Assignment 2: Computational Plasticity

Jordi Parra Porcar
CIMNE
jordiparraporcar@gmail.com

1. 1D Computational Plasticity

In order to carry out all the tests, the material parameters presented below are going to be chosen as the reference ones. Furthermore, in order to study the effect of the parameter aim of study, a higher and lower value of the reference one, are going to be chosen. Remarks: All magnitudes presented are measured in the International System of Units, in except from the time that is given in ms.

E	σ_y	K	H	dt	η
210000	300	50000	50000	0.01	3000

Table 1: Parameters for linear case

σ_∞	δ
500	1000

Table 2: Parameters for exponential saturation law (MPa)

1.1. Perfect plasticity case

For the perfect plasticity case the isotropic and kinematic hardening parameters are set to 0.

1.1.1. Viscosity dependence

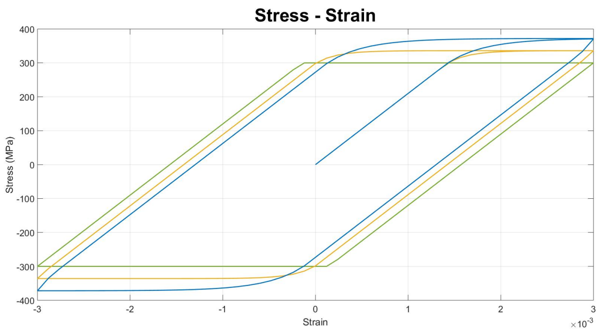

Figure 1: η=0 (green), η=3000 (yellow), η=6000 (blue)

Assessment:

The fact of varying the viscosity from its reference value (η=3000) has the expected effect on the stresses. The higher the viscosity for a fixed simulation time, the higher the stresses will be. Furthermore, Setting the viscosity parameter to 0 means recovering the rate independent case.

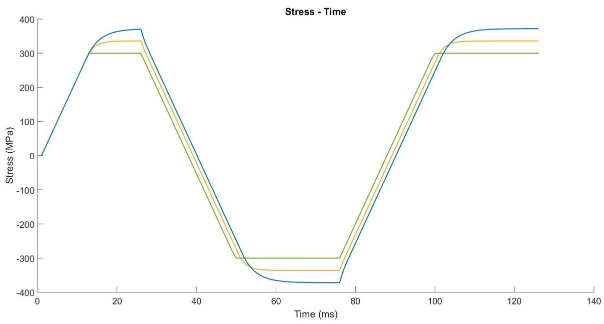

Figure 2: η=0 (green), η=3000 (yellow), η=6000 (blue)

Assessment:
It is possible to observe that the yield strength does not increase with time in neither of the cycles, that is due to the fact that the model has no hardening.

1.1.2. Strain rate dependence

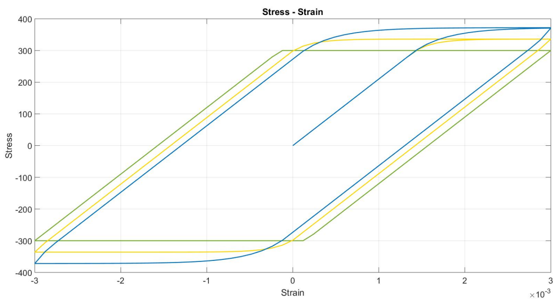

Figure 3: $\dot{\epsilon}$=0.1 (green), $\dot{\epsilon}$=100 (yellow), $\dot{\epsilon}$=200 (blue)

Assessment:
It is shown that low strain rate (quasistatic load) gives similar results to the inviscid case. On the other hand multiplying the strain rate by 2 gives same results as with the double value of viscosity.

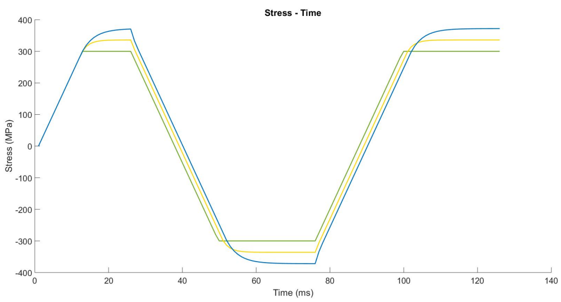

Figure 4: $\dot{\epsilon}$=0.1 (green), $\dot{\epsilon}$=100 (yellow), $\dot{\epsilon}$=200 (blue)

Assessment:
The response of the load vs time it is analogous to the the tests at the previous section, with viscosity variation.

1.2. Isotropic hardening

1.2.1. Rate independent

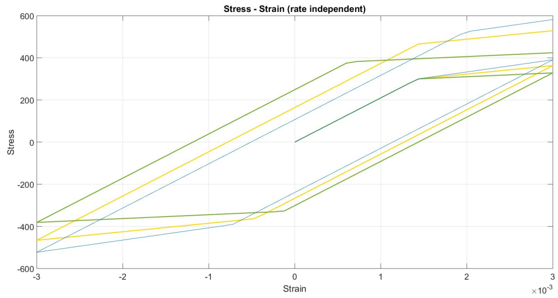

Figure 5: K=20000 (green), K=50000 (yellow), K=80000 (blue)

Assessment:
It can be observed that after the first plastic load the elastic domain increases and in further loads reaching the plastic strain area needs higher values of strain. The hardening isotropic hardening behaviour is however is symmetrical between tension and compression. It is important to remark that the response to the hardening is immediate once the yielding point is reached and increasing linearly from that point on.

1.2.2. Rate dependent

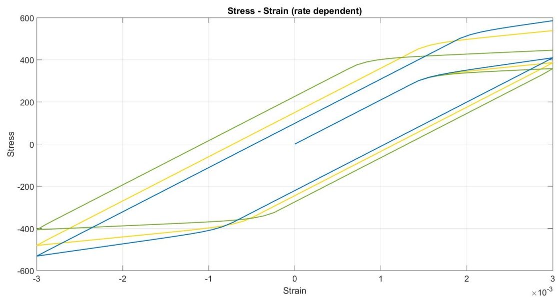

Figure 6: K=20000 (green), K=50000 (yellow), K=80000 (blue)

Assessment:
In figure 4, unlike to the behaviour presented in Figure 3, in this case the viscous effects are present and this fact can be observed once the yielding point is reached. The increase in yielding is no longer linear due to viscosity.

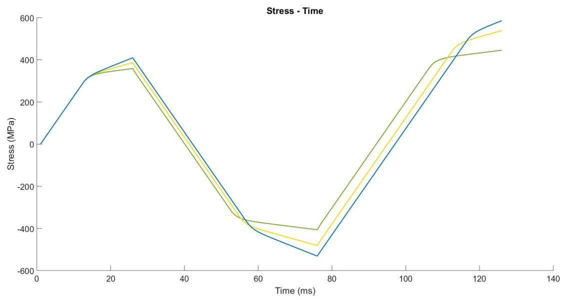

Figure 7: K=20000 (green), K=50000 (yellow), K=80000 (blue)

Assessment:
In this plot of stress vs time it is shown clearly the cyclic hardening effect due to the fact that, the yield strength it is getting higher after each cycle since the elastic domain is growing.

1.3. Isotropic hardening. Non linear saturation law. Rate independent

The isotropic hardening parameter K is set to 0 for the tests carried out in this section.

1.3.1. Variation of σ_∞

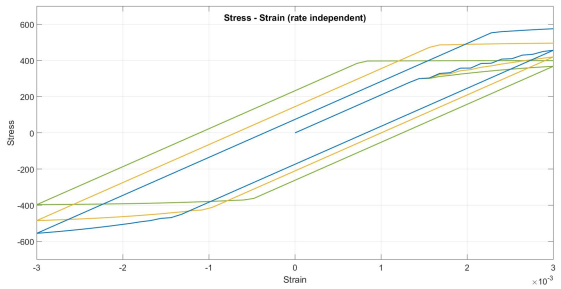

Figure 8: σ_∞=400 (green), σ_∞=500 (yellow), σ_∞=600 (blue)

Assessment:
In figure 5, it can be observed that the higher the σ_∞ value the more difficult is to reach the saturation stress.

1.3.2. Variation of δ

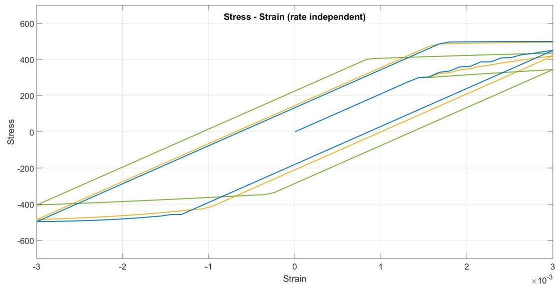

Figure 9: $\delta = 200(green), \delta = 1000(yellow), \delta = 1800(blue)$

Assessment:
In figure 6, it can be observed that the higher the δ parameter value the higher the velocity to reach the value of σ_∞=500 .

1.4. Kinematic hardening

1.4.1. Rate independent

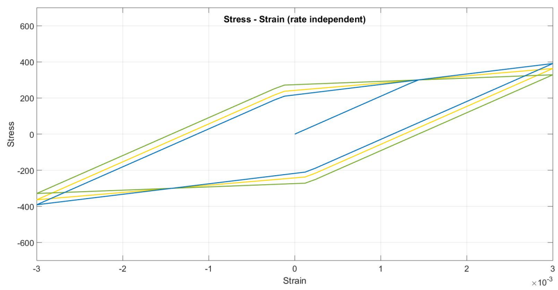

Figure 10: H=20000 (green), H=50000 (yellow), H=80000 (blue)

Assessment:
The higher the value of the kinematic hardening modulus, the higher the difference between yielding in tension an compression, as the material yields earlier in compression after unloading in tension, hence the Bauschinger effect can be captured in a desired manner depending on the type of material that is going to be modeled.

1.4.2. Rate dependent

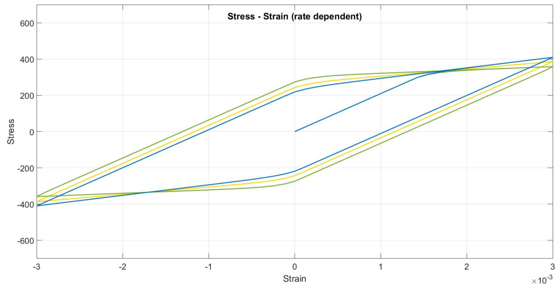

Figure 11: H=20000 (green), H=50000 (yellow), H=80000 (blue)

Assessment:
In apart from the effects explained in the rate independent section, in the rate dependent case the effects viscosity in the transition from elastic to plastic regime can be observed.

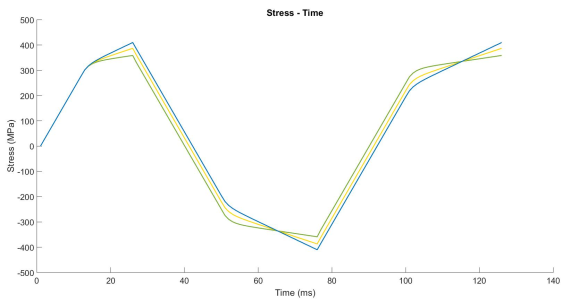

Figure 12: H=20000 (green), H=50000 (yellow), H=80000 (blue)

Assessment:
The plot of stress vs time shows clearly the translation of the elastic domain in each load cycle as well as the fact that, the size of the elastic domain remains constant due to the absence of isotropic hardening.

1.5. Combined hardening

1.5.1. Rate independent

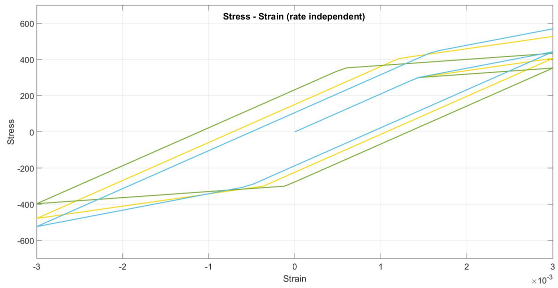

Figure 13: H=K=20000 (green), H=K=50000 (yellow), H=K=80000 (blue)

Assessment:
In figure 9, it is shown that combining the two types of hardening (isotropic and kinematic) gives the expected results since, on the one side the yield domain expands (isotropic hardening effect) and on the other side the domain also translates (kinematic hardening effect).

1.5.2. Rate dependent

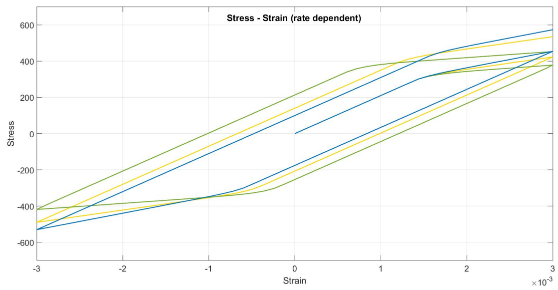

Figure 14: H=K=20000 (green), H=K=50000 (yellow), H=K=80000 (blue)

Assessment:
In the rate dependent case can be observed the effects of the viscosity in the transition from elastic to plastic regime.

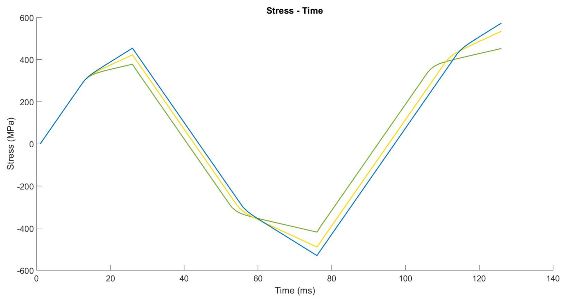

Figure 15: H=K=20000 (green), H=K=50000 (yellow), H=K=80000 (blue)

Assessment:
In the rate dependent case can be observed the effects of the viscosity in the transition from elastic to plastic regime.

2. J2 3D Computational Plasticity

In order to be able to compare the 3D model with the 1D, a uniaxial cyclic loading is going to be chosen, furthermore, the same material parameters used for 1D plasticity are going to be used for J2 3D plasticity; however in this case we must include the Poison ratio, which is set to 0.3 as a reference.

2.1. Perfect plasticity

In perfect plasticity the isotropic and kinematic hardening parameters are set to 0.

2.1.1. Rate independent

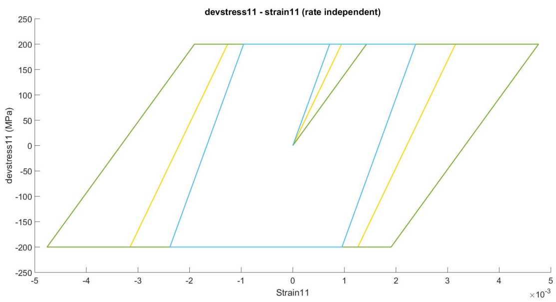

Figure 16: $\nu = 0(green), \nu = 0.3(yellow), \nu = 0.5(blue)$

Assessment:

The perfect plasticity behaviour is captured by means of the behaviour of $dev\sigma$ in respect to the uniaxial ϵ_{11}

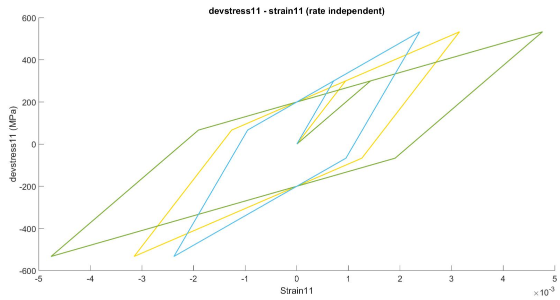

Figure 17: $\nu = 0(green), \nu = 0.3(yellow), \nu = 0.5(blue)$

Assessment:
Due to the affectation of Poisson ratio in the principal stress components, in the plot σ_{11} in respect to ϵ_{11}, it is not possible to observe perfect plasticity

2.1.2. Rate dependent

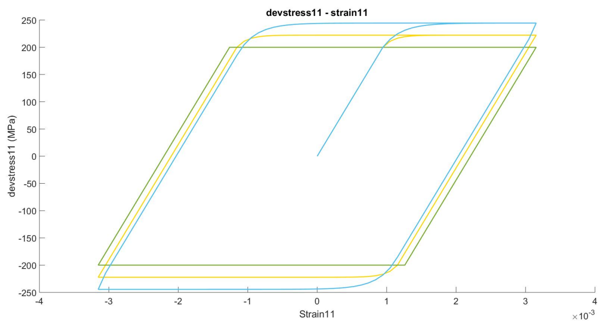

Figure 18: η=0 (green), η=3000 (yellow), η=6000 (blue)

Assessment:
As it occurs in 1D the higher the viscosity for a fixed simulation time, the higher the stresses turn out to be, in this case the deviatoric stress. Furthermore, Setting the viscosity parameter to 0 means recovering the rate independent case.

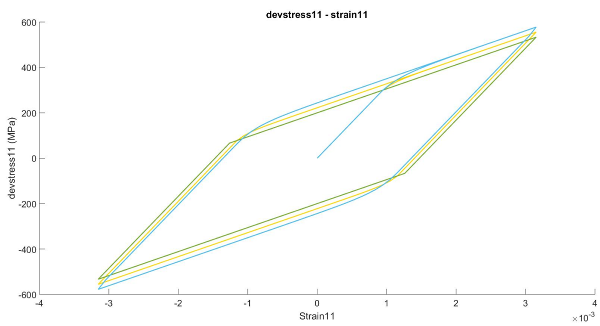

Figure 19: η=0 (green), η=3000 (yellow), η=6000 (blue)

Assessment:
The same assessment carried out for the deviatoric stresses is valid for σ_{11}

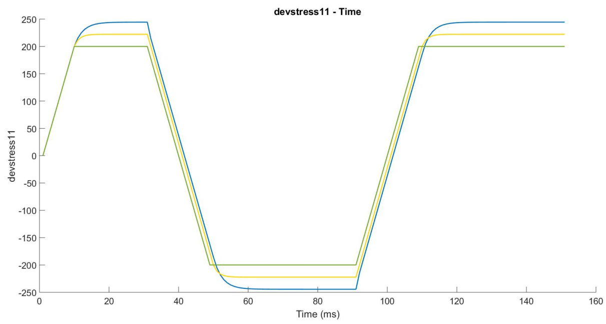

Figure 20: η=0 (green), η=3000 (yellow), η=6000 (blue)

Assessment:
The viscosity effects are shown through the evolution on time and the assessment of figures 13 and 14 is valid for this one as well.

2.2. Isotropic hardening

2.2.1. Rate independent

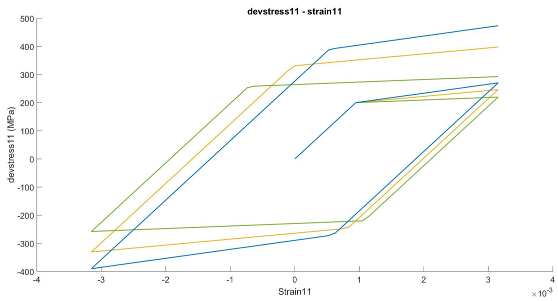

Figure 21: K=20000 (green), K=50000 (yellow), K=80000 (blue)

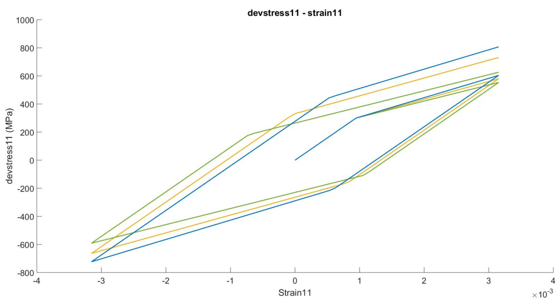

Figure 22: K=20000 (green), K=50000 (yellow), K=80000 (blue)

Assessment:
The behaviour of $dev\sigma_{11}$ and σ_{11} is analogous to the behaviour observed for the 1D case. In each load cycle σ_y increases, so that the load cycles are geometrically opened.

2.2.2. Rate dependent

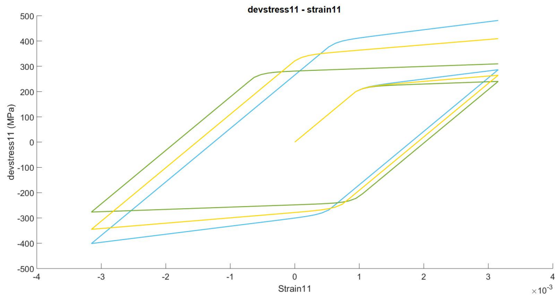

Figure 23: K=20000 (green), K=50000 (yellow), K=80000 (blue)

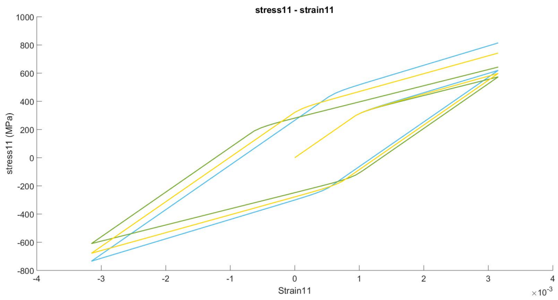

Figure 24: K=20000 (green), K=50000 (yellow), K=80000 (blue)

Assessment:
The behaviour of $dev\sigma_{11}$ and σ_{11} for the viscous case is analogous to the behaviour observed for the 1D case. In each load cycle σ_y increases. The nonlinear behaviour in the plastic region caused by the viscous effects.

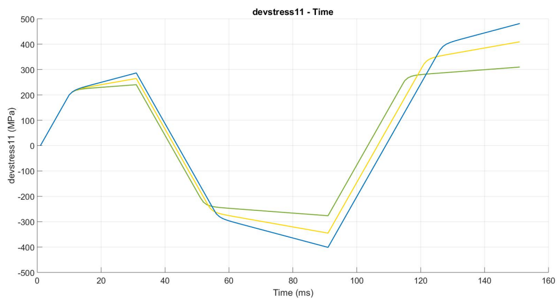

Figure 25: K=20000 (green), K=50000 (yellow), K=80000 (blue)

Assessment:
The behaviour of the $dev\sigma_{11}$ vs time is analogous as the behaviour of stress vs time in the 1D case

2.3. Isotropic hardening with non linear saturation law. Rate independent

The isotropic hardening parameter K is set to 0 for the tests carried out in this section. Only the results for the deviatoric part of stress it is shown in this section.

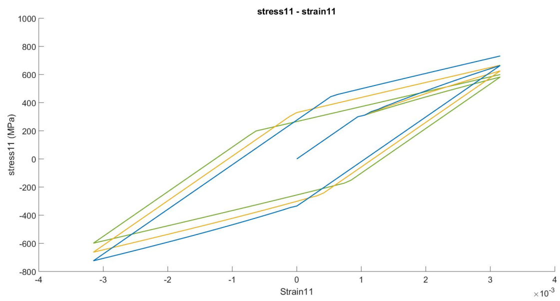

Figure 26: σ_∞=400 (green), σ_∞=500 (yellow), σ_∞=600 (blue)

Assessment:
The higher the σ_∞ value the more difficult is to reach the saturation stress.

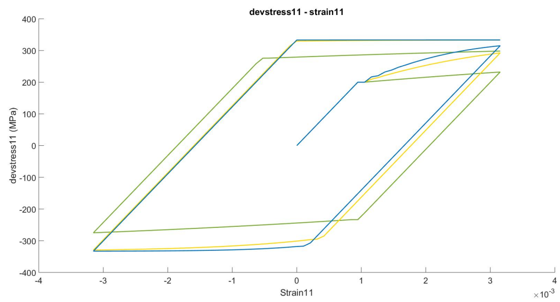

Figure 27: $\delta = 200(green), \delta = 1000(yellow), \delta = 1800(blue)$

Assessment:
The variation of the parameters in the non linear saturation law produces analogous results to the 1D model.

2.4. Kinematic hardening

2.4.1. Rate independent

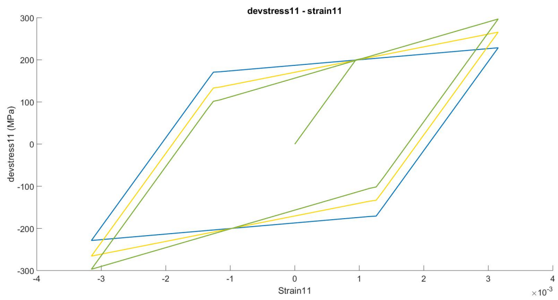

Figure 28: H=20000 (green), H=50000 (yellow), H=80000 (blue)

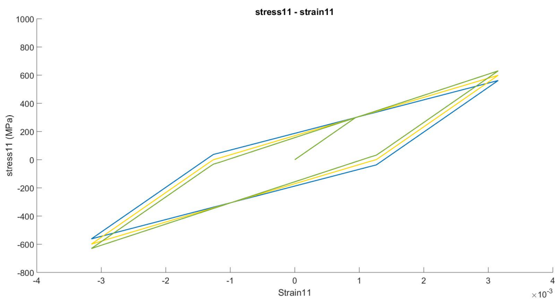

Figure 29: H=20000 (green), H=50000 (yellow), H=80000 (blue)

Assessment:
The results for the kinematic hardening are analogous to the 1D case, that is, an increase in the kinematic hardening modulus causes a higher translation of the elastic domain so that, the difference between tension and compression gets larger.

2.4.2. Rate dependent

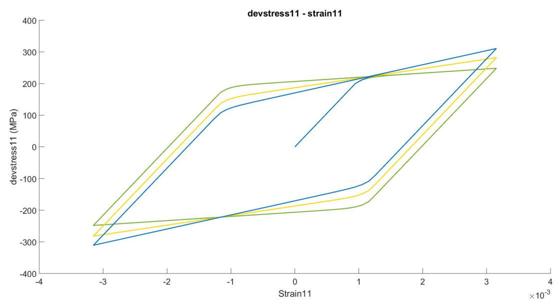

Figure 30: H=20000 (green), H=50000 (yellow), H=80000 (blue)

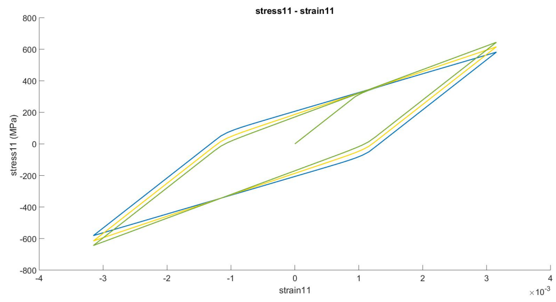

Figure 31: H=20000 (green), H=50000 (yellow), H=80000 (blue)

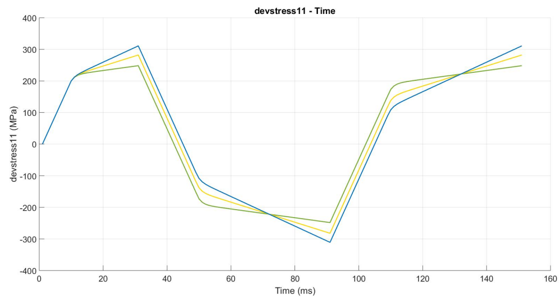

Figure 32: H=20000 (green), H=50000 (yellow), H=80000 (blue)

Assessment:
The effects of viscosity in the kinematic hardening behaviour are analogous to the 1D case.

2.5. Combined hardening. Rate dependent case

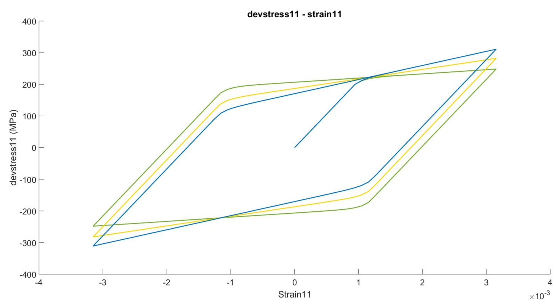

Figure 33: K=H=20000 (green), K=H=50000 (yellow), K=H=80000 (blue)

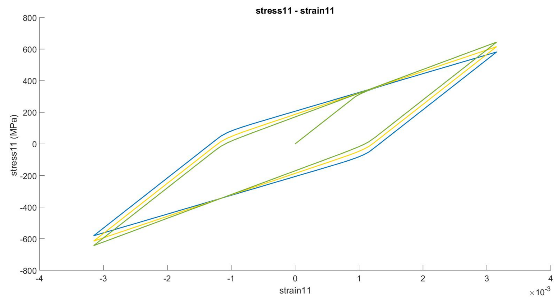

Figure 34: K=H=20000 (green), K=H=50000 (yellow), K=H=80000 (blue)

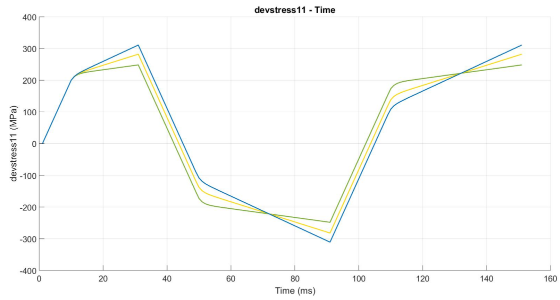

Figure 35: K=H=20000 (green), K=H=50000 (yellow), K=H=80000 (blue)

Assessment:
The combined hardening, on the one side shows a translation in the yield surface due to isotropic hardening and on the other side an increase in size due to isotropic hardening.

LIST OF FILES FOR 1D PLASTICITY

MAIN FILE: main.m

```
clear all
clc

pp = 1; % if pp = 1 -> Pure plasticity case
nlh = 0; % if nlh == 1 -> Non-lineal isotropic hardening
es = 1; % if es == 1 -> Exponential saturation law + linear hardening
E = 210e3; % Young modulus
K= 0; % Isotropic hardening paramenter
H= 0; % Kinematic hardening parameter
yields = 300; % Yield point in tension
yieldsc = -1*yields; % Yield point in compression
stress_inf = 500 ; % Limit for the exponential hardening law
delta = 0; % Second parameter for the exponential hardening law
eta = 0; % Viscosity parameter
dt = 0.01; % Time step for rate dependent case
istep = 25;
v = [E,K,H];
C = diag(v);
%Strain limits for cyclic loading
STRAIN_LOAD = [0 0.003 0 -0.003 0 0.003];
%strain_history
[strain, total_strain_n1] = strain_history( istep, STRAIN_LOAD );
%Additive split of strains
strain_p_n = [0 0 0];  %plastic strain
time = 1:length(strain);
[sigma_n1,strain_p_n, strain_e] = trial(pp,stress_inf,eta,nlh,es,E,K,H,dt,
total_strain_n1, yields, C, strain_p_n,delta);

figure(1)
plot(strain, sigma_n1(2:end,1))
hold on

figure(2)
hold on
plot(time, sigma_n1(2:end,1))
```

FUNCTION: strain_history.m

```
function [strain, total_strain_n1] = strain_history( istep, STRAIN_LOAD )

strain = zeros(1,sum(istep)+1);

tramo_b=[];
for i=1:length(STRAIN_LOAD)-1;
     e = linspace(STRAIN_LOAD(i),STRAIN_LOAD(i+1),istep+1);
     e = e(2:end-1);
     tramo_a = [STRAIN_LOAD(i) e];
     tramo_b = [tramo_b tramo_a] ;
end
tramo_b = [tramo_b STRAIN_LOAD(end)] ;
strain = tramo_b;

total = length(strain)
total_strain_n1 = zeros(total,3);
for j = 1:total
```

```
    total_strain_n1(j,:) = [strain(1,j) 0 0];
end

end
```

FUNCTION: trial.m

```
function [sigma_n1,strain_p_n, strain_e] =
trial(pp,stress_inf,eta,nlh,es,E,K,H,dt, total_strain_n1, yields, C, strain_p_n,
delta)

% Compute the trial state at time n+1
strain_p_n = zeros(length(total_strain_n1(:,1)),3);
strain_e = zeros(length(total_strain_n1(:,1)),3);
sigma_n1 = zeros(length(total_strain_n1(:,1)),3);

if eta == 0
   dt = 1;
end

for i = 1:(length(total_strain_n1(:,1)))
  strain_p_trial_n1 = strain_p_n(i,:);
  strain_e_trial_n1 = total_strain_n1(i,:) - strain_p_trial_n1;
  sigma_trial_n1 = (total_strain_n1(i,:) - strain_p_trial_n1)*C;

  if es == 1 %Exponential saturation law + linear hardening
       sigma_trial_n1(2) = -Exp_sat_law( stress_inf, yields, K, delta,
strain_p_trial_n1);
  end
  %yield function
      f_sigma_trial_n1 = abs(sigma_trial_n1(1) - sigma_trial_n1(3))- yields +
sigma_trial_n1(2);

  %pure plastic case
   if f_sigma_trial_n1 <= 0 %elastic step
      strain_p_n(i+1,:) = strain_p_trial_n1;
      strain_e(i+1,:) = strain_e_trial_n1 ;
      sigma_n1(i+1,:) = sigma_trial_n1;
   else
          if nlh == 1 % Newton-Raphson iterative solution algorithm (Nonlinear
isotropic hardening )
             gama_new = nrm(yields,stress_inf,f_sigma_trial_n1,K,E,H,dt,eta,
delta,strain_p_trial_n1 );
          else
              gama_new = 1/dt*(E+K+H+eta/dt)^-1 * f_sigma_trial_n1;
          end
   sigma_n1(i+1,1) = sigma_trial_n1(1)- dt*gama_new*E*sign(sigma_trial_n1(1) -
sigma_trial_n1(3));
   if nlh == 1;
            sigma_n1(i+1,2) = -Exp_sat_law( stress_inf, yields, K, delta,
strain_p_trial_n1+gama_new*dt);
   else
            sigma_n1(i+1,2) = sigma_trial_n1(2) - dt*gama_new * K;
   end
   sigma_n1(i+1,3) = sigma_trial_n1(3) + dt*gama_new * H*sign(sigma_trial_n1(1)
- sigma_trial_n1(3));

   strain_p_n(i+1,1) =  strain_p_trial_n1(1) +
dt*gama_new*sign(sigma_trial_n1(1)- sigma_trial_n1(3));
   strain_p_n(i+1,2) =  strain_p_trial_n1(2) + dt*gama_new;
   strain_p_n(i+1,3) =  strain_p_trial_n1(3) -
dt*gama_new*sign(sigma_trial_n1(1)- sigma_trial_n1(3));
```

```
        end
    end
end
```

FUNCTION: nrse.m

```
function [ gn1,Dgn1 ] = nrse(yields,stress_inf,f_sigma_trial_n1,
gama_k_n1,K,E,H,dt,nu,delta,strain_p_trial_n1);

%Nonlinear residual scalar equation on the plastic multiplayer
aa = Exp_sat_law( stress_inf, yields, K, delta, strain_p_trial_n1 );
bb = Exp_sat_law( stress_inf, yields, K, delta, strain_p_trial_n1 + gama_k_n1*dt
);
gn1 = f_sigma_trial_n1 - gama_k_n1*dt*(E+H+nu/dt)-(bb - aa);

%delta = delta * strain_p_trial_n1(2);

ddPI = (stress_inf - yields)*(dt*delta*exp(-
1*(delta*(strain_p_trial_n1(2)+gama_k_n1*dt))));
Dgn1 = -(E+H+nu/dt)*dt - ddPI*dt;
end
```

FUNCTION: nrm.m

```
function [ gamma_new] =
nrm(yields,stress_inf,f_sigma_trial_n1,K,E,H,dt,eta,delta_ci,strain_p_trial_n1 )
%UNTITLED4 Summary of this function goes here
%   Detailed explanation goes here
k = 0;
gama_k_n1 = 0.0001;
tol = 1e-8;
[ gn1,Dgn1 ] = nrse(yields,stress_inf,f_sigma_trial_n1,
gama_k_n1,K,E,H,dt,eta,delta_ci,strain_p_trial_n1 );
while  tol < abs(gn1)
   %solve the linearized equation
   gama_k_n1 = gama_k_n1 - gn1/Dgn1;
   [ gn1,Dgn1 ] = nrse(yields,stress_inf,f_sigma_trial_n1,
gama_k_n1,K,E,H,dt,eta,delta_ci,strain_p_trial_n1 );
   k = k+1;
end
   gamma_new = gama_k_n1;

end
```

FUNCTION: Exp_sat_law.m

```
function [ dPI ] = Exp_sat_law( stress_inf, yields, K, delta_ci,
strain_p_trial_n1);
dPI = (stress_inf - yields)*(1-exp(-delta_ci*strain_p_trial_n1(2)))+
K*strain_p_trial_n1(2);
end
```

LIST OF FILES FOR 3D J2 PLASTICITY

MAIN FILE: main.m

```
clear all
clc

pp = 0; % if pp = 1 is pure plasticity case
nlh = 0; % if nlh == 1 -> Non-lineal isotropic hardening
es = 1; % if es == 1 -> Exponential saturation law + linear hardening
E = 210000; % Young modulus
K= 80000; % Isotropic hardening paramenter
H= 80000; % Kinematic hardening parameter
yields = 300; % Yield point in compression
yieldsc = -1*yields;
delta = 1000; % Second parameter for the exponential hardening law
pois = 0.3; %Poison coefficient
eta=3000;
stress_inf = 500;% Limit for the exponential hardening law
dt = 0.01; % Time step for rate dependent case
istep = 30;

HH = H*eye(6);
[ce,mu] = tensor(E,pois);

[total_strain] = strain_history(ce,istep);

[dev_sigma,sigma_n,strain_p_n, strain_e_n] =
trial(mu,stress_inf,eta,nlh,es,E,K,H,HH,dt,delta, total_strain, yields, ce);
time = 1:length(total_strain);

figure(1)
plot(total_strain(:,1),dev_sigma(:,1))
hold on

figure(2)
plot(total_strain(:,1),sigma_n(:,1))
hold on

figure(3)
plot(time, dev_sigma(:,1))
hold on
```

FUNCTION: strain_history.m

```
function [  total_strain  ] = strain_history( ce, istep )

ce_1 = inv(ce);
step=5*istep+1;
stress_load = [0 1000 0 -1000 0 1000];
stress = zeros(6,step);
strain = zeros(6,step);


tramo_b=[];
for i=1:length(stress_load)-1
     e = linspace(stress_load(i),stress_load(i+1),istep+1);
     e = e(2:end-1);
     tramo_a = [stress_load(i) e];
     tramo_b = [tramo_b tramo_a] ;
end
```

```
tramo_b = [tramo_b stress_load(end)] ;
stress(1,:) = tramo_b;
for i=1:step
    strain(:,i) = ce_1*stress(:,i);
end
for i=1:3
total_strain = strain';
end
```

FUNCTION: trial.m

```
function [dev_sigma,sigma_n,strain_p_n, strain_e_n] =
trial(mu,stress_inf,eta,nlh,es,E,K,H,HH,dt,delta, total_strain, yields, ce)
% Compute the trial state at time n+1
total_strain_iso = zeros(length(total_strain(:,1)),1);
total_strain_ki = zeros(length(total_strain(:,1)),6);
strain_p_n = zeros(length(total_strain(:,1)),6);
strain_p_iso = zeros(length(total_strain(:,1)),1);
strain_p_ki = zeros(length(total_strain(:,1)),6);
strain_e_n = zeros(length(total_strain(:,1)),6);
strain_e_iso = zeros(length(total_strain(:,1)),1);
strain_e_ki = zeros(length(total_strain(:,1)),6);
sigma_n = zeros(length(total_strain(:,1)),6);
q_n = zeros(length(total_strain(:,1)),1);
q_bar_n = zeros(length(total_strain(:,1)),6);
dev_sigma = zeros(length(total_strain(:,1)),6);

if eta == 0
   dt = 1;
end

for i = 1:length(total_strain(:,1))-1

strain_p_n_trial = strain_p_n(i,:);
strain_p_iso_trial = strain_p_iso(i,:);
strain_p_ki_trial = strain_p_ki(i,:);

strain_e_n_trial = total_strain(i+1,:) - strain_p_n_trial;
strain_e_iso_trial = total_strain_iso(i+1,:) - strain_p_iso_trial;
strain_e_ki_trial = total_strain_ki(i+1,:) - strain_p_ki_trial;

sigma_n_trial = strain_e_n_trial*ce';
q_n_trial = strain_e_iso_trial*K;
q_bar_n_trial = strain_e_ki_trial*HH;


  %deviatoric part of sigma
   dev_sigma_trial = deviatoric(sigma_n_trial);
   den = dev_sigma_trial - q_bar_n_trial;
   n_trial = den / norm(den);

     if es == 1 %Exponential saturation law + linear hardening
      q_n_trial = -1*Exp_sat_law( stress_inf, yields, K, delta,
strain_p_iso_trial);

   end

   %yield function
      f_sigma_n_trial = norm(den) - sqrt(2/3)*(yields - q_n_trial);

%pure plastic case
   if f_sigma_n_trial <= 0 %elastic step
      gamma = 0;
```

```
      sigma_n(i+1,:) = sigma_n_trial;
      q_n(i+1) = q_n_trial ;
      q_bar_n(i+1,:) = q_bar_n_trial;
      dev_sigma(i+1,:)  = dev_sigma_trial; % deviatoric(sigma_trial_n1);

      strain_p_n(i+1,:) = strain_p_n_trial;
      strain_p_iso(i+1,:) = strain_p_iso_trial;
      strain_p_ki(i+1,:) = strain_p_ki_trial;

      strain_e_n(i+1,:) = strain_e_n_trial;
      strain_e_iso(i+1,:) = strain_e_iso_trial;
      strain_e_ki(i+1,:) = strain_e_iso_trial;

   else

           if nlh == 1 % Newton-Raphson iterative solution algorithm (Nonlinear
isotropic hardening )

              gamma = NRM(yields,stress_inf,f_sigma_n_trial,mu,K,E,H,dt,eta,
delta,strain_p_iso_trial );

           else
              gamma = (2*mu+2/3*K+2/3*H+eta/dt)^(-1) *f_sigma_n_trial/dt;
           end

   sigma_n(i+1,:)= sigma_n_trial - dt*gamma*2*mu*n_trial;
if nlh == 1
   q_n(i+1,:)=
-Exp_sat_law(stress_inf,yields,K,delta,strain_p_iso_trial+gamma*dt*sqrt(2/3));
else
    q_n(i+1,:) = q_n_trial - dt*gamma*K*sqrt(2/3);
end
   q_bar_n(i+1,:)= q_bar_n_trial + dt* gamma*2/3*H*n_trial;
   dev_sigma(i+1,:)  = deviatoric(sigma_n_trial) - dt*gamma*2*mu*n_trial;

   strain_p_n(i+1,:)= strain_p_n_trial + dt*gamma*n_trial;
   strain_p_iso(i+1,:)= strain_p_iso_trial + dt*gamma*sqrt(2/3);
   strain_p_ki(i+1,:)= strain_p_ki_trial - dt*gamma*n_trial;

  % if pp == 1 %pure plasticity

        % if sigma_n(i,1)>0
             % sigma_n(i+1,1) = yields;
            %  sigma_n(i+1,2) = sigma_n(i,2);
             % sigma_n(i+1,3) = sigma_n(i,3);
          % else
              % sigma_n(i+1,1) = -yields;
              % sigma_n(i+1,2) = -sigma_n(i,2);
             % sigma_n(i+1,3) = -sigma_n(i,3);
          % end
       %  dev_sigma(i+1,:)  = deviatoric(sigma_n(i,:));
         %E = E*(1-E*(E+K+H+eta/dt)^-1)
   %end
   end

end
end
```

FUNCTION: deviatoric.m

```
function [ dev_sigma ] = deviatoric(sigma_n_trial  )
%UNTITLED4 Summary of this function goes here
%   Detailed explanation goes here
s = sigma_n_trial ;
trace = s(1)+s(2)+s(3);
dev_sigma= [s(1)-(1/3)*trace,s(2)-(1/3)*trace,s(3)-(1/3)*trace,s(4),s(5),s(6)];
end
```

FUNCTION: tensor.m

```
function [ce,mu] = tensor( E, pois)

mu = E/(2*(1-pois));
lame = pois*E/(1+pois)*(1-2*pois);

        ce    = zeros(6,6);

for i=1:3
     ce(i,i)=2*mu + lame;
end
for i=4:6
    ce(i,i) = mu;
end
        ce(1,2)=lame;
        ce(1,3)=lame;
        ce(2,3)=lame;
        ce(2,1)=lame;
        ce(3,1)=lame;
        ce(3,2)=lame;
end
```

FUNCTION: Exp_sat_law.m

```
function [ dPI ] = Exp_sat_law( stress_inf, yields, K, delta,
strain_p_iso_trial)

dPI = (stress_inf - yields)*(1-exp(-delta*strain_p_iso_trial))+
K*strain_p_iso_trial;
end
```

FUNCTION: nrse.m

```
function [ gn1,Dgn1 ] = nrse(yields,stress_inf,f_sigma_n_trial,
gama_k_n1,mu,K,E,H,dt,eta,delta,strain_p_iso_trial)

%Nonlinear residual scalar equation on the plastic multiplayer
aa = Exp_sat_law( stress_inf, yields, K, delta, strain_p_iso_trial );
bb = Exp_sat_law( stress_inf, yields, K, delta, strain_p_iso_trial +
gama_k_n1*sqrt(2/3)*dt );
gn1 = f_sigma_n_trial - gama_k_n1*dt*(2*mu+2/3*H+eta/dt)-sqrt(2/3)*(bb - aa);
ddPI = (stress_inf - yields)*delta*sqrt(2/3)*dt*exp(-delta*(strain_p_iso_trial +
sqrt(2/3)*gama_k_n1*dt))+K*dt*sqrt(2/3);
Dgn1 = -dt*((2*mu+(2/3)*H+eta/dt) + 2/3*ddPI);
end
```

FUNCTION: ***nrm.m***

```
function [ gamma] =
nrm(yields,stress_inf,f_sigma_n_trial,mu,K,E,H,dt,eta,delta,strain_p_iso_trial )
%UNTITLED4 Summary of this function goes here
%   Detailed explanation goes here
k = 0;
gama_k_n1 = 0;
tol = 1e-6;

[ gn1,Dgn1 ] = nrse(yields,stress_inf,f_sigma_n_trial,
gama_k_n1,mu,K,E,H,dt,eta,delta,strain_p_iso_trial );
while abs(gn1) > tol
   %solve the linearized equation
   gama_k_n1 = gama_k_n1 - gn1/Dgn1;
   [ gn1,Dgn1 ] = nrse(yields,stress_inf,f_sigma_n_trial,
gama_k_n1,mu,K,E,H,dt,eta,delta,strain_p_iso_trial );
   k = k+1;
end
   gamma = gama_k_n1;

end
```