```
% @ 2024 Sichao Ding and Yaru Lv (Hohai University). All Rights Reserved.
% Unauthorized copying, modification, distribution, or commercial use is prohibited without
written permission.
% For licensing, please contact: 240408020003@hhu.edu.cn.
% A compressive rate-dependent hypoplastic constitutive model（Niemunis and Herle, 1996）
clear all
clc

%*******************************************************************
% Input the real-time strain rate time-history curve from the test
load('data1.mat');
data1 = table2array(data1);
%*******************************************************************

% Intergranular strain tensor parameter %inter(0)
% Calcareous sand
rr=8e-4;% Elastic range of initial stiffness
mr=3;
mt=2;
brc=0.4;
xxc=8;

% model parameters of Wolffersdorff (1996)
j=pi/180*(34);% Critical state friction angle
hsc=43000;% kpa（quasi-static）
n=0.76;
ec0=1.389;
ed0=0.695;
ei0=1.736;
a=0.25;
b=1;

% Rate-dependent dynamic parameters
hsmax=5*hsc;
xxmax=5*xxc;
brmin=-0.33;
Ds=600;
u=2.5;

% 图号
picturei=2;

% Test condition
% Void ratio（statev）
es0=1.265;%initial void ratio
es=es0;
% Initial pressure (kPa)
p0=0.0000001;% 没有固结 Convergence
% Initial stress state
t330=-p0;
t110=-p0;

% Initial intergranular strain tensor（A point）Second-order tensor
inter=[0 0 0;0 0 0;0 0 0];% inter(1)

% Initial strain rate tensor
dd=[0 0 0;0 0 0;0 0 0];
```

```
% Initial stress rate tensor
tt=[0 0 0;0 0 0;0 0 0];

% Initial stress tensor
t=[t110-0.000001 0 0;0 t330 0;0 0 t330];

% Initial strain tensor
d=[0 0 0;0 0 0;0 0 0];
% dd11=-1000; % Axial constant-loading strain rate with compression defined as negative
dd33=0;
st=0.00000001;% 设置时间步
i=1;% 循环次数

% Second-order Identity Tensor
i2=[1 0 0;0 1 0;0 0 1];
% Fourth-order Identity Tensor
for k1=1:3
    for k2=1:3
        for k3=1:3
            for k4=1:3
i4(k1,k2,k3,k4)=1/2*(i2(k1,k3)*i2(k2,k4)+i2(k1,k4)*i2(k2,k3));
            end
        end
    end
end


while abs(d(1,1))<0.205 % Specify the axial strain range

% 1.Deviatoric Stress Tensor

%*******************************************************************
time=i*st;
Time(i) = time;
[DD11] = interpolation(data1,Time);
dd11 = -DD11(i);
% dd11=-(-1e22*time^5+8e18*time^4-2e15*time^3+2e11*time^2+3e6*time);
% dd11=-460;
%*******************************************************************

dd=[dd11 0 0;0 0 0;0 0 0]; % Loading condition

pt=t-(1/3)*trace(t)*i2;
if trace(pt^2)==0;% 偏应力张量的所有特征值都为零，即偏应力张量为零张量
npt=pt;
dwpt=pt;
dwt=t;
else
npt=pt/sqrt(trace(pt^2));% 加载方向(unit direction) sqrt(trace(pt^2))为偏应力张量的模（用来求矩阵的模）
dwpt=pt/trace(t);% 正则化偏应力张量
dwt=t/trace(t);% 正则化应力张量
end

% 2.Calculate F 即fff
tanpsi=sqrt(3)*sqrt(trace(dwpt^2));% AppendixA：控制F sqrt(trace(dwpt^2))为正则化偏应力张量的模
if trace(dwpt^2)<0.000000001;% 各项同性应力状态（无偏应力）
```

```
cos3theta=-1;
fff=1;
else
cos3theta=-sqrt(6)*trace(dwpt^3)/(trace(dwpt^2))^1.5;
fff=sqrt((1/8)*tanpsi^2+(2-tanpsi^2)/(2+sqrt(2)*cos3theta*tanpsi))-1/(2*sqrt(2))*tanpsi;
end
% The norm of the volumetric strain rate tensor
D=sqrt(trace(dd^2));

% 3.Calculate fd and fs
% Rate-dependent modification
br=brc+(brmin*(D/Ds)^u)/((D/Ds)^u+1);
xx=xxc+(xxmax*(D/Ds)^u)/((D/Ds)^u+1);
hs=hsc+(hsmax*(D/Ds)^u)/((D/Ds)^u+1);
% mr=mrc+(mrmax*(D/Ds)^u)/((D/Ds)^u+1);
kk=exp(-((-trace(t))/hs)^n);
ei=ei0*kk;
ed=ed0*kk;
ec=ec0*kk;

fd=((es-ed)/(ec-ed))^a;
fe=(ec/es)^b;

aaa=sqrt(3)*(3-sin(j))/(2*sqrt(2)*sin(j));
fb=hs/n*(1+ei)/ei*(ei0/ec0)^b*((-trace(t))/hs)^(1-n)*(3+aaa^2-aaa*sqrt(3)*((ei0-ed0)/(ec0-ed0
))^a)^(-1);
fs=fb*fe;

% 4.Calculate L and N

% L:Tensor of fourth order
for k1=1:3;
    for k2=1:3;
        for k3=1:3;
            for k4=1:3;
L(k1,k2,k3,k4)=fs*1/trace(dwt*dwt)*(fff^2*i4(k1,k2,k3,k4)+aaa^2*dwt(k1,k2)*dwt(k3,k4));

            end
        end
    end
end
% N:Second order tensor
for k1=1:3;
    for k2=1:3;
N(k1,k2)=fs*fd*fff*aaa/trace(dwt*dwt)*(dwt(k1,k2)+dwpt(k1,k2));
    end
end

% 5.Intergranular strain tensor
% 单位（方向）粒间应变张量
lengthinter=sqrt(trace(inter^2));% 粒间应变张量长度（模）
if lengthinter==0;
ninter=[0 0 0;0 0 0;0 0 0];% 同pointA的inter
else;
ninter=inter/lengthinter;% 正则化,带方向二阶张量(粒间应变张量的演化方向)
end;
pp=lengthinter/rr;% 标准化后的粒间应变张量长度ρ(0,1)
```

```
% 6.Calculate M
% 6.1 Calculate L:ninter
for k1=1:3;
    for k2=1:3;
Lninter(k1,k2)=0;%二阶张量
         for k3=1:3;
            for k4=1:3;
Lninter(k1,k2)=Lninter(k1,k2)+L(k1,k2,k3,k4)*ninter(k3,k4);%二阶张量增量形式
            end
        end
    end
end
% 6.2 Calculate M
for k1=1:3;
    for k2=1:3;
        for k3=1:3;
            for k4=1:3;

               if trace(ninter*dd)>0;%判断加载卸载
                  M(k1,k2,k3,k4)=(pp^xx*mt+(1-pp^xx)*mr)*L(k1,k2,k3,k4)+pp^xx*(1-mt)*Lninter(
                  k1,k2)*ninter(k3,k4)+pp^xx*N(k1,k2)*ninter(k3,k4);
               else
                  M(k1,k2,k3,k4)=(pp^xx*mt+(1-pp^xx)*mr)*L(k1,k2,k3,k4)+pp^xx*(mr-mt)*Lninter(
                  k1,k2)*ninter(k3,k4);
               end

            end
        end
    end
end

% 7.Calculate the stress rate (tt)（应变控制）
% Strain Rate Tensor
dd=[dd11 0 0;0 dd33 0;0 0 dd33] %dd11=-1;dd33=0

for k1=1:3;
    for k2=1:3;
        tt(k1,k2)=0;
         for k3=1:3;
            for k4=1:3;
                tt(k1,k2)=tt(k1,k2)+M(k1,k2,k3,k4)*dd(k3,k4);
            end
        end
    end
end

% 9.Evolution of state variables
% 9.1.void ratio
ees=(1+es)*trace(dd);
% 9.2 Evolution of intergranular strain tensor
if trace(ninter*dd)>0;

for k1=1:3;
    for k2=1:3;
dinter(k1,k2)=0;
        for k3=1:3;
```

```
            for k4=1:3;
dinter(k1,k2)=dinter(k1,k2)+(i4(k1,k2,k3,k4)-ninter(k1,k2)*ninter(k3,k4)*pp^br)*dd(k3,k4);
            end
        end
    end
end

elseif trace(ninter*dd)<=0;
dinter=dd;

end

inter=inter+dinter*st;

% Accumulation and update
es=es+st*ees;
t=t+tt*st;
d=d+dd*st;

% Principal Stress
zt=eig(t);
zt11=zt(1);
zt22=zt(2);
zt33=zt(3);

p=-(zt11+zt22+zt33)/3;
q=-(zt11-zt33);%1/sqrt(2)*sqrt((zt11-zt22)^2+(zt22-zt33)^2+(zt11-zt33)^2);

% Principal Strain
zd=eig(d);
zd11=zd(1);
zd22=zd(2);
zd33=zd(3);

dv=zd11+zd22+zd33;
%sqrt(2)/3*sqrt((zd11-zd22)^2+(zd22-zd33)^2+(zd11-zd33)^2);
dy=-(d(1,1)-d(3,3));%1/sqrt(2)*sqrt((zd11-zd22)^2+(zd22-zd33)^2+(zd11-zd33)^2)

% 存入数组
allpt(i)=p;
allqt(i)=q;
alld(i)=-zd11;
alldy(i)=dy*100;
allforce(i)=-zt11/1000;
alles(i)=es;
alldv(i)=-dv;
allhs(i)=hs;
allpp(i)=pp;
alld11(i)=-d(1,1)*100;
alldd11(i)=-dd(1,1);
i=i+1;

end

figure(picturei)
%subplot(2,1,1);plot(alld11,allqt,'b');xlabel('剪应变');ylabel('q');hold on;%xlim([-1.5
1.5]);ylim([-80 80]);hold on
```

```
%subplot(2,1,2);plot(alld11,alldv,'b');xlabel('剪应变');;ylabel('volume');hold on;%xlim([0
220]);ylim([-80 80]);
subplot(2,1,1);plot(alld11,allforce,'r');grid on;xlabel('轴向应变');ylabel('轴向应力');hold on;
%subplot(2,1,2);plot(alldv,allpt,'r');grid on;xlabel('体应变');ylabel('平均应力');hold on;

%*******************************************************************************
figure(5)
plot(Time,DD11,'k.-');
%*******************************************************************************

%*******************************************************************************
function [dd11] = interpolation(data,time)
T = data(1:201,1);
D = data(1:201,2);
dd11 = interp1(T,D,time,'linear');
end
%********************************************************************************
```