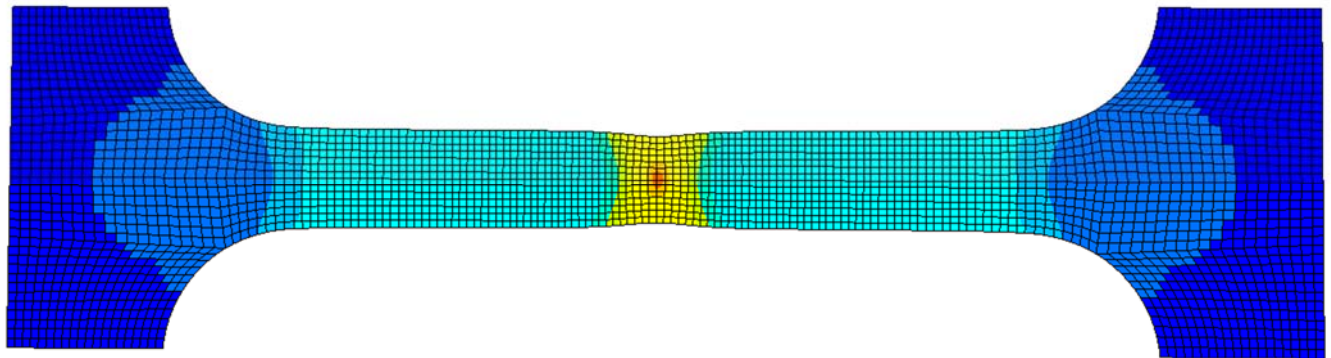


## JRC TECHNICAL REPORTS

# Formulation and Implementation of the VPJC Material Model in EUROPLEXUS

Vegard Aune  
Folco Casadei  
Georgios Valsamos  
Tore Borvik  
2016

MAX: +9.65E-01  
---: +1.00E+00  
---: +9.23E-01  
---: +8.46E-01  
---: +7.69E-01  
---: +6.92E-01  
---: +6.15E-01  
---: +5.38E-01  
---: +4.61E-01  
---: +3.85E-01  
---: +3.08E-01  
---: +2.31E-01  
---: +1.54E-01  
---: +7.69E-02  
---: +0.00E+00  
MIN: +0.00E+00  
ECRO 11 [N/A]





Formulation and Implementation of  
the VPJC Material Model in  
EUROPLEXUS

This publication is a Technical report by the Joint Research Centre, the European Commission's in-house science service. It aims to provide evidence-based scientific support to the European policy-making process. The scientific output expressed does not imply a policy position of the European Commission. Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use which might be made of this publication.

**JRC Science Hub**

<https://ec.europa.eu/jrc>

JRC102336

EUR 27982 EN

ISBN 978-92-79-59746-6

ISSN 1831-9424

doi:10.2788/609529

© European Union, 2016

Reproduction is authorised provided the source is acknowledged.

Printed in Italy

All images © European Union 2016

How to cite: Authors; title; EUR; doi

# Formulation and Implementation of the VPJC Material Model in EUROPLEXUS

Vegard Aune<sup>a,b,\*</sup>, Folco Casadei<sup>c</sup>, Georgios Valsamos<sup>c</sup>, Tore Børvik<sup>a,b</sup>

<sup>a</sup>*Centre for Advanced Structural Analysis (CASA), NTNU, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway*

<sup>b</sup>*Structural Impact Laboratory (SIMLab), Department of Structural Engineering, NTNU, NO-7491 Trondheim, Norway*

<sup>c</sup>*European Commission, Joint Research Centre (JRC), Institute for the Protection and Security of the Citizen (IPSC), European Laboratory for Structural Assessment (ELSA), 21027 Ispra, Italy*

---

## Abstract

A constitutive model of elastic-thermoviscoplasticity and ductile failure is implemented in EUROPLEXUS (EPX) using a fully vectorized version of the forward Euler integration algorithm and a special case of the backward Euler algorithm. The material model is based on the constitutive model originally proposed by Johnson and Cook, and the ductile failure criterion proposed by Cockcroft and Latham. Constitutive equations are formulated in terms of linear elasticity, von Mises yield criterion, associated flow rule, nonlinear isotropic strain hardening, strain-rate hardening, temperature softening due to adiabatic heating, and ductile failure. The performance of the implementation is verified and validated in EPX. First, a verification is performed using single element simulations in uniaxial tension and simple shear. Then, the model is validated against experimental data from quasi-static tension tests and airblast experiments. The model is applicable for one-, two- and three-dimensional stress analysis, and coupled with element deletion options available in EPX.

## Keywords:

Elastic-thermoviscoplasticity, Isotropic hardening, von Mises yield criterion, Adiabatic thermal softening, Cutting plane return map algorithm, Radial return map algorithm, VPJC, EUROPLEXUS

---

## 1. Introduction

In metal plasticity, extreme loading conditions in fast transient dynamics often involve large strains, high strain rates, temperature softening and ductile failure. A widely used design tool for this class of problems is the non-linear finite element (FE) method. In the FE method, the formulation and numerical solution of non-linear problems in continuum mechanics relies on the weak form of the momentum balance equation (also known as the principal of virtual work [1]). The integral form of the principal of virtual work is well suited for direct application to the FE method, and by spatial discretization the solution is integrated in time. In the case of fast transient dynamics, it is often used an explicit time integration based on the central difference method. Iterations for FE analyses involving non-linear material behaviour can in general be divided into two levels, hereby denoted the global and local level. The global level involves the explicit establishment of global equilibrium between internal stresses and external loads, and the local level updates the corresponding stress state in each FE integration point (for a given strain increment) in terms of the governing constitutive equations. Constitutive equations (also known as material models) are mathematical descriptions of material behaviour which gives the stress as a function of the deformation history of the body

---

\*Corresponding author

Email address: [vegard.aune@ntnu.no](mailto:vegard.aune@ntnu.no) (Vegard Aune)

[1]. This implies that the local integration of the constitutive equations controls the accuracy and stability of the global equilibrium, and that the overall solution in the FE model is considerably influenced by the accuracy, robustness and effectiveness of the integration algorithm.

A commonly used scheme for the local stress integration is the predictor-corrector method (often called return mapping). The basis for this return mapping is two successive steps, i.e., the predictor step and the corrector step. The predictor step is used to estimate a trial stress state, while the corrector step applies a flow rule by using a return mapping scheme to ensure the consistency condition. That is, ensuring that the stress state is on the yield surface. The present study uses the explicit cutting plane method proposed by Ortiz and Simo [2] and the implicit closest point projection method proposed by Simo and Taylor [3]. During these approaches an elastic predictor uses the total strain increment and the linear elastic relation to obtain a trial stress state. Then, if the elastic trial stress is outside the plastic domain, the elastically predicted stresses are corrected to return to the updated yield surface by iterations of the plastic corrector. The basis for the cutting plane method is to utilize the known stress state at the last converged time increment to determine the normal to the yield surface, while the implicit closest point projection method utilizes the cylindrical shape of the yield surface and the deviatoric stress space to scale the trial state to fit a suitably updated yield surface by iterations of the plastic corrector [4, 5].

This work considers a well-known constitutive relation proposed by Johnson and Cook [6] and von Mises plasticity is used to update the stresses. A model including elastic-thermoviscoplasticity and ductile failure is implemented as a new material called *VPJC* in EUROPLEXUS (EPX) [7]. The implementation uses a fully vectorized version of the forward Euler integration algorithm and a special case of the backward Euler algorithm. Large deformations are accounted for by using a hypoelastic-plastic formulation in the constitutive equations and co-rotational formulations in EPX (i.e., the Jaumann stress rates [7]). This implies that the constitutive equations are defined in a co-rotational coordinate system in which the basis system rotates with the material. The subroutine interface has been implemented using the Cauchy stress components, and all stress and strain quantities are defined in terms of the rate of deformation tensor. The model is applicable for one- (1D), two- (2D) and three-dimensional (3D) stress analyses (i.e., for bar, shell, solid, axisymmetric and plane strain elements). The model is coupled with the element deletion options available in EPX by introducing a state variable controlling the element erosion. That is, as the failure variable reaches its critical value  $D_C$  the element is removed. The model is finally verified and validated through numerical simulations in EPX. This is first done by a verification using single element simulations in uniaxial tension and simple shear. Then, the model is validated against experimental data from quasi-static tension tests and airblast experiments.

## 2. Constitutive Equations

There are in general two categories within the plasticity theory, i.e. the mathematical theories and the physical theories [8]. Mathematical theories are purely phenomenological and aim to represent experimental observations through mathematical relations. Most of the common material models today are phenomenological. Physical theories try to describe the underlying mechanisms, and quantify the plastic deformation, at a microscopical level. A well-known and acknowledged physical theory is the crystal plasticity theory. This section considers the phenomenological-based formulation of the constitutive equations for the computational model used in this study, starting with the basic theory for infinitesimal strains and then expanding the theory to also include large deformations. The proposed computational model includes linear elasticity, large strains, high strain rates, temperature softening and ductile failure. Assuming negligible elastic strains compared to the plastic strains, an hypoelastic formulation may be adopted through an additive decomposition of the rate of deformation tensor. Since the main objective is the numerical implementation of thermo-viscoplasticity, the constitutive equations are restricted to linear isotropic elastic behavior. Thermodynamic coupling is not included, however, the temperature evolution is assumed to be adiabatic. Ductile failure is considered by using a simple one-parameter failure criterion suggested by Cockcroft and Latham

[9] which is based on plastic work per unit volume.

From a computational point of view, the plastic response is an incremental process which implies that the constitutive equations need to be formulated on rate form. Starting by assuming infinitesimal deformations, the strain rate tensor  $\dot{\boldsymbol{\epsilon}}$  may be decomposed by an additive split

$$\dot{\boldsymbol{\epsilon}} = \dot{\boldsymbol{\epsilon}}^e + \dot{\boldsymbol{\epsilon}}^p \quad (1)$$

where  $\dot{\boldsymbol{\epsilon}}^e$  is the elastic and  $\dot{\boldsymbol{\epsilon}}^p$  is the plastic strain rate tensor, respectively. Using the generalized Hooke's law, the rate of the stress tensor is defined as

$$\dot{\boldsymbol{\sigma}} = \mathbf{C} : \dot{\boldsymbol{\epsilon}}^e \quad (2)$$

where  $\mathbf{C}$  is the fourth order elasticity tensor given by

$$\mathbf{C} = \lambda_{el} \mathbf{I} \otimes \mathbf{I} + 2\mu_{el} \mathbf{I} \quad (3)$$

where  $\mathbf{I}$  is the second order identity tensor, and  $\lambda_{el}$  and  $\mu_{el}$  are the Lamé constants given as

$$\lambda_{el} = \frac{\nu E}{(1 + \nu)(1 - 2\nu)} \quad \text{and} \quad \mu_{el} = \frac{E}{2(1 + \nu)} \quad (4)$$

The elasticity tensor  $\mathbf{C}$  is assumed to be constant,  $E$  is the Young's modulus and  $\nu$  the Poisson's ratio.

In general, assuming only isotropic hardening, the yield criterion is defined as

$$f(\boldsymbol{\sigma}, R) = \varphi(\boldsymbol{\sigma}) - \sigma_y(\sigma_0, R) = \sigma_{eq} - (\sigma_0 + R) \quad (5)$$

where  $\varphi(\boldsymbol{\sigma}) = \sigma_{eq}$  is the equivalent stress,  $\sigma_0$  is the initial yield stress and  $R = R(p)$  is the isotropic hardening variable depending on the equivalent (or accumulated) plastic strain  $p$ . The evolution of the yield surface is therefore governed by the elastic domain, described by the flow stress  $\sigma_y$ . The function  $f$  is assumed to be a continuous function of the stress tensor  $\boldsymbol{\sigma}$ , and results in negative values when the material is in the elastic domain. In the case of rate-independent plasticity and isotropic hardening, it is assumed that  $f(\boldsymbol{\sigma}) > 0$  is inadmissible and, consequently, that plastic deformation occurs when  $f(\boldsymbol{\sigma}) = 0$ .

Furthermore,  $\varphi(\boldsymbol{\sigma})$  is assumed to be an isotropic and positive homogeneous function of order one of the stress tensor. This implies that

$$\varphi(a\boldsymbol{\sigma}) = a\varphi(\boldsymbol{\sigma}) \quad (6)$$

where  $a$  could be any positive scalar. Then, Euler's theorem for homogeneous functions gives [10]

$$\boldsymbol{\sigma} : \frac{\partial \varphi}{\partial \boldsymbol{\sigma}} = \varphi \quad (7)$$

The plastic deformation process is governed by an important physical restriction, i.e., that work has to be done to the body all the time for the deformation to continue. This restriction is expressed as

$$\mathfrak{D}_p = \boldsymbol{\sigma} \dot{\boldsymbol{\epsilon}}^p \geq 0 \quad (8)$$

where  $\mathfrak{D}_p$  expresses the plastic dissipation per unit volume, which is assumed to be dissipated as heat.

The restriction in Eq.(8) is fulfilled by defining the plastic strain rate tensor  $\dot{\boldsymbol{\epsilon}}^p$  in a way that ensures non-negative dissipation. This is done by the plastic flow rule, which in the general case is expressed as

$$\dot{\boldsymbol{\epsilon}}^p = \dot{\lambda} \frac{\partial g}{\partial \boldsymbol{\sigma}} \quad (9)$$

where  $g = g(\boldsymbol{\sigma}) \geq 0$  is the plastic potential function and  $\dot{\lambda} \geq 0$  is a scalar denoted the plastic multiplier.

Combining Eqs.(8) and (9) result in the following restriction on the plastic potential function

$$\boldsymbol{\sigma} : \frac{\partial g}{\partial \boldsymbol{\sigma}} \geq 0 \quad (10)$$

where it is assumed that  $g$  is a positive homogeneous function of order one. Then, by using the Euler's theorem for homogeneous functions, it is seen that this assumption fulfills the dissipation inequality in Eq.(8) as

$$\mathfrak{D}_p = \boldsymbol{\sigma} \dot{\boldsymbol{\varepsilon}}^p = \dot{\lambda} \boldsymbol{\sigma} : \frac{\partial g}{\partial \boldsymbol{\sigma}} = \dot{\lambda} g \geq 0 \quad (11)$$

By further assuming that the plastic potential function  $g$  is defined by the yield function  $f$ , Eq.(9) is expressed as

$$\dot{\boldsymbol{\varepsilon}}^p = \dot{\lambda} \frac{\partial f}{\partial \boldsymbol{\sigma}} \quad (12)$$

and it is seen that the plastic potential function is associated with the yield function  $f$ . Eq.(12) is therefore called the associated flow rule, and ensures non-negative dissipation through Eqs.(5), (7) and (8) as

$$\mathfrak{D}_p = \boldsymbol{\sigma} \dot{\boldsymbol{\varepsilon}}^p = \dot{\lambda} \boldsymbol{\sigma} : \frac{\partial f}{\partial \boldsymbol{\sigma}} = \dot{\lambda} \boldsymbol{\sigma} : \frac{\partial \varphi}{\partial \boldsymbol{\sigma}} = \dot{\lambda} \varphi = \dot{\lambda} \sigma_{eq} \geq 0 \quad (13)$$

This implies, that in the case of the associated flow rule, the equivalent plastic strain rate equals the plastic parameter

$$\dot{p} = \dot{\lambda} \quad (14)$$

where the equivalent plastic strain  $p$  is defined as the energy conjugate variable to the equivalent stress  $\sigma_{eq}$ . The plastic dissipation could then be expressed in terms of the equivalent stress and the plastic strain rate as

$$\mathfrak{D}_p = \boldsymbol{\sigma} \dot{\boldsymbol{\varepsilon}}^p = \sigma_{eq} \dot{p} \quad (15)$$

Hence, assuming the associated flow rule implies that the incremental plastic strain vector  $d\boldsymbol{\varepsilon}^p = \dot{\boldsymbol{\varepsilon}}^p dt$  is parallel to the gradient of the yield surface  $\nabla f$  at  $\boldsymbol{\sigma}$ , and therefore directed along the outward normal  $\mathbf{n}$  of the yield surface (see Figure 1). Moreover, the magnitude of  $\dot{\boldsymbol{\varepsilon}}^p$  is given by the plastic multiplier  $\dot{\lambda}$ .

An important remark to this argumentation is that the shape of the yield surface also determines the direction of the plastic flow, as well as the stress state at which yielding initiates.

The associated flow rule may also be derived from a stronger assumption than the dissipation inequality in Eq.(8), i.e., the principle of maximum plastic dissipation expressed as

$$(\boldsymbol{\sigma} - \boldsymbol{\sigma}^*) \dot{\boldsymbol{\varepsilon}}^p \geq 0 \quad (16)$$

where  $\boldsymbol{\sigma}$  is the stress state at which the plastic dissipation occurs, and  $\boldsymbol{\sigma}^*$  is an arbitrary stress state inside or on the yield surface  $f(\boldsymbol{\sigma}^*) \leq 0$ . As for the dissipation inequality this principle states the normality of plastic flow, and in addition it is seen that the yield surface must be convex [11]. Thus, the use of an associated flow rule ensures that the plastic strain increments  $d\boldsymbol{\varepsilon}^p$  are vectors perpendicular to the potential surface.

To distinguish between elastic loading/unloading and plastic loading, the Kuhn-Tucker conditions are used to determine when plastic flow may occur. These conditions determines the evolution of the yield surface and are given as [11]



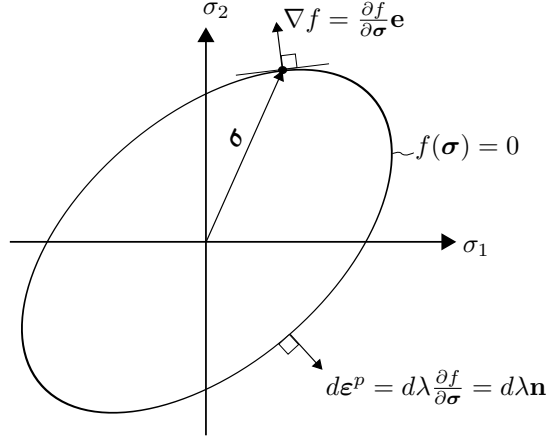


Figure 1: Geometric representation of the associated flow rule for an elliptic yield surface in plane stress space. The orthogonal unit vectors  $\mathbf{e}$  are pointing in the coordinate directions.

$$f \leq 0, \quad \dot{\lambda} \geq 0, \quad \dot{\lambda} f = 0 \quad (17)$$

where the first of these conditions indicates that the stress states must lie on or within the yield surface, the second indicates that the plastic multiplier is non-negative, and the last condition ensures that the stresses lie on the yield surface during plastic loading. From the Kuhn-Tucker conditions, the consistency condition can be stated as [10]

$$\dot{\lambda} \dot{f} = 0 \quad (18)$$

The consistency condition ensures that the stress must be located on the yield surface during continuous plastic flow, and is, in the theory of rate-independent plasticity, used to determine the plastic multiplier  $\dot{\lambda}$ .

To account for viscoplasticity and strain-rate sensitivity, the theory of plasticity has to be extended. The main difference is that in the viscoplastic domain the stress is allowed to move outside the yield surface, by assuming that only elastic strains are developed while the stress is inside or on the yield surface ( $f \leq 0$ ) [11]. This means that plastic strains are assumed to evolve only when the stresses move outside the yield surface (Figure 2b). By using an additive constitutive relation, this implies that the yield function  $f$  is equal to the viscous stress  $\sigma_v(\dot{p})$ , i.e.,

$$f(\boldsymbol{\sigma}) = \sigma_v(\dot{p}) > 0 \quad \iff \quad f_v(\boldsymbol{\sigma}) = \varphi(\boldsymbol{\sigma}) - \sigma_y(\sigma_0, R) - \sigma_v(\dot{p}) = 0 \quad (19)$$

From Eq.(19) it is seen that the viscous stress controls the strain-rate sensitivity of the material (Figure 2). Although the additive formulation is illustrative for the viscoplastic material behavior, a multiplicative formulation may be a preferred choice in some applications. In general, the multiplicative formulation reads

$$f_v(\boldsymbol{\sigma}) = \varphi(\boldsymbol{\sigma}) - \sigma_y(\sigma_0, R)v_p(\dot{p}) = 0 \quad \iff \quad f(\boldsymbol{\sigma}) = \sigma_v(p, \dot{p}) = (v_p - 1)\sigma_y > 0 \quad (20)$$

where  $v_p(\dot{p})$  is determined by the constitutive relation. It should be noted that in both Eq.(19) and Eq.(20) the viscoplastic constraint  $f_v(\boldsymbol{\sigma}) = 0$  is introduced, which allows for an expansion of the elastic domain due to the rate sensitivity of the material. The rate sensitivity of the material is controlled by the constitutive relation for the plastic multiplier, generally expressed as

$$\dot{p} = \dot{\lambda} = \begin{cases} 0 & \text{for } f \leq 0 \\ \phi(\boldsymbol{\sigma}, R) & \text{for } f > 0 \end{cases} \quad (21)$$

where  $\phi(\boldsymbol{\sigma}, R)$  depends on the state of the material. From Eq.(21) it is seen that the consistency condition in Eq.(18) no longer holds in the case of a viscoplastic formulation, and that the plastic flow depends explicitly on time during the evolution of  $\lambda$ .

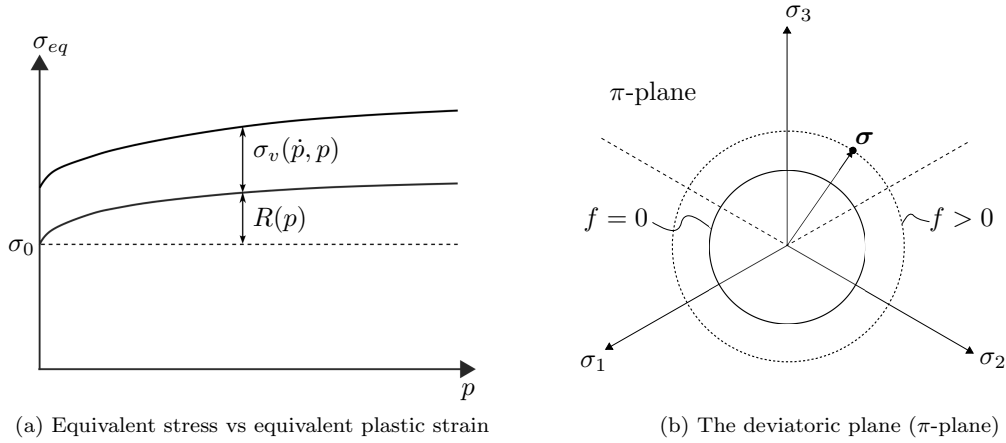


Figure 2: Viscoplastic effects due to strain-rate sensitivity.

To account for large displacement and large strain, the constitutive model is defined in a co-rotational coordinate system in which the basis system rotates with the material [7]. Typically, there are two different approaches of dealing with finite deformations, i.e., the hypoelastic-plastic formulations and hyperelastic-plastic formulations. In contrast to the hyperelastic formulation of finite plasticity models, hypoelastic theories do not rely on the assumption of the existence of a strain energy potential to model the reversible behaviour. Moreover, in the particular case of metal plasticity, the elastic strains are negligible compared to the plastic strains and it is reasonable to adopt an hypoelastic formulation for large deformations. This is analogous to the additive decomposition of the strain-rate tensor  $\dot{\boldsymbol{\epsilon}}$  in Eq. (1) in the theory for infinitesimal deformations, and the rate-of-deformation tensor  $\mathbf{D}$  is decomposed additively as [1]

$$\mathbf{D} = \mathbf{D}^e + \mathbf{D}^p \quad (22)$$

where  $\mathbf{D}^e$  and  $\mathbf{D}^p$  are the elastic and plastic part, respectively. The rate-of-deformation tensor  $\mathbf{D}$  is also called velocity strain and is a rate measure of the deformation and stretching of the elements.

The starting point for hypoelastic models is the formulation of the constitutive equations in terms of objective stress rates [5], in which the hypoelastic relation in terms of the objective stress rate is expressed as

$$\hat{\boldsymbol{\sigma}}^\nabla = \mathbf{C} : \mathbf{D}^e = \lambda_{el} (tr \mathbf{D}^e) \mathbf{I} + 2\mu_{el} \mathbf{D}^e \quad (23)$$

where  $\boldsymbol{\sigma}$  is the Cauchy stress tensor,  $\lambda_{el}$  and  $\mu_{el}$  are the Lamé constants given in Eq.(4), and  $\hat{\boldsymbol{\sigma}}^\nabla$  is an objective (or frame-invariant) stress rate given as

$$\hat{\boldsymbol{\sigma}}^\nabla = \dot{\boldsymbol{\sigma}} - \mathbf{A}\boldsymbol{\sigma} + \boldsymbol{\sigma}\mathbf{A} \quad (24)$$

where  $\dot{\boldsymbol{\sigma}}$  is the time derivative of the Cauchy stress tensor  $\boldsymbol{\sigma}$  and  $\mathbf{A}$  is an appropriate angular velocity tensor. By combining Eqs.(23) and (24) the objective stress rate is obtained as

$$\dot{\boldsymbol{\sigma}} = \mathbf{A}\boldsymbol{\sigma} - \boldsymbol{\sigma}\mathbf{A} + \lambda_{el} (tr \mathbf{D}^e) \mathbf{I} + 2\mu_{el} \mathbf{D}^e \quad (25)$$

The choice of an appropriate angular velocity tensor  $\mathbf{A}$  is typically done by the respective FE codes, where EPX [7] uses the Jaumann rate of the Cauchy stress  $\hat{\boldsymbol{\sigma}}^{\nabla J}$  given as

$$\hat{\sigma}^{\nabla J} = \dot{\sigma} - \mathbf{W}\sigma + \sigma\mathbf{W}, \quad \mathbf{A} = \mathbf{W} = \frac{1}{2}(\mathbf{L} - \mathbf{L}^T) \quad (26)$$

where  $\mathbf{W}$  is the spin tensor indicating the rate of rotation of an element and  $\mathbf{L}$  is the velocity gradient (see e.g. [1] for more details). Another frequently used objective stress is obtained by using the Green-Naghdi stress rate  $\hat{\sigma}^{\nabla G}$  [1] expressed as

$$\hat{\sigma}^{\nabla G} = \dot{\sigma} - \dot{\mathbf{R}}\mathbf{R}^T\sigma + \sigma\dot{\mathbf{R}}\mathbf{R}^T = \dot{\sigma} - \mathbf{\Omega}\sigma + \sigma\mathbf{\Omega}, \quad \mathbf{A} = \mathbf{\Omega} = \dot{\mathbf{R}}\mathbf{R}^T = -\mathbf{R}\dot{\mathbf{R}}^T \quad (27)$$

where  $\mathbf{R}$  is the rotation tensor representing the rigid body motion in the polar decomposition  $\mathbf{F} = \mathbf{R}\mathbf{U}$  (see Figure 3) and  $\mathbf{\Omega}$  is the skew-symmetric rate of the rotation tensor.

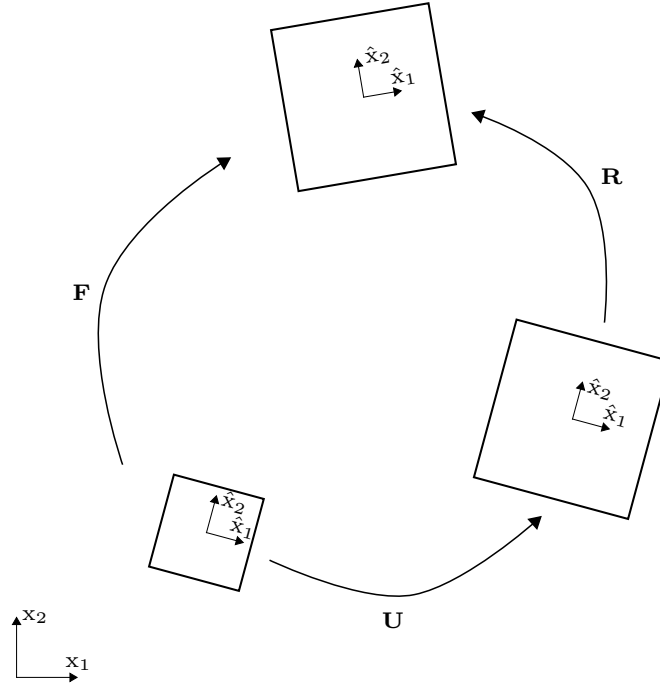


Figure 3: Polar decomposition of a typical homogeneous deformation. The Cartesian coordinate system with coordinate axes  $(x_1, x_2)$  refers to the global reference frame, while  $(\hat{x}_1, \hat{x}_2)$  represents the local (or co-rotated) coordinate system.

From Eqs. (26) and (27) it is seen that the Green-Naghdi rate differs from the Jaumann rate only in using a different measure of the rotation of the material. Despite the difference in Eqs. (26) and (27), it is worth noting that when using polar decomposition  $\mathbf{F} = \mathbf{R}\mathbf{U}$  the spin tensor  $\mathbf{W}$  can be expressed as [12]

$$\mathbf{W} = \dot{\mathbf{R}}\mathbf{R}^T - \frac{1}{2}\mathbf{R}(\dot{\mathbf{U}}\mathbf{U}^{-1} + \mathbf{U}^{-T}\dot{\mathbf{U}})\mathbf{R}^T \quad (28)$$

where  $\mathbf{U}$  is the symmetric right stretch tensor and it is used that  $\dot{\mathbf{F}} = \mathbf{L}\mathbf{F}$ . Thus, for pure rigid motion ( $\dot{\mathbf{U}} = 0$ ) the Green-Naghdi and Jaumann stress rates give the same result since

$$\mathbf{W} = \dot{\mathbf{R}}\mathbf{R}^T = \mathbf{\Omega} \quad (29)$$

whereas in for instance shear one should be aware of the different formulation and how this may influence the results of the FE simulation [1].

As concerns the plastic deformation rate tensor  $\mathbf{D}^p$  this is defined by means of the associated flow rule as

$$\mathbf{D}^p = \dot{\lambda} \frac{\partial f}{\partial \boldsymbol{\sigma}} \quad (30)$$

### 2.1. The von Mises yield criterion

The von Mises yield criterion is an isotropic and pressure-insensitive criterion commonly used for metals and alloys [11]. The pressure insensitivity in metals and alloys is due to the fact that the plastic deformation to a large extent takes place by plastic slip, which is a shear driven deformation mode [10]. Geometrically, the yield criterion  $f(\boldsymbol{\sigma}) = 0$  defines a surface in the stress space which is defined by the components of the stress tensor  $\boldsymbol{\sigma}$  (i.e., the vector space in Figure 4a). The elastic domain is given by the stress space inside this yield surface, while the plastic domain is the surface itself.

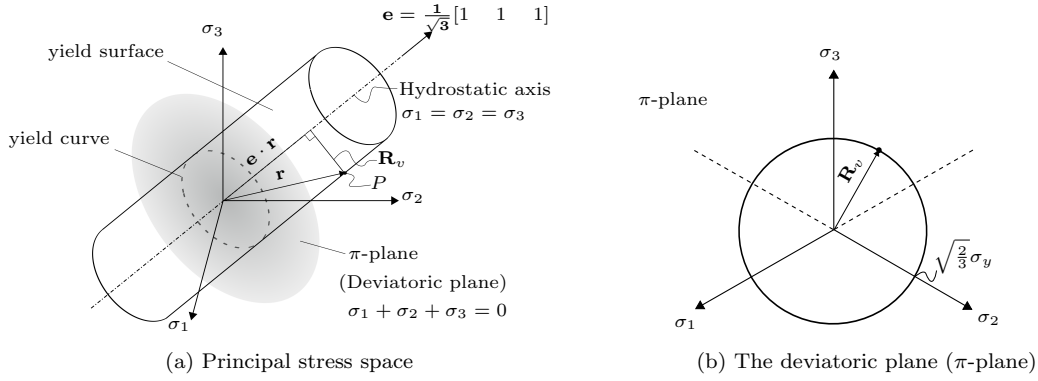


Figure 4: The von Mises yield criterion.

From Figure 4a it is seen that the von Mises yield surface is a circular cylinder of radius  $\mathbf{R}_v = \sqrt{\frac{2}{3}}\sigma_y$ . The stress state on the yield surface could then be represented in the principal stress space as the vector  $\mathbf{r} = [\sigma_1 \ \sigma_2 \ \sigma_3]$ , which has to satisfy the following equation [12]

$$\mathbf{r} \cdot \mathbf{r} - (\mathbf{e} \cdot \mathbf{r})(\mathbf{e} \cdot \mathbf{r}) = \mathbf{R}_v^2 \quad (31)$$

By splitting the stress in deviatoric and hydrostatic parts

$$\boldsymbol{\sigma}' = \boldsymbol{\sigma} - \sigma_H \mathbf{I} \quad \text{and} \quad \sigma_H = \frac{1}{3} \text{tr}(\boldsymbol{\sigma}) = \frac{\sigma_1 + \sigma_2 + \sigma_3}{3} \quad (32)$$

where  $\boldsymbol{\sigma}'$  is the deviatoric stress and  $\sigma_H$  is the hydrostatic stress, the von Mises yield criterion could be expressed as

$$\mathbf{R}_v^2 = \boldsymbol{\sigma} : \boldsymbol{\sigma} - \left( \frac{1}{\sqrt{3}} \text{tr}(\boldsymbol{\sigma}) \right)^2 = (\sigma'_{ij} + \sigma_H \delta_{ij})(\sigma'_{ij} + \sigma_H \delta_{ij}) - 3\sigma_H^2 = \sigma'_{ij} \sigma'_{ij} = \boldsymbol{\sigma}' : \boldsymbol{\sigma}' = \frac{2}{3} \sigma_y^2 \quad (33)$$

Eq.(33) implies that isotropic (or hydrostatic) states of stress do not result in yielding. Thus, the von Mises yield criterion is purely deviatoric and additional hydrostatic stress only translates the stress state along the hydrostatic axis in Figure 4a. Furthermore, the result in Eq.(33) could be presented in terms of the yield function  $f$  from Eq.(5), i.e.,

$$f(\boldsymbol{\sigma}) = \sigma_{eq} - \sigma_y = \sqrt{\frac{3}{2} \boldsymbol{\sigma}' : \boldsymbol{\sigma}'} - \sigma_y = \sqrt{3J_2} - \sigma_y \quad (34)$$

where the von Mises equivalent stress  $\sigma_{eq}$ , in terms of the stress components, reads

$$\sigma_{eq} = \sqrt{\sigma_{11}^2 + \sigma_{22}^2 + \sigma_{33}^2 - \sigma_{11}\sigma_{22} - \sigma_{22}\sigma_{33} - \sigma_{33}\sigma_{11} + 3(\sigma_{12}^2 + \sigma_{23}^2 + \sigma_{31}^2)} \quad (35)$$

and the associated flow rule in Eq.(30) is then given as

$$\mathbf{D}^p = \dot{\lambda} \frac{\partial f}{\partial \boldsymbol{\sigma}} = \dot{\lambda} \frac{3}{2} \frac{\boldsymbol{\sigma}'}{\sigma_{eq}} \quad (36)$$

The rate of the accumulated plastic strain  $\dot{p}$ , in the particular case of associated flow and von Mises isotropic behaviour, is defined as

$$\dot{p} = \sqrt{\frac{2}{3} \mathbf{D}^p : \mathbf{D}^p} \quad (37)$$

## 2.2. The modified Johnson-Cook constitutive model

As discussed earlier, engineering structures subjected to fast transient dynamics may experience large strains, high strain rates and temperature softening. A widely used constitutive relation allowing for these factors is the constitutive model of Johnson and Cook [6], which is empirical and dates back to 1983. The constitutive relation has a multiplicative formulation and is given by

$$\sigma_y = (A + Bp^n) (1 + C \ln(\dot{p}^*)) (1 - T^{*m}) \quad (38)$$

where  $A$ ,  $B$ ,  $C$ ,  $n$  and  $m$  are material constants determined from material tests, and

$$\dot{p}^* = \frac{\dot{p}}{\dot{p}_0} \quad \text{and} \quad T^* = \frac{T - T_r}{T_m - T_r} \quad (39)$$

represents the dimensionless strain rate  $\dot{p}^*$  and the non-dimensional homologous temperature  $T^*$ , respectively. The parameter  $\dot{p}_0$  is a user-defined reference strain rate, while  $T_r$  and  $T_m$  are defined as the room and melting temperature, respectively. The first term in Eq.(38) represents the strain hardening, the second term describes the influence of the strain rate, and the last term represents the temperature softening behaviour. The second and third term shift the hardening curve up or down, depending on the strain rate and the temperature conditions. It is seen from Eq.(38) that the flow stress  $\sigma_y$  increases with increasing plastic strain rate (which is a necessary characteristic in viscoplasticity), while it decreases with increasing temperature. Thus, an increase in the plastic strain rate expands the yield surface whereas an increase in temperature contracts the yield surface.

To avoid non-physical softening effects when  $\dot{p}^* < 1$ , the strain-rate-sensitivity term in the Johnson-Cook model can be adjusted [13]. This leads to the modified Johnson-Cook model, given by [14]

$$\sigma_y = (A + Bp^n) (1 + \dot{p}^*)^C (1 - T^{*m}) \quad (40)$$

This is illustrated by a comparison of Eqs.(38) and (40) in Figure 5, where the ordinate axis refers to the normalized stress  $\sigma_y / [(A + Bp^n) (1 - T^{*m})]$ . Note that, since the formulations in Eqs.(38) and (40) are slightly different, the calibration of the viscoplastic material parameter  $C$  may be somewhat different in the two versions of the Johnson-Cook model.

The first term of the Johnson-Cook model could lead to numerical instabilities when  $n < 1$  due to the initial value of  $p$ , i.e.,  $p$  equal zero before any plasticity occur. This results in an infinite value of the hardening modulus  $H_R = Bnp^{n-1}$  when plasticity occurs. This could be avoided by defining the initial plastic strain

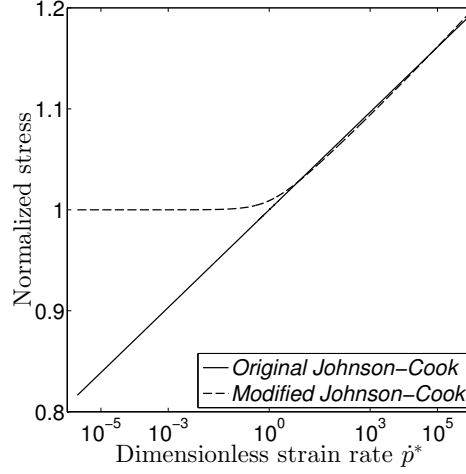


Figure 5: Original vs modified Johnson-Cook model, when  $C_{JC} = 0.014$ ,  $C_{MJC} = 0.013$ , and  $\dot{p}_0 = 5e^{-4}$ .

as an infinite small value. However, it is desirable to formulate the material model as general as possible. Therefore, the first term of Eq.(40), representing the strain hardening, is replaced by another hardening rule than that proposed by Johnson and Cook. By replacing the first term of Eq.(40) with a two terms saturation type of hardening rule proposed by Voce [15], the constitutive model could be expressed as

$$\sigma_y = (A + R(p)) (1 + \dot{p}^*)^C (1 - T^{*m}) = (A + \sum_{k=1}^2 Q_k (1 - \exp(-C_k p))) (1 + \dot{p}^*)^C (1 - T^{*m}) \quad (41)$$

where the constant  $A$  is the initial yield strength of the material, and  $Q_k$  and  $C_k$  ( $k = 1, 2$ ) define the strain hardening. As before, the rate sensitivity is governed by the constant  $C$ , while  $m$  defines the thermal softening behaviour. The hardening modulus  $H_R$  is found from the rate of change in  $R(p)$  given as

$$\dot{R} = \frac{dR}{dt} = \frac{dR}{dp} \frac{dp}{dt} = H_R \frac{dp}{dt} \implies H_R = \frac{dR}{dp} = \sum_{k=1}^2 Q_k C_k e^{(-C_k p)} \quad (42)$$

It is seen from Eq.(42) that no numerical instabilities are expected for the hardening modulus  $H_R$  when the plastic strain equals zero.

Finally, evaluating yielding in the viscoplastic domain and combining Eqs.(34) and (41), the strain-rate sensitivity of the material is controlled by

$$\dot{p} = \dot{\lambda} = \begin{cases} 0 & \text{for } f \leq 0 \\ \dot{p}_0 \left[ \left( \frac{\varphi(\sigma)}{\sigma_y(\sigma_0, R, T)} \right)^{\frac{1}{C}} - 1 \right] & \text{for } f > 0 \end{cases} \quad (43)$$

### 2.3. Adiabatic thermal softening

Assuming that the plastic work is dissipated as heat implies that large plastic deformation results in an increase in temperature. The result of this increase in temperature is a decrease in flow stress. This is already introduced in the last term of Eq.(41), which accounts for the thermal softening of the flow stress at elevated temperatures. However, the evolution of the temperature remains to be established. Since the plastic response of the material is of a very short time duration in fast transient dynamics, the heat transfer is modeled by assuming adiabatic conditions. This implies that there is no heat transfer into or out of the system during the plastic straining. Using the thermal energy balance per unit volume given by [10]

$$\mathfrak{D}_p = \chi \boldsymbol{\sigma} : \mathbf{D}^p = \chi \sigma_{eq} \dot{p} = \rho C_p \dot{T} \quad (44)$$

where  $\rho$  is the material density,  $C_p$  is the specific heat capacity of the solid material and  $\chi$  is the Taylor-Quinney coefficient. The temperature increase due to adiabatic heating can be computed as

$$\dot{T} = \frac{\mathfrak{D}_p}{\rho C_p} \quad (45)$$

From Eq.(44) it is seen that the Taylor-Quinney coefficient  $\chi$  represents the fraction of the plastic power that is converted into heat. The remaining fraction  $1 - \chi$  is assumed to be stored in the material due to structural rearrangements (e.g. elastic "fields" around dislocations). A value of 0.9 is commonly found in the literature for metals, however, a conservative choice would be 1.0 which implies maximum thermal softening. In reality the  $\chi$ -value will not be constant, however, advanced experimental techniques are necessary to determine the evolution of this parameter. Moreover, one should always keep in mind that, while strain hardening (i.e., work hardening due to creation of dislocations) implies an increase of strength locally in the material and distributes the plasticity, softening results in localization of plasticity.

#### 2.4. Ductile failure

A material model is not complete without some form of material degradation or failure. The degradation or failure in a material is usually given in terms of a damage parameter, and failure occurs through damage evolution. Ductile fracture arises from the nucleation, growth and coalescence of microscopic voids that initiate at inclusions and second phase particles. The voids around particles grow when the material is subjected to plastic straining and hydrostatic tension, and fracture occurs when the growing voids reach a critical size, relative to their spacing, resulting in a local plastic instability between the voids [16]. This work considers only ductile failure which is defined as the first sign of fracture, i.e., the nucleation (or initiation) of voids. This means that the fracture criterion is uncoupled from the constitutive equations, and is therefore denoted a failure criterion in the following. The reader is referred to [5, 17] if damage coupling is important in the constitutive model.

Cockcroft and Latham [9] suggested a simple failure criterion which is based on plastic work per unit volume. They reasoned that the failure criterion needed to be based on some combination of stress and strain, and that damage accumulates during plastic straining. To account for hydrostatic tension they based the criterion on the magnitude of the major principal stress  $\sigma_1$ , i.e.,

$$D = \frac{W}{W_c} = \frac{1}{W_c} \int_0^p \langle \sigma_1 \rangle dp \quad (46)$$

where  $W_c$  is the failure parameter which can be found by integrating the major principal stress in a uniaxial tension test during the entire equivalent plastic strain path until the plastic strain at failure  $p_f$ . The expression  $\langle \sigma_1 \rangle$  is equivalent to the function  $\max(0, \sigma_1)$ , which implies that only positive values of the maximum principal stress  $\sigma_1$  (i.e., tensile stress) contribute to the damage evolution. Material failure takes place when the damage parameter  $D$  reaches unity. This failure criterion is therefore easy to implement in a FE code and it requires only one uniaxial tension test for calibration.

The failure criterion by Cockcroft and Latham has proven to give equally good results as more complex criteria [18, 19, 20], and it is also shown that the criterion indirectly accounts for both deviatoric and hydrostatic stress states [21]. This is seen by using an alternative expression for the major principal stress [22], i.e.,

$$\sigma_1 = \sigma_H + \frac{3 + \mu_\sigma}{3\sqrt{3 + \mu_\sigma^2}} \sigma_{eq} = \left( \sigma^* + \frac{3 + \mu_\sigma}{3\sqrt{3 + \mu_\sigma^2}} \right) \sigma_{eq} \quad (47)$$

where the stress triaxiality  $\sigma^*$  and the Lode parameter  $\mu_\sigma$  is defined by

$$\sigma^* = \frac{\sigma_H}{\sigma_{eq}}, \quad \mu_\sigma = \frac{2\sigma_2 - \sigma_1 - \sigma_3}{\sigma_3 - \sigma_1} \quad (48)$$

Remember that  $\sigma_1 \geq \sigma_2 \geq \sigma_3$  are the ordered principal stresses. Eq.(46) can then be expressed as

$$D = \frac{1}{W_c} \int_0^p \left\langle \sigma^* + \frac{3 + \mu_\sigma}{3\sqrt{3 + \mu_\sigma^2}} \right\rangle \sigma_{eq} dp \quad (49)$$

Since material failure occurs when  $D = D_c = 1.0$  and  $p = p_f$ , Eq.(49) can be written as

$$1 = \frac{1}{W_c} \int_0^{p_f} \left\langle \sigma^* + \frac{3 + \mu_\sigma}{3\sqrt{3 + \mu_\sigma^2}} \right\rangle \sigma_{eq} dp \quad (50)$$

Assuming that the stress triaxiality  $\sigma^*$  and Lode parameter  $\mu_\sigma$  remain constant throughout the entire loading history (i.e., proportional loading) and using the Voce hardening rule from Eq.(41) to express the equivalent stress  $\sigma_{eq}$  through the yield condition in Eq.(34), Eq.(50) now reads

$$\sigma^* = \frac{W_c}{Ap_f + \sum_{k=1}^2 (Q_k p_f + \frac{Q_k}{C_k} [\exp(-C_k p_f) - 1])} - \frac{3 + \mu_\sigma}{3\sqrt{3 + \mu_\sigma^2}} \quad (51)$$

The influence of stress triaxiality  $\sigma^*$  and Lode parameter  $\mu_\sigma$  on the plastic failure strain  $p_f$  can then be illustrated as shown in Figure 6, where the material parameters by Holmen et al. [21] are used to generate the respective curves. Various loading scenarios are now represented by different combinations of the stress triaxiality and the Lode parameter. Some typical examples are  $(\mu_\sigma, \sigma^*) = (+1, 1/3)$  which implies uniaxial tension ( $\sigma_1 \geq \sigma_2 = \sigma_3$ ),  $(\mu_\sigma, \sigma^*) = (0, 0)$  corresponding to pure shear ( $2\sigma_2 = \sigma_1 + \sigma_3$ ) and  $(\mu_\sigma, \sigma^*) = (-1, -1/3)$  representing uniaxial compression ( $\sigma_1 = \sigma_2 \geq \sigma_3$ ).

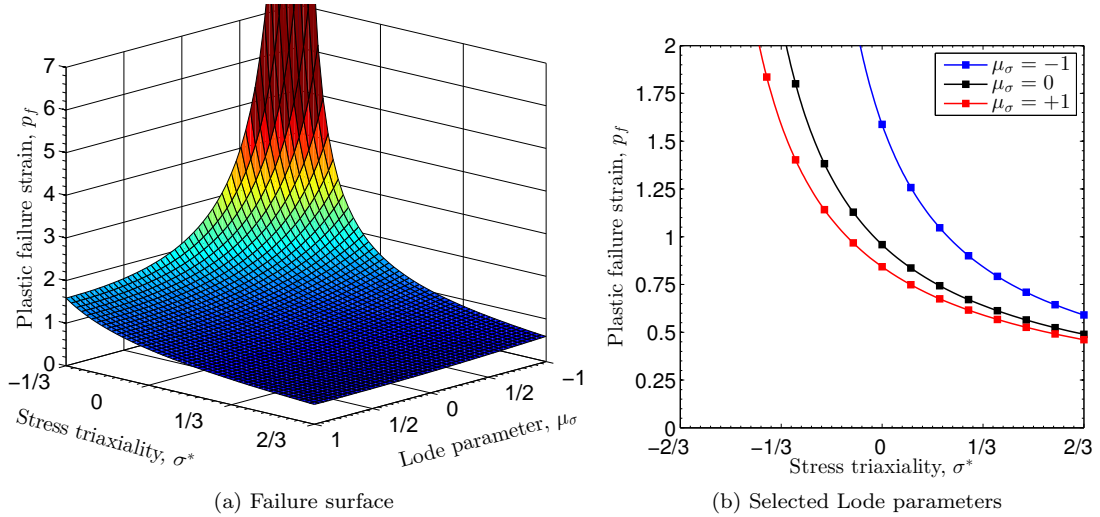


Figure 6: Representation of the plastic failure strain  $p_f$  in the  $(\mu_\sigma, \sigma^*)$  space when using the Cockcroft-Latham failure criterion and assuming proportional loading. Note that the necessary material parameters to be used in Eq. (51) are taken from Table 1.

From Eq.(46) it is seen that, to make the material model as general as possible, it is preferable to compute the principal stresses directly in the subroutine. This is done by following the procedure proposed by [12] which uses the principal deviatoric stresses  $\sigma'_i$  to find the principal stresses  $\sigma_i$ . The principal deviatoric stresses are determined from the eigenvalue problem for the deviatoric stress  $\boldsymbol{\sigma}'$  in Eq. (32), i.e.,



$$\det(\sigma' \mathbf{I} - \boldsymbol{\sigma}') = 0 \quad (52)$$

where  $\det(\cdot)$  is the determinant of the matrix argument  $(\cdot)$ . The result of this operation is a cubic equation called the characteristic equation, given as

$$(\sigma')^3 - I_1'(\sigma')^2 - J_2\sigma' - J_3 = 0 \quad (53)$$

where the principal invariants are given respectively as

$$I_1' = \text{tr}(\boldsymbol{\sigma}') = \sigma'_{ii} = 0, \quad J_2 = -I_2' = \frac{1}{2}\text{tr}(\boldsymbol{\sigma}'^2) = \frac{1}{2}\sigma'_{ij}\sigma'_{ij}, \quad J_3 = I_3' = \det(\boldsymbol{\sigma}') \quad (54)$$

and the  $J$ -symbols for the principal invariants are introduced since these are commonly used in the literature for metal plasticity. The general solution is then given as

$$\sigma'_1 = 2\sqrt{\frac{J_2}{3}} \cos \frac{\theta}{3}, \quad \sigma'_2 = 2\sqrt{\frac{J_2}{3}} \cos \left( \frac{\theta}{3} - \frac{2\pi}{3} \right) \leq \sigma'_1, \quad \sigma'_3 = 2\sqrt{\frac{J_2}{3}} \cos \left( \frac{\theta}{3} + \frac{2\pi}{3} \right) \leq \sigma'_2 \quad (55)$$

where the angle  $\theta$  is given as

$$\cos \theta = \frac{J_3}{2\sqrt{J_2^3/27}}, \quad 0 \leq \theta \leq \pi \quad (56)$$

and the Lode angle is recognized as  $\theta_L = \theta/3$ . Finally, the principal stresses  $\sigma_i$  of the stress tensor  $\boldsymbol{\sigma}$  are determined by

$$\sigma_i = \sigma'_i + \sigma_H = \sigma'_i + \frac{1}{3}\text{tr}(\boldsymbol{\sigma}) \quad (57)$$

where the major principal stress  $\sigma_1$  is used in the Cockcroft-Latham (CL) failure criterion in Eq.(46).

### 3. Two-dimensional stress analysis

This section contains useful relationships when dealing with the particular case of plane stress. Plane stress is defined as the state of stress in which the normal stress  $\sigma_{33}$  and the shear stresses  $\sigma_{13,23}$  are assumed to be zero [11]. From a practical point of view this assumption is typically used in the analysis of bodies in which one of the dimensions (i.e., the thickness) is much smaller than the others and is subjected to loads that result in dominant stresses perpendicular to the thickness direction [5].

#### 3.1. Two-dimensional theory of elasticity

In the particular case of plane stress, the linear elastic stress-strain compliance relationship for an isotropic material can be written as a system of equations in matrix form given as

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 0 \\ 0 \end{bmatrix} = \mathbf{S}\boldsymbol{\sigma} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ 0 \\ \sigma_{12} \\ 0 \\ 0 \end{bmatrix} \quad (58)$$

The three stress components which is assumed to be zero in the stress vector indicate that their associated columns in the compliance matrix  $\mathbf{S}$  can be neglected (i.e., column 3, 5 and 6). Furthermore, if we neglect the rows associated with the strain components in the thickness direction  $\varepsilon_{i3}$  ( $i = 1, 2, 3$ ), the stress-strain compliance relationship reduces to a  $3 \times 3$  matrix given as

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{bmatrix} = \mathbf{S}\boldsymbol{\sigma} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & 0 \\ -\nu & 1 & 0 \\ 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} \quad (59)$$

Then, the elasticity matrix  $\mathbf{C}$  for plane stress is found by inverting the plane stress compliance matrix  $\mathbf{S}$  and given by Hooke's law as

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \mathbf{C}\boldsymbol{\varepsilon} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{bmatrix} \quad (60)$$

It is important to note that the elasticity matrix for plane stress is not found by removing columns and rows from the general isotropic elasticity matrix in Eq.(3).

Finally, in the particular case of plane stress the principal stresses ( $\sigma_1, \sigma_2$ ) may be calculated as

$$\begin{aligned} \sigma_1 &= \frac{\sigma_{11} + \sigma_{22}}{2} + \sqrt{\left(\frac{\sigma_{11} - \sigma_{22}}{2}\right)^2 + \sigma_{12}^2} \\ \sigma_2 &= \frac{\sigma_{11} + \sigma_{22}}{2} - \sqrt{\left(\frac{\sigma_{11} - \sigma_{22}}{2}\right)^2 + \sigma_{12}^2} \end{aligned} \quad (61)$$

where the major principal stress  $\sigma_1$  is used in the CL failure criterion in Eq.(46).

### 3.2. Through thickness strain increment

Although the stress in the thickness direction  $\sigma_{33}$  is assumed to be zero, this is not the case for the corresponding strain  $\varepsilon_{33}$ . The expression for the change in thickness during plane stress may be found by using the relationship between the volumetric strain  $\varepsilon_v (= \varepsilon_{kk})$  and the hydrostatic pressure  $P$  from the theory of three-dimensional elasticity, i.e.,

$$\varepsilon_v = \varepsilon_{kk} = -\frac{1}{K}P, \quad K = \lambda_{el} + \frac{2}{3}\mu_{el} = \frac{E}{3(1-2\nu)} \quad (62)$$

where  $P = -\frac{1}{3}\sigma_{kk}$  is the mean stress and  $K$  is the bulk modulus. Note that the hydrostatic pressure  $P$  is equal to the hydrostatic stress  $\sigma_H$  but has the opposite sign.

The physical meaning of  $\varepsilon_{kk}$  is the relative change of volume

$$\varepsilon_{kk} = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33} = \frac{dV}{V} \quad (63)$$

where experimental evidence has shown that plastic deformation is volume preserving ( $dV = 0$ ) even at high hydrostatic pressures [10]. This means that  $\varepsilon_{kk} = 0$  and that the material is incompressible during plastic deformation. Thus, the strain through the thickness for plates and shells is given as

$$\varepsilon_{33} = -(\varepsilon_{11} + \varepsilon_{22}) \quad (64)$$

where it is seen that the strain in the thickness direction  $\varepsilon_{33}$  is a result of the in-plane strains  $\varepsilon_{11,22}$ . This is typical for thin-walled structures where there are no constraints for the strain in the thickness direction. The only inconsistency is that in the constitutive equations for plates and shells the thickness is considered to be constant while Eq.(64) shows that in reality there will be a small change. This illustrates the importance of defining the thickness strain increment  $\Delta\varepsilon_{33}$  in the material routine to predict the actual change in the thickness of the element.

Using the assumption of a plastic incompressible material (i.e.,  $\varepsilon_{kk}^p = 0$ ) the plastic strain increment through the thickness (due to plastic thinning) is given as

$$\Delta\varepsilon_{33}^p = -(\Delta\varepsilon_{11}^p + \Delta\varepsilon_{22}^p) \quad (65)$$

where  $\Delta\varepsilon_{11}^p$  and  $\Delta\varepsilon_{22}^p$  could be expressed using strain decomposition

$$\begin{aligned} \Delta\varepsilon_{11}^p &= \Delta\varepsilon_{11} - \Delta\varepsilon_{11}^e \\ \Delta\varepsilon_{22}^p &= \Delta\varepsilon_{22} - \Delta\varepsilon_{22}^e \end{aligned} \quad (66)$$

and the elastic strain components are given by Hooke's law in Eq.(59)

$$\begin{aligned} \Delta\varepsilon_{11}^e &= \frac{1}{E}(\Delta\sigma_{11} - \nu\Delta\sigma_{22}) \\ \Delta\varepsilon_{22}^e &= \frac{1}{E}(\Delta\sigma_{22} - \nu\Delta\sigma_{11}) \end{aligned} \quad (67)$$

By combining Eqs.(65) to (67), the plastic strain increment through the thickness is given by

$$\begin{aligned} \Delta\varepsilon_{33}^p &= -(\Delta\varepsilon_{11} - \Delta\varepsilon_{11}^e) - (\Delta\varepsilon_{22} - \Delta\varepsilon_{22}^e) \\ &= \Delta\varepsilon_{11}^e + \Delta\varepsilon_{22}^e - \Delta\varepsilon_{11} - \Delta\varepsilon_{22} \\ &= \frac{1}{E}(\Delta\sigma_{11} - \nu\Delta\sigma_{22} + \Delta\sigma_{22} - \nu\Delta\sigma_{11}) - (\Delta\varepsilon_{11} + \Delta\varepsilon_{22}) \\ &= \frac{1-\nu}{E}(\Delta\sigma_{11} + \Delta\sigma_{22}) - (\Delta\varepsilon_{11} + \Delta\varepsilon_{22}) \end{aligned} \quad (68)$$

Then, when the plastic strain increment  $\Delta\varepsilon_{33}^p$  is known the total through thickness strain increment  $\Delta\varepsilon_{33}$  is finally found by strain decomposition

$$\begin{aligned} \Delta\varepsilon_{33} &= \Delta\varepsilon_{33}^e + \Delta\varepsilon_{33}^p \\ &= -\frac{\nu}{E}(\Delta\sigma_{11} + \Delta\sigma_{22}) + \frac{1-\nu}{E}(\Delta\sigma_{11} + \Delta\sigma_{22}) - (\Delta\varepsilon_{11} + \Delta\varepsilon_{22}) \\ &= \frac{1-2\nu}{E}(\Delta\sigma_{11} + \Delta\sigma_{22}) - (\Delta\varepsilon_{11} + \Delta\varepsilon_{22}) \end{aligned} \quad (69)$$

#### 4. Uniaxial stress case

The case of uniaxial stress, which occurs in bars and in some types of beams, can be treated similarly to the two-dimensional stress analysis case presented in Section 3.1.

In this case one assumes that the two lateral normal stresses are zero ( $\sigma_{22} = \sigma_{33} = 0$ ), so that the corresponding strain increments  $\Delta\varepsilon_{22}$  and  $\Delta\varepsilon_{33}$  must be computed and returned by the constitutive routine.

We assume that the strain increments can be decomposed into elastic and plastic parts, so that

$$\begin{aligned} \Delta\varepsilon_{11}^p &= \Delta\varepsilon_{11} - \Delta\varepsilon_{11}^e \\ \Delta\varepsilon_{22}^p &= \Delta\varepsilon_{22} - \Delta\varepsilon_{22}^e \\ \Delta\varepsilon_{33}^p &= \Delta\varepsilon_{33} - \Delta\varepsilon_{33}^e \end{aligned} \quad (70)$$

and that the plastic part of the deformation occurs at constant volume ( $dV = 0$ ), due to plastic incompressibility

$$\Delta\varepsilon_{11}^p + \Delta\varepsilon_{22}^p + \Delta\varepsilon_{33}^p = 0 \quad (71)$$

In the general 3D case, the elastic strain increments are related to the total stress increments by Hooke's law

$$\begin{aligned}
\Delta\varepsilon_{11}^e &= \frac{1}{E} [\Delta\sigma_{11} - \nu(\Delta\sigma_{22} + \Delta\sigma_{33})] \\
\Delta\varepsilon_{22}^e &= \frac{1}{E} [\Delta\sigma_{22} - \nu(\Delta\sigma_{11} + \Delta\sigma_{33})] \\
\Delta\varepsilon_{33}^e &= \frac{1}{E} [\Delta\sigma_{33} - \nu(\Delta\sigma_{11} + \Delta\sigma_{22})]
\end{aligned} \tag{72}$$

However, in the uniaxial stress case the lateral normal stresses are assumed to vanish ( $\sigma_{22} = \sigma_{33} = 0$ ) and therefore also the stress increments vanish ( $\Delta\sigma_{22} = \Delta\sigma_{33} = 0$ ). Eq. (72) then reads

$$\begin{aligned}
\Delta\varepsilon_{11}^e &= \frac{1}{E} \Delta\sigma_{11} \\
\Delta\varepsilon_{22}^e &= -\frac{\nu}{E} \Delta\sigma_{11} = -\nu \Delta\varepsilon_{11}^e \\
\Delta\varepsilon_{33}^e &= -\frac{\nu}{E} \Delta\sigma_{11} = -\nu \Delta\varepsilon_{11}^e
\end{aligned} \tag{73}$$

which implies that

$$\Delta\varepsilon_{22}^e = \Delta\varepsilon_{33}^e = -\nu \Delta\varepsilon_{11}^e \tag{74}$$

and by utilizing the symmetry it is also assume that

$$\Delta\varepsilon_{22}^p = \Delta\varepsilon_{33}^p \tag{75}$$

and therefore

$$\Delta\varepsilon_{22} = \Delta\varepsilon_{33} \tag{76}$$

$$\varepsilon_{22} = \varepsilon_{33} \tag{77}$$

Then, by replacing Eq. (75) into Eq. (71) one obtains

$$2\Delta\varepsilon_{22}^p = 2\Delta\varepsilon_{33}^p = -\Delta\varepsilon_{11}^p \tag{78}$$

$$\Delta\varepsilon_{22}^p = \Delta\varepsilon_{33}^p = -\frac{1}{2} \Delta\varepsilon_{11}^p = -\frac{1}{2} (\Delta\varepsilon_{11} - \Delta\varepsilon_{11}^e) \tag{79}$$

Using Eq. (73) this now becomes

$$\Delta\varepsilon_{22}^p = \Delta\varepsilon_{33}^p = -\frac{1}{2} \left( \Delta\varepsilon_{11} - \frac{1}{E} \Delta\sigma_{11} \right) = -\frac{1}{2} \Delta\varepsilon_{11} + \frac{1}{2E} \Delta\sigma_{11} \tag{80}$$

Finally, the total lateral strain increment  $\Delta\varepsilon_{22}$  (and  $\Delta\varepsilon_{33}$ ) is obtained by strain decomposition

$$\begin{aligned}
\Delta\varepsilon_{22} &= \Delta\varepsilon_{33} = \Delta\varepsilon_{22}^e + \Delta\varepsilon_{22}^p = \Delta\varepsilon_{33}^e + \Delta\varepsilon_{33}^p \\
&= -\frac{1}{2} \Delta\varepsilon_{11} + \frac{1}{2E} \Delta\sigma_{11} - \nu \Delta\varepsilon_{11}^e \\
&= \frac{1}{2E} \Delta\sigma_{11} - \frac{\nu}{E} \Delta\varepsilon_{11} - \frac{1}{2} \Delta\varepsilon_{11} \\
&= \frac{1-2\nu}{2E} \Delta\sigma_{11} - \frac{1}{2} \Delta\varepsilon_{11}
\end{aligned} \tag{81}$$

## 5. Numerical Return Mapping

Wilkins [23] was one of the first to introduce a return mapping procedure, and a variety of schemes have been proposed following this work (see e.g. [2, 5, 4, 3, 24]). In general, it is possible to categorize the stress integration methods in two different approaches, i.e., the forward Euler method and the backward Euler method. The choice of method is illustrated in Figure 7 and is mainly based on the assumption of the direction of the plastic flow  $\mathbf{n} = \partial f / \partial \boldsymbol{\sigma}$ . The forward Euler method uses the known stress state at the previous or trial configuration to determine the plastic flow direction  $\mathbf{n}$  [2] (see Figure 7a). This assumption is therefore only valid for very small time steps, and is therefore suitable for the explicit time integration FE method. Since the direction is known there is only one unknown during the return mapping procedure, i.e., the plastic multiplier  $\lambda$ . The backward Euler method evaluates the normal  $\mathbf{n}$  to the yield surface at the current unknown stress state [3] (see Figure 7b), which gives good accuracy even for larger time steps making it appropriate for both the explicit and implicit time integration FE methods. However, the variation of the normal to the yield surface must be taken into consideration during the return mapping procedure resulting in a more complex integration, due to the involvement of possible 2<sup>nd</sup> derivatives of the yield function. In the particular case of isotropic von Mises plasticity, the accuracy of the forward and backward Euler method is found to be equal [25]. It should also be emphasized that when considering viscoplasticity the plastic strain rate will change during the local iterations in the forward Euler method. That is, the plastic strain rate will increase during the iterative return mapping (due to an increase in plastic strain). This implies that the return to the yield surface ( $f = 0$ ) occurs at strain rates that are too low. However, the plastic strain rate at the converge increment  $\Delta t_{n+1}$  should be the same as in the backward Euler method.

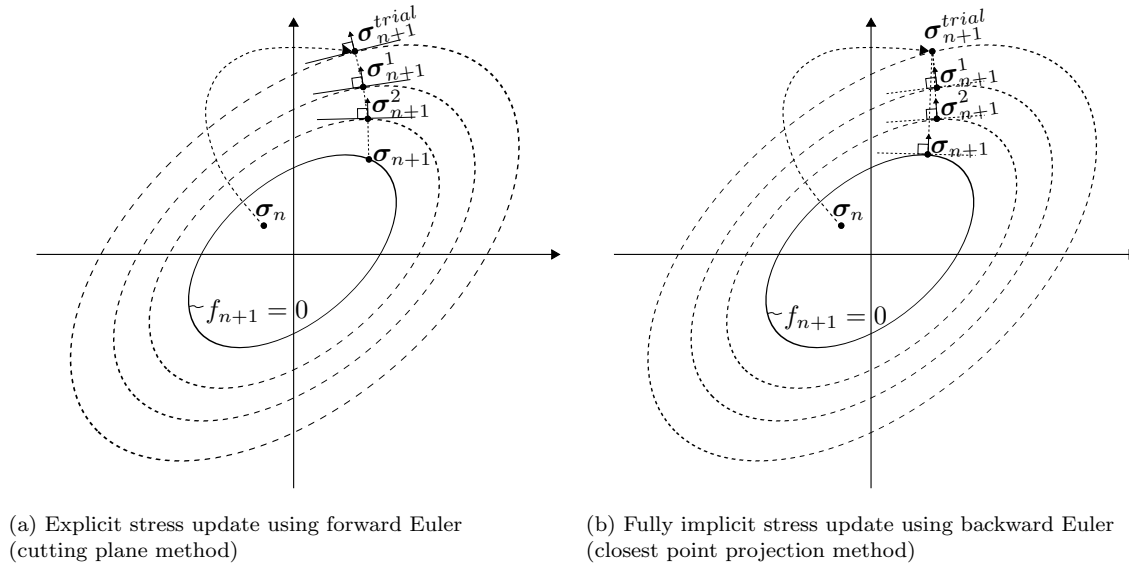


Figure 7: The elastic predictor viscoplastic corrector return map algorithms for the von Mises yield criterion and associated plasticity. Starting from an elastic stress state at the last converged increment  $t_n$  before arriving at the new converged increment  $t_{n+1}$ .

During both approaches an elastic predictor uses the total strain increment and the linear elastic relation to obtain a trial stress state. Then, if the elastic trial stress is outside the plastic domain, the elastically predicted stresses are corrected to fit a suitably updated yield surface by iterations of the plastic corrector. Thus, the proposed integration algorithms are formulated as an elastic predictor viscoplastic corrector stress update using a return mapping scheme. The solution is strain controlled in the sense that the total (co-rotational) strain increment  $\Delta \boldsymbol{\varepsilon}_{n+1} = \mathbf{D}_{n+1} \Delta t_{n+1}$ , within the time interval  $\Delta t_{n+1} = t_{n+1} - t_n$ , is known from the global equilibrium and used to update the stresses and internal variables at step  $n + 1$ . Then, as in

a rate-independent case, it makes sense to first compute an elastic trial state by assuming that the material behaviour is purely elastic within the interval. If the trial state is within the elastic domain ( $f(\boldsymbol{\sigma}) \leq 0$ ), no viscoplastic flow takes place within the considered time step and the trial state is the actual state at the end of the step. Otherwise, the evolution of stresses and internal variables is computed by means of a suitable return mapping method. It should be emphasized that the consistency condition ( $f(\boldsymbol{\sigma}) = 0$ ) no longer holds in the viscoplastic case, since the updated stress state at  $t_{n+1}$  generally lies outside the yield surface ( $f(\boldsymbol{\sigma}) > 0$ ). However, the terminology viscoplastic return mapping is justified in the present case since the updated stresses are obtained by returning (or moving) the trial stress towards the yield surface. Hence, the application of the procedure is similar to that in the rate-independent case, and Eq. (23) reads

$$\Delta\boldsymbol{\sigma} = \mathbf{C} : (\Delta\boldsymbol{\varepsilon} - \Delta\boldsymbol{\varepsilon}^p) = \Delta\boldsymbol{\sigma}^e + \Delta\boldsymbol{\sigma}^p \quad (82)$$

where the stress update at time  $t_{n+1}$  is expressed as

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \Delta\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \mathbf{C} : \Delta\boldsymbol{\varepsilon}_{n+1} - \Delta\lambda_{n+1}\mathbf{C} : \mathbf{n}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} - \Delta\boldsymbol{\sigma}^p \quad (83)$$

in which the viscoplastic corrector  $\Delta\boldsymbol{\sigma}^p$  returns the stress to the yield surface.

All internal variables and functions in Eqs.(20), (23), (34), (36), (42), (43), (45) and (46) are therefore adopted to the return mapping scheme for integration, and replaced with the corresponding incremental values within and at the end of the considered interval  $\Delta t_{n+1}$ . The incremental form of the system of first order differential equations then reads

$$\begin{aligned} \boldsymbol{\varepsilon}_{n+1} &= \boldsymbol{\varepsilon}_n + \Delta\boldsymbol{\varepsilon}_{n+1} \\ \boldsymbol{\varepsilon}_{n+1}^p &= \boldsymbol{\varepsilon}_n^p + \Delta\boldsymbol{\varepsilon}_{n+1}^p = \boldsymbol{\varepsilon}_n^p + \Delta\lambda_{n+1}\mathbf{n}_{n+1} = \boldsymbol{\varepsilon}_n^p + \frac{3}{2}\Delta\lambda_{n+1}\frac{\boldsymbol{\sigma}'_{n+1}}{\sigma_{eq,n+1}} \\ \lambda_{n+1} &= \lambda_n + \Delta\lambda_{n+1} = \lambda_n + \dot{p}_0\Delta t_{n+1} \left[ \left( \frac{\varphi_{n+1}}{\sigma_{y,n+1}} \right)^{\frac{1}{c}} - 1 \right] \\ \boldsymbol{\sigma}_{n+1} &= \boldsymbol{\sigma}_n + \Delta\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \mathbf{C} : (\Delta\boldsymbol{\varepsilon}_{n+1} - \Delta\boldsymbol{\varepsilon}_{n+1}^p) = \boldsymbol{\sigma}_{n+1}^{trial} - \Delta\lambda_{n+1}\mathbf{C} : \mathbf{n}_{n+1} \\ R_{n+1} &= R_n + \Delta R_{n+1} = R_n + \Delta\lambda_{n+1}H_{R,n+1} = R_n + \Delta\lambda_{n+1} \sum_{k=1}^2 Q_k C_k \exp(-C_k \lambda_{n+1}) \\ \sigma_{y,n+1} &= \sigma_0 + R_{n+1} \\ T_{n+1} &= T_n + \Delta T_{n+1} = T_n + \frac{\chi\sigma_{eq,n+1}\Delta\lambda_{n+1}}{\rho C_p} \\ D_{n+1} &= D_n + \Delta D_{n+1} = D_n + \langle \sigma_{1,n+1} \rangle \Delta\lambda_{n+1} \\ f_{v,n+1} &= \varphi_{n+1} - \sigma_{y,n+1} \left( 1 + \frac{\Delta\lambda_{n+1}}{\dot{p}_0\Delta t_{n+1}} \right)^C \left( 1 - \left[ \frac{T_{n+1} - T_r}{T_m - T_r} \right]^m \right) \end{aligned} \quad (84)$$

where  $\Delta\lambda_{n+1}$  is the incremental plastic multiplier at time  $t_{n+1}$ .

The set of nonlinear algebraic equations in Eq. (84) are solved by updating the normal to the yield surface gradient  $\mathbf{n}_{n+1}$  iteratively, and the viscoplastic corrector is applied from the elastic predictor  $\boldsymbol{\sigma}_{n+1}^{trial}$  throughout the iteration until  $f_{v,n+1} = 0$ .

### 5.1. Cutting plane method

The present study focus mainly on the cutting plane method originally proposed by Ortiz and Simo [2]. The basis for this return mapping is an explicit elastic predictor plastic corrector stress update using a forward Euler scheme. That is, the objective of the numerical integration is to use the known stress state at the

previous or trial configuration  $n$  to determine the new converged stress state at step  $n + 1$  (Figure 7a).

By using the normal  $\mathbf{n}_{n+1}^i$  at the previous stress state (see Figure 7a) and assuming weak coupling of the temperature, i.e., using  $T_n$  instead of  $T_{n+1}$ , there is only one unknown during the return mapping procedure and that is the plastic multiplier  $\Delta\lambda_{n+1}^{i+1}$ . This reduces the fully set of equations in Eq. (84) to a single equation which can be solved by a linear Taylor-expansion around  $f_{n+1}(\boldsymbol{\sigma}, \mathbf{X}) = f_{n+1}(\boldsymbol{\sigma}, \sigma_y, v_p(\dot{p}), \Gamma_p(T_n))$ , i.e.,

$$f_{n+1}^{i+1} = f_{n+1}^i + \left. \frac{\partial f}{\partial \boldsymbol{\sigma}} \right|_{n+1}^i \Delta \boldsymbol{\sigma}_{n+1} + \left. \frac{\partial f}{\partial \mathbf{X}} \right|_{n+1}^i \Delta \mathbf{X}_{n+1} \quad (85)$$

which may be written in terms of the internal variables  $\mathbf{X}$  as

$$f_{n+1}^{i+1} = f_{n+1}^i + \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \boldsymbol{\sigma}_{n+1} + \frac{\partial f}{\partial \sigma_{y,n+1}} \delta \sigma_{y,n+1} + \frac{\partial f}{\partial v_{p,n+1}} \delta v_{p,n+1} = 0 \quad (86)$$

where the flow stress  $\sigma_{y,n+1}$  and the following expressions are chosen as the internal variables  $\mathbf{X}$  in the Taylor-expansion

$$v_p(\Delta\lambda_{n+1}) = \left(1 + \frac{\Delta\lambda_{n+1}}{\dot{p}_0 \Delta t_{n+1}}\right)^C \quad \Gamma_p(T_n) = \left(1 - \left[\frac{T_n - T_r}{T_m - T_r}\right]^m\right) \quad (87)$$

Introducing the partial derivatives of the internal variables in Eq. (86) this now reads

$$f_{n+1}^{i+1} = f_{n+1}^i - \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \lambda_{n+1}^{i+1} - v_{p,n+1} \Gamma_{p,n} H_{R,n+1} \delta \lambda_{n+1}^{i+1} - \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \dot{\lambda}_{n+1}} \delta \dot{\lambda}_{n+1}^{i+1} = 0 \quad (88)$$

Moreover, using  $\dot{p} = \dot{\lambda} = \frac{\Delta\lambda}{\Delta t}$  the internal variable controlling the viscoplasticity  $v_p(\dot{p})$  may be expressed as  $v_p(\Delta\lambda)$  and Eq. (88) may be written as

$$f_{n+1}^{i+1} = f_{n+1}^i - \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \lambda_{n+1}^{i+1} - v_{p,n+1} \Gamma_{p,n} H_{R,n+1} \delta \lambda_{n+1}^{i+1} - \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \Delta\lambda_{n+1}} \delta \lambda_{n+1}^{i+1} = 0 \quad (89)$$

Finally, the incremental change  $\delta\lambda$  in the incremental plastic multiplier  $\Delta\lambda$  is then expressed as

$$\delta \lambda_{n+1}^{i+1} = \frac{f_{n+1}^i}{\frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} + v_{p,n+1} \Gamma_{p,n} H_{R,n+1} + \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \Delta\lambda_{n+1}}} \quad (90)$$

where

$$\frac{\partial v_p(\Delta\lambda_{n+1})}{\partial \Delta\lambda} = \frac{C}{\dot{p}_0 \Delta t_{n+1}} \left(1 + \frac{\Delta\lambda_{n+1}}{\dot{p}_0 \Delta t_{n+1}}\right)^{C-1} \quad (91)$$

$$H_{R,n+1} = \frac{dR_{n+1}}{dp_{n+1}} = \sum_{k=1}^2 Q_k C_k \exp(-C_k \lambda_{n+1}) \quad (92)$$

$$\begin{aligned} \left\{ \frac{\partial f}{\partial \boldsymbol{\sigma}} \right\}^T [\mathbf{C}] \left\{ \frac{\partial f}{\partial \boldsymbol{\sigma}} \right\} &= \frac{\partial f}{\partial \sigma_{ij}} C_{ijkl} \frac{\partial f}{\partial \sigma_{kl}} = C_{1111} \left[ \left( \frac{\partial f}{\partial \sigma_{11}} \right)^2 + \left( \frac{\partial f}{\partial \sigma_{22}} \right)^2 + \left( \frac{\partial f}{\partial \sigma_{33}} \right)^2 \right] \\ &+ 2C_{1122} \left[ \frac{\partial f}{\partial \sigma_{11}} \frac{\partial f}{\partial \sigma_{22}} + \frac{\partial f}{\partial \sigma_{22}} \frac{\partial f}{\partial \sigma_{33}} + \frac{\partial f}{\partial \sigma_{33}} \frac{\partial f}{\partial \sigma_{11}} \right] \\ &+ 4C_{1212} \left[ \left( \frac{\partial f}{\partial \sigma_{12}} \right)^2 + \left( \frac{\partial f}{\partial \sigma_{23}} \right)^2 + \left( \frac{\partial f}{\partial \sigma_{31}} \right)^2 \right] \end{aligned} \quad (93)$$

For more details regarding the linear Taylor-expansion, it is referred to Appendix A. It is also emphasized that since the time increment  $\Delta t_{n+1}$  is often very small during explicit time integration, the change in temperature  $\Delta T_{n+1}$  is assumed to have a negligible effect on the update of the state variables. Therefore, the temperature is assumed to have no evolution during the time increment  $\Delta t_{n+1}$ , and a low coupling approach for the temperature is applied. This implies that  $T_n$  is used in the return mapping to update the stress and internal variables at time  $t_{n+1}$ , and then the resulting state variables at  $t_{n+1}$  are used to calculate and update  $T_{n+1}$ . Thus, this assumption simplifies the calculations leading to Eq. (90) without any significant loss of accuracy in the model. This is also the case for the normal  $\mathbf{n}_{n+1} = \partial f / \partial \boldsymbol{\sigma}_{n+1}$  to the yield surface where the use of the previous normal  $\mathbf{n}_{n+1}^i$  instead of  $\mathbf{n}_{n+1}^{i+1}$  violates the associated flow rule in Eqs. (36) and (84) since a unique normal for the given strain increment  $\Delta \boldsymbol{\varepsilon}_{n+1}$  must be found at each stress state. Thus, the deformation does not follow the minimum plastic work path when the cutting plane method is used (see Figure 7). However, when the time increment  $\Delta t_{n+1}$  is very small this is assumed to have negligible effect on the stress state.

The numerical scheme within the time increment  $\Delta t_{n+1}$  may then be summarized as follows (see also Appendix E):

1. Set the initial values of internal variables to the converged values of the previous step at  $t_n$ , check if the integration point under evaluation has already failed, and calculate the speed of sound.
2. Compute the elasticity tensor.

That is, *IF* a 3D stress state is used *THEN* use Eqs. (3) and (4)

$$\mathbf{C} = \lambda_{el} \mathbf{I} \otimes \mathbf{I} + 2\mu_{el} \mathbf{I} = \begin{bmatrix} C_1 & C_4 & C_4 & 0 & 0 & 0 \\ C_4 & C_1 & C_4 & 0 & 0 & 0 \\ C_4 & C_4 & C_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_7 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & C_7 \end{bmatrix}$$

$$C_1 = \frac{(1-\nu)E}{(1+\nu)(1-2\nu)}, \quad C_4 = \frac{\nu C_1}{1-\nu}, \quad C_7 = \frac{1}{2} \frac{E}{1+\nu}$$

*ELSEIF* a 2D stress state is used *THEN* use Eq. (60)

$$\mathbf{C} = \begin{bmatrix} C_1 & C_4 & 0 \\ C_4 & C_1 & 0 \\ 0 & 0 & C_7 \end{bmatrix}, \quad C_1 = \frac{E}{1-\nu^2}, \quad C_4 = \nu C_1, \quad C_7 = \frac{1}{2} \frac{E}{1+\nu}$$

*ELSE* a 1D stress state is used *THEN* use

$$\mathbf{C} = \begin{bmatrix} C_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad C_1 = E$$

3. Use the total strain increment  $\Delta \boldsymbol{\varepsilon}_{n+1}$  from the global equilibrium to compute the elastic predictor

$$\boldsymbol{\sigma}_{n+1}^{trial} = \boldsymbol{\sigma}_n + \mathbf{C} : \Delta \boldsymbol{\varepsilon}_{n+1}$$

4. Compute the von Mises equivalent stress in terms of the elastic trial stress and Eq. (35)

$$\sigma_{eq,n+1}^{trial} = \varphi(\boldsymbol{\sigma}_{n+1}^{trial}) = \sqrt{\frac{3}{2} \boldsymbol{\sigma}_{n+1}^{trial} : \boldsymbol{\sigma}_{n+1}^{trial}}$$



5. Check if temperature softening is included in the material input (i.e.,  $m \neq 0$ ).
6. Set the incremental plastic multiplier  $\Delta\lambda_{n+1}$  equal to zero.
7. Check for plastic admissibility, i.e., *IF*  $f(\boldsymbol{\sigma}_{n+1}^{trial}) > 0$  *THEN* apply return mapping by using the cutting plane algorithm ( $SOLU=1$ ) to find the incremental change in the plastic multiplier  $\Delta\lambda_{n+1}$ . Note that superscript  $i$  denotes the local iteration counter in the following.

- (i) Compute the normal  $\mathbf{n}_{n+1}$  to the yield surface based on the initial (i.e., trial) or previous stress configuration.

$$\mathbf{n}_{n+1} = \frac{3}{2} \frac{\boldsymbol{\sigma}'_{n+1}}{\sigma_{eq}}$$

- (ii) Compute hardening modulus  $H_{R,n+1}$  according to Eq. (42)

$$H_{R,n+1}^i = Q_1 C_1 \exp(-C_1 p_{n+1}^i) + Q_2 C_2 \exp(-C_2 p_{n+1}^i)$$

- (iii) Compute denominator in Eq. (90) using Eqs. (91), (92), (93) and (98)

$$\text{DENOM} = \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}^i} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}^i} + v_{p,n+1}^i \Gamma_{p,n} H_{R,n+1}^i + \sigma_{y,n+1}^i \Gamma_{p,n} \frac{\partial v_{p,n+1}^i}{\partial \Delta\lambda_{n+1}^i}$$

- (iv) Compute incremental change in the plastic multiplier  $\delta\lambda_{n+1}^{i+1}$  in Eq. (90) and update variable  $\Delta\lambda_{n+1}^{i+1}$

$$\delta\lambda_{n+1}^{i+1} = \frac{f_{n+1}^i}{\frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}^i} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}^i} + v_{p,n+1}^i \Gamma_{p,n} H_{R,n+1}^i + \sigma_{y,n+1}^i \Gamma_{p,n} \frac{\partial v_{p,n+1}^i}{\partial \Delta\lambda_{n+1}^i}}$$

$$\Delta\lambda_{n+1}^{i+1} = \Delta\lambda_{n+1}^i + \delta\lambda_{n+1}^{i+1}$$

- (v) Update internal variables dependent on  $\Delta\lambda_{n+1}^{i+1}$

$$p_{n+1}^{i+1} = p_{n+1}^i + \delta\lambda_{n+1}^{i+1}$$

$$R_{n+1}^{i+1} = R_n + H_{R,n+1}^i \Delta\lambda_{n+1}^{i+1}$$

$$\sigma_{y,n+1}^{i+1} = A + H_{R,n+1}^{i+1} \Delta\lambda_{n+1}^{i+1}$$

- (vi) Update stress components using the elastic Hooke's law according to Eq. (83)

$$\boldsymbol{\sigma}_{n+1}^{i+1} = \boldsymbol{\sigma}_{n+1}^{trial} - \Delta\lambda_{n+1}^{i+1} \mathbf{C} : \mathbf{n}_{n+1}^{i+1}$$

Note that the material routine uses a fully vectorized representation of the stress and strain vectors, i.e.,

$$\boldsymbol{\sigma} = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T$$

$$\boldsymbol{\varepsilon} = [\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, \gamma_{12}, \gamma_{23}, \gamma_{31}]^T = [\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, 2\varepsilon_{12}, 2\varepsilon_{23}, 2\varepsilon_{31}]^T$$

Moreover, the application of objective stress rates (i.e., the Jaumann rate in EPX) to update the stress tensor  $\boldsymbol{\sigma}$  in plane stress states allow for the possibility that the normal stress  $\sigma_{33}$  will not be zero (outside the material routine). The material subroutine should therefore initialize the normal stress  $\sigma_{33}$  to zero ( $\sigma_{33} = 0$ ).

- (vii) Compute the updated von Mises equivalent stress using Eq. (35)

$$\sigma_{eq,n+1}^{i+1} = \varphi(\boldsymbol{\sigma}_{n+1}^{i+1}) = \sqrt{\frac{3}{2} \boldsymbol{\sigma}'_{n+1}{}^{i+1} : \boldsymbol{\sigma}'_{n+1}{}^{i+1}}$$

(viii) Update the yield function in Eq. (84) according to the new stress state computed in (vi) and (vii)

$$f_{v,n+1}^{i+1} = \varphi_{n+1}^{i+1} - \sigma_{y,n+1}^{i+1} \left( 1 + \frac{\Delta\lambda_{n+1}^{i+1}}{\dot{p}_0 \Delta t_{n+1}} \right)^C \left( 1 - \left[ \frac{T_n - T_r}{T_m - T_r} \right]^m \right)$$

(viii) Check for convergence, i.e., *IF*

$$\left| \frac{f_v(\Delta\lambda_{n+1}^{i+1})}{\sigma_{y,n+1} v_p \Gamma_p} \right| < \text{TOL}$$

*THEN* continue to 7.

*ELSE*  $i = i + 1$  and *GO TO* (i)

8. Update the temperature according to the low coupling approach using Eq. (45) and check *IF*  $T \geq T_m$

$$T_{n+1} = T_n + \frac{\chi \sigma_{eq,n+1} \Delta\lambda_{n+1}}{\rho C_T}$$

9. Compute the hydrostatic and deviatoric stress respectively according to Eq. (32)

$$\sigma_{H,n+1} = \frac{1}{3} \text{tr}(\boldsymbol{\sigma}_{n+1})$$

$$\boldsymbol{\sigma}'_{n+1} = \boldsymbol{\sigma}_{n+1} - \sigma_{H,n+1} \mathbf{I}$$

10. Computing the principal stresses  $\sigma_{i,n+1}$  based on the principal deviatoric stresses  $\sigma'_{i,n+1}$  as shown in Eqs. (55) and (57)

$$\sigma'_{1,n+1} = 2\sqrt{\frac{J_{2,n+1}}{3}} \cos \frac{\theta}{3}$$

$$\sigma'_{2,n+1} = 2\sqrt{\frac{J_{2,n+1}}{3}} \cos \left( \frac{\theta}{3} - \frac{2\pi}{3} \right) \leq \sigma'_{1,n+1}$$

$$\sigma'_{3,n+1} = 2\sqrt{\frac{J_{2,n+1}}{3}} \cos \left( \frac{\theta}{3} + \frac{2\pi}{3} \right) \leq \sigma'_{2,n+1}$$

where

$$\cos \theta = \frac{J_{3,n+1}}{2\sqrt{J_{2,n+1}^3/27}}, \quad 0 \leq \theta \leq \pi$$

$$\sigma_{i,n+1} = \sigma'_{i,n+1} + \sigma_{H,n+1} = \sigma'_{i,n+1} + \frac{1}{3} \text{tr}(\boldsymbol{\sigma}_{n+1})$$

11. Compute the damage parameter  $D_{n+1}$  by using Eq. (46), i.e., integrating the major principal stress during the entire equivalent plastic strain path

$$W_{n+1} = W_n + \max(\sigma_{1,n+1}, 0) \Delta\lambda_{n+1}$$

$$D_{n+1} = \frac{W_{n+1}}{W_c}$$

12. Update state variables to be returned to the FE code and check for element erosion, i.e., *IF*  $D_{n+1} > D_C$  *THEN* the material point stiffness is deleted.

13. Update the (actual) total strain  $\varepsilon_{33,n+1}$  and incremental strain  $\Delta\varepsilon_{33,n+1}$  through the thickness according to Eq. (69) and return this to the FE code

$$\Delta\varepsilon_{33,n+1} = \frac{1-2\nu}{E}(\Delta\sigma_{11,n+1} + \Delta\sigma_{22,n+1}) - (\Delta\varepsilon_{11,n+1} + \Delta\varepsilon_{22,n+1})$$

$$\varepsilon_{33,n+1} = \varepsilon_{33,n} + \Delta\varepsilon_{33,n+1}$$

In a FE code, this is typically done for each material integration point in the FE assembly. The stresses are updated in the material interface using a fully vectorized version of the forward Euler integration algorithm and a two-state architecture where the initial values at  $t_n$  are stored in the *old* arrays and the new values at  $t_{n+1}$  must be updated and stored in the *new* arrays returned to the global FE analysis. The *VPJC* material routine is valid for 1D, 2D and 3D stress states, i.e., for bar, shell, solid, axisymmetric, plane strain and plane stress elements. The stresses for 3D elements are stored similar to that of symmetric tensors as  $\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T$ , and plane stress, axisymmetric and plane strain elements are stored as  $\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}]^T$ . The deformation gradient and strains are stored similarly to the stresses, however, one should be aware of that the shear strain is stored as engineering shear strains, e.g.  $\gamma_{12} = 2\varepsilon_{12}$ .

## 5.2. Radial return method

The objective of the implicit backward Euler integration algorithm is to find the minimum distance from the trial state to the final state at step  $n+1$  (see Figure 7b). In the particular case of a 3D stress state, von Mises yield criterion with associated flow rule and isotropic hardening, the fully implicit set of equations in Eq. (84) is reduced to a single equation and the radial return algorithm. This is due to the circular cylindrical shape of the yield surface in Figure 4a, which implies that the gradient of the yield surface at the trial state  $\mathbf{n}_{n+1}^{trial}$  is co-linear with the corresponding gradient at the final stress state  $\mathbf{n}_{n+1}$  (Figure 8). The return to the yield surface from the elastic trial state is radial in the deviatoric plane, which simplifies the algorithm and makes the return mapping exceptionally stable and accurate. This makes the radial return mapping one of the most widely used integration algorithms for plain strain and 3D  $J_2$  (or von Mises) elastoplasticity [26].

By considering the deviatoric stress space, the equivalent stress could then be expressed as (see also Appendix B)

$$\sigma_{eq} = \varphi(\boldsymbol{\sigma}'_{n+1}) = \varphi(\boldsymbol{\sigma}'_{n+1}^{trial}) - 3G\Delta\lambda_{n+1}, \quad G = \frac{E}{2(1+\nu)} \quad (94)$$

and the system of equations in Eq. (84) is only dependent on the incremental plastic multiplier  $\Delta\lambda_{n+1}$ , where the viscoplastic constraint  $f_{v,n+1}(\boldsymbol{\sigma}) = 0$  reads

$$f_{v,n+1}(\boldsymbol{\sigma}) = \varphi_{n+1}^{trial}(\boldsymbol{\sigma}) - 3G\Delta\lambda_{n+1} - \sigma_{y,n+1} \left(1 + \frac{\Delta\lambda_{n+1}}{\dot{p}_0\Delta t}\right)^C \left(1 - \left[\frac{T_{n+1} - T_r}{T_m - T_r}\right]^m\right) = 0 \quad (95)$$

Then, the incremental plastic multiplier  $\Delta\lambda_{n+1}$  can be solved by means of a local Newton-Raphson iteration, i.e.,

$$\Delta\lambda_{n+1}^{i+1} = \Delta\lambda_{n+1}^i - \frac{f_v(\Delta\lambda_{n+1}^i)}{\frac{\partial f_v(\Delta\lambda_{n+1}^i)}{\partial \Delta\lambda}} \quad (96)$$

where (again)  $i$  is the local iteration counter.

Since the time increment  $\Delta t_{n+1}$  is often very small during explicit time integration, the change in temperature  $\Delta T_{n+1}$  is assumed to have a negligible effect on the update of the state variables. Hence, the temperature is assumed to have no evolution during the time increment  $\Delta t_{n+1}$ , and a low coupling approach for the temperature is applied. This implies that  $T_n$  is used in the return mapping to update the stress and internal

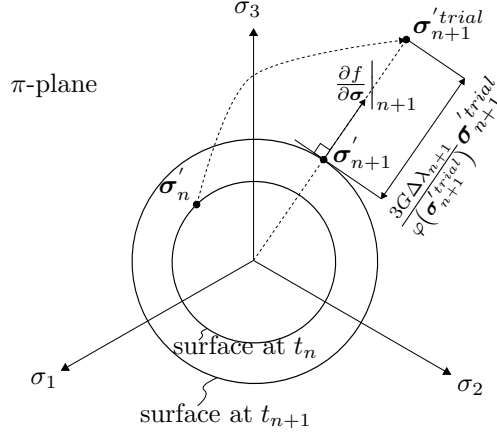


Figure 8: The implicit elastic predictor viscoplastic corrector return map algorithm for the von Mises yield criterion and associated plasticity in the deviatoric plane.

variables at time  $t_{n+1}$ , and then the resulting state variables at  $t_{n+1}$  are used to calculate  $T_{n+1}$ . This assumption reduces the number of equations in Eq. (84), and the residual derivative in Eq. (96) is expressed as

$$\begin{aligned} \frac{\partial f_v(\Delta\lambda_{n+1})}{\partial \Delta\lambda} &= \frac{\partial}{\partial \Delta\lambda} \left[ \varphi(\sigma'_{n+1}) - 3G\Delta\lambda_{n+1} - \sigma_{y,n+1}v_p(\Delta\lambda_{n+1})\Gamma_p(T_n) \right] \\ &= -3G - \frac{\partial \sigma_{y,n+1}}{\partial \Delta\lambda} v_p(\Delta\lambda_{n+1})\Gamma_p(T_n) - \sigma_{y,n+1} \frac{\partial v_p(\Delta\lambda_{n+1})}{\partial \Delta\lambda} \Gamma_p(T_n) \end{aligned} \quad (97)$$

where

$$\begin{aligned} v_p(\Delta\lambda_{n+1}) &= \left( 1 + \frac{\Delta\lambda_{n+1}}{\dot{p}_0 \Delta t} \right)^C \\ \frac{\partial v_p(\Delta\lambda_{n+1})}{\partial \Delta\lambda} &= \frac{C}{\dot{p}_0 \Delta t} \left( 1 + \frac{\Delta\lambda_{n+1}}{\dot{p}_0 \Delta t} \right)^{C-1} \\ \Gamma_p(T_n) &= \left( 1 - \left[ \frac{T_n - T_r}{T_m - T_r} \right]^m \right) \end{aligned} \quad (98)$$

The numerical scheme within the time increment  $\Delta t_{n+1}$  may then be summarized as follows:

1. Set the initial values of internal variables to the converged values of the previous step at  $t_n$ , and set the incremental plastic multiplier  $\Delta\lambda_{n+1}$  equal to zero. Use the total strain increment  $\Delta\epsilon_{n+1}$  from the global equilibrium to compute the elastic predictor

$$\sigma_{n+1}^{trial} = \sigma_n + \mathbf{C} : \Delta\epsilon_{n+1}$$

2. Compute the equivalent stress in terms of the elastic trial stress

$$\sigma_{eq}^{trial} = \varphi(\sigma_{n+1}^{trial}) = \sqrt{\frac{3}{2} \sigma_{n+1}^{trial} : \sigma_{n+1}^{trial}}$$

and the radial normal vector for the yield surface

$$\mathbf{n}_{n+1} = \frac{3}{2} \frac{\sigma_{n+1}^{trial}}{\sigma_{eq}^{trial}}$$

3. Check for plastic admissibility, i.e., *IF*  $f(\boldsymbol{\sigma}_{n+1}^{trial}) > 0$  *THEN* perform local Newton-Raphson iteration to find  $\Delta\lambda_{n+1}$

$$(i) \quad \Delta\lambda_{n+1}^{i+1} = \Delta\lambda_{n+1}^i - \frac{f_v(\Delta\lambda_{n+1}^i)}{\frac{\partial f_v(\Delta\lambda_{n+1}^i)}{\partial \Delta\lambda}}$$

$$(ii) \quad \text{IF } \left| \frac{f_v(\Delta\lambda_{n+1}^{i+1})}{\sigma_{y,n+1} v_p \Gamma_p} \right| > \text{TOL} \text{ THEN } i = i + 1 \text{ and GO TO (i)}$$

4. Update internal variables dependent on  $\Delta\lambda_{n+1}$ . Note that only the deviatoric stress will change during the radial return, and that the hydrostatic stress could be calculated based on the elastic trial stress.
5. Update stress components using the elastic Hooke's law, i.e.,

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} - \Delta\lambda_{n+1} \mathbf{C} : \mathbf{n}_{n+1}$$

6. Check for element erosion, i.e., *IF*  $D > D_C$  *THEN* the material point stiffness is deleted.
7. Update state variables to be returned to the global analysis in EPX.

As for the cutting plane method in Section 5.1, this is done for each material integration point in the FE assembly and the stresses are updated using a two-state architecture and returned to the global analysis in EPX. It must be emphasized that the radial return method is only valid in the case of a 3D stress state (Figure 4), and therefore limited to solid, axisymmetric and plane strain elements. The stresses for three-dimensional elements are stored similar to that of symmetric tensors as  $\boldsymbol{\sigma} = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T$ , and plane strain and axisymmetric elements are stored as  $\boldsymbol{\sigma} = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}]^T$ . The deformation gradient and strains are stored similarly to the stresses, however, one should be aware of that the shear strain is stored as engineering shear strains, e.g.  $\gamma_{12} = 2\varepsilon_{12}$ .

## 6. Verification of Material Subroutine

The performance of the material subroutine is verified and validated in EPX. First, a verification is performed using single element simulations in uniaxial tension and simple shear. Then, the model is validated against experimental data found in the literature for quasi-static tension tests [21] and airblast experiments [27].

The material model requires the user to specify 20 parameters, where 16 are mandatory while the room (or ambient) temperature  $T_r$ , convergence criterion  $TOL$ , maximum number of local iterations  $MXITER$  and preferred solution algorithm  $SOLU$  are taken as default if not specified in the input file [7]. The material constants used for the verification and validation in the following are taken from the literature for a ductile structural steel [21] and summarized in Table 1. Previous studies by De Borst and Feenstra [25] have shown that the choice of return mapping algorithm is more important for orthotropic yield criteria (e.g. the Hill criterion) than for the isotropic von Mises plasticity theory used in this study. The cutting-plane algorithm ( $SOLU = 1$ ) is the default choice and will therefore be used for the return mapping because this is robust and readily implemented for all stress states (1D, 2D and 3D).

### 6.1. Singel element verification

An extensive single element verification of the VPJC material model is performed in Reference [28]. The following single element verification is therefore mainly included for completeness of this study, and it is referred to [28] for more detailed single element verifications. Since the material model is applicable for one- (1D), two- (2D) and three-dimensional (3D) stress states, the scope of this work is limited to one representative bar element in 1D (*FUN3*), one continuum element in 2D (*Q42L*), one shell element in 3D (*Q4GS*) and one continuum element in 3D (*CUBE*). This is based on the findings in [28], which concluded that the

Table 1: Material parameters used in numerical simulations [7, 21].

Elastic constants and density			Yield stress and strain hardening					Strain-rate hardening		Damage	
E	$\nu$	$\rho$	A	$Q_1$	$C_1$	$Q_2$	$C_2$	$\dot{p}_0$	C	$W_C$	$D_C$
[MPa]	[-]	$kg/m^3$	[MPa]	[MPa]	[-]	[MPa]	[-]	$[s^{-1}]$	[-]	[MPa]	[-]
210000	0.33	7850	370	236.4	39.3	408.1	4.5	5e-4	0.01	473	1.0
Adiabatic heating and temperature softening					Convergence		Solution algorithm				
$C_p$	$\chi$	$T_m$	$T_r$	m	TOL	MXITER	SOLU				
[J/kgK]	[-]	[K]	[K]	[-]	[-]	[-]	[1: cutting plane or 2: radial return]				
452	0.9	1800	293	1.0	$10^{-5}$	100	1				

results are more or less independent of element type. Note that the *JAUM* option must be activated when using CEA continuum elements (e.g. *Q42L* and *CUBE*) to treat large strains, and that the *LMST* option must be included to update the thickness for the *Q4GS* elements during plastic straining [7].

The numerical tests of each element are performed in uniaxial tension and simple shear for the 2D and 3D elements, while the 1D element is limited to uniaxial tests only. Note that, in addition to tests in uniaxial tension and simple shear, it is common to validated a material subroutine in terms of uniaxial compression, uniaxial tension/compression in oblique directions, and uniaxial tension/compression with a finite rotation. However, since the formulation of the material model is isotropic the response is the same in tension and compression and independent of loading direction. It is therefore considered sufficient to only test one of these loading conditions. Moreover, since the co-rotational formulation in the material routine is taken care of by EPX outside the material subroutine environment [7], there is no need to test the return map algorithm in terms of finite rotation. The verification of the material routine is therefore limited to the uniaxial tension and simple shear tests in Figure 9.

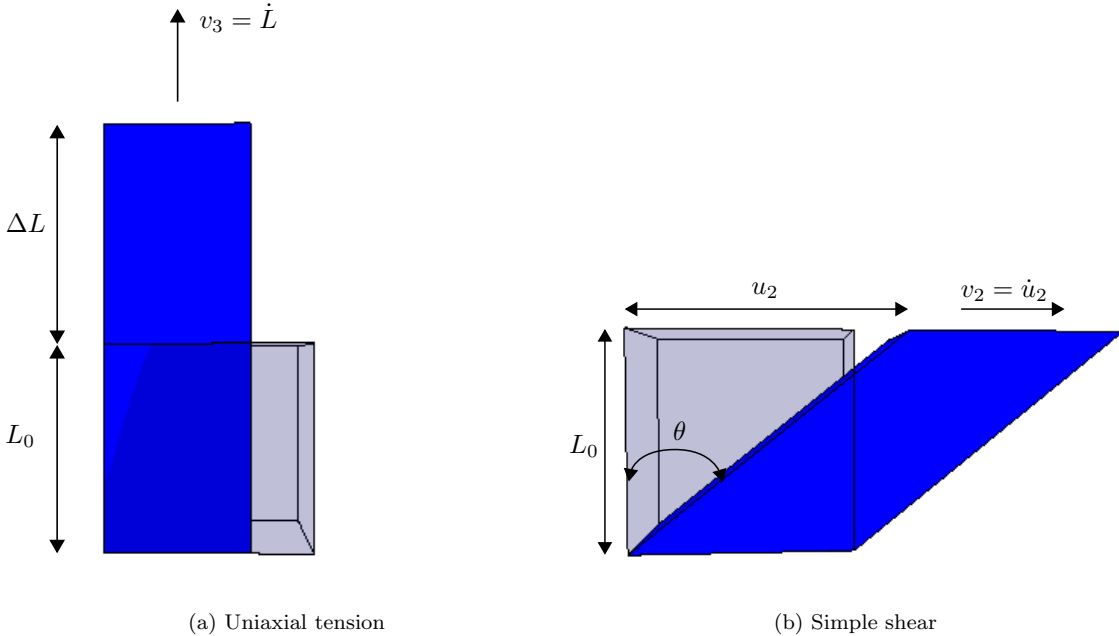


Figure 9: Illustration of the loading cases for the single element tests of the brick element (*CUBE*). Note that  $\gamma = \tan(\theta) = u_2/L_0$ .

The numerical models consist of a simple input file containing one element with element lengths of  $L_0 = 1$  mm. The tests are performed with a prescribed displacement resulting in an equivalent plastic strain of approximately 2.0. This is obtained by applying a constant velocity throughout the analysis (Figure 9), and the results are compared with the analytical solution for the equivalent stress in Eq. (41). In the particular case of including viscoplasticity ( $C \neq 0$ ), it is convenient to test the elements at constant strain rate since the analytical solution of the equivalent stress is readily obtained from Eq. (41). Due to the logarithmic nature of the longitudinal strain in uniaxial tension, i.e.,

$$\varepsilon_{33} = \varepsilon_{33}^e + \varepsilon_{33}^p = \ln\left(\frac{L}{L_0}\right) = \ln\left(1 + \frac{\Delta L}{L_0}\right) \quad (99)$$

It is necessary with an exponential displacement-driven loading to obtain a constant strain rate. By considering large strains and neglecting the elastic part  $\varepsilon_{33}^e$  of the longitudinal strain, Eq. (99) may be simplified to

$$\varepsilon_{33} \approx \varepsilon_{33}^p = p = \dot{p}t = \ln\left(\frac{L(t)}{L_0}\right) \quad (100)$$

which may be expressed as

$$L(t) = L_0 \exp(\dot{p}t) \quad (101)$$

where the displacement is given by

$$d_{33}(t) = L(t) - L_0 = L_0 (\exp(\dot{p}t) - 1) \quad (102)$$

with corresponding velocity

$$v_3(t) = L_0 \dot{p} \exp(\dot{p}t) \quad , \quad v_3(t=0) = v_{3,0} = L_0 \dot{p} \quad (103)$$

Eq. (102) was therefore used to ensure constant strain rate in uniaxial tension, by choosing typical values of the plastic strain rate  $\dot{p}$  and controlling the respective termination times such that the final displacement was the same in all tests. The corresponding viscoplastic response in the simple shear tests was obtained by varying the constant velocity ( $v_2$  in Figure 9) and termination time to obtain an equivalent plastic strain of approximately 2.0. To compare the tests in simple shear with the analytical solution in Eq.(41), it was necessary to scale the loading velocity in simple shear ( $v_2 = \text{constant}$ ) with respect to the initial velocity in uniaxial tension ( $v_3(t=0) = v_{3,0}$ ). The scaling factor was obtained by considering the plastic strain rate  $\dot{p}$  defined in Eq. (37), which for the uniaxial test reads

$$\dot{p} = \sqrt{\frac{2}{3} \mathbf{D}^p : \mathbf{D}^p} = \sqrt{\frac{2}{3} \left[ (\dot{\varepsilon}_{11}^p)^2 + (\dot{\varepsilon}_{22}^p)^2 + (\dot{\varepsilon}_{33}^p)^2 \right]} = \sqrt{\frac{2}{3} \left[ \frac{6}{4} (\dot{\varepsilon}_{33}^p)^2 \right]} = \dot{\varepsilon}_{33}^p \quad (104)$$

when assuming that the material is incompressible during plastic deformation (see Eq. (64)). The corresponding expression in simple shear is

$$\dot{p} = \sqrt{\frac{2}{3} \mathbf{D}^p : \mathbf{D}^p} = \sqrt{\frac{2}{3} \left[ \left( \frac{1}{2} \dot{\gamma}_{12} \right)^2 + \left( \frac{1}{2} \dot{\gamma}_{21} \right)^2 \right]} = \frac{\dot{\gamma}_{12}}{\sqrt{3}} = \frac{\dot{\gamma}}{\sqrt{3}} \quad (105)$$

This implies that the loading velocity in simple shear was  $v_2 = \sqrt{3}v_3$ . In addition, the termination time in the case of simple shear was scaled such that tests in uniaxial tension and simple shear had the same equivalent plastic strain at the end of the simulation (see Table 2).

To illustrate the capabilities of the model, the model is first tested with respect to rate-independent plasticity. Then, the model is tested for its precision in predicting both viscoplastic and thermoviscoplastic

Table 2: Loading in numerical test programme.

Loading	Uniaxial tension				Simple shear			
Velocity [m/s]	$v_{3,0} = 1.0$	0.1	0.01	0.001	$v_2 = 1.732$	0.1732	0.01732	0.001732
Termination [s]	0.002	0.02	0.2	2	0.0021	0.021	0.21	2.1
$\dot{p}$ [s <sup>-1</sup> ]	1000	100	10	1	1000	100	10	1

behaviour, before finally tested with respect to ductile failure. The numerical results were compared to analytical solutions for the constitutive equations, where this was available, to ensure that the overall behaviour of the material model is correct.

### 6.1.1. Rate-independent plasticity

The rate-independent behaviour of the material model is verified by excluding the strain-rate and temperature sensitivity. This is readily done in the material input by choosing  $C = 0$  and  $m = 0$ . The damage threshold was also set to an exaggerated value to avoid element erosion during plastic straining, i.e.,  $D_C \gg 1$ . The analytical solution then reduces to the first term in Eq. (41), which will be used for comparison to the numerical solution.

### 1D bar element

Figure 10 shows the performance of the model in describing the rate-independent behaviour in uniaxial tension for a bar element (*FUN3*).

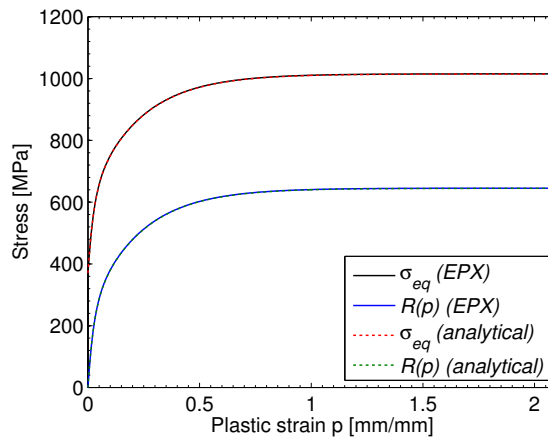


Figure 10: Results from strain-rate and temperature independent 1D single element (*FUN3*) test in uniaxial tension ( $C = 0$ ,  $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to analytical solution in Eq. (41).

### 2D continuum element

Figure 11 shows the capability of the model in describing the rate-independent behaviour in uniaxial tension and simple shear for a 2D continuum and plane stress condition by using the fully integrated quadrilateral *Q42L* element. Note that the *JAUM* option must be activated when using CEA continuum elements (e.g. *Q42L*) to treat large strains.



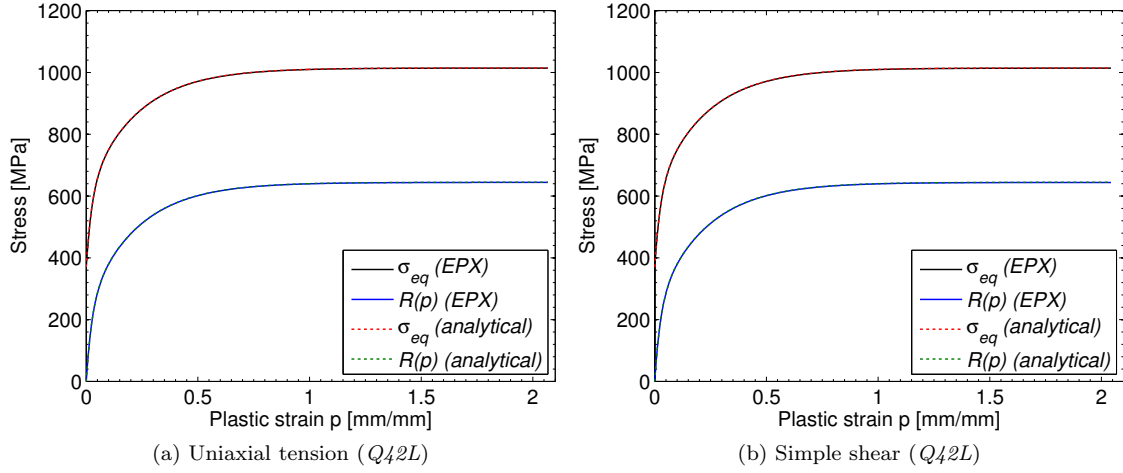


Figure 11: Results from strain-rate and temperature independent 2D single element ( $Q42L$ ) tests ( $C = 0$ ,  $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to analytical solution in Eq. (41).

### 3D shell element

The same tests in uniaxial tension and simple shear are performed also for a 3D (quadrilateral) shell element ( $Q4GS$ ), where the numerical results are compared to the analytical solution from Eq. (41) in Figure 12. Note that the *LMST* option must be included to update the thickness for the  $Q4GS$  elements during plastic straining.

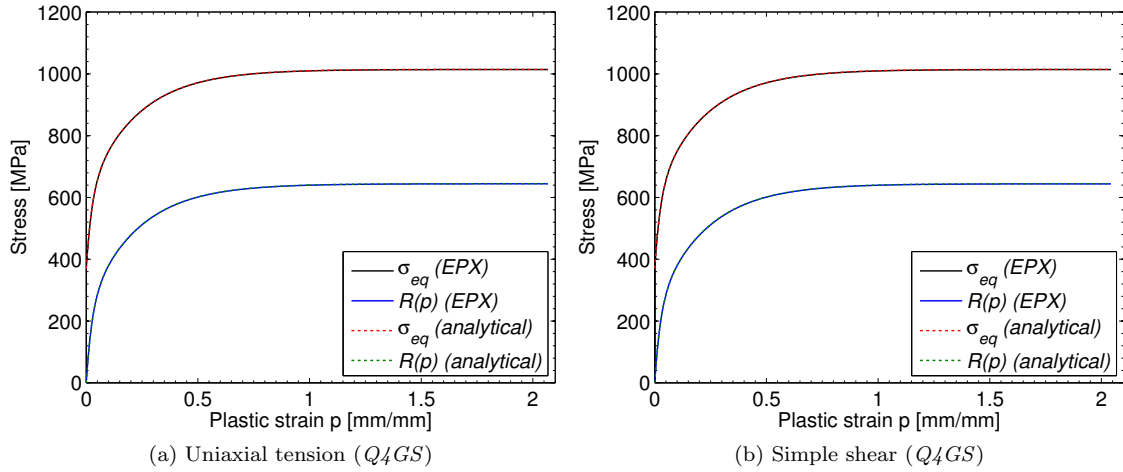


Figure 12: Results from strain-rate and temperature independent 3D single (shell) element ( $Q4GS$ ) tests ( $C = 0$ ,  $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to analytical solution in Eq. (41).

### 3D continuum element

Finally, the material model is verified for a 3D (hexahedron) continuum element ( $CUBE$ ) in uniaxial tension and simple shear (see Figure 13). Again, note that the *JAUM* option must be activated when using CEA continuum elements (e.g.  $CUBE$ ) to treat large strains.

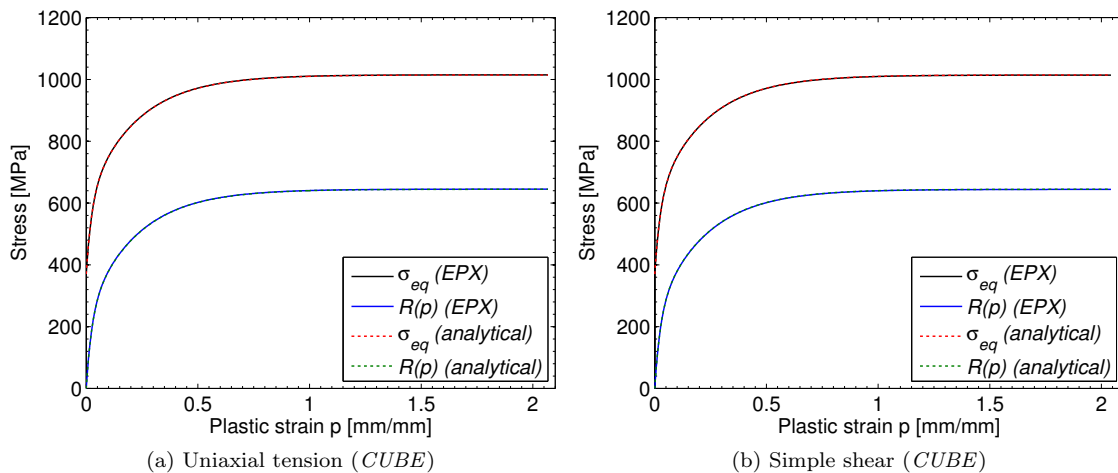


Figure 13: Results from strain-rate and temperature independent 3D single element (*CUBE*) tests ( $C = 0$ ,  $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to analytical solution in Eq. (41).

It is seen that the equivalent stress  $\sigma_{eq}$  from the numerical tests is in excellent agreement with the analytical solution in Eq. (41) for all stress states (Figures 10 - 13), and that the material model is capable of describing the rate-independent behaviour during plastic straining. This is also confirmed by comparing the equivalent stress  $\sigma_{eq}$  and the hardening variable  $R(p)$  at the end of the simulations ( $p = 2.0$ ) in Table 3. Note that negative values of  $\Delta\sigma_{eq}$  and  $\Delta R(p)$  imply that the analytical value is larger than that in the respective numerical test.

Table 3: Summary of single element tests for strain rate and temperature independent behaviour at the end of the simulations ( $p = 2.0$ ).

Element	$\sigma_{eq}$ [MPa]	$\Delta\sigma_{eq}$ [%]	$R(p = 2.0)$ [MPa]	$\Delta R(p = 2.0)$ [%]
Analytical	1014.5	-	644.5	-
<i>FUN3</i> (uniaxial)	1015.1	0.06	645.1	0.10
<i>Q42L</i> (uniaxial)	1014.1	-0.04	644.1	-0.06
<i>Q42L</i> (shear)	1013.9	-0.06	643.9	-0.09
<i>Q4GS</i> (uniaxial)	1014.1	-0.04	644.1	-0.06
<i>Q4GS</i> (shear)	1013.9	-0.06	643.9	-0.09
<i>CUBE</i> (uniaxial)	1014.9	0.04	644.9	0.06
<i>CUBE</i> (shear)	1014.0	-0.04	644.0	-0.07

### 6.1.2. Viscoplasticity

Similar tests are performed by also including viscoplasticity ( $C = 0.01$ ) in the model. This will verify if the model is capable of describing the viscoplastic response at elevated strain rates by varying the plastic strain rate  $\dot{p}$  according to Table 2. That is, starting with a plastic strain rate  $\dot{p} = 1$  and increasing this successively by one order of magnitude until  $\dot{p} = 1000$ . The constant strain rates are obtained in the numerical tests by using Eq. (102) (or Eq. (103)). As for the rate-independent tests, the temperature sensitivity and element erosion are excluded from the numerical tests by choosing  $m = 0$  and  $D_C \gg 1$ . The analytical solution then reduces to the two first terms in Eq. (41), against which the numerical results will be compared. Finally, the rate-independent response ( $\dot{p} \leq 5e^{-4}$ ) from Section 6.1.1 is also plotted for comparison.

### 1D bar element

Figure 14 shows the performance of the model in describing the rate-dependent behaviour in uniaxial tension for a bar element (*FUN3*).

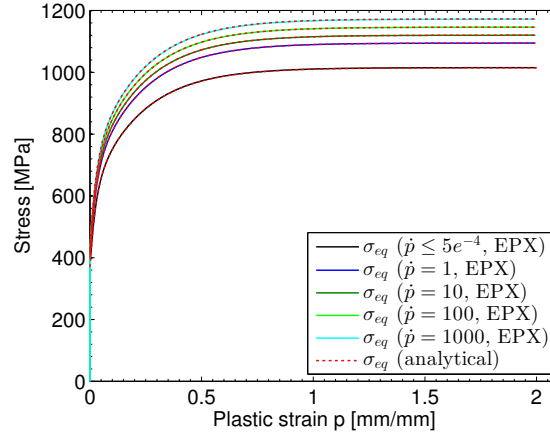


Figure 14: Results from temperature independent 1D single element (*FUN3*) test in uniaxial tension ( $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to the first two terms of the analytical solution in Eq. (41).

### 2D continuum element

Figure 15 shows the capabilities of the model in describing the viscoplastic response at elevated strain rates for the *Q42L* element in uniaxial tension (Figure 15a) and simple shear (Figure 15b).

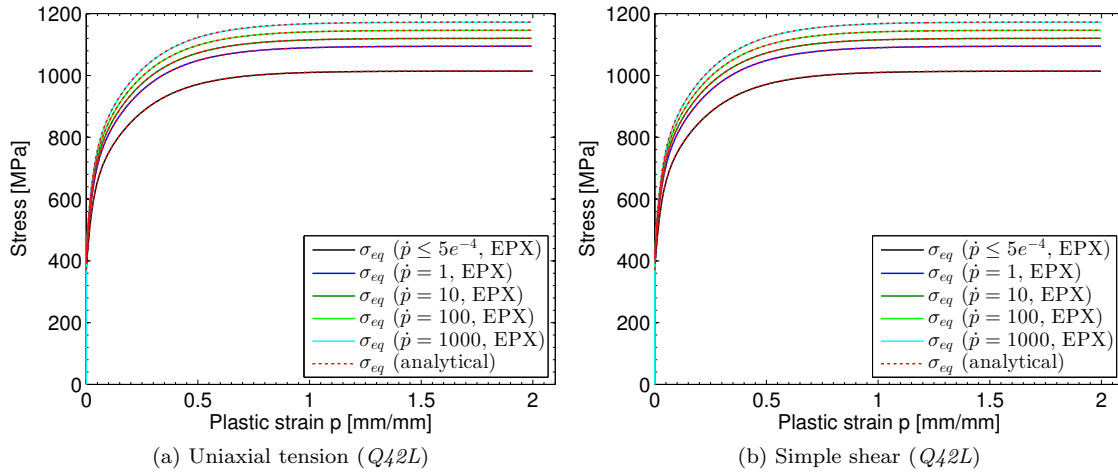


Figure 15: Results from temperature independent 2D single element (*Q42L*) tests ( $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to the first two terms of the analytical solution in Eq. (41).

### 3D shell element

The same tests are performed for the *Q4GS* element in uniaxial tension (Figure 16a) and simple shear

(Figure 16b), where the numerical results are compared to the first two terms of the analytical solution in Eq. (41).

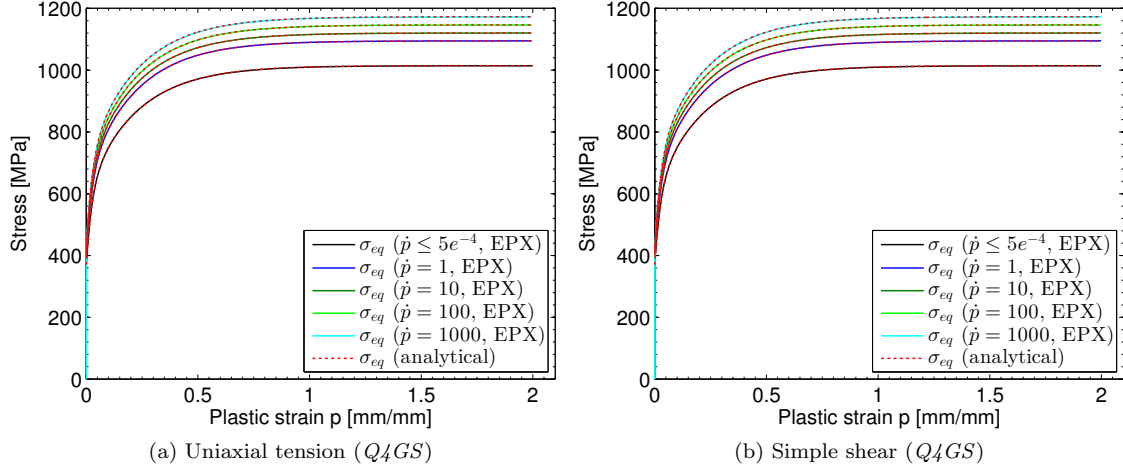


Figure 16: Results from temperature independent 3D single (shell) element ( $Q_4GS$ ) tests ( $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to the first two terms of the analytical solution in Eq. (41).

### 3D continuum element

Finally, the viscoplastic response of the material model is verified for the *CUBE* element in uniaxial tension (Figure 17a) and simple shear (Figure 17b).

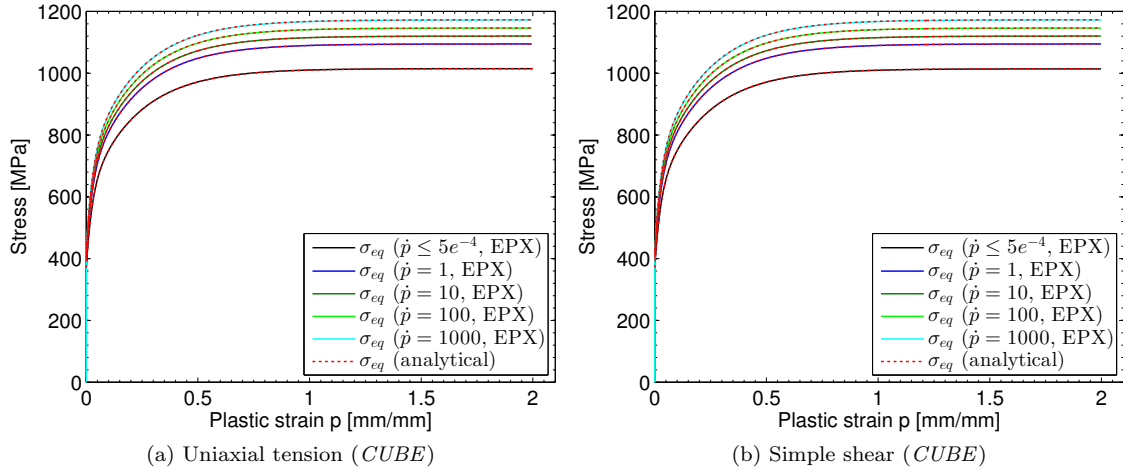


Figure 17: Results from temperature independent 3D single (continuum) element (*CUBE*) tests ( $m = 0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to the first two terms of the analytical solution in Eq. (41).

It is seen that the equivalent stress  $\sigma_{eq}$  from the numerical tests is in excellent agreement with the first two terms of the analytical solution in Eq. (41) for all stress states (Figures 14 - 17), and that the material model is capable of describing the viscoplastic response at elevated strain rates. This is also confirmed by comparing the equivalent stress  $\sigma_{eq}$  and the hardening variable  $R(p)$  at the end of the simulations ( $p = 2.0$ )

in Table 4. By comparing to the results in Table 3 it is seen that the hardening variable  $R(p)$  is only dependent on the plastic straining and independent of strain rate (as expected). Again, note that negative values of  $\Delta\sigma_{eq}$  and  $\Delta R(p)$  imply that the analytical value is larger than that in the respective numerical test. Also note that the analytical reference in Table 4 is found using the plastic strain and temperature history from the uniaxial tests with the *FUN3* element, while the values given for  $\Delta\sigma_{eq}$  and  $\Delta T(p = 2.0)$  for each element and numerical test are found using the corresponding histories from the respective tests. That is, the analytical reference used to compute  $\Delta\sigma_{eq}$  and  $\Delta T(p = 2.0)$  in Table 4 is found using the plastic strain and temperature history from the respective test.

Table 4: Summary of single element tests for temperature independent behaviour at the end of the simulations ( $p = 2.0$ ).

Plastic strain rate [s <sup>-1</sup> ]	Element	$\sigma_{eq}$ [MPa]	$\Delta\sigma_{eq}$ [%]	$R(p = 2.0)$ [MPa]	$\Delta R(p = 2.0)$ [%]
$\leq 5e^{-4}$	Analytical	1014.5	-	644.5	-
	<i>FUN3</i> (uniaxial)	1015.1	0.06	645.1	0.10
	<i>Q42L</i> (uniaxial)	1014.1	-0.04	644.1	-0.06
	<i>Q42L</i> (shear)	1013.9	-0.06	643.9	-0.09
	<i>Q4GS</i> (uniaxial)	1014.1	-0.04	644.1	-0.06
	<i>Q4GS</i> (shear)	1013.9	-0.06	643.9	-0.09
	<i>CUBE</i> (uniaxial)	1014.9	0.04	644.9	0.06
	<i>CUBE</i> (shear)	1014.0	-0.04	644.0	-0.07
1	Analytical	1094.6	-	644.4	-
	<i>FUN3</i> (uniaxial)	1094.6	0.0	644.4	0.0
	<i>Q42L</i> (uniaxial)	1094.6	0.0	644.4	0.0
	<i>Q42L</i> (shear)	1094.6	0.0	644.5	0.0
	<i>Q4GS</i> (uniaxial)	1094.6	0.0	644.4	0.0
	<i>Q4GS</i> (shear)	1094.6	0.0	644.5	0.0
	<i>CUBE</i> (uniaxial)	1094.6	0.0	644.5	0.0
	<i>CUBE</i> (shear)	1094.6	0.0	644.5	0.0
10	Analytical	1120.1	-	644.4	-
	<i>FUN3</i> (uniaxial)	1120.1	0.0	644.4	0.0
	<i>Q42L</i> (uniaxial)	1120.1	0.0	644.4	0.0
	<i>Q42L</i> (shear)	1120.1	0.0	644.5	0.0
	<i>Q4GS</i> (uniaxial)	1120.1	0.0	644.5	0.0
	<i>Q4GS</i> (shear)	1120.1	0.0	644.5	0.0
	<i>CUBE</i> (uniaxial)	1120.1	0.0	644.5	0.0
	<i>CUBE</i> (shear)	1120.1	0.0	644.5	0.0
100	Analytical	1146.1	-	644.4	-
	<i>FUN3</i> (uniaxial)	1146.1	-0.01	644.4	-0.02
	<i>Q42L</i> (uniaxial)	1146.1	-0.01	644.4	-0.01
	<i>Q42L</i> (shear)	1146.1	-0.01	644.4	-0.01
	<i>Q4GS</i> (uniaxial)	1146.1	-0.01	644.4	-0.01
	<i>Q4GS</i> (shear)	1146.1	-0.01	644.4	-0.01
	<i>CUBE</i> (uniaxial)	1146.1	-0.01	644.4	-0.01
	<i>CUBE</i> (shear)	1146.1	0.0	644.4	-0.01
1000	Analytical	1172.8	-	644.4	-
	<i>FUN3</i> (uniaxial)	1171.9	-0.08	643.6	-0.13
	<i>Q42L</i> (uniaxial)	1172.0	-0.07	643.7	-0.11
	<i>Q42L</i> (shear)	1172.2	-0.06	643.9	-0.09
	<i>Q4GS</i> (uniaxial)	1172.0	-0.07	643.7	-0.11
	<i>Q4GS</i> (shear)	1172.2	-0.06	643.9	-0.09
	<i>CUBE</i> (uniaxial)	1172.2	-0.05	643.9	-0.08
	<i>CUBE</i> (shear)	1172.4	-0.04	644.0	-0.07

### 6.1.3. Thermoviscoplasticity

Now, thermoviscoplasticity is also included in the model by specifying  $m = 1.0$  in the material input (according to Table 1). The temperature  $T$  from EPX was used as input in Eq. (41) to obtain the analytical solution for the equivalent stress  $\sigma_{eq}$ , while the incremental plastic strain  $\Delta p$  and equivalent stress  $\sigma_{eq}$  from the numerical solution is used as input in Eq. (45) to obtain an analytical solution for the temperature evolution.

#### 1D bar element

Figure 18 shows the performance of the model in describing the thermoviscoplastic behaviour (Figure 18a) and temperature evolution (Figure 18b) in uniaxial tension for a bar element (*FUN3*).

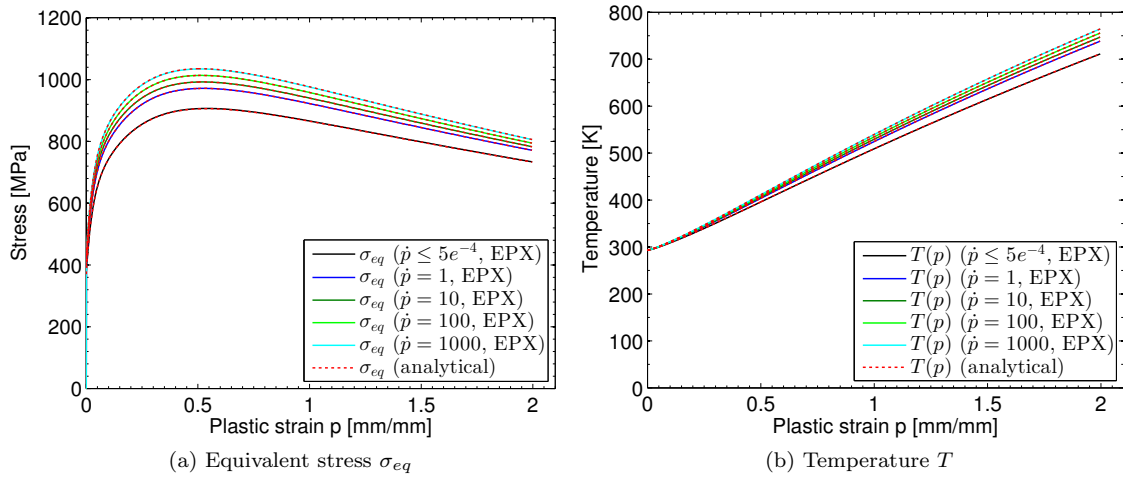


Figure 18: Results from temperature dependent 1D single element (*FUN3*) test in uniaxial tension ( $m = 1.0$  and  $D_C \gg 1$ ). Numerical results in EPX compared to the analytical solution in Eq. (41) and Eq. (45).

#### 2D continuum element

Figure 19 shows the capabilities of the computational model in predicting the thermoviscoplastic response at elevated strain rates for the *Q42L* element in uniaxial tension (Figure 19a) and simple shear (Figure 19b).

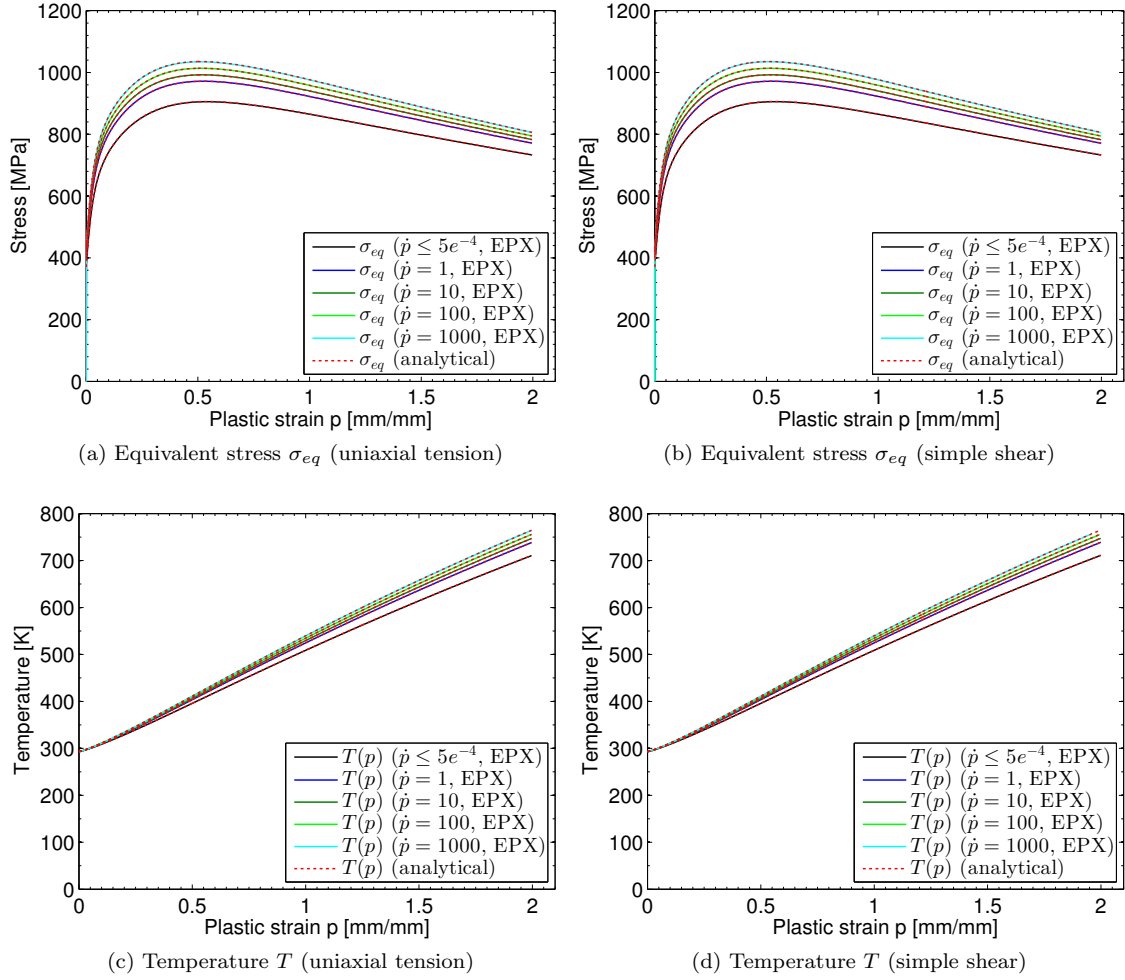


Figure 19: Results from temperature dependent ( $m = 1.0$  and  $D_C \gg 1$ ) 2D single (continuum) element ( $Q42L$ ) tests in uniaxial tension (left) and simple shear (right). Numerical results in EPX compared to the analytical solution in Eq. (41) and Eq. (45).

### 3D shell element

The same tests are performed for the  $Q4GS$  element in uniaxial tension (Figure 20a) and simple shear (Figure 20b).



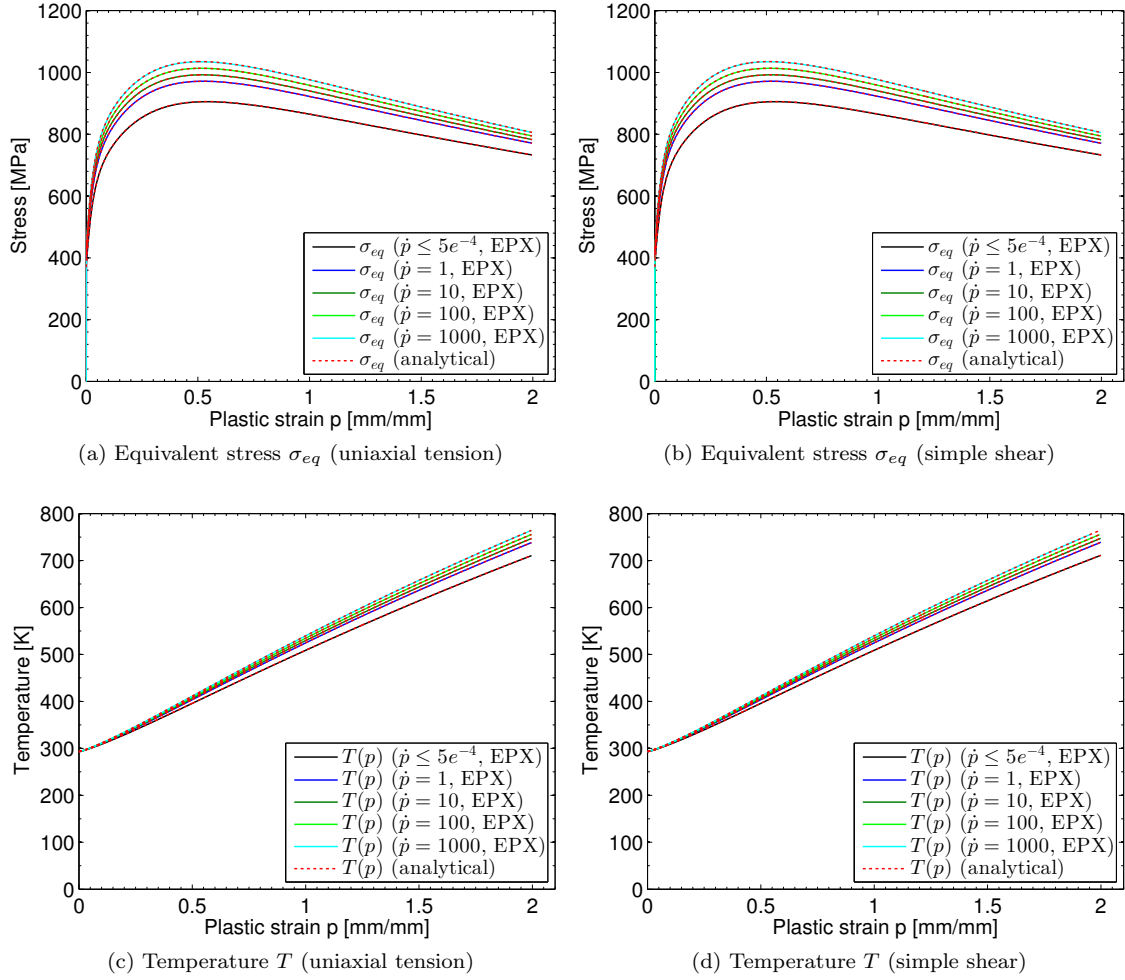


Figure 20: Results from temperature dependent ( $m = 1.0$  and  $D_C \gg 1$ ) 3D single (shell) element ( $Q4GS$ ) tests in uniaxial tension (left) and simple shear (right). Numerical results in EPX compared to the analytical solution in Eq. (41) and Eq. (45).

### 3D continuum element

Finally, the thermoviscoplastic response of the material model is verified for the *CUBE* element in uniaxial tension (Figure 21a) and simple shear (Figure 21b).

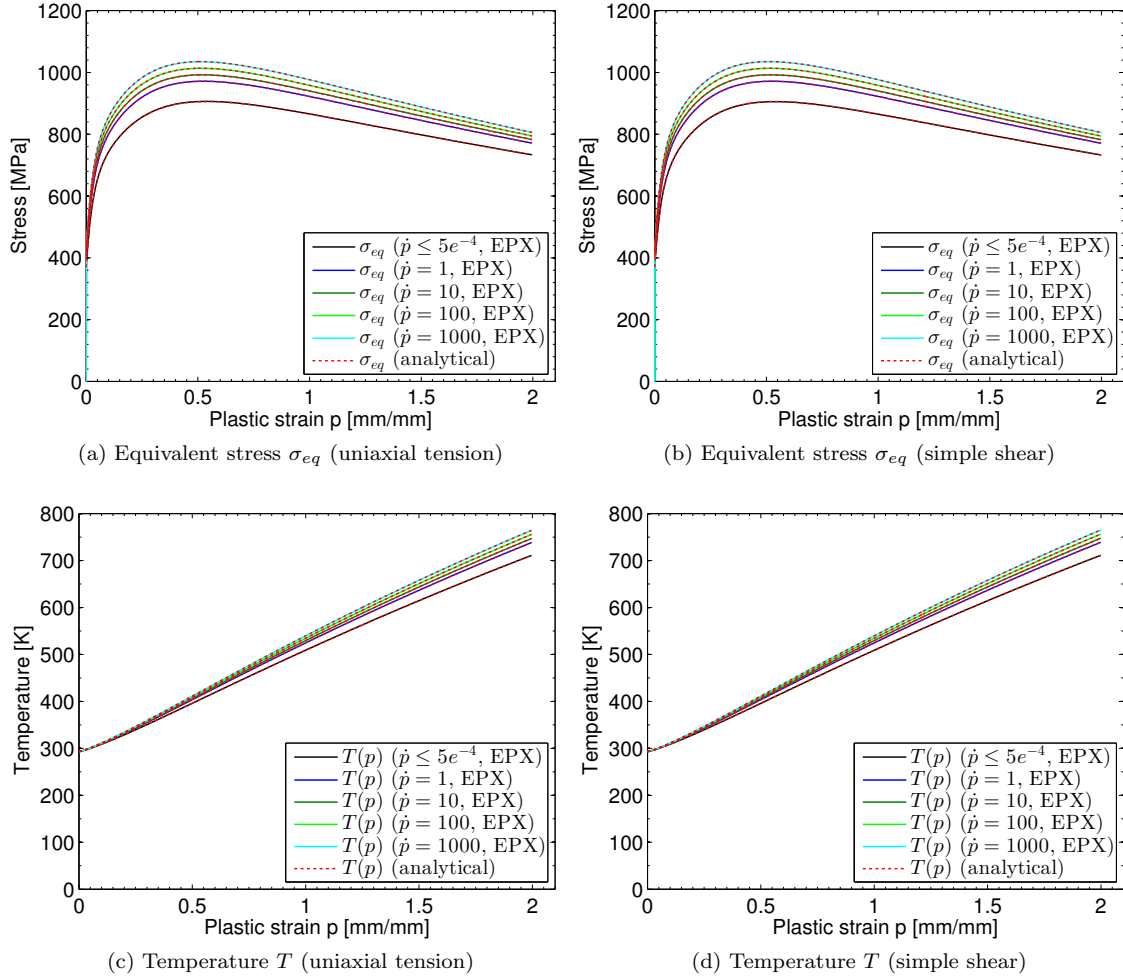


Figure 21: Results from temperature dependent ( $m = 1.0$  and  $D_C \gg 1$ ) 3D single (continuum) element (*CUBE*) tests in uniaxial tension (left) and simple shear (right). Numerical results in EPX compared to the analytical solution in Eq. (41) and Eq. (45).

Again, it is seen that the equivalent stress  $\sigma_{eq}$  from the numerical tests is in excellent agreement with the analytical solution in Eq. (41) for all stress states (Figures 18 - 21), and that the computational model is capable of describing the temperature softening of the equivalent stress due to adiabatic heating. This is also supported by comparing the equivalent stress  $\sigma_{eq}$  and the temperature  $T$  at the end of the simulations ( $p = 2.0$ ) in Table 5. Note that the analytical reference in Table 5 is found using the plastic strain and temperature history from the uniaxial tests with the *FUN3* element, while the values given for  $\Delta\sigma_{eq}$  and  $\Delta T(p = 2.0)$  for each element and numerical test are found using the corresponding histories from the respective tests. That is, the analytical reference used to compute  $\Delta\sigma_{eq}$  and  $\Delta T(p = 2.0)$  in Table 5 is found using the plastic strain and temperature history from the respective test. Also note that in a thermoviscoplastic model there is always a competition between viscoplasticity and thermal softening, where (viscoplastic) strain hardening implies an increase of strength locally in the material and distributes the plasticity while thermal softening results in localization of plasticity.

Table 5: Summary of single element tests for thermo-viscoplastic behaviour at the end of the simulations ( $p = 2.0$ ).

Plastic strain rate [s <sup>-1</sup> ]	Element	$\sigma_{eq}$ [MPa]	$\Delta\sigma_{eq}$ [%]	$T(p = 2.0)$ [K]	$\Delta T(p = 2.0)$ [%]
$\leq 5e^{-4}$	Analytical	733.0	-	711.3	-
	<i>FUN3</i> (uniaxial)	733.7	0.10	711.2	-0.02
	<i>Q42L</i> (uniaxial)	732.9	-0.07	710.5	-0.01
	<i>Q42L</i> (shear)	732.6	-0.05	711.1	-0.01
	<i>Q4GS</i> (uniaxial)	732.9	-0.07	710.5	-0.01
	<i>Q4GS</i> (shear)	732.6	-0.05	711.1	-0.01
	<i>CUBE</i> (uniaxial)	733.5	0.06	710.9	-0.01
	<i>CUBE</i> (shear)	732.7	-0.04	711.1	-0.01
1	Analytical	771.1	-	738.4	-
	<i>FUN3</i> (uniaxial)	771.1	0.0	738.3	-0.02
	<i>Q42L</i> (uniaxial)	771.1	0.0	738.3	-0.02
	<i>Q42L</i> (shear)	770.7	0.0	738.9	-0.02
	<i>Q4GS</i> (uniaxial)	771.1	0.0	738.2	-0.02
	<i>Q4GS</i> (shear)	770.7	0.0	738.9	-0.02
	<i>CUBE</i> (uniaxial)	771.1	0.0	738.3	-0.02
	<i>CUBE</i> (shear)	770.7	0.0	738.9	-0.02
10	Analytical	782.7	-	747.0	-
	<i>FUN3</i> (uniaxial)	782.7	0.0	746.9	-0.01
	<i>Q42L</i> (uniaxial)	782.7	0.0	746.9	-0.01
	<i>Q42L</i> (shear)	782.3	0.0	747.4	-0.02
	<i>Q4GS</i> (uniaxial)	782.7	0.0	746.9	-0.01
	<i>Q4GS</i> (shear)	782.3	0.0	747.4	-0.02
	<i>CUBE</i> (uniaxial)	782.7	0.0	746.9	-0.01
	<i>CUBE</i> (shear)	782.3	0.0	747.5	-0.02
100	Analytical	794.3	-	755.8	-
	<i>FUN3</i> (uniaxial)	794.2	-0.01	755.6	-0.01
	<i>Q42L</i> (uniaxial)	794.2	-0.01	755.6	-0.01
	<i>Q42L</i> (shear)	793.8	-0.01	756.3	-0.02
	<i>Q4GS</i> (uniaxial)	794.3	0.0	755.6	-0.01
	<i>Q4GS</i> (shear)	793.8	-0.01	756.3	-0.02
	<i>CUBE</i> (uniaxial)	794.3	0.0	755.6	-0.01
	<i>CUBE</i> (shear)	793.8	0.0	756.3	-0.02
1000	Analytical	806.0	-	764.5	-
	<i>FUN3</i> (uniaxial)	805.5	-0.06	764.3	-0.02
	<i>Q42L</i> (uniaxial)	805.5	-0.07	764.3	-0.01
	<i>Q42L</i> (shear)	805.2	-0.05	764.9	-0.01
	<i>Q4GS</i> (uniaxial)	805.6	-0.07	764.2	-0.01
	<i>Q4GS</i> (shear)	805.2	-0.05	764.9	-0.01
	<i>CUBE</i> (uniaxial)	805.6	-0.05	764.3	-0.01
	<i>CUBE</i> (shear)	805.2	-0.04	764.9	-0.01

#### 6.1.4. Ductile failure

Finally, ductile failure is included in the model by specifying  $D_c = 1$  in the material input (see Table 1). To couple the model with the element erosion option available in EPX, it is also necessary to include *EROS* in the input file under the directive defining the type of computation (typically *CPLA LAGR EROS 1.0* or *TRID LAGR EROS 1.0*) [7]. Then, when the damage variable in Eq. (46) reaches its critical value  $D = D_c$ , the material (or integration) point stiffness is deleted and the element is removed. Note that the respective material (or integration) point may fail either due to the CL criterion when  $D_c = 1.0$  or when the material melts at  $T = T_m$ . The latter scenario is not included in these single element tests since the material melts at very large strains for the material used in this study. However, the user should be aware that this type of material failure may occur in problems with sufficiently large strains.

#### 1D bar element

Figure 22 shows the performance of the *VPJC* model in predicting ductile failure during the rate-dependent behaviour in uniaxial tension for a bar element (*FUN3*).

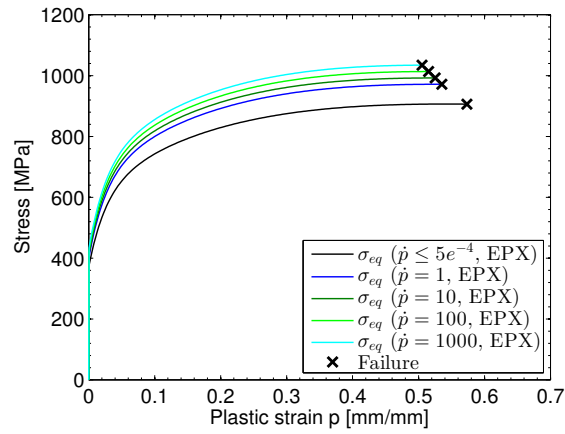


Figure 22: Results from temperature and strain-rate dependent 1D single element (*FUN3*) test in uniaxial tension including failure ( $m = 1.0$  and  $D_c = 1$ ).

#### 2D continuum element

Figure 23 shows the capabilities of the model in predicting ductile failure during the rate-dependent behaviour at elevated strain rates for the *Q42L* element in uniaxial tension (Figure 23a) and simple shear (Figure 23b).

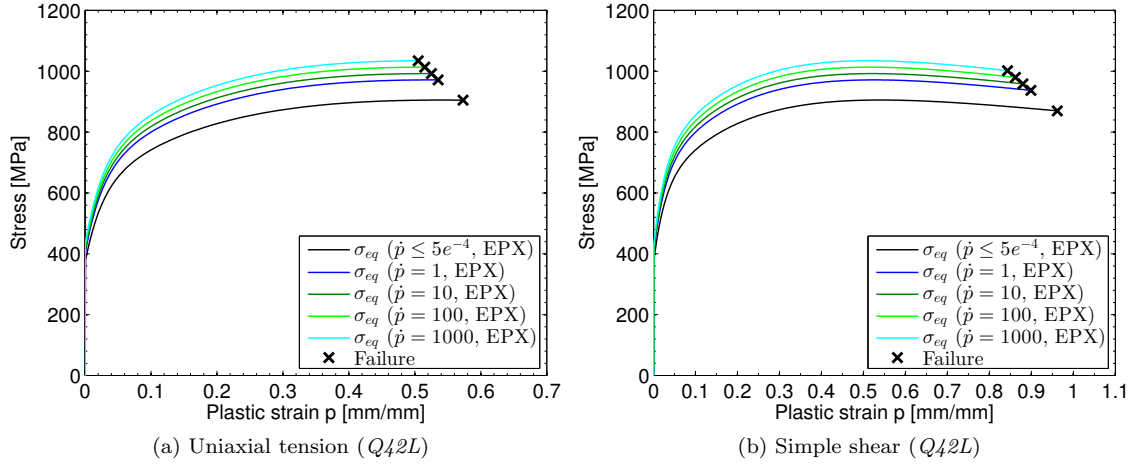


Figure 23: Results from temperature and strain-rate dependent 2D single element ( $Q42L$ ) tests including failure ( $m = 1.0$  and  $D_C = 1$ ).

### 3D shell element

The same tests are performed for the  $Q4GS$  element in uniaxial tension (Figure 24a) and simple shear (Figure 24b).

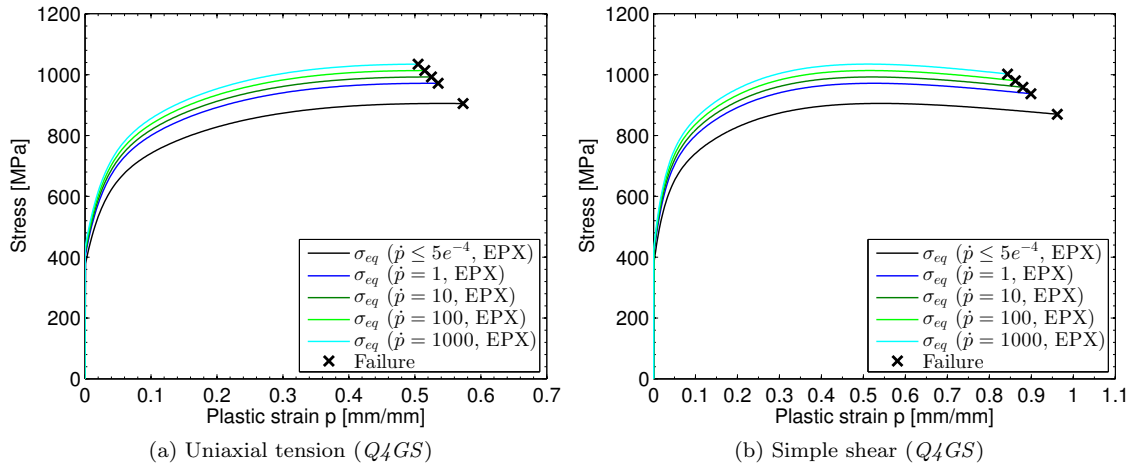


Figure 24: Results from temperature and strain-rate dependent 3D single (shell) element ( $Q4GS$ ) tests including failure ( $m = 1.0$  and  $D_C = 1$ ).

### 3D continuum element

Finally, the models capability in predicting ductile failure is verified for the  $CUBE$  element in uniaxial tension (Figure 25a) and simple shear (Figure 25b).

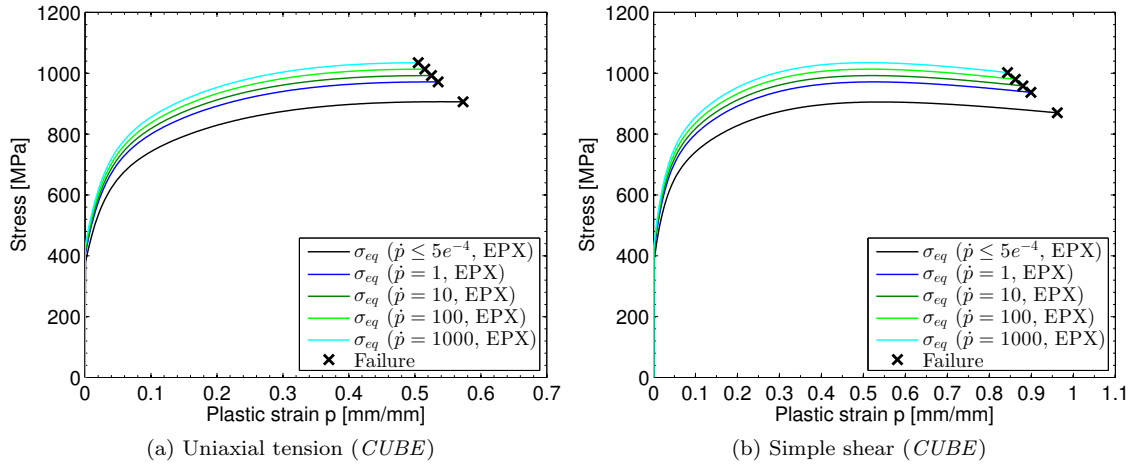


Figure 25: Results from temperature and strain-rate dependent 3D single (continuum) element (*CUBE*) tests including failure ( $m = 1.0$  and  $D_C = 1$ ).

Figures 22 - 25 show that the *VPJC* material model accounts for viscoplasticity when the CL criterion is used to predict ductile failure. That is, elevated strain rates result in reduced elongation to failure due to the viscoplastic increase in the equivalent stress. This is explained by Eq. (20) where an increased strain rate results in an increased flow stress  $\sigma_y(\sigma_0, p, \dot{p})$ . Consequently, the equivalent stress will increase to fulfill the yield criterion ( $f_v(\boldsymbol{\sigma}) = 0$ ). This also results in an increase in the major principal stress  $\sigma_1$  and that the damage parameter  $W_c$  in Eq. (46) is reached at smaller elongations. The difference in elongation at failure  $p_f$  in uniaxial tension and simple shear is mainly explained by the dependence of stress triaxiality  $\sigma^*$  and Lode parameter  $\mu_\sigma$  in Eqs. (46)-(51). Note that the differences observed in failure strain in Figures 23 - 25 corresponds well with the corresponding difference predicted by Eq. (51) in Figure 6.

## 6.2. Tension test

The material parameters ( $Q_1$ ,  $C_1$ ,  $Q_2$  and  $C_2$ ) in the first multiplicative term in Eq. (41) and CL parameter  $W_c$  in Eq. (46) were determined based on inverse modelling of the quasi-static tension tests presented in [21]. These tension tests therefore serve as an excellent basis for validation of the quasi-static part of the *VPJC* material model.

The uniaxial tension tests were modelled based on the experimental geometry (see Figure 26a) and prescribing the same boundary conditions as in the material tests (see Figure 26b). The numerical model was discretized in 3116 4-node quadrilaterals where the initial element size in the gauge area was equal to the thickness (0.8 mm) in an attempt to capture the diffuse and local necking observed in the experiments. Two simulations were performed using both the *Q42L* and *Q4GS* elements, where the former is a plane stress elements with 2 dofs per node and 4 integration points (for full integration) while the latter is a 3D shell element with 6 dofs per node and 20 integration points (5 through the thickness) [7]. Mass-scaling by increasing the density by a factor  $10^9$  ( $\rho = 7850E9$ ) was used to speed up the simulation, because an explicit solution in time results in significant computational times (i.e., a considerable amount of time increments) for quasi-static tests. The quasi-static nature of the test was ensured by comparing the kinetic energy present in the simulations to the plastic dissipation and found negligible. Moreover, since the plastic response of the material cannot be assumed to occur within a short time duration in a quasi-static test, the assumption of an adiabatic condition is no longer valid and the simulation should be run without thermal softening ( $m = 0$ ). It was also assumed negligible viscoplasticity in these tests ( $C = 0$ ), while the remaining material parameters and physical constants were taken from Table 1. The reader is referred to [21, 27] and

[29] for more information about the experimental work and numerical model, respectively, for these tension tests.

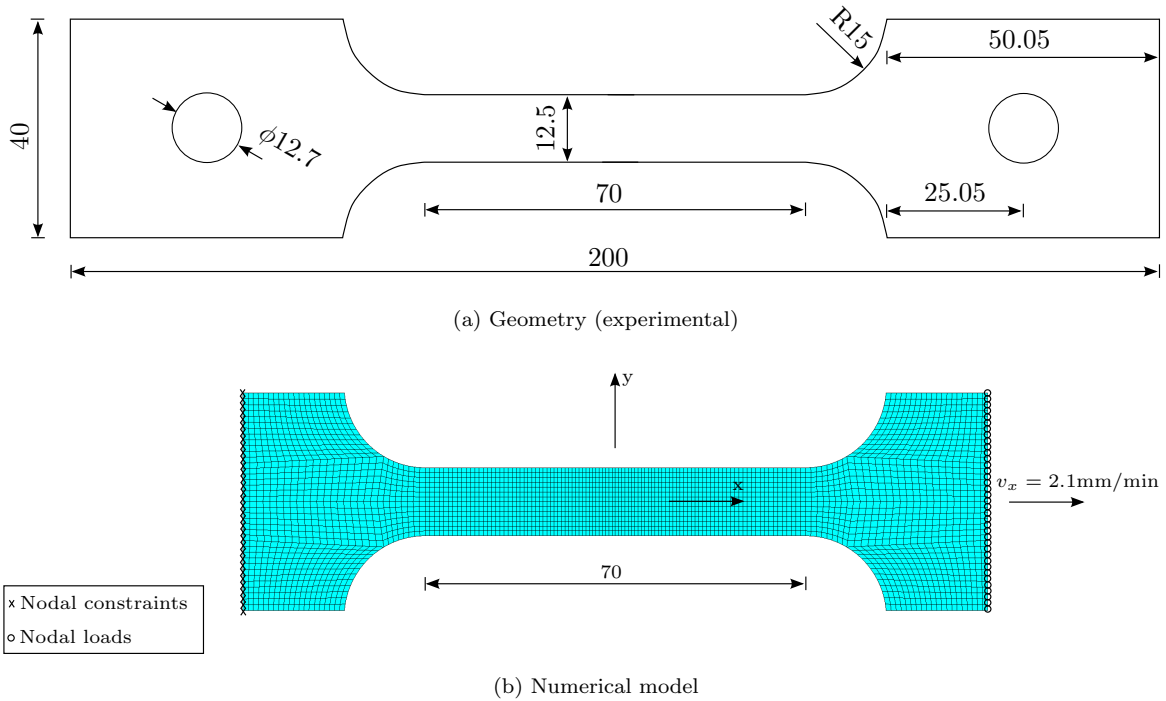


Figure 26: Geometry and numerical model of the uniaxial tension test.

A comparison of the experimental and numerical results in terms of nominal stress-strain curves is presented in Figure 27. It is seen that the numerical simulations are in excellent agreement with the experimental data, and the quasi-static behaviour of the *VPJC* material model may be assumed to be properly implemented in EPX.

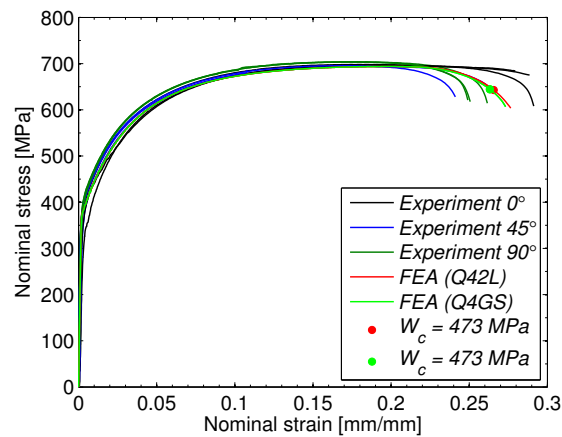
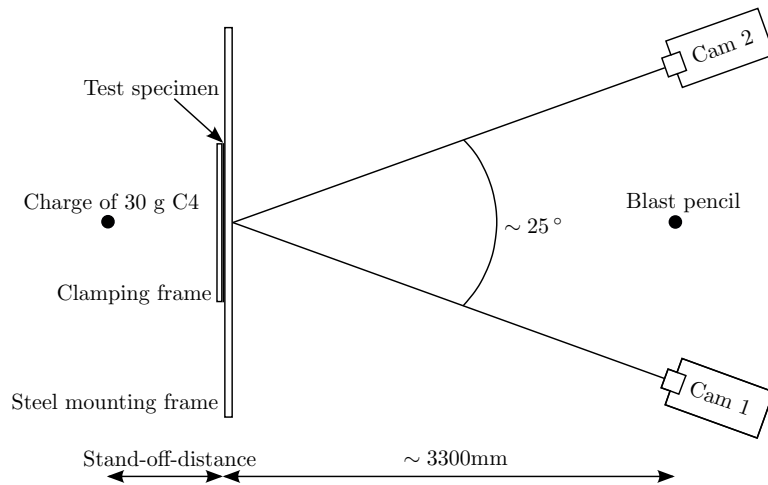


Figure 27: Nominal stress-strain curves from uniaxial tension tests along three different loading directions for the Docol 600 DL steel. Numerical results from EPX (FEA) for both *Q42L* and *Q4GS* elements are compared to the experimental data in [21]. The red and green dots denote the point of failure in the determination of  $W_c$ .

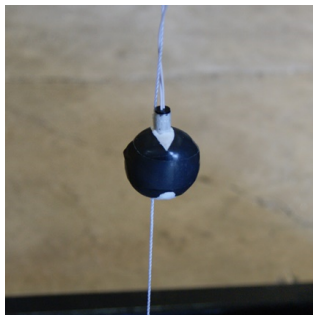
### 6.3. Airblast experiments

A series of airblast experiments on the same steel (Docol 600 DL) material were further used as an validation of the dynamic part of the VPJC material model. Note that the material tests by Holmen et al. [21] were taken from the same plates (i.e., the same batch) as those used in this study. Thus, the material data provided in Table 1 also apply for these airblast experiments.

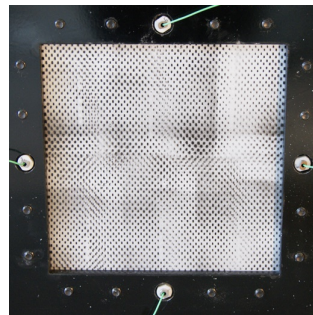
The experimental setup is shown in Figure 28, and consists of a steel mounting frame fixed to the concrete floor with outer dimensions  $1.0 \times 1.0 \times 0.015 \text{ m}^3$  and a square opening of  $0.3 \times 0.3 \text{ m}^2$  at the centre. The square plate specimen with dimensions of  $0.4 \times 0.4 \times 0.0008 \text{ m}^3$  were clamped to the rigid mounting frame using M12 bolts and a clamping frame in an attempt to achieve fixed boundary conditions. The explosive mass  $W$  was positioned at stand-off distances  $R = 0.125 \text{ m}$ ,  $0.250 \text{ m}$  and  $0.375 \text{ m}$  relative to the centre point of the plate. The explosive material was Composition C-4 with a spherical charge, a mass of  $30 \text{ g}$  (equivalent to  $40.2 \text{ g}$  of TNT) and a diameter of approximately  $34.5 \text{ mm}$  (Figure ??). Finally, the plates were painted with a speckle pattern (Figure 28c) and three-dimensional digital image correlation (3D-DIC) analyses were conducted to measure the transient displacement field during the tests. The stereovision setup of the high-speed cameras (Phantom v1610) is illustrated in Figure 28a. The reader is referred to [27] for further details regarding the experimental work.



(a) Sketch of setup



(b) Spherical charge



(c) DIC speckle pattern

Figure 28: Experimental setup airblast test [27]. The speckle pattern in (c) is seen from the cameras.



Figure 29 shows the assembly of the numerical model, where the symmetry of the problem was utilized to model only one quarter of the experimental setup using symmetric boundary conditions. The plate was modelled using a Lagrangian discretization with an element size of approximately 2.5 mm (Figure 29b) and *Q4GS* shell elements, while the bolts and clamping frames were represented by 8-node brick elements (*CUB8*) with 8 integration points and the *VPJC* model with a high elastic limit to ensure elastic behaviour using the physical constants for steel in Table 1. The steel mounting frame and the bolts were modelled as one component (Figure 29a) and the bolts were modelled as stress-free. The pre-tensioning of the bolts was accounted for by applying an external pressure at the contact area ( $A_c = 175 \text{ mm}^2$ ) between the bolt heads and the clamping frame (see Figure 29c). Dividing the pre-tensioning force  $F_p$  in each bolt by this contact area resulted in a contact pressure of 527 MPa between each bolt head and the clamping frame. Contact between the plate, bolts and frames was modelled using a surface-to-surface contact algorithm (*GLIS*). This contact algorithm is based on slave nodes and master surfaces where contact was enforced by Lagrangian multipliers when a slave node penetrated a master surface. The plate was modelled as the slave and the static friction coefficient between the plate and clamping frames was set to 0.15, while the dynamic friction coefficient was equal to 0.10. The blast loading was applied on the plate by using the *AIRBlast* directive in EPX where the mass and positioning of the charge were used as input [7]. A more detailed description of the numerical model is given in [29].

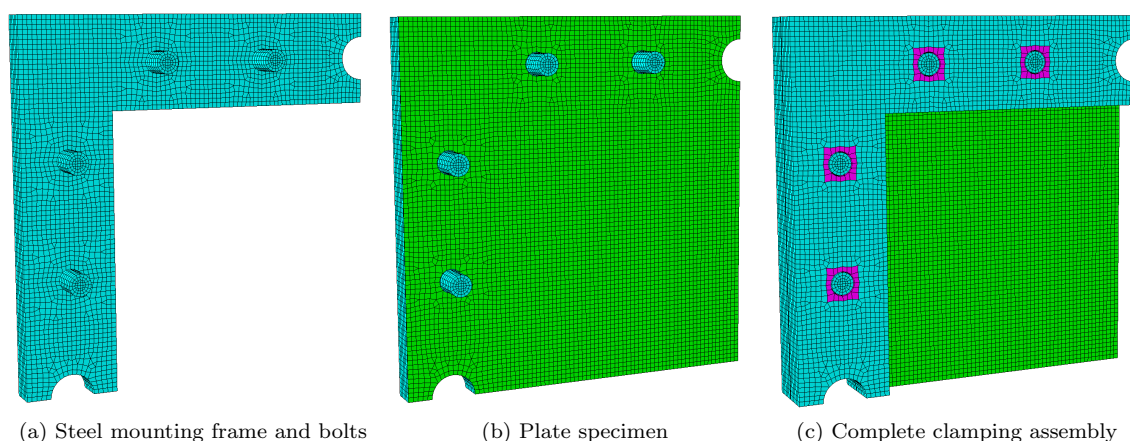


Figure 29: Numerical model of the airblast experiments showing (a) steel mounting frame and bolts as one component (in cyan), (b) plate specimen (in green) and (c) complete assembly including the clamping frame (also in cyan) and contact area between bolt heads and clamping frame (in magenta) used to model the effect of the pre-tensioning of the bolts [29].

A comparison of the experimental and numerical results in terms of mid-point deflection versus time is presented in Figure 30. Considering the complexity in these airblast tests, it is observed a very good agreement between the numerical simulations and experimental observations. In particular, the maximum displacement corresponds very well with the experimental data for the two largest stand-off distances and the reversed snap buckling is captured at the largest stand-off distance. The representation of the maximum displacement is also quite good at the closest stand-off distance, however, the displacement is somewhat underestimated. This is probably due to poorer representation of the actual loading on the plate, since we are outside the range of the empirical methods used in the *AIRB* directive ( $Z = R/W^{1/3} < 0.4 \text{ m/kg}^{1/3}$ ). However, since the scope of this report is to validate the implementation of the *VPJC* model this is not given any more consideration. Thus, the *VPJC* material model is found to perform excellent also in fast transient dynamics.

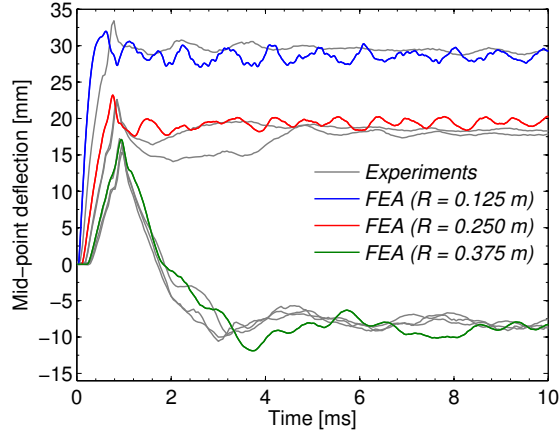


Figure 30: Comparison of experimental and numerical results (FEA) for the airblast tests [29].

## 7. Concluding Remarks

An elastic-thermoviscoplastic Johnson-Cook type constitutive model was implemented in EUROPLEXUS (EPX) and called *VPJC*. The *VPJC* model allows the user to choose between either a fully vectorized version of the forward Euler integration (i.e., cutting plane) algorithm or a special case of the backward Euler (i.e., radial return) algorithm. Ductile failure is also included in the material model through the criterion proposed by Cockcroft and Latham. This allows the model to be coupled with the element erosion option available in EPX. The material model is applicable for one-, two- and three-dimensional stress states and a wide range of elements in EPX.

The performance of the model was first verified using single element simulations in uniaxial tension and simple shear. The model was found to respond very well to all the relevant test cases in terms of rate-independent, viscoplastic and thermoviscoplastic behaviour. The dependence of the stress triaxiality on the failure strain was also shown by including ductile failure in the single element verification. Finally, to ensure a realistic behaviour, the *VPJC* model was validated against experimental data from quasi-static tension tests and airblast experiments. It was found that the numerical simulations were in excellent agreement with the experimental data, and the *VPJC* model is therefore found to be properly implemented in EPX.

## 8. Further work

1. The hardening (Power law) given in Eq. (40) could be included in the model. By introducing a *FLAG* in the material card the user could then choose between Voce hardening (e.g. *FLAG* = 0) or Power law hardening (e.g. *FLAG* = 1). This may be done in the plastic corrector by

```

IF FLAG = 0 THEN use Voce hardening from Eq. (42)
    HR = Q1C1 exp(-C1p) + Q2C2 exp(-C2p)
ELSEIF FLAG = 1 THEN use Power law hardening from Eq. (40)
    HR = Bnpn-1
ELSE Material routine stops and print message:
'VPJC - INVALID TYPE OF HARDENING, FLAG=0 or 1'

```

It should be noted that the material card must be extended to include three more parameters, i.e., *B*, *n* and a *FLAG*.

2. Extend the radial return method to include plane stress. This could be done by constraining the original three-dimensional constitutive equation to obtain a plane stress state. That is, prescribe only the in-plane strains in the three-dimensional equation and use plane stress constraints as an additional condition to determine the in-plane stresses and out-of-plane strains. The reader is referred to [5] (Chapter 9) and [26] for more information on this approach.
3. A substepping scheme could be implemented to increase the stability and accuracy of the return map algorithms.
4. Implement and add an option of using the Johnson-Cook failure criterion [30]. This is done by adding the following during the update of variables depending on *DLAMBDA*

```

C COMPUTE FAILURE STRAIN BASED ON JOHNSON-COOK
  PF = (D1 + D2 * EXP(-D3 * SIGMAH/PHI))
  .* ((1 + DLAMBDA/(PDOT * DT)) ** D4)
  .* (1 + D5 * ((T - TR)/(TM - TR)))
C COMPUTE AND UPDATE DAMAGE VARIABLE
  DJC = DJC + DLAMBDA/PF
*
...continue as is ...
*
990  IF (D >= DC OR DJC >= DC) THEN
*
* the current gp was already failed or is failing now ...
...continue as is ...
  IF (D >= DC) THEN
    WRITE(59,*) 'Element deleted due to CL-criterion'
  IF (DJC >= DC) THEN
    WRITE(59,*) 'Element deleted due to JC-criterion'
...continue as is ...

```

Note that the material input must be extended to include five more parameters, i.e.,  $D_1$ ,  $D_2$ ,  $D_3$ ,  $D_4$  and  $D_5$ . Material constants for a Weldox steel can be found in [14], i.e.,  $D_1 = 0.0705$ ,  $D_2 = 1.732$ ,  $D_3 = -0.54$ ,  $D_4 = -0.015$  and  $D_5 = 0$ .

## References

- [1] T. Belytschko, W. Liu, B. Moran, Nonlinear Finite Elements for Continua and Structures, John Wiley & Sons (2000).
- [2] M. Ortiz, J. Simo, An analysis of a new class of integration algorithms for elastoplastic relations, International Journal of Numerical Methods in Engineering 23 (1986) 353–366.
- [3] J. Simo, R. L. Taylor, Consistent tangent operators for rate-independent elastoplasticity, Computer Methods in Applied Mechanics and Engineering 48 (3) (1985) 101–118.
- [4] R. D. Krieg, D. B. Krieg, Accuracies of numerical solution methods for elastic-perfectly plastic model, ASME Journal of Pressure Vessel Technology 99 (4) (1977) 510–515.
- [5] E. A. de Souza Neto, D. Peric, D. R. J. Owen, Computational methods for plasticity : theory and applications, Wiley, UK (2008).
- [6] G. R. Johnson, W. H. Cook, A Constitutive Model and Data for Metals Subjected to Large Strains, High Strain Rates and High Temperatures, Proceedings of the 7th International Symposium on Ballistics, Hague (1983) 541–547.
- [7] Europlexus, User's manual, Joint Research Centre, [http://europlexus.jrc.ec.europa.eu/public/manual\\_html/index.html](http://europlexus.jrc.ec.europa.eu/public/manual_html/index.html) [cited 07.06.16].

- [8] S. A. Khan, S. Huang, *Continuum Theory of Plasticity*, Wiley-Interscience, 1995.
- [9] M. G. Cockcroft, D. J. Latham, Ductility and the workability of metals, *Journal of the Institute of Metals* 96 (1968) 33–39.
- [10] O. S. Hopperstad, T. Børvik, *Lecture Notes KT8306 Plasticity Theory*, Department of Structural Engineering, SIMLab (2013).
- [11] J. Lubliner, *Plasticity theory*, Dover (2008).
- [12] F. Irgens, *Continuum Mechanics*, Springer-Verlag (2008).
- [13] M. Ortiz, G. T. Camacho, Adaptive Lagrangian modelling of ballistic penetration metallic targets, *Comput. Methods Appl. Mech. Engrg.* 142 (1997) 269–301.
- [14] T. Børvik, O. S. Hopperstad, T. Berstad, M. Langseth, A computational model of viscoplasticity and ductile damage for impact and penetration, *European Journal of Mechanics - A/Solids* 20 (2001) 685–712.
- [15] E. Voce, The relationship between stress and strain for homogenous deformation, *Journal of the Institute of Metals* 74 (1948) 536–562.
- [16] T. L. Anderson, *Fracture Mechanics - Fundamentals and Applications*, Taylor and Francis Group, third edition (2005).
- [17] J. Lemaitre, J.-L. Chaboche, *Mechanics of Solid Materials*, Cambridge University Press.
- [18] T. Børvik, S. Dey, A. H. Clausen, Perforation resistance of five different high-strength plates subjected to small-arms projectiles, *International Journal of Impact Engineering* 36 (2009) 948–964.
- [19] A. Kane, T. Børvik, A. Benallal, O. S. Hopperstad, Failure criteria with unilateral conditions for simulation of plate perforation, *European Journal of Mechanics - A/Solids* 30 (2011) 468–476.
- [20] S. Dey, T. Børvik, O. S. Hopperstad, M. Langseth, On the influence of fracture criterion in projectile impact of steel plates, *Computational Materials Science* 38 (2006) 176–191.
- [21] J. K. Holmen, O. S. Hopperstad, T. Børvik, Low-velocity impact on multi-layered dual-phase steel plates, *International Journal of Impact Engineering* 78 (2015) 161–177.
- [22] J. K. Holmen, J. Johnsen, O. S. Hopperstad, T. Børvik, Influence of fragmentation on the capacity of aluminium alloy plates subjected to ballistic impact, *European Journal of Mechanics A/Solids* 55 (2016) 221–233.
- [23] M. L. Wilkins, Calculation of elastic-plastic flow, Technical report UCRL-7322, Lawrence radiation Laboratory, University of California. Livermore, California April 19.
- [24] R. P. R. Cardoso, J. W. Yoon, Stress integration method for a nonlinear kinematic/isotropic hardening model and its characterization based on polycrystal plasticity, *International Journal of Plasticity* 25 (2009) 1684–1710.
- [25] R. de Borst, P. H. Feenstra, Studies in anisotropic plasticity with reference to the Hill criterion, *International Journal for Numerical Methods in Engineering* 29 (1990) 315–336.
- [26] J. C. Simo, R. L. Taylor, A Return Mapping Algorithm for Plane Stress Elastoplasticity, *International Journal for Numerical Methods in Engineering* 22 (1986) 649–670.
- [27] V. Aune, E. Fagerholt, K. Hauge, M. Langseth, T. Børvik, Experimental study on the response of aluminium and steel plates subjected to free airblast loading, *International Journal of Impact Engineering* 90 (2016) 106–121. doi: 10.1016/j.ijimpeng.2015.11.017.
- [28] F. Casadei, V. Aune, G. Valsamos, M. Larcher, Testing of the Johnson-Cook material model VPJC in EUROPLEXUS, European Commission, Joint Research Centre, Technical Report, JRC 98848 EUR 27594 (OPOCE LB-NA-27594-EN-N) (2015) 1–62. doi:10.2788/02760.
- [29] V. Aune, G. Valsamos, F. Casadei, M. Larcher, M. Langseth, T. Børvik, Numerical study on the structural response of blast-loaded thin aluminium and steel plates, Submitted for journal publication (2016).
- [30] G. R. Johnson, W. H. Cook, Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures, *Engineering Fracture Mechanics* 21 (I) (1985) 31–48.

## Appendix A. Cutting plane algorithm by using a linear Taylor-expansion

A linear Taylor-expansion around  $f_{n+1}(\boldsymbol{\sigma}, \mathbf{X}) = f_{n+1}(\boldsymbol{\sigma}, \sigma_y, v_p(\dot{p}), \Gamma_p(T_n))$  reads

$$f_{n+1}^{i+1} = f_{n+1}^i + \left. \frac{\partial f}{\partial \boldsymbol{\sigma}} \right|_{n+1}^i \Delta \boldsymbol{\sigma}_{n+1} + \left. \frac{\partial f}{\partial \mathbf{X}} \right|_{n+1}^i \Delta \mathbf{X}_{n+1} \quad (\text{A.1})$$

which may be written in terms of the internal variables  $\mathbf{X}$  as

$$f_{n+1}^{i+1} = f_{n+1}^i + \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \boldsymbol{\sigma}_{n+1} + \frac{\partial f}{\partial \sigma_{y,n+1}} \delta \sigma_{y,n+1} + \frac{\partial f}{\partial v_{p,n+1}} \delta v_{p,n+1} = 0 \quad (\text{A.2})$$

where the flow stress  $\sigma_{y,n+1}$  and the following expressions are chosen as the internal variables  $\mathbf{X}$  in the Taylor-expansion

$$v_p(\Delta \lambda_{n+1}) = \left( 1 + \frac{\Delta \lambda_{n+1}}{\dot{p}_0 \Delta t_{n+1}} \right)^C \quad \Gamma_p(T_n) = \left( 1 - \left[ \frac{T_n - T_r}{T_m - T_r} \right]^m \right) \quad (\text{A.3})$$

Introducing the partial derivatives of the internal variables Eq. (A.2) reads

$$f_{n+1}^{i+1} = f_{n+1}^i + \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \underbrace{\delta \boldsymbol{\sigma}_{n+1}}_{-C \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \lambda_{n+1}^{i+1}} + \frac{\partial f}{\partial \sigma_{y,n+1}} \underbrace{\delta \sigma_{y,n+1}}_{(-1)v_{p,n+1} \Gamma_{p,n} \frac{\partial \sigma_{y,n+1}}{\partial \lambda_{n+1}} \delta \lambda_{n+1}^{i+1}} + \frac{\partial f}{\partial v_{p,n+1}} \underbrace{\delta v_{p,n+1}}_{(-1)\sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \dot{p}_{n+1}} \delta \dot{p}_{n+1}^{i+1}} = 0 \quad (\text{A.4})$$

$$f_{n+1}^{i+1} = f_{n+1}^i - \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \lambda_{n+1}^{i+1} - v_{p,n+1} \Gamma_{p,n} \frac{\partial \sigma_{y,n+1}}{\partial \lambda_{n+1}} \delta \lambda_{n+1}^{i+1} - \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \dot{\lambda}_{n+1}} \delta \dot{\lambda}_{n+1}^{i+1} = 0 \quad (\text{A.5})$$

$$f_{n+1}^{i+1} = f_{n+1}^i - \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \lambda_{n+1}^{i+1} - v_{p,n+1} \Gamma_{p,n} H_{R,n+1} \delta \lambda_{n+1}^{i+1} - \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \dot{\lambda}_{n+1}} \delta \dot{\lambda}_{n+1}^{i+1} = 0 \quad (\text{A.6})$$

Moreover, using  $\dot{p} = \dot{\lambda} = \frac{\Delta \lambda}{\Delta t}$  the internal variable controlling the viscoplasticity  $v_p(\dot{p})$  may be expressed as  $v_p(\Delta \lambda)$  and Eq. (A.6) may be written as

$$f_{n+1}^{i+1} = f_{n+1}^i - \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \delta \lambda_{n+1}^{i+1} - v_{p,n+1} \Gamma_{p,n} H_{R,n+1} \delta \lambda_{n+1}^{i+1} - \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \Delta \lambda_{n+1}} \delta \lambda_{n+1}^{i+1} = 0 \quad (\text{A.7})$$

Finally, the incremental change  $\delta \lambda$  in the incremental plastic multiplier  $\Delta \lambda$  is then expressed as

$$\delta \lambda_{n+1}^{i+1} = \frac{f_{n+1}^i}{\frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} \mathbf{C} \frac{\partial f}{\partial \boldsymbol{\sigma}_{n+1}} + v_{p,n+1} \Gamma_{p,n} H_{R,n+1} + \sigma_{y,n+1} \Gamma_{p,n} \frac{\partial v_{p,n+1}}{\partial \Delta \lambda_{n+1}}} \quad (\text{A.8})$$

where

$$\frac{\partial v_p(\Delta \lambda_{n+1})}{\partial \Delta \lambda} = \frac{C}{\dot{p}_0 \Delta t} \left( 1 + \frac{\Delta \lambda_{n+1}}{\dot{p}_0 \Delta t} \right)^{C-1} \quad (\text{A.9})$$

$$H_{R,n+1} = \frac{dR_{n+1}}{dp_{n+1}} = \sum_{k=1}^2 Q_k C_k e^{(-C_k \lambda_{n+1})} \quad (\text{A.10})$$

$$\left\{ \frac{\partial f}{\partial \boldsymbol{\sigma}} \right\}^T [\mathbf{C}] \left\{ \frac{\partial f}{\partial \boldsymbol{\sigma}} \right\} \quad (\text{A.11})$$

$$[\mathbf{C}] = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1112} & C_{1121} & C_{1123} & C_{1132} & C_{1113} & C_{1131} \\ C_{2211} & C_{2222} & C_{2233} & C_{2212} & C_{2221} & C_{2223} & C_{2232} & C_{2213} & C_{2231} \\ C_{3311} & C_{3322} & C_{3333} & C_{3312} & C_{3321} & C_{3323} & C_{3332} & C_{3313} & C_{3331} \\ C_{1211} & C_{1222} & C_{1233} & C_{1212} & C_{1221} & C_{1223} & C_{1232} & C_{1213} & C_{1231} \\ C_{2111} & C_{2122} & C_{2133} & C_{2112} & C_{2121} & C_{2123} & C_{2132} & C_{2113} & C_{2131} \\ C_{2311} & C_{2322} & C_{2333} & C_{2312} & C_{2321} & C_{2323} & C_{2332} & C_{2313} & C_{2331} \\ C_{3211} & C_{3222} & C_{3233} & C_{3212} & C_{3221} & C_{3223} & C_{3232} & C_{3213} & C_{3231} \\ C_{1311} & C_{1322} & C_{1333} & C_{1312} & C_{1321} & C_{1323} & C_{1332} & C_{1313} & C_{1331} \\ C_{3111} & C_{3122} & C_{3133} & C_{3112} & C_{3121} & C_{3123} & C_{3132} & C_{3113} & C_{3131} \end{bmatrix} \quad (\text{A.12})$$

$$\left\{ \frac{\partial f}{\partial \sigma} \right\} = \begin{bmatrix} \frac{\partial f}{\partial \sigma_{11}^1} \\ \frac{\partial f}{\partial \sigma_{22}^2} \\ \frac{\partial f}{\partial \sigma_{33}^3} \\ \frac{\partial f}{\partial \sigma_{12}^2} \\ \frac{\partial f}{\partial \sigma_{21}^1} \\ \frac{\partial f}{\partial \sigma_{23}^3} \\ \frac{\partial f}{\partial \sigma_{32}^2} \\ \frac{\partial f}{\partial \sigma_{13}^3} \\ \frac{\partial f}{\partial \sigma_{31}^1} \end{bmatrix} = \begin{bmatrix} f_{,11} \\ f_{,22} \\ f_{,33} \\ f_{,12} \\ f_{,21} \\ f_{,23} \\ f_{,32} \\ f_{,13} \\ f_{,31} \end{bmatrix} \quad (\text{A.13})$$

$$\left\{ \frac{\partial f}{\partial \sigma} \right\}^T [\mathbf{C}] = \begin{bmatrix} f_{,11}C_{1111} + f_{,22}C_{2211} + f_{,33}C_{3311} + f_{,12}C_{1211} + f_{,21}C_{2111} + f_{,23}C_{2311} + f_{,32}C_{3211} + f_{,13}C_{1311} + f_{,31}C_{3111} \\ f_{,11}C_{1122} + f_{,22}C_{2222} + f_{,33}C_{3322} + f_{,12}C_{1222} + f_{,21}C_{2122} + f_{,23}C_{2322} + f_{,32}C_{3222} + f_{,13}C_{1322} + f_{,31}C_{3122} \\ f_{,11}C_{1133} + f_{,22}C_{2233} + f_{,33}C_{3333} + f_{,12}C_{1233} + f_{,21}C_{2133} + f_{,23}C_{2333} + f_{,32}C_{3233} + f_{,13}C_{1333} + f_{,31}C_{3133} \\ f_{,11}C_{1112} + f_{,22}C_{2212} + f_{,33}C_{3312} + f_{,12}C_{1212} + f_{,21}C_{2112} + f_{,23}C_{2312} + f_{,32}C_{3212} + f_{,13}C_{1312} + f_{,31}C_{3112} \\ f_{,11}C_{1121} + f_{,22}C_{2221} + f_{,33}C_{3321} + f_{,12}C_{1221} + f_{,21}C_{2121} + f_{,23}C_{2321} + f_{,32}C_{3221} + f_{,13}C_{1321} + f_{,31}C_{3121} \\ f_{,11}C_{1123} + f_{,22}C_{2223} + f_{,33}C_{3323} + f_{,12}C_{1223} + f_{,21}C_{2123} + f_{,23}C_{2323} + f_{,32}C_{3223} + f_{,13}C_{1323} + f_{,31}C_{3123} \\ f_{,11}C_{1132} + f_{,22}C_{2232} + f_{,33}C_{3332} + f_{,12}C_{1232} + f_{,21}C_{2132} + f_{,23}C_{2332} + f_{,32}C_{3232} + f_{,13}C_{1332} + f_{,31}C_{3132} \\ f_{,11}C_{1113} + f_{,22}C_{2213} + f_{,33}C_{3313} + f_{,12}C_{1213} + f_{,21}C_{2113} + f_{,23}C_{2313} + f_{,32}C_{3213} + f_{,13}C_{1313} + f_{,31}C_{3113} \\ f_{,11}C_{1131} + f_{,22}C_{2231} + f_{,33}C_{3331} + f_{,12}C_{1231} + f_{,21}C_{2131} + f_{,23}C_{2331} + f_{,32}C_{3231} + f_{,13}C_{1331} + f_{,31}C_{3131} \end{bmatrix} \quad (\text{A.14})$$

$$\left\{ \frac{\partial f}{\partial \sigma} \right\}^T [\mathbf{C}] \left\{ \frac{\partial f}{\partial \sigma} \right\} = (f_{,11}C_{1111} + f_{,22}C_{2211} + f_{,33}C_{3311} + f_{,12}C_{1211} + f_{,21}C_{2111} + f_{,23}C_{2311} + f_{,32}C_{3211} + f_{,13}C_{1311} + f_{,31}C_{3111})f_{,11} \\ + (f_{,11}C_{1122} + f_{,22}C_{2222} + f_{,33}C_{3322} + f_{,12}C_{1222} + f_{,21}C_{2122} + f_{,23}C_{2322} + f_{,32}C_{3222} + f_{,13}C_{1322} + f_{,31}C_{3122})f_{,22} \\ + (f_{,11}C_{1133} + f_{,22}C_{2233} + f_{,33}C_{3333} + f_{,12}C_{1233} + f_{,21}C_{2133} + f_{,23}C_{2333} + f_{,32}C_{3233} + f_{,13}C_{1333} + f_{,31}C_{3133})f_{,33} \\ + (f_{,11}C_{1112} + f_{,22}C_{2212} + f_{,33}C_{3312} + f_{,12}C_{1212} + f_{,21}C_{2112} + f_{,23}C_{2312} + f_{,32}C_{3212} + f_{,13}C_{1312} + f_{,31}C_{3112})f_{,12} \\ + (f_{,11}C_{1121} + f_{,22}C_{2221} + f_{,33}C_{3321} + f_{,12}C_{1221} + f_{,21}C_{2121} + f_{,23}C_{2321} + f_{,32}C_{3221} + f_{,13}C_{1321} + f_{,31}C_{3121})f_{,21} \\ + (f_{,11}C_{1123} + f_{,22}C_{2223} + f_{,33}C_{3323} + f_{,12}C_{1223} + f_{,21}C_{2123} + f_{,23}C_{2323} + f_{,32}C_{3223} + f_{,13}C_{1323} + f_{,31}C_{3123})f_{,23} \\ + (f_{,11}C_{1132} + f_{,22}C_{2232} + f_{,33}C_{3332} + f_{,12}C_{1232} + f_{,21}C_{2132} + f_{,23}C_{2332} + f_{,32}C_{3232} + f_{,13}C_{1332} + f_{,31}C_{3132})f_{,32} \\ + (f_{,11}C_{1113} + f_{,22}C_{2213} + f_{,33}C_{3313} + f_{,12}C_{1213} + f_{,21}C_{2113} + f_{,23}C_{2313} + f_{,32}C_{3213} + f_{,13}C_{1313} + f_{,31}C_{3113})f_{,13} \\ + (f_{,11}C_{1131} + f_{,22}C_{2231} + f_{,33}C_{3331} + f_{,12}C_{1231} + f_{,21}C_{2131} + f_{,23}C_{2331} + f_{,32}C_{3231} + f_{,13}C_{1331} + f_{,31}C_{3131})f_{,13} \quad (\text{A.15})$$

$$[\mathbf{C}] = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & 0 & 0 & 0 & 0 & 0 & 0 \\ C_{2211} & C_{2222} & C_{2233} & 0 & 0 & 0 & 0 & 0 & 0 \\ C_{3311} & C_{3322} & C_{3333} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{1212} & C_{1221} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{2112} & C_{2121} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{2323} & C_{2332} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{3223} & C_{3232} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{1313} & C_{1331} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{3113} & C_{3131} \end{bmatrix} \quad (\text{A.16})$$

Due to the symmetry of the stress,  $\sigma_{ij} = \sigma_{ji}$ , it is seen that also

$$C_{ijkl} = C_{jikl} \quad (\text{A.17})$$

Since the strain tensor also is symmetric,  $\varepsilon_{kl} = \varepsilon_{lk}$ , it is also assumed that

$$C_{ijkl} = C_{ijlk} \quad (\text{A.18})$$

and the 9x9 matrix in Eq. (A.16) may be reduced to the following 6x6 matrix

$$[\mathbf{C}] = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & 0 & 0 & 0 \\ C_{2211} & C_{2222} & C_{2233} & 0 & 0 & 0 \\ C_{3311} & C_{3322} & C_{3333} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{1212} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{2323} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{1313} \end{bmatrix} \quad (\text{A.19})$$

Adding major symmetry, i.e.  $C_{ijkl} = C_{klij}$ , to Eq. (A.19) and assuming an isotropic material where all axes are symmetry axes, i.e.  $c1 = C_{1111} = C_{2222} = C_{3333}$ ,  $c4 = C_{1122} = C_{2211} = C_{1133} = C_{3311} = C_{2233} = C_{3322}$  and  $c7 = C_{1212} = C_{2323} = C_{1313}$ . Then, Eq. (A.15) reduces to

$$\begin{aligned} \left\{ \frac{\partial f}{\partial \sigma} \right\}^T [\mathbf{C}] \left\{ \frac{\partial f}{\partial \sigma} \right\} = & \\ & (f_{,11}C_{1111} + f_{,22}C_{2211} + f_{,33}C_{3311})f_{,11} + (f_{,11}C_{1122} + f_{,22}C_{2222} + f_{,33}C_{3322})f_{,22} \\ & + (f_{,11}C_{1133} + f_{,22}C_{2233} + f_{,33}C_{3333})f_{,33} \\ & + (f_{,12}C_{1212} + f_{,21}C_{2112})f_{,12} + (f_{,12}C_{1221} + f_{,21}C_{2121})f_{,21} \\ & + (f_{,23}C_{2323} + f_{,32}C_{3223})f_{,23} + (f_{,23}C_{2332} + f_{,32}C_{3232})f_{,32} \\ & + (f_{,13}C_{1313} + f_{,31}C_{3113})f_{,13} + (f_{,13}C_{1331} + f_{,31}C_{3131})f_{,31} \\ & = c1 (f_{,11}^2 + f_{,22}^2 + f_{,33}^2) + 2c4 (f_{,11}f_{,22} + f_{,22}f_{,33} + f_{,33}f_{,11}) \\ & + 4c7 (f_{,12}^2 + f_{,23}^2 + f_{,31}^2) \end{aligned} \quad (\text{A.20})$$

However, since the strain tensor is symmetric,  $\varepsilon_{kl} = \varepsilon_{lk}$  when  $k \neq l$ , it is seen that the number of independent coefficients is reduced to 3

$$[\mathbf{C}] = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & 0 & 0 & 0 \\ C_{2211} & C_{2222} & C_{2233} & 0 & 0 & 0 \\ C_{3311} & C_{3322} & C_{3333} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{1212} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{2323} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{1313} \end{bmatrix} = \begin{bmatrix} c_1 & c_4 & c_4 & 0 & 0 & 0 \\ c_4 & c_1 & c_4 & 0 & 0 & 0 \\ c_4 & c_4 & c_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_7 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_7 \end{bmatrix} \quad (\text{A.21})$$

where it is used Voigt notation and symmetric stress tensors

$$\begin{aligned} \sigma_4 &= \sigma_{12} = \sigma_{21} = \sigma_7 \\ \sigma_5 &= \sigma_{23} = \sigma_{32} = \sigma_8 \\ \sigma_6 &= \sigma_{31} = \sigma_{13} = \sigma_9 \end{aligned} \quad (\text{A.22})$$

The symmetry of the stress and strain tensors allows us to express them as column matrices (or vectors) in the form

$$[\boldsymbol{\sigma}] = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix} \quad \text{and} \quad [\boldsymbol{\varepsilon}] = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{12} \\ \gamma_{23} \\ \gamma_{31} \end{bmatrix} \quad (\text{A.23})$$

where  $\gamma_{12} = 2\varepsilon_{12}$ ,  $\gamma_{23} = 2\varepsilon_{23}$  and  $\gamma_{31} = 2\varepsilon_{31}$  are the engineering shear strains.

$$[\mathbf{C}] = \begin{bmatrix} c_1 & c_4 & c_4 & 0 & 0 & 0 \\ c_4 & c_1 & c_4 & 0 & 0 & 0 \\ c_4 & c_4 & c_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_7 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_7 \end{bmatrix} \quad (\text{A.24})$$

which implies that Eq. (A.15) may be written on a more compact form, i.e.

$$\begin{aligned} \left\{ \frac{\partial f}{\partial \boldsymbol{\sigma}} \right\}^T [\mathbf{C}] \left\{ \frac{\partial f}{\partial \boldsymbol{\sigma}} \right\} &= \frac{\partial f}{\partial \sigma_i} C_{ij} \frac{\partial f}{\partial \sigma_j} \\ &= c_1 \left[ \left( \frac{\partial f}{\partial \sigma_1} \right)^2 + \left( \frac{\partial f}{\partial \sigma_2} \right)^2 + \left( \frac{\partial f}{\partial \sigma_3} \right)^2 \right] + 2c_4 \left[ \frac{\partial f}{\partial \sigma_1} \frac{\partial f}{\partial \sigma_2} + \frac{\partial f}{\partial \sigma_2} \frac{\partial f}{\partial \sigma_3} + \frac{\partial f}{\partial \sigma_3} \frac{\partial f}{\partial \sigma_1} \right] \\ &\quad + 4c_7 \left[ \left( \frac{\partial f}{\partial \sigma_4} \right)^2 + \left( \frac{\partial f}{\partial \sigma_5} \right)^2 + \left( \frac{\partial f}{\partial \sigma_6} \right)^2 \right] \end{aligned} \quad (\text{A.25})$$



## Appendix B. Radial Return Algorithm

As discussed in Section 5.2, the system of equations in Eq.(84) can be extensively simplified and the return mapping for the von Mises model, with associated flow rule and isotropic hardening, was reduced to a single nonlinear equation with the incremental plastic multiplier  $\Delta\lambda$  as the only unknown. The objective of this section is to give a theoretical understanding of the result  $\varphi = \varphi^{trial} - 3G\Delta\lambda$ , which leads to this special case called the radial return algorithm. It is readily seen that the radial return algorithm reduce the number of equations and is important to make the state-update procedure more computationally efficient, and to improve the performance of the overall finite element scheme. For simplicity only the isotropic strain hardening is considered in the constitutive equations.

Starting with the general elastic predictor plastic corrector approach, the stress state at time  $t_{n+1}$  may be expressed as

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \mathbf{C} : (\Delta\boldsymbol{\varepsilon}_{n+1} - \Delta\boldsymbol{\varepsilon}_{n+1}^p) = \boldsymbol{\sigma}_{n+1}^{trial} - \mathbf{C} : \Delta\boldsymbol{\varepsilon}_{n+1}^p = \boldsymbol{\sigma}_{n+1}^{trial} - \Delta\tilde{\boldsymbol{\sigma}}^p \quad (\text{B.1})$$

where  $\boldsymbol{\sigma}_{n+1}^{trial}$  is the elastic predictor moving the stress outside the yield surface, and the plastic corrector  $\Delta\tilde{\boldsymbol{\sigma}}^p$  returns the stress to the yield surface (Figure B.31).

Introducing the associated flow rule, Eq.(B.1) reads

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} - \frac{3}{2}\Delta\lambda_{n+1}\mathbf{C} : \frac{\boldsymbol{\sigma}'_{n+1}}{\sigma_{eq}} \quad (\text{B.2})$$

Using the relation (Appendix C)

$$\mathbf{C} : \boldsymbol{\sigma}'_{n+1} = 2G\boldsymbol{\sigma}'_{n+1} \quad (\text{B.3})$$

Eq.(B.2) is given by

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} - 3G\frac{\boldsymbol{\sigma}'_{n+1}}{\sigma_{eq}}\Delta\lambda_{n+1} \quad (\text{B.4})$$

In the case of a three-dimensional stress state, the von Mises yield criterion is a circular cylinder in the principal stress space with the deviatoric plane normal to the hydrostatic axis (Figure 4). The von Mises flow vector  $\tilde{\boldsymbol{\varepsilon}}^p$  is purely deviatoric, and the hydrostatic stress  $\boldsymbol{\sigma}_{H,n+1}$  is unaffected by the plastic flow ( $\boldsymbol{\sigma}_{H,n+1}^{trial} = \boldsymbol{\sigma}_{H,n+1}$ ). This implies that the stress update can be formulated more efficiently by splitting the stress into deviatoric and hydrostatic parts, where only the deviatoric stress component need to be included in the return mapping, i.e.

$$\boldsymbol{\sigma}'_{n+1} = \boldsymbol{\sigma}'_{n+1}{}^{trial} - 3G\frac{\boldsymbol{\sigma}'_{n+1}}{\sigma_{eq}}\Delta\lambda_{n+1} = \boldsymbol{\sigma}'_{n+1}{}^{trial} - 3G\frac{\boldsymbol{\sigma}'_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1})}\Delta\lambda_{n+1} \quad (\text{B.5})$$

Rearranging Eq.(B.5) results in an important result, i.e.

$$\boldsymbol{\sigma}'_{n+1} \left( 1 + \frac{3G\Delta\lambda_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1})} \right) = \boldsymbol{\sigma}'_{n+1}{}^{trial} \quad (\text{B.6})$$

which implies that the trial and updated deviatoric stresses are co-linear, i.e.

$$\frac{\boldsymbol{\sigma}'_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1})} = \frac{\boldsymbol{\sigma}'_{n+1}{}^{trial}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})} \quad (\text{B.7})$$

This means that the gradient of the yield surface coincide at the trial and updated stress states (Figure B.31). Substituting the result in Eq.(B.7) into (B.5) results in the following simplification

$$\boldsymbol{\sigma}'_{n+1} = \boldsymbol{\sigma}'_{n+1}{}^{trial} \left( 1 - \frac{3G\Delta\lambda_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})} \right) \quad (\text{B.8})$$

where  $\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) = \sqrt{3J_2(\boldsymbol{\sigma}'_{n+1}{}^{trial})}$  is the elastic trial von Mises equivalent stress.

It is also noted that  $\boldsymbol{\sigma}'_{n+1}{}^{trial}$  is a constant vector in the return mapping, and that the deviatoric stress  $\boldsymbol{\sigma}'_{n+1}$  is a function only dependent of  $\Delta\lambda$  in Eq.(B.8). Furthermore, Eq.(B.8) implies that, in the radial return algorithm with von Mises yielding, the updated deviatoric stress is obtained by scaling the trial deviatoric stress by a factor  $1 - 3G\Delta\lambda_{n+1}/\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})$ . The geometric representation of this update formula in the deviatoric plane is illustrated in Figure B.31.

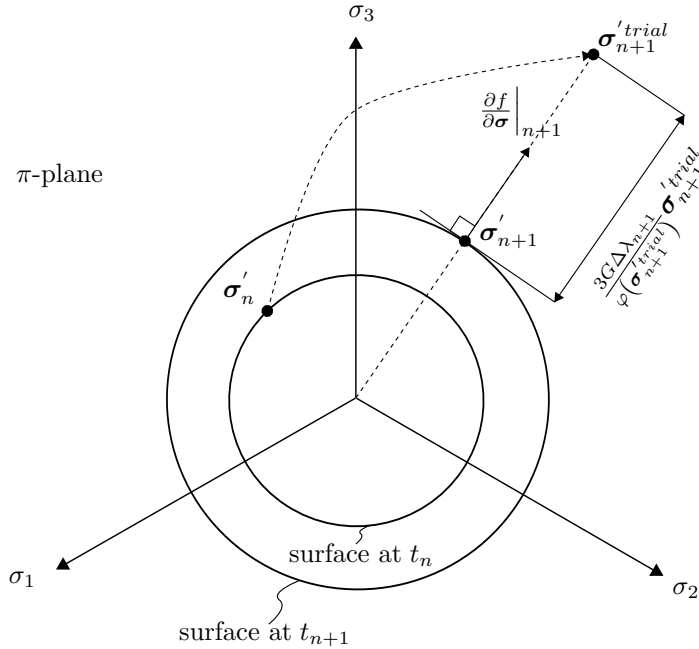


Figure B.31: The implicit elastic predictor plastic corrector radial return map algorithm for von Mises yielding in the deviatoric plane.

Finally, by inserting Eq.(B.8) into the yield criterion, the following nonlinear scalar equation is obtained

$$\begin{aligned} f(\Delta\lambda) &= \varphi(\boldsymbol{\sigma}'_{n+1}) - \sigma_{y,n+1} \\ &= \varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) \left( 1 - \frac{3G\Delta\lambda_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})} \right) - \sigma_{y,n+1} \\ &= \varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) - 3G\Delta\lambda_{n+1} - \sigma_{y,n+1} \\ &= \varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) - 3G\Delta\lambda_{n+1} - (\sigma_{y,n} + H_{R,n+1}\Delta\lambda_{n+1}) = 0 \end{aligned} \quad (\text{B.9})$$

where the incremental plastic multiplier  $\Delta\lambda$  is the only unknown. In Eq.(B.9) it is utilized that  $\varphi(a\boldsymbol{\sigma}') = a\varphi(\boldsymbol{\sigma}')$ , since the function  $\varphi(\boldsymbol{\sigma}')$  is assumed to be a positive homogeneous function of order one of the stress. Thus,

$$\begin{aligned}\varphi(\boldsymbol{\sigma}'_{n+1}) &= \varphi\left(\boldsymbol{\sigma}'_{n+1}{}^{trial} \left[1 - \frac{3G\Delta\lambda_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})}\right]\right) \\ &= \varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) \left(1 - \frac{3G\Delta\lambda_{n+1}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})}\right)\end{aligned}\quad (\text{B.10})$$

By using Eq.(B.9), the incremental plastic multiplier at time  $t_{n+1}$  is expressed as

$$\Delta\lambda_{n+1} = \frac{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) - \sigma_{y,n}}{3G\Delta\lambda_{n+1} + H_{R,n+1}} \quad (\text{B.11})$$

which, in the case of nonlinear hardening, could be solved by a Newton-Raphson method

$$\begin{aligned}\Delta\lambda_{n+1}^{i+1} &= \Delta\lambda_{n+1}^i - \frac{f(\Delta\lambda)}{f'(\Delta\lambda)} \\ &= \Delta\lambda_{n+1}^i - \frac{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial}) - 3G\Delta\lambda_{n+1}^i - \sigma_{y,n+1}}{-3G - H_R}\end{aligned}\quad (\text{B.12})$$

where  $i$  is the local iteration counter. For linear hardening  $\Delta\lambda$  is found directly and no local iterations are necessary.

Finally, the stresses and strains are updated according to Eq.(B.2)

$$\begin{aligned}\boldsymbol{\sigma}_{n+1} &= \boldsymbol{\sigma}_{n+1}{}^{trial} - \frac{3}{2}\Delta\lambda_{n+1}\mathbf{C} : \frac{\boldsymbol{\sigma}'_{n+1}}{\sigma_{eq}} \\ &= \boldsymbol{\sigma}_{n+1}{}^{trial} - \frac{3}{2}\Delta\lambda_{n+1}\mathbf{C} : \frac{\boldsymbol{\sigma}'_{n+1}{}^{trial}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})} \\ p_{n+1} &= p_n + \Delta\lambda_{n+1} \\ \boldsymbol{\varepsilon}_{n+1}^p &= \boldsymbol{\varepsilon}_n^p + \frac{3}{2}\frac{\boldsymbol{\sigma}'_{n+1}{}^{trial}}{\varphi(\boldsymbol{\sigma}'_{n+1}{}^{trial})}\Delta\lambda_{n+1}\end{aligned}\quad (\text{B.13})$$

### Appendix C. Isotropic elastic material

Assuming isotropic linear elastic behavior of the material reduces the stiffness tensor to only depend on the Lamè constants, i.e

$$C_{ijkl} = \lambda_{el}\delta_{ij}\delta_{kl} + \mu_{el}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) \quad (C.1)$$

and the stress is then given by the linear relation

$$\begin{aligned} \sigma_{ij} &= C_{ijkl}\varepsilon_{kl} = \lambda_{el}\delta_{ij}\delta_{kl}\varepsilon_{kl} + \mu_{el}(\delta_{ik}\delta_{jl}\varepsilon_{kl} + \delta_{il}\delta_{jk}\varepsilon_{kl}) \\ &= \lambda_{el}\delta_{ij}\varepsilon_{kk} + \mu_{el}(\varepsilon_{ij} + \varepsilon_{ji}) \\ &= \lambda_{el}\delta_{ij}\varepsilon_{kk} + 2\mu_{el}(\varepsilon_{ij}) \end{aligned} \quad (C.2)$$

where the following relations are used

$$\begin{aligned} \delta_{ik}\delta_{jl}\varepsilon_{kl} &= \delta_{ik}\varepsilon_{jk} = \varepsilon_{ij} \\ \delta_{il}\delta_{jk}\varepsilon_{kl} &= \delta_{il}\varepsilon_{lj} = \varepsilon_{ji} \\ \varepsilon_{ij} &= \varepsilon_{ji} \end{aligned} \quad (C.3)$$

By inserting the expression for the Lamè constants, Eq.(C.1) reads

$$\begin{aligned} C_{ijkl} &= \frac{\nu E}{(1+\nu)(1-2\nu)}\delta_{ij}\delta_{kl} + \frac{E}{2(1+\nu)}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) \\ &= \frac{E}{1+\nu} \left[ \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) + \frac{\nu}{1-2\nu}\delta_{ij}\delta_{kl} \right] \\ &= 2G \left[ \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) + \frac{\nu}{1-2\nu}\delta_{ij}\delta_{kl} \right] \end{aligned} \quad (C.4)$$

and by multiplying the isotropic elasticity tensor in Eq.(C.4) with the deviatoric part of the stress tensor, this product reduces to

$$\begin{aligned} C_{ijkl}\sigma'_{kl} &= 2G \left[ \frac{1}{2}(\delta_{ik}\delta_{jl}\sigma'_{kl} + \delta_{il}\delta_{jk}\sigma'_{kl}) + \frac{\nu}{1-2\nu}\delta_{ij}\delta_{kl}\sigma'_{kl} \right] \\ &= 2G \left[ \frac{1}{2}(\sigma'_{ij} + \sigma'_{ji}) + \frac{\nu}{1-2\nu}\delta_{ij}\sigma'_{kk} \right] \end{aligned} \quad (C.5)$$

Utilizing the symmetry of the stress tensor, i.e.  $\sigma'_{ij} = \sigma'_{ji}$ , and the definition of the deviatoric stress

$$\sigma'_{ij} = \sigma_{ij} - \frac{1}{3}\sigma_{kk}\delta_{ij}, \quad \sigma'_{ii} = \sigma_{ii} - \frac{1}{3}\sigma_{kk}\delta_{ii} = 0 \quad (C.6)$$

Eq.(C.5) reduces to the following expression

$$C_{ijkl}\sigma'_{kl} = 2G\sigma'_{ij} \quad (C.7)$$

which is a useful relation in the radial return map algorithm.

## Appendix D. Two-dimensional stress analysis

This section is an extended version of Section 3 and shows useful relationships when dealing with the particular case of plane stress and three shear stress components. Plane stress is already defined in Section 3 as the state of stress in which the normal stress  $\sigma_{33}$  and the shear stresses  $\sigma_{13,23}$  are assumed to be zero [11]. From a practical point of view this assumption is typically introduced in the analysis of bodies in which one of the dimensions (i.e. the thickness) is much smaller than the others and is subjected to loads that result in dominant stresses perpendicular to the thickness direction [5]. However, the same assumption may also be used when only the normal stress  $\sigma_{33}$  is assumed to be zero. Thus, all the shear components are considered in the calculation.

### Appendix D.1. Two-dimensional theory of elasticity

Recall that, in the particular case of plane stress, the linear elastic stress-strain compliance relationship for an isotropic material can be written as a system of equations in matrix form given as

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{23} \\ 2\varepsilon_{31} \end{bmatrix} = \mathbf{S}\boldsymbol{\sigma} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ 0 \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix} \quad (\text{D.1})$$

The normal stress  $\sigma_{33}$  which is assumed to be zero in the stress vector indicates that its associated column in the compliance matrix  $\mathbf{S}$  can be neglected (i.e. column 3). Furthermore, if we neglect the row associated with the strain components in the thickness direction  $\varepsilon_{i3}$  ( $i = 1, 2, 3$ ), the stress-strain compliance relationship reduces to a  $5 \times 5$  matrix

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \\ 2\varepsilon_{23} \\ 2\varepsilon_{31} \end{bmatrix} = \mathbf{S}\boldsymbol{\sigma} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & 0 & 0 & 0 \\ -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix} \quad (\text{D.2})$$

Then, the elasticity matrix  $\mathbf{C}$  for plane stress is found by inverting the plane stress compliance matrix  $\mathbf{S}$  and given by Hooke's law as

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix} = \mathbf{C}\boldsymbol{\varepsilon} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 & 0 & 0 \\ \nu & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{1-\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1-\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \\ 2\varepsilon_{23} \\ 2\varepsilon_{31} \end{bmatrix} \quad (\text{D.3})$$

It is important to note that the elasticity matrix for plane stress is not found by removing columns and rows from the general isotropic elasticity matrix in Eq. (C.1).

It should also be noted that the procedure in finding the principal stresses  $\sigma_i$  of the stress tensor  $\boldsymbol{\sigma}$  are now determined by using Eq. (57), i.e.

$$\sigma_i = \sigma'_i + \sigma_H = \sigma'_i + \frac{1}{3}\text{tr}(\boldsymbol{\sigma}) \quad (\text{D.4})$$

where the major principal stress  $\sigma_1$  is used in the Cockcroft-Latham failure criterion in Eq. (46). Thus, the procedure in Section 3.1 is not applicable when using all the shear components.

## Appendix E. Implementation of Return Map Algorithm in EUROPLEXUS

This appendix shows a somewhat more detailed version of the numerical scheme given in Section 5.1. Within the time increment  $\Delta t_{n+1}$  the scheme may then be explained as follows:

1. Set the initial values of internal variables to the converged values of the previous step at  $t_n$ , and calculate the speed of sound. Note that the number of shear stress components ( $ITAU$ ) and plane stress conditions ( $IPLANC$ ) (provided by the element routine) are used to *SELECT* the appropriate stress state in the material routine.

### Governing equations

Define useful parameters related to the theory of elasticity

$$T1 = 1 + \nu, \quad T2 = 1 - 2\nu, \quad T3 = 1 - \nu$$

Compute the speed of sound based on the stress state, i.e.,  
IF 3D stress state THEN

$$CSON = \sqrt{\frac{E(1-\nu)}{\rho(1+\nu)(1-2\nu)}}$$

ELSEIF plane stress THEN

$$CSON = \sqrt{\frac{E}{\rho(1-\nu^2)}}$$

ELSEIF uniaxial stress THEN

$$CSON = \sqrt{\frac{E}{\rho}}$$

### Implementation in EUROPLEXUS

```

SUBROUTINE VPJC (ITAU, IPLANC, PROPS, STRAINING, STRESSOLD,
> STRAIN, STATE, STRESSNEW, CSON, FAIL_FLAG)
* sound speed
  T1 = R1 + POISS
  T2 = R1 - R2*POISS
  T3 = R1 - POISS
*
  SELECT CASE (IPLANC)
  CASE (0) ! NO PLANE STRESS CONDITION
    CSON = SQRT (YOUNG*T3 / (RHO*T1*T2))
  CASE (1) ! PLANE STRESS CONDITION (SIG(3)=0)
    CSON = SQRT (YOUNG / (RHO*T1*T3))
  CASE (2) ! UNIAXIAL STRESS CONDITION (SIG(2)=SIG(3)=0)
    CSON = SQRT (YOUNG / RHO)
  CASE DEFAULT ! CHECK THIS ONLY HERE
    CALL ERRMSG ('VPJC', 'WRONG IPLANC')
    STOP 'VPJC - WRONG IPLANC'
  END SELECT
*

```

2. Compute the elasticity tensor.

### Governing equations

IF a 3D stress state is used THEN use Eqs. (3)-(4)

$$\mathbf{C} = \lambda_{el} \mathbf{I} \otimes \mathbf{I} + 2\mu_{el} \mathbf{I} = \begin{bmatrix} C_1 & C_4 & C_5 & 0 & 0 & 0 \\ C_5 & C_2 & C_4 & 0 & 0 & 0 \\ C_5 & C_6 & C_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_7 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_8 & 0 \\ 0 & 0 & 0 & 0 & 0 & C_9 \end{bmatrix}$$

$$C_1 = C_2 = C_3 = \frac{(1-\nu)E}{(1+\nu)(1-2\nu)}, \quad C_4 = C_5 = C_6 = \frac{\nu C_1}{1-\nu},$$

$$C_7 = C_8 = C_9 = \frac{1}{2} \frac{E}{1+\nu}$$

ELSEIF a 2D stress state is used THEN use Eq. (60)

$$\mathbf{C} = \begin{bmatrix} C_1 & C_4 & 0 \\ C_5 & C_2 & 0 \\ 0 & 0 & C_7 \end{bmatrix}$$

$$C_1 = C_2 = \frac{E}{1-\nu^2}, \quad C_4 = C_5 = \nu C_1, \quad C_7 = \frac{1}{2} \frac{E}{1+\nu}$$

ELSE a 1D stress state is used THEN use

$$\mathbf{C} = \begin{bmatrix} C_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad C_1 = E$$

### Implementation in EUROPLEXUS

```

* compute the elasticity tensor
SELECT CASE (IPLANC)
CASE (0) ! NO PLANE STRESS CONDITION
  C1 = YOUNG*T3 / (T1*T2)
  C4 = POISS*C1 / T3
CASE (1) ! PLANE STRESS CONDITION (SIG(3)=0)
  C1 = YOUNG / (T1*T3)
  C4 = POISS*C1
CASE (2) ! UNIAXIAL STRESS CONDITION (SIG(2)=SIG(3)=0)
  C1 = YOUNG
  C4 = R0 ! THESE VALUES HAVE TO BE CHECKED
END SELECT
*
  C2 = C1
  C3 = C1
  C5 = C4
  C6 = C4
  C7 = R5*YOUNG / T1
  C8 = C7
  C9 = C7
*

```

3. & 4. Use the total stain increment  $\Delta \epsilon_{n+1}$  from the global equilibrium to compute the elastic predictor. Then, compute the von Mises equivalent stress in terms of the elastic trial stress and Eq. (35). Note that Eqs.(35) and (34) are used to calculate the gradients, i.e.,  $\sigma_{12} = \sigma_{21}$ ,  $\sigma_{23} = \sigma_{32}$  and  $\sigma_{13} = \sigma_{31}$ .

### Governing equations

$$\sigma_{11,n+1}^{trial} = \sigma_{11,n} + C_1 \Delta \epsilon_{11,n+1} + C_4 \Delta \epsilon_{22,n+1} + C_5 \Delta \epsilon_{33,n+1}$$

IF a 3D stress state is used THEN

$$\sigma_{22,n+1}^{trial} = \sigma_{22,n} + C_4 \Delta \epsilon_{11,n+1} + C_2 \Delta \epsilon_{22,n+1} + C_6 \Delta \epsilon_{33,n+1}$$

$$\sigma_{33,n+1}^{trial} = \sigma_{33,n} + C_5 \Delta \epsilon_{11,n+1} + C_6 \Delta \epsilon_{22,n+1} + C_3 \Delta \epsilon_{33,n+1}$$

$$\sigma_{12,n+1}^{trial} = \sigma_{12,n} + C_7 \Delta \epsilon_{12,n+1}$$

$$\sigma_{23,n+1}^{trial} = \sigma_{23,n} + C_8 \Delta \epsilon_{23,n+1}$$

$$\sigma_{31,n+1}^{trial} = \sigma_{31,n} + C_9 \Delta \epsilon_{31,n+1}$$

$$\frac{3}{2} \sigma_{n+1}^{trial} : \sigma_{n+1}^{trial} = (\sigma_{11,n+1}^{trial})^2 + (\sigma_{22,n+1}^{trial})^2 + (\sigma_{33,n+1}^{trial})^2 - \sigma_{11,n+1}^{trial} \sigma_{22,n+1}^{trial} - \sigma_{33,n+1}^{trial} \sigma_{11,n+1}^{trial} + 3((\sigma_{12,n+1}^{trial})^2 + (\sigma_{23,n+1}^{trial})^2 + (\sigma_{31,n+1}^{trial})^2)$$

ELSEIF a 2D plane stress state is used THEN

$$\sigma_{22,n+1}^{trial} = \sigma_{22,n} + C_4 \Delta \epsilon_{11,n+1} + C_2 \Delta \epsilon_{22,n+1} + C_6 \Delta \epsilon_{33,n+1}$$

$$\sigma_{33,n+1}^{trial} = 0$$

$$\sigma_{12,n+1}^{trial} = \sigma_{12,n} + C_7 \Delta \epsilon_{12,n+1}$$

$$\frac{3}{2} \sigma_{n+1}^{trial} : \sigma_{n+1}^{trial} = (\sigma_{11,n+1}^{trial})^2 + (\sigma_{22,n+1}^{trial})^2 - \sigma_{11,n+1}^{trial} \sigma_{22,n+1}^{trial} + 3((\sigma_{12,n+1}^{trial})^2)$$

ELSE a 1D stress state is used and

$$\sigma_{22,n+1}^{trial} = 0$$

$$\sigma_{33,n+1}^{trial} = 0$$

$$\frac{3}{2} \sigma_{n+1}^{trial} : \sigma_{n+1}^{trial} = (\sigma_{11,n+1}^{trial})^2$$

END IF

$$\sigma_{eq,n+1}^{trial} = \varphi(\sigma_{n+1}^{trial}) = \sqrt{\frac{3}{2} \sigma_{n+1}^{trial} : \sigma_{n+1}^{trial}}$$

### Implementation in EUROPLEXUS

```
* compute the elastic trial stress and the trial equivalent stress
*
  STRESS(1) = STRESSOLD(1) + C1*STRAININC(1) + C4*STRAININC(2)
  > + C5*STRAININC(3)
*
  SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
  CASE (0) ! NO PLANE STRESS CONDITION
  STRESS(2) = STRESSOLD(2) + C4*STRAININC(1) + C2*STRAININC(2)
  > + C6*STRAININC(3)
  STRESS(3) = STRESSOLD(3) + C5*STRAININC(1) + C6*STRAININC(2)
  > + C3*STRAININC(3)
  PHITRIAL = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
  > + STRESS(3)*STRESS(3) - STRESS(1)*STRESS(2)
  > - STRESS(2)*STRESS(3) - STRESS(3)*STRESS(1)
  CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
  STRESS(2) = STRESSOLD(2) + C4*STRAININC(1) + C2*STRAININC(2)
  > + C6*STRAININC(3)
  STRESS(3) = 0
  PHITRIAL = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
  > - STRESS(1)*STRESS(2)
  CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
  STRESS(2) = 0
  STRESS(3) = 0
  PHITRIAL = STRESS(1)*STRESS(1)
  END SELECT
*
  SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
  CASE (0) ! NO SHEAR STRESS IS PRESENT
  CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
  STRESS(4) = STRESSOLD(4) + C7*STRAININC(4) ! THE GAMMAS ARE USED
  PHITRIAL = PHITRIAL + R3*(STRESS(4)*STRESS(4))
  CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
  STRESS(4) = STRESSOLD(4) + C7*STRAININC(4) ! SHEAR STRAINS
  STRESS(5) = STRESSOLD(5) + C8*STRAININC(5) ! ARE THE GAMMAS, NOT
  STRESS(6) = STRESSOLD(6) + C9*STRAININC(6) ! THE EPSILONS
  PHITRIAL = PHITRIAL + R3*(STRESS(4)*STRESS(4)
  > + STRESS(5)*STRESS(5)
  > + STRESS(6)*STRESS(6))
  CASE DEFAULT ! WE CHECK THIS ONLY IN THE FIRST SELECT CASE ON ITAU
  CALL ERRMSG ('VPEC', 'WRONG ITAU')
  STOP 'VPEC - WRONG ITAU'
  END SELECT
*
  PHITRIAL = SQRT (PHITRIAL)
*
```

5. Define initial values of internal variables, to be used in return mapping algorithm, to the converged values of the previous step at  $t_n$ . Check if temperature softening is included by the user (i.e.,  $m \neq 0$  in material input), compute a value for the yield function  $f(\sigma_{n+1}^{trial})$  and set the incremental plastic multiplier  $\Delta\lambda_{n+1}$  equal to zero.

### Governing equations

$$\begin{aligned} p_n &= \text{STATE}(3) \\ R_n &= \text{STATE}(6) \\ \sigma_{y,n} &= A + R_n \\ W_n &= \text{STATE}(14) \end{aligned}$$

IF temperature softening is specified in material input ( $m \neq 0$ ) THEN

$$\begin{aligned} T^* &= \frac{T_n - T_r}{T_m - T_r} \\ \Gamma_p(T_n) &= (1 - T^{*m}) \\ f_{n+1} &= \sigma_{eq,n+1}^{trial} - \sigma_{y,n} \Gamma_p(T_n) \end{aligned}$$

ELSE  $m = 0$  and the CPU time can be reduced by skipping the softening term

$$f_{n+1} = \sigma_{eq,n+1}^{trial} - \sigma_{y,n}$$

ENDIF

$$\begin{aligned} \Delta\lambda_{n+1} &= 0 \\ \delta\lambda_{n+1} &= 0 \end{aligned}$$

IF  $\varphi_{n+1}^{trial} = 0$  THEN

$$\varphi_{n+1} = 1$$

ELSE

$$\varphi_{n+1} = \varphi_{n+1}^{trial} = \sigma_{eq,n+1}^{trial}$$

ENDIF

### Implementation in EUROPLEXUS

```

-----
PN = STATE(3)
RP = STATE(6)
SIGMAY = SY + RP
*
* import old value damage variables
WE = STATE(14)
*
IF (TEMP_SOFT) THEN
  TSTAR = (T-TR) / (TM-TR)
  TSOFT = R1 - TSTAR**M
  F = PHITRIAL - SIGMAY*TSOFT
ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
  F = PHITRIAL - SIGMAY
ENDIF
DLAMBDA = RO
DDLAMBDA = RO
IF (PHITRIAL == RO) THEN
  PHI = R1
ELSE
  PHI = PHITRIAL
ENDIF
*

```



6. Check for plastic admissibility.

Governing equations

IF  $f(\sigma_{n+1}^{trial}) > 0$  THEN apply return mapping by using the cutting plane algorithm ( $SOLU = 1$ ) to find the incremental change in the plastic multiplier  $\Delta\lambda_{n+1}$ .  
Superscript  $i$  denotes the local iteration counter.

Implementation in EUROPLEXUS

```

IF (F > RO) THEN
*
* plastic step: apply return mapping
* the cutting plane method uses values from the previous
* iteration to update the plastic parameter, dlambda.
* thus, all subsequent values (until dlambda) are related to
* the previous iteration, "1".
-----
P = PN
LOOP_NR : DO NRITER = 1, MXITER

```

- (i) Compute the normal  $\mathbf{n}_{n+1}$  to the yield surface based on the initial (i.e., trial) or previous stress configuration. Note that Eqs.(35) and (34) are used to calculate the gradients, i.e.,  $\sigma_{12} = \sigma_{21}$ ,  $\sigma_{23} = \sigma_{32}$  and  $\sigma_{13} = \sigma_{31}$ .

Governing equations

IF a 3D stress state is used THEN

$$n_{11} = \partial f / \partial \sigma_{11} = (\sigma_{11} - 0.5(\sigma_{22} + \sigma_{33})) / \sigma_{eq}$$

$$n_{22} = \partial f / \partial \sigma_{22} = (\sigma_{22} - 0.5(\sigma_{33} + \sigma_{11})) / \sigma_{eq}$$

$$n_{33} = \partial f / \partial \sigma_{33} = (\sigma_{33} - 0.5(\sigma_{11} + \sigma_{22})) / \sigma_{eq}$$

$$n_{12} = \partial f / \partial \sigma_{12} = 3\sigma_{12} / \sigma_{eq}$$

$$n_{23} = \partial f / \partial \sigma_{23} = 3\sigma_{23} / \sigma_{eq}$$

$$n_{31} = \partial f / \partial \sigma_{31} = 3\sigma_{31} / \sigma_{eq}$$

ELSEIF a plane stress state is used THEN

$$n_{11} = \partial f / \partial \sigma_{11} = (\sigma_{11} - 0.5\sigma_{22}) / \sigma_{eq}$$

$$n_{22} = \partial f / \partial \sigma_{22} = (\sigma_{22} - 0.5\sigma_{11}) / \sigma_{eq}$$

$$n_{33} = 0$$

$$n_{12} = \partial f / \partial \sigma_{12} = 3\sigma_{12} / \sigma_{eq}$$

ELSE a 1D stress state is used and

$$n_{11} = \partial f / \partial \sigma_{11} = \sigma_{11} / \sigma_{eq}$$

Implementation in EUROPLEXUS

```

* compute the respective gradients of the von mises yield surface
RNV(2:6) = RO ! INITIALIZE TO ZERO "OPTIONAL" COMPONENTS
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
RNV(1) = (STRESS(1) - R5*(STRESS(2) + STRESS(3))) / PHI
RNV(2) = (STRESS(2) - R5*(STRESS(3) + STRESS(1))) / PHI
RNV(3) = (STRESS(3) - R5*(STRESS(1) + STRESS(2))) / PHI
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
RNV(1) = (STRESS(1) - R5*STRESS(2)) / PHI
RNV(2) = (STRESS(2) - R5*STRESS(1)) / PHI
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
RNV(1) = STRESS(1) / PHI ! TO BE CHECKED ...
END SELECT
*
SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
RNV(4) = R3*STRESS(4) / PHI
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
RNV(4) = R3*STRESS(4) / PHI
RNV(5) = R3*STRESS(5) / PHI
RNV(6) = R3*STRESS(6) / PHI
END SELECT

```

- (ii) Compute hardening modulus  $H_{R,n+1}$  according to Eq. (42)

Governing equations

$$H_{R,n+1}^i = Q_1 C_1 \exp(-C_1 p_{n+1}^i) + Q_2 C_2 \exp(-C_2 p_{n+1}^i)$$

Implementation in EUROPLEXUS

```

* compute hardening modulus
HR = QR1*CR1*EXP(-CR1*P) + QR2*CR2*EXP(-CR2*P)

```

(iii) Compute denominator in Eq. (90) using Eqs. (91), (92), (93) and (98)

**Governing equations**

*IF* a 3D stress state is used *THEN*

$$\mathbf{n}^T \mathbf{Cn} = C_1 [(n_{11})^2 + (n_{22})^2 + (n_{33})^2] + 2C_4 [n_{11}n_{22} + n_{22}n_{33} + n_{33}n_{11}] + 4C_7 [(n_{12})^2 + (n_{23})^2 + (n_{31})^2]$$

*ELSEIF* a plane stress state is used *THEN*

$$\mathbf{n}^T \mathbf{Cn} = C_1 [(n_{11})^2 + (n_{22})^2] + 2C_4 [n_{11}n_{22}] + 4C_7 [(n_{12})^2]$$

*ELSE* a 1D stress state is used and

$$\mathbf{n}^T \mathbf{Cn} = C_1 (n_{11})^2$$

$$v_{p,n+1}^i = \left( 1 + \frac{\Delta\lambda_{n+1}^i}{\dot{p}_0 \Delta t_{n+1}} \right)^C$$

$$\frac{\partial v_{p,n+1}^i}{\partial \Delta\lambda_{n+1}^i} = \frac{C}{\dot{p}_0 \Delta t_{n+1}} \left( 1 + \frac{\Delta\lambda_{n+1}^i}{\dot{p}_0 \Delta t_{n+1}} \right)^{C-1}$$

*IF* temperature softening is activated ( $m \neq 0$ ) *THEN*

$$\Gamma_p(T_n) = \left( 1 - \left[ \frac{T_n - T_r}{T_m - T_r} \right]^m \right)$$

$$\text{DENOM} = \mathbf{n}^T \mathbf{Cn} + v_{p,n+1}^i \Gamma_{p,n} H_{R,n+1}^i + \sigma_{y,n+1}^i \Gamma_{p,n} \frac{\partial v_{p,n+1}^i}{\partial \Delta\lambda_{n+1}^i}$$

*ELSE*  $m = 0$  and skip the softening term

$$\text{DENOM} = \mathbf{n}^T \mathbf{Cn} + v_{p,n+1}^i H_{R,n+1}^i + \sigma_{y,n+1}^i \frac{\partial v_{p,n+1}^i}{\partial \Delta\lambda_{n+1}^i}$$

*ENDIF*

**Implementation in EUROPLEXUS**

```

* compute denominator in the residual derivative of dlambda
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
DENOMCP = C1*(RNV(1)*RNV(1) + RNV(2)*RNV(2) + RNV(3)*RNV(3))
+ R2*C4*(RNV(1)*RNV(2) + RNV(2)*RNV(3) + RNV(3)*RNV(1))
>
CASE (1) ! PLANE STRESS CONDITION (RNV(3)=0)
DENOMCP = C1*(RNV(1)*RNV(1) + RNV(2)*RNV(2))
+ R2*C4*(RNV(1)*RNV(2))
>
CASE (2) ! UNIAXIAL STRESS CONDITION (RNV(2)=RNV(3)=0)
DENOMCP = C1*(RNV(1)*RNV(1)) ! TO BE CHECKED ...
END SELECT

*
SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (RNV(4)) IS PRESENT
DENOMCP = DENOMCP + R4*C7*(RNV(4)*RNV(4))
CASE (3) ! ALL THREE SHEAR STRESSES (RNV(4,6)) ARE PRESENT
DENOMCP = DENOMCP + R4*C7*(RNV(4)*RNV(4) + RNV(5)*RNV(5)
+ RNV(6)*RNV(6))
END SELECT

*
DENOMV = (R1 + DLAMBDA / (PDOT*DT))*C
DENOMDV = (C / (PDOT*DT))*(R1 + DLAMBDA / (PDOT*DT))*C - R1
IF (TEMP_SOFT) THEN
TSTAR = (T - TR) / (TM - TR)
TSOFT = R1 - TSTAR*M
DENOM = DENOMCP + HR*DENOMV*TSOFT + SIGMAY*DENOMDV*TSOFT
ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
DENOM = DENOMCP + HR*DENOMV + SIGMAY*DENOMDV
ENDIF

```

(iv) Compute incremental change in the plastic multiplier  $\delta\lambda_{n+1}^{i+1}$  in Eq. (90) and update variable  $\Delta\lambda_{n+1}^{i+1}$

**Governing equations**

$$\delta\lambda_{n+1}^{i+1} = \frac{f_{n+1}^i}{\text{DENOM}}$$

$$\Delta\lambda_{n+1}^{i+1} = \Delta\lambda_{n+1}^i + \delta\lambda_{n+1}^{i+1}$$

**Implementation in EUROPLEXUS**

```

* compute residual increment dlambda and update variable dlambda
DDLAMBDA = F / DENOM
DLAMBDA = DLAMBDA + DDLAMBDA

```

(v) Update internal variables dependent on  $\Delta\lambda_{n+1}^{i+1}$

### Governing equations

$$\begin{aligned} p_{n+1}^{i+1} &= p_{n+1}^i + \delta\lambda_{n+1}^{i+1} \\ R_{n+1}^{i+1} &= R_n + H_{R,n+1}^i \Delta\lambda_{n+1}^{i+1} \\ \sigma_{y,n+1}^{i+1} &= A + R_{n+1}^{i+1} \end{aligned}$$

### Implementation in EUROPLEXUS

```
* update internal variables dependent on dlambda
P = P + DDLAMBDA
R = RP + HR*DLAMBDA
SIGMAY = SY + R
```

(vi) Update stress components using the elastic Hooke's law according to Eq. (83). Note that the material routine uses a fully vectorized representation of the stress and strain vector, i.e.,  $\boldsymbol{\sigma} = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T$  and  $\boldsymbol{\varepsilon} = [\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, \gamma_{12}, \gamma_{23}, \gamma_{31}]^T = [\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, 2\varepsilon_{12}, 2\varepsilon_{23}, 2\varepsilon_{31}]^T$

### Governing equations

*IF* a 3D stress state is used *THEN*

$$\begin{aligned} \sigma_{11,n+1}^{i+1} &= \sigma_{11,n+1}^i - \delta\lambda_{n+1}^{i+1} (C_1 n_{11} + C_4 n_{22} + C_5 n_{33}) \\ \sigma_{22,n+1}^{i+1} &= \sigma_{22,n+1}^i - \delta\lambda_{n+1}^{i+1} (C_4 n_{11} + C_2 n_{22} + C_6 n_{33}) \\ \sigma_{33,n+1}^{i+1} &= \sigma_{33,n+1}^i - \delta\lambda_{n+1}^{i+1} (C_5 n_{11} + C_6 n_{22} + C_3 n_{33}) \\ \sigma_{12,n+1}^{i+1} &= \sigma_{12,n+1}^i - \delta\lambda_{n+1}^{i+1} C_7 n_{12} \\ \sigma_{23,n+1}^{i+1} &= \sigma_{23,n+1}^i - \delta\lambda_{n+1}^{i+1} C_8 n_{23} \\ \sigma_{31,n+1}^{i+1} &= \sigma_{31,n+1}^i - \delta\lambda_{n+1}^{i+1} C_9 n_{31} \end{aligned}$$

*ELSEIF* a plane stress state is used *THEN*

$$\begin{aligned} \sigma_{11,n+1}^{i+1} &= \sigma_{11,n+1}^i - \delta\lambda_{n+1}^{i+1} (C_1 n_{11} + C_4 n_{22} + C_5 n_{33}) \\ \sigma_{22,n+1}^{i+1} &= \sigma_{22,n+1}^i - \delta\lambda_{n+1}^{i+1} (C_4 n_{11} + C_2 n_{22} + C_6 n_{33}) \\ \sigma_{33,n+1}^{i+1} &= 0 \\ \sigma_{12,n+1}^{i+1} &= \sigma_{12,n+1}^i - \delta\lambda_{n+1}^{i+1} C_7 n_{12} \end{aligned}$$

*ELSE* a 1D stress state is used and

$$\sigma_{11,n+1}^{i+1} = \sigma_{11,n+1}^i - \delta\lambda_{n+1}^{i+1} C_1 n_{11}$$

### Implementation in EUROPLEXUS

```
* update stress components using the elastic (hooke's) law
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
  STRESS(1) = STRESS(1) -
    > DDLAMBDA*(C1*RVN(1) + C4*RVN(2) + C5*RVN(3))
  STRESS(2) = STRESS(2) -
    > DDLAMBDA*(C4*RVN(1) + C2*RVN(2) + C6*RVN(3))
  STRESS(3) = STRESS(3) -
    > DDLAMBDA*(C5*RVN(1) + C6*RVN(2) + C3*RVN(3))
CASE (1) ! PLANE STRESS CONDITION (RVN(3)=0)
  STRESS(1) = STRESS(1) -
    > DDLAMBDA*(C1*RVN(1) + C4*RVN(2))
  STRESS(2) = STRESS(2) -
    > DDLAMBDA*(C4*RVN(1) + C2*RVN(2))
CASE (2) ! UNIAXIAL STRESS CONDITION (RVN(2)=RVN(3)=0)
  STRESS(1) = STRESS(1) - DDLAMBDA*C1*RVN(1) ! TO BE CHECKED ..
END SELECT

*
SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (RVN(4)) IS PRESENT
  STRESS(4) = STRESS(4) - DDLAMBDA*C7*RVN(4)
CASE (3) ! ALL THREE SHEAR STRESSES (RVN(4:6)) ARE PRESENT
  STRESS(4) = STRESS(4) - DDLAMBDA*C7*RVN(4)
  STRESS(5) = STRESS(5) - DDLAMBDA*C8*RVN(5)
  STRESS(6) = STRESS(6) - DDLAMBDA*C9*RVN(6)
END SELECT

*
```

(vii) Compute the updated von Mises equivalent stress using Eq. (35)

### Governing equations

*IF* a 3D stress state is used *THEN*

$$\frac{3}{2}\sigma'_{n+1} : \sigma'_{n+1} = (\sigma_{11,n+1}^{i+1})^2 + (\sigma_{22,n+1}^{i+1})^2 + (\sigma_{33,n+1}^{i+1})^2 - \sigma_{11,n+1}^{i+1}\sigma_{22,n+1}^{i+1} + \sigma_{22,n+1}^{i+1}\sigma_{33,n+1}^{i+1} - \sigma_{33,n+1}^{i+1}\sigma_{11,n+1}^{i+1} + 3((\sigma_{12,n+1}^{i+1})^2 + (\sigma_{23,n+1}^{i+1})^2 + (\sigma_{31,n+1}^{i+1})^2)$$

*ELSEIF* a plane stress state is used *THEN*

$$\frac{3}{2}\sigma'_{n+1} : \sigma'_{n+1} = (\sigma_{11,n+1}^{i+1})^2 + (\sigma_{22,n+1}^{i+1})^2 - \sigma_{11,n+1}^{i+1}\sigma_{22,n+1}^{i+1} + 3((\sigma_{12,n+1}^{i+1})^2)$$

*ELSE* a 1D stress state is used and

$$\frac{3}{2}\sigma'_{n+1} : \sigma'_{n+1} = (\sigma_{11,n+1}^{i+1})^2$$

*ENDIF*

$$\sigma_{eq,n+1}^{i+1} = \varphi(\sigma_{n+1}^{i+1}) = \sqrt{\frac{3}{2}\sigma'_{n+1} : \sigma'_{n+1}}$$

### Implementation in EUROPLEXUS

```

* compute updated equivalent stress
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
  PHI = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
  > + STRESS(3)*STRESS(3) - STRESS(1)*STRESS(2)
  > - STRESS(2)*STRESS(3) - STRESS(3)*STRESS(1)
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
  PHI = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
  > - STRESS(1)*STRESS(2)
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
  PHI = STRESS(1)*STRESS(1)
END SELECT

*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
  PHI = PHI + R3*(STRESS(4)*STRESS(4))
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4;6)) ARE PRESENT
  PHI = PHI + R3*(STRESS(4)*STRESS(4)
  > + STRESS(5)*STRESS(5)
  > + STRESS(6)*STRESS(6))
END SELECT

*
PHI = SQRT (PHI)

```

(viii) Update the yield function in Eq. (84) according to the new stress state computed in (vi) and (vii). Check for convergence.

### Governing equations

*IF* temperature softening ( $m \neq 0$ ) *THEN*

$$f_{v,n+1}^{i+1} = \varphi_{n+1}^{i+1} - \sigma_{y,n+1}^{i+1}(\bullet)$$

$$(\bullet) = \left(1 + \frac{\Delta\lambda_{n+1}^{i+1}}{\dot{p}_0\Delta t_{n+1}}\right)^C \left(1 - \left[\frac{T_n - T_r}{T_m - T_r}\right]^m\right)$$

$$RESNOR = \left| \frac{f_v(\Delta\lambda_{n+1}^{i+1})}{\sigma_{y,n+1}^{i+1} v_{p,n+1}^{i+1} \Gamma_p} \right|$$

*ELSE*  $m = 0$  and skip the softening term

$$f_{v,n+1}^{i+1} = \varphi_{n+1}^{i+1} - \sigma_{y,n+1}^{i+1}(\bullet)$$

$$(\bullet) = \left(1 + \frac{\Delta\lambda_{n+1}^{i+1}}{\dot{p}_0\Delta t_{n+1}}\right)^C$$

$$RESNOR = \left| \frac{f_v(\Delta\lambda_{n+1}^{i+1})}{\sigma_{y,n+1}^{i+1} v_{p,n+1}^{i+1}} \right|$$

*ENDIF*

*IF*  $RESNOR < TOL$  *THEN* continue to 7.

*ELSE*  $i = i + 1$  and *GO TO* (i)

### Implementation in EUROPLEXUS

```

* compute yield function and new residual
IF (TEMP_SOFT) THEN
  TSTAR = (T - TR) / (TM - TR) ! THIS RECALCULATION OF TSTAR
  TSOFT = R1 - TSTAR**M ! AND TSOFT SEEMS REDUNDANT HERE
  F = PHI - SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C)*TSOFT
*
* check convergence
RESNOR = F / (SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C)*TSOFT)
RESNOR = ABS (RESNOR)
ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
  F = PHI - SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C)
* check convergence
RESNOR = F / (SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C))
RESNOR = ABS (RESNOR)
ENDIF
*
IF (RESNOR <= TOL) THEN

```

7. Update the temperature according to the low coupling approach using Eq. (45) and erode element if  $T > T_m$

#### Governing equations

$$T_{n+1} = T_n + \frac{\chi \sigma_{eq,n+1} \Delta \lambda_{n+1}}{\rho C_T}$$

#### Implementation in EUROPLEXUS

```
* update temperature (only if temperature softening is activated)
IF (TEMP_SOFT) THEN
  T = T + TQ*PHI*DLAMBDA / (RHO*CP)
  IF (T >= TM) THEN ! CHECK IF MELTING OCCURS
    STATE(13) = TM
    D = DC ! this will cause the GP to fail (see 990)
    GO TO 990
  ENDDIF
ENDIF
ENDIF
```

8. Compute the hydrostatic stress according to Eq. (32)

#### Governing equations

$$\sigma_{H,n+1} = \frac{1}{3} \text{tr}(\sigma_{n+1}) = \frac{1}{3} (\sigma_{11,n+1} + \sigma_{22,n+1} + \sigma_{33,n+1})$$

#### Implementation in EUROPLEXUS

```
* compute hydrostatic stress based on new stress state,
SIGMAH = STRESS(1)
SELECT CASE (IPLANC)
CASE (0) ! NO PLANE STRESS CONDITION
  SIGMAH = SIGMAH + STRESS(2) + STRESS(3)
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
  SIGMAH = SIGMAH + STRESS(2)
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
  END SELECT
*
SIGMAH = SIGMAH / R3
```

9. Computing the principal stresses  $\sigma_{i,n+1}$  based on the principal deviatoric stresses  $\sigma'_{i,n+1}$  as shown in Eqs. (55) and (57)

#### Governing equations

$$\begin{aligned} \sigma'_{11,n+1} &= \sigma_{11,n+1} - \sigma_{H,n+1} \\ \sigma'_{22,n+1} &= \sigma_{22,n+1} - \sigma_{H,n+1} \\ \sigma'_{33,n+1} &= \sigma_{33,n+1} - \sigma_{H,n+1} \\ J_{2,n+1} &= \frac{1}{2} \left( (\sigma'_{11,n+1})^2 + (\sigma'_{22,n+1})^2 + (\sigma'_{33,n+1})^2 \right) \\ &\quad + (\sigma_{12,n+1})^2 + (\sigma_{23,n+1})^2 + (\sigma_{31,n+1})^2 \\ J_{3,n+1} &= \sigma'_{11,n+1} (\sigma_{23,n+1})^2 + \sigma'_{22,n+1} (\sigma_{31,n+1})^2 \\ &\quad + \sigma'_{33,n+1} (\sigma_{12,n+1})^2 - \sigma'_{11,n+1} \sigma'_{22,n+1} + \sigma'_{33,n+1} \\ &\quad - 2\sigma_{12,n+1} \sigma_{23,n+1} \sigma_{31,n+1} \\ \theta &= \arccos \left( \frac{J_{3,n+1}}{2\sqrt{J_{2,n+1}^3/27}} \right), \quad 0 \leq \theta \leq \pi \\ \sigma'_{1,n+1} &= 2\sqrt{\frac{J_{2,n+1}}{3}} \cos \frac{\theta}{3} \\ \sigma'_{2,n+1} &= 2\sqrt{\frac{J_{2,n+1}}{3}} \cos \left( \frac{\theta}{3} - \frac{2\pi}{3} \right) \\ \sigma'_{3,n+1} &= 2\sqrt{\frac{J_{2,n+1}}{3}} \cos \left( \frac{\theta}{3} + \frac{2\pi}{3} \right) \\ \sigma_{1,n+1} &= \sigma_{H,n+1} + \sigma'_{1,n+1} \\ \sigma_{2,n+1} &= \sigma_{H,n+1} + \sigma'_{2,n+1} \\ \sigma_{3,n+1} &= \sigma_{H,n+1} + \sigma'_{3,n+1} \end{aligned}$$

#### Implementation in EUROPLEXUS

```
* 3d stress state and compute deviatoric stress
S1 = STRESS(1) - SIGMAH
S2 = STRESS(2) - SIGMAH
S3 = STRESS(3) - SIGMAH
*
* finding principal stress based on the deviatoric stress and invariants
A1 = R5*(S1*S1 + S2*S2 + S3*S3)
*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0) ! NO SHEAR STRESS IS PRESENT
  A2 = -S1*S2*S3 ! DETERMINANT OF THE DEVIATORIC STRESS TENSOR
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
  A1 = A1 + STRESS(4)*STRESS(4)
  A2 = S3*STRESS(4)*STRESS(4) - S1*S2*S3
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
  A1 = A1 +
> STRESS(4)*STRESS(4) + STRESS(5)*STRESS(5) +
> STRESS(6)*STRESS(6)
  A2 = S1*STRESS(5)*STRESS(5) +
> S2*STRESS(6)*STRESS(6) +
> S3*STRESS(4)*STRESS(4) -
> S1*S2*S3 - R2*STRESS(4)*STRESS(5)*STRESS(6)
  END SELECT
*
A3 = -SQRT (R27 / A1)*A2*R5 / A1
A3 = SIGN (MIN (ABS (A3), R1), A3)
A4 = ACOS (A3) / R3
*
* compute (chronological) principal stresses
AUX = SQRT (A1 / R3)
S1 = SIGMAH + R2*AUX*COS(A4)
S2 = SIGMAH + R2*AUX*COS(A4-R2*PI/R3) ! only for postprocessing
S3 = SIGMAH + R2*AUX*COS(A4+R2*PI/R3) ! only for postprocessing
*
```

10. Compute the damage parameter  $D_{n+1}$  by using Eq. (46), i.e. integrating the major principal stress during the entire equivalent plastic strain path

#### Governing equations

$$W_{n+1} = W_n + \max(\sigma_{1,n+1}, 0) \Delta\lambda_{n+1}$$

$$D_{n+1} = \frac{W_{n+1}}{W_c}$$

#### Implementation in EUROPLEXUS

```
*
WE = WE + MAX (S1, RO)*DLAMBDA
D = WE / WC
*
```

11. Update state variables to be returned to the element routine in EUROPLEXUS and check for element erosion.

#### Governing equations

```
STATE(1) =  $\sigma_{H,n+1}$ 
STATE(2) =  $\sigma_{eq,n+1}$ 
STATE(3) =  $p_{n+1}$ 
STATE(4) =  $\sigma_{eq,n+1}^{trial}$ 
STATE(5) =  $f_{n+1}$ 
STATE(6) =  $R_{n+1}$ 
STATE(7) =  $\delta\lambda_{n+1}$ 
STATE(8) =  $\Delta\lambda_{n+1}$ 
STATE(9) = NRITER
STATE(10) =  $\dot{p}_{n+1} = \Delta\lambda_{n+1} / \Delta t_{n+1}$ 
STATE(11) =  $D_{n+1}$ 
STATE(13) =  $T_{n+1}$ 
STATE(14) =  $W_{n+1}$ 
STATE(15) =  $CSON_{n+1}$ 
STATE(16) =  $\sigma_{1,n+1}$ 
STATE(17) =  $\sigma_{2,n+1}$ 
STATE(18) =  $\sigma_{3,n+1}$ 
IF  $D_{n+1} > D_C$  THEN
failure and the material point stiffness is deleted.
STATE(12) = 1
ELSE
no failure and the material point stiffness should
not be deleted.
STATE(12) = 0
```

#### Implementation in EUROPLEXUS

```
* update state variables before exit
STATE(1) = SIGMAH
STATE(2) = PHI
STATE(3) = P
STATE(4) = PHITRIAL
STATE(5) = F
STATE(6) = R
STATE(7) = DDLAMBDA
STATE(8) = DLAMBDA
STATE(9) = NRITER
STATE(10) = DLAMBDA / DT
STATE(11) = D

*
STATE(13) = T
STATE(14) = WE
STATE(15) = CSON

*
STATE(16) = S1
STATE(17) = S2
STATE(18) = S3

*
* decide if material point stiffness is zero, i.e. failed
-----
990 IF (D >= DC) THEN
*
* the current gp was already failed or is failing now :
* state(1:10,12,13), the relevant components
* of stressnew and cson are set to 0.
* the total strains (strain) are not set to 0 so they continue to be
* updated as long as the element is not eroded, i.e. as long
* as the element has some virgin gauss points.
*
SELECT CASE (ITAU)
CASE (0)
STRESSNEW(1:3) = RO
CASE (1)
STRESSNEW(1:4) = RO
CASE (2)
STRESSNEW(1:5) = RO
CASE (3)
STRESSNEW(1:6) = RO
END SELECT
STATE(1:10) = RO
STATE(12) = RO
* state(11) (damage d) keeps last value it had before failure
STATE(12) = RO
* state(13) (absolute temperature t) keeps last value before failure
* state(14) (damage accumulation we) keeps last value before failure
CSON = RO
STATE(15) = CSON
ELSE
*
* no failure
*
STATE(12) = R1
*
```

12. Update the actual total strain  $\varepsilon_{33,n+1}$  and incremental strain  $\Delta\varepsilon_{33,n+1}$  through the thickness according to Eq. (69) and return this to the element routine in EUROPLEXUS

### Governing equations

*IF* plane stress *THEN* use Eq.(69) to compute actual strain through thickness

$$\Delta\varepsilon_{33,n+1} = \frac{1-2\nu}{E}(\Delta\sigma_{11,n+1} + \Delta\sigma_{22,n+1}) - (\Delta\varepsilon_{11,n+1} + \Delta\varepsilon_{22,n+1})$$

$$\varepsilon_{33,n+1} = \varepsilon_{33,n} + \Delta\varepsilon_{33,n+1}$$

### Implementation in EUROPLEXUS

```
* strain increment(s) and total strain(s) corresponding to
* the imposed zero stress condition(s)
*
  SELECT CASE (IPLANC)
  CASE (0) ! NO PLANE STRESS CONDITION
  CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
* store provisional strain increment
  AUX3 = STRAININC(3)
* compute and update the actual strain increment
  FACTOR = T2 / YOUNG
  STRAININC(3) = FACTOR*(STRESSNEW(1) - STRESSOLD(1)
  > + STRESSNEW(2) - STRESSOLD(2))
  > - STRAININC(1) - STRAININC(2)
* update the total strain
  STRAIN(3) = STRAIN(3) - AUX3 + STRAININC(3)
  CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
  AUX2 = STRAININC(2)
  AUX3 = STRAININC(3)
  FACTOR = T2 / YOUNG
  STRAININC(2) = 0.5D0*(FACTOR*(STRESSNEW(1) - STRESSOLD(1))
  > - STRAININC(1))
  > STRAININC(3) = STRAININC(2)
  STRAIN(2) = STRAIN(2) - AUX2 + STRAININC(2)
  STRAIN(3) = STRAIN(3) - AUX3 + STRAININC(3)
  END SELECT
ENDIF
*
* erosion (optional)
*
IF (L_EROSION) THEN
  IF (D >= DC) FAIL_FLAG = .TRUE.
ENDIF
*
END SUBROUTINE VPJC
```

In EPX, this is done for each material integration point in the FE assembly. The stresses are updated in the material interface using a fully vectorized version of the forward Euler integration algorithm and a two-state architecture where the initial values at  $t_n$  are stored in the *old* arrays and the new values at  $t_{n+1}$  must be updated and stored in the *new* arrays returned to the global FE analysis. The *VPJC* material routine is valid for 1D, 2D and 3D stress states, i.e., for bar, shell, solid, axisymmetric, plane strain and plane stress elements. The stresses for 3D elements are stored similar to that of symmetric tensors as  $\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T$ , and plane stress, axisymmetric and plane strain elements are stored as  $\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}]^T$ . The deformation gradient and strains are stored similarly to the stresses, however, one should be aware of that the shear strain is stored as engineering shear strains, e.g.  $\gamma_{12} = 2\varepsilon_{12}$ .

## Appendix F. Material routine in EUROPLEXUS

The constitutive model of elastic-thermoviscoplasticity and ductile failure is implemented in EUROPLEXUS (EPX) [7] using a fully vectorized version of the forward Euler integration algorithm and a special case of the backward Euler integration algorithm as presented in this report. The FORTRAN implementation of the numerical scheme in Appendix E is presented in this section. The model is applicable for 1D, 2D and 3D stress states, i.e., for 1D, 2D and 3D stress states, i.e., for bar, shell, solid, axisymmetric, plane strain and plane stress elements. The model is coupled with an element deletion option available in EPX by introducing a state variable controlling the element erosion, i.e., as the damage variable reaches its critical value  $D_C$  the stiffness in the material point is removed. An element will be removed from the mesh when a user-defined fraction of the material points in the element are deleted. This is specified in the input file for EPX using the *EROS* directive where the user should specify a (proportional) number between 0 and 1. The default value is 1 and indicates that an element is eroded when all its Gauss points have failed, while the value 0 indicate that an element is eroded when any one of its Gauss points fails.

The procedure described in Section 5.1 and Appendix E is formulated in terms of the material subroutine *VPJC*, which communicates with the element routine (or even a higher-order routine) in EPX. During an

FE analysis the material subroutine *VPJC* is called by the chosen element routine for each material point at each increment. When the material subroutine is called, it is provided with the state at the start of the increment (stress and solution-dependent state variables at  $t_n$ ). Based on the geometrical calculations performed by the element routine it is also provided with the strain increment  $\Delta\epsilon$  and total strains  $\epsilon$  at the beginning and the end of the increment. Thus, the *VPJC* uses a two-state architecture for the stress tensor  $\sigma$  where the initial values at  $t_n$  are stored in the *old* arrays and the new values at  $t_{n+1}$  must be updated and stored in the *new* arrays returned to the global analysis in EPX. The stresses for 3D elements are stored similar to that of symmetric tensors as  $\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T$ , while plane stress, plane strain and axisymmetric elements are stored as  $\sigma = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}]^T$ . The deformation gradient and strains are stored similarly to the stresses, however, one should be aware of that the shear strain is stored as tensor shear strains, e.g.  $\epsilon_{12} = \frac{1}{2}\gamma_{12}$ . That is,  $\epsilon = [\epsilon_{11}, \epsilon_{22}, \epsilon_{33}, 2\epsilon_{12}, 2\epsilon_{23}, 2\epsilon_{31}]^T$  and  $\epsilon = [\epsilon_{11}, \epsilon_{22}, \epsilon_{33}, 2\epsilon_{12}]^T$  in the 3D and 2D cases, respectively.

The coding of material routines in EUROPLEXUS follow FORTRAN 90 conventions. In addition to the vectorized interface, it is essential that all variables are defined and initialized properly using a routine that reads the material constants from the input data set. Moreover, a conversion routine is used and called by the element routine just before and after the material routine, to make the arrangement of stress and strain in *VPJC* compatible with the one used in the element routine. Thus, the conversion routine ensures that all tensors have the same number of components and the same organization for a given element type.

The FORTRAN source of the material routine for EPX is listed below.

```

C M_MATERIAL_VPJC          TOUS GVALSAMOS 16/06/06 20:09:13 #3077
  MODULE M_MATERIAL_VPJC
*
* material of type vpjc
*
* the vpjc subroutine can be used for all stress states, by providing
* appropriate values of itau and iplanc
*
  USE M_MATERIALS_ARRAY
*
  IMPLICIT NONE
  SAVE
*
  PRIVATE
  PUBLIC ::
  > MAVPJC, ! PROPERTIES READING ROUTINE
  > VPJC    ! MATERIAL ROUTINE
*
  CONTAINS
=====
  SUBROUTINE MAVPJC (INUMLDC)
-----
*
*                                vegard aune, folco casadei 11-14
* material "vpjc" (visco-plastic johnson-cook)
-----
* mat : index of material read in the input file
*
* organization of newmat%matval: (material properties)
* -----
*   matval(1) - rho      - material density
*   matval(2) - young    - elastic modulus
*   matval(3) - nu       - poisson ratio
*   matval(4) - elas     - initial yield stress
*   matval(5) - tol      - tolerance newton iteration
*   matval(6) - mxit     - maximum number of iterations
*   matval(7) - qr1      - asymptote first term voce hardening
*   matval(8) - cr1      - hardening parameter first term voce
*   matval(9) - qr2      - asymptote 2nd term voce hardening
*   matval(10) - cr2     - hardening parameter 2nd term voce
*   matval(11) - pdot    - reference strain-rate
*   matval(12) - c       - hardening parameter visco term
*   matval(13) - tq      - taylor-quinney coefficient
*   matval(14) - cp      - specific heat capacity solid material
*   matval(15) - tr      - room temperature

```



```

*   matval(16)- tm      - melting temperature
*   matval(17)- m      - hardening parameter temperature term
*   matval(18)- dc     - upper limit of d when softening
*   matval(19)- wc     - cockcroft-latham failure criterion
*   matval(20)- solu   - 1=cutting plane (default) explicit
*                       requires N-R iterations,
*                       2=radial return (implicit) requires N-R iterations
*
*   INCLUDE 'NONE.INC'
*
*   INCLUDE 'CARMA.INC'
*   INCLUDE 'CONSTA.INC'
*   INCLUDE 'CUNIT.INC'
*   INCLUDE 'REDCOM.INC'
*
*   INTEGER, INTENT(IN) :: INUMLDC
*
* local variables
*
*   INTEGER, PARAMETER :: IFIX = 4
*
*   INTEGER, PARAMETER :: NMOT = 20
*
*   CHARACTER*4 :: MOT(NMOT)
*   INTEGER :: KOPT(NMOT), I
*
*   DATA MOT /'RO ', 'YOUN', 'NU ', 'ELAS', 'TOL ',
>   'MXIT', 'QR1 ', 'CR1 ', 'QR2 ', 'CR2 ',
>   'PDOT', 'C ', 'TQ ', 'CP ', 'TR ',
>   'TM ', 'M ', 'DC ', 'WC ', 'SOLU'/
*
* the syntax of this material is particularly simple, because
* it contains only a sequence of keywords, each followed by a value.
* the first 4 keywords are mandatory, the others are optional
*
* create new material structure newmat of the needed length
*
*   CALL CREATE_MATERIAL (NMOT, 0, 0)
*   NEWMAT%NAME = 'VPJC : VISCO-PLASTIC JOHNSON-COOK'
*   NEWMAT%TYPE = 120 ! TYPE OF THE VPJC MATERIAL
*   NEWMAT%NUMLDC = INUMLDC
*   NEWMAT%LGECCR = LGECCR(NEWMAT%TYPE)
*
*   CALL LIRVAL (NMOT, MOT, NEWMAT%MATVAL, KOPT)
*
* input data complete ? (all keywords are mandatory except tol, mxit, tr, solu)
*
*   DO I = 1, NMOT
*     SELECT CASE (I)
*       CASE (5, 6, 15, 20) ! TOL, MXIT, TR, SOLU ARE OPTIONAL
*       CASE DEFAULT      ! ALL OTHERS ARE MANDATORY
*         IF (KOPT(I) == 0) GO TO 9
*       END SELECT
*     END DO
*
* set default values in the optional properties which have not been read
*
*   IF (KOPT(5) == 0) NEWMAT%MATVAL(5) = 1.E-4 ! TOL
*   IF (KOPT(6) == 0) NEWMAT%MATVAL(6) = 20 ! MXIT
*   IF (KOPT(15) == 0) NEWMAT%MATVAL(15) = 293.DO ! TR
*   IF (KOPT(20) == 0) NEWMAT%MATVAL(20) = 1.DO ! SOLU
*
* printouts
*
*   WRITE(TAPOUT,1000) NEWMAT%NUMLDC
*
*   WRITE(TAPOUT,1001) NEWMAT%MATVAL(1)
*   WRITE(TAPOUT,1002) NEWMAT%MATVAL(2)
*   WRITE(TAPOUT,1003) NEWMAT%MATVAL(3)
*   WRITE(TAPOUT,1004) NEWMAT%MATVAL(4)
*   WRITE(TAPOUT,1005) NEWMAT%MATVAL(5)
*   WRITE(TAPOUT,1006) NINT (NEWMAT%MATVAL(6))
*   WRITE(TAPOUT,1007) NEWMAT%MATVAL(7)
*   WRITE(TAPOUT,1008) NEWMAT%MATVAL(8)
*   WRITE(TAPOUT,1009) NEWMAT%MATVAL(9)
*   WRITE(TAPOUT,1010) NEWMAT%MATVAL(10)
*   WRITE(TAPOUT,1011) NEWMAT%MATVAL(11)

```



```

> R27=27.D0
* defining the number pi
  REAL(8), PARAMETER :: PI = 3.1415926535D0
*
  INTEGER :: MXITER, WRITER, SOLU
  LOGICAL :: TEMP_SOFT ! IS TEMPERATURE SOFTENING ACTIVATED (IS M /= 0?)
  REAL(8) :: TSTAR, TSOFT
*
* input parameters to generalize the present material routine:
* itau : number of shear stress components (stress(4),stress(5),stress(6))
*      0 = no shear stress is present
*      1 = only tau_xy (stress(4)) is present
*      3 = all three shear stresses (stress(4:6)) are present
* iplanc : 0 = no plane stress condition
*          1 = plane stress condition (sigma_z=stress(3)=0)
*          2 = uniaxial stress condition (sigma_y=stress(2)=0 and
*          sigma_z=stress(3)=0)
*
* organization of stress and strain arrays
* -----
* stress(1)- xx : normal components
* stress(2)- yy
* stress(3)- zz
* stress(4)- xy : shear components
* stress(5)- yz
* stress(6)- xz
*
* attention: the shear components of strain and straininc arrays are the
* "engineering" values (gamma), i.e. twice the "tensor" values (epsilon)!
*
* list of material properties sent to the subroutine
* -----
* props(1) - rho - material density
* props(2) - young - elastic modulus
* props(3) - poiss - poisson ratio
* props(4) - sy - initial yield stress
* props(5) - tol - tolerance newton iteration
* props(6) - mxiter - maximum number of iterations
* props(7) - qr1 - asymptote first term voce hardening
* props(8) - cr1 - hardening parameter first term voce
* props(9) - qr2 - asymptote 2nd term voce hardening
* props(10) - cr2 - hardening parameter 2nd term voce
* props(11) - pdot - reference strain-rate
* props(12) - c - hardening parameter visco term
* props(13) - tq - taylor-quinney coefficient
* props(14) - cp - specific heat capacity solid material
* props(15) - tr - room temperature
* props(16) - tm - melting temperature
* props(17) - m - hardening parameter temperature term
* props(18) - dc - upper limit of d when softening
* props(19) - wc - cockcroft-latham failure criterion
* props(20) - solu - 1=cutting plane (default) requires N-R iterations,
*                  2=radial return (direct, no iterations)
*                  (not available in plane or uniaxial stress!)
*
* list of state variables (old and new)
* -----
* description name intent
* -----
* state(1) - hydrostatic pressure (1/3 sigma_kk) sigmah new
* state(2) - von mises equivalent stress phi new
* state(3) - equivalent plastic strain p old+new
* state(4) - elastic trial equivalent stress phitrial new
* state(5) - yield function (should be near zero) f new
* state(6) - total hardening of the material r old+new
* state(7) - change of dlambda over the step ddlambda new
* state(8) - incremental plastic multiplier dlambda new
* state(9) - n of iterations for convergence nriter new
* state(10) - plastic strain rate dlambda / dt new
* state(11) - damage d old+new
* state(12) - failure: 1 = virgin g.p., 0 = failed g.p. fa new
* state(13) - absolute temperature t old+new
* state(14) - cockcroft-latham damage accumulation we old+new
* state(15) - sound speed cson new
* state(16) - first principal stress s1 new
* state(17) - second principal stress s2 new
* state(18) - third principal stress s3 new

```

```

*
* fail_flag : output only!
*           in input it is always .false.
*           in output it is .true. if:
*           - the current Gauss point is failed, **and**
*           - erosion has been activated by the user
*             (i.e. l_erosion == .true.)
*
* NOTE:
* - the material routine is called also for an already failed gp, if the
*   element to which the gp belongs has not been eroded yet!
* - however, the material routine is not called at all for an eroded element!
*
! Thread variable
  INTEGER :: NTH
*
  DC = PROPS(18)
  D = STATE(11)
  T = STATE(13) ! T IS INITIALIZED TO TR IN ECROU.FF
*
* check previous failure of current gp
  IF (D >= DC) THEN
* current gauss point was previously failed, so let's just skip it ...
  GO TO 990
  ENDIF
*
* current gp was not previously failed, but it might fail now ...
*
  CALL PILOT_GET_THREAD_NUM (NTH)
! End thread variable
*
  CALL ELEM_DT1 (IEL(NTH), DT)
*
  RHO      = PROPS(1)
  YOUNG    = PROPS(2)
  POISS    = PROPS(3)
  SY       = PROPS(4)
  TOL      = PROPS(5)
  MXITER   = NINT (PROPS(6))
  QR1      = PROPS(7)
  CR1      = PROPS(8)
  QR2      = PROPS(9)
  CR2      = PROPS(10)
  PDOT     = PROPS(11)
  C        = PROPS(12)
  TQ       = PROPS(13)
  CP       = PROPS(14)
  TR       = PROPS(15)
  TM       = PROPS(16)
  M        = PROPS(17)
  WC       = PROPS(19)
  SOLU     = NINT (PROPS(20))
*
  IF (SOLU == 2) THEN
* radial return is not available in either plane stress or uniaxial stress
  IF (IPLANC > 0) THEN
    CALL ERRMSS ('VPJC', 'SOLU 2 N/A IN PLANE/UNIAXIAL STRESS')
    STOP 'VPJC - SOLU 2 N/A IN PLANE/UNIAXIAL STRESS'
  ENDIF
  ENDIF
*
  TEMP_SOFT = (M /= 0.D0) ! TEMPERATURE SOFTENING IS ACTIVATED ONLY
                    ! IF M /= 0 (EXPONENTIALS ARE EXPENSIVE ...)
*
* sound speed
  T1 = R1 + POISS
  T2 = R1 - R2*POISS
  T3 = R1 - POISS
*
  SELECT CASE (IPLANC)
  CASE (0) ! NO PLANE STRESS CONDITION
    CSON = SQRT (YOUNG*T3 / (RHO*T1*T2))
  CASE (1) ! PLANE STRESS CONDITION (SIG(3)=0)
    CSON = SQRT (YOUNG / (RHO*T1*T3))
  CASE (2) ! UNIAXIAL STRESS CONDITION (SIG(2)=SIG(3)=0)
    CSON = SQRT (YOUNG / RHO)
  CASE DEFAULT ! WE CHECK THIS ONLY IN THE FIRST SELECT CASE ON IPLANC

```

```

        CALL ERRMSS ('VPJC', 'WRONG IPLANC')
        STOP 'VPJC - WRONG IPLANC'
    END SELECT

*
*-----
* state update procedure for the von mises elasto-
* thermoviscoplastic material model with non-linear isotropic
* hardening goverened by the modified johnson-cook model:
* explicit elastic predictor/return mapping algorithm.
*-----
*
* elastic predictor: compute elastic trial stress state
*-----
*
* compute the elasticity tensor
SELECT CASE (IPLANC)
CASE (0)  ! NO PLANE STRESS CONDITION
    C1 = YOUNG*T3 / (T1*T2)
    C4 = POISS*C1 / T3
CASE (1)  ! PLANE STRESS CONDITION (SIG(3)=0)
    C1 = YOUNG / (T1*T3)
    C4 = POISS*C1
CASE (2)  ! UNIAXIAL STRESS CONDITION (SIG(2)=SIG(3)=0)
    C1 = YOUNG
    C4 = R0  ! THESE VALUES HAVE TO BE CHECKED
END SELECT

*
C2 = C1
C3 = C1
C5 = C4
C6 = C4
C7 = R5*YOUNG / T1
C8 = C7
C9 = C7
R3G = R3*C7

*
* compute the elastic trial stress and the elastic trial equivalent stress
*
    STRESS(1) = STRESSOLD(1) + C1*STRAININC(1) + C4*STRAININC(2)
    >          + C5*STRAININC(3)
*
    SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
    CASE (0)  ! NO PLANE STRESS CONDITION
        STRESS(2) = STRESSOLD(2) + C4*STRAININC(1) + C2*STRAININC(2)
    >          + C6*STRAININC(3)
        STRESS(3) = STRESSOLD(3) + C5*STRAININC(1) + C6*STRAININC(2)
    >          + C3*STRAININC(3)
        PHITRIAL = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
    >          + STRESS(3)*STRESS(3) - STRESS(1)*STRESS(2)
    >          - STRESS(2)*STRESS(3) - STRESS(3)*STRESS(1)
    CASE (1)  ! PLANE STRESS CONDITION (STRESS(3)=0)
        STRESS(2) = STRESSOLD(2) + C4*STRAININC(1) + C2*STRAININC(2)
    >          + C6*STRAININC(3)
        STRESS(3) = R0
        PHITRIAL = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
    >          - STRESS(1)*STRESS(2)
    CASE (2)  ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
        STRESS(2) = R0
        STRESS(3) = R0
        PHITRIAL = STRESS(1)*STRESS(1)
    END SELECT

*
    SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
    CASE (0)  ! NO SHEAR STRESS IS PRESENT
    CASE (1)  ! ONLY TAU_XY (STRESS(4)) IS PRESENT
        STRESS(4) = STRESSOLD(4) + C7*STRAININC(4) ! THE GAMMAS ARE USED!
        PHITRIAL = PHITRIAL + R3*(STRESS(4)*STRESS(4))
    CASE (3)  ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
        STRESS(4) = STRESSOLD(4) + C7*STRAININC(4) ! SHEAR STRAIN AND STRAININC
        STRESS(5) = STRESSOLD(5) + C8*STRAININC(5) ! ARE THE GAMMAS, NOT
        STRESS(6) = STRESSOLD(6) + C9*STRAININC(6) ! THE EPSILONS
        PHITRIAL = PHITRIAL + R3*(STRESS(4)*STRESS(4)
    >          + STRESS(5)*STRESS(5)
    >          + STRESS(6)*STRESS(6))
    CASE DEFAULT ! WE CHECK THIS ONLY IN THE FIRST SELECT CASE ON ITAU
        CALL ERRMSS ('VPJC', 'WRONG ITAU')
        STOP 'VPJC - WRONG ITAU'

```

```

END SELECT
*
PHITRIAL = SQRT (PHITRIAL)
*
* check for plastic admissibility
*-----
PN = STATE(3)
RP = STATE(6)
SIGMAY = SY + RP
*
* import old value damage variables
WE = STATE(14)
*
IF (TEMP_SOFT) THEN
TSTAR = (T-TR) / (TM-TR)
TSOFT = R1 - TSTAR**M
F = PHITRIAL - SIGMAY*TSOFT
ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
F = PHITRIAL - SIGMAY
ENDIF
DLAMBDA = R0
DDLAMBDA = R0
IF (PHITRIAL == R0) THEN
PHI = R1
ELSE
PHI = PHITRIAL
ENDIF
*
IF (SOLU == 2) GO TO 200
*
*=====
* solu == 1 : cutting plane solution strategy (default)
*=====
*
IF (F > R0) THEN
*
* plastic step: apply return mapping
* the cutting plane method uses values from the previous
* iteration to update the plastic parameter, dlambda.
* thus, all subsequent values (until dlambda) are related to
* the previous iteration, "i".
*-----
P = PN
LOOP_NR : DO WRITER = 1, MXITER
*
* compute the respective gradients of the von mises yield surface
RNV(2:6) = R0 ! INITIALIZE TO ZERO "OPTIONAL" COMPONENTS
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
RNV(1) = (STRESS(1) - R5*(STRESS(2) + STRESS(3))) / PHI
RNV(2) = (STRESS(2) - R5*(STRESS(3) + STRESS(1))) / PHI
RNV(3) = (STRESS(3) - R5*(STRESS(1) + STRESS(2))) / PHI
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
RNV(1) = (STRESS(1) - R5*STRESS(2)) / PHI
RNV(2) = (STRESS(2) - R5*STRESS(1)) / PHI
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
RNV(1) = STRESS(1) / PHI ! TO BE CHECKED ...
END SELECT
*
SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
RNV(4) = R3*STRESS(4) / PHI
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
RNV(4) = R3*STRESS(4) / PHI
RNV(5) = R3*STRESS(5) / PHI
RNV(6) = R3*STRESS(6) / PHI
END SELECT
*
* compute hardening modulus
HR = QR1*CR1*EXP(-CR1*P) + QR2*CR2*EXP(-CR2*P)
*
* compute denominator in the residual derivative of dlambda
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
DENOMCP = C1*(RNV(1)*RNV(1) + RNV(2)*RNV(2) + RNV(3)*RNV(3))
> + R2*C4*(RNV(1)*RNV(2) + RNV(2)*RNV(3) + RNV(3)*RNV(1))

```

```

CASE (1) ! PLANE STRESS CONDITION (RNV(3)=0)
  DENOMCP = C1*(RNV(1)*RNV(1) + RNV(2)*RNV(2))
> + R2*C4*(RNV(1)*RNV(2))
CASE (2) ! UNIAXIAL STRESS CONDITION (RNV(2)=RNV(3)=0)
  DENOMCP = C1*(RNV(1)*RNV(1)) ! TO BE CHECKED ...
END SELECT

*

SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (RNV(4)) IS PRESENT
  DENOMCP = DENOMCP + R4*C7*(RNV(4)*RNV(4))
CASE (3) ! ALL THREE SHEAR STRESSES (RNV(4:6)) ARE PRESENT
  DENOMCP = DENOMCP + R4*C7*(RNV(4)*RNV(4) + RNV(5)*RNV(5)
> + RNV(6)*RNV(6))
END SELECT

*

DENOMV = (R1 + DLAMBDA / (PDOT*DT))**C
DENOMDV = (C / (PDOT*DT))*(R1 + DLAMBDA / (PDOT*DT))**(C - R1)
IF (TEMP_SOFT) THEN
  TSTAR = (T - TR) / (TM - TR)
  TSOFT = R1 - TSTAR**M
  DENOM = DENOMCP + HR*DENOMV*TSOFT + SIGMAY*DENOMDV*TSOFT
ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
  DENOM = DENOMCP + HR*DENOMV + SIGMAY*DENOMDV
ENDIF

*
* compute residual increment ddlambda and update variable dlamba
DDLAMBDA = F / DENOM
DLAMBDA = DLAMBDA + DDLAMBDA

*
* update internal variables dependent on dlamba
P = P + DDLAMBDA
R = RP + HR*DLAMBDA
SIGMAY = SY + R

*
* update stress components using the elastic (hooke's) law
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
  STRESS(1) = STRESS(1) -
> DDLAMBDA*(C1*RNV(1) + C4*RNV(2) + C5*RNV(3))
  STRESS(2) = STRESS(2) -
> DDLAMBDA*(C4*RNV(1) + C2*RNV(2) + C6*RNV(3))
  STRESS(3) = STRESS(3) -
> DDLAMBDA*(C5*RNV(1) + C6*RNV(2) + C3*RNV(3))
CASE (1) ! PLANE STRESS CONDITION (RNV(3)=0)
  STRESS(1) = STRESS(1) -
> DDLAMBDA*(C1*RNV(1) + C4*RNV(2))
  STRESS(2) = STRESS(2) -
> DDLAMBDA*(C4*RNV(1) + C2*RNV(2))
CASE (2) ! UNIAXIAL STRESS CONDITION (RNV(2)=RNV(3)=0)
  STRESS(1) = STRESS(1) - DDLAMBDA*C1*RNV(1) ! TO BE CHECKED ...
END SELECT

*

SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (RNV(4)) IS PRESENT
  STRESS(4) = STRESS(4) - DDLAMBDA*C7*RNV(4)
CASE (3) ! ALL THREE SHEAR STRESSES (RNV(4:6)) ARE PRESENT
  STRESS(4) = STRESS(4) - DDLAMBDA*C7*RNV(4)
  STRESS(5) = STRESS(5) - DDLAMBDA*C8*RNV(5)
  STRESS(6) = STRESS(6) - DDLAMBDA*C9*RNV(6)
END SELECT

*
* compute updated equivalent stress
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
  PHI = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
> + STRESS(3)*STRESS(3) - STRESS(1)*STRESS(2)
> - STRESS(2)*STRESS(3) - STRESS(3)*STRESS(1)
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
  PHI = STRESS(1)*STRESS(1) + STRESS(2)*STRESS(2)
> - STRESS(1)*STRESS(2)
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
  PHI = STRESS(1)*STRESS(1)
END SELECT

*

SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT

```

```

CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
      PHI = PHI + R3*(STRESS(4)*STRESS(4))
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
      PHI = PHI + R3*(STRESS(4)*STRESS(4)
>          + STRESS(5)*STRESS(5)
>          + STRESS(6)*STRESS(6))
      END SELECT
*
      PHI = SQRT (PHI)
*
* compute yield function and new residual
      IF (TEMP_SOFT) THEN
          TSTAR = (T - TR) / (TM - TR) ! THIS RECALCULATION OF TSTAR
          TSOFT = R1 - TSTAR**M ! AND TSOFT SEEMS REDUNDANT HERE
          F = PHI - SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C)*TSOFT
*
* check convergence
          RESNOR = F / (SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C)*TSOFT)
          RESNOR = ABS (RESNOR)
          ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
          F = PHI - SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C)
*
* check convergence
          RESNOR = F / (SIGMAY*((R1 + DLAMBDA / (PDOT*DT))**C))
          RESNOR = ABS (RESNOR)
      ENDIF
*
      IF (RESNOR <= TOL) THEN
*
* update stress components based on plastic step
*-----
          STRESSNEW(1) = STRESS(1)
          STRESSNEW(2:4) = RO ! INITIALIZE TO ZERO "OPTIONAL" COMPONENTS
          SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
          CASE (0) ! NO PLANE STRESS CONDITION
              STRESSNEW(2) = STRESS(2)
              STRESSNEW(3) = STRESS(3)
          CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
              STRESSNEW(2) = STRESS(2)
          CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
          END SELECT
*
          SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
          CASE (0) ! NO SHEAR STRESS IS PRESENT
          CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
              STRESSNEW(4) = STRESS(4)
          CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
              STRESSNEW(4) = STRESS(4)
              STRESSNEW(5) = STRESS(5)
              STRESSNEW(6) = STRESS(6)
          END SELECT
*
* update remaining variables dependent of dlambda
*-----
* update temperature (only if temperature softening is activated)
      IF (TEMP_SOFT) THEN
          T = T + TQ*PHI*DLAMBDA / (RHO*CP)
          IF (T >= TM) THEN ! CHECK IF MELTING OCCURS
              STATE(13) = TM
              D = DC ! this will cause the GP to fail (see 990)
              GO TO 990
          ENDIF
      ENDIF
*
* compute hydrostatic stress based on new stress state,
      SIGMAH = STRESS(1)
      SELECT CASE (IPLANC)
      CASE (0) ! NO PLANE STRESS CONDITION
          SIGMAH = SIGMAH + STRESS(2) + STRESS(3)
      CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
          SIGMAH = SIGMAH + STRESS(2)
      CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
      END SELECT
*
      SIGMAH = SIGMAH / R3
*
* 3d stress state and compute deviatoric stress

```



```

S1 = STRESS(1) - SIGMAH
S2 = STRESS(2) - SIGMAH
S3 = STRESS(3) - SIGMAH
*
* finding principal stress based on the deviatoric stress and invariants
A1 = R5*(S1*S1 + S2*S2 + S3*S3)
*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0) ! NO SHEAR STRESS IS PRESENT
A2 = -S1*S2*S3 ! DETERMINANT OF THE DEVIATORIC STRESS TENSOR
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
A1 = A1 + STRESS(4)*STRESS(4)
A2 = S3*STRESS(4)*STRESS(4) - S1*S2*S3
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
A1 = A1 +
> STRESS(4)*STRESS(4) + STRESS(5)*STRESS(5) +
> STRESS(6)*STRESS(6)
A2 = S1*STRESS(5)*STRESS(5) +
> S2*STRESS(6)*STRESS(6) +
> S3*STRESS(4)*STRESS(4) -
> S1*S2*S3 - R2*STRESS(4)*STRESS(5)*STRESS(6)
END SELECT
*
A3 = -SQRT (R27 / A1)*A2*R5 / A1
A3 = SIGN (MIN (ABS (A3), R1), A3)
A4 = ACOS (A3) / R3
*
* compute (chronological) principal stresses
AUX = SQRT (A1 / R3)
S1 = SIGMAH + R2*AUX*COS(A4)
S2 = SIGMAH + R2*AUX*COS(A4-R2*PI/R3) ! only for postprocessing
S3 = SIGMAH + R2*AUX*COS(A4+R2*PI/R3) ! only for postprocessing
*
WE = WE + MAX (S1, R0)*DLAMBDA
D = WE / WC
*
* update state variables before exit
STATE(1) = SIGMAH
STATE(2) = PHI
STATE(3) = P
STATE(4) = PHITRIAL
STATE(5) = F
STATE(6) = R
STATE(7) = DDLAMBDA
STATE(8) = DLAMBDA
STATE(9) = NRITER
STATE(10) = DLAMBDA / DT
STATE(11) = D
*
STATE(13) = T
STATE(14) = WE
STATE(15) = CSON
*
STATE(16) = S1
STATE(17) = S2
STATE(18) = S3
*
EXIT LOOP_NR
ELSE ! (RESNOR >= TOL)
IF (NRITER == MXITER) THEN
CALL ERRMSS ('VPJC', 'MAX NO. OF N-R ITERATIONS REACHED')
STOP 'VPJC - MAX NO. OF N-R ITERATIONS REACHED'
ENDIF
ENDIF ! (RESNOR < TOL)
*
END DO LOOP_NR
*
ELSE ! (F <= R0)
*
* if not plasticity, elastic update
STRESSNEW(1) = STRESS(1)
STRESSNEW(2:4) = R0 ! INITIALIZE TO ZERO "OPTIONAL" COMPONENTS
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
STRESSNEW(2) = STRESS(2)
STRESSNEW(3) = STRESS(3)
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)

```

```

        STRESSNEW(2) = STRESS(2)
CASE (2)      ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
END SELECT
*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0)      ! NO SHEAR STRESS IS PRESENT
CASE (1)      ! ONLY TAU_XY (STRESS(4)) IS PRESENT
        STRESSNEW(4) = STRESS(4)
CASE (3)      ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
        STRESSNEW(4) = STRESS(4)
        STRESSNEW(5) = STRESS(5)
        STRESSNEW(6) = STRESS(6)
END SELECT
*
SIGMAH = STRESS(1)
SELECT CASE (IPLANC)
CASE (0)      ! NO PLANE STRESS CONDITION
        SIGMAH = SIGMAH + STRESS(2) + STRESS(3)
CASE (1)      ! PLANE STRESS CONDITION (STRESS(3)=0)
        SIGMAH = SIGMAH + STRESS(2)
CASE (2)      ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
END SELECT
*
SIGMAH = SIGMAH / R3
*
* 3d stress state and compute deviatoric stress
S1 = STRESS(1) - SIGMAH
S2 = STRESS(2) - SIGMAH
S3 = STRESS(3) - SIGMAH
*
* finding principal stress based on the deviatoric stress and invariants
A1 = R5*(S1*S1 + S2*S2 + S3*S3)
*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0)      ! NO SHEAR STRESS IS PRESENT
        A2 = -S1*S2*S3 ! DETERMINANT OF THE DEVIATORIC STRESS TENSOR
CASE (1)      ! ONLY TAU_XY (STRESS(4)) IS PRESENT
        A1 = A1 + STRESS(4)*STRESS(4)
        A2 = S3*STRESS(4)*STRESS(4) - S1*S2*S3
CASE (3)      ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
        A1 = A1 +
>          STRESS(4)*STRESS(4) + STRESS(5)*STRESS(5) +
>          STRESS(6)*STRESS(6)
        A2 = S1*STRESS(5)*STRESS(5) +
>          S2*STRESS(6)*STRESS(6) +
>          S3*STRESS(4)*STRESS(4) -
>          S1*S2*S3 - R2*STRESS(4)*STRESS(5)*STRESS(6)
END SELECT
*
A3 = -SQRT (R27 / A1)*A2*R5 / A1
A3 = SIGN (MIN (ABS (A3), R1), A3)
A4 = ACOS (A3) / R3
*
* compute (chronological) principal stresses
AUX = SQRT (A1 / R3)
S1 = SIGMAH + R2*AUX*COS(A4)
S2 = SIGMAH + R2*AUX*COS(A4-R2*PI/R3) ! only for postprocessing
S3 = SIGMAH + R2*AUX*COS(A4+R2*PI/R3) ! only for postprocessing
*
STATE(1) = SIGMAH ! COMPUTE SIGMAH AND INSERT IT HERE
STATE(2) = PHITRIAL ! COMPUTE PHI AND INSERT IT HERE
STATE(3) = PN ! COMPUTE P AND INSERT IT HERE
STATE(4) = PHITRIAL
STATE(5) = F
STATE(6) = RP
STATE(7) = R0
STATE(8) = R0
STATE(9) = R0
STATE(10) = R0
STATE(11) = D
STATE(12) = R1
STATE(13) = T
STATE(14) = WE
STATE(15) = CSON
*
STATE(16) = S1
STATE(17) = S2

```

```

STATE(18) = S3
*
ENDIF ! (F > R0)
*
GO TO 990
*
200 CONTINUE
*
=====
* solu == 2 : radial return solution strategy
=====
*
C COMPUTE THE GRADIENT OF THE VON MISES YIELD SURFACE
RNV(2:6) = R0 ! INITIALIZE TO ZERO "OPTIONAL" COMPONENTS
SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
CASE (0) ! NO PLANE STRESS CONDITION
RNV(1) = (STRESS(1) - R5*(STRESS(2) + STRESS(3))) / PHI
RNV(2) = (STRESS(2) - R5*(STRESS(3) + STRESS(1))) / PHI
RNV(3) = (STRESS(3) - R5*(STRESS(1) + STRESS(2))) / PHI
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
RNV(1) = (STRESS(1) - R5*STRESS(2)) / PHI
RNV(2) = (STRESS(2) - R5*STRESS(1)) / PHI
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
RNV(1) = STRESS(1) / PHI ! TO BE CHECKED ...
END SELECT
*
SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
RNV(4) = R3*STRESS(4) / PHI
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
RNV(4) = R3*STRESS(4) / PHI
RNV(5) = R3*STRESS(5) / PHI
RNV(6) = R3*STRESS(6) / PHI
END SELECT
*
IF (F > R0) THEN
*
C PLASTIC STEP: APPLY RADIAL RETURN MAPPING - USE NEWTON-RAPHSON
C ALGORITHM TO SOLVE THE RETURN MAPPING EQUATION
C-----
P = PN
LOOP_NR2 : DO WRITER = 1, MXITER
*
C COMPUTE HARDENING MODULUS
HR = QR1*CR1*EXP(-CR1*P) + QR2*CR2*EXP(-CR2*P)
C COMPUTE RESIDUAL DERIVATIVE
DENOMV = (R1 + DLAMBDA/(PDOT*DT))*C
DENOMDV = (C/(PDOT*DT))*(R1+DLAMBDA/(PDOT*DT))*C*(C-R1)
IF (TEMP_SOFT) THEN
TSTAR = (T - TR) / (TM - TR)
TSOFT = R1 - TSTAR**M
ELSE ! M = 0 --> THE CODE ASSUMES TSTAR=0, SO THAT TSOFT=1
TSTAR = R0
TSOFT = R1
ENDIF
DENOM = R3G + HR*DENOMV*TSOFT + SIGMAY*DENOMDV*TSOFT
C COMPUTE NEWTON-RAPHSON INCREMENT AND UPDATE VARIABLE DLAMBDA
DDLAMBDA = F / DENOM
DLAMBDA = DLAMBDA + DDLAMBDA
C COMPUTE NEW RESIDUAL
P = P + DDLAMBDA
R = RP + HR*DLAMBDA
SIGMAY = SY + R
DENOMV = (R1 + DLAMBDA/(PDOT*DT))*C
F = PHITRIAL - R3G*DLAMBDA - SIGMAY*DENOMV*TSOFT
C CHECK CONVERGENCE
RESNOR = F / (SIGMAY*DENOMV*TSOFT)
RESNOR = ABS (RESNOR)
*
IF (RESNOR <= TOL) THEN
*
C UPDATE VARIABLES DEPENDENT OF DLAMBDA
C-----
C UPDATE TEMPERATURE (ONLY IF TEMPERATURE SOFTENING IS ACTIVATED)

```

```

IF (TEMP_SOFT) THEN
  PHIEFF = PHITRIAL - R3G*DLAMBDA ! EQUIVALENT (EFFECTIVE) STRESS
  T = T + TQ*PHIEFF*DLAMBDA / (RHO*CP)
  IF (T >= TM) THEN ! CHECK IF MELTING OCCURS
    STATE(13) = TM
    D = DC ! this will cause the GP to fail (see 990)
    GO TO 990
  ENDIF
ENDIF
C COMPUTE HYDROSTATIC STRESS BASED ON TRIAL STRESS STATE,
C I.E. UTILIZE THAT THE HYDROSTATIC STRESS IS THE SAME
C FOR BOTH TRIAL STATE AND FINAL STATE. ONLY THE DEVIATORIC
C STRESS STATE WILL CHANGE.
  SIGMAH = STRESS(1)
  SELECT CASE (IPLANC)
  CASE (0) ! NO PLANE STRESS CONDITION
    SIGMAH = SIGMAH + STRESS(2) + STRESS(3)
  CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
    SIGMAH = SIGMAH + STRESS(2)
  CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
  END SELECT
  SIGMAH = SIGMAH / R3
C COMPUTE DEVIATORIC STRESS
  S1 = STRESS(1) - SIGMAH
  S2 = STRESS(2) - SIGMAH
  S3 = STRESS(3) - SIGMAH
C FINDING PRINCIPAL STRESS BASED ON THE DEVIATORIC STRESS AND INVARIANTS
  A1 = R5*(S1*S1 + S2*S2 + S3*S3)
  SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
  CASE (0) ! NO SHEAR STRESS IS PRESENT
    A2 = -S1*S2*S3 ! DETERMINANT OF THE DEVIATORIC STRESS TENSOR
  CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
    A1 = A1 + STRESS(4)*STRESS(4)
    A2 = S3*STRESS(4)*STRESS(4) - S1*S2*S3
  CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
    A1 = A1 +
    > STRESS(4)*STRESS(4) + STRESS(5)*STRESS(5) +
    > STRESS(6)*STRESS(6)
    A2 = S1*STRESS(5)*STRESS(5) +
    > S2*STRESS(6)*STRESS(6) +
    > S3*STRESS(4)*STRESS(4) -
    > S1*S2*S3 - R2*STRESS(4)*STRESS(5)*STRESS(6)
  END SELECT
  A3 = -SQRT (R27 / A1)*A2*R5 / A1
  A3 = SIGN (MIN (ABS (A3), R1), A3)
  A4 = ACOS (A3) / R3
C COMPUTE (CHRONOLOGICAL) PRINCIPAL STRESSES
  AUX = SQRT (A1 / R3)
  S1 = SIGMAH + R2*AUX*COS(A4)
  S2 = SIGMAH + R2*AUX*COS(A4-R2*PI/R3) ! only for postprocessing
  S3 = SIGMAH + R2*AUX*COS(A4+R2*PI/R3) ! only for postprocessing
  WE = WE + MAX (S1, RO)*DLAMBDA
  D = WE / WC
*
C UPDATE STRESS COMPONENTS USING THE ELASTIC (HOOKE'S) LAW
C-----
  SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
  CASE (0) ! NO PLANE STRESS CONDITION
    STRESSNEW(1) = STRESS(1) -
    > DLAMBDA*(C1*RVN(1) + C4*RVN(2) + C5*RVN(3))
    STRESSNEW(2) = STRESS(2) -
    > DLAMBDA*(C4*RVN(1) + C2*RVN(2) + C6*RVN(3))
    STRESSNEW(3) = STRESS(3) -
    > DLAMBDA*(C5*RVN(1) + C6*RVN(2) + C3*RVN(3))
  CASE (1) ! PLANE STRESS CONDITION (RVN(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
    STRESSNEW(1) = STRESS(1) -
    > DLAMBDA*(C1*RVN(1) + C4*RVN(2))
    STRESSNEW(2) = STRESS(2) -
    > DLAMBDA*(C4*RVN(1) + C2*RVN(2))
  CASE (2) ! UNIAXIAL STRESS CONDITION (RVN(2)=RVN(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
    STRESSNEW(1) = STRESS(1) -
    > DLAMBDA*C1*RVN(1) ! TO BE CHECKED ...
  END SELECT

```

```

*
      SELECT CASE (ITAU) ! TREAT SHEAR COMPONENTS
      CASE (0) ! NO SHEAR STRESS IS PRESENT
      CASE (1) ! ONLY TAU_XY (RNV(4)) IS PRESENT
        STRESSNEW(4) = STRESS(4) - DLAMBDA*C7*RNV(4)
      CASE (3) ! ALL THREE SHEAR STRESSES (RNV(4:6)) ARE PRESENT
        STRESSNEW(4) = STRESS(4) - DLAMBDA*C7*RNV(4)
        STRESSNEW(5) = STRESS(5) - DLAMBDA*C8*RNV(5)
        STRESSNEW(6) = STRESS(6) - DLAMBDA*C9*RNV(6)
      END SELECT

*
C COMPUTE UPDATED EQUIVALENT STRESS
      SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST
      CASE (0) ! NO PLANE STRESS CONDITION
        PHI = STRESSNEW(1)*STRESSNEW(1) +
          > STRESSNEW(2)*STRESSNEW(2) +
          > STRESSNEW(3)*STRESSNEW(3) -
          > STRESSNEW(1)*STRESSNEW(2) -
          > STRESSNEW(2)*STRESSNEW(3) -
          > STRESSNEW(3)*STRESSNEW(1)
      CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
        PHI = STRESSNEW(1)*STRESSNEW(1) +
          > STRESSNEW(2)*STRESSNEW(2) -
          > STRESSNEW(1)*STRESSNEW(2)
      CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
        PHI = STRESSNEW(1)*STRESSNEW(1)
      END SELECT

*
      SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
      CASE (0) ! NO SHEAR STRESS IS PRESENT
      CASE (1) ! ONLY TAU_XY (STRESSNEW(4)) IS PRESENT
        PHI = PHI + R3*(STRESSNEW(4)*STRESSNEW(4))
      CASE (3) ! ALL THREE SHEAR STRESSES (STRESSNEW(4:6)) ARE PRESENT
        PHI = PHI + R3*(STRESSNEW(4)*STRESSNEW(4)
          > + STRESSNEW(5)*STRESSNEW(5)
          > + STRESSNEW(6)*STRESSNEW(6))
      END SELECT

*
      PHI = SQRT (PHI)

*
C UPDATE STATE VARIABLES (SDV) BEFORE EXIT
      STATE(1) = SIGMAH
      STATE(2) = PHI
      STATE(3) = P
      STATE(4) = PHITRIAL
      STATE(5) = F
      STATE(6) = R
      STATE(7) = DDLAMBDA
      STATE(8) = DLAMBDA
      STATE(9) = WRITER
      STATE(10) = DLAMBDA / DT
      STATE(11) = D

*
      STATE(13) = T
      STATE(14) = WE
      STATE(15) = CSON

*
      STATE(16) = S1
      STATE(17) = S2
      STATE(18) = S3

*
      EXIT LOOP_NR2
      ELSE ! (RESNOR >= TOL)
      IF (WRITER == MXITER) THEN
        CALL ERRMSS ('VPJC', 'MAX NO. OF N-R ITERATIONS REACHED')
        STOP 'VPJC - MAX NO. OF N-R ITERATIONS REACHED'
      ENDIF
      ENDIF ! (RESNOR < TOL)

*
      END DO LOOP_NR2

*
      ELSE ! (F <= R0)

*
C IF NOT PLASTICITY, ELASTIC UPDATE
      STRESSNEW(1) = STRESS(1)
      STRESSNEW(2:4) = R0 ! INITIALIZE TO ZERO "OPTIONAL" COMPONENTS
      SELECT CASE (IPLANC) ! TREAT NORMAL COMPONENTS FIRST

```

```

CASE (0) ! NO PLANE STRESS CONDITION
STRESSNEW(2) = STRESS(2)
STRESSNEW(3) = STRESS(3)
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
STRESSNEW(2) = STRESS(2)
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
END SELECT
*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0) ! NO SHEAR STRESS IS PRESENT
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
STRESSNEW(4) = STRESS(4)
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
STRESSNEW(4) = STRESS(4)
STRESSNEW(5) = STRESS(5)
STRESSNEW(6) = STRESS(6)
END SELECT
*
SIGMAH = STRESS(1)
SELECT CASE (IPLANC)
CASE (0) ! NO PLANE STRESS CONDITION
SIGMAH = SIGMAH + STRESS(2) + STRESS(3)
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
SIGMAH = SIGMAH + STRESS(2)
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
* THIS CASE IS ILLEGAL FOR SOLU == 2 (SEE TEST AT BEGINNING OF ROUTINE)
END SELECT
SIGMAH = SIGMAH / R3
*
* 3d stress state and compute deviatoric stress
S1 = STRESS(1) - SIGMAH
S2 = STRESS(2) - SIGMAH
S3 = STRESS(3) - SIGMAH
*
* finding principal stress based on the deviatoric stress and invariants
A1 = R5*(S1*S1 + S2*S2 + S3*S3)
*
SELECT CASE (ITAU) ! ADD SHEAR COMPONENTS IF PRESENT
CASE (0) ! NO SHEAR STRESS IS PRESENT
A2 = -S1*S2*S3 ! DETERMINANT OF THE DEVIATORIC STRESS TENSOR
CASE (1) ! ONLY TAU_XY (STRESS(4)) IS PRESENT
A1 = A1 + STRESS(4)*STRESS(4)
A2 = S3*STRESS(4)*STRESS(4) - S1*S2*S3
CASE (3) ! ALL THREE SHEAR STRESSES (STRESS(4:6)) ARE PRESENT
A1 = A1 +
> STRESS(4)*STRESS(4) + STRESS(5)*STRESS(5) +
> STRESS(6)*STRESS(6)
A2 = S1*STRESS(5)*STRESS(5) +
> S2*STRESS(6)*STRESS(6) +
> S3*STRESS(4)*STRESS(4) -
> S1*S2*S3 - R2*STRESS(4)*STRESS(5)*STRESS(6)
END SELECT
*
A3 = -SQRT (R27 / A1)*A2*R5 / A1
A3 = SIGN (MIN (ABS (A3), R1), A3)
A4 = ACOS (A3) / R3
*
* compute (chronological) principal stresses
AUX = SQRT (A1 / R3)
S1 = SIGMAH + R2*AUX*COS(A4)
S2 = SIGMAH + R2*AUX*COS(A4-R2*PI/R3) ! only for postprocessing
S3 = SIGMAH + R2*AUX*COS(A4+R2*PI/R3) ! only for postprocessing
*
STATE(1) = SIGMAH ! compute SIGMAH and insert it here
STATE(2) = PHITRIAL ! compute PHI and insert it here
STATE(3) = PN ! compute P and insert it here
STATE(4) = PHITRIAL
STATE(5) = F
STATE(6) = RP
STATE(7) = R0
STATE(8) = R0
STATE(9) = R0
STATE(10) = R0
STATE(11) = D

```

```

STATE(12) = R1
STATE(13) = T
STATE(14) = WE
STATE(15) = CSON
*
STATE(16) = S1
STATE(17) = S2
STATE(18) = S3
*
ENDIF ! (F > R0)
*
* decide if material point stiffness is zero, i.e. failed
*-----
990 IF (D >= DC) THEN
*
* the current gp was already failed or is failing now :
* state(1:10,12,13), the relevant components
* of stressnew and cson are set to 0.
* the total strains (strain) are not set to 0 so they continue to be
* updated as long as the element is not eroded, i.e. as long
* as the element has some virgin gauss points.
*
SELECT CASE (ITAU)
CASE (0)
STRESSNEW(1:3) = R0
CASE (1)
STRESSNEW(1:4) = R0
CASE (2)
STRESSNEW(1:5) = R0
CASE (3)
STRESSNEW(1:6) = R0
END SELECT
STATE(1:10) = R0
* state(11) (damage d) keeps last value it had before failure
STATE(12) = R0
* state(13) (absolute temperature t) keeps last value it had before failure
* state(14) (damage accumulation we) keeps last value it had before failure
CSON = R0
STATE(15) = CSON
ELSE
*
* no failure
*
STATE(12) = R1
*
* strain increment(s) and total strain(s) corresponding to
* the imposed zero stress condition(s)
*
SELECT CASE (IPLANC)
CASE (0) ! NO PLANE STRESS CONDITION
CASE (1) ! PLANE STRESS CONDITION (STRESS(3)=0)
* store provisional strain increment
AUX3 = STRAININC(3)
* compute and update the actual strain increment
FACTOR = T2 / YOUNG
STRAININC(3) = FACTOR*(STRESSNEW(1) - STRESSOLD(1)
> + STRESSNEW(2) - STRESSOLD(2))
> - STRAININC(1) - STRAININC(2)
* update the total strain
STRAIN(3) = STRAIN(3) - AUX3 + STRAININC(3)
CASE (2) ! UNIAXIAL STRESS CONDITION (STRESS(2)=STRESS(3)=0)
AUX2 = STRAININC(2)
AUX3 = STRAININC(3)
FACTOR = T2 / YOUNG
STRAININC(2) = 0.5DO*(FACTOR*(STRESSNEW(1) - STRESSOLD(1))
> - STRAININC(1))
STRAININC(3) = STRAININC(2)
STRAIN(2) = STRAIN(2) - AUX2 + STRAININC(2)
STRAIN(3) = STRAIN(3) - AUX3 + STRAININC(3)
END SELECT
ENDIF
*
* erosion (optional)
*
IF (L_EROSION) THEN
IF (D >= DC) FAIL_FLAG = .TRUE.
ENDIF

```

```
*  
  END SUBROUTINE VPJC  
*=====
```

```
  END MODULE M_MATERIAL_VPJC
```



Europe Direct is a service to help you find answers to your questions about the European Union

Free phone number (\*): 00 800 6 7 8 9 10 11

(\*): Certain mobile telephone operators do not allow access to 00 800 numbers or these calls may be billed.

A great deal of additional information on the European Union is available on the Internet.

It can be accessed through the Europa server <http://europa.eu>

#### **How to obtain EU publications**

Our publications are available from EU Bookshop (<http://bookshop.europa.eu>), where you can place an order with the sales agent of your choice.

The Publications Office has a worldwide network of sales agents.

You can obtain their contact details by sending a fax to (352) 29 29-42758.

## JRC Mission

As the Commission's in-house science service, the Joint Research Centre's mission is to provide EU policies with independent, evidence-based scientific and technical support throughout the whole policy cycle.

Working in close cooperation with policy Directorates-General, the JRC addresses key societal challenges while stimulating innovation through developing new methods, tools and standards, and sharing its know-how with the Member States, the scientific community and international partners.

*Serving society  
Stimulating innovation  
Supporting legislation*

