Technical University of Denmark

Non-linear finite element modeling

Mikkelsen, Lars Pilgaard

Publication date:
2007

Document Version
Publisher's PDF, also known as Version of record

Risø-R-Report

RISØ

Non-linear finite element modeling

Lars Pilgaard Mikkelsen

Risø-R-1625(EN)

Risø National Laboratory
Technical University of Denmark
Roskilde, Denmark
November 2007

DTU

Author: Lars Pilgaard Mikkelsen
Title: Non-linear finite element modeling
Department: Material Research Department

Risø-R-1625(EN)
November 2007

Abstract (max. 2000 char.):

The note is written for courses in "Non-linear finite element method". The note has been used by the author teaching non-linear finite element modeling at Civil Engineering at Aalborg University, Computational Mechanics at Aalborg University Esbjerg, Structural Engineering at the University of the Southern Denmark and in Medicine and Technology at the Technical University of Denmark. The note focus on the applicability to actually code routines with the purpose to analyze a geometrically or material non-linear problem. The note is tried to be kept on so brief a form as possible, with the main focus on the governing equations and methods of implementing.

ISSN 0106-2840
ISBN 978-87-550-3644-4

Contract no.:

Group's own reg. no.:
1615064-06

Sponsorship:

Cover :

Pages:
Tables:
References:

Information Service Department
Risø National Laboratory
Technical University of Denmark
P.O.Box 49
DK-4000 Roskilde
Denmark
Telephone +45 46774004
bibl@risoe.dk
Fax +45 46774013
www.risoe.dk

Non-linear finite element modeling

Lars Pilgaard Mikkelsen
Material Research Department, Risø National Laboratory
Technical University of Denmark, 4000 Roskilde

1. INTRODUCTION

The note is written for courses in "Non-linear finite element method". The note has been used by the author at Civil Engineering at Aalborg University, Computational Mechanics at Aalborg University Esbjerg, Structural Engineering at the University of the Southern Denmark and in Medicine and Technology at the Technical University of Denmark. The note focus on the applicability to actually code routines with the purpose to analyze a geometrically or material non-linear problem. The note is tried to be kept on so brief a form as possible, with the main focus on the governing equations and methods of implementing. For a more complete exposition please see the references in the end of the notes. Specially, [1] and [2] is suited as complementary textbooks.

Though out the whole document, a conventional tensor notation has been used where repeated indices indicate summations and where Latin indices range from 1 to 3, while Greek indices range from 1 to 2. The notation $(\ )_{,i}$ indicate partial differentiation with respect to the coordinate x_i . The finite element notation is highly inspired by [3, 4] but with reference to a usual matrix notation a'la [5]. Note, that "indices" in parentheses shall not be treated as indices in a tensor notation sense but indicate a degree of freedom, element number etc.

First is included a brief repetition for a linear finite element model. This is done in order to refer back to a common language in the subsequently derivations.

Enjoy ...

2. LINEAR CONTINUUM

Equilibrium for a linear elastic three-dimensional continuum where the assumption of small strains, $|\varepsilon_{ij}| \ll 1$, and small displacements gradients, $|u_{i,j}| \ll 1$, is satisfied, can e.g. be expressed by the principal of virtual work

$$\int_V \sigma_{ij} \delta\varepsilon_{ij} dV = \int_S T_i \delta u_i dS \tag{2.1}$$

In (2.1), the volume forces is neglected while T_i denotes the surface traction. The tensor components of the strain and stress tensor is given by

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right) \tag{2.2}$$

$$\sigma_{ij} = \mathcal{L}_{ijkl}\varepsilon_{kl} \tag{2.3}$$

respectively, where $\mathcal{L}_{ijkl}$ represents the elastic modulus. For a linear elastic material following the generalized Hooks low, the elastic modulus is given by

$$\mathcal{L}_{ijkl} = \frac{E}{1+\nu}\left\{\frac{1}{2}\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) + \frac{\nu}{1-2\nu}\delta_{ij}\delta_{kl}\right\} \tag{2.4}$$

with Young's modulus E and Poisson's ratio ν .

Even in the linear case, it is rare that an exact solution to (2.1-2.3) exists and approximative methods is necessary. The "Finite Element Method" is such a method. In the finite element method the volume V is divided into K sub-volumes (elements), such that

$$V = \sum_{k=1}^{K} V_{(k)} \ , \quad S = \sum_{k=1}^{K} S_{(k)} \tag{2.5}$$

If element no. k does not contain part of the surface S, we have $S_{(k)} = 0$. Inside the volume $V_{(k)}$ the displacements components u_i are approximated by F_e shape functions $U^{(n)}$ such that the displacements in a given point inside the element is given by

$$u_i = \sum_{n=1}^{F_e} U_i^{(n)} D_{(n)} \tag{2.6}$$

where F_e is the number of degrees of freedom for the element, $U_i^{(n)}$ is the value of the shape function in a particular point and $D_{(n)}$ is the corresponding node point displacements. The shape functions is assumed to be specified such that the displacements, u_i are continues over the element boundaries when the common node points have the same displacements values, $D_{(n)}$. For a element with G_{No} nodes where the three displacement components u_i is approximated by the same shape function, $\phi_{(g)}, g = 1, \ldots, G_{(No)}$, inside an element and with only one degree of freedom per node per displacement component, $U_i^{(n)}$ can be written as

$$\begin{aligned}
U_i^{(1)} &= (\phi_{(1)}, 0, 0) \\
U_i^{(2)} &= (0, \phi_{(1)}, 0) \\
U_i^{(3)} &= (0, 0, \phi_{(1)}) \\
&\vdots \\
U_i^{(F_e-2)} &= (\phi_{(G_{No})}, 0, 0) \\
U_i^{(F_e-1)} &= (0, \phi_{(G_{No})}, 0) \\
U_i^{(F_e)} &= (0, 0, \phi_{(G_{No})})
\end{aligned} \tag{2.7}$$

After substitution of (2.6) in (2.2) the following approximation of the strains

$$\varepsilon_{ij} = \sum_{n=1}^{F_e} E_{ij}^{(n)} D_{(n)} \tag{2.8}$$

with

$$E_{ij}^{(n)} = \frac{1}{2}\left(U_{i,j}^{(n)} + U_{j,i}^{(n)}\right) \tag{2.9}$$

is obtained.

If the internal virtual work is denoted $\delta W_{int.}$ and the external virtual work is denoted $\delta W_{ext.}$ the principle of virtual work (2.1) can be written as

$$\delta W_{int.} = \delta W_{ext.} \tag{2.10}$$

which after substitution of (2.5) in (2.10) can be written as

$$\sum_{k=1}^{K} (\delta W_{int.})_{(k)} = \sum_{k=1}^{K} (\delta W_{ext.})_{(k)} \tag{2.11}$$

Approximating both the real and the virtual displacement by the finite element approximation (2.6, 2.8) such that

$$\begin{aligned} u_i = \textstyle\sum_{n=1}^{F_e} U_i^{(n)} D_{(n)} \quad \delta u_i = \textstyle\sum_{m=1}^{F_e} U_i^{(m)} \delta D_{(m)} \\ \varepsilon_{ij} = \textstyle\sum_{n=1}^{F_e} E_{ij}^{(n)} D_{(n)} \quad \delta \varepsilon_{ij} = \textstyle\sum_{m=1}^{F_e} E_{ij}^{(m)} \delta D_{(m)} \end{aligned} \tag{2.12}$$

the contribution from element no. (k) to the internal and external virtual work can be written as

$$(\delta W_{int.})_{(k)} = \sum_{n=1}^{F_e} \sum_{m=1}^{F_e} S_{(mn)}^{(k)} D_{(n)} \delta D_{(m)} \tag{2.13}$$

$$(\delta W_{ext.})_{(k)} = \sum_{m=1}^{F_e} P_{(m)}^{(k)} \delta D_{(m)} \tag{2.14}$$

where the element stiffness matrix can be written as

$$S_{(mn)}^{(k)} = \int_{V_{(k)}} \mathcal{L}_{ijkl} E_{kl}^{(n)} E_{ij}^{(m)} dV \tag{2.15}$$

while the element load vector can be written as

$$P_{(m)}^{(k)} = \int_{S_{(k)}} T_i U_i^{(m)} dS \tag{2.16}$$

Often, the shape functions (2.7) is given in local element coordinates ξ_i running between e.g. $\xi_i \in [-1 : 1]$. Therefore, during the differentiation (2.9) and the integration (2.15) and (2.16) a mapping between the local and the local coordinate system should be performed, e.g. chapter 6 in [5]. For Isoparametric elements where both the displacement of a point in the element and the global coordinates of a point in the element are interpolated from the nodal values, respectively using the same shape functions, the mapping between the two coordinate system is rather straightforward. The governing equation can be found in e.g. chapter 6 in [5]. The integration (2.15) and (2.16) is often based on a numerical integration using the Gaussian integration scheme.

The element stiffness matrix $S_{(mn)}^{(k)}$ and the element load vectors $P_{(m)}^{(k)}$ for each element $(k = 1, \ \ldots, \ K)$ can now be collected into one common equations system in agreement with (2.11). This can be done when the correlation between the local degree of freedoms ($m, n = 1, \ldots, F_e$) for each element and the global degrees of freedoms ($M, N = 1, \ldots, F$) for the mesh is known. The total number of freedoms is denoted by F. The resulting equations system

$$\sum_{M=1}^{F} \sum_{N=1}^{F} S_{(MN)} D_{(N)} \delta D_{(M)} = \sum_{M=1}^{F} P_{(M)} \delta D_{(M)} \tag{2.17}$$

can now be written as F independent equations by approximating of the virtual displacement field by one nodal displacement $\delta D_{(M)} = 1$, with all other nodal displacements equal to zero. Doing this successive for the F degree of freedoms we get F linear independent equations

$$\sum_{N=1}^{F} S_{(MN)} D_{(N)} = P_{(M)} \ , \quad M = 1, 2, \ldots, F \tag{2.18}$$

After including the kinematic boundary condition directly into (2.18), the solution of (2.18) will give the nodal displacements $D_{(N)}$ corresponding to the load vector $P_{(M)}$. From the nodal displacement $D_{(N)}$ it is successively possible to calculate the displacements, the strains and the stresses in the the structure using (2.6), (2.8) and (2.3), respectively.

2.1. Matrix notation

The above shown governing equations for the finite element model is possible to incorporate directly into a programming code such as `Fortran 77`. Nevertheless, in higher level programming language such as `Fortran 90`, `Matlab` and `C++`, it can be advantage to formulate the finite element equations in matrix notation. By arranging the shape functions $U_i^{(n)}$ and the thereof derived $E_{ij}^{(n)}$ in the matrixes

$$\mathbf{N}_{(3\times F_e)} = \begin{pmatrix} U_1^{(1)} & \cdots & U_1^{(F_e)} \\ U_2^{(1)} & \cdots & U_2^{(F_e)} \\ U_3^{(1)} & \cdots & U_3^{(F_e)} \end{pmatrix} \ , \quad \mathbf{B}_{(6\times F_e)} = \begin{pmatrix} E_{11}^{(1)} & \cdots & E_{11}^{(F_e)} \\ E_{22}^{(1)} & \cdots & E_{22}^{(F_e)} \\ E_{33}^{(1)} & \cdots & E_{33}^{(F_e)} \\ 2E_{12}^{(1)} & \cdots & 2E_{12}^{(F_e)} \\ 2E_{13}^{(1)} & \cdots & 2E_{13}^{(F_e)} \\ 2E_{23}^{(1)} & \cdots & 2E_{23}^{(F_e)} \end{pmatrix} \tag{2.19}$$

it follows that the displacements and strains vectors in a given point in element (k), see (2.6) and (2.8) can be written as

$$\mathbf{u} = \mathbf{N}\mathbf{D}^{(k)} \ , \quad \boldsymbol{\varepsilon} = \mathbf{B}\mathbf{D}^{(k)} \tag{2.20}$$

where the following vectors are defined ($\boldsymbol{\sigma}$ will be used later)

$$\mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \ , \quad \mathbf{D}^{(k)} = \begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ D_{F_e} \end{pmatrix} \ , \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{23} \end{pmatrix} \ , \quad \boldsymbol{\sigma} = \begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{pmatrix} \tag{2.21}$$

Using these matrixes, it is possible to write the element stiffness matrix (2.15) and the element load vector (2.16) as

$$\mathbf{S}^{(k)} = \int_{V_{(k)}} \mathbf{B}^T \boldsymbol{\mathcal{L}} \mathbf{B} dV \ , \quad \mathbf{P}^{(k)} = \int_{S_{(k)}} \mathbf{N}^T \mathbf{T} dS \tag{2.22}$$

with

$$\boldsymbol{\mathcal{L}} = \begin{pmatrix} \mathcal{L}_{1111} & \mathcal{L}_{1122} & \cdots & \mathcal{L}_{1113} \\ \mathcal{L}_{2211} & \mathcal{L}_{2222} & \cdots & \mathcal{L}_{2213} \\ \vdots & \vdots & \ddots & \\ \mathcal{L}_{1311} & \mathcal{L}_{1322} & & \mathcal{L}_{1313} \end{pmatrix} \ , \quad \mathbf{T} = \begin{pmatrix} T_1 \\ T_2 \\ T_3 \end{pmatrix} \tag{2.23}$$

In the matrix for $\boldsymbol{\mathcal{L}}$ the symmetry of $\mathcal{L}_{ijkl} = \mathcal{L}_{jikl}$ and $\mathcal{L}_{ijkl} = \mathcal{L}_{ijlk}$ is implicit included, while the symmetry $\mathcal{L}_{ijkl} = \mathcal{L}_{klij}$ will result in a symmetric $\boldsymbol{\mathcal{L}}$ matrix. From the constitutive matrix $\boldsymbol{\mathcal{L}}$, the stresses in a element can be found as

$$\boldsymbol{\sigma} = \boldsymbol{\mathcal{L}} \boldsymbol{\varepsilon} = \boldsymbol{\mathcal{L}} \mathbf{B} \mathbf{D}^{(k)} \tag{2.24}$$

Collecting the element stiffness matrix and the element load vector into the global stiffness matrix and load vector the following equation system is obtained

$$\mathbf{S}\mathbf{D} = \mathbf{P} \tag{2.25}$$

with

$$\mathbf{D} = (D_1, \ D_2, \ \cdots, D_F)^T \tag{2.26}$$

which can be solved similar to (2.18).

2.2. Numerical implementation

A linear finite element program can be considered as consist of three parts, a pre-processing part, an analyzing part and a post-processing part. In the following, the variable used is just examples which meaning will not been explained here. An example, building up a linear finite element model in `Matlab`/`Calfem` [6] can be found in [7]. The list shown below is thought as a check-list when building up a finite element program.

- Pre-processing
  - Definition of variable used in the model
    - Geometrical parameters: lgeo, hgeo, bgeo, ...
    - Topology parameters: Ndof, Nel, Nnode
    - Properties Emodul, SigmaY, Nexp, ...
    - Defining the case: Udisp, ...
  - Topology of the model
    - Define the node coordinates
    - Correlate the global node numbers to local node number in the element
    - Correlate local degree of freedom numbers with the global degree of freedom numbers

        * Identify nodes (and elements) along external boundaries
        * Correlate material and element properties to element numbers
    - Plot the undeformed structure

- Analyzing part
    - Initialize global stiffness matrix and right hand side: $\mathbf{S} = \mathbf{0}$, $\mathbf{P} = \mathbf{0}$
    - For each element:
        * Build up the local element stiffness matrix: $\mathbf{S}^{(k)}$
        * Build up eventually the local element load vector (static boundary condition): $\mathbf{P}^{(k)}$
        * Assemble the local element matrices $\mathbf{S}^{(k)}$, $\mathbf{P}^{(k)}$ into the global matrices $\mathbf{S}$, $\mathbf{P}$
    - Kinematic boundary condition
    - Solve the system: $\mathbf{SD} = \mathbf{P}$
    - Extract the displacement, strains, stresses, etc. from the solution in the integration points for each element: $\mathbf{u} = \mathbf{ND}^{(k)}$, $\boldsymbol{\varepsilon} = \mathbf{BD}^{(k)}$, $\boldsymbol{\sigma} = \boldsymbol{\mathcal{L}}\mathbf{BD}^{(k)}$

- Post-processing part
    - Plot the deformed structure
    - Plot contours of strains, stresses, etc.

3. KINEMATIC NON-LINEAR PROBLEM

Even inside the approximation of a linear elastic material behaviour, a number of problems can not be solved satisfactory by a linear finite element model. Examples is post-buckling analysis, finite strain problems, specimen exposed for large rotations, etc.

Here will, as an example on a kinematic (geometric) nonlinear problem, be treated the case of a three dimensional continuum where the assumption for small strains, $|\varepsilon_{ij}| \ll 1$, holds while the displacement gradients, $u_{i,j}$, no longer can be considered as small. For this case, the governing equations can be written as

$$\int_V \sigma_{ij}\delta\varepsilon_{ij}dV = \int_S T_i\delta u_i dS \tag{3.1}$$

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i} + u_{k,i}u_{k,j}\right) \tag{3.2}$$

$$\sigma_{ij} = \mathcal{L}_{ijkl}\varepsilon_{kl} \tag{3.3}$$

If we make a finite element approach of the total displacements similar to the one done in chapter 2 with (2.6)

$$u_i = \sum_{n=1}^{F_e} U_i^{(n)} D_{(n)} \tag{3.4}$$

we will no longer get a system of linear equations as (2.25), but a set of non-linear equations

$$\mathbf{G}(\mathbf{D}) = \mathbf{P} \tag{3.5}$$

Another way to write the governing equations is in term of a small increment which in the following will be denoted by a dot $(\dot{})$. In order to do this we will write the displacements, the stresses and the prescribed surface forces after a small increment as

$$\bar{u}_i = u_i + \dot{u}_i \tag{3.6}$$

$$\bar{\sigma}_{ij} = \sigma_{ij} + \dot{\sigma}_{ij} \tag{3.7}$$

$$\bar{T}_i = T_i + \dot{T}_i \tag{3.8}$$

respectively. After substitute (3.6) in (3.2) the total strain components after a small increment is found to

$$\bar{\varepsilon}_{ij} = \varepsilon_{ij} + \dot{\varepsilon}_{ij} + \frac{1}{2}\dot{u}_{k,i}\dot{u}_{k,j} \tag{3.9}$$

with

$$\dot{\varepsilon}_{ij} = \frac{1}{2}\left(\dot{u}_{i,j} + \dot{u}_{j,i} + \dot{u}_{k,i}u_{k,j} + u_{k,i}\dot{u}_{k,j}\right) \tag{3.10}$$

The equilibrium after this small increment can similar to (3.1) with the new total quantities be written as

$$\int_V \bar{\sigma}_{ij}\delta\bar{\varepsilon}_{ij}dV = \int_S \bar{T}_i\delta\bar{u}_i dS \tag{3.11}$$

After substitution of (3.6-3.9) into (3.11) and rearrangement the terms we can write

$$\int_V \left\{\dot{\sigma}_{ij}\delta\dot{\varepsilon}_{ij} + \sigma_{ij}\dot{u}_{k,j}\delta\dot{u}_{k,i}\right\} dV = \int_S \dot{T}_i\delta\dot{u}_i dS - \left[\int_V \sigma_{ij}\delta\dot{\varepsilon}_{ij}dV - \int_S T_i\delta\dot{u}_i dS\right] \tag{3.12}$$

where it has been make use of that $(\dot{})(\dot{}) \ll (\dot{})$ (small increment), and that $\delta(\varepsilon_{ij} + \dot{\varepsilon}_{ij}) = \delta\dot{\varepsilon}_{ij}$ and corresponding $\delta(u_i + \dot{u}_i) = \delta(\dot{u}_i)$ which comes from the fact that the displacement/strain state is known before the increment and the variation thereof therefore vanishes. The bracket term vanishes if equilibrium is satisfied before the increment, but will normally be included in a numerical procedure in order to avoid the numerical solution to drift away from equilibrium. The virtual work on incremental form will together with increment of the strains (3.10) and the constitutive relation (2.3) on incremental form

$$\dot{\sigma}_{ij} = \mathcal{L}_{ijkl}\dot{\varepsilon}_{ij} \tag{3.13}$$

be the governing equations for the finite element formulation.

In the incremental finite element model, it is not the total displacements u_i , but the increment thereof $\dot{u}_i$ which is approximated by the shape functions $U_i^{(n)}$. Therefore, $\dot{u}_i$ and $\dot{\varepsilon}_{ij}$ can be written as (2.6)

$$\dot{u}_i = \sum_{n=1}^{F_e} U_i^{(n)}\dot{D}_{(n)} \tag{3.14}$$

and

$$\dot{\varepsilon}_{ij} = \sum_{n=1}^{F_e} E_{ij}^{(n)}\dot{D}_{(n)} \tag{3.15}$$

with

$$E_{ij}^{(n)} = \frac{1}{2}\left(U_{i,j}^{(n)} + U_{j,i}^{(n)} + U_{k,i}^{(n)}u_{k,j} + u_{k,i}U_{k,j}^{(n)}\right) \tag{3.16}$$

where $E_i^{(n)}$ depends on the total deformation state, u_i. Following the procedure from section 2, the element stiffness matrix, the element load vector and the element load correction vector can be found as

$$S_{(mn)}^{(k)} = \int_{V_{(k)}} \left\{\mathcal{L}_{ijkl}E_{kl}^{(n)}E_{ij}^{(m)} + \sigma_{ij}U_{k,j}^{(n)}U_{k,i}^{(m)}\right\} dV \tag{3.17}$$

$$\dot{P}_{(m)}^{(k)} = \int_{S_{(k)}} \dot{T}_i U_i^{(m)} dS \tag{3.18}$$

$$R_{(m)}^{(k)} = \int_{V_{(k)}} \sigma_{ij}E_{ij}^{(m)} dV - \int_{S_{(k)}} T_i U_i^{(m)} dS \tag{3.19}$$

respectively, which components generally differing from increment to increment. After collection the element stiffness matrix and the load vectors in the system stiffness matrix and system load vectors, and approximating the virtual displacement fields similar to what occurs between (2.17) and (2.18), the following system of linear independent equations

$$\sum_{N=1}^{F} S_{(MN)}\dot{D}_{(N)} = \dot{P}_{(M)} - R_M \;, \quad M = 1, 2, \ldots, F \tag{3.20}$$

is obtained.

The solution of (3.20) gives the increment of the node displacement, $\dot{D}_{(N)}$. Having these, the strain increment, $\dot{\varepsilon}_{ij}$ and therefrom the stress increments $\dot{\sigma}_{ij}$ are calculated in the integrations points, element by element, from equations (3.15) and (3.13), respectively. The current values of the stresses after the increment are updated by calculating $\sigma_{ij} + \dot{\sigma}_{ij}$ in each integration point, and the current values of the nodal displacement are updated by calculating $D_{(N)} + \dot{D}_{(N)}$. Subsequently, a new increment can be calculated repeating the procedure shown above. The total displacement u_i in the integration point is calculated from (3.4), and if the total strains are wanted at some point in the loading history, equation (3.2) can be used for this with

$$\dot{u}_{i,j} = \sum_{n=1}^{F_e} U_{i,j}^{(n)} \dot{D}_{(n)} \tag{3.21}$$

3.1. Matrix notation

In the linear case in section 2, it was straight forward to rewrite the governing equations into matrix notation. For the non-linear cases case, it can be more difficult but of course possible. It is the second term in (3.17) which require some creative matrix manipulations. In addition to defining the incremental displacement components vector and node displacements vector as

$$\dot{\mathbf{u}} = \begin{pmatrix} \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \end{pmatrix} \;, \quad \dot{\mathbf{D}}^{(k)} = \begin{pmatrix} \dot{D}_1 \\ \dot{D}_2 \\ \vdots \\ \dot{D}_{F_e} \end{pmatrix} \tag{3.22}$$

and the strain and stress vector as

$$\boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{23} \end{pmatrix} \quad , \quad \boldsymbol{\sigma} = \begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{pmatrix} \tag{3.23}$$

it is necessary to define an incremental displacement gradient vector with 9 elements

$$\dot{\hat{\mathbf{u}}} = \begin{pmatrix} \dot{u}_{1,1} & \dot{u}_{1,2} & \dot{u}_{1,3} & \dot{u}_{2,1} & \dot{u}_{2,2} & \dot{u}_{2,3} & \dot{u}_{3,1} & \dot{u}_{3,2} & \dot{u}_{3,3} \end{pmatrix}^T \tag{3.24}$$

Doing this, the increment of the displacement vector, the increment of the strain vector and the increment of displacement gradient vector can be written as

$$\dot{\mathbf{u}} = \mathbf{N}\dot{\mathbf{D}}^{(k)} \ , \quad \dot{\boldsymbol{\varepsilon}} = \mathbf{B}\dot{\mathbf{D}}^{(k)} \ , \quad \dot{\hat{\mathbf{u}}} = \hat{\mathbf{B}}\dot{\mathbf{D}}^{(k)} \tag{3.25}$$

where $\mathbf{N}$ and $\mathbf{B}$ is given like (2.19) but with $E_{ij}^{(n)}$ from (2.9) replaced by the components from (3.16) whereby $\mathbf{N}$ will depends on the total instantaneous displacements u_i. The matrix $\hat{\mathbf{B}}$ is given by

$$\hat{\mathbf{B}}_{(9\times F_e)} = \begin{pmatrix} U_{1,1}^{(1)} & \cdots & U_{1,1}^{(F_e)} \\ U_{2,1}^{(1)} & \cdots & U_{2,1}^{(F_e)} \\ U_{3,1}^{(1)} & \cdots & U_{3,1}^{(F_e)} \\ U_{1,2}^{(1)} & \cdots & U_{1,2}^{(F_e)} \\ \vdots & \vdots & \vdots \\ U_{3,3}^{(1)} & \cdots & U_{3,3}^{(F_e)} \end{pmatrix} \tag{3.26}$$

which with $\phi_{(g)}$ introduced in (2.7) can be written as

$$\hat{\mathbf{B}}_{(9\times F_e)} = \begin{pmatrix} \phi_{(1),1}\mathbf{I} & \cdots & \phi_{(G_{No}),1}\mathbf{I} \\ \phi_{(1),2}\mathbf{I} & \cdots & \phi_{(G_{No}),2}\mathbf{I} \\ \phi_{(1),3}\mathbf{I} & \cdots & \phi_{(G_{No}),3}\mathbf{I} \end{pmatrix} \quad , \quad \mathbf{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3.27}$$

If we in a similar way define a "expanded" stress matrix,

$$\hat{\boldsymbol{\sigma}}_{(9\times 9)} = \begin{pmatrix} \sigma_{11}\mathbf{I} & \sigma_{12}\mathbf{I} & \sigma_{13}\mathbf{I} \\ \sigma_{21}\mathbf{I} & \sigma_{22}\mathbf{I} & \sigma_{23}\mathbf{I} \\ \sigma_{31}\mathbf{I} & \sigma_{32}\mathbf{I} & \sigma_{33}\mathbf{I} \end{pmatrix} \tag{3.28}$$

we can write the element stiffness matrix (3.17), the element load vector (3.18) and the element load correction vector (3.19) as

$$\mathbf{S}^{(k)} = \int_{V_{(k)}} \left\{ \mathbf{B}^T \boldsymbol{\mathcal{L}} \mathbf{B} + \hat{\mathbf{B}}^T \hat{\boldsymbol{\sigma}} \hat{\mathbf{B}} \right) dV \tag{3.29}$$

$$\dot{\mathbf{P}}^{(k)} = \int_{S_{(k)}} \mathbf{N}^T \dot{\mathbf{T}} dS \tag{3.30}$$

$$\mathbf{R}^{(k)} = \int_{V_{(k)}} \mathbf{B}^T \boldsymbol{\sigma} dV - \int_{S_{(k)}} \mathbf{N}^T \mathbf{T} dS \tag{3.31}$$

In (3.29), both two terms depends on the point in the loading history, the first term through $\mathbf{B}$ dependency on the total displacement state u_i and the second term through $\hat{\boldsymbol{\sigma}}$.

Collected into the global stiffness matrix and load vectors the linear equations system (3.20) can in matrix notation be written as

$$\mathbf{S}\dot{\mathbf{D}} = \dot{\mathbf{P}} - \mathbf{R} \tag{3.32}$$

3.2. Numerical implementation

Similar to the linear finite element model, the numerical implementation of the incremental finite element model as described above can be considered as consisting of three parts; the pre-processing, the analyzing and the post-processing part. The incremental finite element solution can be structured as:

- Pre-processing
  - Definition of variable used in the model
    - Geometrical parameters: lgeo, hgeo, bgeo, ...
    - Topology parameters: Ndof, Nel, Nnode
    - Properties Emodul, SigmaY, Nexp, ...
    - Defining the case: dUdisp, Ustop, ...
  - Topology of the model
    - Define the node coordinates
    - Correlate the global node numbers to local node number in the element
    - Correlate local degree of freedom numbers with the global degree of freedom numbers
    - Identify nodes (and elements) along external boundaries
    - Correlate material and element properties to element numbers
  - Plot the undeformed structure
  - Initialize the state variable in the integration points in the elements: $\mathbf{D} = \mathbf{0}$, $\boldsymbol{\sigma} = \mathbf{0}$, $\boldsymbol{\varepsilon} = \mathbf{0}$
- Analyzing part (repeat for each new increment)
  - Initialize global stiffness matrix and right hand side: $\mathbf{S} = \mathbf{0}$, $\dot{\mathbf{P}} = \mathbf{0}$
  - For each element:
    - Build up the local element stiffness matrix: $\mathbf{S}^{(k)}$
    - Build up eventually the increment of the local element load vector (static boundary condition): $\dot{\mathbf{P}}^{(k)}$.
    - Assemble the local element matrices $\mathbf{S}^{(k)}$, $\dot{\mathbf{P}}^{(k)}$ into the global matrices $\mathbf{S}$, $\dot{\mathbf{P}}$
  - Kinematic boundary condition

– Solve the system: $\mathbf{S}\dot{\mathbf{D}} = \dot{\mathbf{P}}$

– Extract the increment of the displacement, strains, stresses, etc. in the integration points from the solution: $\dot{\mathbf{u}} = \mathbf{N}\dot{\mathbf{D}}^{(k)}$, $\dot{\boldsymbol{\varepsilon}} = \mathbf{B}\dot{\mathbf{D}}^{(k)}$, $\dot{\boldsymbol{\sigma}} = \boldsymbol{\mathcal{L}}\mathbf{B}\dot{\mathbf{D}}^{(k)}$

– Update state variable in the integration points: $\mathbf{D} = \mathbf{D} + \dot{\mathbf{D}}$, $\boldsymbol{\sigma} = \boldsymbol{\sigma} + \dot{\boldsymbol{\sigma}}$, $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon} + \dot{\boldsymbol{\varepsilon}}$

– New increment?

- Post-processing part

  – Plot the deformed structure

  – Plot contours of strains, stresses, etc.

3.3. Incremental and iterative methods

The increment model presented above, is a Euler integration method including a equilibrium correction term. It is a quite straight forward analysis to performed, but will usually require a large number of increment in order to give an accurate solution. A number of much more sophisticated methods has been developed. One of the more simple method is a combined Euler integration and a full or modified Newton-Raphson iteration scheme, see e.g. [1, chapter 1].

Passing a maximum value of the prescribed parameter (load or displacement) in a numerical incremental scheme require some specific concern. A number of different numerical treatment has been proposed. Two of the more successful method is a combined Rayleigh-Ritz/Finite element method as described in [8], [9, page 132] or the very stable Riks/Arc-length methods, see e.g. [1, chapter 9.3].

References

[1] M. A. Crisfield. *Non-linear Finite Element Analysis of Solids and Structures*, volume 1. John Wiley & Sons, 1991.

[2] S. Krenk. *Non-linear analysis with finite elements.* Aalborg University, Aalborg, Denmark, 1993.

[3] V. Tvergaard. *Plasticitet og Krybning i Konstruktionsmaterialer (in Danish).* Dept. of Solid Mechanics, Technical University of Denmark, Lyngby, Denmark, 1993.

[4] V. Tvergaard. Notes on finite element formulations for finite strain plasticity or viscoplasticity. Lecture note from Dept. of Solid Mechanics, Technical University of Denmark, Lyngby, Denmark, 1999.

[5] R. D. Cook, D. S. Malkus, M. E. Plesha, and R. J. Witt. *Concepts and applications of finite element analysis.* John Wiley & Sons, 4rd edition, 2001.

[6] P.-E. Austrell, H. Carlsson, O. Dahlblom, K.-G. Olsson, K. Persson, A. Peterson, H Petersson, M. Ristimaa, and G. Sandberg. *CALFEM—A Finite Element Toolbox to MATLAB, Version 3.2.* Division of Structural Mechanics and Division of Solid Mechanics, LTH, Lund Unversity, 1997.

[7] E. Byskov. A short introductory example of use of calfem. Technical report, Dept. of Build. Tech. and Struct. Engng., Aalborg University, 1999.

[8] V. Tvergaard. Effect of thickness inhomogeneities in internally pressurized elastic-plastic spherical shells. *J. Mech. Phys. Solids*, 24:291–304, 1976.

[9] A. Needleman and V. Tvergaard. Finite element analysis of localization in plasticity. In J.T. Oden and G.F. Carey, editors, *Finite Elements - Special Problems in Solid Mechanics*, pages 94–157. Prentice-Hall, Englewood Cliffs, New Jersey, 1985.

Risø's research is aimed at solving concrete problems in the society.

Research targets are set through continuous dialogue with business, the political system and researchers.

The effects of our research are sustainable energy supply and new technology for the health sector.