

Two-step Exact Solution for Mohr-Coulomb Viscoplastic Consistency Model with Linear Hardening/Softening: a Proof with Maple Software

Timo Saksala

Department of Mechanics and Design, Tampere University of Technology, Tampere, P.O. Box 589, FI-33101, Finland

Abstract It has been observed numerically that the viscoplastic consistency model by Wang (1997) with a linear yield surface and a linear hardening/softening rule converges, using the standard stress return mapping, with two steps. In this paper this numerical observation is proved analytically using the Maple software. The proof is carried out for the Mohr-Coulomb viscoplastic consistency model. However, a similar proof can be performed for other linear models, such as the Rankine and Tresca models, as well.

Keywords Viscoplastic Consistency Model, Mohr-Coulomb Criterion, Stress Return Mapping, Exact Solution

1. Introduction

The classical viscoplasticity formulations, namely the Perzyna and Duvaut-Lions models, do not utilize the consistency condition[1]. Consequently, trial stress states violating the yield criterion are not returned to the yield surface defined by the yield criterion. In those models, the viscoplastic strain rate depends on the amount of violation[1]. In contrast, the consistency condition is imposed in the more recent viscoplastic consistency model by Wang[2]. Thereby, the trial stresses violating the yield function are returned to the yield surface as in the classical rate-independent plasticity. The major difference to the rate-independent plasticity is the dependence of the yield function on the rate of the internal variable(s). The basic components of the viscoplastic consistency model are[2]:

$$\begin{aligned} f_{vp} &= f(\boldsymbol{\sigma}, \kappa, \dot{\kappa}) \\ \dot{\boldsymbol{\epsilon}}^{vp} &= \dot{\lambda} \frac{\partial g_{vp}}{\partial \boldsymbol{\sigma}} \\ \dot{\kappa} &= \dot{\lambda} k(\boldsymbol{\sigma}, \kappa) \\ f_{vp} &\leq 0, \quad \dot{\lambda} \geq 0, \quad \dot{\lambda} f_{vp} = 0 \end{aligned} \quad (1)$$

where $\kappa, \dot{\kappa}$ are the internal variable and its rate, respectively. Moreover, g_{vp} is the viscoplastic potential, $\boldsymbol{\epsilon}^{vp}$ is the viscoplastic strain tensor (vector), and $\dot{\lambda}$ is the viscoplastic increment. The yield function f_{vp} can be called a dynamic yield function since it can expand and shrink in relation to

the loading-unloading conditions of Kuhn-Tucker form must hold.

As a consequence, the robust stress return algorithms of rate-independent computational plasticity can be employed with minor modifications concerning the presence of the rate of the internal variable(s). It is shown in this paper that a two-step exact solution exists for a viscoplasticity problem with a linear yield surface and a linear hardening/softening rule. This fact has been, of course, observed in numerical simulations but a formal proof is given here. The proof is performed for the Mohr-Coulomb (MC) viscoplasticity model with the Maple technical computing software. The MC model is chosen due to its wide usage in geotechnical engineering analyses.

2. Return Mapping Scheme for Viscoplastic Consistency Model

Stress integration, i.e. return mapping, algorithms for classical plasticity and viscoplasticity are considered, e.g. in[1,4], while algorithms for consistent viscoplasticity are developed in[2,4,5]. For present purposes, the stress integration algorithm by Wang et al.[2] is utilized. As all the terms involving the second or mixed derivatives with respect to stress or internal variables disappear due to the linearity (in stress) assumption, the algorithm therein simplifies into a form given below.

Return mapping algorithm for viscoplastic consistency model

Compute trial state: $\boldsymbol{\sigma}_{trial} = \mathbf{E}(\boldsymbol{\epsilon}_{t+\Delta t} - \boldsymbol{\epsilon}_t^{vp})$,

$$f_{vp}^{trial} = f(\boldsymbol{\sigma}_{trial}, \lambda_t, \dot{\lambda}_t)$$

* Corresponding author:

timo.saksala@tut.fi (Timo Saksala)

Published online at <http://journal.sapub.org/ajcam>

Copyright © 2011 Scientific & Academic Publishing. All Rights Reserved

its static position depending on the loading rate. Moreover,

Set: $\Delta\lambda_t^{(0)} = 0, \Delta\lambda_t^{(0)} = 0, \sigma_{\text{cor}}^{(0)} = \sigma_{\text{trial}}, \dot{\lambda}_t^{(0)} = 0, \dot{\lambda}_t^{(0)} = 0,$
 $\Delta\epsilon^{\text{vp},(0)} = \mathbf{0}$ and go to 1.

Local iteration:

1. $\delta\lambda = \frac{f_{\text{vp}}^{\text{trial}}}{(\partial_{\sigma} f_{\text{vp}})^T \mathbf{E} \partial_{\sigma} \mathbf{g}_{\text{vp}} - \partial_{\lambda} f_{\text{vp}} - \frac{1}{\Delta t} \partial_{\dot{\lambda}} f_{\text{vp}}}$
 2. $\Delta\lambda_t^{(i+1)} = \Delta\lambda_t^{(i)} + \delta\lambda$
 3. $\dot{\lambda}_t^{(i+1)} = \Delta\lambda_t^{(i+1)} / \Delta t$
 4. $\Delta\epsilon^{\text{vp},(i+1)} = \Delta\epsilon^{\text{vp},(i)} + \delta\lambda \partial_{\sigma} \mathbf{g}_{\text{vp}}$
 5. $\sigma_{\text{cor}}^{(i+1)} = \sigma_{\text{trial}} - \mathbf{E} \Delta\epsilon^{\text{vp},(i+1)}$
 6. $f_{\text{vp}}^{(i+1)} = f_{\text{vp}}(\sigma_{\text{cor}}^{(i+1)}, \lambda_t + \Delta\lambda_t^{(i+1)}, \dot{\lambda}_t^{(i+1)})$
- If $f_{\text{vp}}^{(i+1)} < \text{TOL}$ then go to Update.

Else set $i \leftarrow i + 1$ and go to 1.

Update: $\sigma_{t+\Delta t} = \sigma_{\text{cor}}^{(i+1)}, \epsilon_{t+\Delta t}^{\text{vp}} = \epsilon_t^{\text{vp}} + \Delta\epsilon^{\text{vp},(i+1)},$

$\lambda_{t+\Delta t} = \lambda_t + \Delta\lambda_t^{(i+1)}$ and exit.

In the above algorithm symbol ∂_x denotes the derivative with respect to a variable (scalar or vector) x . Moreover, \mathbf{E} denotes the elasticity matrix.

In fact, in this form the algorithm above is similar to the generalized cutting plane algorithm[3]. Therefore, it is not limited to linear yield surfaces but can be used for stress integration of nonlinear models as well.

3. A Two-Step Solution for the MC Viscoplasticity Problem

The MC yield criterion with its viscoplastic potential reads

$$\begin{aligned} f_{\text{MC}}(\sigma, \lambda_{\text{MC}}, \dot{\lambda}_{\text{MC}}) &= \sigma_1 - \sigma_3 + (\sigma_1 + \sigma_3) \sin \phi - \\ &\quad 2c(\lambda_{\text{MC}}, \dot{\lambda}_{\text{MC}}) \cos \phi \\ g_{\text{MC}}(\sigma) &= \sigma_1 - \sigma_3 + (\sigma_1 + \sigma_3) \sin \psi \end{aligned} \quad (2)$$

where σ_1, σ_3 are the major and minor principal stresses, ϕ and ψ are the internal friction and dilatation angle, respectively, and c is the cohesion of the material. The rate-dependent softening/hardening rule is assumed linear by

$$c = c_0 + h_{\text{MC}} \lambda_{\text{MC}} + s_{\text{MC}} \dot{\lambda}_{\text{MC}} \quad (3)$$

where c_0 is the initial (intact) value of cohesion while h_{MC} and s_{MC} are the softening/hardening and viscosity moduli, respectively. Due to its linearity, the gradients of the MC yield surface and plastic potential are constant vectors (having the angles ϕ and ψ as parameters).

Next, it is shown by Maple software that a two-step exact solution, using the algorithm above, exists for the MC viscoplasticity problem with the linear hardening/softening rule (3). The proof begins at the onset of viscoplasticity. Vector notation is used and only the relevant intermediate results of the Maple command executions are shown.

First time step: transition from elasticity to viscoplasticity, i.e. $\epsilon_t^{\text{vp}} = \mathbf{0}, \lambda_{\text{MC},t} = 0$, is treated as shown in Figure 1.

```
1> restart;
2> with(linalg):
3> Em:=evalm(E/((1+nu)*(1-2*nu))*matrix
  ([[1-nu,nu,nu],[nu,1-nu,nu],[nu,nu,1-nu]])):
4> f:=s1-s3+(s1+s3)*sin(phi)-2*c*cos(phi):
5> g:=s1-s3+(s1+s3)*sin(psi):
6> df:=vector([diff(f,s1),diff(f,s2),diff(f,s3)]):
7> dg:=vector([diff(g,s1),diff(g,s2),diff(g,s3)]):
8> lamda0:=0: Dlamda0:=0: Depsvp:=vector([0,0,0]):
9> dL:=f/(evalm(transpose(df)&* (evalm(Em &*
  dg)))+2*cos(phi)*(h_MC+s_MC/dt)):
10> Dlamda1:=Dlamda0+dL:
11> lamdarate:=Dlamda1/dt:
12> depsvp:=evalm(dL*dg):
13> Depsvp1:=evalm(Depsvp &+ depsvp):
14> signal:=evalm(vector([s1,s2,s3])-evalm(Em
  &* Depsvp1)):
15> c1:=c+h_MC*(lamda0+Dlamda1)+s_MC*lamdarate:
16> simplify(subs({s1=signal[1],s3=signal[3],c=c1},f));
```

Figure 1. Maple code for the first iteration involving a transition from elasticity to viscoplasticity

Thus, $f_{\text{MC}}(\sigma_{\text{cor}}^{(1)}, \Delta\lambda_{\text{MC}}^{(1)}, \dot{\lambda}_{\text{MC},t}^{(1)}) = 0$. Therefore, at the onset of viscoplasticity, a single step solution exists. Before continuing, the Maple script in Figure 1 is elaborated in some detail. After loading the linear algebra package in line 2, the linear elasticity matrix is defined in line 3. Lines 4-7 define the trial value of the MC yield function, the plastic potential, and their gradients determined with the `diff` command. In line 8 the initial values are set for the viscoplastic multiplier and strain. Then, the viscoplastic increment is solved in line 9 corresponding to step 1 in the algorithm. Lines 10 and 11 corresponds to steps 2 and 3 in the algorithm. Lines 12 and 13 do the same operations as step 4 in the algorithm. Line 14 corrects the stress according to step 5. Line 15 updates the value of cohesion and, finally, line 16 computes the new value of the MC yield function (step 6 in the algorithm).

In the next time step: $\epsilon_{t+\Delta t}^{\text{vp}} = \Delta\epsilon^{\text{vp},(1)}, \lambda_{\text{MC},t} = \Delta\lambda_{\text{MC}}^{(1)}$. This is a general time step involving softening/hardening and strain rate effects. The Maple code is shown in Figure 2.

```
17> Dlamda0:=0: Depsvp:=vector([0,0,0]):
18> lamda1:=lamda0+Dlamda1:
19> f2:=s1-s3+(s1+s3)*sin(phi)-2*c1*cos(phi):
20> dL:=f2/(evalm(transpose(df)&* (evalm(Em &*
  dg)))+2*cos(phi)*(h_MC+s_MC/dt)):
21> Dlamda1:=Dlamda0+dL:
22> lamdarate:=Dlamda1/dt:
23> depsvp:=evalm(dL*dg):
24> Depsvp1:=evalm(Depsvp &+ depsvp):
25> signal:=evalm(vector([s1,s2,s3])-evalm(Em &*
  Depsvp1)):
26> c1:=c+h_MC*(lamda1+Dlamda1)+s_MC*lamdarate:
27> f2new:=simplify(subs({s1=signal[1],s3=signal[3],
  c=c1},f));
```

Figure 2. Maple code for the first iteration involving viscoplasticity

In line 19 above s_1 and s_2 denote the trial principal stress components. Moreover, the Maple command in line 27 yields a result that can be written in form:

$$\begin{aligned} f_{2\text{new}} = \\ \frac{(1-\nu-2\nu^2)\cos\phi s_{\text{MC}}(\sigma_1-\sigma_3+\sin\phi(\sigma_1+\sigma_3)-2c\cos\phi)}{\Delta t E(1-2\nu+\sin\phi\sin\psi)+(1+\Delta t)\cos\phi h_{\text{MC}}(1-\nu-2\nu^2)} \end{aligned} \quad (4)$$

Thus, in the end of the first iteration the value of the yield

function is given by (4). Accordingly, the exact solution is obtained with the first step if the viscosity modulus s_{MC} is set to zero which is the rate-independent case. In the rate-dependent (viscous) case another step is needed:

```

28> dL:=f2new/(evalm(transpose(df) &* (evalm(Em
&*dg))) + 2*cos(phi)*(h_MC+s_MC/dt)):
29> Dlamda2:=Dlamda1+dL:
30> lamdarate:=Dlamda2/dt:
31> depsvp:=evalm(dL*dg):
32> Depsvp2:=evalm(Depsvp1 &+ depsvp):
33> sigma2:=evalm(vector([s1,s2,s3]) - evalm(Em
&* Depsvp2)):
34> c2:=c+h_MC*(lamda1+Dlamda2)+s_MC*lamdarate:
35>
f2newnew:=simplify(subs({s1=sigma2[1],s3=sigma2[3],
c=c2},f));

f2newnew :=0

```

Figure 3. Maple code for the second iteration involving viscoplasticity

Thus, $f_{MC}(\sigma_{cor}^{(2)}, \lambda_{MC,t} + \Delta\lambda_{MC,t}^{(2)}, \dot{\lambda}_{MC,t}^{(2)}) = 0$. Therefore, only two steps are required for the exact solution which was to be proven.

4. Conclusions

The purpose of this paper was to prove that a two-step exact solution exist for the rate-dependent softening continua modelled with the Mohr-Coulomb viscoplastic consistency model with linear hardening/softening rule. According to the proof with the Maple software, the two-step nature of the consistent viscoplasticity model is located in the invention of the viscosity parameter to the hardening/softening rule. On setting the viscosity modulus to zero, the classical rate-independent MC plasticity with a single-step exact solution is recovered.

Even though the proof was carried out with the MC model, it can be performed for all other models with linear yield surfaces, such as the Rankine and Tresca models, and linear hardening/softening rules.

Finally, it has been observed in numerical simulations that also the Drucker-Prager (DP) viscoplastic consistency model with linear hardening/softening rule has the same two-step nature with respect to convergence. This clearly stems from the fact that a closed form solution exists for the rate-independent DP plasticity problem. However, the proof is somewhat more complicated due to the nonlinearity of the DP criterion.

REFERENCES

- [1] Sluys L.J., Wave propagation, localisation and dispersion in softening solids, PhD Thesis, Delft University of Technology, 1992
- [2] Wang W.M., Sluys L.J., De Borst R., Viscoplasticity for instabilities due to strain softening and strain-rate softening, International Journal for Numerical Methods in Engineering 1997; 40: 3839-3864
- [3] Simo J.C., Hughes T.J.R., Computational inelasticity, Springer, 1998
- [4] Barpi F., Impact behaviour of concrete: a computational approach. Engineering Fracture Mechanics 2004; 71: 2197-2213
- [5] Winnicki A., Pearce C.J., Bićanić N., Viscoplastic Hoffman consistency model for concrete. Computers & Structures 2001; 79: 7-19