Universitat Politcnica de
Catalunya.BarcelonaTech

Finite Deformation Nonlinear Elasticity

Computational Solid Mechanics

Author:
Yuyang Wang
Christian Rossi
Master on Numerical Methods in
Engieering

Supervisor:
Prof.Marino Arroyo

June 9, 2016

Contents

List of Figures

1 Kirchhoff Saint-Venant material model

1.1 Stress tensor

Isotropic linear elastic can be derived from balance of linear momentum, the linearized strain displacement relation $\epsilon = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T)$, and the stored elastic energy function

$$W(\epsilon) = \frac{\lambda}{2}(tr\varepsilon)^2 + \mu\, tr(\varepsilon^2) = \frac{\lambda}{2}(\varepsilon_{ii})^2 + \mu\varepsilon_{jk}\varepsilon_{jk}$$

using the linear elastic expression $\sigma = C : \varepsilon$ written as $\sigma = \lambda(tr\varepsilon)1 + 2\mu\varepsilon$, we can write the previous formula as:

$$\sigma_{ij} = \frac{\partial}{\partial \epsilon_{ij}}(\frac{\lambda}{2}(\varepsilon_{kk})^2 + \mu\varepsilon_{ij}\varepsilon_{ij}) = \lambda\varepsilon_{kk}\delta_{ij} + 2\mu\varepsilon_{ij}$$

1.2 Isotropic verification

The definition of isotropic model is that any isotropic strain energy function can be written in terms of the principal invariants of C.

$$W(C) = W(I_1(C), I_2(C), I_3(C))$$

with

$$I_1(C) = traceC \quad I_2(C) = \frac{1}{2}[(traceC)^2 - traceC^2] \quad I_3(C) = detC = J^2$$

We here assume the material is isotropic and we could write the energy equation W in terms of the principal invariants of C.

$$\frac{\partial W}{\partial C} = \frac{\partial W}{\partial I_1}\frac{\partial I_1}{\partial C} + \frac{\partial W}{\partial I_2}\frac{\partial I_2}{\partial C} + \frac{\partial W}{\partial I_3}\frac{\partial I_3}{\partial C}$$

$\longrightarrow$

$$\frac{\partial W}{\partial C} = c_1 I + c_2(I_2(CI - C^T)) + c_3 I_3 C^{-T}$$

grouping different terms,

$$\frac{\partial W}{\partial C} = p_0 I + p_1 C^T + p_2 C^{-T}$$

with $C^{-1} = C^{-1}(I, C, C^2)$ continually, we group the last equation again,

$$\frac{\partial W}{\partial C} = p_0 I + p_1 C^T + p_2 C^2$$

As we know that the PK-2 stress tensor(S) can be yielded from above equation as

$$S = 2\frac{\partial W}{\partial C} = 2(p_0 I + p_1 C^T + p_2 C^2)$$

On the other hand, the isotropic Cauchy stress tensor has the form

$$\sigma = 2\rho(s_1 I + s_2 B + s_3 B^2)$$

with B a right Cauchy-Green deformation tensor. We can compare the last two equations sharing the same form, thereby concluding that the original model is isotropic.

Actually, on the other hand, we could write

$$W(I_1, I_2) = \frac{1}{2}(\lambda + 2\mu)[\frac{1}{2}(I_1 - 3)]^2 - 2\mu[\frac{1}{4}(-2I_1 + I_2 + 3)]$$
$$= \frac{1}{8}(\lambda + 2\mu)(I_1 - 3)^2 - \frac{\mu}{2}(-2I_1 + I_2 + 3)$$

We again prove that the model is isotropic model!

1.3 Second Piola-Kirchhoff stress

The second Piola-Kirchhoff stress follow the next equation:

$$S = \frac{\partial W}{\partial E}$$

And after differentiating the energy equation we obtain

$$S = \lambda\, tr(E)\mathbf{1} + 2\mu E$$

1.4 Nominal stress $\mathbf{P}(\Lambda)$

According to the known condition, we could compute **F**

$$F = \begin{bmatrix} \Lambda & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

With the definition of 2nd Piola-Kirchhoff stress: $\mathbf{S} = \mathbf{F^{-1}P}$. We can compute the nominal normal stress **P=FS**. Especially, the xX component of the first Piola-Kirchhoff stress and the stretch ratio Λ,as is shown is figure 1.

$$E = \frac{1}{2}(F^T F - I) = \frac{1}{2}\begin{pmatrix} \Lambda^2 - 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$P = \Lambda\lambda(tr\mathbf{E}) + 2\Lambda\mu\mathbf{E}$$

And the first nominal stress is given by:

$$P_{11} = \frac{\Lambda(\Lambda^2 - 1)}{2}(\lambda + \mu)$$

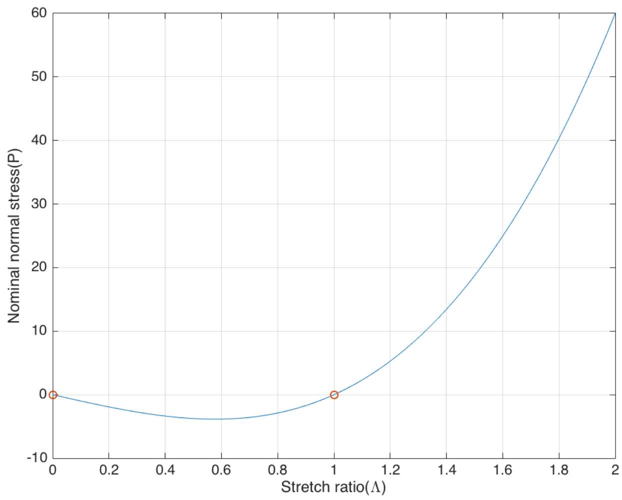


Figure 1: The nominal normal stress P with respect to the stretch ratio

1.5 Monotonic verification

The relation $P(\Lambda)$ is not monotonic as is shown in figure 1.When $\Lambda \leq 0.5$, $P(\Lambda)$ shown a decreasing trends while later it increase to zero and then goes up monotonically. With the last equation, we could compute the derivative of P with respect to Λ

$$\frac{dP}{d\Lambda} = \frac{3\Lambda^2 - 1}{2}(\lambda + \mu) = 0$$

Hence, the point fails with zeros stiffness which does not dependent on the elastic constant,

$$\Lambda_{crit} = \frac{1}{\sqrt{3}}$$

We know that $J = detF = \Lambda$. When $J \to 0^+$,meaning $\Lambda \to 0^+$,we could obtain the expression W in terms of Λ

$$W = \frac{1}{2}(trE)^2 + \mu trE^2$$
$$= \frac{1}{4}(\frac{\lambda}{2} + \mu)$$

When $\Lambda \to 0^+$,$W \to \frac{1}{4}(\frac{\lambda}{2} + \mu)$.

1.6 Modified Kirchhoff Saint-Venant material model

With the modified Kirchhoff Saint-Venant material model:

$$W(E) = \frac{\lambda}{2}(lnJ)^2 + \mu tr(E^2)$$

and noting $J = \sqrt{detE}$ and especially in 1D case,$J = \sqrt{detC} = det(2E + 1)$,$tr(E^2) = E^2$ with $E = \frac{1}{2}(\Lambda^2 - 1)$
$\longrightarrow$

$$W(E) = \frac{\lambda}{8}(ln(2E + 1))^2 + \mu E^2$$

The second Piola-Kirchhoff stress is

$$\frac{\partial W}{\partial E} = \frac{\lambda}{8}\frac{ln(2E + 1)}{2E + 1} + 2\mu E$$
$$= \frac{\lambda}{8}\frac{ln\Lambda}{\Lambda} + \mu(\Lambda^2 - 1)$$

Thus, the nominal normal stress is

$$P(\Lambda) = FS = \frac{\lambda}{8}ln(\Lambda) + \mu\Lambda(\Lambda^2 - 1)$$

W can be also written in term of Λ

$$W = \frac{1}{2}(ln\Lambda)^2 + \frac{\mu}{4}(\Lambda^2 - 1)^2$$

We can see from figure 2 that $P(\Lambda)$ is monotonic. However,when $\Lambda = 1$,$P(\Lambda) = 0$,which follows the actual fact that there is no stress for undeformed configuration.On the other hand, when $J \longrightarrow 0^+$,meaning $\Lambda \longrightarrow 0^+$,we have $W \longrightarrow +\infty$, which circumvent the drawbacks of the previous mode

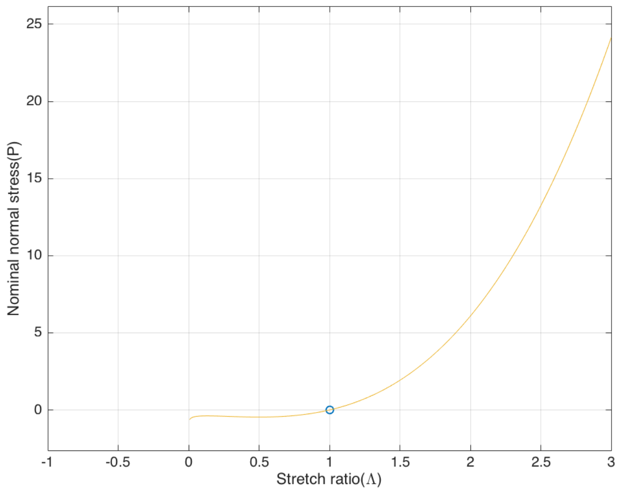


Figure 2: The modified nominal normal stress P with respect to the stretch ratio

1.7 Implementation of Kirchhoff Saint-Venant material model

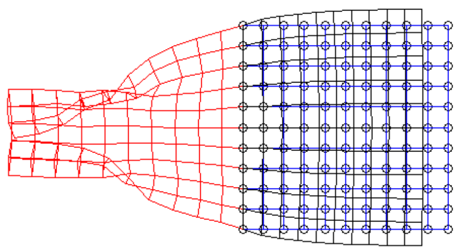
The Matlab function called**KSV** (Kirchhoff Saint-Venant model) has been implemented in the given code. The function computes the following tasks:

- Green-Lagrange deformation tensor (E)
- Stored elastic energy (W)
- Piola-Kirchhoff stress tensor (S)
- Tangent elastic constitutive tensor (C)

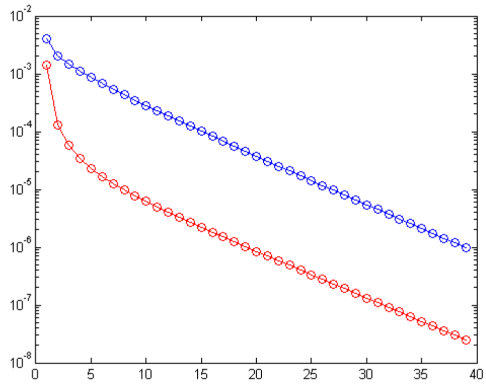
Once the function has been implemented, the consistency test has been performed with the **Check derivatives.m** function. In order to check the problem a new material was created and linked with the **KSV** function. The new material has the following properties:

- $\mu = 0.9$
- $\lambda = 150$
- $potential = 2$

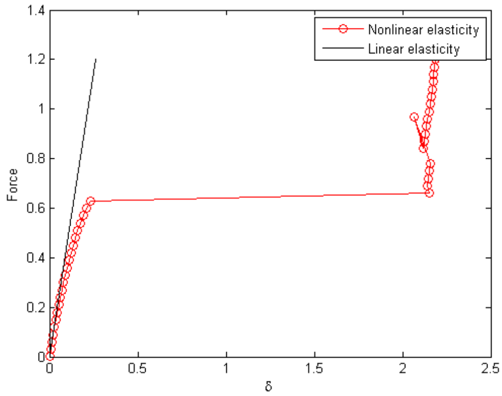
The example run was *Upsetting of a block, dead load* and the consistency test gave us an relative error of 0.0098 %. Figure 3a shows the deformation shape for a linear and non-linear case. It can be noticed how the non-linear response gets unstable. Figure 3b shows the relative error (logaritmic scale) in term of number of iterations and it can be point out how the non-linear model converge. From Figure 3c the non-linear behaviour is shown: it can be appreciated how the force is linear for a small deformation and than, after a almost a constant response, it increases drastically.



(a) Deformation mesh.



(b) Log(relative error) vs iterations.



(c) Force vs deformation diagram.

Figure 3: Implementation of KSV model

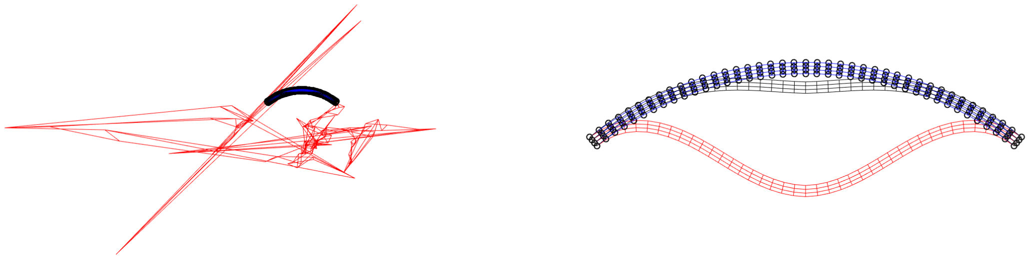
2 Implementation of line-search

Implement a line-search algorithm to be used in combination with Newton's method.For this, we resort to Matlab's function **fminbd**, which performs 1D nonlinear minimization with bounds.Besides, we define a function Ener_1D that evaluates the energy energy along the line that passes through x in the direction s(the descent search direction).

Firstly we summarize the algorithm of Newton's method with line search proceeds as:

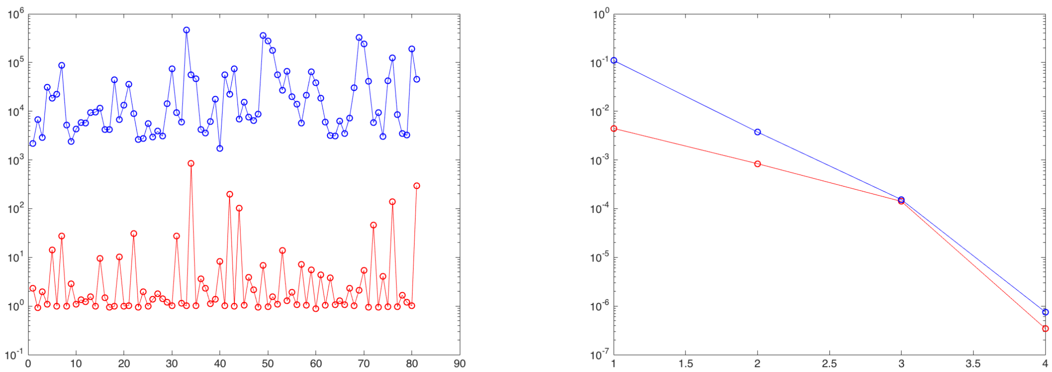
- Solve $\mathbf{J}(\mathbf{x^k})\triangle\mathbf{x^k} = -\mathbf{r}(\mathbf{x^k})$
- Consider an energy-descent search direction, $\mathbf{s^k} = \triangle\mathbf{x^k}$ if $\mathbf{r}(\mathbf{x^k})^\mathbf{T}\triangle\mathbf{x^k} \leq \mathbf{0}$, and $\mathbf{s^k} = -\triangle\mathbf{x^k}$ if $\mathbf{r}(\mathbf{x^k})^\mathbf{T}\triangle\mathbf{x^k} > \mathbf{0}$
- Solve the 1D minimization problem $min_\alpha\Pi(\mathbf{x^k} + \alpha s^k)$
- Update $\mathbf{x^{k+1}} = \mathbf{x^k} + \alpha\mathbf{s^k}$

The code with line-search method is verified with example 4(arch, dead load at center of the arch). The results using normal Newton's method and normal Newton's method with line search are shown in the figure 4.It is obvious that we are unable to obtain a the deformed



(a) Without line-search (b) With line-search

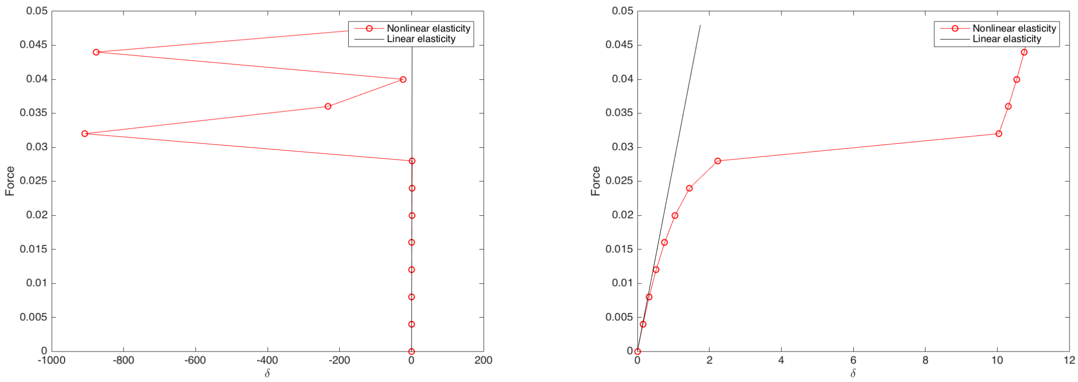
Figure 4: Arch, dead load at center of the arch



(a) Without line-search (b) With line-search

Figure 5: The number of iteration for convergence

arch with normal Newton's method.It is thus advisable to solve the same problem by means of the Newton's method with line-search previously.For the same tolerance, with line-search, the solution is converged at 4 steps while that for Newton's method is 80 steps.Moreover, if the deformed shape,figure 4b is studied, it can be seen how buckling phenomena appears yielding a stable equilibrium situation in which all the eigenvalues are positive.Furthermore, it is also interesting to stress how the fore-displacement curve changes due to buckling,figure 6.



(a) Without line-search

(b) With line-search

Figure 6: Force vs. displacement

3 Implementation of a material model

The code you are given implements a plane-strain finite element method for finite deformation elasticity. A compressible Neo-Hookean material is already in place(modeling a slightly porous rubber for instance),whose strain energy density(or hyper-elastic potential) is

$$W(\mathbf{C}) = \frac{1}{2}(lnJ)^2 - \mu_0 lnJ + \frac{1}{2}\mu_0(trace\mathbf{C} - 3)$$

This constitutive model is isotropic.Note that,since we are considering plane strain,we can use a 2×2 right Cauchy-Green deformation tensor and replace trace**C**-3 by trace**C**-2 in the above equation.

We want to consider now an anisotropic material,more specifically,a transversely isotropic material.We consider a material constitutive law for a rubber reinforce by fibers, all alighed in the same direction in such a way that perpendicular to the fibers,the material remains isotropic.The orientation of the fibers is given in the reference configuration by a unit vector N^{fib}.Such a model depends on the principal invariants of **C**,and additionally by the fourth invariant

$$I_4(\mathbf{C}) = \mathbf{N^{fib}} \cdot \mathbf{C} \cdot \mathbf{N^{fib}} = C_{IJ} N_I^{fib} N_J^{fib}$$

More specifically,

$$W(\mathbf{C}) = \frac{1}{2}\mu(trace\mathbf{C} - 3) - \mu_0 lnJ + \kappa\mathcal{G}(J) + c_0\{exp[c_1(\sqrt{I_4(\mathbf{C})})^4] - 1\}$$

where μ_0,κ,and c_1 are material parameters, and $\mathcal{G}(J)$ provide the volumetric response of the material.We consider

$$\mathcal{G}(J) = \frac{1}{4}(J^2 - 1 - 2lnJ)$$

The last term in the strain energy function specifies the contribution to the deformation energy function specifies the contribution to the deformation energy of the fibers, and as typical in biological fibers, with this model these become stiffer the more deformed they are.

3.1 Implementation of anisotropic material model

The new material implemented in the **preprocessing** Matlab function has the properties as is shown in table 1. The Matlab code implementation is shown in the annex.

Table 1: New material properties

μ	1
c_0	80
c_1	5
κ	100
θ	$[0, \pi/2, \pi/4, \pi/6]$
N_{fib}	$[cos(\theta); sin(\theta)]$

3.2 Implementation example=0

In order to check the correctness of the code, we run *Check_Derivatives.m* with material=2 and luckily everything goes well,meaning no warning shown in the command window.

Figure 7 shows how Newton's method for anisotropic model (for $\theta = \pi/2$) converges quadratically in four iteration: from the first iteration to the third iteration, the convergence analysis shows a linear behaviour, which than changes to quadratic, with a order's relative error of 10^{-10}.

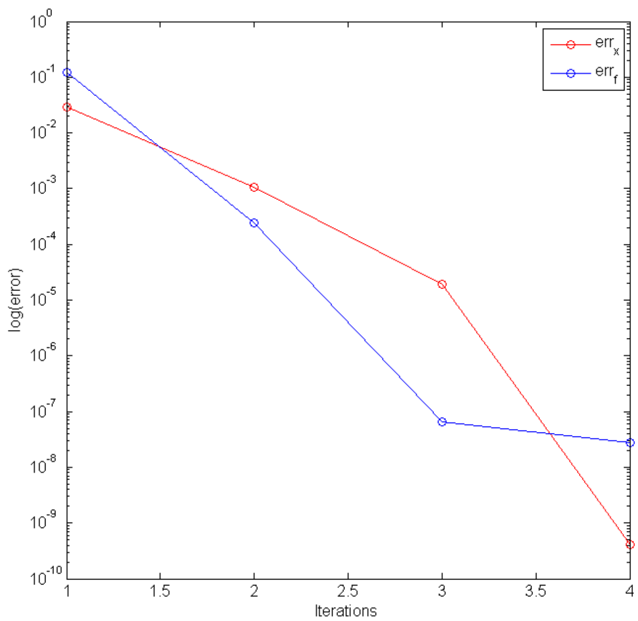


Figure 7: Error Plot showing Quadratic Convergence

3.3 Implementation example=0

Here,we implement example with a dead load applied on an elastic block in tension,and with a few representative orientations of the fibers,$\theta = 0 \quad \theta = pi/6 \quad \theta = pi/4 \quad \theta = pi/2$. The general goal to to check the strength of a material changing with the fiber orientation.For example,in textile industry, the strength of non-woven materials or other fiber-reinforced composite materials made of fibers is usually affected by the orientation.So here this simulation help develop the relationship between strength corresponding to orientation.

For each case, Figure (a) shows the comparison between the original mesh and the mesh after the non-linear response. Figure (b) shows the relation between force and displacement and it can be appreciated how for a small deformation, the slope for the non-linear case coincides with the linear one. As the deformation increases, the non-linear case has changes in slope depending the value of θ.

We can see from figure9a,10a,11a and 12a, with an increasing orientation angle, the deformation becomes large with in anisotropic nonlinear model in red mesh,compared to the black mesh for linear elasticity.The red mesh displays the real fact. Because we know that a fiber can only be applied a tensile load in along its axis direction, when the angle between the load and fiber orientation is small,meaning near 0, almost 100% of fiber strength can be used for the material,thereby the deformation is small.In an extremely case, $\theta = pi/2$, fibers strength takes no contribution to the material strength, so the we obtain the largest deformation of the material, as shown in figure8.The result is consistent with the literature result($http://www.intechopen.com/books/polyester/fibre-reinforced-polyester-composites$),figure 8.

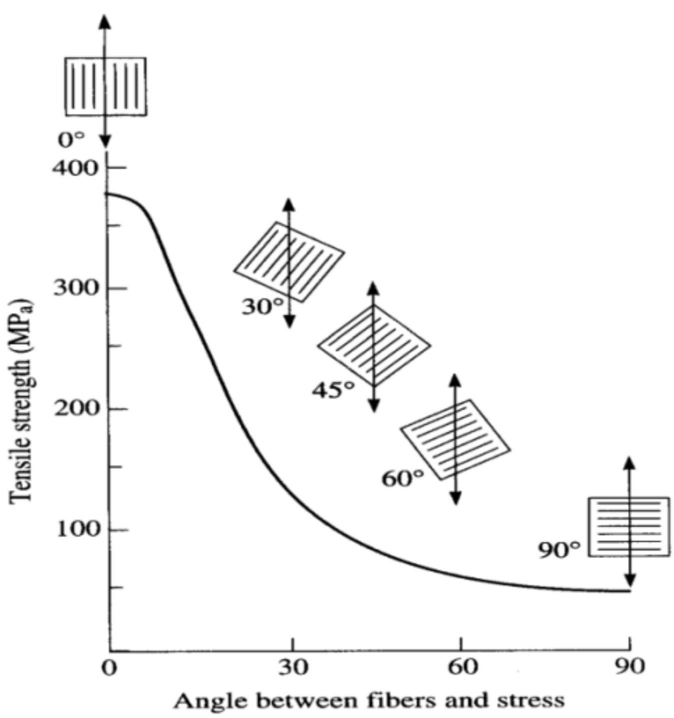
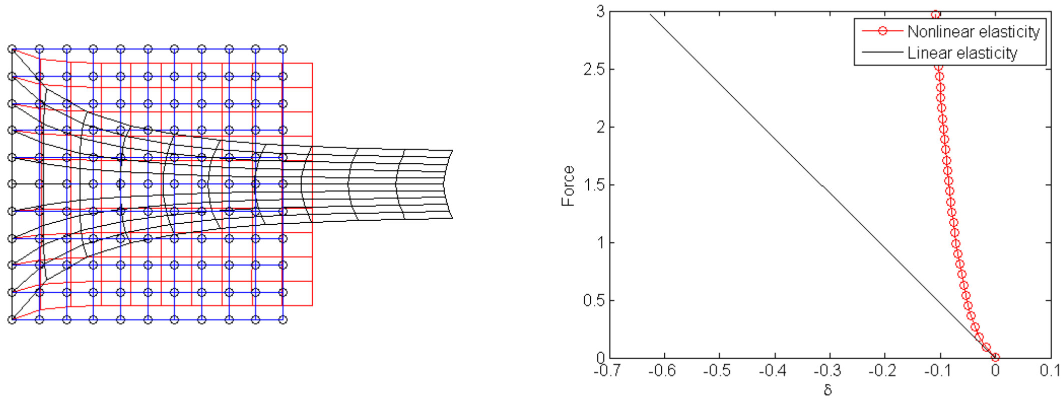
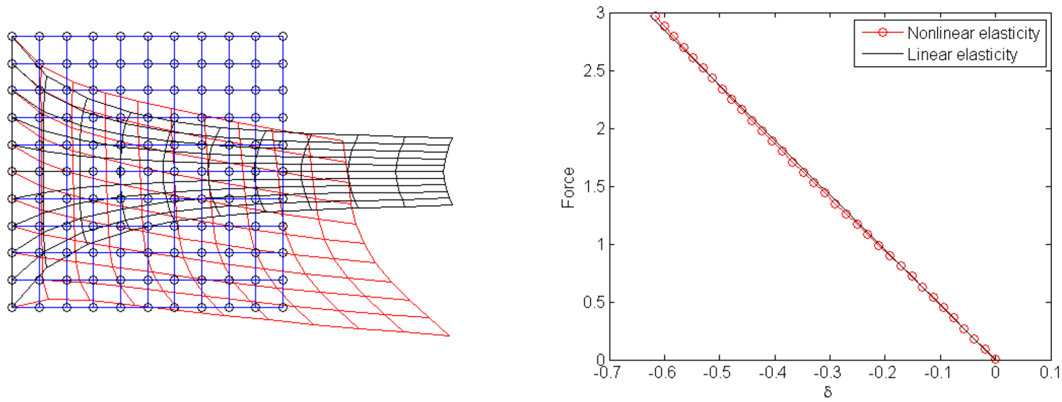


Figure 8: Effect of fiber orientation on the tensile strength



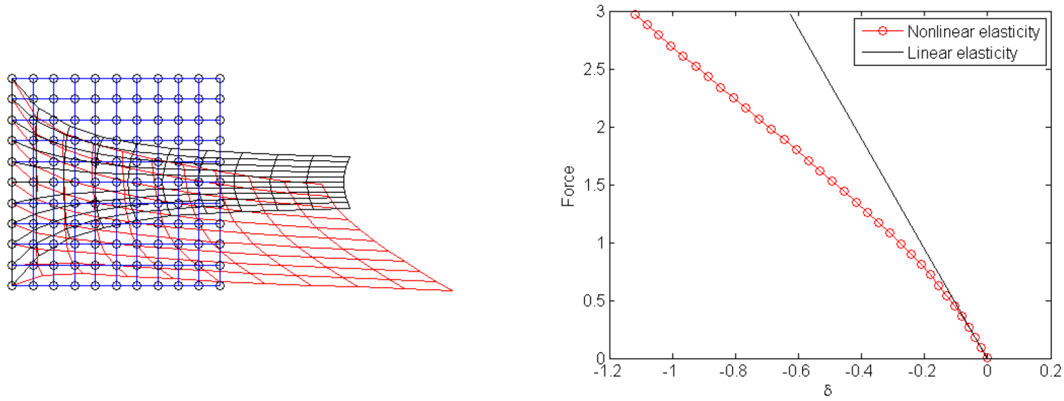
(a) Deformation mesh. (b) Force vs deformation diagram.

Figure 9: Implementation of anisotropic model (theta = 0)



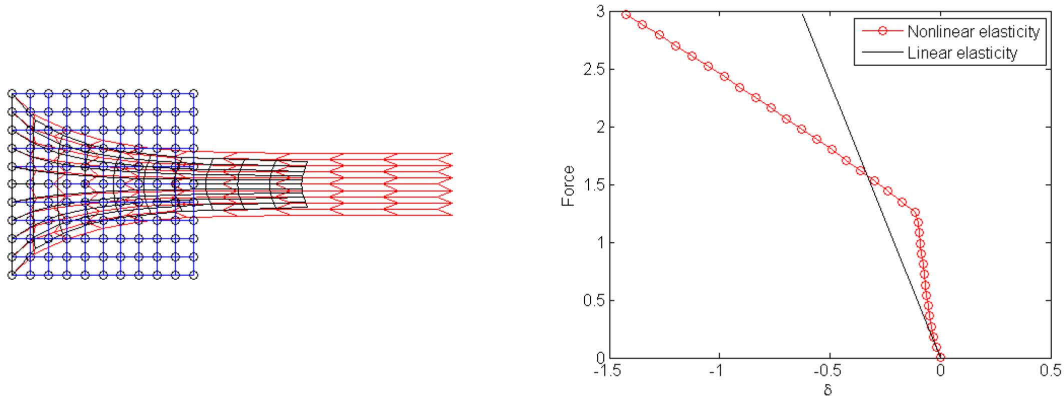
(a) Deformation mesh. (b) Force vs deformation diagram.

Figure 10: Implementation of anisotropic model (theta = $\pi/6$)



(a) Deformation mesh. (b) Force vs deformation diagram.

Figure 11: Implementation of anisotropic model (theta = $\pi/4$)



(a) Deformation mesh. (b) Force vs deformation diagram.

Figure 12: Implementation of anisotropic model (theta = $\pi/2$)

4 Annex

4.1 Kirchho Saint-Venant model

```
function [W,S,CC]=KSV(C,lambda,mu,icode)
C          =[C(1),C(3);
             C(3),C(2)] ;
% Green-Lagrange deformation tensor
E        = 0.5*( C - eye(2) ) ;
% Stored elastic energy
W        = lambda/2*(trace(E))^2 + mu*trace(E*E) ;
% Piola-Kirchhoff stress tensor
t_S  = lambda*trace(E)*eye(2) + 2*mu*E ;

S(1)     = t_S(1,1);
S(2)     = t_S(2,2);
S(3)     = t_S(1,2);
% Tangent elastic constitutive tensor in two dimensions
CC(1,1)  = lambda + 2*mu ;
CC(1,2)  = lambda ;
CC(1,3)  = 0 ;
CC(2,1)  = CC(1,2);
CC(2,2)  = lambda + 2*mu ;
CC(2,3)  = 0 ;
CC(3,1)  = CC(1,3);
CC(3,3)  = 2*mu ;
CC(3,2)  = CC(2,3);
```

4.2 Line search function

```
function [x_short, t]= LineSearch(x_short,p,Ener,dir_der,options)

switch options.type_LS
    case 1, % simple backtracking

        iter = 0;
        alfa=options.alfa;
        t=1;
        [Ener_t] = Ener_short(x_short+t*p,1);
        while ((iter<=options.n_iter_max_LS) && ...
                (Ener_t > Ener + alfa*t*dir_der))
            iter =iter+1;
            t=t*options.beta;
            [Ener_t] = Ener_short(x_short+t*p,1);
        end
        x_short=x_short+t*p;
        if (iter == options.n_iter_max_LS)
            disp('Warning, LS did not succeed!')
        end
    case 2, %MATLAB function
        t=1;
        opts=optimset('TolX',options.TolX,'MaxIter',options.n_iter_max_LS);
        t = fminbnd(@(t) Ener_1D(t,x_short,p),0,100,opts);
        x_short=x_short+t*p;

    otherwise,
```

```
        error('This option does not exist');
end
```

4.3 Transversely isotropic material

```
function [W,S,CC] = anisotropic(C,kappa,mu,c0,c1,N_fib,icode)
syms C11;
syms C12;
syms C22;
C        =[C(1,1), C(2,2),2* C(1,2)];
[W, S,CC] = deri( N_fib,kappa,mu,c0,c1,icode);

W = subs(W,{C11,C22,C12},{C(1),C(2),C(3)})
S = subs(S,{C11,C22,C12},{C(1),C(2),C(3)});
CC = subs(CC,{C11,C22,C12},{C(1),C(2),C(3)});
W = double(W);
S = double(S);
CC = double(CC);
end

function [W, S,CC] = deri( N_fib,kappa,mu,c0,c1,icode)

syms C11;
syms C12;
syms C22;
 Cx        =[C11,C12;
            C12,C22] ;
%fourth invariant
I_4 = (N_fib' * Cx )* N_fib;
J = (C11*C22-C12*C12);
JJ=sqrt(J);
G = 1/4*(J - 1 - 2*log(JJ));

W = 1/2*mu*((C11+C22)-2) - mu * log(JJ) + kappa * ...
   G + c0 * (exp(c1*(sqrt(I_4)-1)^4)-1);

   S11 = 2*diff(W,C11);
   S12 = diff(W,C12);
   S22 = 2*diff(W,C22);

   S = [S11, S22,S12];


   CC1111 = 2*diff(S11,C11);
   CC1112 = diff(S11,C12);
   CC1122 = 2*diff(S11,C22);
   CC1211 = 2*diff(S12,C11);
   CC1212 = diff(S12,C12);
   CC1222 = 2*diff(S12,C22);
   CC2211 = 2*diff(S22,C11);
   CC2212 = diff(S22,C12);
   CC2222 = 2*diff(S22,C22);

   CC = [CC1111,CC1122,CC1112; CC2211 , CC2222,CC2212 ;CC1211,CC1222,CC1212
       ];
```

```
end
```

4.4 Implementation of a new material

```
%Model parameters
switch material
       case 1
           mod1.potential = 1; % 1 is Neohookean
           mod1.mu=1;
           mod1.lambda=1000;
       case 2
           mod1.potential = 2; % 2 is Kirchkoff-Saint Venant
           mod1.mu=1;
           mod1.lambda=100;
       case 3
           mod1.potential = 3; % 3 is transversely isotropic model
           mod1.mu=1.;
           mod1.c0=80;
           mod1.c1=5;
           mod1.kappa=100;
           theta=pi*0;
           mod1.N_fib=[cos(theta); sin(theta)];
     otherwise
         error('Material not implemented')
end
```