Short documentation of subroutines (UMAT, UHARD) for gradient-enhanced damage modeling (Version 2)

Andreas Seupel (Andreas.Seupel@tu-freiberg.de)

June 8, 2018

Institute of Mechanics and Fluid Dynamics, TU Bergakademie Freiberg, Lampadiusstrasse 4, 09599 Freiberg, Germany

1 Changes to Version 1

- The source code is rewritten into Fortran fixed form to avoid compilation problems.
- Parallelization (multiple CPUs) with default settings of Abaqus 6.14 and Abaqus 2016 has been tested. Usage of the subroutines with multiple CPUs is possible.

2 Content and Technical Hints

We provide a minimal model (UHARD) and a full model (UMAT), which contain the formulation of gradient-enhanced ductile damage described within the paper [1]. The implementation is done in fixed form Fortran for Abaqus-version 6.14. However, the subroutines are also applicable for Abaqus 2016.

2.1 Compilation and Test Setting

As compiler, Fortran Intel(R) 64 version 11.0 is utilized. Simulations and testing have been carried out using Abaqus installations on a Linux64 system.

2.2 Files

The following files are attached:

- UHARD.f (Fortran subroutine for user-defined hardening.)
- UMAT.f (Fortran subroutine for user-defined material without element deletion. This has been used for simulations until initiation of an incipient crack and the convergence study.)

- UMAT_ELEMENT_EROSION.f (Fortran subroutine for user-defined material with element deletion. This has been used for the crack growth example.)
- SHEAR_BAND_SPECIMEN_UHARD.inp (Abaqus input-file containing the 2D-shear band specimen example with $b_{\mathrm{e}}/L = 1/4$ for UHARD-usage)
- SHEAR_BAND_SPECIMEN_UMAT.inp (Abaqus input-file containing the 2D-shear band specimen example with $b_{\mathrm{e}}/L = 1/4$ for UMAT-usage)
- DOUBLE_NOTCH_TENSILE_TEST.inp (Abaqus input-file containing the 2D double-notched tensile test with $b_{\mathrm{e}}/L = 1/2$ for UMAT-usage with element deletion)
- All input-files of the mesh-convergence study, global convergence study and numerical examples conducted in the corresponding paper [1]

2.3 Parallelization

The UMATs/UHARD have been tested using one and several CPUs in Abaqus 6.14 and Abaqus 2016. For both versions, identical results have been obtained using multiple or single CPUs. In Abaqus 6.14 no parallelization of the element operations is applied by default in case of thermo-mechanical simulations and even if the option "standard_parallel=all" is chosen. In contrast, parallelization of element operations *and* solver are utilized by default in Abaqus 2016, which therefore provides a possible better performance and speed-up.

2.4 Starting a Simulation

To start a simulation, the UHARD or UMAT-subroutine and the input-files have to be provided in the same folder. The following terminal command can be used to start the test example for UHARD on a local workstation with an Abaqus-installation:

```
[abqVersion] job=SHEAR_BAND_SPECIMEN_UHARD.inp user=UHARD.f
```

The UMAT-example is similarly executed:

```
[abqVersion] job=SHEAR_BAND_SPECIMEN_UMAT.inp user=UMAT.f
```

```
[abqVersion] job=DOUBLE_NOTCH_TENSILE_TEST.inp user=UMAT_ELEMENT_EROSION.f cpus=4
```

Please insert the appropriate alias for the Abaqus command instead of [abqVersion]. More information concerning the examples are given within the subsequent sections.

3 Application

Both implementations (UHARD, UMAT) of gradient-enhanced damage need the definition of a coupled temperature displacement load step in Abaqus. A steady-state heat conduction has to be applied. The experienced reader is encouraged to have a look into the provided input-files. A definition with automatic step-size control is given by:

```
*Step, name=Step-1, nlgeom=YES, inc=1000
*Coupled Temperature-displacement, creep=none, steady state
0.001, 1., 1e-12, 0.001
```

For the double-notched tensile test, a numerical damping is used. The step definition is therefore:

```
*Step, name=Step-1, nlgeom=YES, inc=8000
*Coupled Temperature-displacement, ..., stabilize, factor=0.0002,
allsdtol=0, continue=NO
0.001, 1., 1e-12, 0.001
```

3.1 Material definition of UHARD

In order to signal the usage of a user-defined hardening (UHARD), the material definition reads:

```
*Material, name=Material-1
*Conductivity
0.25,
*Density
7.85e-09,
*Depvar
6,
*Heat Generation
*Elastic
525., 0.3
*Plastic, hardening=USER, properties=4
1., 0.75, 0., 0.4
```

Six user-defined state variables are defined (*Depvar). Four properties of hardening/softening are specified in the *Plastic-modul. Default description of elasticity is utilized (*Elastic). The *Conductivity is related to the internal length L:

$$\text{*Conductivity} = L^2.$$

The *Heat Generation is needed to incorporate the HETVAL-subroutine. The properties and depvar definitions are summarized in Tab. 1. If five properties are declared, the calculation is interrupted at formation of an incipient crack ($D = D_\mathrm{c}$). Otherwise, the calculations goes on, but without element deletion due to the described problem with element deletion [1].

Table 1: Properties and Depvar (STATEV in post-processing) of UHARD-subroutine.

PROP No.	Symbol (as in paper)	Description
1	σ_0	initial yield stress
2	H	linear hardening modulus
3	ε_i	damage parameter, strain at damage initiation
4	ε_c	damage parameter, strain at failure
(5)	-	arbitrary, the calculation is interrupted at D_c
STATEV No.	Symbol (as in paper)	Description
1	r	source term (heat source)
2	κ	damage driving strain
3	D	damage variable
4	-	element deletion indicator (=0->deletion)
5	-	indicator for damage initiation (>0->initiation)
6	ε_eq	equivalent plastic strain

3.2 Material definition of UMAT

In order to signal the usage of a user-defined material routine (UMAT), the material definition reads for example:

```
*Material, name=Material-1
*Conductivity
0.0625,
*Density
7.85e-09,
*Depvar,delete=12
12,
*Heat Generation
*User Material, constants=15, unsymm
525., 0.3, 1., 2., 1., 0., 0., 0.0
0.0, 0.0, 0., 0.0, 0.4, 0., 1.
```

Twelve user-defined state variables have to be defined (*Depvar). Depvar number 12 is chosen to act as element deletion signal. For visualization reasons, the field output STATUS has to be declared (see input-file). The *Conductivity-value is again related to the internal length L (see above). The *Heat Generation is needed to incorporate the HETVAL-subroutine. Fiveteen material parameters have to be defined and an unsymmetric material tangent (key word unsymm) has to be stated. As strain hardening law, a more general implementation compared to the paper is applied:

$$R = \sigma_0 + H\varepsilon_\mathrm{eq} + R_\infty \left(1 - \exp\left(-a\varepsilon_\mathrm{eq}\right)\right). \quad (1)$$

For $R_\infty = 0$ the linear hardening is realized. The properties and depvar definitions are summarized in Tab. 2.

Table 2: Properties and Depvar (STATEV in post-processing) of UMAT-subroutine.

PROP No.	Symbol (as in paper and Eq. (1))	Description
1	E	Youngs-modulus
2	ν	Poissons-ratio
3	σ_0	initial yield stress
4	H	hardening modulus
5	N	free hardening exponent
6	R_∞	saturation stress
7	a	hardening exponent
8	C_1	Johnson-Cook-parameter
9	C_2	Johnson-Cook-parameter
10	C_3	Johnson-Cook-parameter
11	C_4	not used, set 0
12	-	not used, set 0
13	ε_c	damage parameter, strain at failure
14	DDS	indicator: stop at $D = D_\mathrm{c}$->set DDS=1
15	DAMAGE_CASE	=1->non-local case; =0->local case
STATEV No.	Symbol (as in paper)	Description
1	ε_eq	equivalent plastic strain
2	ε_eq	hardening variable
3	D	damage variable
4	-	indicator for damage initiation (>0->initiation)
5	$\varepsilon_\mathrm{i}^\mathrm{fix}$	strain at damage initiation
6	-	equivalent deviatoric strain
7	σ_eq	von Mises stress
8	κ	damage driving strain
9	h	stress triaxiality
10	r	source term (heat source)
11	$\frac{\mathrm{d}\, r}{\mathrm{d} \triangle \bar{\varepsilon}_\mathrm{d}}$	derivative of source term
12	-	element deletion indicator (=0->deletion)

3.3 Flowchart of the UMAT-subroutine

The UMAT-subroutine is divided into several sub-programs. The most important are:

- UMAT (main program): Organizes the elastic-plastic operator split, calculates updates of stress, internal variables, heat source and material tangent entries
- DERIVATIVES_NL/DERIVATIVES: Calculates residual functions and Jacobian-matrix for the non-local and local case
- PLASTICITY_MODUL_NL/PLASTICITY_MODUL: Solves the non-linear equation, returns updated stress and material tangent entries
- STRAIN_HARDENING: Definition of strain hardening rule and damage locus

3.4 Element deletion

The UMAT-subroutine UMAT_ELEMENT_EROSION.f can be used for crack growth simulations with element deletion. Three ingredients are needed:

1. Input-file: a Depvar number has to be defined, which signals material failure by the key word: delete=12 (see input-file)
2. Input-file: to visualize element deletion the field output variable STATUS has to be requested
3. Material properties (PROPS): the property No. 14 has to be set to 0. Setting the value to 1 stops the simulation when material failure is reached at an integration point

3.5 Error and warning messages of UMAT/UHARD

Error messages are written to the .dat and .msg-files. The simulation is directly aborted if the number of PROPS or STATEV is wrong. The UMAT sends also a kill signal in the case of plane stress state. Problems concerning convergence are also written to the .dat-file.

3.6 Control parameters

For the crack growth simulations, some control settings of Abaqus are redefined. Firstly, the number of cutbacks of time increment is increased (20 allowed). Secondly, the increase in time increment while convergence is increased. Additionally, the convergence control for the displacement and the temperature degree of freedom are changed according to the paper. Please see input-file, keyword *Control.

References

[1] A. Seupel, G. Hütter, and M. Kuna. An efficient FE-implementation of implicit gradient-enhanced damage models to simulate ductile failure. *Engineering Fracture Mechanics*, 199:41–60, 2018.