Implementation of a Material Model with Shear Rate and Temperature Dependent Viscosity

Vingaard, Mathias; Endelt, Benny Ørtoft; Christiansen, Jesper De C.

Published in:
Proceedings of 6th European LS-DYNA Users' Conference

Publication date:
2007

Document Version
Peer reviewed version

Implementation of a material model with shear rate and temperature dependent viscosity: Keyword file and FORTRAN code

Mathias Vingaard
This work was carried out at
Department of Production, Aalborg University
as part of a Ph.D. study.

October 9, 2010

Abstract

A user defined material with shear rate and, optionally, temperature dependent viscosity was implemented and validated in reference [1] from 6th European LS-DYNA Users' Conference, copyright © by Engineering Research AB (ERAB AB). Shear rate dependence was expressed with a Yasuda function and temperature dependence with an Arrhenius function. This document describes the keyword data blocks and the FORTRAN code for the user material model.

1 Introduction

The implementation of the user material model with variable viscosity, was carried out with the purpose of simulating polymer flow in LS-DYNA. In polymer flow, the viscosity often shows large variations due to shear thinning, cooling or heat dissipation. In LS-DYNA 971, the material models
MAT_ELASTIC_WITH_VISCOSITY and MAT_ALE_VISCOUS
are available, and they include temperature dependent and shear rate dependent viscosity, respectively. The ambition with the user material model was to take both shear and temperature dependency of a viscosity tensor into account.

2 Viscosity, shear rate and temperature

In the user defined material, the equations 1 to 8 are applied in the order of numbering. The normal strain rates are calculated as:

$$\dot{\varepsilon}_{ii} = \frac{\Delta\varepsilon_{ii}}{\Delta t} \tag{1}$$

The stored shear strains, γ, are halved to deviatoric strains, ε, in calculation of deviatoric strain rate [2]:

$$\dot{\varepsilon}_{ij} = \frac{1}{2}\dot{\gamma}_{ij} = \frac{\Delta\gamma_{ij}}{2\Delta t} \tag{2}$$

The viscosity tensor is dependent on the shear rate, and on the temperature if it is specified, i.e. the absolute temperature is above 273K. Shear rate dependency of the viscosity is expressed with a *Yasuda* function, also known as a *generalised Cross/Carreau* function [3, 4]:

$$\mu_{ij}(\dot{\gamma}_{ij}) = \mu_\infty + (\mu_0 - \mu_\infty)(1 + (k\ \dot{\gamma}_{ij})^a)^{(n-1)/a} \tag{3}$$

where

μ_∞	=	lower limit for viscosity at infinite shear rate,
μ_0	=	upper limit for viscosity at zero shear rate,
k	=	characteristic time,
a, n	=	material constants.

The *Carreau* model is a special case of equation 3, where the parameter, a, is fixed at the value of 2 rather than fitted to the current material. Other special cases of the Yasuda function is *Cross*, where $a = 1 - n$, and *modified Carreau*, where $a = 1$. If temperature dependence is included, it is expressed through the zero shear viscosity which is calculated from an Arrhenius expression, specifically known as the Arrhenius-Frenkel-Eyring formula [5]:

$$\mu_0(T) = A \exp\left(\frac{E_a}{R\ T}\right) \tag{4}$$

where

A	=	material constant,
E_a	=	activation energy,
R	=	universal gas constant,
T	=	temperature.

The function for temperature dependence of viscosity has some theoretical justification and is described as semi-empirical, where the functions for shear rate dependence must be considered empirical [5, 6]. Bulk viscosity depends on the viscosity, here average normal viscosity, according to Stokes hypothesis:

$$\mu_B = -\frac{2}{3}\frac{\mu_{kk}}{3} \tag{5}$$

Bulk viscosity can often be neglected, and in the user material model the value of it can be held within specified limits or set to zero. The viscous stress tensor is determined as:

$$\tau_{ij} = 2\mu_{ij}\dot{\varepsilon}_{ij} - \mu_B\dot{\varepsilon}_{kk}\delta_{ij} \tag{6}$$

Pressure is calculated from the hydrostatic stress from the previous time step and the volume strain increment from the current time step:

$$p = \frac{-\sigma_{kk}}{3} - B\ \varepsilon_{kk} \tag{7}$$

Total stress tensor:

$$\sigma_{ij} = \tau_{ij} - p\ \delta_{ij} \tag{8}$$

where

δ_{ij} = Kronecker delta,
Δt = time step,
B = bulk modulus.

3 Keyword file input blocks

Bulk modulus, B, is calculated from Young's modulus, E, and Poisson's ratio, ν from the relation; $B = E/(3(1-2\nu))$. Likewise shear modulus, G, is determined; $G = E/(2(1+\nu))$. In the material data shown here, density has been scaled to reduce computation times.

- `RO` = density,
- `YOUNG` = Young's modulus,
- `POISSON` = Poisson's ratio,
- `BULKMOD` = bulk modulus,
- `SHEARMOD` = shear modulus,
- `BULKVISC` = bulk viscosity,
- `CHTIME` = k in equation 3,
- `N-EXP` = n in equation 3,
- `A-EXP` = a in equation 3,
- `MUINF` = μ_∞ in equation 3,
- `MU0` = μ_0 in equation 3,
- `A` = A in equation 4 ,
- `ACT-E` = E_a in equation 4,
- `MUMAX` = upper numerical limit for viscosity,
- `BULKVMIN` = lower numerical limit for bulk viscosity.

3.1 Isothermal

In an isothermal simulation, the viscosity is calculated with equation 3 with the units *mm* and *MPa*. The data shown in the input block below is for polystyrene at 453 K written according the specifications in reference [3].

```
*MAT_USER_DEFINED_MATERIAL_MODELS
$      MID        RO        MT       LMC       NHV    IORTHO     IBULK        IG
         1     1E-05        43        16        50         0         3         4
$    IVECT     IFAIL    ITHERM    IHYPER      IEOS
         1         0         1         1
$    YOUNG   POISSON   BULKMOD  SHEARMOD
$    [MPa]       [-]     [MPa]     [MPa]
     2.4e3       0.3     2.0e3    9.23e2
$   CHTIME     N-EXP     A-EXP     MUINF       MU0              MUMAX  BULKVMIN
$      [s]       [-]       [-]   [MPa s]   [MPa s]            [MPa s]   [MPa s]
 5.18E+000 4.71E-001         2 7.00E-005 7.82E-002          7.82E-002         0
```

3.2 Thermal

In a thermal simulation the viscosity is calculated with equation 4 inserted in equation 3 with the units m and Pa. The data shown in the input block below is for polystyrene with a temperature from 453 K to 531 K.

```
*MAT_USER_DEFINED_MATERIAL_MODELS
$      MID        RO        MT       LMC       NHV    IORTHO     IBULK        IG
         1     1E+05        43        16        50         0         3         4
$    IVECT     IFAIL    ITHERM    IHYPER      IEOS
         1         0         1         1
$    YOUNG   POISSON   BULKMOD  SHEARMOD
$     [Pa]       [-]      [Pa]      [Pa]
     2.4e9       0.3     2.0e9    9.23e8
$   CHTIME     N-EXP     A-EXP     MUINF         A     ACT-E     MUMAX  BULKVMIN
$      [s]       [-]       [-]    [Pa s]    [Pa s]   [J/mol]    [Pa s]    [Pa s]
  5.968778  .4815149         2        70 7.573e-11 1.305e+05  7.82e+04 -5.21e+04
```

4 FORTRAN code

The user material model can only handle simulations with solid elements. Here the subroutine added to the FORTRAN file supplied by LSTC, `dyn21.f`, is printed:

```
c*************************************************************
      subroutine umat43 (cm,eps,sig,epsp,hsv,dt1,capa,etype,tt,
     1 temper,failel,crv)
c*************************************************************
c     cm(3)=BULKMOD
c     cm(09)= CHTIME
c     cm(10)= N-EXP
c     cm(11)= A-EXP
c     cm(12)= MUINF
c     cm(13)= MU0 (isothermal) or A (thermal)
c     cm(14)= ACT-E
c     cm(15)= MUMAX
c     cm(16)= BULKVMIN
c
c     eps(1)=local x  strain increment
c     eps(2)=local y  strain increment
c     eps(3)=local z  strain increment
c     eps(4)=local xy strain increment
c     eps(5)=local yz strain increment
c     eps(6)=local zx strain increment
c
c     sig(1)=total x  stress
c     sig(2)=total y  stress
c     sig(3)=total z  stress
```

```
c     sig(4)=total xy stress
c     sig(5)=total yz stress
c     sig(6)=total zx stress
c
c     dt1=current time step size
c     tt=current problem time
c     temper=current temperature
c
c     hsv(16..21)       = srg(mvi): Shear strain rate gradient
c     hsv(22..27)       = mu(mvi): Viscosity
c     hsv(28..33)       = dst(mvi): Deviatoric stress tensor
c     hsv(34)           = phyd: Pressure
c     hsv(35..40)       = sig(mvi): Total stress tensor
c
      include 'iounits.inc'
      character*5 etype
      dimension cm(*),eps(*),sig(*),hsv(*),crv(101,2,*)
      real dt1,temper,temp
      logical failel
c Local declacrations
      dimension sg(6), srg(6), dst(6)
      real Rgas, mu(6), mub, phyd
      integer kron, mvi
c Gas constant [J/(K*mol)]
      Rgas = 8.314510
c
ccccc Normal strain rate vector
      do 20 mvi=1,3
        if(dt1.le.0.or.eps(mvi).eq.0) then
        srg(mvi) = 0
        else
        srg(mvi) = eps(mvi)/dt1
        endif
        hsv(15+mvi)=srg(mvi)
20    continue
c
ccccc Deviatoric strain rate vector
      do 25 mvi=4,6
        if(dt1.le.0.or.eps(mvi).eq.0) then
        srg(mvi) = 0
        else
        srg(mvi) = 0.5 * eps(mvi)/dt1
        endif
        hsv(15+mvi)=srg(mvi)
25    continue
c
```

```
ccccc Viscosity vector
      do 30 mvi=1,6
c       thermal
        if (temper.ge.273) then
        mu(mvi) = cm(12) +
     1      ( cm(13) * exp(cm(14)/(Rgas * temper)) - cm(12)) /
     2      ( 1.0 + (cm(09) * abs(2*srg(mvi)) )**cm(11))
     3      **((1-cm(10))/cm(11))
c       isothermal
        else
        mu(mvi) = cm(12) +
     1      ( cm(13)  - cm(12) ) /
     2      ( 1.0 + (cm(09) * abs(2*srg(mvi)) )**cm(11))
     3      **((1-cm(10))/cm(11))
        endif
        if (mu(mvi).gt.cm(15)) then
        mu(mvi) = cm(15)
        endif
        hsv(21+mvi)=mu(mvi)
30    continue
c
ccccc Bulk viscosity
      mub = - 2.0/3.0 * (mu(1) + mu(2) + mu(3))/3.0
      if (mub.lt.cm(16)) then
      mub = cm(16)
      endif
c
ccccc Deviatoric stress vector
      do 40 mvi=1,6
        if (mvi.le.3) then
        kron=1
        else
        kron=0
        endif
        dst(mvi) = 2 * mu(mvi) * srg(mvi) -
     1           mub * (srg(1) + srg(2) + srg(3)) * kron
        hsv(27+mvi)=dst(mvi)
40    continue
c
ccc Pressure scalar
      phyd = -(sig(1) + sig(2) + sig(3))/3 -
     1        cm(3) * (eps(1) + eps(2) + eps(3))
      hsv(34)=phyd
c
ccccc Total stress vector
      do 70 mvi=1,6
```

```
        if (mvi.le.3) then
        kron=1
        else
        kron=0
        endif
      sig(mvi) = - phyd * kron + dst(mvi)
      hsv(34+mvi)=sig(mvi)
   70 continue
c
      return
      end
```

`dyn21.f` with the subroutine printed above, `umat43`, was compiled with Intel FORTRAN compiler.

5 Summary

With the purpose of modelling molten polymers, a user defined material was programmed with a viscosity depending on shear rate by and on temperature. The FORTRAN code was added to the file from LSTC and compiled to a version of LS-DYNA, where the material data was set in input blocks in a keyword file.

References

[1] M. Vingaard, B. Endelt and J.deC. Christiansen, "Implementation of a material model with shear rate and temperature dependent viscosity", 6th European LS-DYNA Users' Conference (copyright © by ERAB AB) Gothenburg, 2007. www.dynalook.com/european-conf-2007/implementation-of-a-material-model-with-shear-rate.pdf

[2] LS-DYNA Theory Manual, Livermore Software Technology Corporation, Livermore, CA, 2006.

[3] C.A. Hieber and H.H. Chiang, "Some correlations involving shear viscosity of polystyrene melts", Rheol Acta 28, 321.332, 1989.

[4] FIDAP 8 Theory Manual, Fluent Inc., Lebanon, NH, 1998.

[5] P.A. Tanguy, L. Choplin and P. Hurez, "Compensation effects in viscosity-temperature dependence of polymer melts", Polymer Engineering and Science 28(8), 529.533, 1988.

[6] C.-Y. Liu, J. He, R. Keunings and C. Bailly, "New linearized relation for the universal viscosity - temperature behavior of polymer melts", Macromolecules 39, 8867.8869, 2006.