

---

---

# COMPUTATIONAL SOLID MECHANICS

---

---

## CONTINUUM DAMAGE MECHANICS

BY

JOHN HANNA

*MSc. in Computational Mechanics*  
*Universitat Politècnica de Catalunya*

13<sup>rd</sup> April, 2019

# Contents

<b>1</b>	<b>General Introduction</b>	<b>2</b>
<b>2</b>	<b>Inviscid Model</b>	<b>2</b>
2.1	Introduction and Implementation . . . . .	2
2.2	Results and Discussion . . . . .	4
2.2.1	1st case (Uniaxial tension-compression-tension) . . . . .	4
2.2.2	2nd case (Uniaxial Tension, biaxial compression-tension) . . . . .	5
2.2.3	3rd case (Biaxial tension-compression-tension) . . . . .	7
<b>3</b>	<b>Viscous Model</b>	<b>8</b>
3.1	Introduction . . . . .	8
3.2	Results and Discussion . . . . .	8
<b>4</b>	<b>Conclusion</b>	<b>11</b>
<b>5</b>	<b>Appendix (MATLAB codes)</b>	<b>12</b>
5.1	Drawing the damage surfaces . . . . .	12
5.2	Trial internal variable calculation for all cases . . . . .	13
5.3	Modification done in the main damage code for viscosity . . . . .	14
5.4	Calculation of internal and hardening variables for all cases . . . . .	15

# List of Figures

1	Tension-only surface for 1st loading case . . . . .	3
2	Unsymmetric surface for 1st loading case . . . . .	3
3	q vs r for exponential hardening/softening . . . . .	3
4	$\sigma_1$ vs $\epsilon_1$ for symmetric model . . . . .	4
5	$\sigma_1$ vs $\epsilon_1$ for unsymmetric model . . . . .	4
6	$\sigma_1$ vs $\epsilon_1$ for Tension only model . . . . .	5
7	Damage vs time for the three models . . . . .	5
8	$\sigma_1$ vs $\epsilon_1$ for Tension only model . . . . .	6
9	$\sigma_2$ vs $\epsilon_2$ for Tension only model . . . . .	6
10	$\sigma_1$ vs $\epsilon_1$ for unsymmetric model . . . . .	6
11	$\sigma_2$ vs $\epsilon_2$ for unsymmetric model . . . . .	7
12	$\sigma_1$ vs $\epsilon_1$ for symmetric model . . . . .	7
13	$\sigma_1$ vs $\epsilon_1$ for unsymmetric model . . . . .	8
14	$\sigma_1$ vs $\epsilon_1$ for Tension model . . . . .	8
15	Damage surface for uniaxial tension . . . . .	9
16	$\sigma_1$ vs $\epsilon_1$ for different strain rates . . . . .	9
17	Evolution of damage for different strain rates . . . . .	9
18	$\sigma_1$ vs $\epsilon_1$ for different viscosities . . . . .	10
19	$\sigma_1$ vs $\epsilon_1$ for different integration parameters . . . . .	10
20	11 component of tangent modulus evolution for different $\alpha$ . . . . .	10
21	11 component of algorithmic tensor evolution for different $\alpha$ . . . . .	11
22	Evolution of tangent and algorithmic tensors with time . . . . .	11

# 1 General Introduction

Continuum damage mechanics is a tool that is used to model materials that suffer from loss of stiffness due to micro pores and cracks that are initiated under certain amount of loading. It is a thermodynamically consistent tool built on the Clausius-Duhem inequality. It is computationally favoured by many researches and used to model different materials due to the present simple algorithms that are widely used.

The main objective of continuum damage mechanics is to provide a continuum mechanics framework to model micro-defects at the macroscopic scale and study their effect on the material properties. This is usually done through introducing a damage parameter ( $d$ ) which takes a value from 0 to 1, 0 being a virgin material, and 1 for a completely damaged material. This parameter takes into account the introduction of voids and micocracks due to the material deformation. An effective stress is defined to account for the damaged area of the material.

Several computational issues appear in attempting to solve a damage problem using the Finite Element Method. The main problem arises from the softening effect that the material pertains in damage. The solution is usually highly dependant on the used mesh because of the inability to capture the strain localization effect. Moreover, using Newton-Raphson algorithm can diverge due to the negative slope of the stress-strain curve.

In this report, inviscid and viscous damage models are implemented using MATLAB. The input is strain which is fed to the software in an indirect way; the loading path is given in the principal stress space, then the strains are calculated using the inverse linear elastic constitutive model. For the rate independent case, Tension-only and Unsymmetric Compression-Tension models are being implemented. Exponential hardening/softening law is implemented along with the linear law. The viscous case is only implemented for the symmetric damage model; the model is then tested using different viscosities and strain rates.

## 2 Inviscid Model

### 2.1 Introduction and Implementation

The first part of the report is about the rate independent damage model. In this case, the viscosity doesn't play a role in the model and stress doesn't depend on the rate by which the material is being deformed. Two damage models are being implemented: Unsymmetric Tension-Compression model and Tension only model. Exponential hardening law is implemented and tested through plotting the hardening parameter ( $q$ ) versus the internal variable ( $r$ ) for the hardening and softening case in figure 3.

To implement these models in the provided code, two things need to be added: the model geometry in the principal stress space and the calculation of the internal variable. To draw the geometry, polar coordinates are used to calculate the principal stresses. The radius is been calculated as the damage surface function divided by the stress norm using  $\theta$  stress vector. Loop is applied over  $\theta = 0$  to  $360^\circ$  to draw the geometry. The principal stresses are then calculated as  $\sigma_1 = r\cos(\theta)$ ,  $\sigma_2 = r\sin(\theta)$ . The internal variable is calculated for both models. For the tension-only model, the Macaulay is used for the calculation. For the nonsymmetric model, the compression-tension ratio and a variable defining the tension-compression state are being used for the calculation.

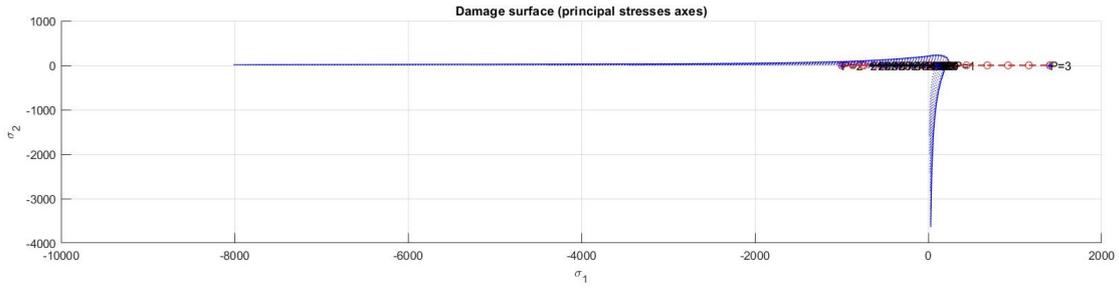


Figure 1: Tension-only surface for 1st loading case

The tension-only model surface is shown above. As expected, the surface has a certain limit in tension beyond which damage occurs, while the surface goes to infinity in compression meaning that the material doesn't suffer damage in compressive loads. In the above graph, the surface is getting smaller since softening is applied to the material.

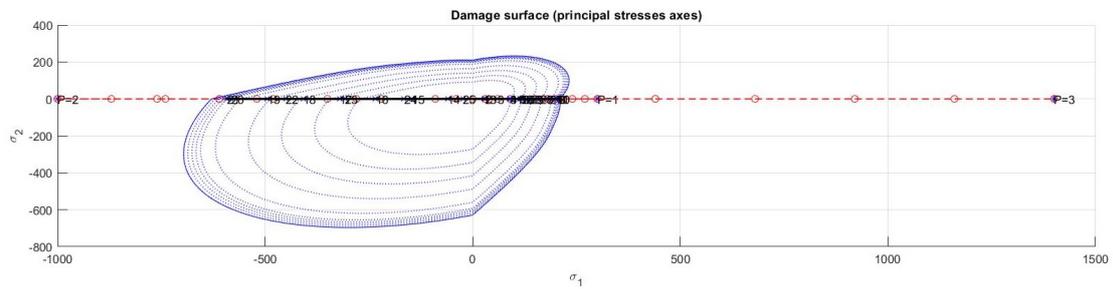


Figure 2: Unsymmetric surface for 1st loading case

The unsymmetric tension-compression model surface is shown above in the principal stresses space. The ratio used is 3 making the stress limit beyond which damage starts is 3 times larger in compression than that of tension. In the loading case, the surface is getting smaller because of the softening effect. In figure 1 and 2, the stress path is either inside the elastic zone or on the damage surface, when the damage surface changes due to softening the path moves with the surface; that happens due to the persistency condition.

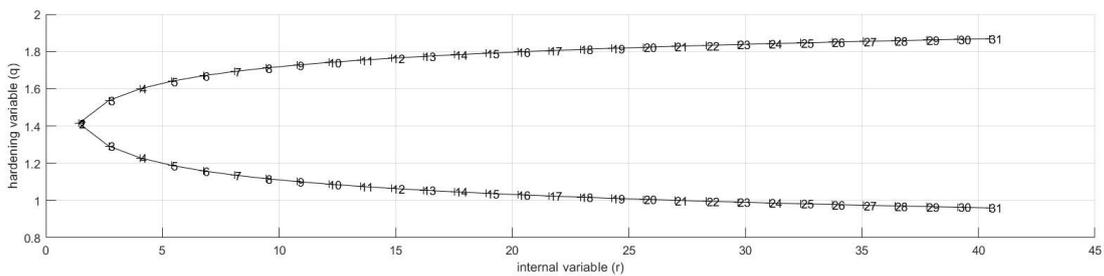


Figure 3: q vs r for exponential hardening/softening

Figure 3 shows the hardening variable versus the internal variable for exponential hardening/softening. At the initial point, the internal variable is equal to the hardening variable. Hardening starts with a certain slope  $H$ , then the slope decreases exponentially reaching almost zero where  $q$  has a limiting value. The graph was plotted for an extensive tension case to show how the curves have a limiting value.

## 2.2 Results and Discussion

### 2.2.1 1st case (Uniaxial tension-compression-tension)

The first case is a purely uniaxial one in which tension is first applied till damage starts and softening takes place. Afterwards, the material is compressed till damage and finally, tension is applied again. The values are given to the code in the principal stress space with values:

$$\begin{aligned} \Delta\sigma_1^{(1)} &= 300, & \Delta\sigma_2^{(1)} &= 0 \\ \Delta\sigma_1^{(2)} &= -1300, & \Delta\sigma_2^{(2)} &= 0 \\ \Delta\sigma_1^{(3)} &= 2400, & \Delta\sigma_2^{(3)} &= 0 \end{aligned}$$

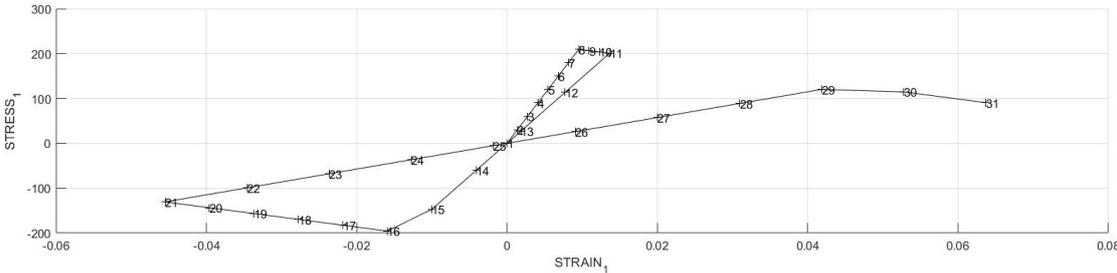


Figure 4:  $\sigma_1$  vs  $\epsilon_1$  for symmetric model

In the symmetric model, the first load path starts with a linear elastic part till damage starts at the stress limit followed by material softening. The second path is compression; it should be noted that this path has different slope due to the occurred damage. Moreover, the path passes by the zero which should not be confused with plasticity where there is plastic strain. The last path is tension again where damage takes place at a lower stress level due to the softening effect.

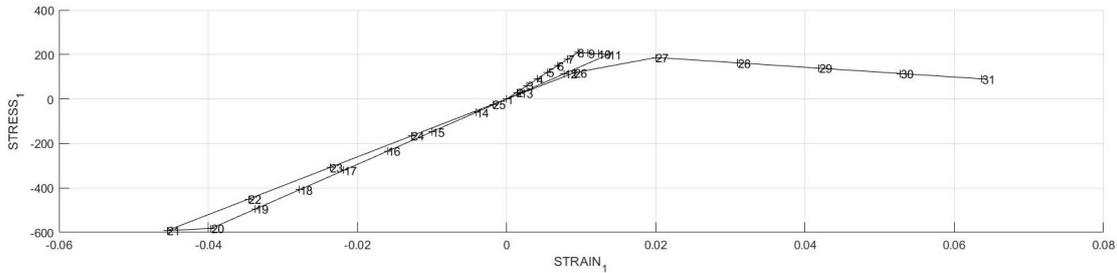


Figure 5:  $\sigma_1$  vs  $\epsilon_1$  for unsymmetric model

As in the symmetric case, the first path has the same exact behavior in the unsymmetric case. However, most of the second path is purely elastic, and damage starts at a higher compressive stress than tensile (about 3 times) that's because of the unsymmetry of the damage surface. Since less damage takes place in this model (less softening), damage starts in the third path at a higher value than it was in the symmetric model.

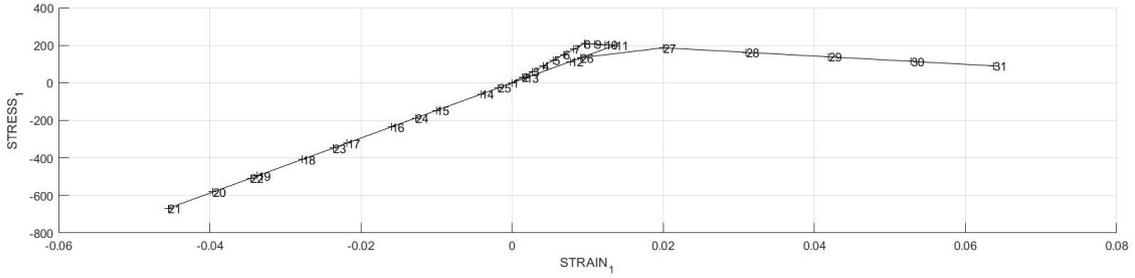


Figure 6:  $\sigma_1$  vs  $\epsilon_1$  for Tension only model

In the tension-only model, the first stress path is the same as in the symmetric and unsymmetric case. However, the second path is purely elastic since damage occurs only in tension and the damage surface goes to infinity in compression. The third stress path has the same behaviour except that damage starts again at a higher stress value than in the other two models since less softening takes place. In general, the overall damage is the same at the end since the softening effect plays a role in increasing the strain needed to start damage again making all the models have the same damage at the end of the 3 stress paths. More details are shown in the next figure.

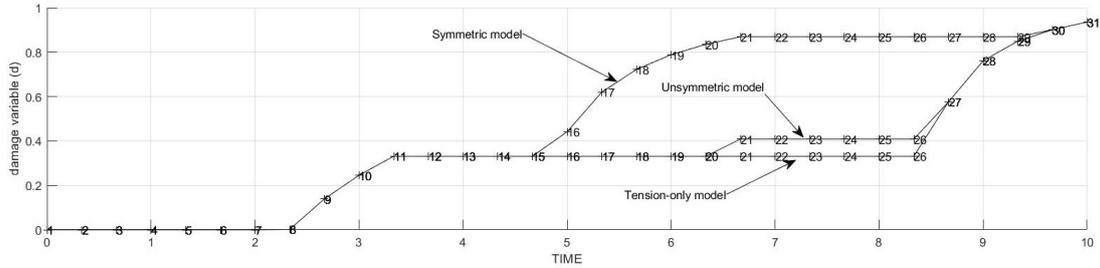


Figure 7: Damage vs time for the three models

The above graph shows the damage parameter vs time for the three models. It should be noted that time doesn't play a role in the inviscid model but it represents the evolution of the loading.

All the models have the same first two parts of the plot which represents the first stress path; the damage variable is constant and equals to zero in the linear elastic part, followed by an increase after the stress exceeded the damage stress limit. The second part, representing the second stress path is different in all the models. It starts by constant damage representing the elastic unloading and first part of the elastic compression which is the same in all the models. Damage in the symmetric models starts early with high evolution, followed by the unsymmetric model with a slight damage evolution, while, no damage occurs in the tension-only model. The third stress path starts with constant damage in all the models; followed by damage which starts the earliest in the tension-only model, followed by the unsymmetric model, and finally the symmetric model, all of them reaching the same value of the damage parameter.

### 2.2.2 2nd case (Uniaxial Tension, biaxial compression-tension)

The 2<sup>nd</sup> case starts with uniaxial loading, followed by biaxial compression, and finally biaxial tension is applied.

$$\begin{aligned} \Delta\sigma_1^{(1)} &= 400, & \Delta\sigma_2^{(1)} &= 0 \\ \Delta\sigma_1^{(2)} &= -1800, & \Delta\sigma_2^{(2)} &= -1800 \end{aligned}$$

$$\Delta\sigma_1^{(3)} = 3100, \quad \Delta\sigma_2^{(3)} = 3100$$

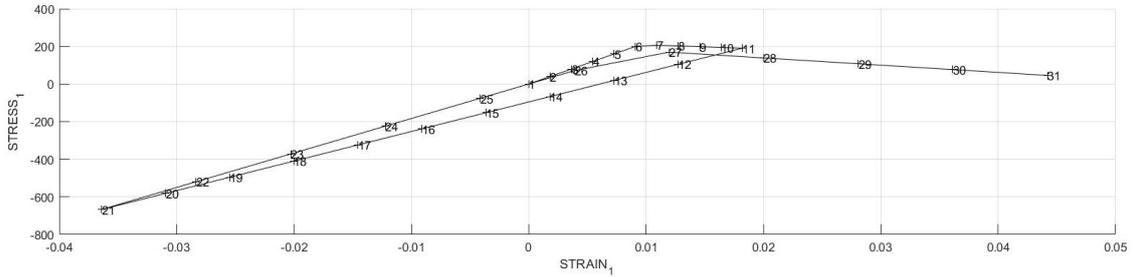


Figure 8:  $\sigma_1$  vs  $\epsilon_1$  for Tension only model

For the stress-strain curve in the first direction, the material is stretched and damage takes place. Afterwards, compression starts making the material follows a line not passing through the zero due to the poisson's effect. No damage occurs in compression. In the final tension path, the material passes the zero since the stress state is the same in both directions. It should be noted that damage started earlier due to the softening effect that the material underwent.

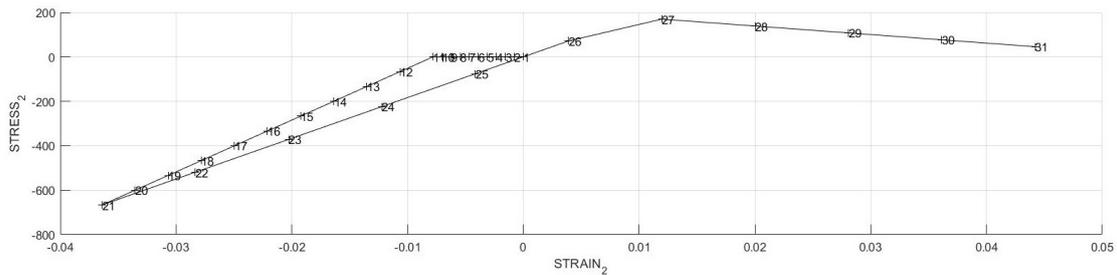


Figure 9:  $\sigma_2$  vs  $\epsilon_2$  for Tension only model

In the second stress direction, the stress is initially zero since the material undergoes uniaxial tension, however, the material starts straining due to Poisson's effect. No damage takes place in the compression part, however, damage occurs at the biaxial tension where softening takes place.

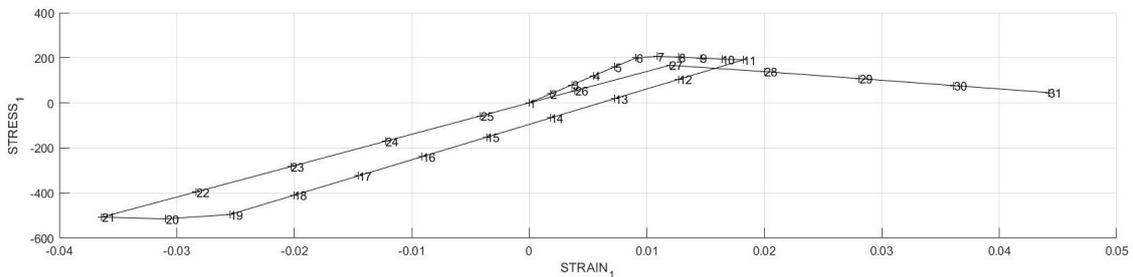


Figure 10:  $\sigma_1$  vs  $\epsilon_1$  for unsymmetric model

For the unsymmetric model, uniaxial tension causes the stress to increase in the first direction till damage occurs. Biaxial compression is then followed where the material don't pass through the zero strain due to the Poisson's effect. It should be noted that damage occurs after a higher stress value in compression since the model is unsymmetric. Biaxial tension is then followed and damage takes place earlier due to the softening effect.

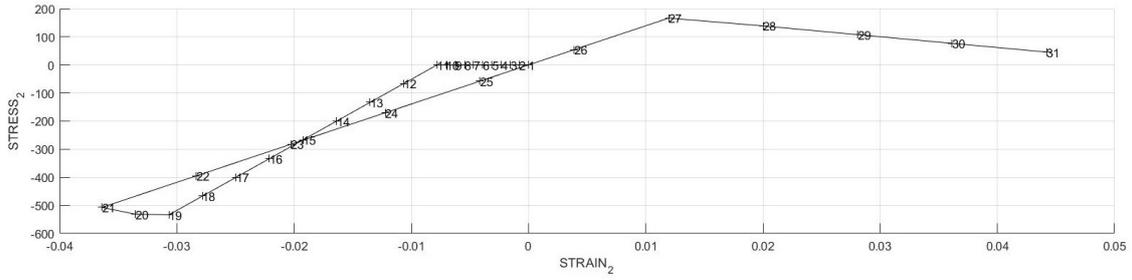


Figure 11:  $\sigma_2$  vs  $\epsilon_2$  for unsymmetric model

For the other stress direction, strain occurs with zero stress due to the Poisson's effect in the uniaxial step. Biaxial compression is followed till damage takes place, afterwards biaxial tension is applied where the material followed a different line due to the previously occurred damage. In tension, damage starts again at a value lower than 200 due to the softening effect.

### 2.2.3 3rd case (Biaxial tension-compression-tension)

In the 3<sup>rd</sup> case, biaxial tension is applied followed by biaxial compression, and finally biaxial tension is applied. The stress-strain curves are plotted in one direction only since the deformation is the same in both directions.

$$\begin{aligned} \Delta\sigma_1^{(1)} &= 400, & \Delta\sigma_2^{(1)} &= 400 \\ \Delta\sigma_1^{(2)} &= -1800, & \Delta\sigma_2^{(2)} &= -1800 \\ \Delta\sigma_1^{(3)} &= 3100, & \Delta\sigma_2^{(3)} &= 3100 \end{aligned}$$

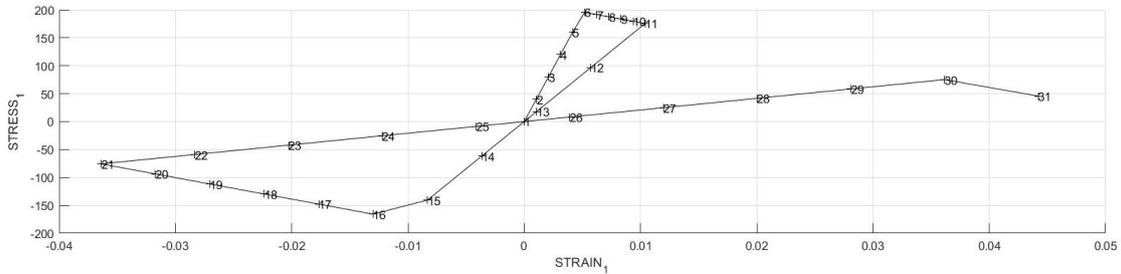


Figure 12:  $\sigma_1$  vs  $\epsilon_1$  for symmetric model

The symmetric model is plotted in the above figure. The softening effect is severe in this case since deformation is done in both directions. The unloading path goes through the zero point since the deformation is the same in both directions and Poisson's effect will not appear. The stress level is the same for tension and compression since the used model is symmetric. In the third stress path, damage starts at a much lower stress value and the slope of the curve is much less indicating that severe damage has taken place.

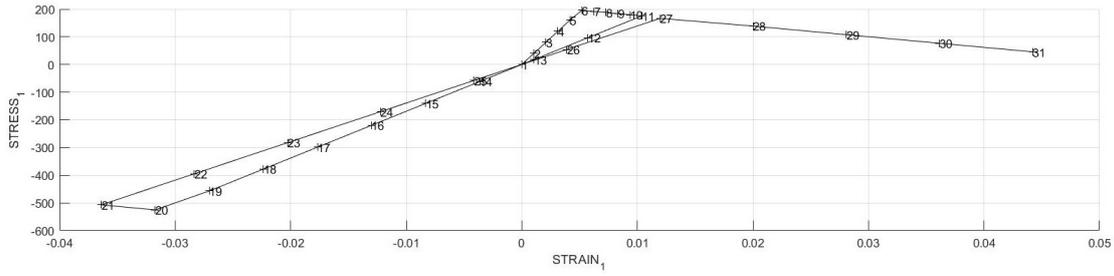


Figure 13:  $\sigma_1$  vs  $\epsilon_1$  for unsymmetric model

The unsymmetric model shows similar effect. As usual, the stress limit is 3 times higher in compression than in tension.

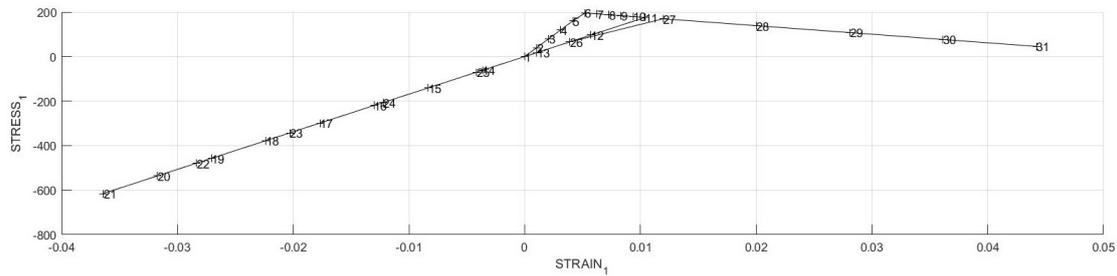


Figure 14:  $\sigma_1$  vs  $\epsilon_1$  for Tension model

The tension-only model doesn't make the material reach damage in compression. Damage in tension less severe since the material is only linear elastic in compression.

### 3 Viscous Model

#### 3.1 Introduction

In the case of the viscous model, the viscosity of the material plays a role in the stress path. Moreover, the rate by which the material is straining also changes the stress path. For that, time will play a role in the governing equations so as the material viscosity. The viscous part is developed only in damage, which means that it has no effect in the linear elastic part. To do the time integration, the famous  $\theta$  method is used, where  $0 \leq \theta \leq 1$ , 0 for the explicit scheme, 1 for the implicit method and 0.5 for Crank-Nicholson method. The last method is usually preferred since it has second order accuracy which is not the case for the other methods.

The rate dependant case is implemented in MATLAB for the symmetric model only. It is implemented by introducing a different trial internal variable that's a combination between the old and new strain norms. The new internal variable is calculated differently being dependant on the viscosity, strain rate and the time integration method. The stress-strain curves for different viscosities, strain rates and integration methods are plotted to test the correctness of the implementation.

#### 3.2 Results and Discussion

Uniaxial tension case is solved using the viscous model in order to see the effect of changing different parameters on the stress behaviour. In general,  $\alpha = 0.5$ ,  $\eta = 0.5$ ,  $total - time = 10$  and one parameter of them is changing at a time while the others are fixed at these values.

$$\Delta\sigma_1^{(1)} = \Delta\sigma_1^{(2)} = \Delta\sigma_1^{(3)} = 300$$

$$\Delta\sigma_2^{(1)} = \Delta\sigma_2^{(2)} = \Delta\sigma_2^{(3)} = 0$$

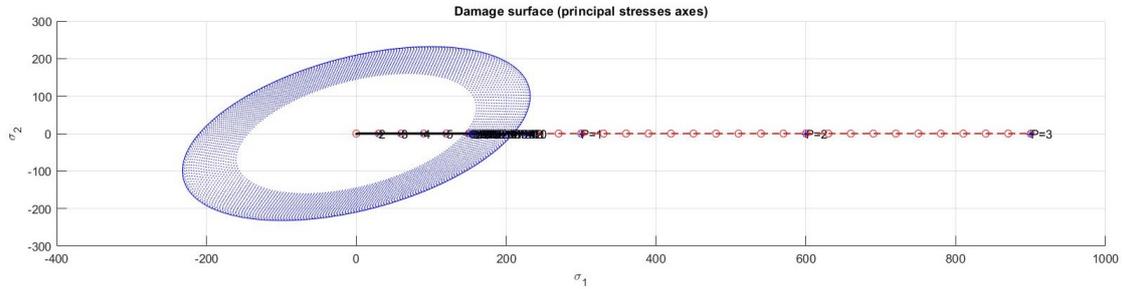


Figure 15: Damage surface for uniaxial tension

The above figure shows the stress path in the principal stress space along with the damage surface for the viscous model. It must be pointed out that the persistency condition is no longer available in the viscous case and the stress path is free not to be on the damage surface on the contrary to the inviscid case.

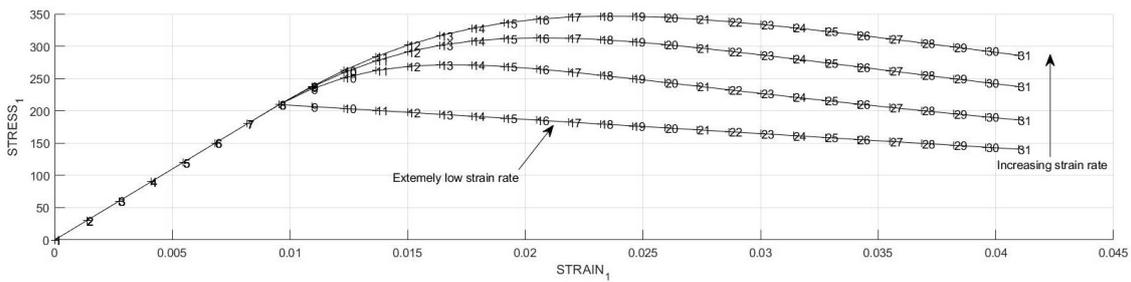


Figure 16:  $\sigma_1$  vs  $\epsilon_1$  for different strain rates

The stress-strain curve for different strain rates shows how the strain rate plays an important role in the stress development and in damage. It should be noted that all the curves coincide in the elastic part since the viscous part is only implemented with damage. Moreover, the stress level increases as the strain rate increases; that means if the material is being deformed fast, a higher level of stress will be achieved. For very low strain rates, the inviscid behaviour is retained.

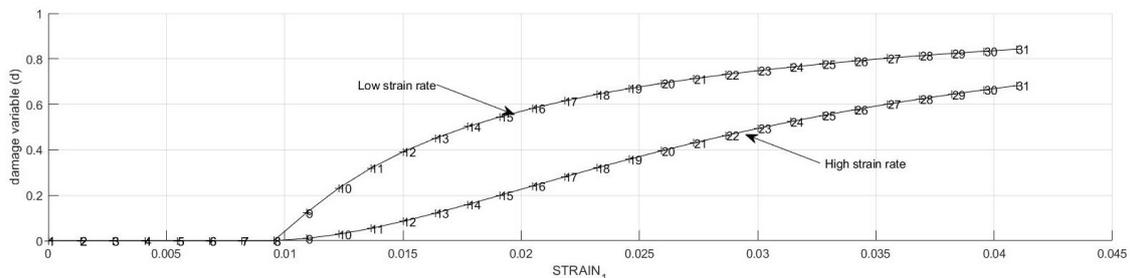


Figure 17: Evolution of damage for different strain rates

The damage parameter will evolve faster in the high strain rate, as expected, but in the overall process low strain rate will produce more damage for the same strain path over longer time.

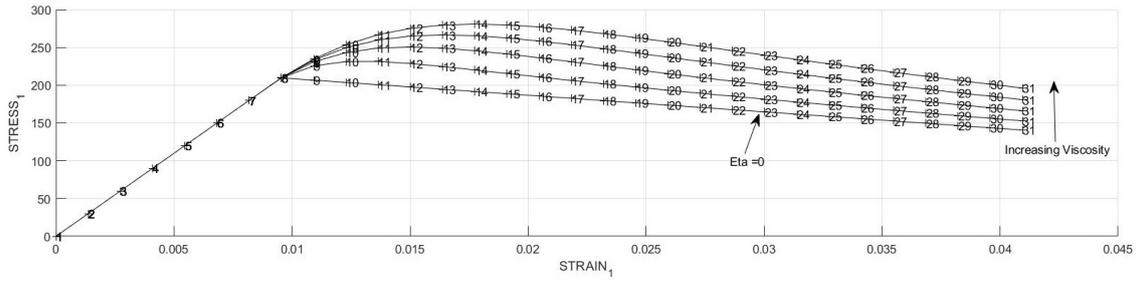


Figure 18:  $\sigma_1$  vs  $\epsilon_1$  for different viscosities

The above figure shows different stress-strain curves for different material viscosities. Higher stress level is obtained as the material viscosity increases. It should be noted that the inviscid model is retained for a zero viscosity and an integration parameter  $\alpha$  of 1.

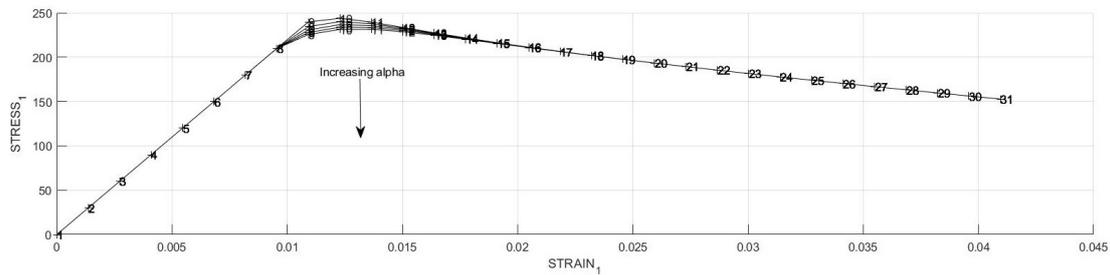


Figure 19:  $\sigma_1$  vs  $\epsilon_1$  for different integration parameters

The above figure shows the effect of changing the integration parameter  $\alpha$  on the stress-strain curve. The parameter affects the calculation of  $r$ -trial in which it's higher as a higher integration parameter is used. Therefore, damage can start earlier using large value of  $\alpha$ . Moreover, the parameter affects the calculation of the internal variable which affects the hardening/softening specially in the beginning.

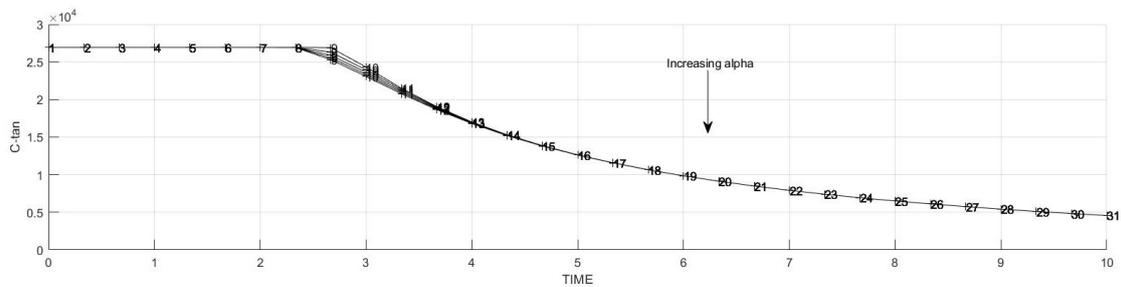


Figure 20: 11 component of tangent modulus evolution for different  $\alpha$

The tangent modulus evolution with time is divided into two parts: elastic part where it is equal to the elasticity tensor and the damage part which is affected by the damage variable. For different integration parameters, the first part of all the curves is the same since there is no damage, however the second part is different; as the integration parameter increases, damage starts earlier and hence, the tangent modulus is reduced faster.

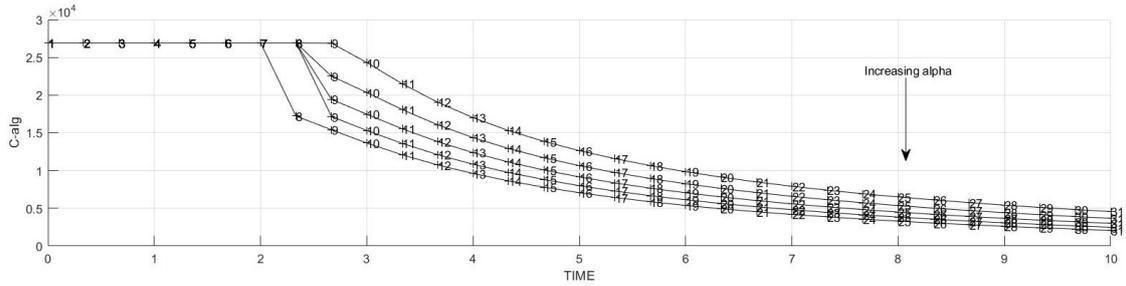


Figure 21: 11 component of algorithmic tensor evolution for different  $\alpha$

The evolution of the algorithmic tensor for different integration parameters is plotted above. It is noteworthy that increasing the value of  $\alpha$  will result in an earlier damage, and thus, faster reduction in the tensor. Moreover, the algorithmic tensor coincides with the tangent modulus for  $\alpha = 0$ .

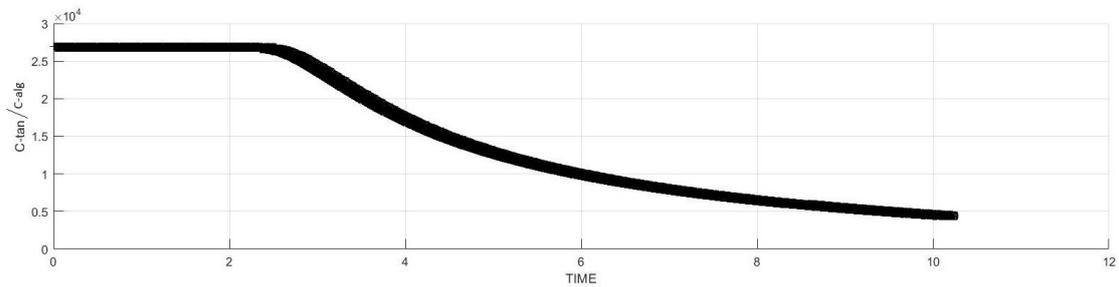


Figure 22: Evolution of tangent and algorithmic tensors with time

The above figure shows the evolution of tangent modulus and the algorithmic tensor, the last is plotted for a very small time step. The figure shows that both tensors coincide as the time step tends to zero, which is a conclusion that we can draw from the algorithmic tensor equation.

## 4 Conclusion

In conclusion, material voids and internal cracks can be modelled macroscopically through damage mechanics. Damage models are easy to implement on a Gauss point level, therefore, it is favoured by many researchers. Several damage models can be implemented: inviscid and viscous models and for both of them symmetric, unsymmetric and tension-only models exist. An issue that needs to be taken care of in finite elements is the softening behaviour associated with damage.

## 5 Appendix (MATLAB codes)

### 5.1 Drawing the damage surfaces

```
% POLAR COORDINATES
if MDtype==1
    tetha=[0:0.01:2*pi];
    %*****
    %* RADIUS
    D=size(tetha); %* Range
    m1=cos(tetha); %*
    m2=sin(tetha); %*
    Contador=D(1,2); %*

    radio = zeros(1,Contador) ;
    s1 = zeros(1,Contador) ;
    s2 = zeros(1,Contador) ;

    for i=1:Contador
        radio(i)= q/sqrt([m1(i) m2(i) 0 nu*(m1(i)+m2(i))]*....
            .....
            ce_inv*[m1(i) m2(i) 0 nu*(m1(i)+m2(i))]' );

        s1(i)=radio(i)*m1(i);
        s2(i)=radio(i)*m2(i);

    end
    hplot =plot(s1,s2,tipo_linea);

elseif MDtype==2
    tetha=[0:0.01:2*pi];
    %*****
    %* RADIUS
    D=size(tetha); %* Range
    m1=cos(tetha); %*
    m2=sin(tetha); %*
    Contador=D(1,2); %*

    radio = zeros(1,Contador) ;
    s1 = zeros(1,Contador) ;
    s2 = zeros(1,Contador) ;

    for i=1:Contador

        radio(i)= q/sqrt([max(m1(i),0) max(m2(i),0) 0....
            .....
            max(nu*(m1(i)+m2(i)),0)]* ce_inv*[m1(i) m2(i) 0 ...
            .....
            nu*(m1(i)+m2(i))]' ); % max function is used as a macaulay bracket
```

```

        s1(i)=radio(i)*m1(i);
        s2(i)=radio(i)*m2(i);

    end
    hplot =plot(s1,s2,tipo_linea);

elseif MDtype==3
    tetha =[0:0.01:2*pi];

    %* RADIUS
    D=size(tetha); %* Range
    m1=cos(tetha); %*
    m2=sin(tetha); %*
    Contador=D(1,2); %*

    radio = zeros(1,Contador) ;
    s1 = zeros(1,Contador) ;
    s2 = zeros(1,Contador) ;

    for i=1:Contador

        set=(max(m1(i),0)+max(m2(i),0)+max(nu*(m1(i)+m2(i)),0))....
        .....
        /(abs(m1(i))+abs(m2(i))+.....
        ....
        abs(nu*(m1(i)+m2(i))))); % variable to define tension-
        % compression state

        radio(i)= q/((set+(1-set)/n)*sqrt([m1(i) m2(i) 0...
        .....
        nu*(m1(i)+m2(i))] * ce_inv * [m1(i) m2(i) 0 nu*(m1(i)+m2(i))] '));

        s1(i)=radio(i)*m1(i);
        s2(i)=radio(i)*m2(i);

    end
    hplot =plot(s1,s2,tipo_linea);

end

```

## 5.2 Trial internal variable calculation for all cases

```

%*****
if (MDtype==1) %* Symmetric

    if viscpr==1
        tau_n= sqrt(eps_n*ce*eps_n'); % strain norm at step n
        tau_n1= sqrt(eps_n1*ce*eps_n1'); % strain norm at step n+1
        rtrial=(1-alp)*tau_n+alp*tau_n1;
    else
        rtrial= sqrt(eps_n1*ce*eps_n1');
    end

```



```

end
% GLOBAL VARIABLES
% *****
% Stress
% -----
m_sigma=[sigma_n1(1)  sigma_n1(3)  0;sigma_n1(3)...
        .....
        sigma_n1(2) 0 ; 0 0  sigma_n1(4)];
sigma_v{i} = m_sigma ;

% VARIABLES TO PLOT (set label on cell array LABELPLOT)
% -----
vartoplot{i}(1) = hvar_n(6) ; % Hardening variable (q)
vartoplot{i}(2) = hvar_n(5) ; % Internal variable (r)
vartoplot{i}(3) = 1-hvar_n(6)/hvar_n(5) ; % Damage variable (d)
vartoplot{i}(4) = C_tan(1,1) ; % tangent modulus
vartoplot{i}(5) = C_alg(1,1) ; % Algorithmic tensor
end
end

```

#### 5.4 Calculation of internal and hardening variables for all cases

```

%*      initializing
r0 = sigma_u/sqrt(E);
zero_q=1.d-6*r0;
inf_q=r0+(r0-zero_q);
Aexp=H*r0/(inf_q-r0); % constant for exponential hard/soft
                        %(for softening zeo_q is used)

%*****
%*****
%*      Damage surface
[rtrial] = Modelos_de_dano1 (MDtype,ce,viscpr,eps_n,eps_n1,n);
%*****

%*****
%*      Ver el Estado de Carga
%*
%*      ----->      fload=0 : elastic unload
%*
%*      ----->      fload=1 : damage (compute algorithmic constitutive tensor)
%*
fload=0;
sig_eff=ce*eps_n1'; % effective stress

if(rtrial > r_n)
    %*      Loading
    if viscpr==1
        r_n1= (eta-delta_t*(1- alp))*r_n/(eta+alp*delta_t)...
        ....
                +delta_t*rtrial/(eta+alp*delta_t);
    else

```

```

        r_n1=rtrial;
    end
    fload=1;
    delta_r=r_n1-r_n;
    if hard_type == 0
        % Linear
        q_n1= q_n+ H*delta_r;

        %%% Limits for hardening and softening
        if H<0
            if (q_n1<zero_q)
                q_n1=zero_q;
            end
        else
            if (q_n1>inf_q)
                q_n1=inf_q;
            end
        end

    else
        % Exponential (for softening zero_q is used)
        q_n1=inf_q -((inf_q-r_n)*exp(Aexp*(1-r_n1/r_n)));
    end

    % damage parameter
    dano_n1 = 1.d0-(q_n1/r_n1);
    % Computing stress and constitutive tensors
    % *****
    sigma_n1 =(1.d0-dano_n1)*ce*eps_n1';
    C_tan = (1-dano_n1)*ce;
    C_alg = C_tan + (alp*delta_t*(H*r_n1-q_n1)/((eta+alp*delta_t)*tau_n1*r_n
    % if (q_n1<zero_q) %%% question here (softening limit do we need a har
    % q_n1=zero_q;
    % end
    else
        %* Elastic load/unload
        fload=0;
        r_n1= r_n;
        q_n1= q_n ;

        dano_n1 = 1.d0-(q_n1/r_n1);
    % Computing stress and constitutive tensors
    % *****
    sigma_n1 =(1.d0-dano_n1)*ce*eps_n1';
    C_tan = (1-dano_n1)*ce;
    C_alg = C_tan;
    end

```