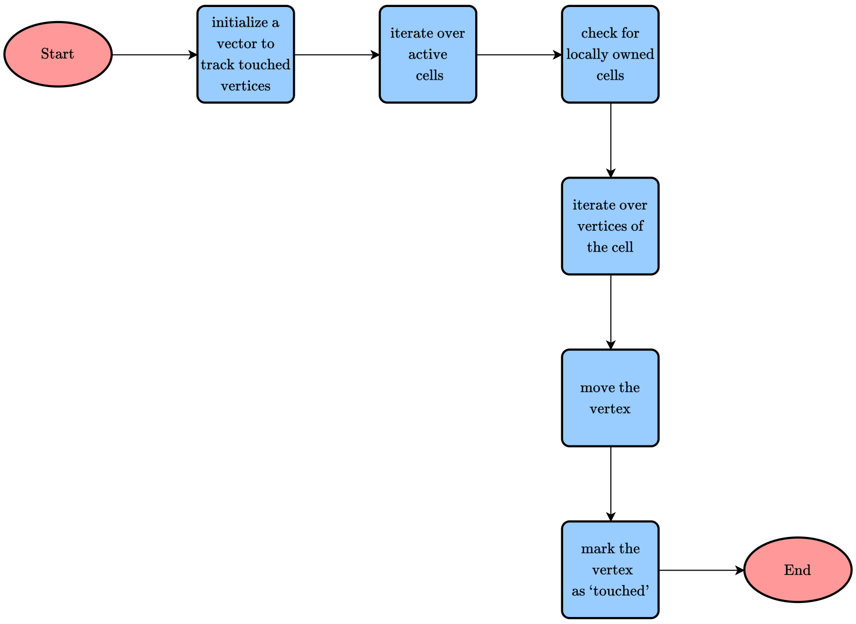
`solve_newton(unsigned int n_t_steps = 1, unsigned int t_step = 0)`

The function is responsible for solving the nonlinear system of equations using the Newton-Raphson method. This function iteratively assembles the system matrix and right-hand side vector, solves the linear system, and updates the solution until convergence is achieved.



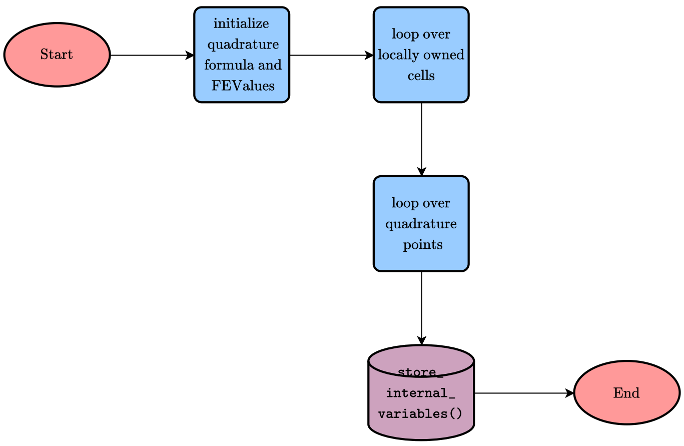
`move_mesh(const TrilinosWrappers::MPI::Vector &displacement) const`

The function is responsible for updating the mesh geometry by displacing its vertices according to the computed displacement field. This function iterates over the cells of the mesh, applies the displacement to the vertices, and ensures that each vertex is only moved once.



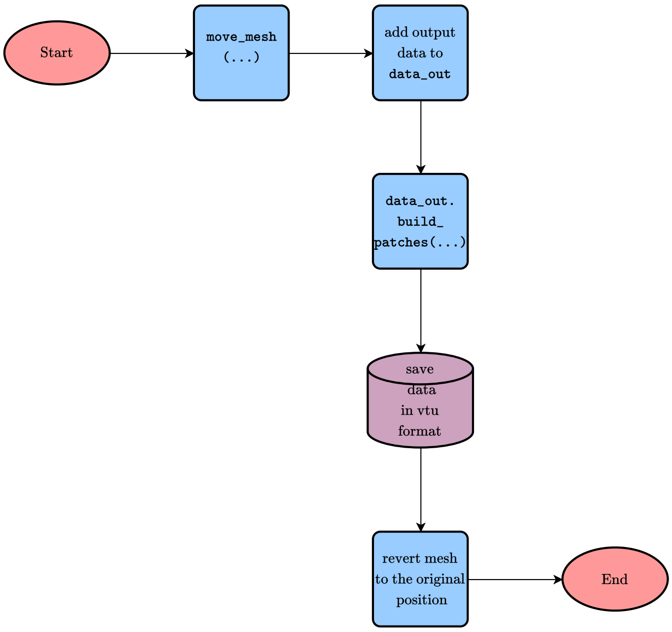
`store_internal_variables()`

The function is responsible for storing the internal state variables at each quadrature point for the current time step. This function iterates over the cells of the mesh, retrieves the quadrature point history, and stores the current values of the internal variables.



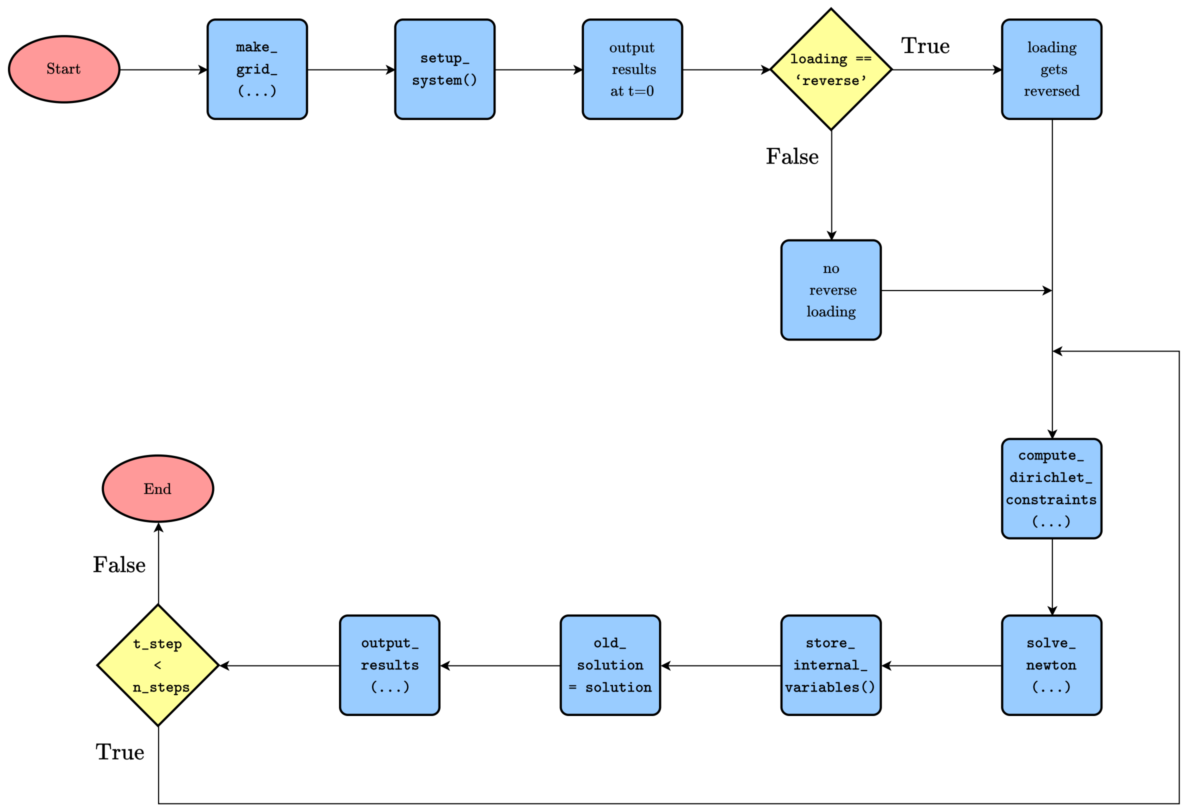
`output_results(const unsigned int current_time_step, const std::string output_name = "solution")`

The function is responsible for generating and writing the graphical output of the simulation results at a given time step. This function moves the mesh according to the current solution, attaches the degrees of freedom to the data output object, and writes the results to a file.



`run()`

The run function in the `PlasticityProblem` class orchestrates the entire simulation process. It begins by resetting the computing timer and generating the initial mesh using the `make_grid()` function. The system is then set up by calling `setup_system()`, which initializes the degrees of freedom, constraints, and other system components. Various constraints are initialized and closed, including hanging node constraints and Dirichlet constraints. The function then generates and writes the initial graphical output of the mesh and solution using the `output_results()` function. The main simulation loop follows, managing the time-stepping process, updating the displacement, and solving the system at each time step.



3.1.3 `ConstitutiveLaw` Class

The class is responsible for defining the material behaviour under different loading conditions. It encapsulates the constitutive equations that describe how the material responds to stress and strain, including both elastic and plastic deformations.

```
derivative_of_isotropic_tensor(Tensor<1, dim> x, Tensor<2, dim> e,
SymmetricTensor<2, dim> Y, SymmetricTensor<2, dim> dy_dx) const
```

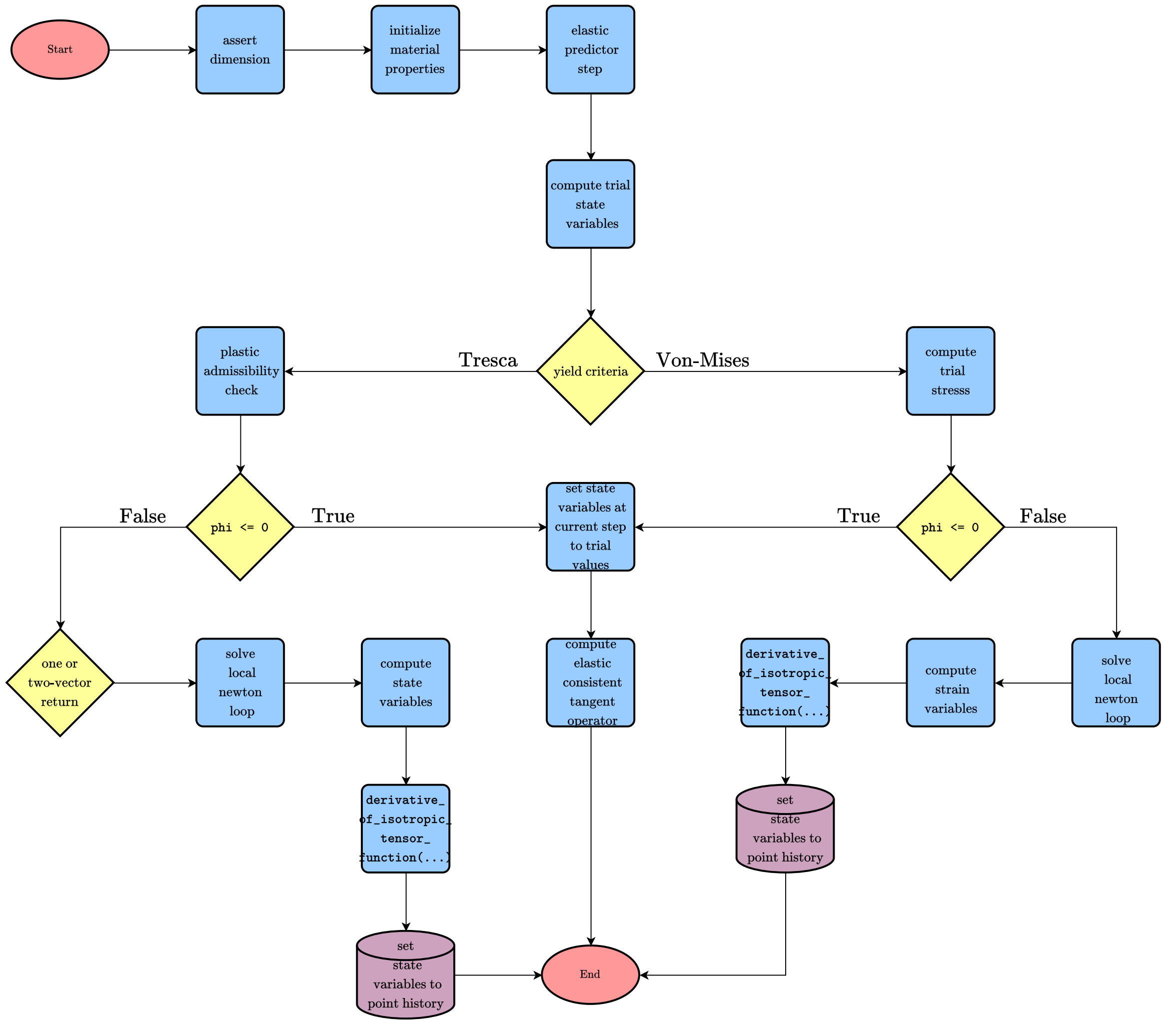
The function computes the derivative of an isotropic tensor function with respect to its input tensor. This function is used to determine the Tresca consistent tangent operator. Refer to **Box A.6**, for the algorithm on computation of general isotropic tensor function, in [1].

```
return_mapping_and_derivative_stress_strain(const SymmetricTensor<2,
dim> &delta_strain, std::shared_ptr<PointHistory<dim>> &qph, std::string
yield_criteria) const
```

The function performs the return-mapping algorithm to update the stress and strain state variables based on the given strain increment. It also computes the consistent tangent operator for the VM yield condition, which is the derivative of the stress with respect to the strain. The function handles both the T and VM yield criteria, determining whether the material response is elastic or plastic and updating the internal variables accordingly. The return-mapping algorithms can be found in [1]

The return-mapping for the VM yield criteria is always a one-vector return due to the shape of the yield surface. The return-mapping for the T yield criteria could be a one or two-vector return. The VM and T return-mapping was set up to handle general isotropic hardening, even though the return-mapping could be solved in closed form for the PP and LIH cases (which were tested).

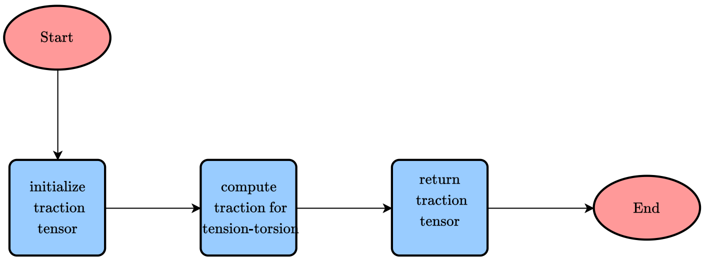
Additionally, the principal stress derivatives are computed for the T yield criteria. These principal stress derivatives are passed as arguments to the `derivative_of_isotropic_tensor(...)` function, which computes the consistent tangent operator for the T yield-criteria.



3.1.4 `BoundaryForce` Class

`value(const Point<dim> &p)`

The function is responsible for computing the traction (force per unit area) at a given point on the boundary of the domain. This function calculates the traction, for the tension and torsion of a hollow-cylinder problem, based on the parameters such as inner radius, outer radius, normal force, and torque.



3.1.5 `PointHistory` Class

The `PointHistory` class in the `PlasticityModel` namespace is responsible for storing and managing the state variables at each quadrature point in the finite element mesh. These state variables include stress, strain, elastic strain, plastic strain, and other internal variables necessary for the plasticity model. The class provides methods to set and get these variables, as well as to store and retrieve the state variables from the previous time step. This is essential for the return-mapping algorithm used in the plasticity model to update the stress and strain state variables based on the given strain increment.

3.2 Mixed Hardening Implementation

Real-life materials experience a combination of isotropic and kinematic hardening during plastic loading [1]. The yield surface changes shape and translates through stress space.

3.2.1 Von-Mises

This subsection will present an explanation on how the return-mapping and other parts of the `plasticity_model.cc` will have to be adjusted to incorporate isotropic and kinematic hardening effects. Refer to **Section 7.6**, for a detailed explanation of the algorithm, in [1].

Two key variables for the kinematic hardening implementation, which is not essential to isotropic hardening, are the back stress ($\boldsymbol{\beta}$) and the relative stress ($\boldsymbol{\eta}$). They would have to be accounted for in the `store_internal_variables()` function. The back stress is a function of the accumulated plastic strain.

$$\boldsymbol{\eta} = \boldsymbol{s} - \boldsymbol{\beta} \tag{15}$$

Therefore the corresponding variables, getter functions and setter functions will need to added to the `PointHistory` class. This would make it possible to retrieve and save the back stress and and relative stress between time-steps.

At the start of the return-mapping the trial back stress and relative stress will need to be initialized and set equal to the back stress and relative stress, at time step n; which can be retrieved from the `PointHistory` class, respectively. Additionally, the trial relative effective stress ($\bar{q}^{\text{trial}}_{n+1}$) equation is updated to use the relative stress instead of the deviatoric stress only; which was the case for isotropic hardening.

$$\bar{q}^{\text{trial}}_{n+1} \equiv \sqrt{\frac{3}{2}}\,\|\boldsymbol{\eta}^{\text{trial}}_{n+1}\| \tag{16}$$

As before, if the step is elastic then the state variables at $n+1$ is set to their corresponding trial values at time step $n+1$. If the step is plastic then a local Newton loop needs to be solved. The residual equation, which needs to be solved in the local Newton loop, for the plastic step changes from the one which only incorporates isotropic hardening to:

$$\widetilde{\Phi}(\Delta\gamma) \equiv \bar{q}^{\text{trial}}_{n+1} - 3G\Delta\gamma - \bar{\beta}(\bar{\varepsilon}^{p}_{n} + \Delta\gamma) + \bar{\beta}_n - \sigma_y(\bar{\varepsilon}^{p}_{n} + \Delta\gamma) = 0. \tag{17}$$

After solving for $\Delta\gamma$ the state variables at time $n+1$ can be computed with

$$\bar{\varepsilon}^{p}_{n+1} := \bar{\varepsilon}^{p}_{n} + \Delta\gamma, \tag{18}$$

$$\bar{\beta}_{n+1} := \bar{\beta}(\bar{\varepsilon}^{p}_{n+1}), \tag{19}$$

$$\boldsymbol{\beta}_{n+1} := \boldsymbol{\beta}_n + \sqrt{\frac{2}{3}}(\bar{\beta}_{n+1} - \bar{\beta}_n)\frac{\boldsymbol{\eta}^{\text{trial}}_{n+1}}{\|\boldsymbol{\eta}^{\text{trial}}_{n+1}\|}, \tag{20}$$

$$p_{n+1} := p^{\text{trial}}_{n+1}, \tag{21}$$

$$\boldsymbol{s}_{n+1} := \boldsymbol{s}^{\text{trial}}_{n+1} - 2G\Delta\gamma\sqrt{\frac{3}{2}}\frac{\boldsymbol{\eta}^{\text{trial}}_{n+1}}{\|\boldsymbol{\eta}^{\text{trial}}_{n+1}\|}, \tag{22}$$

$$\boldsymbol{\sigma}_{n+1} := \boldsymbol{s}_{n+1} + p_{n+1}\boldsymbol{I}, \tag{23}$$

$$\boldsymbol{\varepsilon}^{e}_{n+1} := \frac{1}{2G}\boldsymbol{s}_{n+1} + \frac{1}{3}\varepsilon^{\text{e trial}}_{\text{v}\ n+1}\boldsymbol{I}. \tag{24}$$

After computing the new state variables they should get stored in the point history.

3.2.2 Tresca

For the T yield criteria the residual equation for the one-vector and two-vector (left and right corner) will need to be adjusted accordingly (in a similar manner to Equation [18] which is for the VM yield criteria case) in order to incorporate kinematic hardening.

4 Results and Discussion

Refer to Appendix [7.3] for the parameters in the `.prm` file which was used to generate the numerical results.

4.1 Material Properties

The following parameters were obtained from [2]. These parameters were used to obtain the analytical and numerical results presented in this section.

- $E = 200$ GPa
- $\nu = 0.3$
- $\kappa = 166.67$ GPa
- $\mu = 76.92$ GPa
- $\sigma_0 = 400$ MPa
- $H = 1.55$ GPa

4.2 Analytical Solutions

4.2.1 Uniaxial Stress

The first analytical solution, which the model was tested against, was a uniaxial stress problem. A unit cube with an applied displacement on the top face, with the opposite face fixed, was used, as shown in Figure [5].

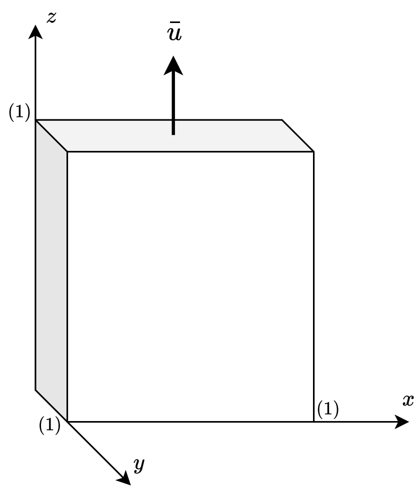


Figure 5: Uniaxial test case

This is a one-dimensional problem therefore the relationship between the stress and strain (in the axial direction) reduces to the following scalar equation

$$\sigma = E\varepsilon. \tag{25}$$

The strain can be computed with

$$\varepsilon = \frac{\bar{u}}{L_0} \tag{26}$$

where $\bar{u}$ is the applied displacement. Yielding will start when

$$\bar{u} = \frac{\sigma_0 L_0}{E}. \tag{27}$$

The stress-strain curve in the plastic region has the following slope for LIH:

$$E_{\text{tan}} = \frac{EH}{E+H}. \tag{28}$$

Using Equations (27) and (28) the stress-strain relationship, in the plastic region, was derived to be

$$\sigma = \left(\varepsilon - \frac{\sigma_0}{E}\right) E_{\text{tan}} + \sigma_0. \tag{29}$$

Therefore for the uniaxial stress case, with LIH, the stress-strain relationship takes on the following piecewise relationship

$$\sigma = \begin{cases} E\varepsilon, & \text{if } \bar{u} \leq \frac{\sigma_0 L_0}{E}, \\ \left(\varepsilon - \frac{\sigma_0}{E}\right) E_{\text{tan}} + \sigma_0, & \text{if } \bar{u} > \frac{\sigma_0 L_0}{E}. \end{cases} \tag{30}$$

The cube experiences volumetric strain during elastic loading which can be computed with

$$\frac{\Delta V}{V_0} = \varepsilon(1-2\nu). \tag{31}$$

The volume of the rectangular prism, during elastic loading, can be determined with

$$V_e = \frac{\bar{u}}{L_0}(1-2\nu)V_0 + V_0. \tag{32}$$

Therefore, the volume, at the onset of yielding, can be computed as

$$V_p = \frac{\sigma_0}{E}(1-2\nu)V_0 + V_0. \tag{33}$$

The prism remains at this volume during plastic loading because it is incompressible. Figure [6] shows the top face, of the prism, with labels representing the displacements on each of the edges.

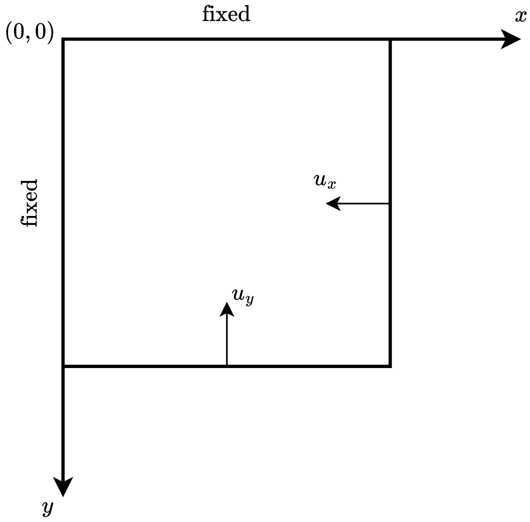


Figure 6: Top face of the prism

Due to symmetry; $u_x = u_y$.

$$u_x = u_y = \begin{cases} \sqrt{\frac{V_e}{1+\bar{u}}} - 1, & \text{if } \bar{u} \leq \frac{\sigma_0 L_0}{E}, \\ \sqrt{\frac{V_p}{1+\bar{u}}} - 1, & \text{if } \bar{u} > \frac{\sigma_0 L_0}{E}. \end{cases} \tag{34}$$

4.2.2 Plane Shear

The plane shear problem can be simplified to two dimensions. The problem set-up is a square which is fixed at the bottom, in all directions, and an applied displacement (δ) at the top in the x-direction. See Figure [7].

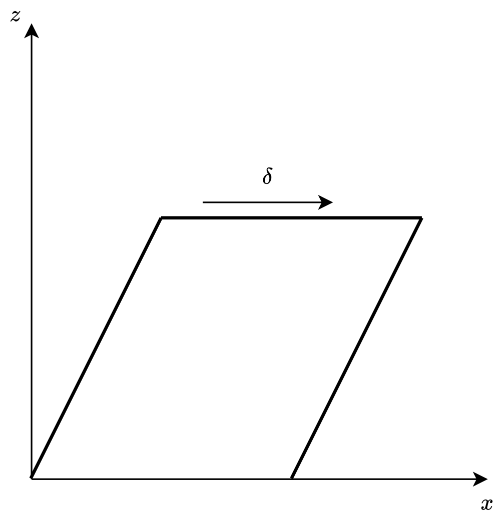


Figure 7: Plane shear

$$u_y = u_z = 0 \tag{35}$$
$$u_x = \alpha z \tag{36}$$
$$\alpha = \frac{\delta}{h} \tag{37}$$

Using the small strain assumption the strain tensor reduces to

$$\boldsymbol{\varepsilon} = \begin{bmatrix} 0 & 0 & \frac{1}{2}\alpha \\ 0 & 0 & 0 \\ \frac{1}{2}\alpha & 0 & 0 \end{bmatrix} \tag{38}$$

The stress tensor, for the elastic region, can be determined with

$$\sigma_{ij} = \lambda \delta_{ij} \varepsilon_{kk} + 2\mu \varepsilon_{ij}. \tag{39}$$

Therefore

$$\boldsymbol{\sigma} = \begin{bmatrix} 0 & 0 & \mu\alpha \\ 0 & 0 & 0 \\ \mu\alpha & 0 & 0 \end{bmatrix} \tag{40}$$

τ will be used to represent the shear stress.

$$\tau = \sigma_{13} = \sigma_{31} \tag{41}$$

The equations needed to determine the shear stress, in the plastic region, and the accumulated plastic strain are presented below. Refer to Appendix [7.1] for the full derivation.

Von-Mises

Initial yield will occur when

$$\tau = \frac{\sigma_0}{\sqrt{3}}. \tag{42}$$

The accumulated plastic strain can be computed with

$$\gamma = \int_{\frac{\sigma_0}{\sqrt{3}}}^{\sigma_f} \frac{3}{\sqrt{3}H}\, d\sigma. \tag{43}$$

The shear stress equation for the elastic and plastic regions can be seen below

$$\tau = \begin{cases} \mu\delta, & \text{if } \delta \leq \frac{\sigma_0}{\sqrt{3}\mu}, \\ \frac{\sigma_0}{\sqrt{3}} + E_{\tan}(\frac{1}{2}\delta - \frac{\sigma_0}{2\sqrt{3}\mu}), & \text{if } \delta > \frac{\sigma_0}{\sqrt{3}\mu}, \end{cases} \tag{44}$$

where

$$E_{\tan} = \frac{2\sqrt{3}\mu H}{\sqrt{3}H + 6\mu}. \tag{45}$$

Tresca

Initial yield will occur when

$$\tau = \frac{\sigma_0}{2}. \tag{46}$$

The accumulated plastic strain can be computed with

$$\gamma = \int_{\frac{\sigma_0}{2}}^{\sigma_f} \frac{2}{H}\, d\sigma. \tag{47}$$

The shear stress equation for the elastic and plastic regions can be seen below

$$\tau = \begin{cases} \mu\delta, & \text{if } \delta \leq \frac{\sigma_0}{2\mu}, \\ \frac{\sigma}{2} + E_{\tan}(\frac{1}{2}\delta - \frac{\sigma_0}{4\mu}), & \text{if } \delta > \frac{\sigma_0}{2\mu}, \end{cases} \tag{48}$$

where

$$E_{\tan} = \frac{2\mu H}{H + 4\mu}. \tag{49}$$

4.2.3 Tension and Torsion of a Thin-Walled Cylinder

Below is a representation of the thin-walled cylinder geometry.

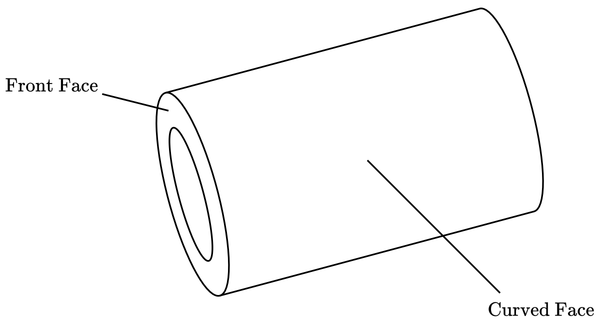


Figure 8: Thin-walled cylinder

The thickness (t) is much smaller than the outer radius (R_o). The Boundary conditions are comprised of a total normal force (N) spread across the front face and a torque (T) which is applied on the front face. The back face was fixed in all directions for the numerical simulations. The only non-zero components of stress are σ_{zz} and $\sigma_{z\theta}$.

$$\sigma_{zz} = \frac{N}{2\pi Rt}$$

$$\sigma_{z\theta} = \frac{T}{2\pi R^2 t}$$

The applied load takes a linear path (from 0 to some final stress $\boldsymbol{\sigma}_f$) which can be seen in Figure [9].

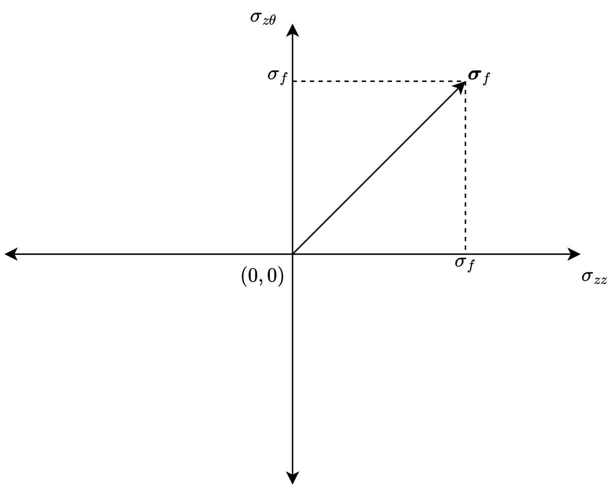


Figure 9: Load path

Set $\sigma_{z\theta} = \sigma_{zz} = \sigma$.

$$
\begin{aligned}
&\text{Principal stresses} \quad && \sigma_1, \sigma_2 = \tfrac{1}{2}\left[\sigma_{zz} \pm \sqrt{\sigma_{zz}^2 + 4\sigma_{z\theta}^2}\right] \\
&\text{Von-Mises in 2D} && \sigma_y - \sigma_1\sigma_2 + \sigma_2^2 \leq \sigma_y^2 \\
&\text{Tresca in 2D} && \max\{|\sigma_1 - \sigma_2|, |\sigma_1|, |\sigma_2|\} \leq \sigma_y
\end{aligned}
\tag{50}
$$

Using Equations (50) and assuming isotropic hardening; the following yield functions is obtained for the VM and T yield criteria

$$
f(\boldsymbol{\sigma}) = \sigma_{zz}^2 + a\sigma_{z\theta}^2 - (\sigma_Y + q)^2 \leq 0 \tag{51}
$$

$$
a = \begin{cases} 3 & \text{VM,} \\ 4 & \text{T.} \end{cases}
$$

Using Equation (51) it can be shown that yielding will start when $\sigma = \frac{\sigma_0}{\sqrt{1+a}}$. See Appendix [7.2] for the derivation of the plastic strain (the components can be seen in Equations (52) and (53) from the the point of yield up to some final stress).

$$
\varepsilon_{zz}^p = \frac{1+a}{2H\sqrt{1+a^2}} \left[\sigma\right]_{\frac{\sigma_0}{\sqrt{1+a}}}^{\sigma_f} \tag{52}
$$

$$
\varepsilon_{z\theta}^p = \frac{a(1+a)}{2H\sqrt{1+a^2}} \left[\sigma\right]_{\frac{\sigma_0}{\sqrt{1+a}}}^{\sigma_f} \tag{53}
$$

The accumulated plastic strain can be computed with

$$
\gamma = \sqrt{(\varepsilon_{zz}^p)^2 + (\varepsilon_{z\theta}^p)^2}. \tag{54}
$$

For the numerical simulation the traction boundary condition applied to the front face was,

$$
t_x = \frac{-Ty}{J}, \tag{55}
$$

$$
t_y = \frac{Tx}{J}, \tag{56}
$$

$$
t_z = \frac{N}{\pi(R_o^2 - R_i^2)}. \tag{57}
$$

The relationship between the stress components in cylindrical and polar coordinates are:

$$
\sigma_{xz} = -\sin(\theta)\sigma_{\theta z}, \tag{58}
$$

$$
\sigma_{yz} = \cos(\theta)\sigma_{\theta z}, \tag{59}
$$

$$
\sigma_{\theta z} = \sqrt{\sigma_{xz}^2 + \sigma_{yz}^2}. \tag{60}
$$

4.3 Comparison between analytical and numerical solutions

4.3.1 Uniaxial Stress

Below is a comparison between the numerical (produced by the plasticity model) and analytical solutions using the VM and T yield criteria with LIH. A tensile uniaxial displacement was applied to the top face.

$$\bar{u} = 0.01$$

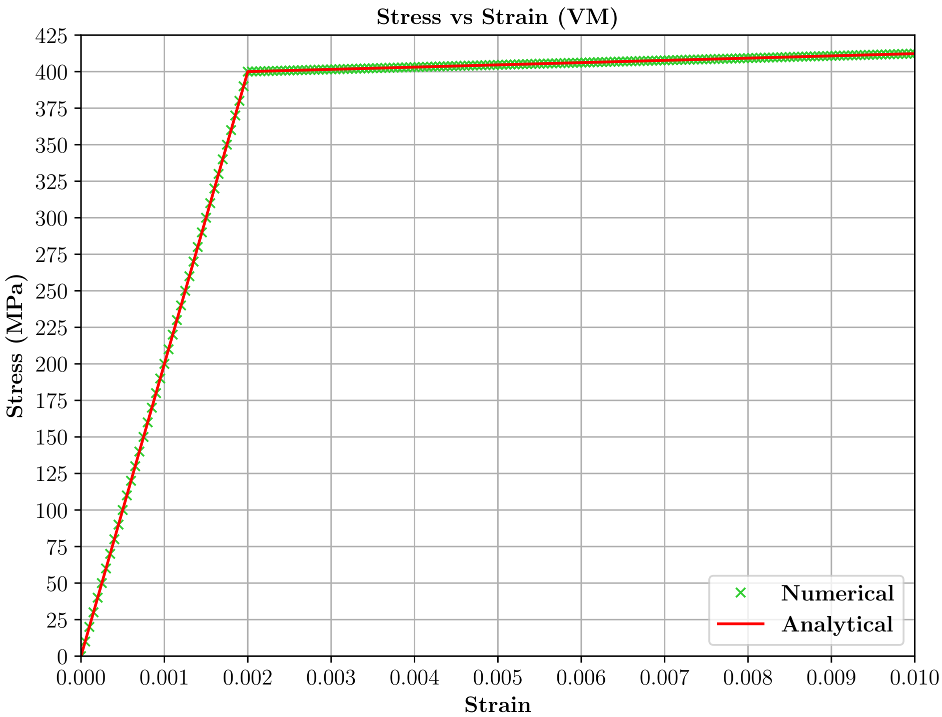


Figure 10: Uniaxial stress-strain curve for VM yield criteria with LIH

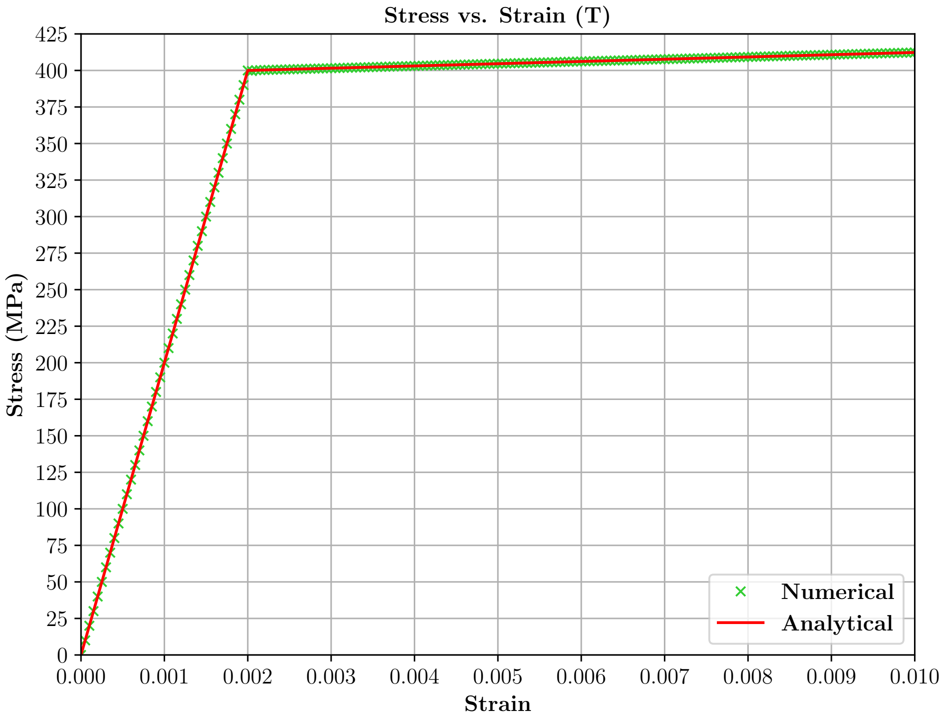


Figure 11: Uniaxial stress-strain curve for T yield criteria with LIH

Figures [10] and [11] show that the numerical results for the uniaxial stress, for both yield criteria, matches the analytical results.

Additionally, reverse loading was investigated. Below are the plots of reverse loading for both yield criteria.

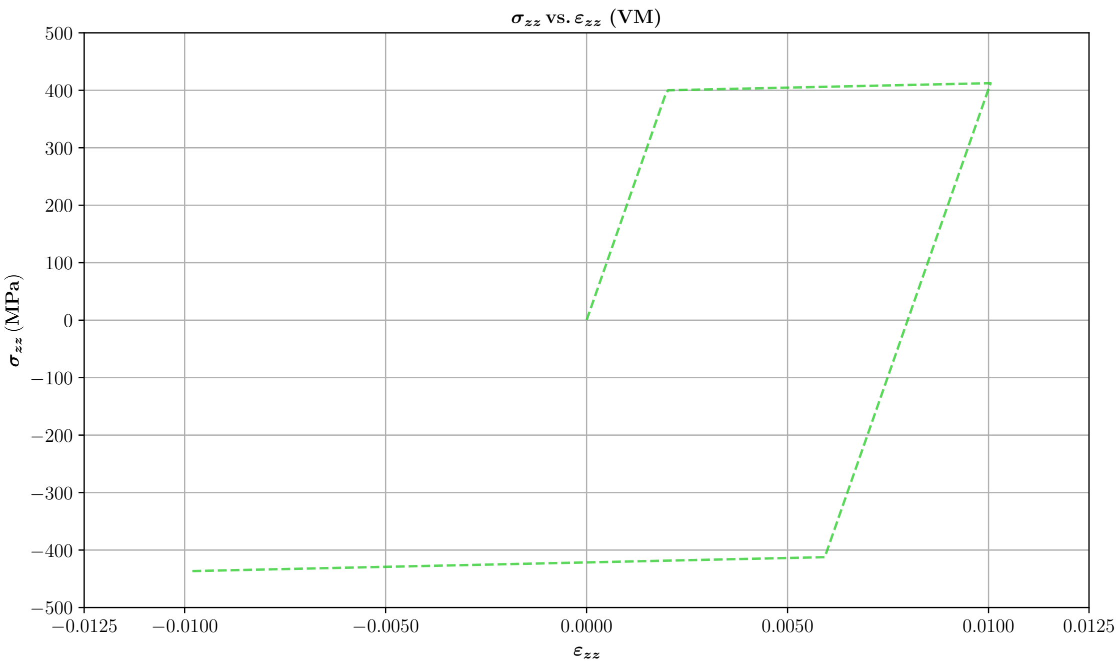


Figure 12: Uniaxial reversed loading stress-strain curve for VM yield criteria with LIH

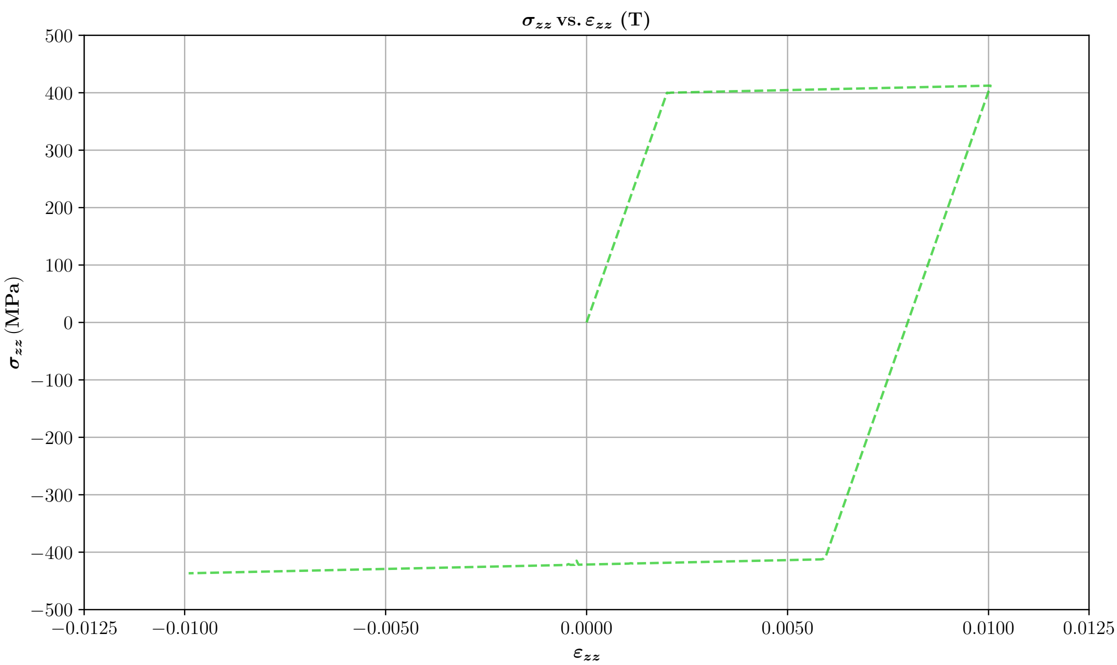


Figure 13: Uniaxial reversed loading stress-strain curve for T yield criteria with LIH

Looking at Figures [11] and [12] the model correctly captures the reversed loading stress response. The loading starts at $(0, 0)$ and the cube starts to deform elastically until the yield point is reached. Thereafter, it starts to deform plastically and the material hardens until the loading is reversed (from tensile to compressive). The material then deforms elastically again until it reaches the negative yield-point. Once the negative yield-point is reached it deforms plastically again.

In Figure [13] around the point $(0, -400)$ there is a 'kink' in the linear hardening region. This is an example of how the Newton loop fails to converge for some steps. These could result in the failed convergence of the entire numerical solution.

The model yields accurate results for small applied displacements and forces however it is not suitable for large strains. This is not surprising because it was built under the small strain assumption.

Below is a graph which compares the analytical and numerical displacement of one of the side-face in the x-direction.

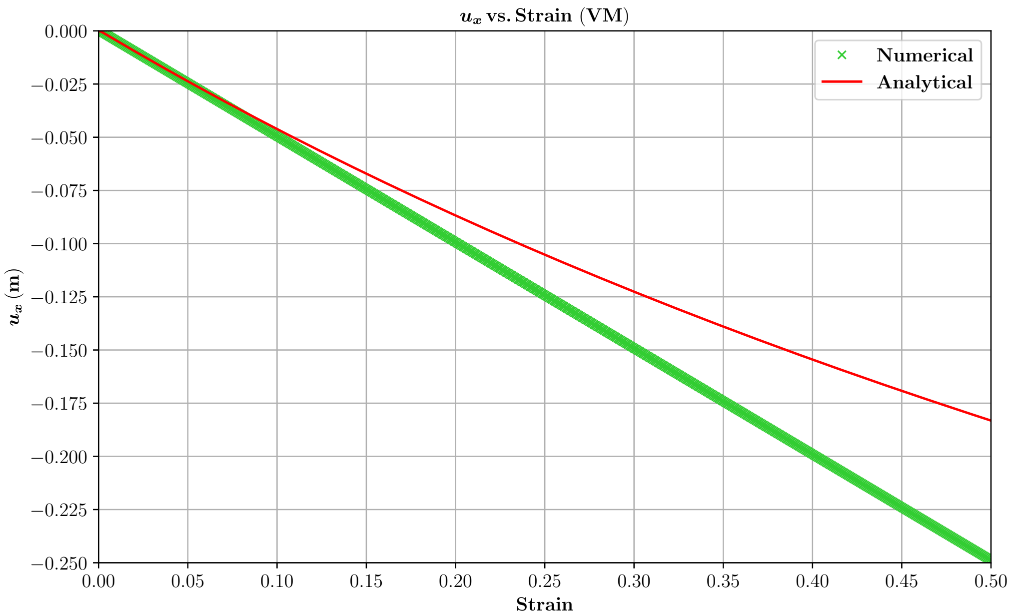


Figure 14: Analytical vs. numerical displacement for VM yield criteria with LIH

The same result is produced for the displacement in y-direction. Figure [14] displays that the displacement numerical solution is acceptable for strain values lower than 10%. A similar trend in displacement accuracy versus strain was observed for all the numerical solutions generated with the model using the VM yield criteria.

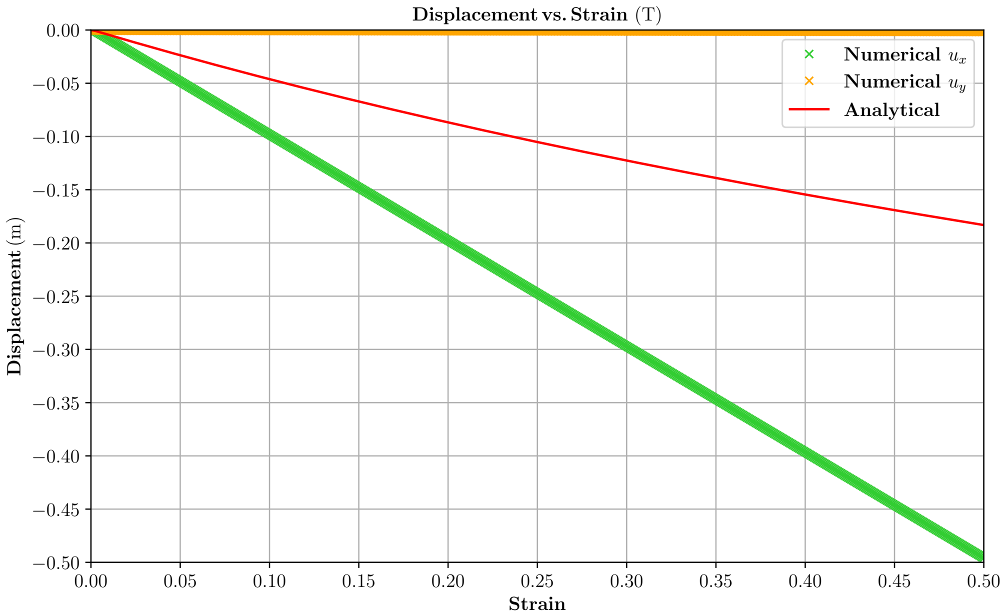


Figure 15: Analytical vs. numerical displacement for T yield criteria with LIH

The numerical displacement in the x and y-directions, for the T yield condition, is significantly different from their analytical counterparts. This can be seen in Figure [15]. It was expected that $u_x = u_y$ however all the displacement (in response to the applied displacement) took place on the face which was allowed to move in the x-direction. For small strains, half of u_x would equal the analytical displacement.

Below are the PP ($H = 0$) results produced by the plasticity model with the VM yield criteria. The model failed to converge for the T yield criteria case.

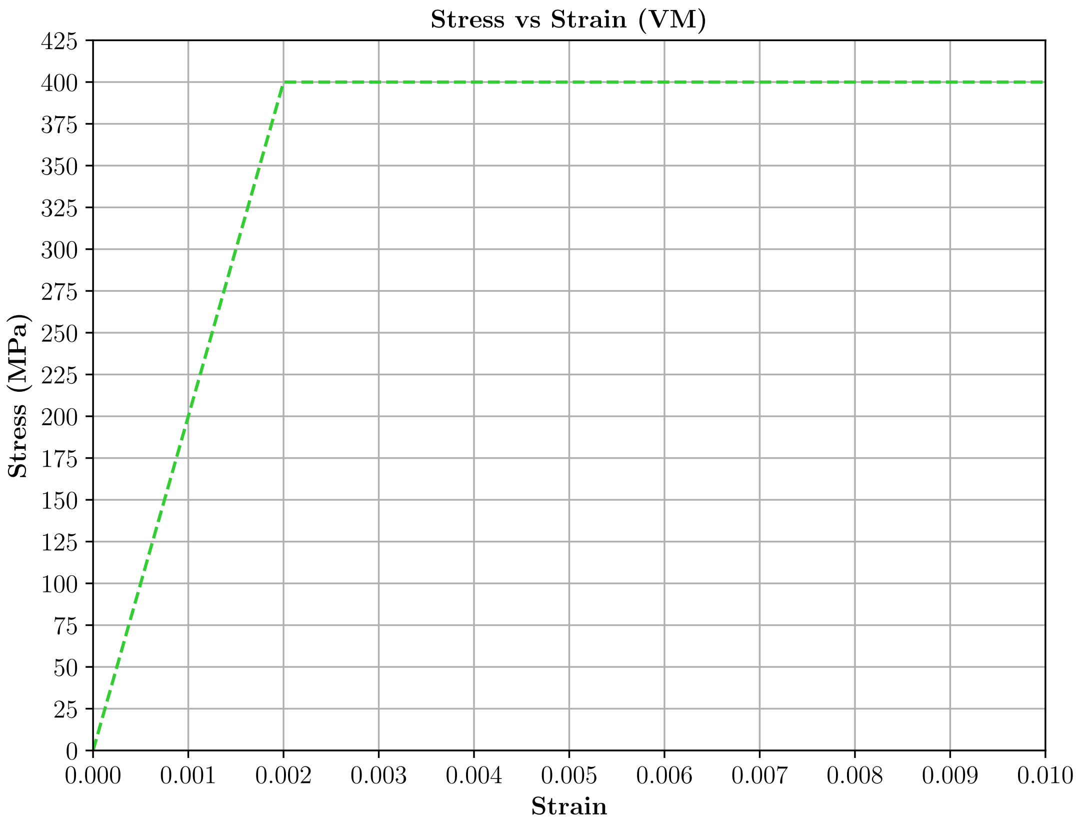


Figure 16: Uniaxial stress-strain curve for VM yield criteria with PP

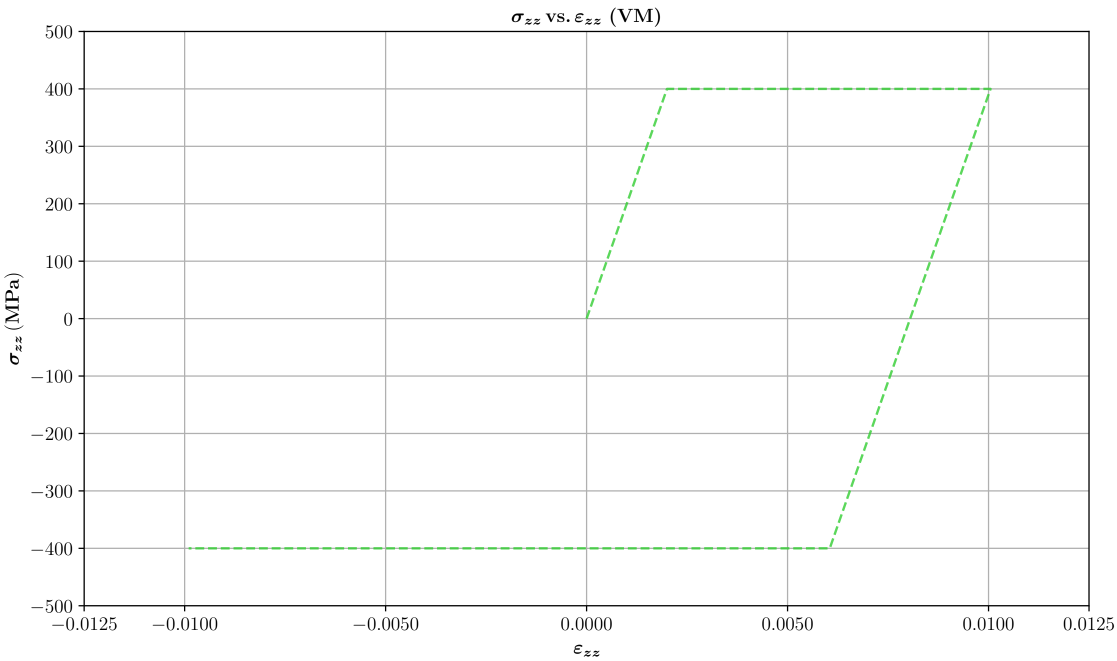


Figure 17: Uniaxial reversed loading stress-strain curve for VM yield criteria with PP

Figures [16] and [17] is evidence that the plasticity model captures the correct uniaxial stress behaviour for VM yield criteria with PP.

4.3.2 Plane Shear

Below are the results for the plane shear problem of a unit cube with $\delta = 0.01$.

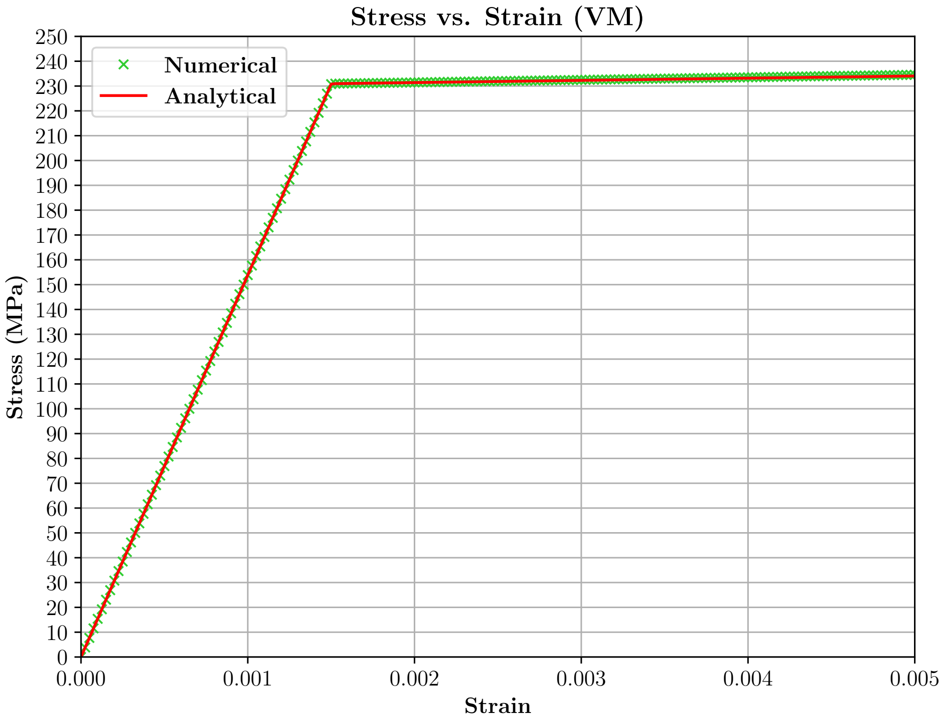


Figure 18: Plane shear stress-strain curve for VM yield criteria with LIH

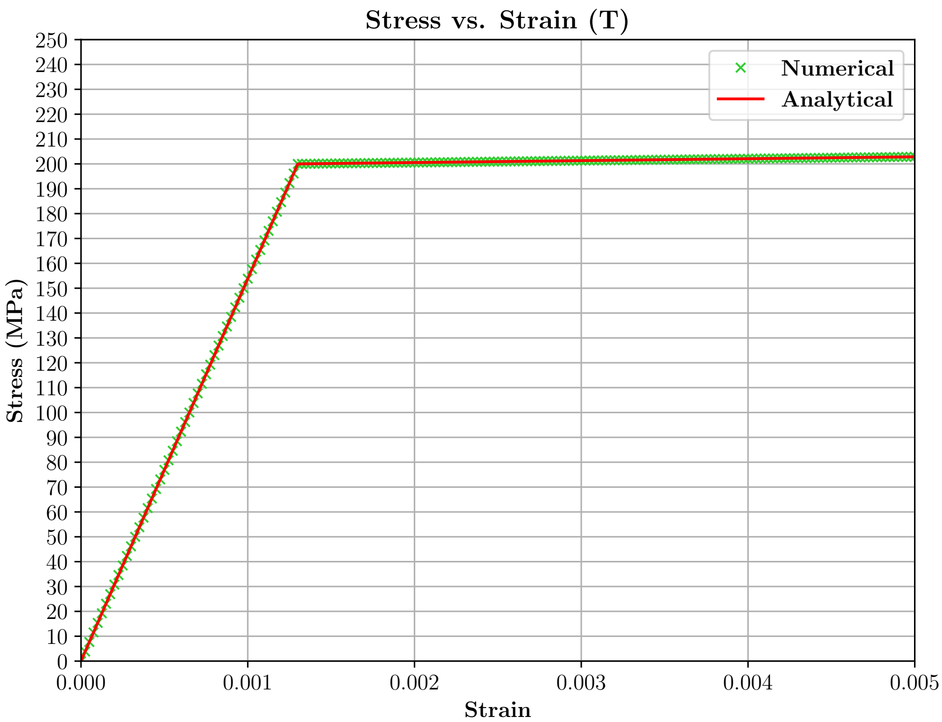


Figure 19: Plane shear stress-strain curve for T yield criteria with LIH

Unlike the uniaxial stress problem the yield point for the VM and T yield criteria are the not the same. This was correctly captured by the model which can be seen in Figures [18] and [19].

Reverse loading for the plane shear problem was also investigated. The numerical results are presented below.

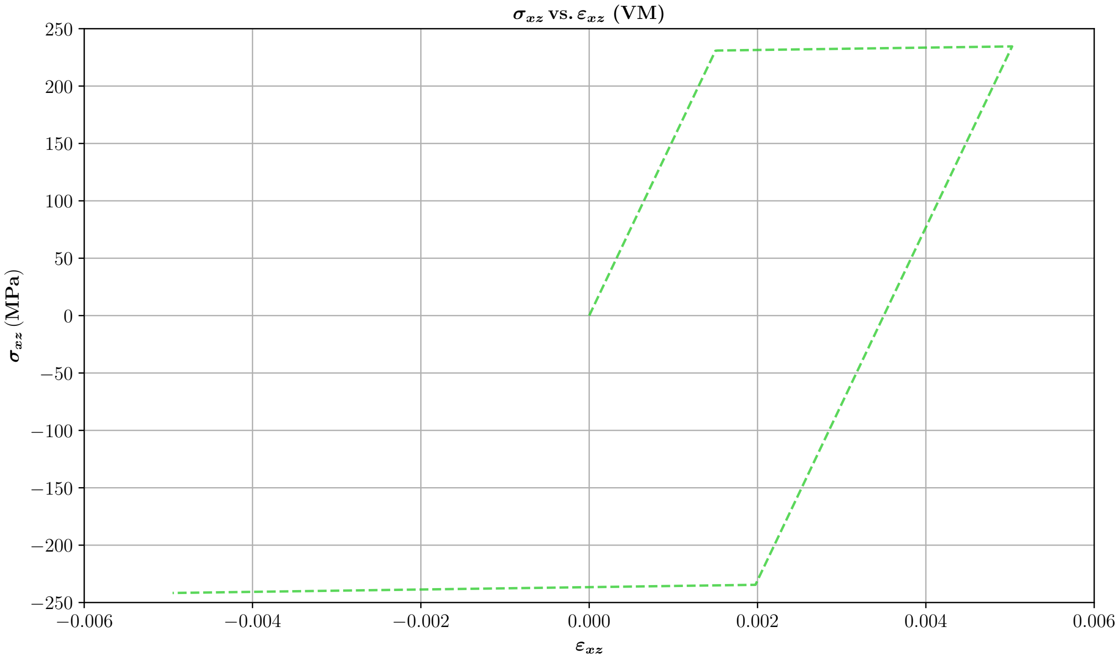


Figure 20: Plane shear reversed-loading stress-strain curve for VM yield criteria with LIH

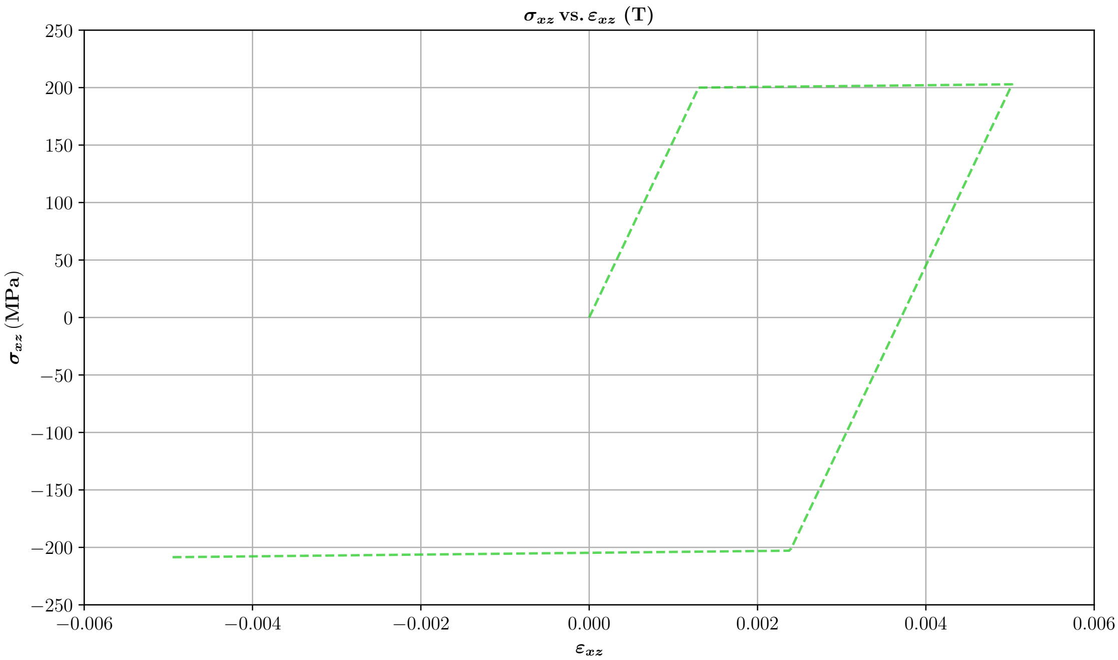


Figure 21: Plane shear reversed-loading stress-strain curve for T yield criteria with LIH

The model produced the desired shear stress evolution with respect to strain for the reversed loading plane shear problem which can be seen in Figures [20] and [21].

Below is a comparison between the analytical and numerical accumulated plastic strain for the VM and T yield criteria.

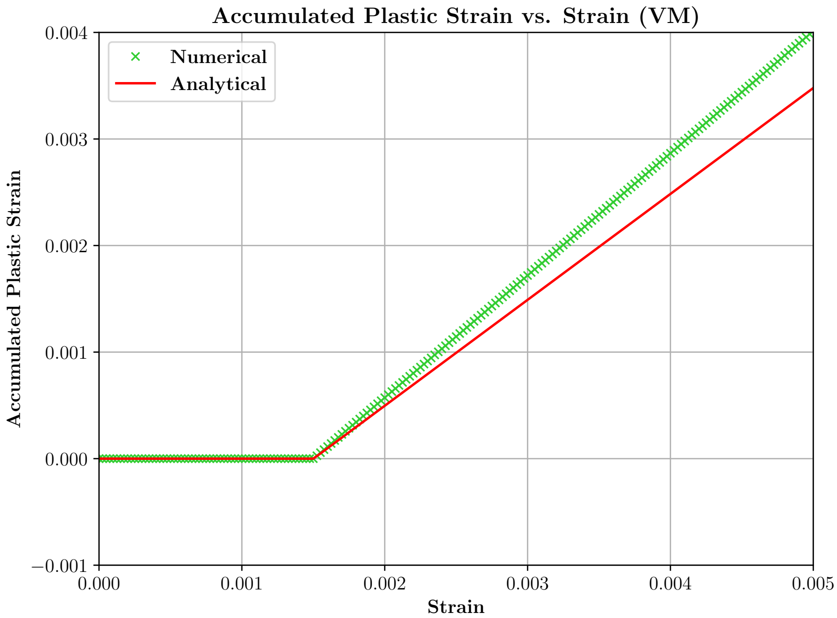


Figure 22: Plane shear accumulated plastic strain vs. strain curve for VM yield criteria with LIH

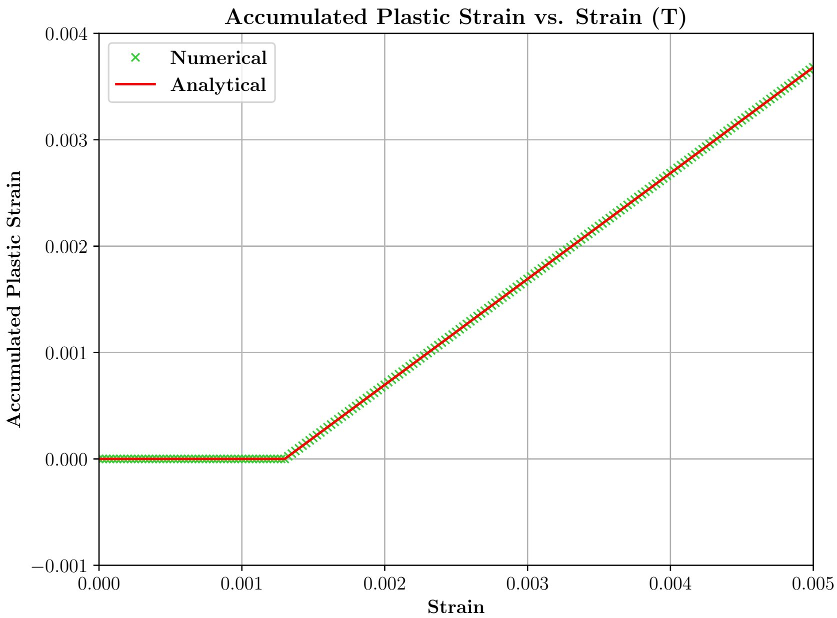


Figure 23: Plane shear accumulated plastic strain vs. strain curve for T yield criteria with LIH

The following are the PP plane shear numerical results. They were achieved by setting H equal to a small value. The model encountered convergence problems when $H = 0$.

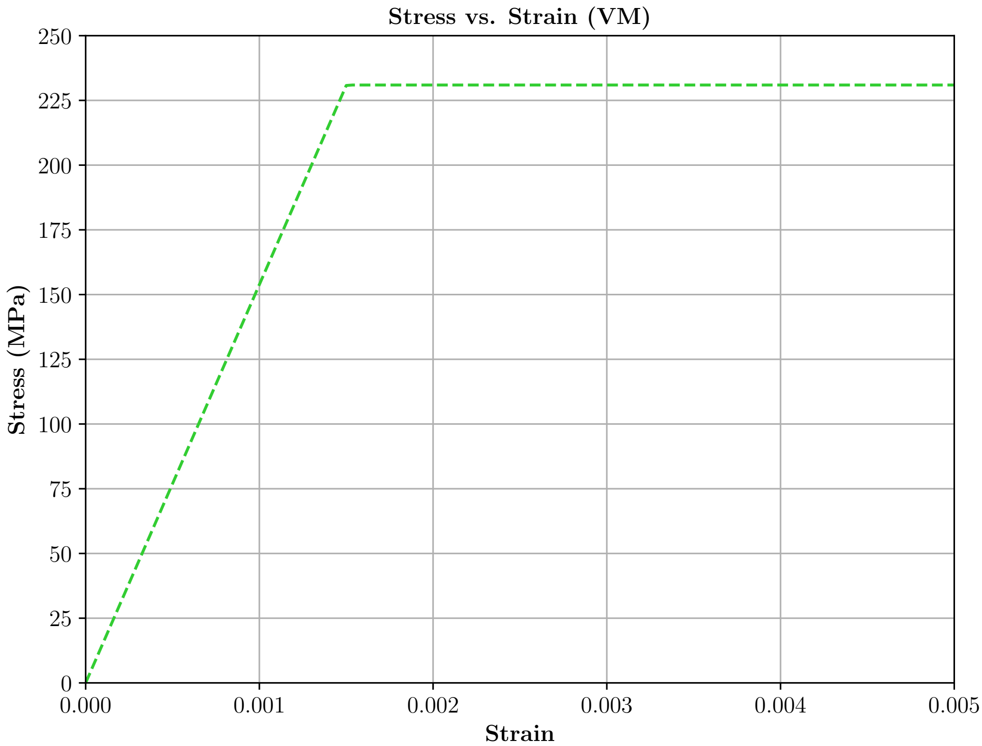


Figure 24: Plane shear stress-strain curve for VM yield criteria with PP

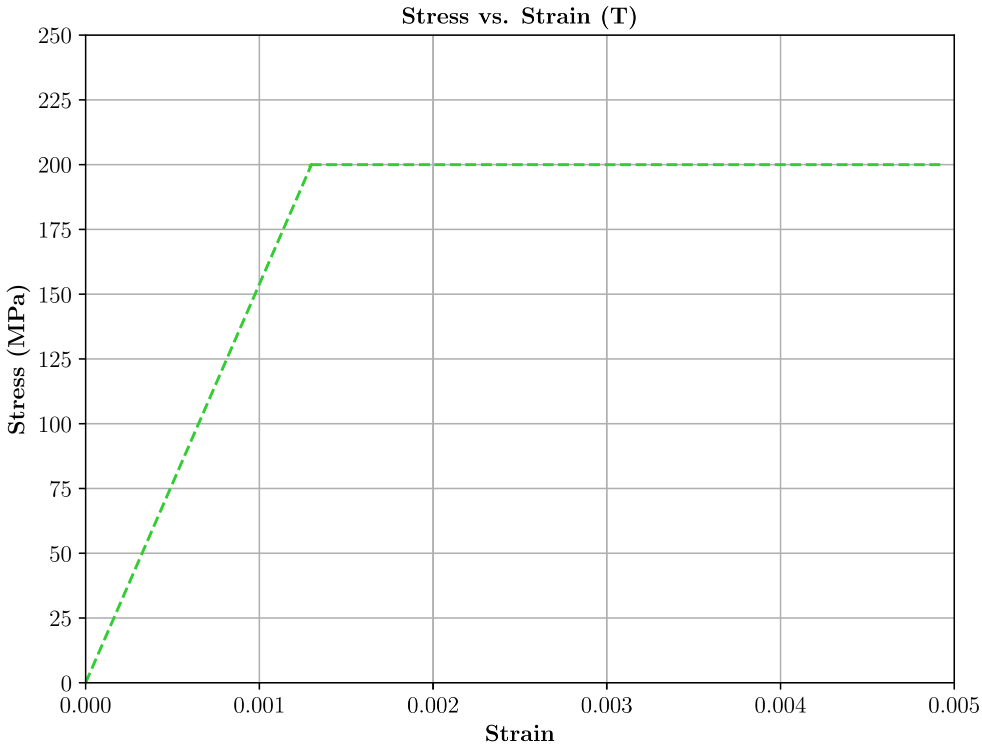


Figure 25: Plane shear stress-strain curve for T yield criteria with PP

Additionally, the reversed loading with PP plane shear problem was analysed.

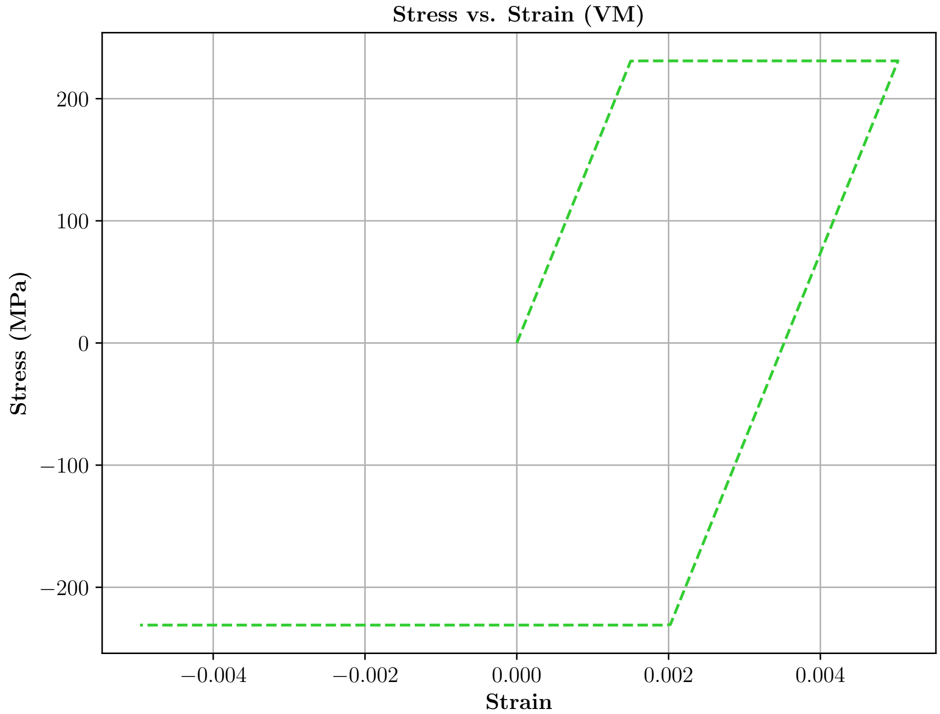


Figure 26: Plane shear reversed-loading stress-strain curve for VM yield criteria with PP

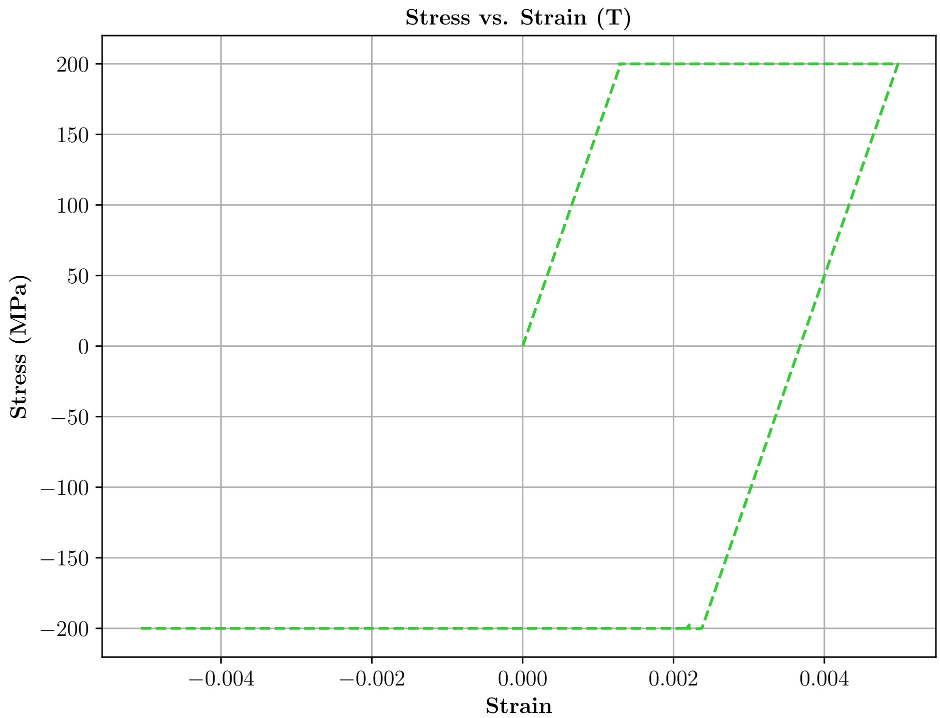


Figure 27: Plane shear reversed-loading stress-strain curve for T yield criteria with PP

4.3.3 Tension and Torsion of a Thin-Walled Cylinder

The analytical soutions, which were derived in Appendix [7.2], made some assumptions therefore the numerical results were expected to differ slightly. Stress values and accumulated plastic strain were recorded at the outer radius on the face which was at $z = 1$ (in the reference configuration), which can be seen in Figure [28], for both yield criteria.

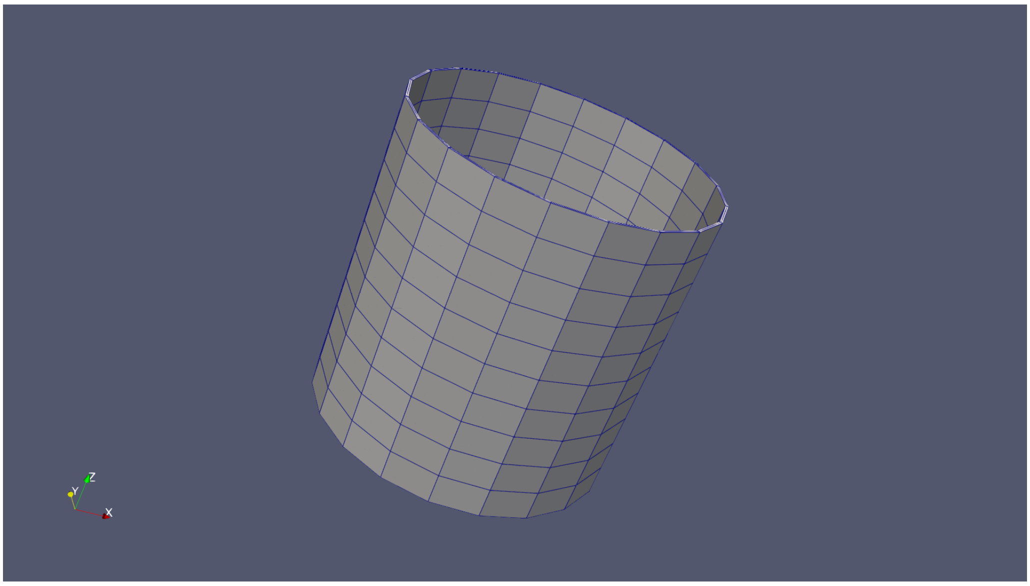


Figure 28: Thin-walled cylinder domain in the reference configuration

The results were obtained by varying the applied torque. The normal force was calculated, as a function of the applied torque, to ensure that the normal stress and shear stress are as equal as possible, throughout the loading, because the analytical solution was computed under the assumption that $\sigma_{\theta z} = \sigma_{zz}$.

T	$\sigma_{\theta z}$	σ_{zz}	γ
4	254.6	254.6	0.07051
5	318.3	318.3	0.1527
6	382.0	382.0	0.2348
7	445.6	445.6	0.3169
8	509.3	509.3	0.3991
9	573.0	573.0	0.4812
10	636.6	636.6	0.5634

Table 1: Analytical solution results for VM yield-criteria with LIH

T	$\sigma_{\theta z}$	σ_{zz}	γ
4	260.9	255.7	0.07470
5	326.5	318.5	0.1578
6	392.1	380.3	0.2408
7	457.7	441.4	0.3235
8	523.3	502.0	0.4062
9	588.9	562.3	0.4888
10	654.5	622.3	0.5713

Table 2: Numerical solution for the VM yield-criteria with LIH

In the numerical results the normal stress (σ_{zz}) and shear stress ($\sigma_{\theta z}$) values were close in value but not equal. The stresses (which were used in the analytical solution) were calculated without the thin-walled assumption.

	γ	
T	Analytical	Numerical
4	0.07051	0.07470
5	0.1527	0.1578
6	0.2348	0.2408
7	0.3169	0.3235
8	0.3991	0.4062
9	0.4812	0.4888
10	0.5634	0.5713

Table 3: Analytical and numerical accumulated plastic strain for the VM yield criteria with LIH

The RMSE of the analytically computed and numerically solved accumulated plastic strains in Table [3] is 0.65%.

The following graphs were plotted with the data of a nodal point at $z = 1$ (in the reference configuration) with $T = 5$.

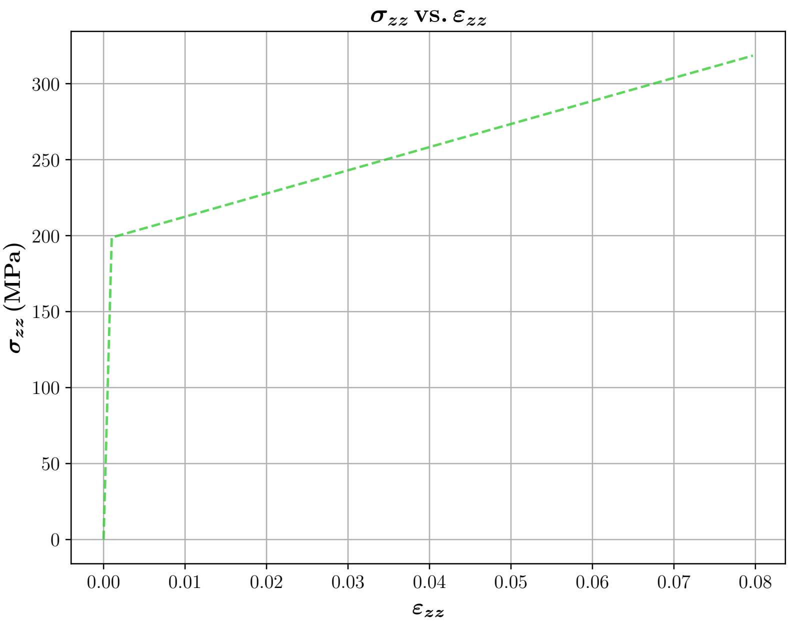


Figure 29: Normal stress-strain curve for VM yield criteria with LIH

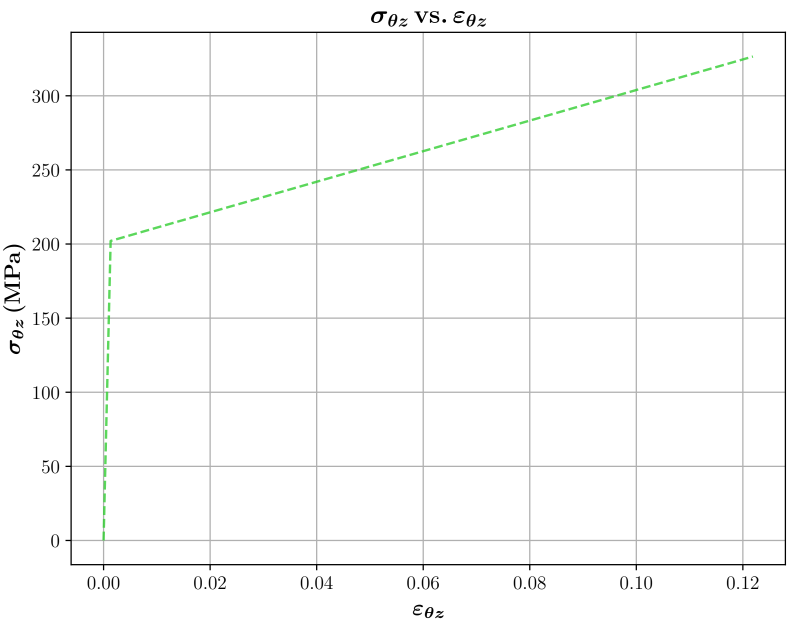


Figure 30: Shear stress-strain curve for VM yield criteria with LIH

Looking at Figures [29] and [30], yielding starts at approximately 200 MPa which is equal to the analytically derived yield point. Additionally, the normal stress and shear stress are equal throughout the loading; which is expected. Lastly, the material is hardening linearly. This supports the argument that the plasticity model is generating accurate results.

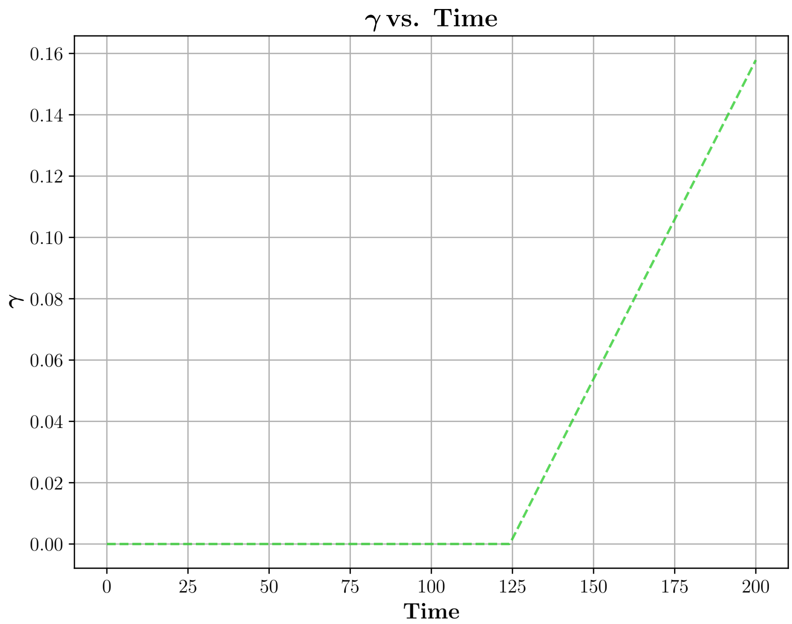


Figure 31: Plastic strain curve for VM yield criteria with LIH

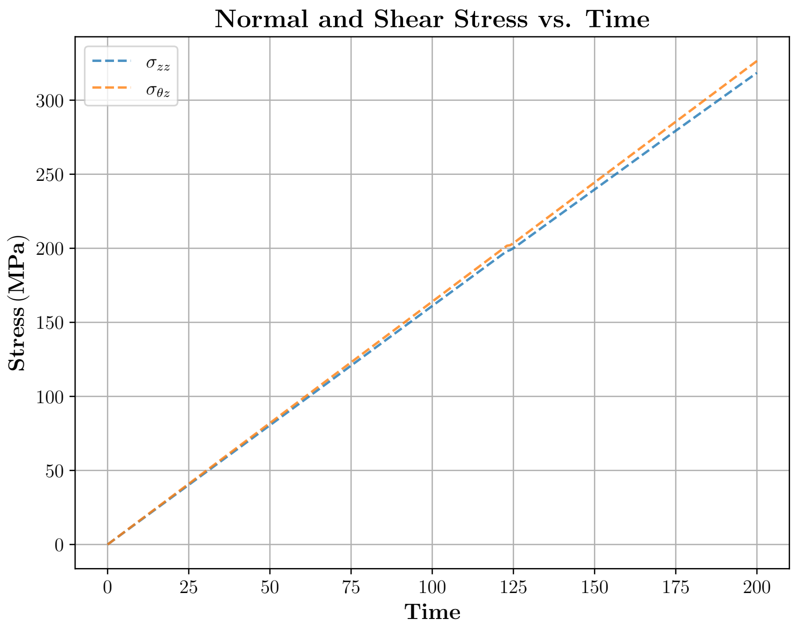


Figure 32: Normal and shear stress vs. time

Figures [31] and [32] shows that the accumulated plastic strain remains at 0 until time 125. At this time the normal and shear stress have values of 200 MPa; which is the stress at which yielding is expected to start.

5 Conclusion

5.1 Recap of Objectives

The main objectives of the project were as follows:

1. Develop an infinitesimal strain plasticity model in C++ using `deal.ii`.
2. Implement VM and T yield-criteria.
3. Incorporate Perfect Plasticity PP and LIH.
4. Apply constant and linearly varying loads (displacement and force).
5. Analyse uniaxial stress, plane shear, and tension and torsion of a thin-walled cylinder problems.

5.2 Assessment of Objectives

1. **Infinitesimal Strain Plasticity Model:** The project demonstrates the feasibility of implementing classical plasticity in the small strain regime using `deal.ii`.
2. **Yield Criteria:**
   - **Von-Mises:** The VM yield-criterion was successfully implemented for all test cases. The model produced accurate stress, strain, and displacement results within the small strain range.
   - **Tresca:** The Tresca yield-criterion produced inconsistent results in some test cases. For example, as shown in Figures [10] and [15], the stress response for the uniaxial stress problem was accurate, but the displacement response was not.
3. **Hardening Models:** LIH was implemented successfully. PP results were achieved by setting H to 0 or a small value (e.g, 10^{-6} MPa).
4. **Load Applications:** Incremental load application was employed to improve convergence. Both constant and linearly varying loads were implemented successfully. Desired results were obtained for reverse loading in uniaxial stress and plane shear problems.
5. **Test Cases:** The uniaxial stress, plane shear, and tension and torsion of a thin-walled cylinder test cases were run successfully for the VM yield-criterion, producing accurate results. However, the tension and torsion test encountered convergence issues with the Tresca yield-criterion, making results unattainable for that case.

6 References

[1] Eduardo A de Souza Neto, Djordje Peric, and David RJ Owen. *Computational methods for plasticity: theory and applications.* John Wiley & Sons, 2011.

[2] Jörg Frohne, Timo Heister, and Wolfgang Bangerth. Efficient numerical methods for the large-scale, parallel solution of elastoplastic contact problems. *International Journal for Numerical Methods in Engineering*, 105(6):416–439, 2016.

7 Appendix

7.1 Accumulated Plastic Strain of Plane Shear Problem

7.1.1 Von-Mises

$$\boldsymbol{s} = \boldsymbol{\sigma} - \frac{1}{3}\mathrm{tr}(\boldsymbol{\sigma})\mathrm{I} \tag{61}$$

$$J_2 = \frac{1}{2}\mathrm{tr}(\boldsymbol{s^2}) \tag{62}$$

Using Equations (61) and (62) the VM yield function, with LIH, reduces to

$$f_{VM}(\sigma, q) = \tau^2 - \frac{(\sigma_0 + q)^2}{3} \leq 0. \tag{63}$$

Using Equation (63) and the fact that $f = 0$, during yielding, it can be shown that

$$\tau = \frac{\sigma_0}{\sqrt{3}}. \tag{64}$$

Using the following equations:

$$df = \frac{\partial f}{\partial \sigma} d\sigma + \frac{\partial f}{\partial q} dq = 0 \tag{65}$$

$$q = H\gamma \tag{66}$$

$$d\gamma = |d\varepsilon_p| \tag{67}$$

$$\varepsilon_p = \xi \frac{\partial f}{\partial \sigma} \tag{68}$$

$$d\varepsilon = d\varepsilon_e + d\varepsilon_p \tag{69}$$

the tangent modulus was computed as

$$E_{\tan} = \frac{2\sqrt{3}\mu H}{\sqrt{3}H + 6\mu}. \tag{70}$$

Additionally, the plastic strain can be computed with

$$\gamma = \int_{\frac{\sigma_0}{\sqrt{3}}}^{\sigma_f} \frac{3}{\sqrt{3}H} d\sigma. \tag{71}$$

7.1.2 Tresca

$$\sigma_1 = \tau \tag{72}$$

$$\sigma_2 = -\tau \tag{73}$$

Using Equations (72) and (73) the T yield function reduces to

$$f_T(\sigma, q) = 2\tau - (\sigma_0 + q) \leq 0. \tag{74}$$

Using Equation (74) and the fact that $f = 0$, during yielding, it can be shown that

$$\tau = \frac{\sigma_0}{2}. \tag{75}$$

Using the following equations:

$$df = \frac{\partial f}{\partial \sigma}\, d\sigma + \frac{\partial f}{\partial q}\, dq = 0 \tag{76}$$

$$q = H\gamma \tag{77}$$

$$d\gamma = |d\varepsilon_p| \tag{78}$$

$$\varepsilon_p = \xi \frac{\partial f}{\partial \sigma} \tag{79}$$

$$d\varepsilon = d\varepsilon_e + d\varepsilon_p \tag{80}$$

the tangent modulus was computed as

$$E_{\tan} = \frac{2\mu H}{H + 4\mu}. \tag{81}$$

Additionally, the plastic strain can be computed with

$$\gamma = \int_{\sigma_0/2}^{\sigma_f} \frac{2}{H}\, d\sigma. \tag{82}$$

7.2 Accumulated Plastic Strain of Thin-Walled Cylinder Under Tension and Torsion

$$df = \frac{\partial f}{\partial \sigma}\, d\sigma + \frac{\partial f}{\partial q}\, dq = 0 \quad \text{(consistency condition)}$$

$$df = 2\sigma d\sigma + 2a\sigma d\sigma - 2(\sigma_y + q)\, dq = 0$$

$$f = 0 \quad \therefore \quad 0 = \sigma^2 + a\sigma^2 - (\sigma_y + q)^2 \quad \text{(yielding)}$$

$$\sigma_y + q = 2\sigma$$

$$dq = \xi H \left| \frac{\partial f}{\partial \sigma} \right| \quad \text{(isotropic hardening)}$$

$$\frac{\partial f}{\partial \sigma} = (2\sigma, 2a\sigma)$$

$$\left| \frac{\partial f}{\partial \sigma} \right| = 2\sigma\sqrt{1 + a^2}$$

$$df = 2\sigma d\sigma + 2a\sigma d\sigma - 2(\sigma_y + q) dq = 0$$

$$\sigma_y + q = 2\sigma$$

$$dq = \frac{1+a}{2} d\sigma$$

$$dq = \xi H \left| \frac{\partial f}{\partial \sigma} \right|$$

$$\xi = \frac{1+a}{4\sigma H\sqrt{1+a^2}} d\sigma$$

$$d\varepsilon^p_{zz} = \xi \frac{\partial f}{\partial \sigma_{zz}} = \frac{1+a}{4\sigma H\sqrt{1+a^2}} d\sigma \times 2\sigma_{zz} = \frac{1+a}{2H\sqrt{1+a^2}} d\sigma$$

$$\varepsilon^p_{zz} = \int_{\frac{\sigma_0}{\sqrt{1+a}}}^{\sigma_f} \frac{1+a}{2H\sqrt{1+a^2}} d\sigma = \frac{1+a}{2H\sqrt{1+a^2}} \left[\sigma\right]_{\frac{\sigma_0}{\sqrt{1+a}}}^{\sigma_f}$$

$$d\varepsilon^p_{z\theta} = \xi \frac{\partial f}{\partial \sigma_{z\theta}} = \frac{1+a}{4\sigma H\sqrt{1+a^2}} d\sigma \times 2a\sigma_{z\theta} = \frac{a(1+a)}{2H\sqrt{1+a^2}} d\sigma$$

$$\varepsilon^p_{z\theta} = \int_{\frac{\sigma_0}{\sqrt{a(1+a)}}}^{\sigma_f} \frac{1+a}{2H\sqrt{1+a^2}} d\sigma = \frac{a(1+a)}{2H\sqrt{1+a^2}} \left[\sigma\right]_{\frac{\sigma_0}{\sqrt{1+a}}}^{\sigma_f}$$

7.3 Parameter Files for Numerical Results

This section contains the .prm files needed to generate the numerical results presented in this report. The corresponding figure and/or table number/s is/are above the .prm file. Text in red is not part of the .prm file.

7.3.1 Uniaxial Stress

Figure [10] and Figure [14]:

```
# problem parameters
set output directory                    =   Von_Mises
set base mesh                           =   box
set problem                             =   tensile
set loading                             =   no reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1550.0
set yield criteria                      =   Von-Mises

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   1
set number of time-steps                =   200
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [11] and Figure [15]:

```
# problem parameters
set output directory                    =   Tresca
set base mesh                           =   box
set problem                             =   tensile
set loading                             =   no reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1550.0
set yield criteria                      =   Tresca

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   2
set number of time-steps                =   200
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [12]:

```
# problem parameters
set output directory                    =   Von_Mises
set base mesh                           =   box
set problem                             =   tensile
set loading                             =   reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1550.0
set yield criteria                      =   Von-Mises

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   1
set number of time-steps                =   600
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [13]:

```
# problem parameters
set output directory                    =   Tresca
set base mesh                           =   box
set problem                             =   tensile
set loading                             =   reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1550.0
set yield criteria                      =   Tresca

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   2
set number of time-steps                =   600
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [16]:

```
# problem parameters
set output directory                     =   Von_Mises
set base mesh                            =   box
set problem                              =   tensile
set loading                              =   no reverse
set applied displacement                 =   0.01
set isotropic hardening parameter        =   0.0
set yield criteria                       =   Von-Mises

# solver parameters
set polynomial degree                    =   1
set number of initial refinements        =   1
set number of time-steps                 =   200
set newton loop tolerance                =   1.0e-5
set newton loop maximum iterations       =   250
```

Figure [17]:

```
# problem parameters
set output directory                     =   Von_Mises
set base mesh                            =   box
set problem                              =   tensile
set loading                              =   reverse
set applied displacement                 =   0.01
set isotropic hardening parameter        =   0.0
set yield criteria                       =   Von-Mises

# solver parameters
set polynomial degree                    =   1
set number of initial refinements        =   1
set number of time-steps                 =   600
set newton loop tolerance                =   1.0e-5
set newton loop maximum iterations       =   250
```

7.3.2 Plane Shear

Figure [18] and Figure [22]:

```
# problem parameters
set output directory                    =   Von_Mises
set base mesh                           =   box
set problem                             =   shear
set loading                             =   no reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1550.0
set yield criteria                      =   Von-Mises

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   1
set number of time-steps                =   200
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [19] and Figure [23]:

```
# problem parameters
set output directory                    =   Tresca
set base mesh                           =   box
set problem                             =   shear
set loading                             =   no reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1550.0
set yield criteria                      =   Tresca

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   1
set number of time-steps                =   200
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [20]:

```
# problem parameters
set output directory                        =   Von_Mises
set base mesh                               =   box
set problem                                 =   shear
set loading                                 =   reverse
set applied displacement                    =   0.01
set isotropic hardening parameter           =   1550.0
set yield criteria                          =   Von-Mises

# solver parameters
set polynomial degree                       =   1
set number of initial refinements           =   1
set number of time-steps                    =   600
set newton loop tolerance                   =   1.0e-5
set newton loop maximum iterations          =   250
```

Figure [21]:

```
# problem parameters
set output directory                        =   Tresca
set base mesh                               =   box
set problem                                 =   shear
set loading                                 =   reverse
set applied displacement                    =   0.01
set isotropic hardening parameter           =   1550.0
set yield criteria                          =   Tresca

# solver parameters
set polynomial degree                       =   1
set number of initial refinements           =   1
set number of time-steps                    =   600
set newton loop tolerance                   =   1.0e-5
set newton loop maximum iterations          =   250
```

Figure [24]:

```
# problem parameters
set output directory                     =   Von_Mises
set base mesh                            =   box
set problem                              =   shear
set loading                              =   no reverse
set applied displacement                 =   0.01
set isotropic hardening parameter        =   1.0e-8
set yield criteria                       =   Von-Mises

# solver parameters
set polynomial degree                    =   1
set number of initial refinements        =   1
set number of time-steps                 =   200
set newton loop tolerance                =   1.0e-5
set newton loop maximum iterations       =   250
```

Figure [25]:

```
# problem parameters
set output directory                     =   Tresca
set base mesh                            =   box
set problem                              =   shear
set loading                              =   no reverse
set applied displacement                 =   0.01
set isotropic hardening parameter        =   1.0e-8
set yield criteria                       =   Tresca

# solver parameters
set polynomial degree                    =   1
set number of initial refinements        =   1
set number of time-steps                 =   200
set newton loop tolerance                =   1.0e-5
set newton loop maximum iterations       =   250
```

Figure [26]:

```
# problem parameters
set output directory                    =   Von_Mises
set base mesh                           =   box
set problem                             =   shear
set loading                             =   reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1.0e-8
set yield criteria                      =   Von-Mises

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   1
set number of time-steps                =   600
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

Figure [27]:

```
# problem parameters
set output directory                    =   Tresca
set base mesh                           =   box
set problem                             =   shear
set loading                             =   reverse
set applied displacement                =   0.01
set isotropic hardening parameter       =   1.0e-8
set yield criteria                      =   Tresca

# solver parameters
set polynomial degree                   =   1
set number of initial refinements       =   1
set number of time-steps                =   600
set newton loop tolerance               =   1.0e-5
set newton loop maximum iterations      =   250
```

7.3.3 Tension and Torsion of a Thin-Walled Cylinder

Table [2] and Table [3]:

```
# problem parameters
set output directory                      =   Von_Mises
set base mesh                             =   cylinder
set problem                               =   torsion
set isotropic hardening parameter         =   1550.0
set yield criteria                        =   Von-Mises

# solver parameters
set polynomial degree                     =   1
set number of initial refinements         =   1
set number of time-steps                  =   200
set newton loop tolerance                 =   1.0e-5
set newton loop maximum iterations        =   250

# parameters for 'torsion' problem
set outer radius                          =   0.5
set inner radius                          =   0.49
set torque                                =   4.0            (4.0 - 10.0)
```

Figure [29], Figure [30], Figure [31] and Figure [32]:

```
# problem parameters
set output directory                      =   Von_Mises
set base mesh                             =   cylinder
set problem                               =   torsion
set isotropic hardening parameter         =   1550.0
set yield criteria                        =   Von-Mises

# solver parameters
set polynomial degree                     =   1
set number of initial refinements         =   1
set number of time-steps                  =   200
set newton loop tolerance                 =   1.0e-5
set newton loop maximum iterations        =   250

# parameters for 'torsion' problem
set outer radius                          =   0.5
set inner radius                          =   0.49
set torque                                =   5.0
```