ASSIGNMENT 1

Computational Solid Mechanics

Sumit Maharjan

Background:

The main objective of standard Continuum Damage Mechanics is to propose a continuum-mechanics based framework allowing to characterize, represent and model at the macroscopic scale the effects of distributed defects and their growth on the material behaviour.

Section 1:

(a) The continuum isotropic damage "non-symmetric tension-compression damage" model

This type of damage models are used to simulate concrete, rocks and other granular materials where the tensile and compressive elastic limits are different. A continuum "non-symmetric tension-compression damage" model has been implemented. The code for the implementation is presented in the Annex. The damage surface for the implemented model considering the ratio of the uniaxial elastic limit in compression /tension as 3, has been presented below in the **Figure 1** in principal stresses axes. It can be observed that the elastic limits of the elastic domain are different for tension and compression.
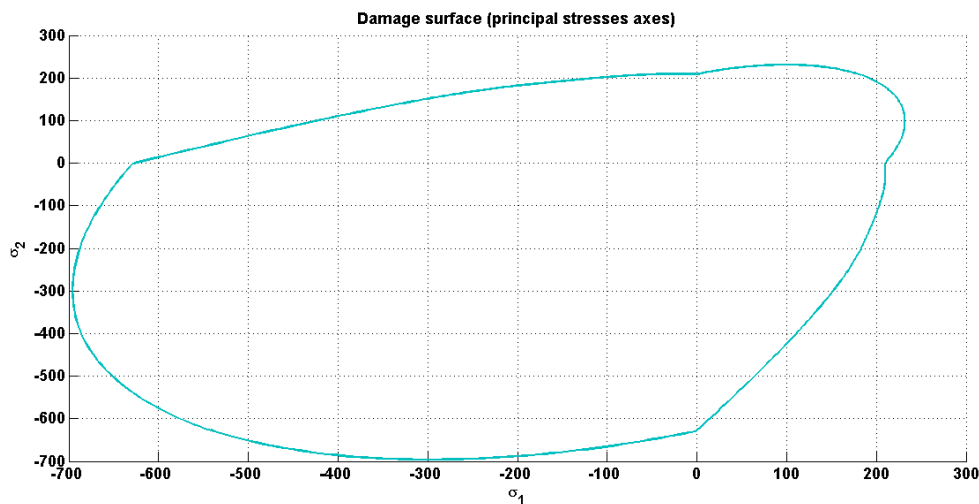

Figure 1: Damage surface for the non-symmetric tension-compression damage model

The "tension-only" damage model

The 'tension-only damage model' model the material that can only fail by tension. The elastic limit due to compression can't be reached in this case. When all the principal stresses are negative elastic limit is reached at infinite. A "tension-only" damage model has been implemented. The code has been described in the Annex. The damage surface for the implemented model is shown below in the **Figure 2**. It can be observed that where all the principal stresses are positive, the damage is associated to only the tensile behaviour.
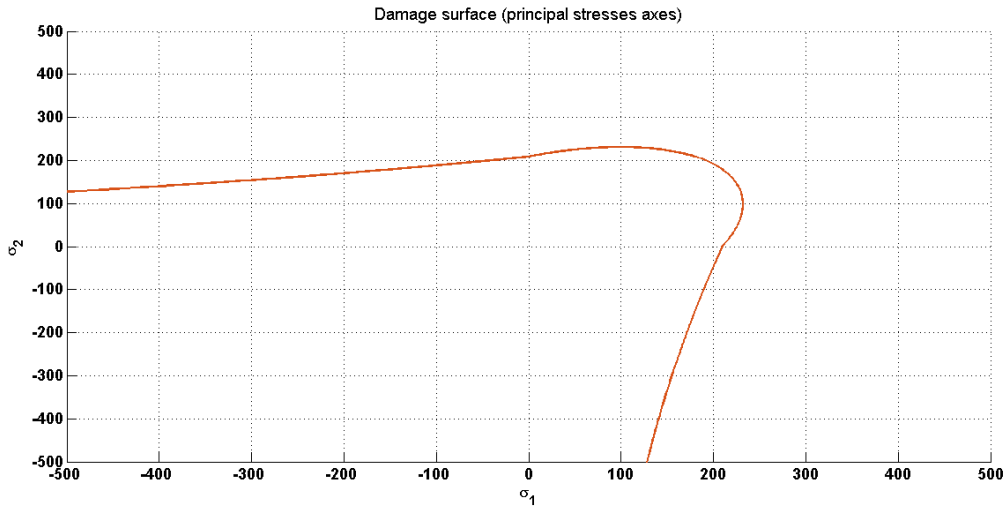
Universitat Politècnica De Catalunya BARCELONATECH

Escola Tècnica Superior d'Enginyers de Camins, Canals i Ports de Barcelona

Centre Internacional de Mètodes Numèrics en Enginyeria

CIMNE

Damage surface (principal stresses axes)

Figure: 2 Damage surface for the implemented Tensile-damage model

On the other hand, previously implemented Symmetric tension compression damage model is effective when the material behave the same both with tension or and compression. The damage surface evolves symmetrically both for tension and compression.

(b) Exponential hardening/softening

An exponential hardening/softening model has been implemented choosing $q_\infty = 10^{-6} r_0$. It can never be negative. For the implemented model, when r→ ∞, q→ q_∞. In the case of hardening the $q_\infty > r_0$, and for the case of softening $q_\infty < r_0$. **The figure 3**, below, shows the variation of q with respect to r for H=8 for hardening and H= -8 for softening. The implemented code has been described in the Annex.
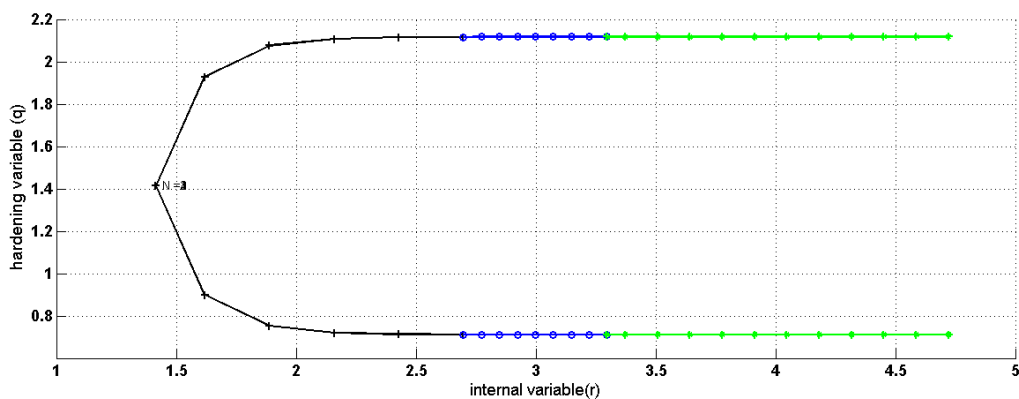

Figure 3: Variation of q vs. r in the Exponential model

(C) **Examine the correctness of the implementation of the models:**

First Case

$(\overline{\Delta\sigma_1}\,(1) = \alpha\,,\overline{\Delta\sigma_2} = 0), (\overline{\Delta\sigma_1}\,(1) = -\beta\,,\overline{\Delta\sigma_2} = 0), (\overline{\Delta\sigma_1}\,(1) = \gamma\,,\overline{\Delta\sigma_2} = 0)$

"Non-symmetric tension-compression damage" model

$(\overline{\Delta\sigma_1}\,(1) = 400\,,\overline{\Delta\sigma_2} = 0), (\overline{\Delta\sigma_1}\,(1) = -1800\,,\overline{\Delta\sigma_2} = 0), (\overline{\Delta\sigma_1}\,(1) = 1400\,,\overline{\Delta\sigma_2} = 0)$

The hardening modulus has been considered to be 0. Initially, till the stress state reaches the elastic domain the effective and the actual stresses are same. After which when the stress path leaves the elastic domain, and this becomes the case of loading ($\dot{d} > 0$). After this when the unloading occurs, the actual stress path tends along a straight line with slope $E^{Sec} = (1 - d)E$ along the secant path. The stress path continues to be along on the same line till the stress state reaches the elastic limit for the compression. After which, when the stress path leaves the elastic limit of compression, again the this become the case where, $\dot{d} > 0\,(Loading)$. Following this the uniaxial elastic tensile loading occurs. The actual stress path again becomes along a straight line with a slope $E^{Sec} = (1 - d_{new})E$, till it reaches the origin. The evolution of the damage surface in the stress space has been shown in the **Figure 4**. **The Figure 5** shows the above explained occurrences.
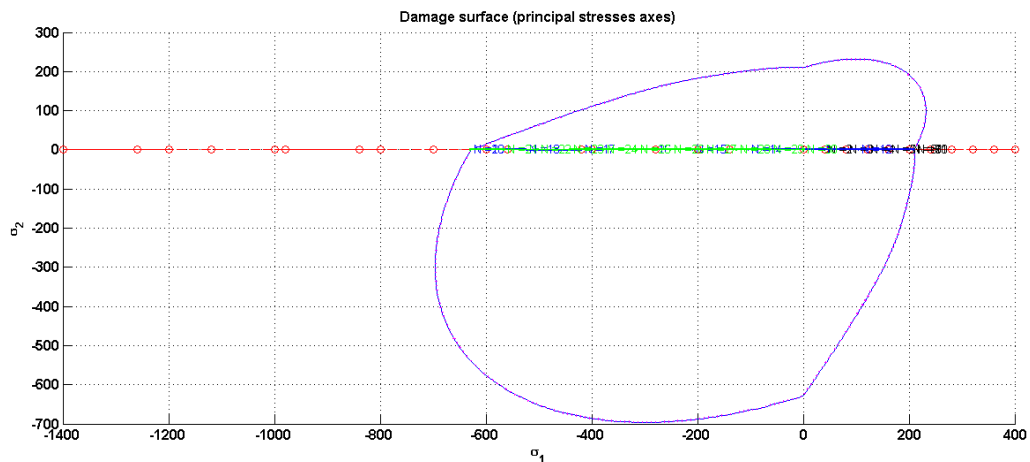

Figure 4:- Path at the stress space for the "Non-symmetric damage model"
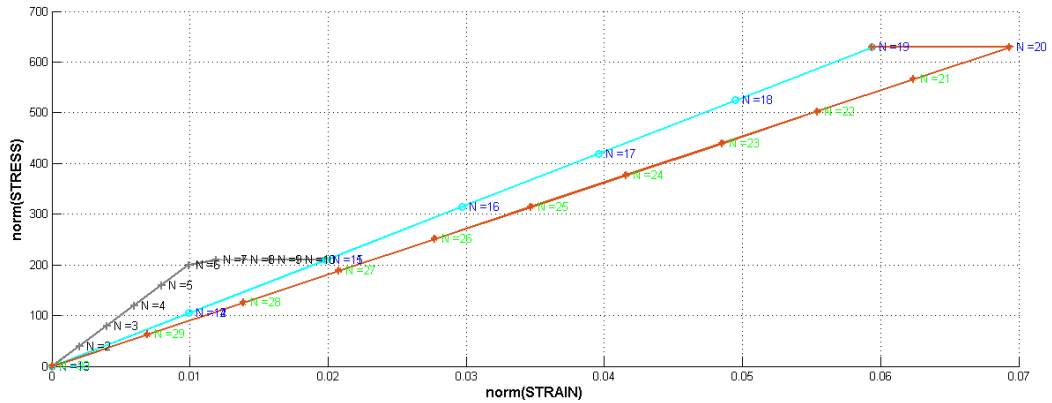
Figure 5:- The stress-strain curve for the uniaxial "Non-symmetric damage model"

The "tension-only" damage model

$(\overline{\Delta\sigma_1}\ (1) = 400\ ,\ \overline{\Delta\sigma_2} = 0),\ (\overline{\Delta\sigma_1}\ (1) = -1800\ ,\ \overline{\Delta\sigma_2} = 0),\ (\overline{\Delta\sigma_1}\ (1) = 1400,\ \overline{\Delta\sigma_2} = 0)$

The hardening modulus has been considered to be 0. At the beginning, till the stress state reaches the elastic domain the effective and the actual stresses are same. Following this when the stress path leaves the elastic domain, it becomes the case of loading. After this when the unloading occurs , the actual stress path tends along a straight line with slope $E^{Sec} = (1 - d)E$ along the secant. The actual stress path continues to be along on the same line. The elastic limit in compression can't be reached. After that when the uniaxial elastic tensile loading occurs, the actual stress path again follows the same straight line. The evolution of the damage surface in the stress space has been shown in the **Figure 6**.
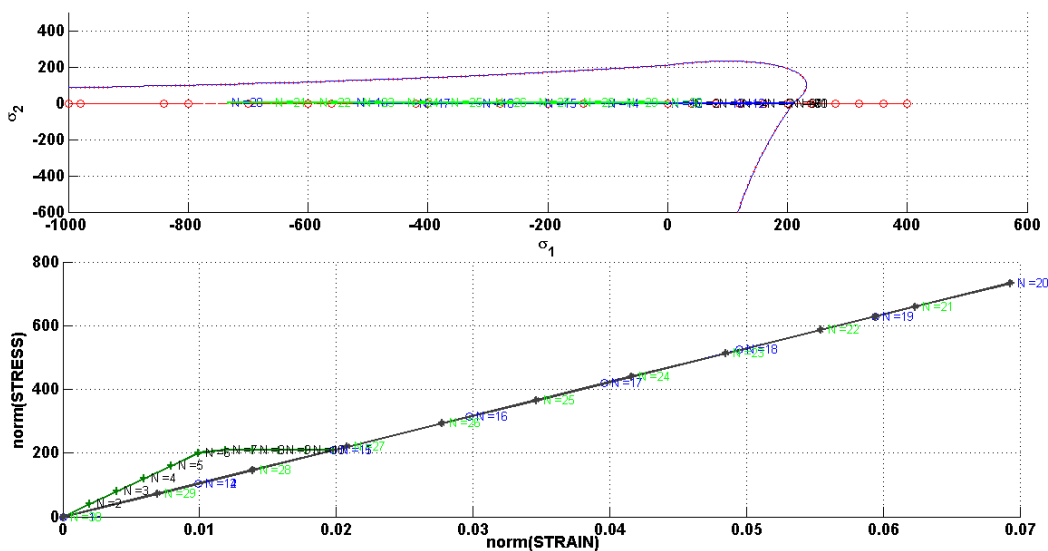

Figure 6:- Path at the stress spacefor the "tension-only" model

Second case

$(\overline{\Delta\sigma_1}\,(1) = \alpha \,, \overline{\Delta\sigma_2} = 0)$, $(\overline{\Delta\sigma_1}\,(1) = -\beta \,, \overline{\Delta\sigma_2} = -\beta)$, $(\overline{\Delta\sigma_1}\,(1) = \gamma \,, \overline{\Delta\sigma_2} =)$

"Non-symmetric tension-compression damage" model

$(\overline{\Delta\sigma_1}\,(1) = 400 \,, \overline{\Delta\sigma_2} = 0)$, $(\overline{\Delta\sigma_1}\,(1) = -2000 \,, \overline{\Delta\sigma_2} = -2000)$, $(\overline{\Delta\sigma_1}\,(1) = 1600 \,, \overline{\Delta\sigma_2} = 1600)$

The hardening modulus has been considered to be -0.05. Initially, up to the elastic domain during the uniaxial elastic loading the effective and the actual stresses are same. After which when the stress path leaves the elastic domain for tension, this become the case where, $\dot{d} > 0$ (*Loading*). After that when biaxial tensile unloading/compressive loading occurs. After which, when the stress path leaves the elastic limit of compression, loading occurs. Following this, the biaxial compressive unloading/tensile loading takes place. The evolution of the damage surface in the stress space has been shown in the **Figure 7** and **Figure 8**.
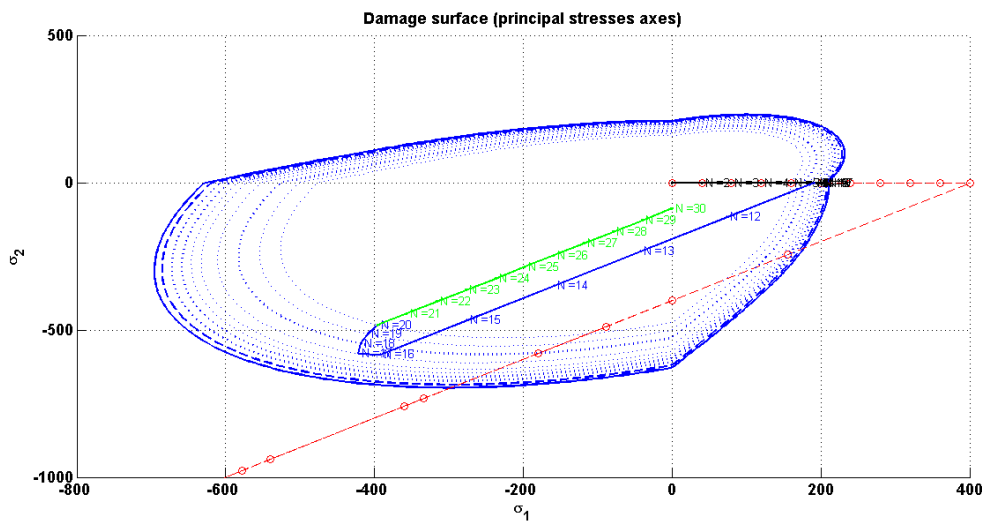

Figure 7:- Path at the stress spacefor the "Non-symmetric tension-compression damage" model
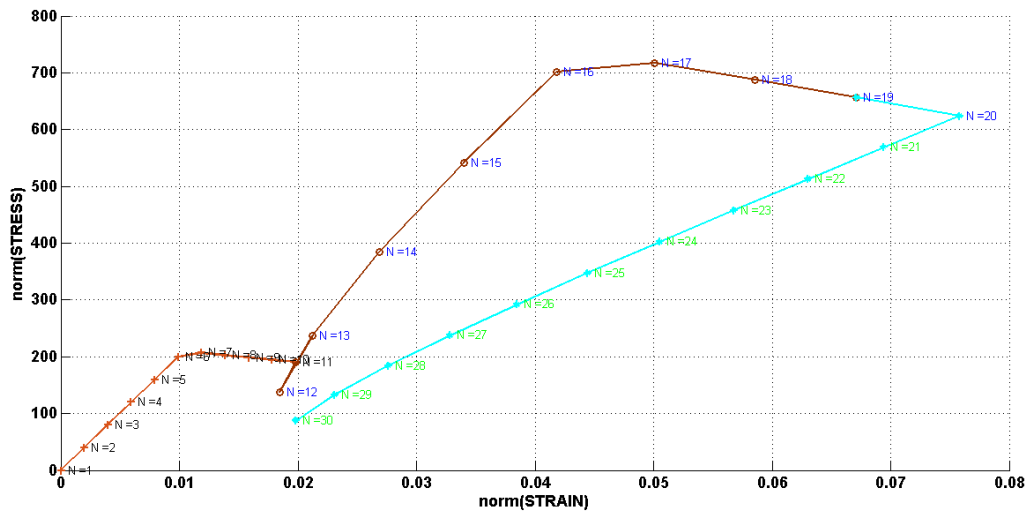
Figure 8:- The stress-strain curve for the biaxial "Non-symmetric tension-compression damage" model

The "tension-only" damage model

$(\overline{\Delta \sigma_1} \ (1) = 300 \ , \ \overline{\Delta \sigma_2} = 0), \ (\overline{\Delta \sigma_1} \ (1) = -800 \ , \ \overline{\Delta \sigma_2} = -800), \ (\overline{\Delta \sigma_1} \ (1) = 500, \ \overline{\Delta \sigma_2} = 500)$

The hardening modulus has been considered to be -0.05. At the starting, during the uniaxial elastic loading, inside the elastic domain, the effective and the actual stresses are same. After which when the stress path leaves the elastic domain for tension, loading occurs. Following this when biaxial tensile unloading/compressive loading occurs and the biaxial elastic limit due to compression can never be reached. Following this, the biaxial compressive unloading/tensile loading takes place.
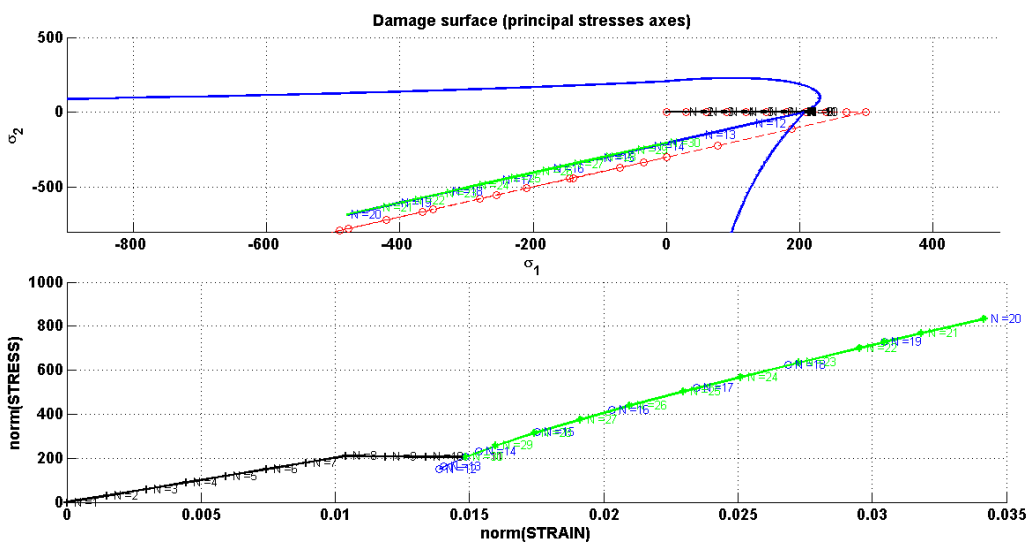

Figure 9:- Path at the stress spacefor the "tension-only" damage model

Third Case:

$(\overline{\Delta\sigma_1}\ (1) = \ \alpha\ ,\ \overline{\Delta\sigma_2} = \alpha),\ (\overline{\Delta\sigma_1}\ (1) = -\beta\ ,\ \overline{\Delta\sigma_2} = -\beta),\ (\overline{\Delta\sigma_1}\ (1) = \gamma,\ \overline{\Delta\sigma_2} = \gamma)$

"Non-symmetric tension-compression damage" model

$(\overline{\Delta\sigma_1}\ (1) = 300\ ,\ \overline{\Delta\sigma_2} = 300),\ (\overline{\Delta\sigma_1}\ (1) = -1500\ ,\ \overline{\Delta\sigma_2} = -1500),\ (\overline{\Delta\sigma_1}\ (1) = 1200,\ \overline{\Delta\sigma_2} = 1200)$

The hardening modulus has been considered to be 0. Initially, during the biaxial elastic loading, till the stress state reaches the elastic domain due to tension, the effective and the actual stresses are same. After which when the stress path leaves the elastic domain for tension loading occurs. . Following this when biaxial tensile unloading/compressive loading occurs, the actual stress follows the secant path till the stress path reaches the biaxial compressive elastic limit. After which, when the stress path leaves the elastic limit of compression, again loading occurs. Following this, the biaxial compressive unloading/tensile loading takes place.The evolution of the damage surface in the stress space has been shown in the **Figure 10**.
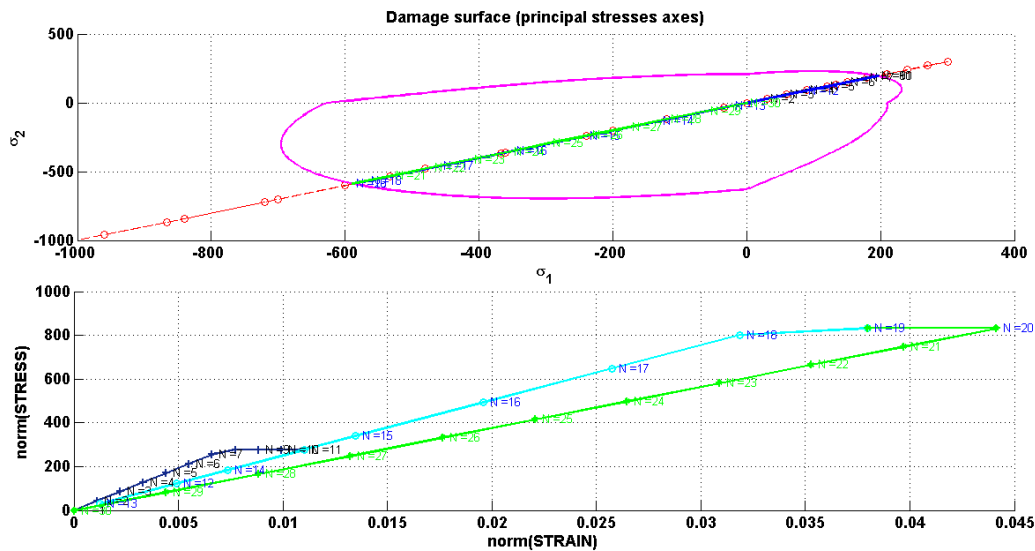

Figure 10:- Path at the stress spacefor the "Non-symmetric tension-compression damage"

The "tension-only" damage model

$(\overline{\Delta\sigma_1}\ (1) = 400\ ,\ \overline{\Delta\sigma_2} = 400),\ (\overline{\Delta\sigma_1}\ (1) = -900\ ,\ \overline{\Delta\sigma_2} = -900),\ (\overline{\Delta\sigma_1}\ (1) = 500,\ \overline{\Delta\sigma_2} = 500)$

The hardening modulus has been considered to be 0.1. Initially up to the elastic limit, during the uniaxial elastic loading, the effective and the actual stresses are same. After which when the stress path leaves the elastic domain for tension, loading occurs. After that when biaxial tensile unloading/compressive loading occurs, the actual stress follows the secant path and the biaxial elastic limit due to compression can never be reached. The evolution of the damage surface in the stress space has been shown in the Figure 11.
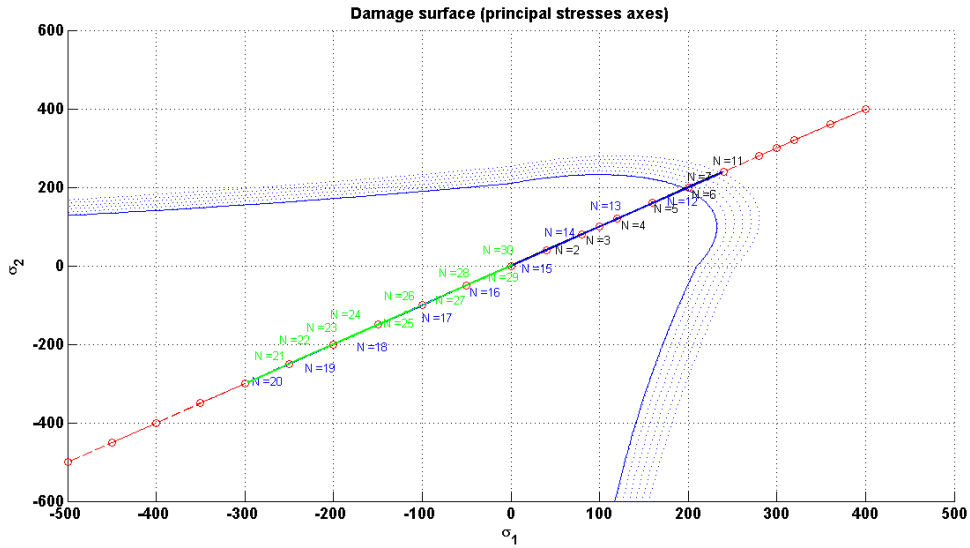
Figure 11(a):- Path at the stress spacefor the "tension-only" damage model
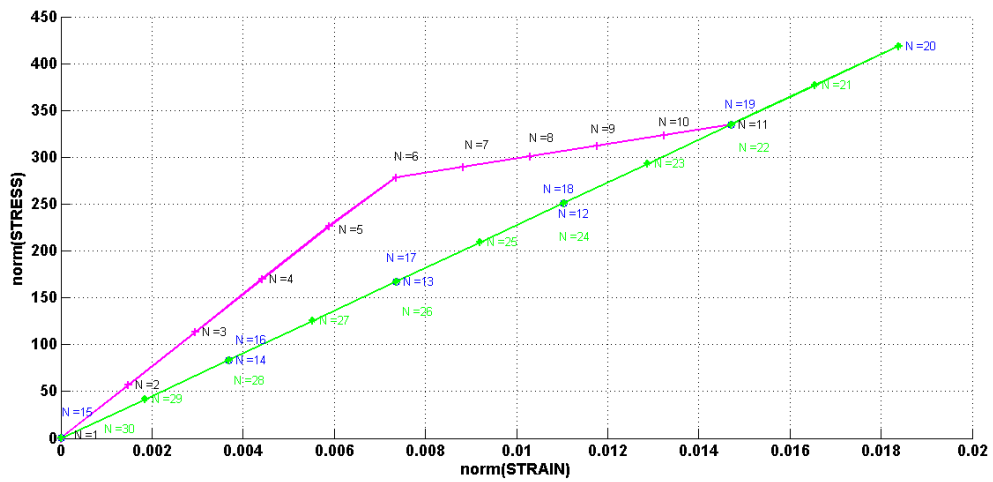

Figure 11(b):- The stress-strain curve for the biaxial "The tension-only damage model"

PART 2

(D) The integration algorithm for the continuum isotropic viscous-damage "symmetric tension compression" model has been implemented successfully. The code is presented in the Annex. In this type of models the rate effects are accommodated into the in-viscid damage model. The stresses become dependent on both the strains and the rate of the strains. The evolution of the stresses are shown in the **Figure 12**. For this observation the effective stress path is chosen such that $(\overline{\Delta\sigma_1}(1) = 200, \overline{\Delta\sigma_2} = 0)$, $(\overline{\Delta\sigma_1}(1) = 400, \overline{\Delta\sigma_2} = 0)$, $(\overline{\Delta\sigma_1}(1) = -600, \overline{\Delta\sigma_2} = 0)$. The viscosity value has been taken $\eta = 5$, and other parameters are set as $\alpha = 5, \gamma = 0.3, H = -0.02$.
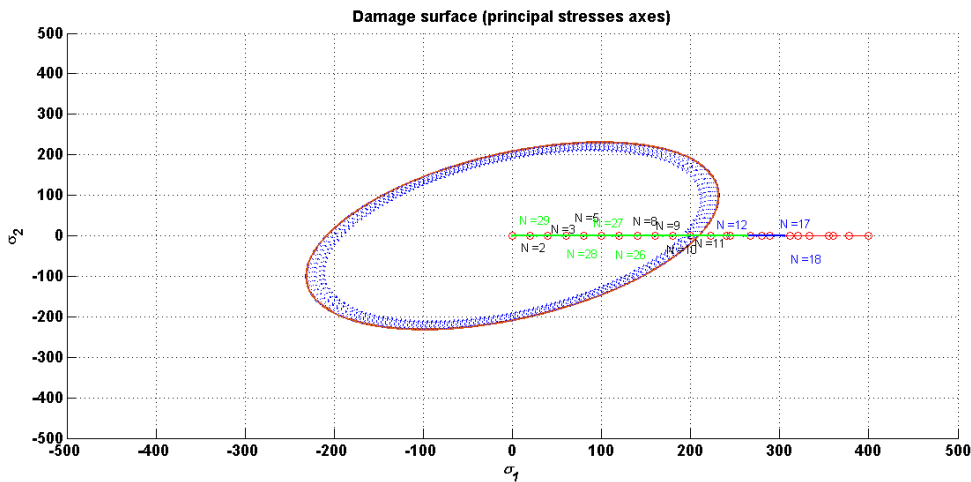

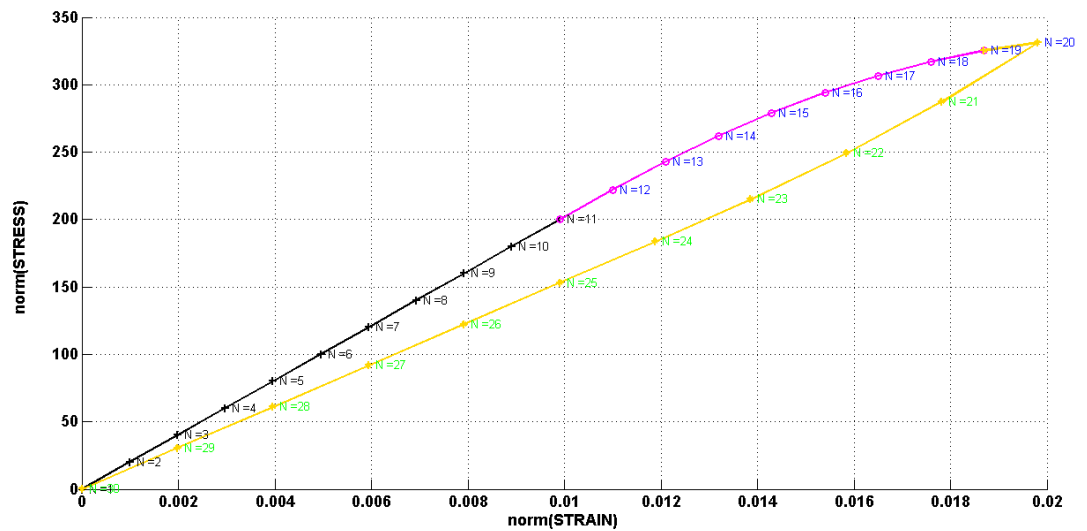**Figure 12**: The evolution of the actual stresses for a viscous damage model

Figure 13: Norm of stresses vs. norm of strains for a viscous damage model

Effect of Different viscosity parameters η

Outside the elastic region with increase in viscosity the stress value increases at a particular strain value. Inside the elastic region stress is independent of viscosity. This behaviour is plotted in the figure below, where stress norms are plotted vs. strain norms for different values of viscosity parameters η. For the observed case the considered parameters are: $\alpha = 0.5$, H=-0.02; $\gamma = 0.3$, The effective stress path is considered as (200,0); (400,0); (800,0).
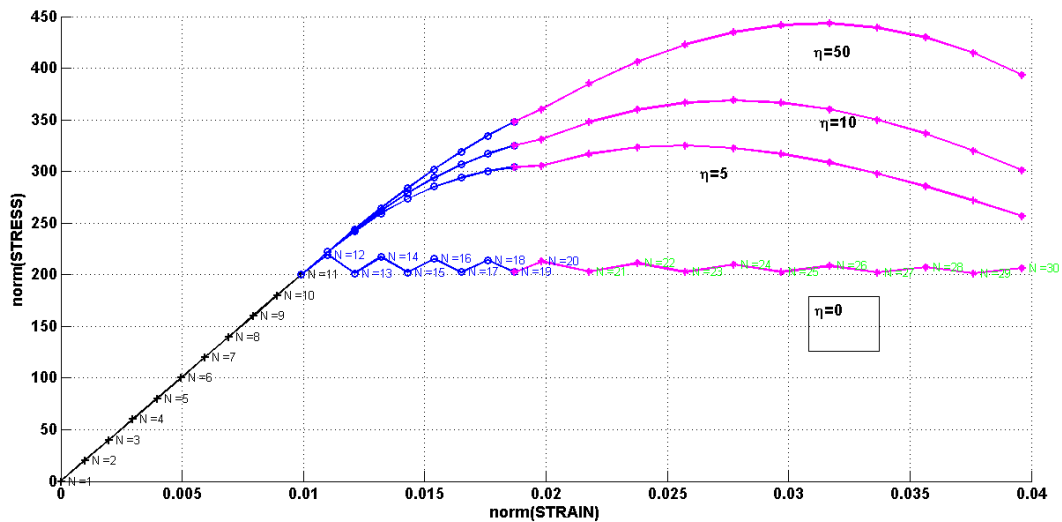
Figure 14:- Variation of Stress vs. Strain for different values of η

Effect of different strain rates:

It is found that outside the elastic region, with the increase in strain rate the stress at a particular strain value increases. Inside the elastic region stresses are independent of strain rates. H=0, $\gamma = 0.3$, $\eta = 5$ are considered here. The effective stress path: (200, 0); (400, 0); (800, 0).

Figure 15:- Variation of Stress vs. Strain for different values of the strain rates

Effect of different alpha values:

For the observationthe parameters are: H=0; $\gamma = 0.3, \eta = 20$. The effective stress path is taken as: (200, 0); (400,0); (800,0).The observations a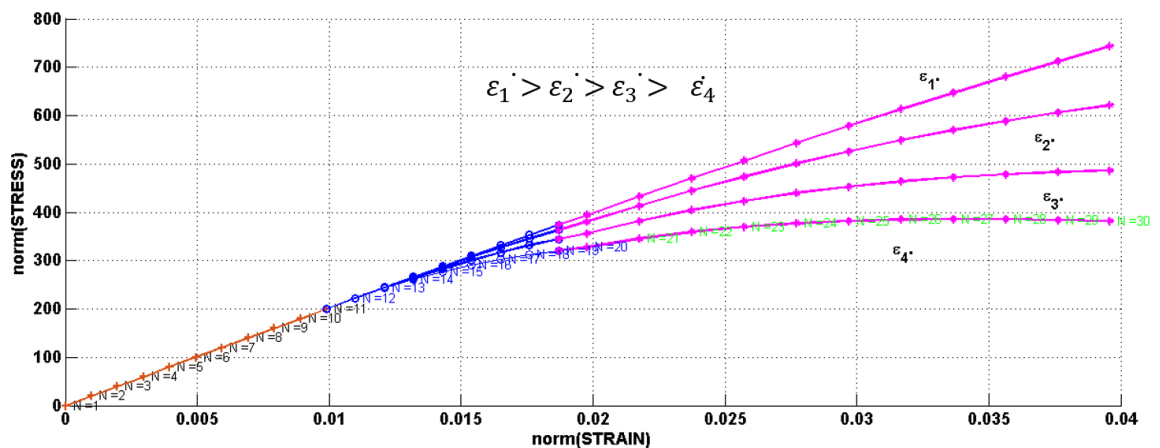re shown in the stress-strain plot below It is observed that for alpha=0 the numerical integration scheme is conditionally stable, and this explicit method gives first order accuracy. With the bigger time step size, this scheme becomes unstable. The similar effects are also observed for α = 0.25. It is an explicit scheme and gives first order accuracy and it is conditionally stable. For α= 0.5, the numerical scheme becomes unconditionally stable and it gives second order accuracy. For α= 0.75, the numerical scheme is unconditionally stable and it gives first order accuracy. For α= 1, the numerical scheme is unconditionally stable and it gives first order accuracy. It is an implicit scheme.
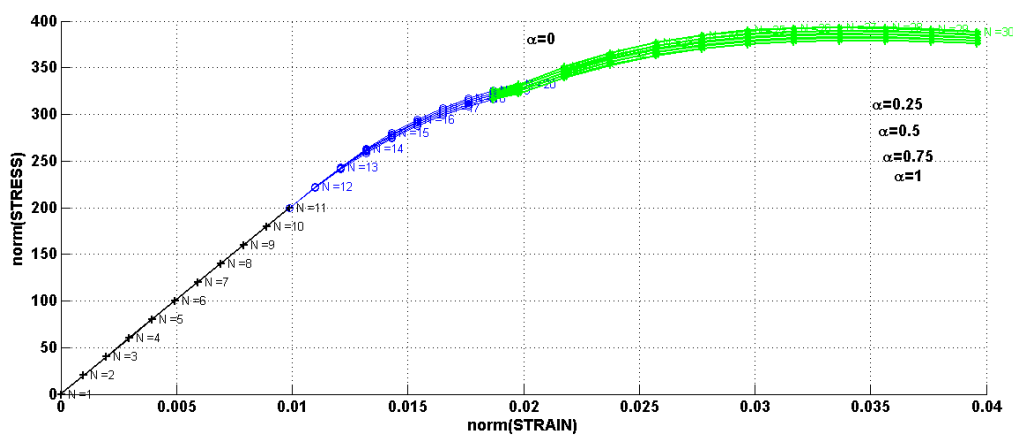
Figure 16:- Stress vs. Strain plot for different values of α

Effect of α on the evolution of the along time of the C11 component of the tangent and algorithmic constitutive operators:

The evolution of the C11 component of the tangent and algorithmic constitutive operators along the time has been studied for the different values of the 'α'. H= -0.02, $\gamma = 0.3, \eta = 10$ have been considered. The considered effective stress path: (200,0); (400,0); (800,0). It is investigated that in the elastic domain no evolution takes place for the C11 component of both the tangent and algorithmic constitutive operators along the time. Outside the elastic domain it is found that for higher alpha values at a particular time, the values of the C11 component of the tangent and algorithmic constitutive operators' decreases. The reduction is more for the algorithmic constitutive operators than the tangent constitutive operators at a particular time. When we consider α =0; both have same value. These observations are shown in the figures below.
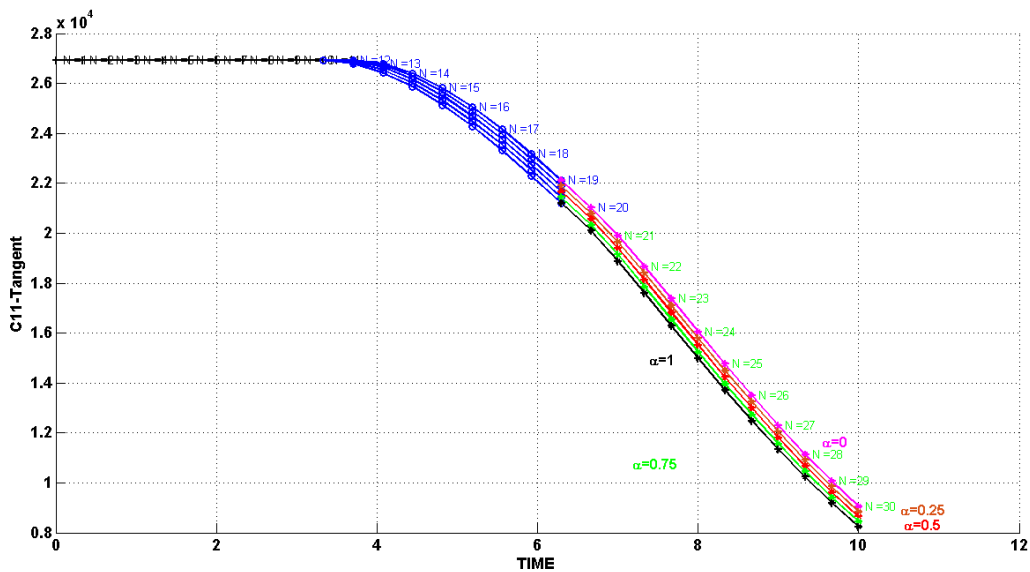
Figure 17: Evolution of the C11 component of the tangent constitutive operators along time for the different values of α
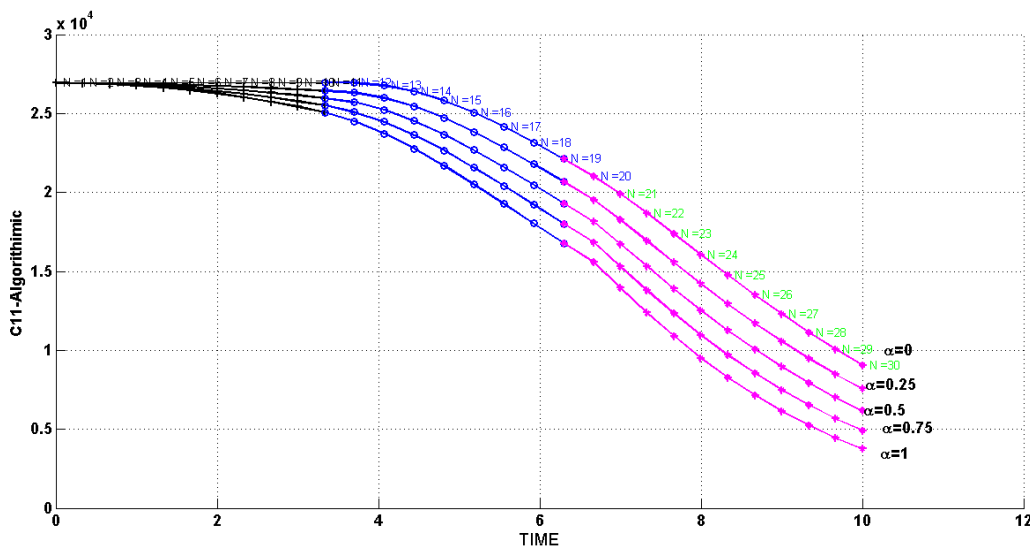
Figure 18: Evolution of the C11 component of the algorithmic constitutive operators along time for the different values of α

ANNEX:

A

```
elseif (MDtype==2)  %* Only tension

sigmab=(eps_n1*ce);
sigmabpos=sigmab.*(sigmab>0);

rtrial=sqrt(sigmabpos*eps_n1');
```

```
elseifMDtype==2
tetha=[(-pi/2)*0.9999:0.01:pi*0.9999];
%***************************************************************************
************
%* 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) ;

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

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

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

B

```
elseif (MDtype==3)  %*Non-symmetric

sigma=(eps_n1*ce);
sigmapos=sigma.*(sigma>0);
sigmaabs=abs(sigma);
tita=sum(sigmapos)/sum(sigmaabs);
    C=(tita+(1-tita)/n);
rtrial= C*sqrt(eps_n1*ce*eps_n1');


end
```

```
elseifMDtype==3
tetha=[0:0.01:2*pi];
%***************************************************************************
************
%* RADIUS
    D=size(tetha);                          %*  Range
```

```matlab
    m1=cos(tetha);                                    %*
    m2=sin(tetha);                                    %*
Contador=D(1,2);                                      %*


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


fori=1:Contador

sigma=[m1(i) m2(i) 0 nu*(m1(i)+m2(i))];
sigmapos=sigma.*(sigma>0);
tita=sum(sigmapos)/sum(abs(sigma));

radio(i)= (q/sqrt(sigma*ce_inv*sigma'))/(tita+(1-tita)/n);

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

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

end
%*********************************************************************
************
```

C AND D

```matlab
hvar_n1 = hvar_n;
r_n       = hvar_n(5);
q_n       = hvar_n(6);
E         = Eprop(1);
nu        = Eprop(2);
H         = Eprop(3);
sigma_u = Eprop(4);
hard_type = Eprop(5) ;
eta = Eprop(7);
ALPHA_COEFF = Eprop(8);
HARDSOFT_MOD = Eprop(3);
%*********************************************************************
************


%*********************************************************************
************
%*        initializing                                               %*
 r0 = sigma_u/sqrt(E);
zero_q=1.d-6*r0;
% if(r_n<=0.d0)
%     r_n=r0;
%     q_n=r0;
% end
%*********************************************************************
************
```

```matlab
%*************************************************************************
***********
%*       Damage
surface                                                                 %*
[rtrial_n] = Modelos_de_dano1 (MDtype,ce,eps_n0,n); % TauEps_n (Viscous
model)
[rtrial] = Modelos_de_dano1 (MDtype,ce,eps_n1,n);
[rtrial_nalpha]=(1-ALPHA_COEFF)*rtrial_n+ALPHA_COEFF*rtrial ; %
TauEps_n+alpha (Viscous model)
%*************************************************************************
***********


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

ifviscpr == 1
if (rtrial_nalpha>r_n)
%*   Loading
fload=1;
delta_r=rtrial_nalpha-r_n;
        r_n1=((eta-delta_t*(1-
ALPHA_COEFF))/(eta+ALPHA_COEFF*delta_t))*r_n+...
            (delta_t*rtrial_nalpha)/(eta+ALPHA_COEFF*delta_t);
ifhard_type == 0
%  Linear
            q_n1= q_n+ H*delta_r;
else
%Exponential

q_inf=r0+(r0-zero_q);
if HARDSOFT_MOD>0
                q_n1=q_n+((H*(q_inf-r0)/r0)*exp(H*(1-
rtrial_nalpha/r0)))*delta_r;
else
                q_n1=q_n+((H*(q_inf-r0)/r0)*(1/exp(H*(1-
rtrial_nalpha/r0))))*delta_r;
end

end

if(q_n1<zero_q)
            q_n1=zero_q;
end

else
%*       Elastic load/unload
fload=0;
        r_n1= r_n  ;
        q_n1= q_n  ;
```

```matlab
    end

else
if(rtrial>r_n)
%*    Loading

fload=1;
delta_r=rtrial-r_n;
        r_n1= rtrial   ;
ifhard_type == 0
%  Linear
            q_n1= q_n+ H*delta_r;
else
%Exponential

q_inf=r0+(r0-zero_q);
if HARDSOFT_MOD>0
                q_n1=q_n+((H*(q_inf-r0)/r0)*exp(H*(1-rtrial/r0)))*delta_r;
else
                q_n1=q_n+((H*(q_inf-r0)/r0)*(1/exp(H*(1-
rtrial/r0))))*delta_r;
end
end

if(q_n1<zero_q)
            q_n1=zero_q;
end


else

%*      Elastic load/unload
fload=0;
        r_n1= r_n   ;
        q_n1= q_n   ;


end

end

% Damage variable
% ---------------
dano_n1   = 1.d0-(q_n1/r_n1);
%  Computing stress
%  ***************
sigma_n1  =(1.d0-dano_n1)*ce*eps_n1';
%hold on
%plot(sigma_n1(1),sigma_n1(2),'bx')

%*************************************************************************
***********


%*************************************************************************
***********
%* Updating historic
variables                                                    %*
```

```
%  hvar_n1(1:4)  = eps_n1p;
hvar_n1(5)= r_n1 ;
hvar_n1(6)= q_n1 ;
%************************************************************************
***********
```

```
%************************************************************************
***********
%* Auxiliar
variables                                                                %*
aux_var(1) = fload;
aux_var(2) = q_n1/r_n1;
%*aux_var(3) = (q_n1-H*r_n1)/r_n1^3;
%************************************************************************
***********
```