

A TWO-DIMENSIONAL FEM CODE FOR THE ANALYSIS OF LARGE DEFORMATIONS OF HYPERELASTIC VISCOPLASTIC SOLIDS

S. Badrinarayanan and N. Zabaras
Sibley School of Mechanical and Aerospace Engineering
188 Engineering and Theory Center Bldg.
Cornell University
Ithaca, NY 14853-3801

Technical Report MM 93-05

August 30, 1993

Abstract

Based on a constitutive model by Anand [1], plane strain and axially symmetric large deformations of an isotropic, hyperelastic-viscoplastic solid are analysed using an updated-Lagrangian FEM formulation. The algorithms for the solution of two independent problems : the constitutive problem and the kinematic problem, are formulated and solved. The constitutive problem involves determining the stress and the state variables, knowing the history of deformation of the solid. A radial return method is employed to solve the governing equations by Euler-backward integration method. The kinematic problem tackles the determination of the motion of the body subjected to certain traction and displacement boundary conditions, knowing the material constitutive model. The consistent material moduli are employed while solving the linearized principle of virtual work equation (PVW). Based on the linearization of the PVW equation, a finite element scheme is described, which is used to solve the quasi-static boundary value problem. To check the accuracy and stability of the algorithm, some numerical examples are worked out.

1 Introduction

Analysis of large deformations mechanics problems is extremely important in metal forming operations. Usually, cold and hot metal forming operations result in a high amount of plastic flow. An extensive review of the subject can be found in references [2-5]. The elasticity was

modelled as hypoelastic and formulations were dealt with in [2,4,5]. Since hypoelastic models resulted in spurious stress values under large rotations, a hyper elastic formulation was considered by Weber and Anand[3]. In this work, the elasticity is modelled as hyperelastic, and the kinematic problem is solved by using an updated lagrangian reference frame.

Initially, an isotropic hyperelastic-viscoplastic constitutive model for finite deformations of polycrystalline metals is described. Thermal effects are ignored and throughout this work, the deformations are considered isothermal. The constitutive model that is used in this work is taken from [1]. The deformation gradient is decomposed by means of Lee's multiplicative decomposition into a plastic and an elastic part [6]. The following sections deal separately with the constitutive problem, the kinematic problem, the finite element implementation and the numerical examples.

2 Constitutive equations

The most commonly used phenomenological model for infinitesimal elastic-plastic deformation is the classical J_2 flow theory with isotropic hardening. The set of constitutive equations considered here are essentially, a generalization of this theory to model large, finite elastic and plastic deformations and rate dependent plastic flow.

The total deformation gradient \mathbf{F} is decomposed by Lee's decomposition as (figure (1))

$$\mathbf{F} = \mathbf{F}^e \bar{\mathbf{F}}^p, \quad \det \mathbf{F}^e > 0 \quad (1)$$

where \mathbf{F} is the total deformation gradient, \mathbf{F}^e , the elastic deformation gradient and $\bar{\mathbf{F}}^p$, the plastic deformation gradient, where $\det \bar{\mathbf{F}}^p = 1$.

Here, the intermediate destressed configuration is *not* a physically achievable one but rather an imaginary configuration which nonetheless aids in visualising the individual mechanisms that separately result in plastic and elastic deformations. From the undeformed state, we assume that the intermediate configuration is reached by means of slip and glide of dislocation planes.

The constitutive equation for stress (based on the hyperelastic model by Anand[1]) is given by

$$\bar{\mathbf{T}} = \mathcal{L}^e [\bar{\mathbf{E}}^e] \quad (2)$$

where $\bar{\mathbf{E}}^e$ the strain measure, is defined with respect to the intermediate (unstressed) configuration as

$$\bar{\mathbf{E}}^e = \ln \mathbf{U}^e \quad (3)$$

while the conjugate stress measure $\bar{\mathbf{T}}$ is the pullback of the Kirchoff stress with respect to \mathbf{R}^e , ie.,

$$\bar{\mathbf{T}} = \mathbf{R}^{eT} \det(\mathbf{U}^e) \mathbf{T} \mathbf{R}^e \quad (4)$$

Here \mathbf{U}^e and \mathbf{R}^e are calculated from the polar decomposition of \mathbf{F}^e , ie.,

$$\mathbf{F}^e = \mathbf{R}^e \mathbf{U}^e \quad (5)$$

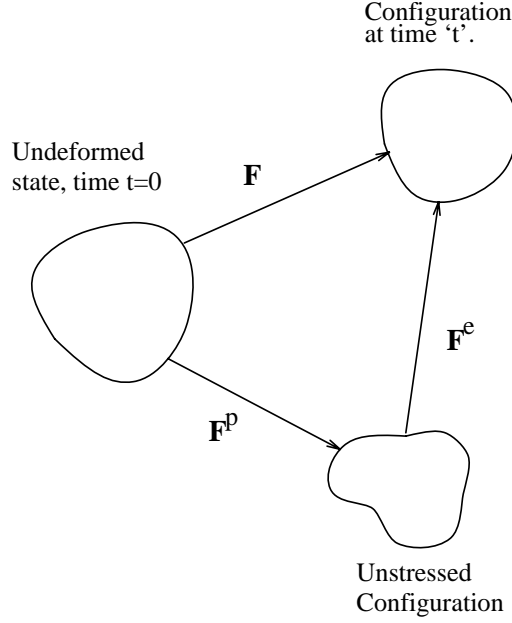


Figure 1: Multiplicative decomposition of the deformation gradient

The elastic isotropic moduli \mathcal{L}^e are defined as

$$\mathcal{L}^e = 2\mu \mathcal{I} + \left(\kappa - \frac{2}{3}\mu \right) \mathbf{I} \otimes \mathbf{I} \quad (6)$$

A flow rule is given in the form of the evolution of $\bar{\mathbf{F}}^p$ with zero spin of the intermediate configuration:

$$\bar{\mathbf{L}}^p = \bar{\mathbf{D}}^p = \dot{\bar{\mathbf{F}}^p} (\bar{\mathbf{F}}^p)^{-1} = \sqrt{\frac{3}{2}} \dot{\epsilon}^p \bar{\mathbf{N}}^p(\bar{\mathbf{T}}', \tilde{\sigma}) \quad (7)$$

$$\bar{\mathbf{W}}^p = 0 \quad (8)$$

where

$$\bar{\mathbf{N}}^p(\bar{\mathbf{T}}', \tilde{\sigma}) = \sqrt{\frac{3}{2}} \frac{\bar{\mathbf{T}}'}{\tilde{\sigma}} \quad (9)$$

and $\tilde{\sigma}$ is given by

$$\tilde{\sigma} = \sqrt{\frac{3}{2} \bar{\mathbf{T}}' \cdot \bar{\mathbf{T}}'} \quad (10)$$

and

$$\bar{\mathbf{T}}' = \bar{\mathbf{T}} - \frac{\text{tr } \bar{\mathbf{T}}}{3} \mathbf{I} \quad (11)$$

The evolution of the equivalent plastic strain $\dot{\epsilon}^p$ is specified via uniaxial experiment as:

$$\dot{\epsilon}^p = f(\tilde{\sigma}, s) \quad (12)$$

while the evolution of the isotropic isothermal scalar resistance s is defined as:

$$\dot{s} = g(\tilde{\sigma}, s) \quad (13)$$

3 An implicit time integration procedure

Let us suppose that the kinematic problem is completely solved and that we completely know the history of the deformation. We are interested in evaluating the history of the stress and the state variables. Since the stress and the state variables are history dependent, we have to integrate the evolution laws over discrete time steps and update them. The accuracy with which the stress can be calculated depends on the amount of time step. Given two successive configurations over a discrete time step and knowing the state completely at one of the configurations, we have to evaluate the state at the later configuration. Here the new configuration is specified with respect to the previous configuration. Once the state of the new configuration is specified, the lagrangian reference frame is updated to this new configuration and the method is repeated. Therefore, this method is called updated lagrangian reference frame method.

In figure (2), the configuration at time t_n and t_{n+1} are shown as well as the evolution of the intermediate unstressed configurations. Let \mathbf{F}_u be the known relative deformation gradient between the two configurations, ie.,

$$\mathbf{F}_u = \mathbf{F}_{n+1} (\mathbf{F}_n)^{-1} \quad (14)$$

With Euler backward integration of the flow rule (7), we have

$$\bar{\mathbf{F}}_{n+1}^p = \exp\left(\Delta t \bar{\mathbf{D}}_{n+1}^p\left(\bar{\mathbf{T}}'_{n+1}, s_{n+1}\right)\right) \bar{\mathbf{F}}_n^p \quad (15)$$

Similarly for the evolution of the state variable

$$s_{n+1} - s_n - \Delta t g(\bar{\sigma}_{n+1}, s_{n+1}) = 0 \quad (16)$$

Let us define the incremental elastic deformation gradient \mathbf{F}_*^e by assuming that there is no additional plastic deformation during the time increment (in a sense, this is the trial elastic deformation gradient). Then,

$$\mathbf{F}_*^e = \mathbf{F}_u \mathbf{F}_n^e = \mathbf{F}_{n+1}^e \Delta \bar{\mathbf{F}}^p \quad (17)$$

where

$$\Delta \bar{\mathbf{F}}^p = \exp\left(\Delta t \bar{\mathbf{D}}_{n+1}^p\left(\bar{\mathbf{T}}'_{n+1}, s_{n+1}\right)\right)$$

From the uniqueness of the polar decomposition, the above equation implies

$$\mathbf{R}_*^e = \mathbf{R}_{n+1}^e \quad (18)$$

and

$$\mathbf{U}_*^e = \mathbf{U}_{n+1}^e \exp\left(\Delta t \bar{\mathbf{D}}_{n+1}^p\left(\bar{\mathbf{T}}'_{n+1}, s_{n+1}\right)\right) = \mathbf{U}_{n+1}^e \Delta \bar{\mathbf{F}}^p \quad (19)$$

Taking logarithms of the last equation and using the definitions of the logarithmic elastic strain, we arrive at:

$$\bar{\mathbf{E}}_{n+1}^e + \Delta t \bar{\mathbf{D}}_{n+1}^p\left(\bar{\mathbf{T}}'_{n+1}, s_{n+1}\right) = \bar{\mathbf{E}}_*^e \quad (20)$$

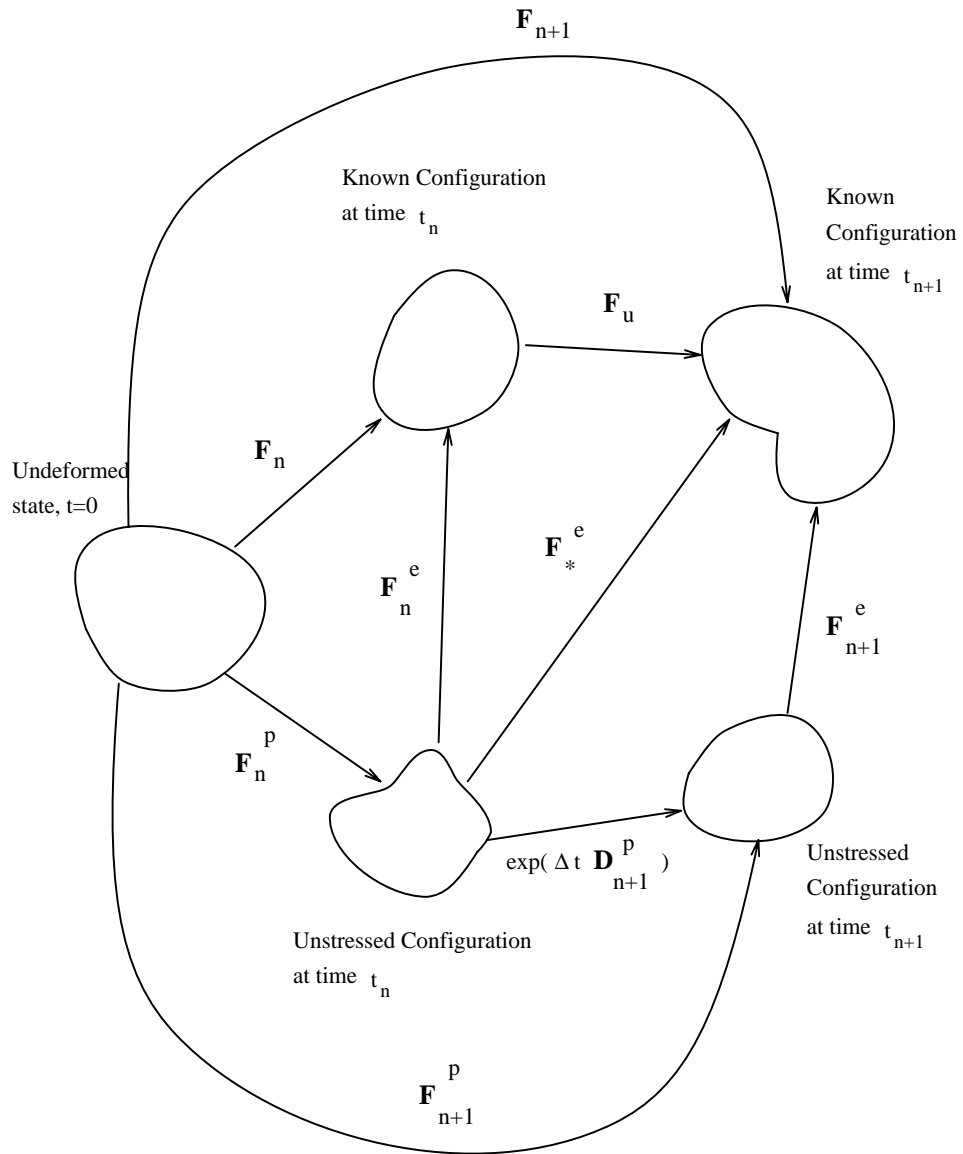


Figure 2: Deformation at the n^{th} and $(n + 1)^{th}$ configurations

This last equation, together with the hyperelastic model discussed earlier, gives:

$$\bar{\mathbf{T}}_{n+1} = \mathcal{L}^e [\bar{\mathbf{E}}_{n+1}^e] = \mathcal{L}^e [\bar{\mathbf{E}}_*^e] - \mathcal{L}^e \left[\Delta t \bar{\mathbf{D}}_{n+1}^p \left(\bar{\mathbf{T}}'_{n+1}, s_{n+1} \right) \right] \quad (21)$$

Let us define the trial stress $\bar{\mathbf{T}}_*$ as follows:

$$\bar{\mathbf{T}}_* = \mathcal{L}^e [\bar{\mathbf{E}}_*^e] \quad (22)$$

This along with the flow rule (7) and the definition of elastic moduli (6), results in

$$\bar{\mathbf{T}}_{n+1} = \bar{\mathbf{T}}_* - 2\mu \sqrt{\frac{3}{2}} \dot{\epsilon}^p \Delta t \bar{\mathbf{N}}_{n+1}^p \left(\bar{\mathbf{T}}'_{n+1}, s_{n+1} \right) \quad (23)$$

Using the definition of $\bar{\mathbf{N}}_{n+1}^p$ we can write:

$$\bar{\mathbf{T}}_{n+1} = \bar{\mathbf{T}}_* - 3\mu \dot{\epsilon}^p \Delta t \frac{\bar{\mathbf{T}}'_{n+1}}{\tilde{\sigma}_{n+1}} \quad (24)$$

and taking the deviatoric parts of the above equation and rearranging, we have:

$$\left(1 + \frac{3\mu \dot{\epsilon}^p \Delta t}{\tilde{\sigma}_{n+1}} \right) \bar{\mathbf{T}}'_{n+1} = \bar{\mathbf{T}}_* \quad (25)$$

ie., $\bar{\mathbf{T}}'_{n+1}$ and $\bar{\mathbf{T}}_*$ have the same direction. Taking the magnitude on both sides of the above equation results in

$$\tilde{\sigma}_{n+1} \left(1 + \frac{3\mu \dot{\epsilon}^p \Delta t}{\tilde{\sigma}_{n+1}} \right) = \tilde{\sigma}_* \quad (26)$$

where

$$\tilde{\sigma}_* = \sqrt{\frac{3}{2} \bar{\mathbf{T}}_* \cdot \bar{\mathbf{T}}_*} \quad (27)$$

and

$$\tilde{\sigma}_{n+1} = \sqrt{\frac{3}{2} \bar{\mathbf{T}}'_{n+1} \cdot \bar{\mathbf{T}}'_{n+1}} \quad (28)$$

or, simplifying,

$$\tilde{\sigma}_{n+1} - \tilde{\sigma}_* + 3\mu \Delta t f(\tilde{\sigma}_{n+1}, s_{n+1}) = 0 \quad (29)$$

This together with equation (16) forms a system of two nonlinear algebraic equations for $\tilde{\sigma}_{n+1}$ and s_{n+1} . The solution to these equations provides $\bar{\mathbf{T}}'_{n+1}$. This method is called the radial return method, since the correction direction from the trial stress follows a radial path with direction:

$$\bar{\mathbf{N}}_{n+1}^p = \sqrt{\frac{3}{2}} \left(\frac{\bar{\mathbf{T}}'_*}{\tilde{\sigma}_*} \right) \quad (30)$$

The radial return factor is given by

$$\eta_{n+1} = \frac{\tilde{\sigma}_{n+1}}{\tilde{\sigma}_*} \quad (31)$$

ie.,

$$\bar{\mathbf{T}}'_{n+1} = \eta_{n+1} \bar{\mathbf{T}}'_* \quad (32)$$

It remains to calculate the pressure

$$\bar{p}_{n+1} \equiv -\frac{1}{3} \text{tr}(\bar{\mathbf{T}}_{n+1})$$

Taking the trace of equation (24) we get

$$\bar{p}_{n+1} = \bar{p}_* \equiv -\frac{1}{3} \text{tr}(\bar{\mathbf{T}}_*) \quad (33)$$

and finally:

$$\bar{\mathbf{T}}_{n+1} = \eta_{n+1} \bar{\mathbf{T}}'_* - \bar{p}_* \mathbf{I} \quad (34)$$

Now using equations (4) and (18) we can calculate \mathbf{T}_{n+1} as

$$\mathbf{T}_{n+1} = [\det(\mathbf{U}_{n+1}^e)]^{-1} \mathbf{R}_*^e \bar{\mathbf{T}}_{n+1} \mathbf{R}_*^{eT} \quad (35)$$

Using the identity

$$[\det(\mathbf{U}_{n+1}^e)]^{-1} = [\det(\exp(\bar{\mathbf{E}}_{n+1}^e))]^{-1} = \exp(-\text{tr}\bar{\mathbf{E}}_{n+1}^e) \quad (36)$$

The above equation, along with equations (2), (6) and (33) results in

$$\text{tr}\bar{\mathbf{T}}_{n+1} = 3\kappa \text{tr}\bar{\mathbf{E}}_{n+1}^e = -3\bar{p}_{n+1} \quad (37)$$

we obtain:

$$[\det(\mathbf{U}_{n+1}^e)]^{-1} = \exp\left(\frac{\bar{p}_{n+1}}{\kappa}\right) = \exp\left(\frac{\bar{p}_*}{\kappa}\right) \quad (38)$$

So finally,

$$\mathbf{T}_{n+1} = \exp\left(\frac{\bar{p}_*}{\kappa}\right) \mathbf{R}_*^e \bar{\mathbf{T}}_{n+1} \mathbf{R}_*^{eT} \quad (39)$$

Now one can update the \mathbf{F}_{n+1}^e as part of the constitutive integration algorithm. Infact, from equation (32) and equation (2) we arrive at the conclusion that

$$2\mu \left(\ln \mathbf{U}_{n+1}^e - \frac{1}{3} \text{tr} \ln \mathbf{U}_{n+1}^e \mathbf{I} \right) = \eta_{n+1} 2\mu \left(\ln \mathbf{U}_*^e - \frac{1}{3} \text{tr} \ln \mathbf{U}_*^e \mathbf{I} \right) \quad (40)$$

which implies that \mathbf{U}_*^e and \mathbf{U}_{n+1}^e have the same direction. From the spectral theorem, we obtain

$$\mathbf{U}_{n+1}^e = \sum \lambda^i \mathbf{e}_*^i \otimes \mathbf{e}_*^i \quad (41)$$

and

$$\mathbf{U}_*^e = \sum \lambda_*^i \mathbf{e}_*^i \otimes \mathbf{e}_*^i \quad (42)$$

The last three equations above result upon simplification,

$$\ln \left[\frac{\lambda^i}{(\lambda^1 \lambda^2 \lambda^3)^{1/3}} \right] = \eta_{n+1} \ln \left[\frac{\lambda_*^i}{(\lambda_*^1 \lambda_*^2 \lambda_*^3)^{1/3}} \right]$$

or, upon further simplification,

$$\frac{\lambda^i}{(\lambda^1 \lambda^2 \lambda^3)^{1/3}} = \left[\frac{\lambda_*^i}{(\lambda_*^1 \lambda_*^2 \lambda_*^3)^{1/3}} \right]^{\eta_{n+1}}$$

Also we have, by incompressibility of $\Delta \bar{\mathbf{F}}^p$ and equation (19),

$$\lambda^1 \lambda^2 \lambda^3 = \lambda_*^1 \lambda_*^2 \lambda_*^3$$

Hence we finally arrive at:

$$\lambda^i = \frac{(\lambda_*^i)^{\eta_{n+1}}}{(\lambda_*^1 \lambda_*^2 \lambda_*^3)^{(\eta_{n+1}-1)/3}} \quad (43)$$

From the above we can calculate \mathbf{F}_{n+1}^e .

Summary of the time-integration constitutive algorithm:

1. Calculate the trial elastic deformation gradient

$$\mathbf{F}_*^e = \mathbf{F}_u \mathbf{F}_n^e$$

2. Perform the polar decomposition

$$\mathbf{F}_*^e = \mathbf{R}_*^e \mathbf{U}_*^e$$

3. Compute the trial elastic strain $\bar{\mathbf{E}}_*^e$. To do this, first compute the spectral decomposition of \mathbf{U}_*^e

$$\mathbf{U}_*^e = \sum \lambda_*^i \mathbf{e}_*^i \otimes \mathbf{e}_*^i$$

where λ_*^i are the eigen values and \mathbf{e}_*^i the eigen vectors of \mathbf{U}_*^e , and then

$$\bar{\mathbf{E}}_*^e = \sum (\ln \lambda_*^i) \mathbf{e}_*^i \otimes \mathbf{e}_*^i$$

4. Calculate the trial stress

$$\bar{\mathbf{T}}_* = \mathcal{L}^e [\bar{\mathbf{E}}_*^e]$$

5. Calculate the mean normal pressure and the deviatoric part of $\bar{\mathbf{T}}_*$

$$\bar{p}_* = -\frac{1}{3} \text{tr} \left(\bar{\mathbf{T}}_* \right)$$

$$\bar{\mathbf{T}}_*' = \bar{\mathbf{T}}_* + \bar{p}_* \mathbf{I}$$

6. Calculate the trial equivalent tensile stress $\tilde{\sigma}_*$

$$\tilde{\sigma}_* = \sqrt{\frac{3}{2} \bar{\mathbf{T}}_*' \cdot \bar{\mathbf{T}}_*'}$$

7. Calculate $\tilde{\sigma}_{n+1}$ and s_{n+1} by solving the following equations (Appendix A)

$$\tilde{\sigma}_{n+1} - \tilde{\sigma}_* + 3\mu \Delta t f(\tilde{\sigma}_{n+1}, s_{n+1}) = 0$$

$$s_{n+1} - s_n - \Delta t g(\tilde{\sigma}_{n+1}, s_{n+1}) = 0$$

8. Obtain the radial return factor η_{n+1}

$$\eta_{n+1} = \frac{\tilde{\sigma}_{n+1}}{\tilde{\sigma}_*}$$

9. Update the stress $\bar{\mathbf{T}}_{n+1}$

$$\bar{\mathbf{T}}_{n+1} = \eta_{n+1} \bar{\mathbf{T}}_*' - \bar{p}_* \mathbf{I}$$

10. Transform $\bar{\mathbf{T}}_{n+1}$ to \mathbf{T}_{n+1}

$$\mathbf{T}_{n+1} = \exp\left(\frac{\bar{p}_*}{\kappa}\right) \mathbf{R}_*^e \bar{\mathbf{T}}_{n+1} \mathbf{R}_*^{eT}$$

11. Update \mathbf{F}_{n+1}^e

$$\mathbf{F}_{n+1}^e = \mathbf{R}_*^e \left(\sum \frac{(\lambda_*^i)^{\eta_{n+1}}}{(\lambda_*^1 \lambda_*^2 \lambda_*^3)^{(\eta_{n+1}-1)/3}} \mathbf{e}_*^i \otimes \mathbf{e}_*^i \right)$$

4 Variational formulation and solution procedure

Let us suppose that the configuration \mathbf{B}_n of the body at time $t = t_n$ is known and is under equilibrium with the forces acting on it. Then the incremental quasi-static boundary value problem at time $t = t_{n+1}$ is to find the incremental (with respect to the configuration \mathbf{B}_n) displacement field $\mathbf{u}(\mathbf{x}_n, t_{n+1}) \equiv \mathbf{u}_{n+1}$ such that

$$G(\mathbf{u}_{n+1}, \tilde{\mathbf{u}}(\mathbf{x}_n)) = \int_{\mathbf{B}_n} \mathbf{P}_u \cdot \frac{\partial \tilde{\mathbf{u}}}{\partial \mathbf{x}_n} dV - \left(\int_{\partial \mathbf{B}_{n+1}} \hat{\mathbf{t}} \cdot \tilde{\mathbf{u}} ds + \int_{\mathbf{B}_{n+1}} \hat{\mathbf{b}} \cdot \tilde{\mathbf{u}} dV \right) = 0 \quad (44)$$

for each test vector field $\tilde{\mathbf{u}}(\mathbf{x}_n)$, which is zero on the portion of the boundary where kinematic boundary conditions are applied. The above equation is a mixed form of the principle of virtual work. The internal work is expressed in the reference configuration \mathbf{B}_n using the Piola-Kirchoff I stress, $\mathbf{P}_u = \det \mathbf{F}_u \mathbf{T}_{n+1} \mathbf{F}_u^{-T}$, while the external work is expressed in the current configuration where the applied surface tractions, $\hat{\mathbf{t}}$, and body forces, $\hat{\mathbf{b}}$, are given.

In order to solve the above set of non-linear equations for the incremental displacement field $\mathbf{u}(\mathbf{x}_n, t_{n+1})$, an iterative scheme is used (figure (3)). In the present work, a Newton-Raphson scheme is employed, which requires linearization of the above equation about the unknown field \mathbf{u}_{n+1} . Let $\mathbf{u}_{n+1}^{(k+1)}$ be the displacement field at the end of the $(k+1)^{th}$ step and $\mathbf{u}_{n+1}^{(k)}$ be the displacement field at the end of the k^{th} step during the Newton-Raphson iterative process. Then, linearizing the equation (44), at any iteration step k , we obtain

$$G(\mathbf{u}_{n+1}^{(k)}, \tilde{\mathbf{u}}) + \frac{\partial G}{\partial \mathbf{u}_{n+1}^{(k)}} (\mathbf{u}_{n+1}^{(k+1)} - \mathbf{u}_{n+1}^{(k)}) = 0 \quad (45)$$

where

$$dG = \int_{\mathbf{B}_n} d\mathbf{P}_u \cdot \frac{\partial \tilde{\mathbf{u}}}{\partial \mathbf{x}_n} dV + d(\text{bodyforces} + \text{applied traction}) \quad (46)$$

The derivative of body forces and applied traction, and their contribution to the stiffness matrix has been described in [7]. We however, treat the external applied tractions and body forces in a different manner (see Appendix B). Here, we will consider only the first term in the above equation.

$$d\mathbf{P}_u = d(\det \mathbf{F}_u \mathbf{T}_{n+1} \mathbf{F}_u^{-T})$$

which results in

$$d\mathbf{P}_u = d(\det \mathbf{F}_u) \mathbf{T}_{n+1} \mathbf{F}_u^{-T} + \det \mathbf{F}_u d\mathbf{T}_{n+1} \mathbf{F}_u^{-T} + \det \mathbf{F}_u \mathbf{T}_{n+1} d\mathbf{F}_u^{-T} \quad (47)$$

Now,

$$d(\det \mathbf{F}_u) = \det \mathbf{F}_u \text{tr} (d\mathbf{F}_u \mathbf{F}_u^{-1})$$

and

$$d\mathbf{F}_u^{-T} = - (d\mathbf{F}_u \mathbf{F}_u^{-1})^T \mathbf{F}_u^{-T}$$

Hence

$$d\mathbf{P}_u = \det \mathbf{F}_u \left[\text{tr} (d\mathbf{F}_u \mathbf{F}_u^{-1}) \mathbf{T}_{n+1} + d\mathbf{T}_{n+1} - \mathbf{T}_{n+1} (d\mathbf{F}_u \mathbf{F}_u^{-1})^T \right] \mathbf{F}_u^{-T} \quad (48)$$

We have to evaluate $d\mathbf{T}_{n+1}$ now. Differentiating equation (39) we obtain

$$\begin{aligned} d\mathbf{T}_{n+1} = \exp\left(\frac{p_*}{\kappa}\right) & \left(-\frac{1}{3\kappa} \text{tr}(d\bar{\mathbf{T}}) \mathbf{R}_*^\varepsilon \bar{\mathbf{T}}_{n+1} (\mathbf{R}_*^\varepsilon)^T + \mathbf{R}_*^\varepsilon \bar{\mathbf{T}}_{n+1} d(\mathbf{R}_*^\varepsilon)^T + \right. \\ & \left. \mathbf{R}_*^\varepsilon d\bar{\mathbf{T}}_{n+1} (\mathbf{R}_*^\varepsilon)^T + d\mathbf{R}_*^\varepsilon \bar{\mathbf{T}}_{n+1} (\mathbf{R}_*^\varepsilon)^T \right) \end{aligned} \quad (49)$$

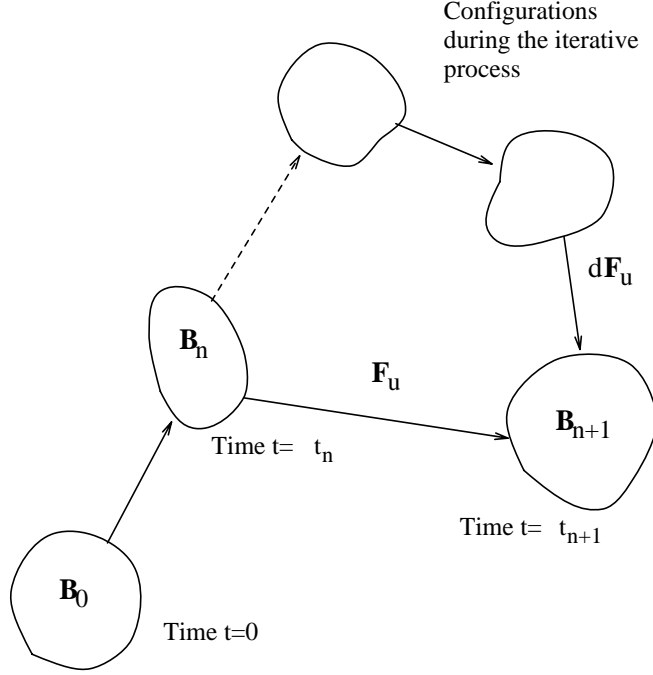


Figure 3: Newton-Raphson iteration scheme

Here

$$d\bar{\mathbf{T}}_{n+1} = \frac{\partial \bar{\mathbf{T}}_{n+1}}{\partial \bar{\mathbf{E}}_*^e} [d\bar{\mathbf{E}}_*^e] = \mathcal{C} [d\bar{\mathbf{E}}_*^e]$$

where the elastoplastic linearized moduli \mathcal{C} are defined as

$$\mathcal{C} \equiv \frac{\partial \bar{\mathbf{T}}_{n+1}}{\partial \bar{\mathbf{E}}_*^e}$$

Now we have to evaluate $d\bar{\mathbf{E}}_*^e$, $d\mathbf{R}_*^e$ and the moduli \mathcal{C} . Let us first consider $d\bar{\mathbf{E}}_*^e$.

$$\bar{\mathbf{E}}_*^e = \ln \mathbf{U}_*^e$$

By using first Padé approximation of $\ln \mathbf{U}_*^e$, we can approximate $\bar{\mathbf{E}}_*^e$ as

$$\bar{\mathbf{E}}_*^e \simeq 2(\mathbf{U}_*^e - \mathbf{I})(\mathbf{U}_*^e + \mathbf{I})^{-1} \quad (50)$$

This approximation is valid in the range of $0.7 < \lambda_* < 1.2$, where λ_* is the stretch (eigen value) corresponding to \mathbf{U}_*^e . Differentiating this, we obtain

$$d\bar{\mathbf{E}}_*^e = 4(\mathbf{U}_*^e + \mathbf{I})^{-1} d\mathbf{U}_*^e (\mathbf{U}_*^e + \mathbf{I})^{-1} \quad (51)$$

In order to evaluate $d\mathbf{U}_*^e$, we have

$$(\mathbf{U}_*^e)^2 = (\mathbf{F}_n^e)^T (\mathbf{F}_u)^T \mathbf{F}_u \mathbf{F}_n^e$$

which upon differentiation and simplification results in,

$$\mathbf{U}_*^e d\mathbf{U}_*^e + d\mathbf{U}_*^e \mathbf{U}_*^e = 2 (\mathbf{F}_n^e)^T \text{sym} \left((\mathbf{F}_u^e)^T d\mathbf{F}_u^e \right) \mathbf{F}_n^e \quad (52)$$

Here, ‘‘sym’’ refers to the symmetric part of the tensor. Consider the Biot strain,

$$\mathbf{E}_*^B \simeq \mathbf{U}_*^e - \mathbf{I}$$

Upon expanding $(\mathbf{U}_*^e)^{-1}$ in terms of \mathbf{E}_*^B we arrive at

$$(\mathbf{U}_*^e)^{-1} = \mathbf{I} - \mathbf{E}_*^B + \mathcal{O}(\mathbf{E}_*^{B^2}) \quad (53)$$

Now, premultiplying equation (52) by $(\mathbf{U}_*^e)^{-1}$, substituting equation (53) and taking the symmetric part on both the sides, we arrive at

$$d\mathbf{U}_*^e = \text{sym} \left((\mathbf{U}_*^e)^{-1} (\mathbf{F}_n^e)^T \text{sym} \left(\mathbf{F}_u^T d\mathbf{F}_u \right) \mathbf{F}_n^e \right) + \mathcal{O}(\mathbf{E}_*^{B^2}) \quad (54)$$

Substituting the above equation in equation (51) we obtain

$$d\bar{\mathbf{E}}_*^e = 4(\mathbf{U}_*^e + \mathbf{I})^{-1} \text{sym} \left((\mathbf{U}_*^e)^{-1} (\mathbf{F}_n^e)^T \text{sym} \left(\mathbf{F}_u^T d\mathbf{F}_u \right) \mathbf{F}_n^e \right) (\mathbf{U}_*^e + \mathbf{I})^{-1} + \mathcal{O}(\mathbf{E}_*^{B^2}) \quad (55)$$

Similarly for evaluating $d\mathbf{R}_*^e$

$$\mathbf{R}_*^e \mathbf{U}_*^e = \mathbf{F}_*^e = \mathbf{F}_u \mathbf{F}_n^e$$

Hence differentiating,

$$d\mathbf{R}_*^e \mathbf{U}_*^e + \mathbf{R}_*^e d\mathbf{U}_*^e = d\mathbf{F}_u \mathbf{F}_n^e$$

Now substituting equation (54) and simplifying the above, we arrive at:

$$d\mathbf{R}_*^e (\mathbf{R}_*^e)^T = d\mathbf{F}_u \mathbf{F}_u^{-1} - \mathbf{R}_*^e \text{sym} \left((\mathbf{U}_*^e)^{-1} (\mathbf{F}_n^e)^T \text{sym} \left(\mathbf{F}_u^T d\mathbf{F}_u \right) \mathbf{F}_n^e \right) (\mathbf{U}_*^e)^{-1} (\mathbf{R}_*^e)^T \quad (56)$$

Thus, $d\mathbf{P}_u$ can now be written as

$$\begin{aligned} d\mathbf{P}_u &= \det \mathbf{F}_u \left\{ \text{tr} \left(d\mathbf{F}_u \mathbf{F}_u^{-1} \right) \mathbf{T}_{n+1} - \mathbf{T}_{n+1} \left(d\mathbf{F}_u \mathbf{F}_u^{-1} \right)^T - \text{tr} \left(\frac{1}{3\kappa} \mathbf{C} \left[d\bar{\mathbf{E}}_*^e \right] \right) \mathbf{T}_{n+1} + \right. \\ &\left. \exp \left(\frac{p_*}{\kappa} \right) \mathbf{R}_*^e \mathbf{C} \left[d\bar{\mathbf{E}}_*^e \right] (\mathbf{R}_*^e)^T + \left(d\mathbf{R}_*^e (\mathbf{R}_*^e)^T \right) \mathbf{T}_{n+1} - \mathbf{T}_{n+1} \left(d\mathbf{R}_*^e (\mathbf{R}_*^e)^T \right)^T \right\} \mathbf{F}_u^{-T} \quad (57) \end{aligned}$$

The linearized moduli \mathbf{C} .

Consider equation (24) with $\dot{\epsilon}^p = f(\tilde{\sigma}_{n+1}, s_{n+1})$. Differentiating this equation with respect to $\bar{\mathbf{E}}_*^e$, we have (we drop the functional dependency of the function f for convenience)

$$\mathbf{C} = \frac{\partial \bar{\mathbf{T}}_*}{\partial \bar{\mathbf{E}}_*^e} - 3\mu \Delta t \frac{\partial}{\partial \bar{\mathbf{E}}_*^e} \left(\frac{f}{\dot{\sigma}_*} \bar{\mathbf{T}}_*' \right)$$

which results in

$$\mathbf{C} = \mathcal{L}^e - 3\mu\Delta t \left[\frac{f}{\tilde{\sigma}_*} \frac{\partial \bar{\mathbf{T}}_*'}{\partial \bar{\mathbf{E}}_*^e} + \frac{\partial}{\partial \bar{\mathbf{E}}_*^e} \left(\frac{f}{\tilde{\sigma}_*} \right) \otimes \bar{\mathbf{T}}_*' \right] \quad (58)$$

Here,

$$\frac{\partial \bar{\mathbf{T}}_*'}{\partial \bar{\mathbf{E}}_*^e} = \mathcal{L}^e - \frac{1}{3} \text{tr}(\mathcal{L}^e[\mathbf{I}]) \mathbf{I} \otimes \mathbf{I} \equiv \mathcal{L}' \quad (59)$$

Differentiating the equations (16) and (29) and simplifying, we obtain

$$d\tilde{\sigma}_{n+1} = c d\tilde{\sigma}_* \quad (60)$$

where

$$\begin{aligned} c &= \frac{b_2}{a_1 b_2 + a_2 b_1} \\ a_1 &\equiv 1 + 3\mu\Delta t \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \\ a_2 &\equiv 3\mu\Delta t \frac{\partial f}{\partial s_{n+1}} \\ b_1 &\equiv \Delta t \frac{\partial g}{\partial \tilde{\sigma}_{n+1}} \\ b_2 &\equiv 1 - \Delta t \frac{\partial g}{\partial s_{n+1}} \end{aligned}$$

Differentiation of equation (27), results in

$$\frac{\partial \tilde{\sigma}_*}{\partial \bar{\mathbf{E}}_*^e} = \frac{3}{2\tilde{\sigma}_*} \mathcal{L}'[\bar{\mathbf{T}}_*'] \quad (61)$$

The above equation, along with equation (60) results in

$$\frac{\partial}{\partial \bar{\mathbf{E}}_*^e} \left(\frac{f}{\tilde{\sigma}_*} \right) = \alpha \mathcal{L}'[\bar{\mathbf{T}}_*'] \quad (62)$$

where

$$\alpha = \frac{1-c}{2\mu\Delta t \tilde{\sigma}_*^2} - \frac{3f}{2\tilde{\sigma}_*^3}$$

Now we can write \mathbf{C} as

$$\mathbf{C} = \mathcal{L}^e - 3\mu\Delta t \left[\frac{f}{\tilde{\sigma}_*} \mathcal{L}' + \alpha \mathcal{L}'[\bar{\mathbf{T}}_*'] \otimes \bar{\mathbf{T}}_*' \right] \quad (63)$$

For an isotropic material, the linearized moduli can be further simplified as given in Appendix C.

5 Finite element implementation

A finite element program has been written in order to test the algorithm described above. Both axially symmetric and plane strain problems can be solved by using the code. The domain of the problem is discretized by a finite element mesh and the displacement field is expressed by means of the finite element shape functions. The nodal displacements along with the finite element shape functions describe the displacement field. The boundary conditions are either fixed displacements or tractions. The analysis involves obtaining the incremental displacements by solving the kinematic problem and then solving for the stresses and the state variables at each stage. The Newton-Raphson linearizations then produce a system of linearized algebraic equations for the incremental nodal displacements. The stiffness matrix is formed by computing the global Jacobian matrix. The various tensor operations are converted to equivalent operations of square matrices on column vectors so that the unknown components of the incremental deformation gradient tensor or the incremental nodal displacements can be expressed as a column vector (see appendix D). The combined effect of various operations then automatically results in the global Jacobian. The global Jacobian essentially consists of three terms: the elasto-plastic stiffness term $[K]$, the stiffness $[K_G]$ due to other terms in the expansion of $d\mathbf{P}_u$ in equation (57) and the stiffness term $[K_L]$ due to the linearization of follower forces [Appendix B]. Some of the important features of the finite element program are described below.

The convergence criteria

During each time increment, the incremental plastic work

$$\Delta W^p = \int_{t_n}^{t_{n+1}} \int_{\mathbf{B}_n} \tilde{\sigma} \dot{\tilde{\epsilon}}^p dV dt$$

is obtained. A simple trapezoidal rule is used to calculate the time integral above and the integration over the domain is performed by the finite element discretization using gaussian quadrature. This incremental plastic work is used as the criteria for convergence of the Newton-Raphson iterations. Within a given time step, for each of the Newton-Raphson iterations, the incremental plastic work is evaluated and compared with the corresponding value in the previous iteration. The Newton-Raphson iterations are assumed to be converged when the incremental plastic work in two successive iterations falls within 1% (or any suitable tolerance).

An automatic time stepping algorithm

Whenever needed, an automatic time stepping algorithm discussed in [3] is employed so that the incremental plastic strain in every time increment is kept close to a prescribed fixed value. Let $\Delta \dot{\tilde{\epsilon}}_s^p$ be the prescribed value and $\Delta \dot{\tilde{\epsilon}}_{max}^p$ the maximum attainable effective plastic strain increment over the previous time step. Then the factor R

$$R \equiv \frac{\Delta \dot{\tilde{\epsilon}}_{max}^p}{\Delta \dot{\tilde{\epsilon}}_s^p}$$

If $R > 1.25$, then the solution is rejected, a new time increment which is smaller by a factor of $(0.85/R)$ is taken and the whole step is repeated. If $R \leq 1.25$, then the solution is accepted and the time increment for the subsequent step is determined so that the ratio R is made close to 1.0. The following algorithm is used.

If $0.8 < R \leq 1.25$ then $\Delta t_{n+1} = \Delta t_n / R$
 If $0.5 < R \leq 0.80$ then $\Delta t_{n+1} = 1.25 \Delta t_n$
 If $R \leq 0.50$ then $\Delta t_{n+1} = 1.50 \Delta t_n$

The contact algorithm for a forging die

The driving force for most of the metal forming problems is the contact at the die-workpiece interface. The die is modelled as rigid. In case of closed or open die forging, a few boundary nodes will be in contact with the die initially. As deformation progresses, more and more nodes will come into contact. One can model the friction conditions at the interface in many ways depending upon the nature of the problem. In case of forging, we model the contact friction as sticking friction. When a node comes into contact with the die, the node “sticks” to the die and does not have any relative displacement with respect to the die.

A list of nodes, which are likely to come into contact with the die surface will be provided by the user. These nodes are monitored during each Newton-Raphson iteration within a time step. If during one of the Newton-Raphson iterations, few nodes cross the die surface, then the iterations for the entire time step are repeated after assigning the nodes the correct amount of displacement boundary conditions so that at the end of the time step, these nodes come into contact with the die exactly. Then, these nodes are deleted from the list we maintain.

Because the nodes come into contact with the dies in an abrupt manner, if we change the time step for the problem, the behavior of the nodes coming into contact with the die will be slightly different.

The forging dies can be of any shape. For the finite element code to recognize the die shape, the equation of the die surface should be given in a user defined function with the name *die_eq(x, y)* where (x, y) are the coordinates of a point on the die. Further, the equation of the die should be given in the following format:

Let the coordinate axes be as shown in figure (4). Let us assume that the equation of the die surface can be written as

$$y = g(x)$$

Then, the function *die* is written as

$$die = f(x, y) = -y + g(x)$$

For every point (x, y) on the die surface, the function *die* (x, y) returns zero. If the value of the function is positive, then the point has not yet come into contact with the die and if negative, the point has crossed the die.

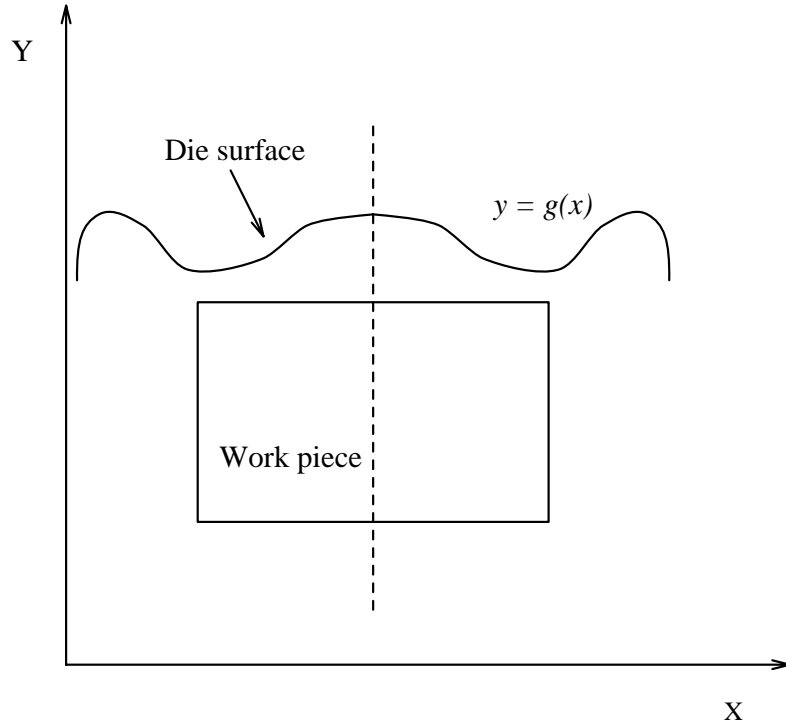


Figure 4: A curved forging die in axisymmetric forging

The contact algorithm for an extrusion problem

In case of extrusion, we assume a sliding friction with a constant friction coefficient. As before, the equation of the extrusion die has to be specified in a user defined subroutine. In extrusion problems, we have to force the boundary nodes that are in contact with the die to follow the given die surface. In reasonably large deformations, imposing this constraint becomes very important. Because the die is considered rigid, the average normal displacement of the nodes touching the die are zero and the tangential traction is determined by the friction coefficient, normal traction and the direction of motion.

To write down the boundary condition, the direction of the normal to the displacement vector has to be calculated. This is done by first estimating approximately the displacement vector of the boundary nodes during the first Newton-Raphson iteration in a time step (since the displacement is the unknown in the problem). We estimate the displacement vector in the following way:

For every node that is following the die, give an axial displacement equal to the displacement at the entry. Then, evaluate the position of the point by forcing the point to lie on the die surface. From the new position obtained and the original position coordinates, obtain the displacement vector.

During subsequent iterations in the same time step, the displacement vector calculated at the end of the previous iteration is employed. Notice that this approximation is used only

to estimate the normal to the displacement and thus to apply the boundary condition. Thus when the solution converges, the boundary condition would have also been applied correctly.

If the coefficient of friction at the die-workpiece interface is given a value zero, we can simulate the frictionless extrusion process.

Boundary Traction

In displacement controlled problems, evaluation of the boundary tractions are quite important. In forging problem, for example, we will be interested in calculating the force required for the process at each time. This forging force is calculated by integrating the traction forces along the die surface. From the finite element solution obtained, stresses are evaluated at all the gauss points. Then, from these values, nodal stresses are obtained by a global least squares interpolation. Now, the stress value at any point inside a given element is evaluated using the nodal values and the finite element shape functions. Then, the force integration is performed either by using a trapezoidal rule like integration scheme or by using gauss integration over each boundary segment that falls in an element and summing them.

The finite element algorithm

A brief sketch of the algorithm for the finite element program is given below. For a generic $(n + 1)^{th}$ step,

1. Initialize the $(n + 1)^{th}$ configuration to that of n^{th} configuration. Start the Newton-Raphson iteration:
2. Obtain the stiffness and the unbalanced forces, element by element and assemble them to get the global Jacobian and unbalanced load vector for the k^{th} step.
 - Evaluate the deformation gradient at each gauss point on a given element.
 - Evaluate the consistent tangent moduli.
 - Evaluate the element stiffness by calculating the terms inside the integral in equation (57).
 - Evaluate the load correction from the previous iteration from the Piola-Kirchoff stress.
 - Evaluate the stiffness and the force term due to the boundary tractions and the body forces.
 - Assemble the global stiffness matrix and force vector.
 - Impose the displacement boundary constraints.
3. Solve for the incremental displacements.
4. Check for the convergence criterion based on the incremental plastic work. If not converged repeat from step 2 to 4. If converged, goto step 5.
5. Compute the stress and the state variables.

6. Check if the current time is within the final time.
7. Update the current configuration and goto step 1.

6 Numerical examples

The constitutive models used

Two constitutive models were predominantly used for solving all the numerical examples. They are given below.

Model 1: Power law with no hardening

The flow function is given by

$$f(\tilde{\sigma}, s) = \tilde{\epsilon}_0 \left(\frac{\tilde{\sigma}}{s_0} \right)^{1/m} \quad (64)$$

and the hardening law

$$g(\tilde{\sigma}, s) = 0 \quad (65)$$

The specific values for the parameters that were used in the numerical examples are:

$$\begin{aligned} s_0 &= 0.006 \\ \nu &= 0.499 \\ \tilde{\epsilon}_0 &= 0.002 \text{ sec}^{-1} \\ m &= 0.2 \\ E &= 1 \end{aligned} \quad (66)$$

where E is the Young's modulus and ν , the Poisson's ratio.

Model 2: Anand's sin hyperbolic model for Al 1100-O at 673 K

The flow rule is given by

$$f(\tilde{\sigma}, s) = A \left(\sinh \left(\xi \frac{\tilde{\sigma}}{s} \right) \right)^{1/m} \quad (67)$$

and the hardening function is given by

$$g(\tilde{\sigma}, s) = h(\tilde{\sigma}, s) f(\tilde{\sigma}, s) \quad (68)$$

where the function h is given by

$$h(\tilde{\sigma}, s) = h_0 \left| 1 - \frac{s}{s^*} \right|^a \quad (69)$$

and

$$s^* = \tilde{s} \left(\frac{\dot{\tilde{\epsilon}}^p}{A} \right)^n \quad (70)$$

Material Parameter	Value
A	$4.75 \times 10^{-7} \text{ sec}^{-1}$
ξ	7.0
m	0.23348
s_0	29.7 MPa
h_0	1115.6 MPa
a	1.3
\tilde{s}	18.92 MPa
n	0.07049
μ	20.2 GPa
κ	66.0 GPa

Table 1: Material parameters for Al 1100-O at 673 K

The specific values of the material parameters are given in table(1).

Example 1: Uniaxial tension to test the accuracy of the integration scheme

The performance of the time-integration procedure for the chosen constitutive function was first evaluated using a uniaxial simulation. A simple finite element mesh with 16, 3 noded triangular elements were considered for an axisymmetric case. A constant strain rate was employed and the boundary nodes were given constant incremental displacement. The finite element mesh used is shown in figure (5). Three different equivalent plastic strain increment were applied by having three different prescribed incremental displacements. The material model 2 described above was used for modeling. The results are presented in figure (6). The results agree well with the ones presented in [3]. Based on the results, we can clearly see that if the plastic strain increment stays within 5%, the accuracy of the algorithm is very high. Hence wherever automatic time stepping algorithm is used, the upper limit on maximum plastic strain increment is maintained at 5%.

Example 2: Cylinder under a constant radial displacement rate at the inner wall

Here the numerical procedure is used to carry out the creep analysis of a thick-walled cylinder with a specified constant radial displacement rate at the inner wall. The geometry of the problem is given in figure (6). The material model 1 given above is chosen. The ‘steady-state’ solution for the radial stress can be obtained in a closed form as

$$\mathbf{T}_{rr} = \left[\left\{ \frac{s_0}{\tilde{\epsilon}_0^m} \right\} \left\{ \frac{2a\dot{U}}{\sqrt{3}b^2} \right\}^m \frac{1}{m\sqrt{3}} \right] \left[1 - \left(\frac{b}{r} \right)^{2m} \right] \quad (71)$$

Here a and b are the radii of the inner and outer walls of the cylinder, r is the radial coordinate and \dot{U} is a constant rate of expansion prescribed at the inner radius $r = a$. When plastic deformations are large compared to the elastic deformations, the solution of

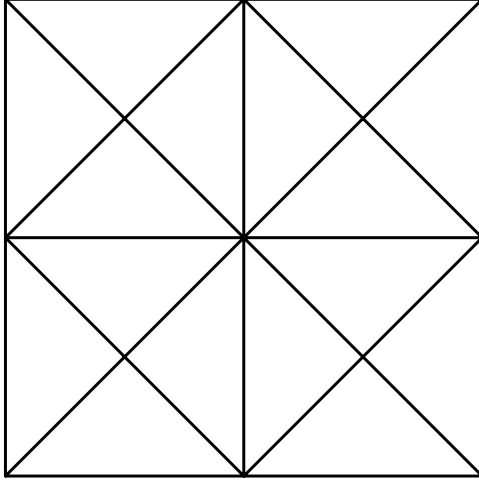


Figure 5: Mesh for the axisymmetric tension example

Figure 6: Uniaxial results for Al 1100-O at 673 K with a constant strain rate of 0.01 s

our finite element simulations should compare well with the above equation. The selection of a Poisson's ratio close to 0.5 results in material being almost incompressible. The finite element mesh chosen is given in figure (8). With this mesh, two simulations are done, one for small values of $(U/a_0\tilde{\epsilon}_0)$ and the other for large values of the same. In both the cases the quantity $(\dot{U}/a_0\tilde{\epsilon}_0)$ was equal to 1.0. The pressure at the inner radius $p = -\mathbf{T}_{rr} |_{r=a}$ is plotted in figures (9) and (10). The results compare well with the analytical solution.

Example 3. Upset forging

As a simple example of a large deformation metal-forming problem, the isothermal upset forging of a cylindrical billet is considered. The dies are modelled as rigid surfaces and sticking friction is assumed between the die and the work piece during contact. The material is modeled as model 2 described above. The deformation is non-homogeneous. The symmetry of the problem enabled modeling of just one fourth of the geometry. A cylinder of height 3 mm and diameter 2 mm was considered. A nominal strain rate of 0.01 was applied. The finite element mesh after 60% of deformation is shown in figure (11) along with the undeformed mesh. An automatic time-stepping algorithm [2] was used so that the increment in effective plastic strain does not go beyond 0.05%. Figures (12) and (13) show the contours of equivalent plastic strain and the internal state variable respectively. The load versus deflection curve is shown in figure (14). The results compare well with [2,3].

Example 4. Extrusion problem

Another important metal forming problem considered is an axisymmetric extrusion of a workpiece through a die with the reduction in area of 25%. The die is modelled as a quintic polynomial which has zero slope and zero curvature at the inlet and the outlet. The workpiece was pushed through the die at a constant strain rate of 0.01sec^{-1} . The material was modelled based on model 2 described above. The deformed mesh is shown in figure (15). The contours of equivalent plastic strain and the internal variable are shown in figures (16) and (17).

7 Conclusion

An updated lagrangian hyperelastic viscoplastic formulation for the analysis of large deformations problem was presented. A radial return integration scheme and full linearization of the principle of virtual work employing the consistent material moduli was implemented. A finite element program was developed incorporating the above theory. The performance of this finite element code in solving large deformations problem was demonstrated by means of many numerical examples. Further, with reasonable modifications, this code can be used in solving inverse and design boundary value problems in metal forming processes.

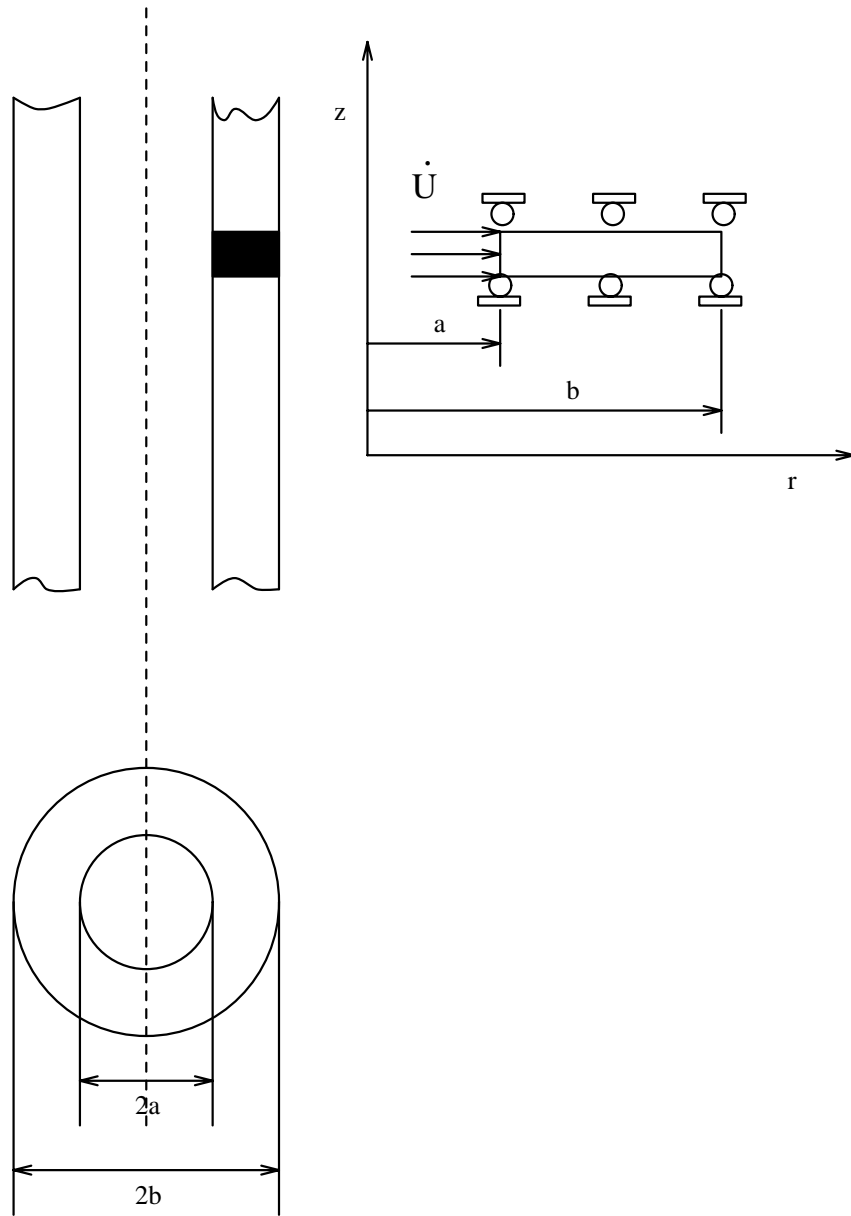


Figure 7: Schematic of a thick-walled cylinder under prescribed velocity conditions at its inner wall

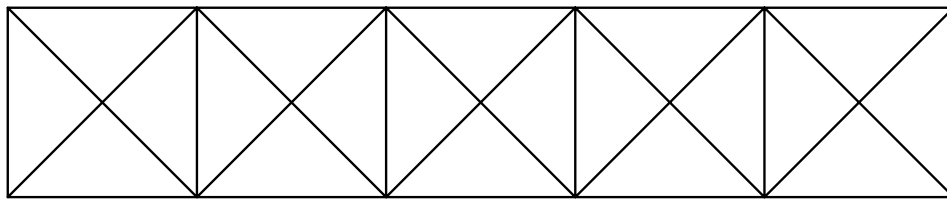


Figure 8: Mesh for the thick walled cylinder example

Figure 9: Normalized pressure (p/s_0) versus normalized ‘strain’ ($U/a_0\tilde{\epsilon}_0$) for a thick-walled cylinder with prescribed radial velocity \dot{U} at its inner wall. Small values of ($U/a_0\tilde{\epsilon}_0$). The finite element solution is obtained by variable time stepping.

Figure 10: Normalized pressure (p/s_0) versus normalized ‘strain’ ($U/a_0\tilde{\epsilon}_0$) for a thick-walled cylinder with prescribed radial velocity \dot{U} at its inner wall. Large values of ($U/a_0\tilde{\epsilon}_0$). The finite element solution is obtained by taking fixed values of $\Delta U/a_0 = 10\%$.

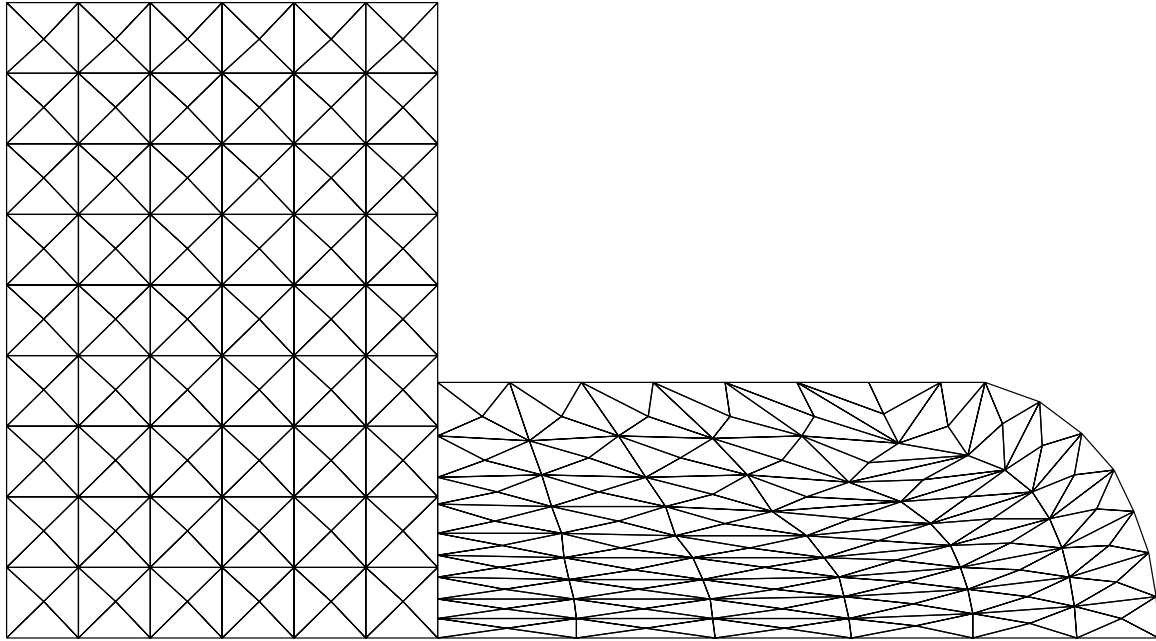


Figure 11: Finite element mesh for an axisymmetric upsetting problem. Deformed mesh is shown after 60% upsetting.

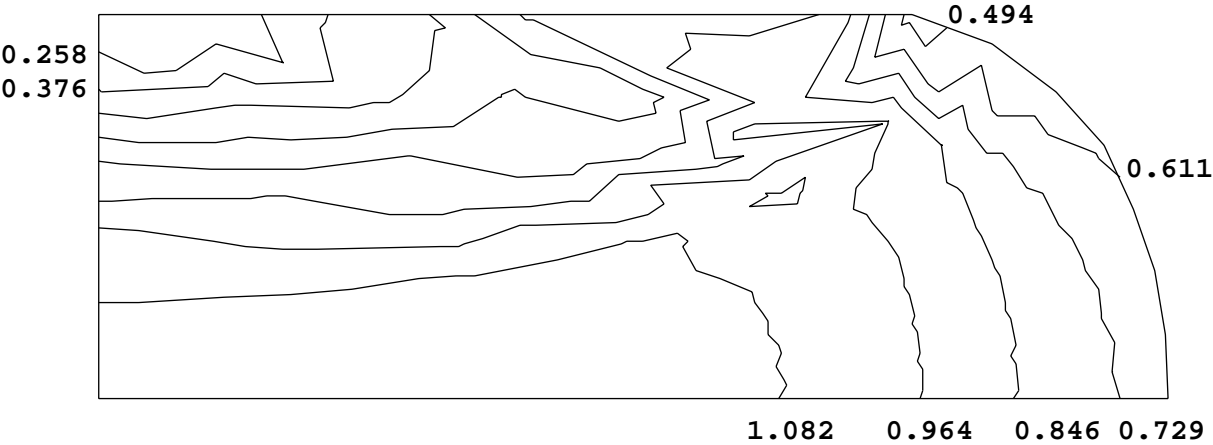


Figure 12: Contours of effective plastic strain $\tilde{\epsilon}^p$ after 60% upsetting.

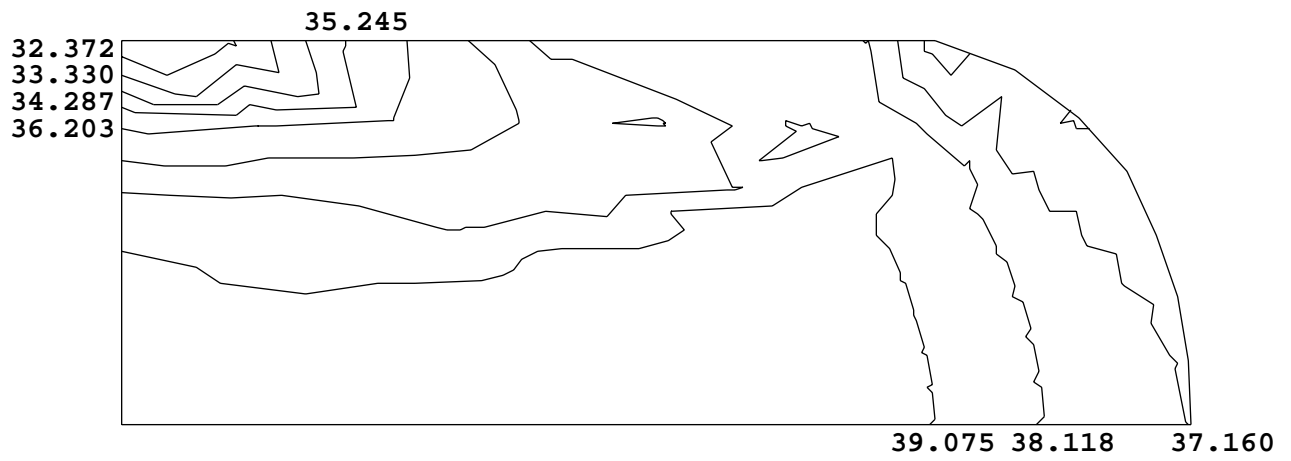


Figure 13: Contours of internal state variable s after 60% upsetting

Figure 14: Load versus deflection for the axisymmetric upsetting problem

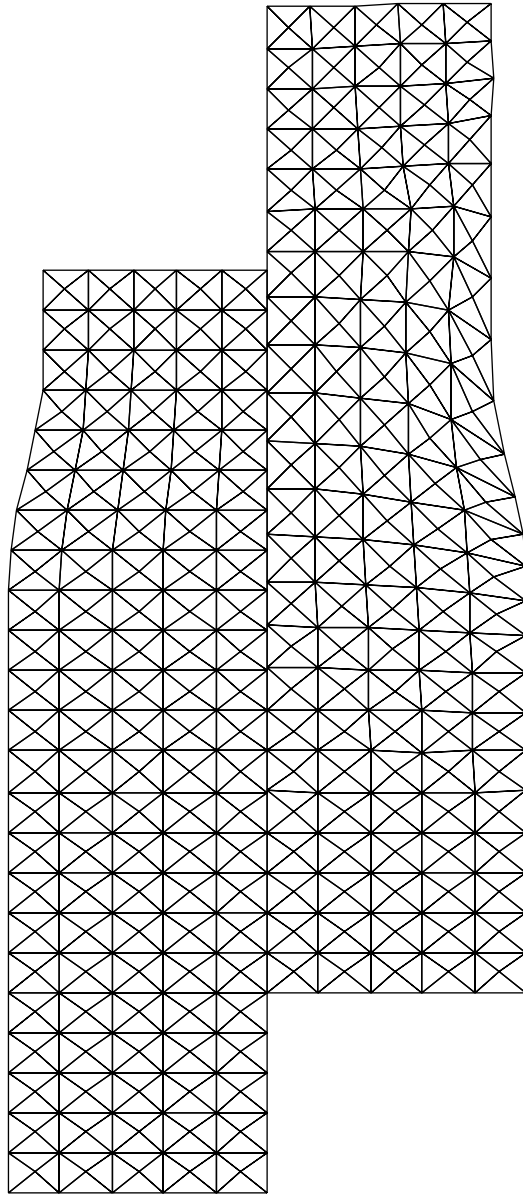


Figure 15: Finite element mesh for an axisymmetric extrusion problem



Figure 16: Contours of effective plastic strain $\tilde{\epsilon}^p$ at the end of extrusion

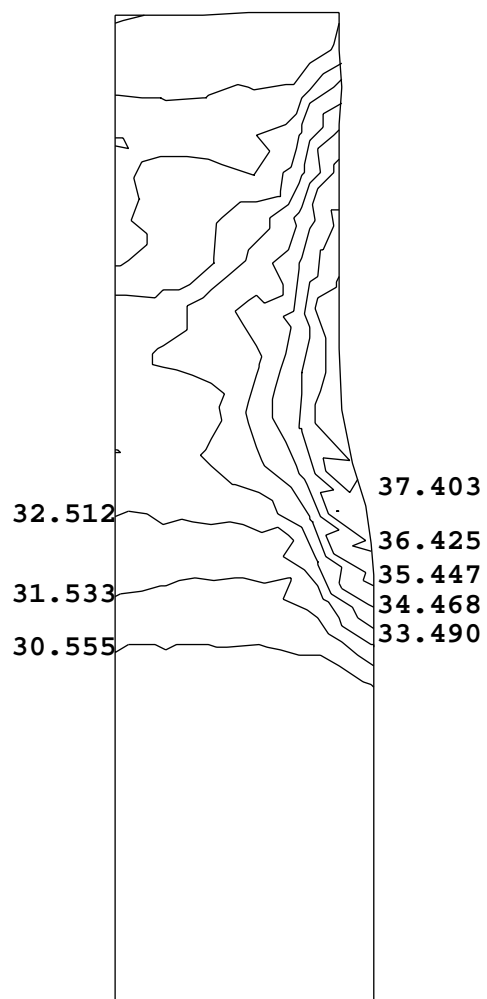


Figure 17: Contours of internal state variable s at the end of extrusion

8 Acknowledgements

This work was funded by NSF grant DDM-9157189 to Cornell University. The computing for this project was supported by the Cornell National Supercomputing facility, which receives major funding by the NSF and IBM corporation, with additional support from New York State.

References

- [1] L. Anand, Constitutive equations for hot-working of metals, *International Journal of Plasticity* 1 (1985) 213-231
- [2] A.M. Lush, G. Weber, and L. Anand, An implicit time-integration procedure for a set of internal variable constitutive equations for isotropic elasto-viscoplasticity, *International Journal of Plasticity* 5 (1989) 521-549
- [3] G. Weber and L. Anand, Finite deformation constitutive equations and a time integration procedure for isotropic, hyperelastic-viscoplastic solids, *Computer Methods in Applied Mechanics and Engineering* 79 (1990) 173-202
- [4] N. Zabaras and A.F.M. Arif, A family of integration algorithms for constitutive equations in finite deformation elasto-viscoplasticity, *International Journal for Numerical Methods in Engineering*, 33 (1992) 59-84
- [5] A.F.M. Arif and N. Zabaras, On the performance of two tangent operators for finite element analysis of large deformation inelastic problems, *International Journal for Numerical Methods in Engineering*, 35 (1992) 369-389
- [6] E.H. Lee and D.T. Liu, Finite-Strain elastic-plastic theory with application to plane-wave analysis, *Journal of Applied Physics*, 38 (1967) 19-27
- [7] H.D. Hibbit, Some follower forces and load stiffness, *International Journal for Numerical Methods in Engineering*, 14 (1979) 937-941
- [8] H. Flanders, Differentiation under the integral sign, *American Mathematical Monthly*, 80 (1973) 615-627

A Solution to the simultaneous equations by a modified Newton-Raphson scheme

In order to solve the nonlinear simultaneous system of algebraic equations arising out of the radial return method, the following method (given in [2]) is employed.

We have to solve the following two equations:

$$s_{n+1} - s_n - \Delta t g(\tilde{\sigma}_{n+1}, s_{n+1}) = 0 \quad (72)$$

$$\tilde{\sigma}_{n+1} - \tilde{\sigma}_* + 3\mu \Delta t f(\tilde{\sigma}_{n+1}, s_{n+1}) = 0 \quad (73)$$

These equations can be solved only by iterative means and because of their highly non-linear nature, the choice of the algorithm will heavily influence the number of iterations needed for convergence. To start with, an initial estimate of the solution is needed and starting from that point, linearization followed by solving the linear equations will provide the answer. The Newton-Raphson method is not applied on both equations simultaneously but for one equation at a time. A concise algorithm is given below.

1. First, the tolerances required for the convergence on $\tilde{\sigma}_{n+1}$ and s_{n+1} are fixed. A good choice is

$$\text{TOL}_s = 1 \times 10^{-5} s_n$$

$$\text{TOL}_\sigma = 1 \times 10^{-5} \tilde{\sigma}_*$$

2. Make an initial estimate of $(\tilde{\sigma}_{n+1}, s_{n+1})$ by solving the equations obtained by a forward gradient approximations of the governing equations. The functions f and g are approximated by Taylor series expansion about $(\tilde{\sigma}_n, s_n)$. This results in

$$\tilde{\sigma}_{n+1} \simeq \tilde{\sigma}_n + \left\{ \frac{b_2 (\tilde{\sigma}_* - \tilde{\sigma}_n - 3\mu \Delta t f(\tilde{\sigma}_n, s_n)) - a_2 \Delta t g(\tilde{\sigma}_n, s_n)}{b_2 a_1 + a_2 b_1} \right\}$$

and

$$s_{n+1} \simeq s_n + \left\{ \frac{b_1 (\tilde{\sigma}_* - \tilde{\sigma}_n - 3\mu \Delta t f(\tilde{\sigma}_n, s_n)) + a_1 \Delta t g(\tilde{\sigma}_n, s_n)}{b_2 a_1 + a_2 b_1} \right\}$$

where

$$a_1 \equiv 1 + 3\mu \Delta t \frac{\partial f}{\partial \tilde{\sigma}_n}$$

$$a_2 \equiv 3\mu \Delta t \frac{\partial f}{\partial s_n}$$

$$b_1 \equiv \Delta t \frac{\partial g}{\partial \tilde{\sigma}_n}$$

$$b_2 \equiv 1 - \Delta t \frac{\partial g}{\partial s_n}$$

However, with certain models, (specifically, the sin hyperbolic model for Aluminium at 673 K) the value of $\tilde{\sigma}_{n+1}$ calculated by the above equation becomes so high that further substitution in the functions f and g blows up. Hence in such cases the value of $\tilde{\sigma}_{n+1}$ is taken to be equal to 0 or $\tilde{\sigma}_n$. This does not affect the algorithm in a big way.

3. Now level 1 iterations are performed to obtain subsequent values for s_{n+1} . The iterations are performed till convergence is obtained on s_{n+1} . Consider a generic k^{th} step. Assume that the function f is invertible to give flow stress. ie., assume that there exists a function f^{-1} such that

$$\dot{\epsilon}^p = f(\tilde{\sigma}, s)$$

implies

$$\tilde{\sigma} = f^{-1}(\dot{\epsilon}^p, s)$$

- (a) Calculate an upper bound on $\tilde{\sigma}_{n+1}$, by evaluating the smaller of $f^{-1}\left(\frac{\tilde{\sigma}_*}{3\mu \Delta t}, s\right)$ and $\tilde{\sigma}_*$. Assign this value to $\tilde{\sigma}_u$.
- (b) Fix the lower bound for $\tilde{\sigma}_{n+1}$ as 0.
- (c) Do level 2 iterations to update $\tilde{\sigma}_{n+1}$ and to evaluate $(d\tilde{\sigma}_{n+1}/ds_{n+1})$ (described below).
- (d) Evaluate the error associated with s_{n+1}

$$E_s^k = s_{n+1}^k - s_n - \Delta t g_{n+1}^k$$

where

$$g_{n+1}^k = g(\tilde{\sigma}_{n+1}^k, s_{n+1}^k)$$

- (e) Compare this error with the tolerance TOL_s . Iteration is assumed to be converged if

$$|E_s^k| \leq \text{TOL}_s$$

Then $s_{n+1} = s_{n+1}^k$ and $\tilde{\sigma}_{n+1} = \tilde{\sigma}_{n+1}^k$. Otherwise the iteration is continued.

- (f) If the iteration continues, the correction Δs^k is calculated.

$$\Delta s^k = \frac{-E_s^k}{dE_s^k/ds_{n+1}^k}$$

where

$$\frac{dE_s^k}{ds_{n+1}^k} = 1 - \Delta t \left\{ \frac{\partial g_{n+1}^k}{\partial s_{n+1}^k} + \frac{\partial g_{n+1}^k}{\partial \tilde{\sigma}_{n+1}^k} \frac{d\tilde{\sigma}_{n+1}^k}{ds_{n+1}^k} \right\}$$

where $d\tilde{\sigma}_{n+1}^k/ds_{n+1}^k$ is calculated during the level 2 iterations.

- (g) The estimate for the next level iteration is

$$s_{n+1}^{k+1} = s_{n+1}^k + \Delta s^k$$

- (h) The initial estimate of $\tilde{\sigma}_{n+1}$ to start the next level 2 iterations is

$$\tilde{\sigma}_{n+1}^{k+1} = \tilde{\sigma}_{n+1}^k + \frac{d\tilde{\sigma}_{n+1}^k}{ds_{n+1}^k} \Delta s^k$$

4. Level 2 iterations are performed to obtain the updates on $\tilde{\sigma}_{n+1}^k$. Consider the generic step i within the k^{th} loop above.

(a) Evaluate the error associated with $\tilde{\sigma}_{n+1}$

$$E_{\tilde{\sigma}}^{k,i} = \tilde{\sigma}_{n+1}^k - \tilde{\sigma}_* + 3\mu \Delta t f_{n+1}^{k,i}$$

where

$$f_{n+1}^{k,i} = g\left(\tilde{\sigma}_{n+1}^{k,i}, s_{n+1}^k\right)$$

(b) Compare this error with the tolerance. If

$$\left|E_{\tilde{\sigma}}^{k,i}\right| \leq \text{TOL}_\sigma$$

then the scheme has converged. Assign $\tilde{\sigma}_{n+1}^k = \tilde{\sigma}_{n+1}^{k,i}$. Evaluate $d\tilde{\sigma}_{n+1}^k/ds_{n+1}^k$,

$$\frac{d\tilde{\sigma}_{n+1}^k}{ds_{n+1}^k} = \frac{-3\mu \Delta t \left(\partial f_{n+1}^{k,i}/\partial s_{n+1}^k\right)}{1 + 3\mu \Delta t \left(\partial f_{n+1}^{k,i}/\partial \tilde{\sigma}_{n+1}^k\right)}$$

for using in level 1 iterations.

(c) If not converged, Calculate the Newton-Raphson correction factor $\Delta\tilde{\sigma}_{NR}^{k,i}$

$$\Delta\tilde{\sigma}_{NR}^{k,i} = \frac{-E_{\tilde{\sigma}}^{k,i}}{dE_{\tilde{\sigma}}^{k,i}/d\tilde{\sigma}_{n+1}^k}$$

where

$$\frac{dE_{\tilde{\sigma}}^{k,i}}{d\tilde{\sigma}_{n+1}^k} = 1 + 3\mu \Delta t \frac{\partial f_{n+1}^{k,i}}{\partial \tilde{\sigma}_{n+1}^k}$$

(d) Depending upon the sign of $\Delta\tilde{\sigma}_{NR}^{k,i}$, update either the upper bound $\tilde{\sigma}_u$ or the lower bound $\tilde{\sigma}_l$ for $\tilde{\sigma}_{n+1}^k$ and then calculate the maximum allowable correction, as given by a quasi-bisection scheme. If $\Delta\tilde{\sigma}_{NR}^{k,i} < 0$, then

$$\tilde{\sigma}_u = \tilde{\sigma}_{n+1}^{k,i}$$

and

$$\Delta\tilde{\sigma}_{max}^{k,i} = \frac{1}{2} \left(\tilde{\sigma}_l - \tilde{\sigma}_{n+1}^{k,i}\right)$$

If $\Delta\tilde{\sigma}_{NR}^{k,i} > 0$, then

$$\tilde{\sigma}_l = \tilde{\sigma}_{n+1}^{k,i}$$

and

$$\Delta\tilde{\sigma}_{max}^{k,i} = \tilde{\sigma}_u - \tilde{\sigma}_{n+1}^{k,i}$$

(e) Determine the correction to be used. If $|\Delta\tilde{\sigma}_{NR}^{k,i}| > |\Delta\tilde{\sigma}_{max}^{k,i}|$, then

$$\Delta\tilde{\sigma}^{k,i} = \Delta\tilde{\sigma}_{max}^{k,i}$$

else

$$\Delta\tilde{\sigma}^{k,i} = \Delta\tilde{\sigma}_{NR}^{k,i}$$

(f) The estimate for the next iteration is

$$\tilde{\sigma}_{n+1}^{k,i+1} = \tilde{\sigma}_{n+1}^{k,i} + \Delta\tilde{\sigma}^{k,i}$$

This algorithm works very well for the sin hyperbolic constitutive model.

For a rate independent model, one of the equations will become

$$\tilde{\sigma}_{n+1} = s_{n+1}$$

Hence, we have only one nonlinear equation to solve. In that case,

$$E_{\tilde{\sigma}}^i = \tilde{\sigma}_{n+1}^i - \tilde{\sigma}_* + 3\mu \Delta t f_{n+1}^i$$

and

$$\Delta\tilde{\sigma}_{n+1} = \frac{-E_{\tilde{\sigma}}^i}{1 + 3\mu \Delta t \left(\frac{\partial f}{\partial \tilde{\sigma}} + \frac{\partial f}{\partial s} \right)}$$

This is continued till the convergence is achieved.

B Stiffness contribution from the boundary tractions and the body forces

Boundary tractions

Let $\hat{\mathbf{t}}$ be the prescribed traction on the boundary dependent on the current coordinates. We need to evaluate,

$$d \left(\int_{\partial \mathbf{B}_{n+1}} \hat{\mathbf{t}} \cdot \tilde{\mathbf{u}} \, ds \right) = d \left(\int_{\partial \mathbf{B}_{n+1}} (\hat{\mathbf{t}} \cdot \tilde{\mathbf{u}}) \mathbf{n} \cdot d\mathbf{s} \right)$$

where \mathbf{n} is the normal to the surface, ds is the infinitesimal area element and $d\mathbf{s}$ is the vectorial area element associated with the normal to the surface. The integral is evaluated over a boundary which is changing continuously. Following [8], we find that, if \mathbf{B} is a regular region, then

$$\frac{d}{dt} \int_{\partial \mathbf{B}} \mathbf{F} \cdot d\mathbf{s} = \int_{\partial \mathbf{B}} \mathbf{F} \cdot d\mathbf{s} + \int_{\partial \mathbf{B}} (\nabla \cdot \mathbf{F}) \mathbf{v} \cdot d\mathbf{s}$$

Here \mathbf{v} is the velocity as defined by

$$\mathbf{v} = \frac{d}{dt} \mathbf{x}_{n+1}$$

We need to take the differential of the boundary traction as a function of the differential of the incremental displacements. Hence if we assume that successive configurations are functions of a scalar variable λ and taking total derivative with respect to this λ and simplifying, we will arrive at the required relation. Substituting the traction forces and simplifying, we arrive at

$$d \left(\int_{\partial \mathbf{B}_{n+1}} \hat{\mathbf{t}} \cdot \tilde{\mathbf{u}} \, ds \right) = \int_{\partial \mathbf{B}_{n+1}} \hat{\mathbf{t}} \cdot \tilde{\mathbf{u}} \, ds + \int_{\partial \mathbf{B}_{n+1}} [(\nabla \hat{\mathbf{t}}) \mathbf{n} \cdot \tilde{\mathbf{u}} + (\hat{\mathbf{t}} \cdot \tilde{\mathbf{u}})(\nabla \cdot \mathbf{n})] (d\mathbf{u}_{n+1} \cdot \mathbf{n}) \, ds$$

The last term on the RHS of the above equation contributes to the stiffness matrix.

Body forces

Let $\hat{\mathbf{b}}$ be the prescribed body force as a function of the current coordinates. We need to evaluate,

$$d \left(\int_{\mathbf{B}_{n+1}} \hat{\mathbf{b}} \cdot \tilde{\mathbf{u}} \, dV \right)$$

Again, following [8], we have

$$\frac{d}{dt} \int_{\mathbf{B}} F \, dV = \int_{\mathbf{B}} \left[\nabla \cdot (F \mathbf{v}) + \frac{\partial F}{\partial t} \right] dV$$

Following the previous section, we arrive at

$$d \left(\int_{\mathbf{B}_{n+1}} \hat{\mathbf{b}} \cdot \tilde{\mathbf{u}} \, dV \right) = \int_{\mathbf{B}_{n+1}} (2(\nabla \hat{\mathbf{b}}) d\mathbf{u}_{n+1} \cdot \tilde{\mathbf{u}}) \, dV$$

This term also contributes to the stiffness matrix.

C A simplified form of the linearized elasto-plastic moduli

In section 4, we derived the value for linearized elasto plastic moduli \mathcal{C} as

$$\mathcal{C} = \mathcal{L}^e - 3\mu \Delta t \left[\frac{f}{\tilde{\sigma}_*} \mathcal{L}' + \alpha \mathcal{L}' [\bar{\mathbf{T}}_*'] \otimes \bar{\mathbf{T}}_*' \right]$$

The above equation becomes much simpler when the elastic material modulus is isotropic. Then

$$\mathcal{L}^e = 2\mu \mathcal{I} + \left(\kappa - \frac{2}{3}\mu \right) \mathbf{I} \otimes \mathbf{I}$$

Now by using the definition of \mathcal{L}' , we obtain

$$\mathcal{L}' = 2\mu \mathcal{I} - \frac{2}{3}\mu \mathbf{I} \otimes \mathbf{I}$$

Also

$$\mathcal{L}' [\bar{\mathbf{T}}'_*] = 2\mu\bar{\mathbf{T}}'_*$$

since $\bar{\mathbf{T}}'_*$ is traceless. With these simplifications, we arrive at

$$\mathbf{C} = 2\mu' \mathcal{I} + \left(\kappa - \frac{2}{3}\mu' \right) \mathbf{I} \otimes \mathbf{I} + \beta \bar{\mathbf{T}}'_* \otimes \bar{\mathbf{T}}'_*$$

where

$$\mu' = \eta_{n+1} \mu$$

and

$$\beta = \frac{3\mu}{\tilde{\sigma}_*^2} (c - \eta_{n+1})$$

The value for c is derived in section 4. In order to derive the above, note that

$$\tilde{\sigma}_{n+1} = \tilde{\sigma}_* - 3\mu \Delta t f$$

implies

$$\eta_{n+1} = \frac{\tilde{\sigma}_{n+1}}{\tilde{\sigma}_*} = 1 - \frac{3\mu \Delta t f}{\tilde{\sigma}_*}$$

D Equivalent matrices for arbitrary tensorial operations

When the Principle of Virtual Work equation is linearized by using Newton-Raphson method, we end up with a linear system of algebraic equations, with the variable being incremental variations on the relative deformation gradient. However, the form of the equation is still not amenable to solving by means of a well known linear system of equation solver. The equation which we obtain, though linear, involves a lot of tensorial operations with known tensors. In order to automatically resolve this equation into a simplified problem, the following procedure is employed.

Suppose the equation we have to solve is of the form,

$$\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{B} + \text{tr}(\mathbf{C}\mathbf{X}\mathbf{D})\mathbf{E} + \text{sym}(\mathbf{X}\mathbf{F}) + \text{skew}(\mathbf{G}\mathbf{X}\mathbf{H}) = \mathbf{O}$$

where \mathbf{X} is the unknown tensor and all the other tensors are known. \mathbf{O} is the zero tensor. All the second order tensors can be represented as 3×3 matrices in an orthogonal reference frame. Because of the specific nature of the problem being considered *viz.* the plane strain and axi-symmetric assumptions, for any given tensor \mathbf{A} , the components

$$\mathbf{A}_{31} = \mathbf{A}_{32} = \mathbf{A}_{23} = \mathbf{A}_{13} = 0$$

A relation

$$\mathbf{A}\mathbf{X} = \mathbf{B}$$

can be re-written as

$$\mathbf{A}_l \mathbf{X}_c = \mathbf{B}_c$$

where any matrix \mathbf{X} can be expressed as \mathbf{X}_c as,

$$\mathbf{X}_c = \begin{bmatrix} \mathbf{X}_{11} \\ \mathbf{X}_{12} \\ \mathbf{X}_{21} \\ \mathbf{X}_{22} \\ \mathbf{X}_{33} \end{bmatrix}$$

Now simple computation shows that \mathbf{A}_l is a 5×5 matrix whose components are given by

$$\mathbf{A}_l = \begin{bmatrix} \mathbf{A}_{11} & 0 & \mathbf{A}_{12} & 0 & 0 \\ 0 & \mathbf{A}_{11} & 0 & \mathbf{A}_{12} & 0 \\ \mathbf{A}_{21} & 0 & \mathbf{A}_{22} & 0 & 0 \\ 0 & \mathbf{A}_{21} & 0 & \mathbf{A}_{22} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{A}_{33} \end{bmatrix}$$

Here the subscript l is to denote the original tensor operation is a multiplication from the left. Following similar logic, the tensor multiplication from the right side,

$$\mathbf{X} \mathbf{A} = \mathbf{B}$$

is expressed as

$$\mathbf{A}_r \mathbf{X}_c = \mathbf{B}_c$$

where

$$\mathbf{A}_r = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{21} & 0 & 0 & 0 \\ \mathbf{A}_{12} & \mathbf{A}_{22} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{A}_{11} & \mathbf{A}_{21} & 0 \\ 0 & 0 & \mathbf{A}_{12} & \mathbf{A}_{22} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{A}_{33} \end{bmatrix}$$

The relation

$$\mathbf{A} \mathbf{X}^T = \mathbf{B}$$

is expressed as

$$\mathbf{A}_{lt} \mathbf{X}_c = \mathbf{B}_c$$

where

$$\mathbf{A}_{lt} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{A}_{11} & \mathbf{A}_{12} & 0 \\ \mathbf{A}_{21} & \mathbf{A}_{22} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{A}_{21} & \mathbf{A}_{22} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{A}_{33} \end{bmatrix}$$

Three other important operations that repeat often are “symmetry”, “skew” and “trace”. The relation

$$\text{sym}(\mathbf{A}\mathbf{X}\mathbf{B}) = \mathbf{Y}$$

is initially reduced to

$$\text{sym}(\mathbf{P}\mathbf{X}_c) = \mathbf{Y}_c$$

which is further simplified as

$$\mathbf{P}_s\mathbf{X}_c = \mathbf{Y}_c$$

Here,

$$\mathbf{P} = \mathbf{A}_l\mathbf{B}_r = \mathbf{B}_r\mathbf{A}_l$$

Though, in general matrix multiplications are *NOT* commutative, in this case (for all these equivalence relations) the matrices obtained commute. This can be proved by long hand multiplication. Alternately, looking at

$$\mathbf{A}\mathbf{X}\mathbf{B} = \mathbf{Y}$$

Defining, $\mathbf{X}\mathbf{B} = \mathbf{Z}$, this results in,

$$\mathbf{A}\mathbf{Z} = \mathbf{Y}$$

Hence,

$$\mathbf{A}_l\mathbf{Z}_c = \mathbf{Y}_c$$

But

$$\mathbf{Y}_c = \mathbf{B}_r\mathbf{X}_c$$

and therefore,

$$\mathbf{A}_l\mathbf{B}_r\mathbf{X}_c = \mathbf{Y}_c$$

On the other hand, with $\mathbf{A}\mathbf{X} = \mathbf{W}$,

$$\mathbf{W}\mathbf{B} = \mathbf{Y}$$

which implies,

$$\mathbf{B}_r\mathbf{W}_c = \mathbf{Y}_c$$

and hence,

$$\mathbf{B}_r\mathbf{A}_l\mathbf{X}_c = \mathbf{Y}_c$$

Hence \mathbf{A}_l and \mathbf{B}_r commute.

Now coming back to \mathbf{P}_s , suppose

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} & \mathbf{P}_{13} & \mathbf{P}_{14} & \mathbf{P}_{15} \\ \mathbf{P}_{21} & \mathbf{P}_{22} & \mathbf{P}_{23} & \mathbf{P}_{24} & \mathbf{P}_{25} \\ \mathbf{P}_{31} & \mathbf{P}_{32} & \mathbf{P}_{33} & \mathbf{P}_{34} & \mathbf{P}_{35} \\ \mathbf{P}_{41} & \mathbf{P}_{42} & \mathbf{P}_{43} & \mathbf{P}_{44} & \mathbf{P}_{45} \\ \mathbf{P}_{51} & \mathbf{P}_{52} & \mathbf{P}_{53} & \mathbf{P}_{54} & \mathbf{P}_{55} \end{bmatrix}$$

then

$$\mathbf{P}_s = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} & \mathbf{P}_{13} & \mathbf{P}_{14} & \mathbf{P}_{15} \\ (\mathbf{P}_{21} + \mathbf{P}_{31})/2 & (\mathbf{P}_{22} + \mathbf{P}_{32})/2 & (\mathbf{P}_{23} + \mathbf{P}_{33})/2 & (\mathbf{P}_{24} + \mathbf{P}_{34})/2 & (\mathbf{P}_{25} + \mathbf{P}_{35})/2 \\ (\mathbf{P}_{21} + \mathbf{P}_{31})/2 & (\mathbf{P}_{22} + \mathbf{P}_{32})/2 & (\mathbf{P}_{23} + \mathbf{P}_{33})/2 & (\mathbf{P}_{24} + \mathbf{P}_{34})/2 & (\mathbf{P}_{25} + \mathbf{P}_{35})/2 \\ \mathbf{P}_{41} & \mathbf{P}_{42} & \mathbf{P}_{43} & \mathbf{P}_{44} & \mathbf{P}_{45} \\ \mathbf{P}_{51} & \mathbf{P}_{52} & \mathbf{P}_{53} & \mathbf{P}_{54} & \mathbf{P}_{55} \end{bmatrix}$$

The skew portion \mathbf{P}_{sk} of \mathbf{P} is obtained simply as

$$\mathbf{P}_{sk} = \mathbf{P} - \mathbf{P}_s$$

where, \mathbf{P}_s is the symmetric part as explained previously. Now consider,

$$\text{tr}(\mathbf{A}\mathbf{X}\mathbf{B})\mathbf{D} = \mathbf{Y}$$

This has to be reduced to the form

$$\mathbf{P}_t \mathbf{X}_c = \mathbf{Y}_c$$

First $\mathbf{A}\mathbf{X}\mathbf{B}$ is reduced to the form $\mathbf{P}\mathbf{X}_c$ and then \mathbf{P}_t is obtained as below. First construct \mathbf{Q} as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \\ \mathbf{Q}_4 \\ \mathbf{Q}_5 \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{11} + \mathbf{P}_{41} + \mathbf{P}_{51} \\ \mathbf{P}_{12} + \mathbf{P}_{42} + \mathbf{P}_{52} \\ \mathbf{P}_{13} + \mathbf{P}_{43} + \mathbf{P}_{53} \\ \mathbf{P}_{14} + \mathbf{P}_{44} + \mathbf{P}_{54} \\ \mathbf{P}_{15} + \mathbf{P}_{45} + \mathbf{P}_{55} \end{bmatrix}$$

Then, we have

$$\mathbf{P}_t = \begin{bmatrix} \mathbf{Q}_1 \mathbf{D}_{11} & \mathbf{Q}_2 \mathbf{D}_{11} & \mathbf{Q}_3 \mathbf{D}_{11} & \mathbf{Q}_4 \mathbf{D}_{11} & \mathbf{Q}_5 \mathbf{D}_{11} \\ \mathbf{Q}_1 \mathbf{D}_{12} & \mathbf{Q}_2 \mathbf{D}_{12} & \mathbf{Q}_3 \mathbf{D}_{12} & \mathbf{Q}_4 \mathbf{D}_{12} & \mathbf{Q}_5 \mathbf{D}_{12} \\ \mathbf{Q}_1 \mathbf{D}_{21} & \mathbf{Q}_2 \mathbf{D}_{21} & \mathbf{Q}_3 \mathbf{D}_{21} & \mathbf{Q}_4 \mathbf{D}_{21} & \mathbf{Q}_5 \mathbf{D}_{21} \\ \mathbf{Q}_1 \mathbf{D}_{22} & \mathbf{Q}_2 \mathbf{D}_{22} & \mathbf{Q}_3 \mathbf{D}_{22} & \mathbf{Q}_4 \mathbf{D}_{22} & \mathbf{Q}_5 \mathbf{D}_{22} \\ \mathbf{Q}_1 \mathbf{D}_{33} & \mathbf{Q}_2 \mathbf{D}_{33} & \mathbf{Q}_3 \mathbf{D}_{33} & \mathbf{Q}_4 \mathbf{D}_{33} & \mathbf{Q}_5 \mathbf{D}_{33} \end{bmatrix}$$

Once we have derived these transformations, getting the global Jacobian becomes a very simple task.