# Chapter 5 - Finite element method for nonlinear problems

In this chapter, the FEM is applied to solve nonlinear problems:

1. The FEM can also solve boundary value problems for inelastic solids. In this section, we show how to extend the FEM to nonlinear problems.
2. As always, example codes are also discussed so how the method is actually implemented to explore useful predictions.

## 5.1 Newton-Raphson method for solving nonlinear equations

1. General form of the finite element equilibrium equation
   (1) The equations of equilibrium for the finite element mesh usually have the general form:

$$\boldsymbol{F}^{(b)}(\boldsymbol{u}^{(a)}) = \boldsymbol{0} \tag{5.1}$$

   where $\boldsymbol{F}^{(b)}()$ denotes a set of $b = 1, 2, ..., N$ vector functions of the nodal displacements $\boldsymbol{u}^{(a)}(\boldsymbol{x})$, $a = 1, 2, ..., N$, and $N$ is the number of nodes in the mesh.
2. The the Newton-Raphson method is suggested to solve the above nonlinear equations.
   (1) First, a guess for the solution is delivered to the equations, e.g. $\boldsymbol{u}^{(a)} = \boldsymbol{w}^{(a)}$. Certainly, $\boldsymbol{w}$ will not satisfy the equations, so the solution needs to be improved through adding a small correction $d\boldsymbol{w}$. Ideally, the correction should be chosen so that:

$$\boldsymbol{F}^{(b)}(\boldsymbol{w}^{(a)} + d\boldsymbol{w}^{(a)}) = \boldsymbol{0} \tag{5.2}$$

   Then, a Taylor expansion is taken to get:

$$\boldsymbol{F}^{(b)}(\boldsymbol{w}^{(a)} + d\boldsymbol{w}^{(a)}) \approx \boldsymbol{F}^{(b)}(\boldsymbol{w}^{(a)}) + \frac{d\boldsymbol{F}^{(b)}}{d\boldsymbol{w}^{(a)}}(\boldsymbol{w}^{(a)})\, d\boldsymbol{w}^{(a)} = \boldsymbol{0} \tag{5.3}$$

   The result is a system of linear equations of the form:

$$\boldsymbol{F}^{(b)}(\boldsymbol{w}^{(a)}) + \boldsymbol{K}\, d\boldsymbol{w}^{(a)} = \boldsymbol{0} \tag{5.4}$$

   where $\boldsymbol{K} = \dfrac{d\boldsymbol{F}^{(b)}}{d\boldsymbol{w}^{(a)}}(\boldsymbol{w}^{(a)})$ is a matrix called the *equation stiffness*.
   (2) Second, the equations (Eq. 5.4) can now be solved for $d\boldsymbol{w}^{(a)}$.
   (3) Third, the guess for $\boldsymbol{w}$ can be corrrected as a new guess:

$$\boldsymbol{w}'^{\,(a)} = \boldsymbol{w}^{(a)} + d\boldsymbol{w}^{(a)} \tag{5.5}$$

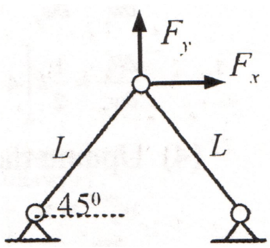
   (4) Again, deliver the new guess to the Eq. 5.5 ($\boldsymbol{w}^{(a)} \to \boldsymbol{w}'^{\,(a)}$) and solve the corresponding corrrection $d\boldsymbol{w}^{(a)}$.
   (5) The iteration is repeated until:

$$\boldsymbol{F}^{(b)}(\boldsymbol{w}'^{\,(a)}) = \epsilon \tag{5.6}$$

   where $\epsilon$ is a small tolerance ( $\epsilon \to \boldsymbol{0}$ ) to obtain accurate estimation of nodal displacements $\boldsymbol{u}^{(a)}$.

## 5.2 Illustrative example of Newton-Raphson method for solving nonlinear equations

1. Consider the shock absorber shown in the figure with two axial bars. The two bars have identical axial stiffness $k$ and length $L$. The external loading $F_x$ and $F_y$ induce displacements $u_x$ and $u_y$ at the loading joint.




2. As discussed in Section 2.2, it is equivalent to replacing the equilibrium equations with the principle of minimum potential energy. Here, we are trying to derive the equilibrium equations by minimizing the total potential energy under finite displacements (without assuming small deflection). The total potential energy of the shock absorber system can be written as:

$$
\begin{aligned}
W(u_x,u_y) &= \frac{1}{2}k\left[L-\sqrt{\left(\frac{L}{\sqrt{2}}+u_x\right)^2+\left(\frac{L}{\sqrt{2}}+u_y\right)^2}\,\right]^2+\frac{1}{2}k\left[L-\sqrt{\left(\frac{L}{\sqrt{2}}-u_x\right)^2+\left(\frac{L}{\sqrt{2}}+u_y\right)^2}\,\right]^2-F_x\,u_x-F_y\,u_y \\
&= \frac{k}{2}\left[\left(L-\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(u_x+u_y)}\right)^2+\left(L-\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(-u_x+u_y)}\right)^2\right]-F_xu_x-F_yu_y
\end{aligned}
\tag{5.7}
$$

3. The equilibrium state of the absorber system requires that:

$$
\begin{cases}
\dfrac{\partial W(u_x,u_y)}{\partial u_x}=0 & : \ R_x(u_x,u_y)=F_x \\
\dfrac{\partial W(u_x,u_y)}{\partial u_y}=0 & : \ R_y(u_x,u_y)=F_y
\end{cases}
\tag{5.8}
$$

where $R_x(u_x,u_y)$ and $R_y(u_x,u_y)$ are

$$
R_x=-k\left[\frac{(-\sqrt{2}L+2u_x)(L-\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(-u_x+u_y)}\,)}{2\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(-u_x+u_y)}}+\frac{(\sqrt{2}L+2u_x)(L-\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(u_x+u_y)}\,)}{2\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(u_x+u_y)}}\right]
\tag{5.9}
$$

$$
R_y=-k\left[\frac{(\sqrt{2}L+2u_y)(L-\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(-u_x+u_y)}\,)}{2\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(-u_x+u_y)}}+\frac{(\sqrt{2}L+2u_y)(L-\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(u_x+u_y)}\,)}{2\sqrt{L^2+u_x^2+u_y^2+\sqrt{2}L(u_x+u_y)}}\right]
\tag{5.10}
$$

Note that the equilibrium equations are *nonlinear* with respect to the unknowns $u_x$ and $u_y$.

4. The Newton-Raphson method is applied to solve the above nonlinear equilibrium equations for $u_x$ and $u_y$.

(1) Perturbation for guessing by $u_x \rightarrow w_x + dw_x$ and $u_y \rightarrow w_y + dw_y$:

$$
\begin{cases}
R_x(u_x,u_y)=F_x \\
R_y(u_x,u_y)=F_y
\end{cases}
\quad\rightarrow\quad
\begin{cases}
R_x(w_x+dw_x,w_y+dw_y)=F_x \\
R_y(w_x+dw_x,w_y+dw_y)=F_y
\end{cases}
\tag{5.11}
$$

(2) Ignoring higher order terms, the Taylor series expansion are:

$$
\begin{cases}
R_x(w_x,w_y)+\dfrac{\partial R_x}{\partial w_x}dw_x+\dfrac{\partial R_x}{\partial w_y}dw_y=F_x \\
R_y(w_x,w_y)+\dfrac{\partial R_y}{\partial w_x}dw_x+\dfrac{\partial R_y}{\partial w_y}dw_y=F_y
\end{cases}
\tag{5.12}
$$

(3) Rewrite the above equations for solving the incremental guessing $dw_x$ and $dw_y$:

$$
\begin{bmatrix}
\dfrac{\partial R_x}{\partial w_x} & \dfrac{\partial R_x}{\partial w_y} \\
\dfrac{\partial R_y}{\partial w_x} & \dfrac{\partial R_y}{\partial w_y}
\end{bmatrix}
\begin{bmatrix} dw_x \\ dw_y \end{bmatrix}
=
\begin{bmatrix} -R_x(w_x,w_y+F_x \\ -R_y(w_x,w_y)+F_y \end{bmatrix}
\tag{5.13}
$$

where $\dfrac{\partial R_x}{\partial w_x}$, $\dfrac{\partial R_x}{\partial w_y}$, $\dfrac{\partial R_y}{\partial w_x}$, and $\dfrac{\partial R_y}{\partial w_y}$ are:

$$
\frac{\partial R_x}{\partial w_x}=\frac{k}{2}\left\{4+L(L^2+2\sqrt{2}Lw_y+2w_y^2)\left[\frac{-1}{(L^2+w_x^2+w_y^2+\sqrt{2}L(-w_x+w_y))^{3/2}}-\frac{1}{(L^2+w_x^2+w_y^2+\sqrt{2}L(w_x+w_y))^{3/2}}\right]\right\}
\tag{5.14}
$$

$$
\frac{\partial R_x}{\partial w_y}=\frac{kL(\sqrt{2}L+2w_y)}{4}\left[\frac{-\sqrt{2}L+2w_x}{(L^2+w_x^2+w_y^2+\sqrt{2}L(-w_x+w_y))^{3/2}}+\frac{\sqrt{2}L+2w_x}{(L^2+w_x^2+w_y^2+\sqrt{2}L(w_x+w_y))^{3/2}}\right]
\tag{5.15}
$$

$$
\frac{\partial R_y}{\partial w_x}=\frac{kL(\sqrt{2}L+2w_y)}{4}\left[\frac{-\sqrt{2}L+2w_x}{(L^2+w_x^2+w_y^2+\sqrt{2}L(-w_x+w_y))^{3/2}}+\frac{\sqrt{2}L+2w_x}{(L^2+w_x^2+w_y^2+\sqrt{2}L(w_x+w_y))^{3/2}}\right]
\tag{5.16}
$$

$$
\frac{\partial R_y}{\partial w_y}=\frac{k}{2}\left[4+\frac{-L^3-2Lw_x^2+2\sqrt{2}L^2w_x}{(L^2+w_x^2+w_y^2+\sqrt{2}L(-w_x+w_y))^{3/2}}+\frac{-L^3-2Lw_x^2-2\sqrt{2}L^2w_x}{(L^2+w_x^2+w_y^2+\sqrt{2}L(w_x+w_y))^{3/2}}\right]
\tag{5.17}
$$

(4) Update the guessing:

$$\begin{cases} w_x' = w_x + dw_x \\ w_y' = w_y + dw_y \end{cases} \tag{5.18}$$

(5) Repeat steps (3) and (4) until:

$$\begin{cases} \left| F_x - R_x\left(w_x', w_y'\right) \right| < \epsilon \\ \left| F_y - R_y\left(w_x', w_y'\right) \right| < \epsilon \end{cases} \tag{5.19}$$

( $\epsilon$ : error tolerance)

5. MATLAB code demonstration

The code written in MATLAB is going to be tested by using the following parameters: $L = \sqrt{2}$, $k = 1$, $F_{x0} = 0$, $F_{y0} = 0$, $\Delta F_x = 0$, $\Delta F_y = -0.025$, and `tol` = $10^{-8}$.

(1) *main* program

```
L = sqrt{2};
k = 1;
Fx0 = 0;
Fy0 = 0;
dFx = 0;
dFy = -0.025;
tol = 1e-8;
nstep = 12;
Fdat = zeros(2,nstep+1);
udat = zeros(2,nstep+1);
Fx = Fx0;
Fy = Fy0;
for i = 2:nstep+1
  Fx = Fx+dFx;
  Fy = Fy+dFy;
  wx = 0;
  wy = 0;
  dw = [0; 0];
  rhs = [100*tol; 100*tol];
  while(abs(rhs(1)) > tol || abs(rhs(2)) > tol)
    wx = wx+dw(1);
    wy = wy+dw(2);
    lhs = lhs(L,k,wx,wy);
    rhs = rhs(L,k,wx,wy,Fx,Fy);
    dw = lhs\rhs;
  end
  Fdat(1,i) = Fx;
  Fdat(2,i) = Fy;
  udat(1,i) = wx;
  udat(2,i) = wy;
end
```

(2) *lhseq* function

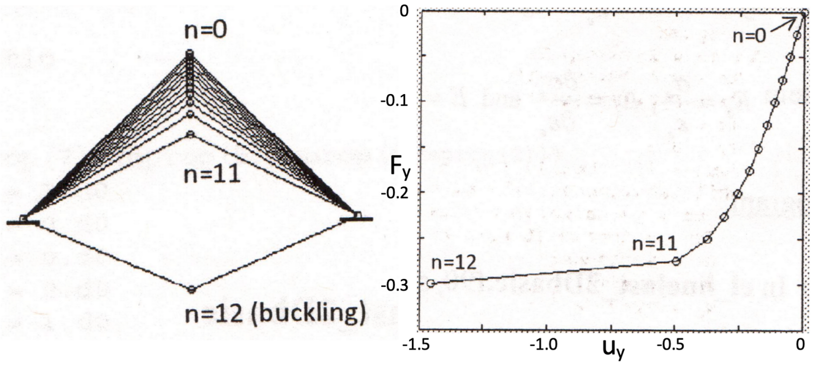
```
function lhs = lhseq(L,k,wx,wy)
    sqrt2 = sqrt(2);
    deno1 = sqrt(L*L+wx*wx+wy*wy+sqrt2*L*(-wx+wy))*(L*L+wx*wx+wy*wy+sqrt2*L*(-wx+wy));
    deno2 = sqrt(L*L+wx*wx+wy*wy+sqrt2*L*(wx+wy))*(L*L+wx*wx+wy*wy+sqrt2*L*(wx+wy));
    lhs = zeros(2,2);
    lhs(1,1) = 0.5*k*(4+L*(L*L+2*sqrt2*L*wy+2*wy*wy)*(-1/deno1-1/deno2));
    lhs(1,2) = 0.25*k*L*(sqrt2*L+2*wy)*((-sqrt2*L+2*wx)/deno1+(sqrt2*L+2*wx)/deno2);
    lhs(2,1) = lhs(1,2);
    lhs(2,2) = 0.5*k*(4+(-L*L*L-2*L*wx*wx+2*sqrt2*L*L*wx)/deno1+...
      (-L*L*L-2*L*wx*wx-2*sqrt2*L*L*wx)/deno2);
  end
```

(3) *rhseq* function

```
function rhs = rhseq(L,k,wx,wy,Fx,Fy)
    sqrt2 = sqrt(2);
    atmp1 = L*L+wx*wx+wy*wy+sqrt2*L*(-wx+wy);
    atmp2 = L*L+wx*wx+wy*wy+sqrt2*L*(wx+wy);
    nume1 = L-sqrt(atmp1);
    nume2 = L-sqrt(atmp2);
    deno1 = 2*sqrt(atmp1);
    deno2 = 2*sqrt(atmp2);
    rhs = zeros(2,1);
    rhs(1) = k*(-sqrt2*L+2*wx)*nume1/deno1+k*(sqrt2*L+2*wx)*nume2/deno2+Fx;
    rhs(2) = k*(sqrt2*L+2*wy)*nume1/deno1+k*(sqrt2*L+2*wy)*nume2/deno2+Fy;
end
```

6. Results ( n is the step counter. )

By plotting the deformed shock absorber over the step counters n, the system exhibits snap through buckling, which is highly nonlinear deformation.

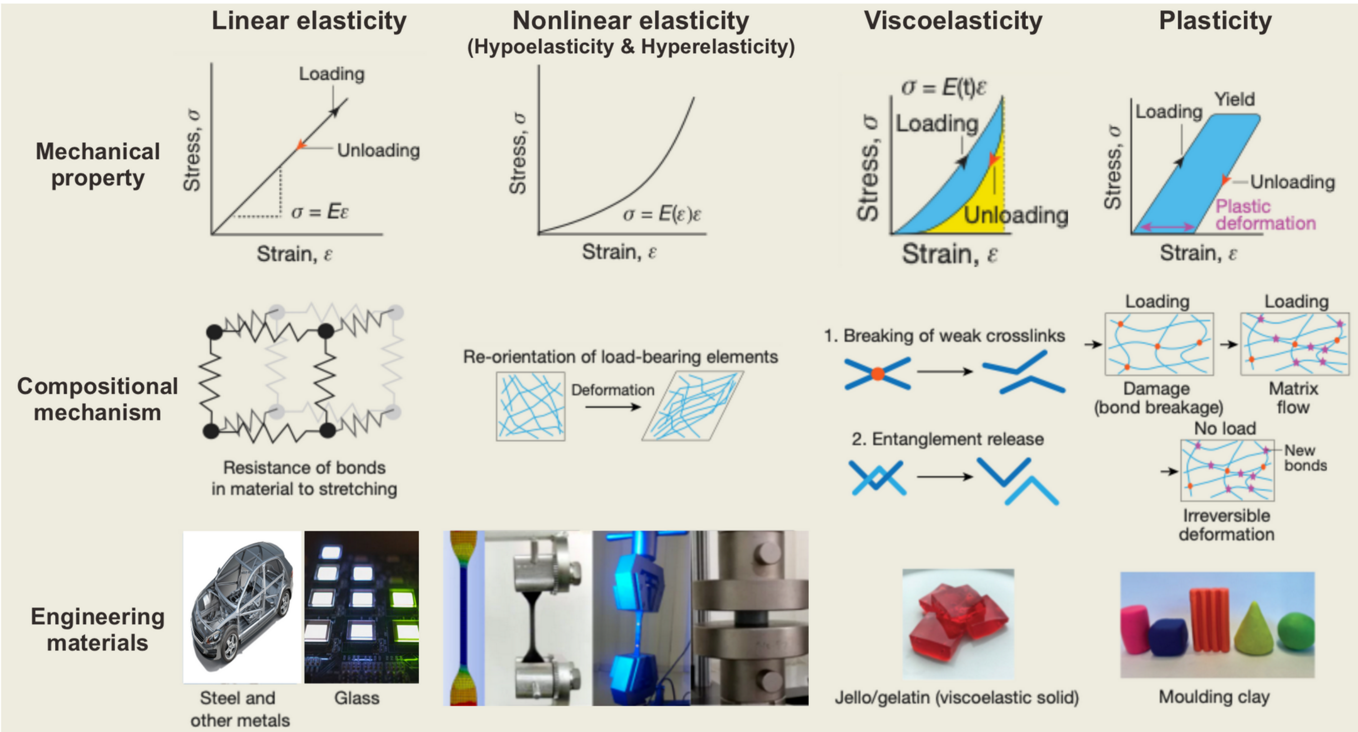



---

## 5.3 FEM for hypoelastic materials

1. General concepts

(1) The stress-strain behavior of the hypoelastic material can be described by the nonlinear constitutive law introduced in this section.



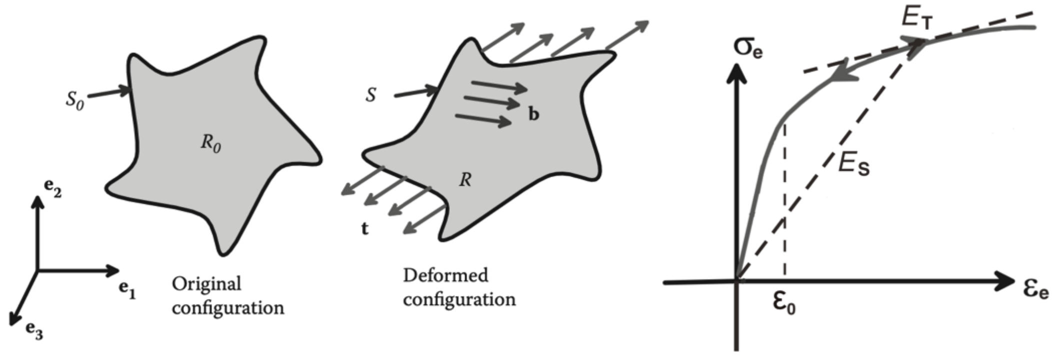

(2) To solve boundary value problems of solids made of hypoelastic materials, the following informations are required:

a. The shape of the solid in its unloaded condition.

b. A body force distribution acting on the solid.

c. Boundary conditions, specifying displacements $\boldsymbol{u}^*(\boldsymbol{x})$ on a portion $\partial_1 R$ and tractions on a portion $\partial_2 R$ of the boundary of $R$.

d. The material constants $n$, $\sigma_0$, and $\varepsilon_0$ for the following hypoelastic constitutive law.




![[FEM5.4.png|600]]

2. Hypoelastic constitutive law

(1) In hypoelastic constitutive law, the calculated displacements, strains, and stresses of such nonlinear materials from FEM need to satisfy the following equations:

a. Strain-displacement equation

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{5.20}$$

b. Static equilibrium for stresses

$$\frac{\partial \sigma_{ij}}{\partial x_i} + b_j = 0 \tag{5.21}$$

c. Boundary conditions on displacement and stress

$$u_i = u_i^*$$

on $\partial_1 R$ and

$$\sigma_{ij} n_i = t_j^* \tag{5.22}$$

on $\partial_2 R$

d. Hypoelastic constitutive law, which relates stress to strain

$$\sigma_{ij} = S_{ij} + \sigma_{kk}\frac{\delta_{ij}}{3} \tag{5.23}$$

where

$$S_{ij} = \frac{2}{3}\sigma_e \frac{e_{ij}}{\varepsilon_e} \tag{5.24}$$

$$\sigma_{kk} = \frac{E}{1-2\nu}\frac{1}{3}\varepsilon_{kk} \tag{5.25}$$

$$e_{ij} = \varepsilon_{ij} - \frac{1}{3}\varepsilon_{kk}\delta_{ij} \tag{5.26}$$

$$\varepsilon_e = \sqrt{\frac{2}{3}e_{ij}e_{ij}} \tag{5.27}$$

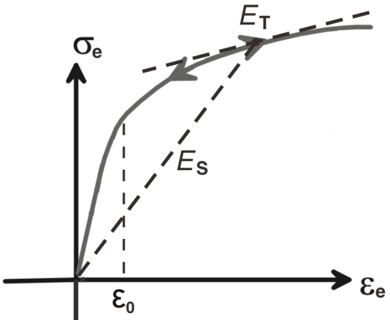
$$\frac{\sigma_e}{\sigma_0} = \begin{cases} \sqrt{\dfrac{1+n^2}{(n-1)^2} - \left(\dfrac{n}{n-1} - \dfrac{\varepsilon_e}{\varepsilon_0}\right)^2} - \dfrac{1}{n-1} & \varepsilon_e \le \varepsilon_0 \\ \left(\dfrac{\varepsilon_e}{\varepsilon_0}\right)^{1/n} & \varepsilon_e \ge \varepsilon_0 \end{cases} \tag{5.28}$$

$$E = \frac{n\sigma_0}{\varepsilon_0} \tag{5.29}$$

Note that $E$ is the slope of the uniaxial stress-strain curve at $\varepsilon_e = 0$.

(2) The uniaxial stress-strain curve for the hypoelastic material is illustrated in the following figure. The material is nonlinearly elastic, in that it is perfectly reversible, but the stresses are related to strains by a nonlinear function. This material model is sometimes used to approximate the more

complicated stress-strain laws for complex elastic materials.




3. Governing equations in terms of the virtual work principle

(1) As in FEM analysis discussed before, the stress equilibrium equation is replaced by the equivalent statement of the principle of virtual work. Thus, $u_i$, $\varepsilon_{ij}$, and $\sigma_{ij}$ are determined as follows:

a. First, the calculated displacement fields that satisfie

$$\int_R \sigma_{ij}(u_k)\frac{\partial \delta v_i}{\partial x_i}dV - \int_R b_i\delta v_i\, dV - \int_{\partial_2 R} t_i^*\delta v_i\, dA = 0 \tag{5.30}$$

$u_i = u_i^*$ on $\partial_1 R$ for all virtual velocity fields $\delta v_i$ that satisfy $\delta v_i = 0$ on $\partial_1 R$. Here, the notation $\sigma_{ij}(u_k)$ is used to show that the stress in the solid depends on the displacement field through the strain-displacement relation and the constitutive equations.

b. Compute the strains from the definition:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

c. Compute the stresses from the stress-strain law.

Based on the virtual work principle, the stress will automatically satisfy the equation of equilibrium, so all the field equations and boundary conditions will be satisfied in the solid domain.

(2) The procedure to solve the above equations is conceptually identical to the linear elastic solution discussed before. The only complication is that the stress is now a *nonlinear* function of the strains, so the virtual work equation is a nonlinear function of the displacement field. It must therefore be solved by *iteration*.

4. Finite element equations

(1) The finite element solution follows almost exactly the same procedure as before. First, the displacement field is discretized by choosing to calculate the displacement field at a set of $n$ nodes, as shown in the figure. The coordinates of these nodal points by $x_i^a$, where the superscript a ranges from $1$ to $n$. The unknown displacement vector at each nodal point will be denoted by $u_i^a$. Then, the finite element equations can be set up as follows.

(2) The displacement field at an arbitrary point within the solid is again specified by interpolating between nodal values in same convenient way:

$$u_i(\boldsymbol{x}) = \sum_{a=1}^{n} N^a(\boldsymbol{x})\, u_i^a \tag{5.31}$$

$$\delta v_i(\boldsymbol{x}) = \sum_{a=1}^{n} N^a(\boldsymbol{x})\, \delta v_i^a \tag{5.32}$$

(3) As usual, the stress corresponding to a given displacement field can be obtained by first computing the strain:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) = \frac{1}{2}\sum_{a=1}^{n}\left(\frac{\partial N^a}{\partial x_j}u_i^a + \frac{\partial N^a}{\partial x_i}u_j^a\right) \tag{5.33}$$

and then using the constitutive law to compute the stresses. This functional relationship is written as:

$$\sigma_{ij} = \sigma_{ij}(\,\varepsilon_{k\ell}(u_i^a)\,) \tag{5.34}$$

(4) By substituting into the principle of virtual work,

$$\left[\int_R \sigma_{ij}(\,\varepsilon_{k\ell}(u_i^a)\,)\frac{\partial N^a}{\partial x_j}dV - \int_R b_i N^a\, dV - \int_{\partial R_2} t_i^* N^a\, dA\right]\delta v_i^a = 0 \tag{5.35}$$

and, since this must hold for all $\delta v_i^a$,

$$\int_R \sigma_{ij}(\,\varepsilon_{k\ell}(u_i^b)\,)\frac{\partial N^a}{\partial x_j}dV - \int_R b_i N^a\, dV - \int_{\partial R_2} t_i^* N^a\, dA = 0 \tag{5.36}$$

$\forall\{a,i\} : x_i^a$ not on $\partial_1 R$ and $u_i^a = u_i^*(x_i^a)\ \forall\{a,i\} : x_i^a$ on $\partial_1 R$

Now, the equations are *nonlinear*, since the stress is a nonlinear function of the unknown nodal displacements $u_i^a$.

5. Solving the finite element equations using the Newton - Raphson method

The nonlinear equations can be solved through iteration in the Newton - Raphson method.

(1) Start with some initial guess for $u_i^b$, e.g., $w_i^b$.

(2) Then try to correct this guess to bring it closer to the proper solution by setting $w_i^b \rightarrow w_i^b + dw_i^b$. Ideally, the correction satisfies:

$$\int_R \sigma_{ij}(\varepsilon_{k\ell}(w_i^b + dw_i^b))\frac{\partial N^a}{\partial x_j}dV - \int_R b_i N^a\, dV - \int_{\partial R_2} t_i^* N^a\, dA = 0 \tag{5.37}$$

(3) This equation cannot be solved directly, so the stress term is linearized in $dw_i^b$:

$$\int_R \left[\sigma_{ij}(\varepsilon_{k\ell}(w_i^b)) + \frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell m}}\frac{\partial \varepsilon_{\ell m}}{\partial w_k^b}dw_k^b\right]\frac{\partial N^a}{\partial x_j}dV - \int_R b_i N^a\, dV - \int_{\partial R_2} t_i^* N^a\, dA = 0 \tag{5.38}$$

(4) Recall that

$$\varepsilon_{\ell m} = \frac{1}{2}\sum_{a=1}^{n}\left(\frac{\partial N^a}{\partial x_m}u_\ell^a + \frac{\partial N^a}{\partial x_\ell}u_m^a\right) \tag{5.39}$$

$$\frac{\partial \varepsilon_{\ell m}}{\partial u_k^b} = \frac{1}{2}\sum_{a=1}^{n}\left(\frac{\partial N^a}{\partial x_m}\delta_{ab}\,\delta_{\ell k} + \frac{\partial N^a}{\partial x_\ell}\delta_{ab}\,\delta_{mk}\right) \tag{5.40}$$

(5) In addition, we know $\varepsilon_{mk} = \varepsilon_{km}$ and derive:

$$\begin{aligned}
\frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell m}}\frac{\partial \varepsilon_{\ell m}}{\partial u_k^b} &= \frac{1}{2}\frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell m}}\sum_{a=1}^{n}\left(\frac{\partial N^a}{\partial x_m}\delta_{ab}\,\delta_{\ell k} + \frac{\partial N^a}{\partial x_\ell}\delta_{ab}\,\delta_{mk}\right)\\
&= \frac{1}{2}\sum_{a=1}^{n}\left(\frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell m}}\frac{\partial N^a}{\partial x_m}\delta_{ab}\,\delta_{\ell k} + \frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell m}}\frac{\partial N^a}{\partial x_\ell}\delta_{ab}\,\delta_{mk}\right)\\
&= \frac{1}{2}\left(\frac{\partial \sigma_{ij}}{\partial \varepsilon_{km}}\frac{\partial N^b}{\partial x_m} + \frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell k}}\frac{\partial N^b}{\partial x_\ell}\right)\\
&= \frac{1}{2}\left(\frac{\partial \sigma_{ij}}{\partial \varepsilon_{km}}\frac{\partial N^b}{\partial x_m} + \frac{\partial \sigma_{ij}}{\partial \varepsilon_{mk}}\frac{\partial N^b}{\partial x_m}\right)\\
&= \frac{\partial \sigma_{ij}}{\partial \varepsilon_{km}}\frac{\partial N^b}{\partial x_m}
\end{aligned}$$

Hence,

$$\frac{\partial \sigma_{ij}}{\partial \varepsilon_{\ell m}}\frac{\partial \varepsilon_{\ell m}}{\partial u_k^b} = \frac{\partial \sigma_{ij}}{\partial \varepsilon_{km}}\frac{\partial N^b}{\partial x_m} = \frac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}}\frac{\partial N^b}{\partial x_\ell}$$

Finally,

$$\int_R \frac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}}\frac{\partial N^a}{\partial x_j}\frac{\partial N^b}{\partial x_\ell}dV\, dw_k^b + \int_R \sigma_{ij}(\varepsilon_{k\ell}(w_i^b))\frac{\partial N^a}{\partial x_j}dV - \int_R b_i N^a\, dV - \int_{\partial R_2} t_i^* N^a\, dA = 0$$

(6) This is evidently a system of linear equations for the correction $dw_k^b$ of the form:

$$K_{aibk}\, dw_k^b - R_i^a + F_i^a = 0 \tag{5.41}$$

where

$$K_{aibk} = \int_R \frac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}}\frac{\partial N^a}{\partial x_j}\frac{\partial N^b}{\partial x_\ell}dV \tag{5.42}$$

$$R_i^a = \int_R b_i N^a\, dV + \int_{\partial R_2} t_i^* N^a\, dA \tag{5.43}$$

$$F_i^a = \int_R \sigma_{ij}(\varepsilon_{k\ell}(w_i^b))\frac{\partial N^a}{\partial x_j}dV \tag{5.44}$$

a. This expression above is nearly identical to the equation we needed to solve for linear elastostatic problems.

b. There are only two differences:

(a) The stiffness contains the (strain-dependent) material tangent moduli $\dfrac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}}$ instead of the elastic constants $C_{ijk\ell}$.

(b) We need to compute an extra term $F_i^a$ in the *residual force vector*.

Again, those integrals are divided up into contributions from each element and evaluated numerically using Gaussian quadrature at the corresponding integration points.

6. Tangent moduli for the hypoelastic solids

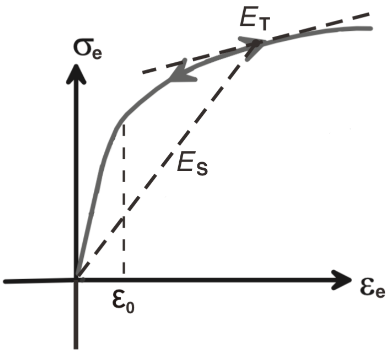
As seen before, nonlinear FEM usually needs the material tangent moduli $\dfrac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}}$. For the hypoelastic constitutive law used in this section, through algegraic deliberations, it is can be shown that:

$$\frac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}} = \begin{cases} \dfrac{4}{9}(E_T - E_S)\dfrac{e_{ij}e_{k\ell}}{\varepsilon_e^2} + \dfrac{2}{3}E_S\left(\delta_{ik}\delta_{j\ell} - \dfrac{\delta_{ij}\delta_{k\ell}}{3}\right) + \dfrac{E}{9(1-2\nu)}\delta_{ij}\delta_{k\ell} & \varepsilon_e > 0 \\ \dfrac{2}{3}E_T\left(\delta_{ik}\delta_{j\ell} - \dfrac{\delta_{ij}\delta_{k\ell}}{3}\right) + \dfrac{E}{9(1-2\nu)}\delta_{ij}\delta_{k\ell} & \varepsilon_e = 0 \end{cases} \tag{5.45}$$

where $E_S$ and $E_T$ are the *secant elastic modulus* and *tangent elastic modulus* of the uniaxial stress-strain curve as shown in the figure respectively:

$$E_S = \frac{\sigma_e}{\varepsilon_e} \tag{5.46}$$

$$E_T = \frac{d\sigma_e}{d\varepsilon_e} \tag{5.47}$$




And the tangent elastic modulus can be calculated by using the relationships between $\sigma_e$ and $\varepsilon_e$ given in the hypoelastic constitutive law (Eq. 5.28):

$$E_T = \frac{d\sigma_e}{d\varepsilon_e} = \begin{cases} \dfrac{\sigma_0\left(\dfrac{n}{n-1} - \dfrac{\varepsilon_e}{\varepsilon_0}\right)}{\varepsilon_0\sqrt{\dfrac{1+n^2}{(n-1)^2} - \left(\dfrac{n}{n-1} - \dfrac{\varepsilon_e}{\varepsilon_0}\right)^2}} & \varepsilon_e \le \varepsilon_0 \\ \dfrac{\sigma_0}{n\varepsilon_e}\left(\dfrac{\varepsilon_e}{\varepsilon_0}\right)^{\frac{1}{n}} & \varepsilon_e \ge \varepsilon_0 \end{cases} \tag{5.48}$$

7. Summary of the Newton – Raphson procedure for hypoelastic solids

Now, we are trying to extend a linear elasticity code to nonlinear problems of hypoelastic materials. The analysis procedure of the Newton – Raphson method for such materials is:

(1) Start with an initial guess for the solution $w_i^a$.

(2) For the current guess, compute $K_{aibk}$, $R_i^a$ and $F_i^a$

where

$$K_{aibk} = \int_R \frac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}}\frac{\partial N^a}{\partial x_j}\frac{\partial N^b}{\partial x_\ell} dV$$

$$R_i^a = \int_R b_i N^a \, dV + \int_{\partial R_2} t_i^* N^a \, dA$$

$$F_i^a = \int_R \sigma_{ij}(\varepsilon_{k\ell}(w_i^b))\frac{\partial N^a}{\partial x_j} dV$$

$$\frac{\partial \sigma_{ij}}{\partial \varepsilon_{k\ell}} = \begin{cases} \dfrac{4}{9}(E_T - E_S)\dfrac{e_{ij}e_{k\ell}}{\varepsilon_e^2} + \dfrac{2}{3}E_S\left(\delta_{ik}\delta_{j\ell} - \dfrac{\delta_{ij}\delta_{k\ell}}{3}\right) + \dfrac{E}{9(1-2\nu)}\delta_{ij}\delta_{k\ell} & \varepsilon_e > 0 \\ \dfrac{2}{3}E_T\left(\delta_{ik}\delta_{j\ell} - \dfrac{\delta_{ij}\delta_{k\ell}}{3}\right) + \dfrac{E}{9(1-2\nu)}\delta_{ij}\delta_{k\ell} & \varepsilon_e = 0 \end{cases}$$

$$E_T = \frac{d\sigma_e}{d\varepsilon_e} = \begin{cases} \dfrac{\sigma_0 \left( \dfrac{n}{n-1} - \dfrac{\varepsilon_e}{\varepsilon_0} \right)}{\varepsilon_0 \sqrt{\dfrac{1+n^2}{(n-1)^2} - \left( \dfrac{n}{n-1} - \dfrac{\varepsilon_e}{\varepsilon_0} \right)^2}} & \varepsilon_e \leq \varepsilon_0 \\ \dfrac{\sigma_0}{n\varepsilon_e} \left( \dfrac{\varepsilon_e}{\varepsilon_0} \right)^{\frac{1}{n}} & \varepsilon_e \geq \varepsilon_0 \end{cases}$$

(3) Modify the system of equations to enforce any displacement boundary constraints.

(4) Solve $K_{aibk}\, dw_k^b = R_i^a - F_i^a$

(5) Let $w_i^a = w_i^a + dw_i^a$

(6) Check for convergence. Go to step 2, if the solution has not yet converged.