Best Practices

Modeling Techniques in Abaqus for LS-Dyna Material Models

3DEXPERIENCE®

Version 1.0 - 5/11/2022

Written by: Shashwat SINHA
Validated by: David Palmer
Edited by: Arati DESAI

SIMULIA Abaqus 2021

Executive Summary

Accurate material modeling is a key to realistic finite element (FE) simulations. Different material models are used for different classes of material, each aiming to capture the correct physical behavior.

A variety of commercial FE software packages are available in the market. Each may have equivalent models that represent the same material behavior. Use of a particular material model may be preferred due to legacy, ease of use, availability of data, and standard practice in a particular industry.

This document provides guidance on developing equivalent material modeling techniques in Abaqus for GISSMO and MAT_ARUP_ADHESIVE models in LS-Dyna. These models are popular in the industry but currently not available in Abaqus. GISSMO model is widely used in the automotive industry for crash simulations. This document describes the GISSMO model and its equivalent modeling approach in Abaqus. The MAT_ARUP_ADHESIVE material model is discussed briefly, only in terms of usage in Abaqus.

Abaqus is a unified finite element analysis (FEA) code, and is a part of the SIMULIA brand of DASSAULT SYSTEMES. SIMULIA also offers simulation products in other technology domains such as fluids, electromagnetics, and so on.

This Best Practices document is applicable to all releases from SIMULIA Abaqus 2021 and above.

Target audience: FE analysts, LS-Dyna users, Abaqus users

Table of contents

1. Overview

Abaqus offers an extensive library of material models to address a wide range of simulation needs. Nevertheless, a desired material model available in other commercial finite element (FE) codes may not be directly available in Abaqus. In this case, an equivalent approach needs to be developed within the Abaqus modeling framework to replicate the behavior of the model in question. A user subroutine or additional procedures may sometimes be needed to produce full equivalence.

Due to the differences in implementation and syntax, it may be difficult to identify the equivalent parameters and concepts used in Abaqus and other codes. This document bridges that gap for certain material models from the LS-Dyna commercial finite element code. This document also describes the equivalent techniques/approaches in Abaqus for the following LS-Dyna material models:

- GISSMO Model
- MAT_ARUP_ADHESIVE Model

2. GISSMO Model

The GISSMO material model (**G**eneralized **I**ncremental **S**tress **S**tate Dependent **Mo**del) is a popular model used in production simulations for crashworthiness. The model incrementally accumulates damage and accounts for material softening and final failure. The attractiveness of the model comes from its flexibility of application for a wide range of metals and material models. The damage prediction can include material instability and can be adjusted through different material parameters. Full details of the model are available in the LS-Dyna Manual, Vol II [1].

Abaqus provides a very general and comprehensive material modeling framework to include the effects of material damage and failure. In GISSMO model, the approach for damage and failure is slightly different than in Abaqus. The GISSMO model is not directly available in Abaqus, but the approach of GISSMO model can be replicated in Abaqus using its material modeling framework. The discussion here is strictly in regards to constructing an equivalent modeling approach for the GISSMO model in Abaqus.

2.1. Parameters

The GISSMO model has several parameters, some of which may not be critical for an equivalent material definition in Abaqus. The critical and relevant parameters are only those parameters that contain material/experimental data. All other parameters required for model adjustments, syntactical, or referential reasons in LS-Dyna are not considered in this document. Refer to the LS-Dyna Manual, Vol II [1] for full reference on material model parameters.

The parameters relevant for material definition are described in Table 2.1. The description of the parameters is taken directly from the LS-Dyna documentation.

Parameter Name	Description
LCSDG	Failure strain curve/table or function
ECRIT	Flag for determining point of material instability
DMGEXP	Exponent for nonlinear damage accumulation
DCRIT	Critical damage at the point of material instability
FADEXP	Exponent for damage-related stress fadeout
LCREDG	Load curve ID or Table ID defining element size dependent regularization factors for equivalent plastic strain to failure.

Table 2.1: GISSMO model parameters

LCSDG : refers to a table of equivalent plastic strain to failure and triaxiality, or a function of triaxiality and *Lode* parameter. This parameter is the critical input needed for the model. The damage accumulation is based on this parameter.

ECRIT : is a flag whose value determines how the point of instability is obtained in simulation. The flag can point to three different ways of determining the point of instability as explained in the next section.

LCREDG : refers to the table for mesh regularization. Regularization factors defined in the table mitigate the mesh sensitivity of the GISSMO model. Additional parameters, SHRF and BIAXF are available, which are reduction factors for regularization for shear and equi-biaxial stress cases, respectively. The values of SHRF and BIAXF generally lie between 0.0 and 1.0 where, 0.0 means full regularization and 1.0 means no regularization under shear (triaxiality = 0.0 for SHRF = 1.0) or biaxial tension (triaxiality = 2/3 for BIAXF = 1.0). Any other intermediate triaxiality follows a linear interpolation between triaxiality 0.0 and 1/3, and between triaxiality 1/3 and 2/3 [1]. Optimizing these parameters is an iterative task.

2.2. Formulation

Damage accumulation in the GISSMO model occurs as follows:

$$\Delta D = \frac{DMGEXP \times D^{\left(1-\frac{1}{DMGEXP}\right)}}{\varepsilon_f}\Delta\varepsilon_p \tag{2.1}$$

The model is based on incremental accumulation of damage based on incremental change in equivalent plastic strain. The table referred to by LCSDG supplies the ε_f values in the above formulation. The damage variable D is calculated continuously throughout the analysis. Once the value of D reaches the critical value (DCRIT), stress is coupled with damage and degraded according to the form:

$$\sigma = \tilde{\sigma}\left[1-\left(\frac{D-DCRIT}{1-DCRIT}\right)^{FADEXP}\right] \tag{2.2}$$

Where, $\tilde{\sigma}$ is undamaged stress.

DCRIT is obtained by one of the following three methods:

- Through the material instability curve given by the form:

$$\Delta F = \frac{DMGEXP}{\varepsilon_{p,loc}} F^{\left(1-\frac{1}{DMGEXP}\right)} \Delta\varepsilon_p \tag{2.3}$$

  Where, $\varepsilon_{p,loc}$ is the equivalent plastic strain at the point of instability. This condition corresponds to ECRIT < 0.

  When this curve is specified, DCRIT is the value of D when $F = 1$. F and D are calculated independently and simultaneously. After this point, stress is coupled with damage according to Equation 2.2.

- Through direct specification of equivalent plastic strain for ECRIT. This condition corresponds to ECRIT > 0.

  When the equivalent plastic strain reaches the value specified under ECRIT, DCRIT is the value of D at that point. After this point, stress is coupled with damage according to Equation 2.2.

- Through direct specification of the value of DCRIT. This condition corresponds to ECRIT = 0.

  When the value of D reaches the value of DCRIT, stress is coupled with damage according to Equation 2.2.

The previous two sections briefly describe the core concepts in the GISSMO model; it is essentially based on equivalent plastic strain and stress triaxiality. The fracture strain is the equivalent plastic strain at failure. While there are other parameters and options, that are not critical in understanding the main concepts and implementation of the model.

2.3. Equivalent Abaqus Approach

Unlike the approach taken in the GISSMO model, in Abaqus the concept of damage and failure of materials is implemented through two distinct phases: *Damage Initiation* and *Damage Evolution*. The DIEM (**D**amage **I**nitiation and **E**volution **M**odel) model in LS-Dyna corresponds directly to this concept. In Abaqus, during the damage initiation phase, the damage initiation variable accumulates in a manner similar to that of D in the GISSMO model. However, instead of going until failure, the accumulation of the damage initiation variable is considered only until the point where the material reaches the damage initiation criterion, corresponding to $\omega = 1$. The GISSMO and DIEM approaches depart from one another at this point. From this point, the damage evolution phase begins in Abaqus and is based on fracture energy concepts.

The three key concepts that apply in establishing the equivalence of the Abaqus approach to the GISSMO model are as follows:

- The damage initiation framework in Abaqus is utilized to capture the damage accumulation through failure curve in the GISSMO model shown in Equation 2.1. With a suitable damage initiation criterion, the failure strain curve of the GISSMO model can be directly specified in Abaqus to continuously accrue the damage initiation variable ω.

  The evolving damage initiation variable ω will be equivalent to D in Equation 2.1. (The built-in fracture based damage evolution concept is not used with this failure curve.)

- Variable $DCRIT$ in Equation 2.2 is obtained either through direct specification or by using another damage initiation criterion for the instability curve (as in Equation 2.3).

- A user subroutine is used to capture stress degradation after the point of instability. A relationship with the form shown in Equation 2.2 is coded. The degrading stress part in the stress-strain curve is analogous to the concept of damage evolution in Abaqus. The subroutine runs with the solver and the form is thus integrated in the overall damage framework.

2.3.1. Damage Initiation Criteria

Accumulation of the damage initiation variable in Abaqus can be done in two ways:

- Linear Damage Initiation

  By default, Abaqus accumulates the damage initiation variable in a linear fashion based on the increment of equivalent plastic strain, given by the form:

$$\omega_D = \int \frac{d\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_D^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl})} = 1 \tag{2.4}$$

- Nonlinear Damage Initiation

  For many scenarios, linear accumulation may not be suitable. To better match the experimental observations, nonlinear accumulation of the damage initiation variable can be used, given by the form:

$$\omega_D = \int n\left(\frac{\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_D^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl})}\right)^{n-1} \frac{d\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_D^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl})} = 1 \tag{2.5}$$

2.3.2. Damage Evolution

When the onset of material damage occurs, the material starts to degrade and the stress-strain relationship no longer accurately represents the material's behavior. Using a softening stress-strain relationship to capture material degradation introduces a strong mesh dependency; that is, the energy dissipated decreases with mesh refinement. To mitigate this effect, the softening response after damage initiation is characterized by a stress-displacement response rather than a stress-strain response based on Hillerborg's (1976) fracture energy proposal [3].

The implementation of the stress-displacement concept in a finite element model uses a characteristic length, *L*, associated with an integration point. The fracture energy is formulated as:

$$G_f = \int_{\bar{\varepsilon}_0^{pl}}^{\bar{\varepsilon}_f^{pl}} L\sigma_y \, d\bar{\varepsilon}^{pl} = \int_0^{\bar{u}_f^{pl}} \sigma_y \, d\bar{u}^{pl} \tag{2.6}$$

G_f is the energy required to open a unit area of crack, and is a material parameter. This expression introduces the definition of the equivalent plastic displacement $\bar{u}^{pl}$, as the fracture work conjugate of the yield stress after the onset of damage (work per unit area of the crack).

Specification of damage evolution in Abaqus can thus be based on either equivalent plastic displacement or fracture energy. Multiple options for each case are available according to the requirement.

For clarity, let the total damage in Abaqus be indicated by variable d.

$$\sigma = (1 - d)\bar{\sigma} \tag{2.7}$$

If multiple damage evolution criteria are in use, then by default, Abaqus picks up the criterion that has maximum degradation. However, a multiplicative criterion is also provided, where the effect of all degradations is taken into account. In this case, an intermediate variable, d_{mult}, is created and then the maximum of all damage variables including the intermediate variable is chosen:

$$\begin{gathered} d_{mult} = 1 - \prod (1 - d_k) \\ d = max\{d_{mult}, max(d_k)\} \end{gathered} \tag{2.8}$$

In the present approach, only one damage evolution criterion based on equivalent plastic displacement will be used.

2.3.3. Equivalence Methodology

To establish an Abaqus model that is equivalent to the GISSMO model, the damage framework in Abaqus is utilized, supplemented with a user subroutine. Table 2.2 shows the equations of the Abaqus damage framework used to achieve the equivalent material degradation effect of the GISSMO model.

GISSMO Model	Abaqus Equivalent Approach
$\Delta D = \frac{DMGEXP \times D^{\left(1-\frac{1}{DMGEXP}\right)}}{\varepsilon_f}\Delta\varepsilon_p$	$\omega_1 = \int n\left(\frac{\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_f^{pl}(\eta,\dot{\bar{\varepsilon}}^{pl})}\right)^{n-1}\frac{d\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_f^{pl}(\eta,\dot{\bar{\varepsilon}}^{pl})}$
$\Delta F = \frac{DMGEXP}{\varepsilon_{p,loc}}F^{\left(1-\frac{1}{DMGEXP}\right)}\Delta\varepsilon_p$	$\omega_2 = \int n\left(\frac{\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_{p,loc}^{pl}(\eta,\dot{\bar{\varepsilon}}^{pl})}\right)^{n-1}\frac{d\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_{p,loc}^{pl}(\eta,\dot{\bar{\varepsilon}}^{pl})}$
$\sigma = \tilde{\sigma}\left[1-\left(\frac{D-DCRIT}{1-DCRIT}\right)^{FADEXP}\right]$	$\sigma = \bar{\sigma}(1-d)$
$\left(\frac{D-DCRIT}{1-DCRIT}\right)^{FADEXP}$	$d\ (\omega_1,\omega_2)$
LCREDG Table	Regularization Table

Table 2.2: Equivalence Table

For nonlinear damage accumulation, Abaqus uses Equation 2.5 as shown in the first 2 rows of the right column of Table 2.2. In the first row, ω_1 is for accumulation of damage based on equivalent plastic strain at fracture. In the second row, ω_2 is for accumulation of damage based on equivalent plastic strain until the point of material instability. The Abaqus equivalent form for damage initiation is an integral expression, but the damage in the simulation is accrued incrementally.

For stress degradation, as in Equation 2.2 of the GISSMO model, Equation 2.7 is used in Abaqus as shown in the third row of Table 2.2. Variable d in Equation 2.7 for Abaqus can be calculated as per the expression in parentheses in Equation 2.2. This calculation is done in Abaqus through user subroutine VUSDFLD. This way, the equivalence of Equations 2.2 and 2.7 is established. The critical damage variable $DCRIT$ and the accumulated damage variable D (or ω_1) are continuously tracked in the subroutine for calculation of d.

As described in Section 2.3, the critical damage variable, $DCRIT$ is obtained by 3 different methods in GISSMO model. In Abaqus, $DCRIT$ is obtained as follows:

- ***When $DCRIT$ is directly specified in the GISSMO model (ECRIT = 0):***
  The value of $DCRIT$ is specified as data for an Abaqus keyword.
- ***When ECRIT is specified as a positive value in GISSMO model (ECRIT > 0):***
  The equivalent plastic strain at the point of material instability is specified as data for an Abaqus keyword. When the equivalent plastic strain during simulation reaches the value of ECRIT, the value of damage variable D (i.e., ω_1) at that instant is recorded as $DCRIT$.
- ***When the instability curve is specified in the GISSMO model (ECRIT < 0):***
  The instability curve is specified in Abaqus using another damage initiation criterion, for example, ω_2, that is different from the damage initiation criterion ω_1 used for the calculation of variable D. Both damage initiation criteria, ω_1 and ω_2, are handled simultaneously and independently in Abaqus during simulation. When the instability curve is specified, DCRIT is the value of D when $F = 1$ (that is, DCRIT is the value of

ω_1 when $\omega_2 = 1$). If D reaches a value of 1 before $F = 1$, then instant failure is assumed to occur.

Note that the instability curve is relevant only for plane stress applications. For bulk materials, the value of ECRIT is nearly constant for all values of stress triaxialities, and therefore a constant value of ECRIT is assumed in such cases.

To conform to the Abaqus damage framework, a plastic displacement and field variable based damage evolution is defined that drives the damage variable d. The field variable is continuously updated based on calculations of d in the VUSDFLD subroutine. The field variable returned from the subroutine directly corresponds to the damage variable d in Equation 2.7.

To activate damage evolution in Abaqus, at least one damage initiation criterion must be satisfied; that is, the value of a damage initiation variable must equal 1. Damage evolution is then driven by the field variable from the subroutine. To accomplish this, a third damage initiation criterion (ω_3) is defined with a very small value of equivalent plastic strain. This allows the damage initiation criterion to be satisfied $(\omega_3 = 1)$ as soon as plasticity ensues and gets damage evolution activated.

With three damage initiation criteria, a damage evolution definition, and a subroutine, an Abaqus equivalent to GISSMO is established. The Abaqus keywords are discussed in Section 2.3.5. “Keyword Interface”.

2.3.4. Mesh Regularization

As mentioned before, mesh sensitivity effects can be mitigated automatically when using built-in damage evolution in Abaqus. However, since the fracture energy concepts are not applied in the GISSMO approach, mesh-sensitivity remains. Mesh refinement studies are recommended and finer meshes are normally needed with this approach. With finer meshes, mesh regularization may not be needed. However, with larger element sizes mesh regularization is needed to have similar results for all mesh sizes. A table of *element size* vs *regularization factors* is needed for Abaqus to apply regularization. This table can be entered as part of the material model parameters.

Variation of fading exponential $(FADEXP)$ with element size is not considered.

2.3.5. Keyword Interface

The corresponding Abaqus keywords for the equivalent methodology in the previous section are described below. They are all defined under the ***MATERIAL** keyword. For more details, refer to SIMULIA User’s Guide for Abaqus Keywords [5].

The keywords are denoted in bold upper case letters, with their parameters in purple. Usage examples include the associated data lines. Comments start with `**`.

- ***DAMAGE INITIATION**, CRITERION=DUCTILE, LODE DEPENDENT, ACCUMULATION POWER=n

This keyword is used for specifying the damage/failure curve as in the GISSMO model, and is used to evaluate ω_1. Omit the LODE DEPENDENT parameter if Lode angle dependence is not to be considered. Each data line contains an equivalent plastic strain and stress triaxiality data pair. Through this keyword, damage $(\omega_1 = D)$ is continuously tracked. The failure curve must be defined in all cases.

ACCUMULATION POWER corresponds to $FADEXP$.

Usage example:

```
*DAMAGE INITIATION, CRITERION=DUCTILE, LODE DEPENDENT, ACCUMULATION
POWER=2

**Equiv Pl Strain, stress triaxiality, Lode angle, strain rate

0.1,  0.05
```

(Lode angle and strain rate dependence are optional)

- ***DAMAGE INITIATION,** CRITERION=MSFLD, DEFINITION=MSFLD

This keyword specifies the instability curve and is used to evaluate ω_2. This additional damage initiation definition works in parallel with the previous damage initiation definition for the damage curve. Each data line for this keyword contains an equivalent plastic strain and minor-to-major principal strain (α) ratio pair. The GISSMO model in LS-Dyna accepts data in the form of equivalent plastic strain and stress triaxiality (η). To convert η to α for use in this Abaqus keyword, the following form is used:

$$\alpha = \frac{\dot{\varepsilon}_{minor}}{\dot{\varepsilon}_{major}} = \frac{3\eta^2 - 2 + \eta\sqrt{3(4 - 9\eta^2)}}{2 - 6\eta^2} \qquad -\frac{2}{3} \leq \eta \leq \frac{2}{3} \tag{2.9}$$

Note that this expression is valid only for plane stress conditions, that are valid for sheet metal applications.

Usage example:

```
*DAMAGE INITIATION, CRITERION=MSFLD, DEFINITION=MSFLD

**Equiv Pl Strain, Minor/Major princ strain, strain rate

0.1, 0.05
```

(strain rate dependence is optional)

- ***DAMAGE INITIATION,** CRITERION=SHEAR, ks=0

This third damage initiation criterion, ω_3, is introduced to ensure that damage evolution becomes active during the simulation. Specifying a small value of equivalent plastic strain (e.g., 1.0e-6) as data for this keyword ensures that the damage criterion is met as soon as plasticity starts. After the damage criterion is met, damage evolution becomes active, and it is controlled by a field variable computed in the VUSDFLD subroutine. Recall that the primary purpose of the other two damage initiation keywords is to track the amount of damage ($\omega_1 = D$) and the point of material instability ($\omega_2 = F$) in the simulation.

Usage example:

```
*DAMAGE INITIATION, CRITERION=SHEAR, ks=0

1.0e-6
```

The data line for this keyword *should not be changed* as it is used to activate damage evolution. If it is changed, a corresponding change must be made in the data of the *DAMAGE EVOLUTION keyword.

- ***DAMAGE EVOLUTION,** TYPE=DISPLACEMENT, SOFTENING=TABULAR, DEPENDENCIES=1

  The damage evolution keyword defines a one-to-one relation between the damage variable and the field variable (FV1) calculated in VUSDFLD. The DEPENDENCIES parameter makes the damage variable field variable dependent. The field variable directly corresponds to variable d in Equation 2.7, thus establishing a direct correspondence between damage variable d and the field variable.

  Usage example:

```
*DAMAGE EVOLUTION, TYPE=DISPLACEMENT, SOFTENING=TABULAR, DEPENDENCIES=1

**Data for damage variable = 0.0

0.0,          0.0,          ,0.0

0.0,          1.0e-6,       ,0.0

**Data for damage variable = 1.0

0.0,          0.0,          ,1.0

1.0,          1.0e-6,       ,1.0
```

  Abaqus interpolates the values in between the data given on the data lines. The data defined in this keyword *should not be changed*. As previously explained, it captures evolution of damage as per the GISSMO model.

- ***USER DEFINED FIELD,** PROPERTIES=176

  This keyword invokes the user subroutine VUSDFLD. The PROPERTIES parameter defines the number of material properties defined on the data lines, which are then available in the subroutine.

  The first data line specifies 7 material parameters: DCRIT, ECRIT, FADEXP, stress triaxiality, field variable number, flag for failure curve, and flag for instability curve, in that order.

The regularization table is specified from the 2nd data line as shown below.

The PROPERTIES parameter must be equal to the number of data lines multiplied by 8.

Usage example:

```
*USER DEFINED FIELD, PROPERTIES=40

**DCRIT, ECRIT, FADEXP, STRESS-TRIAXIALITY, FV#, Flag_failure_curve,
Flag_instability_curve

0, 0, 2, 0.33, 1, 0, 0

**Regularization table (based on stress-triaxiality 0.33) in the **following
format

**elem size, regularization factor

0.1,   2.0

0.5,   1.075

1.0,   0.87

1.5,   0.78
```

The order of data specification *must be maintained,* as the damage calculations will be based on these values and their correct identification. The rules for data analysis are as follows:

- If DCRIT (first entry on first data line) is defined, then DCRIT will be considered as the point of material instability at which damage couples with stress
  - ECRIT and instability curve will be ignored
  - If the DCRIT field is left blank or defined as 0, it will not be considered
- If ECRIT (second entry on first data line) is defined, then damage at ECRIT will be considered as the point of material instability at which damage couples with stress
  - Instability curve will be ignored
- If both DCRIT and ECRIT fields are left blank or both are specified as 0, then the instability curve will determine the point of material instability at which damage couples with stress
  - If the damage curve is satisfied before the instability curve, instant failure will be assumed
- If no DCRIT, ECRIT or instability curve is specified, then no coupling of stress with damage will occur
  - In this case no element failure/removal will take place
- If the stress triaxiality in the first data line (4th entry) is zero or less than zero, no regularization will occur.
  - Interpolation for other stress-triaxiality values is carried out between (0, 1/3) and (1/3, 2/3).

        - If the stress triaxiality defined is outside the range of (0, 2/3), full regularization based on the defined regularization table is assumed. No interpolation based on stress triaxiality is done.
    - If the flag for the damage/failure curve (sixth entry on first data line) is specified as 0, it indicates that the failure curve has been specified for the ductile criterion.

      ```
      *DAMAGE INITIATION, CRITERION=DUCTILE
      ```

        - If a value other than zero is specified, it indicates that the failure curve has been specified for Hosford-Coulomb criterion. The Hosford-Coulomb criterion provides an analytical expression to define the failure strain as function of stress triaxiality and Lode parameter.

          ```
          *DAMAGE INITIATION, CRITERION=HC
          ```

    - If the flag for the instability curve (seventh entry on first data line) is specified as 0, it indicates that the instability curve has been specified for MSFLD criterion.

      ```
      *DAMAGE INITIATION, CRITERION=MSFLD
      ```

        - If value other than zero is specified, it indicates that the instability curve has been specified using the ductile criterion.

          ```
          *DAMAGE INITIATION, CRITERION=Ductile
          ```

- ***DEPVAR**

  This keyword is used in conjunction with the *USER DEFINED FIELD to define the number of variables that can be continuously updated (SDVs) and reported. SDVs can be visualized in Abaqus/Viewer.

  Usage example:

  ```
  *DEPVAR
  6
  1, DAMAGE, “Damage variable”
  2, DUCTCRT, “Ductile criterion”
  3, INSTCRT, “Instability criterion”
  4, DCRIT, “DCRIT”
  5, REG, “Regularization factor”
  6, TRIAX, “Stress-triaxiality”
  ```

  The data on this keyword *must not be changed*.

2.3.6. User Subroutine

User subroutine VUSDFLD is utilized to capture the form of Equation 2.2 and calculate the damage variable d. This variable is returned as field variable and is used for damage evolution. The logic for determining the critical value of damage (as described under the *USER DEFINED FIELD section) is built into the subroutine.

The subroutine has direct access to all data defined under the *USER DEFINED FIELD keyword. In addition, the subroutine also accesses output variables of damage, equivalent plastic strain and stress triaxiality to perform the necessary calculations. An example of output request in the input file is shown below:

```
*ELEMENT OUTPUT, DIRECTIONS=YES

DMICRT, LE, PE, PEEQ, PEEQMAX, S, SDV, FV,
STATUS, SDEG, TRIAX
```

A compiled Fortran subroutine for VUSDFLD and the input file for the example model will be provided; contact your sales representative to obtain the subroutine and the model.

The subroutine is not discussed further. Refer to SIMULIA User's Guide for Abaqus Subroutines [4] for further details on its structure and syntax.

2.3.7. Example

A simple tensile test model is created to demonstrate the methodology. The model presented here is only for demonstration purpose. The test specimen in the model is a simple dog-bone shaped geometry used commonly to perform tensile tests. It is a plane stress model defined with shell elements.

- Model Description

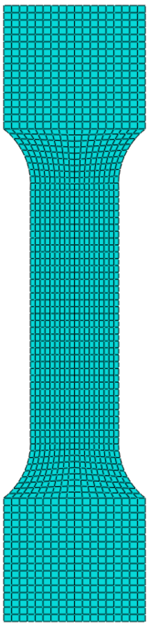
Test specimen	Model details
	Geometry details: Specimen length: 174.64 mm Specimen thickness: 1.2 mm Flange length: 35 mm Flange width: 40 mm Web length: 70mm Web width: 25 mm Radius of transition: 20 mm FE details: Number of nodes: 1729 Number of elements: 1620 Global element size: 2 mm Element type: S4R (general purpose shell element) Element topology: Quadrilateral Element order: Linear

Table 2.3: Example model details

- Material Properties

The material data is from [2].

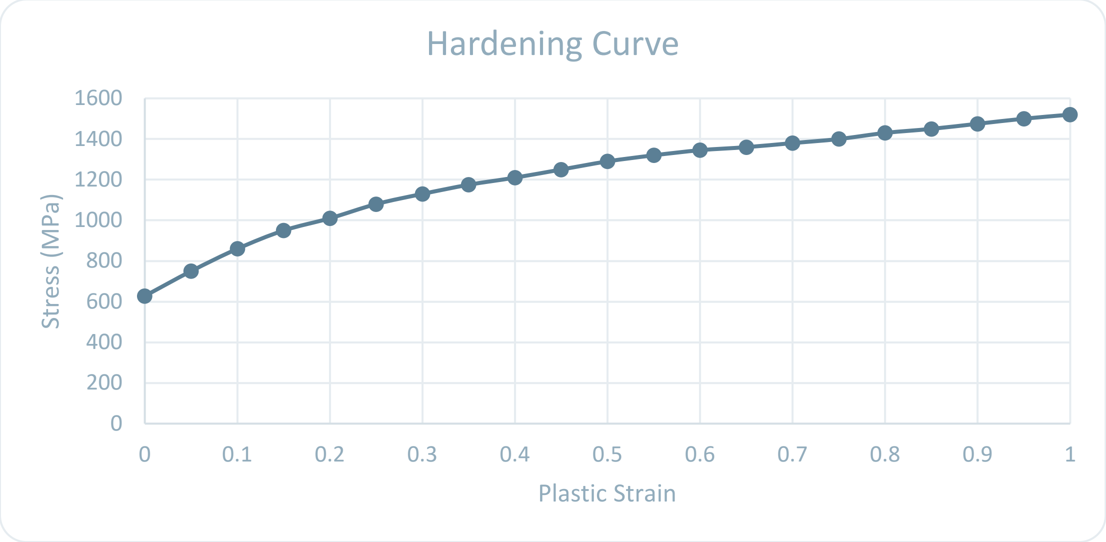


Fig 2.1: True stress-strain curve

Assuming 1% strain offset, Young's Modulus (E) = 62800 MPa

Assumed Poisson's Ratio (v) = 0.29

Assumed density (ρ) = 8.0 x 10^{-9} tonne/mm^3

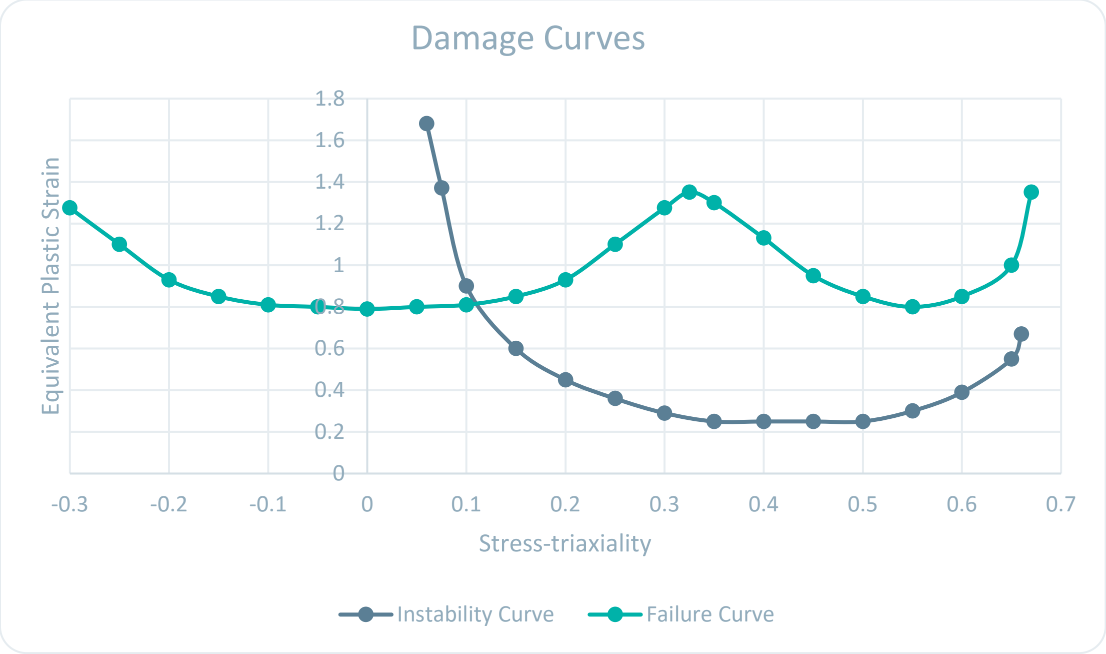


Fig 2.2: Curves for damage characterization

To convert stress triaxiality (η) to ratio of minor to major principal strain (α) for the instability curve, Equation 2.9 is used. The converted data of α along with equivalent plastic strain is then used for the Abaqus keyword (using `criterion=MSFLD`).

Material definition through input file keywords is as follows:

```
*Material, name=Gissmo

*Elastic
62800., 0.29

*Density
8.0e-9

*Plastic
 628.,   0.
 750., 0.05
 860.,  0.1
 950., 0.15
1010.,  0.2
1080., 0.25
1130.,  0.3
1175., 0.35
1210.,  0.4
1250., 0.45
1290.,  0.5
1320., 0.55
1345.,  0.6
1360., 0.65
1380.,  0.7
1400., 0.75
1430.,  0.8
1450., 0.85
1475.,  0.9
1500., 0.95
1520.,   1.

*Damage Initiation, criterion=DUCTILE, accumulation power=2
 1.275,  -0.3,    0.
   1.1, -0.25,    0.
  0.93,  -0.2,    0.
  0.85, -0.15,    0.
  0.81,  -0.1,    0.
   0.8, -0.05,    0.
  0.79,    0.,    0.
   0.8,  0.05,    0.
  0.81,   0.1,    0.
  0.85,  0.15,    0.
  0.93,   0.2,    0.
   1.1,  0.25,    0.
 1.275,   0.3,    0.
  1.35, 0.325,    0.
   1.3,  0.35,    0.
  1.13,   0.4,    0.
  0.95,  0.45,    0.
  0.85,   0.5,    0.
   0.8,  0.55,    0.
  0.85,   0.6,    0.
    1.,  0.65,    0.
  1.35,  0.67,    0.

*Damage Initiation, criterion=MSFLD, definition=MSFLD
 1.68, -0.900828,        0.
 1.37, -0.877288,        0.
  0.9, -0.838922,        0.
  0.6, -0.764723,        0.
 0.45, -0.692666,        0.
 0.36, -0.621337,        0.
 0.29, -0.549272,        0.
 0.25, -0.474777,        0.
 0.25, -0.395661,        0.
 0.25, -0.308734,        0.
 0.25, -0.208712,        0.
  0.3, -0.085284,        0.
 0.39, 0.0876242,        0.
 0.55,   0.43396,        0.
 0.67,  0.604102,        0.

*Damage Initiation, criterion=SHEAR
 1e-06,0.,0.

*Damage Evolution, type=DISPLACEMENT, dependencies=1, softening=TABULAR
0.,    0., ,    0.
0., 1e-06, ,    0.
0.,    0., ,    1.
1., 1e-06, ,    1.

*User Defined Field, Properties=176
0,0,2,0.33,1,0,0
**Regularization factors
0.1,  2.0
0.5,  1.075
1.0,  0.87
1.5,  0.78
2.0,  0.73
2.5,  0.69
3.0,  0.66
3.5,  0.64
4.0,  0.62
4.5,  0.60
5.0,  0.585
5.5,  0.57
6.0,  0.56
6.5,  0.55
7.0,  0.54
7.5,  0.53
8.0,  0.52
8.5,  0.51
9.0,  0.50
9.5,  0.49
10.0, 0.485

*Depvar
6,
1, DAMAGE, "Damage variable, d"
2, DUCTCRT, "Ductile criterion"
3, INSTCRT, "Instability criterion"
4, DCRIT, "DCRIT"
5, REG, "Regularization factor"
6, TRIAX, "Stress-triaxiality"
```

- Boundary Conditions

  The test specimen model was constrained to prevent any rigid body motion. The bottom edge was constrained and the top edge was applied a displacement of 50 mm through an amplitude curve.

- Procedure

  Dynamic Explicit procedure with a step time of 0.05 s was used for the analysis. This ensured a quasi-static response. Explicit procedure also allows for failure of material and removal of failed elements.

- Results

  The results of the model are shown in Fig 2.3. The specimen failed as expected.

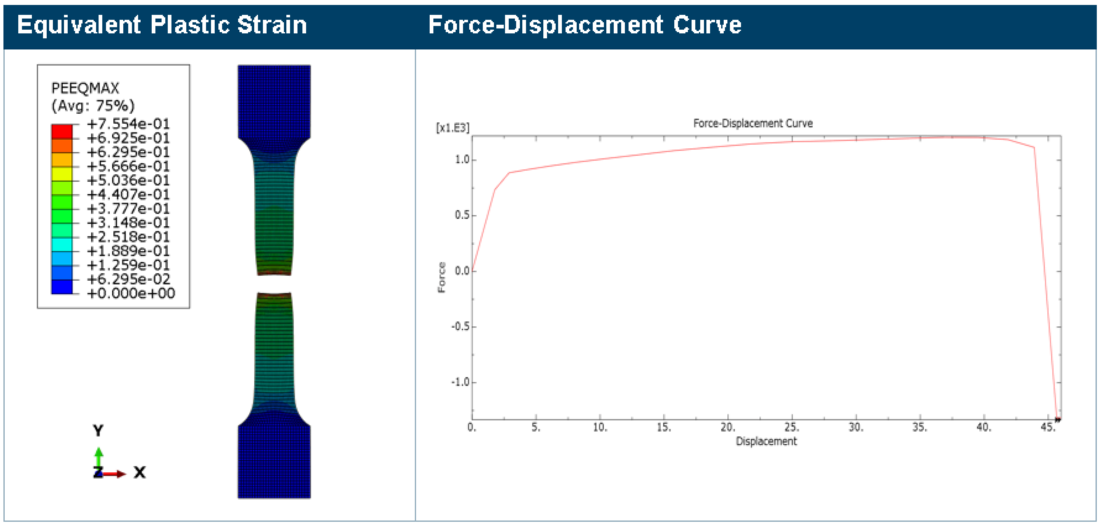


Fig 2.3: Example specimen results

2.4. Conclusion

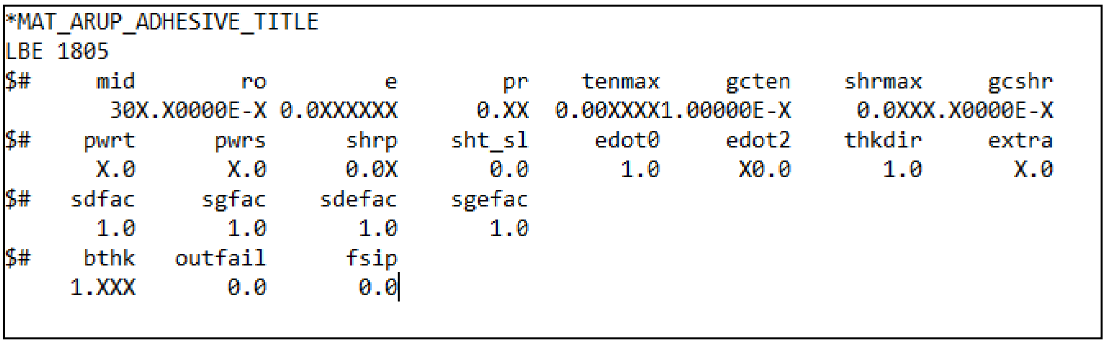
A method to replicate the behavior of the LS-Dyna GISSMO model in Abaqus/Explicit is presented. There are a few minor differences in the approach from LS-Dyna (i.e., no variation of $FADEXP$ with element size; no use of SHRF or BIAXF; no higher order point of material instability calculation) but the main concepts are captured in this approach. You are advised to carefully analyze all aspects of this equivalent approach for GISSMO model before using it. User is also encouraged to use Abaqus' *Damage Initiation and Damage Evolution* approach and compare results. Experimental data must be used to verify simulation results.

3. MAT_ARUP_ADHESIVE Model

This material model is in adhesive bonding for aluminum materials. The material failure is based on traction-separation response, where a pseudo yield surface is used for damage initiation [1]. Damage evolution is based on fracture energy as the interface fails.

3.1.   Usage of Model

The LS-Dyna keyword interface for the material definition is as shown in Fig 3.1.



```
*MAT_ARUP_ADHESIVE_TITLE
LBE 1805
$#       mid        ro         e        pr    tenmax     gcten    shrmax     gcshr
        30X.X0000E-X 0.0XXXXXX      0.XX 0.00XXXX1.00000E-X  0.0XXX.X0000E-X
$#      pwrt      pwrs      shrp    sht_sl     edot0     edot2    thkdir     extra
         X.0       X.0      0.0X       0.0       1.0      X0.0       1.0       X.0
$#     sdfac     sgfac    sdefac    sgefac
         1.0       1.0       1.0       1.0
$#      bthk   outfail      fsip
       1.XXX       0.0       0.0
```

Fig 3.1: Keyword interface in LS-Dyna

3.2.   Equivalent Approach in Abaqus

The parameters used in LS-Dyna directly correspond to parameters used in a traction-separation material definition in Abaqus. Thus, a direct equivalence is established. Table 3.2 describes the LS-Dyna parameters in the Abaqus framework.

Dyna Parameters	Abaqus Equivalent Description
Mid	N/A
Ro	Mass density
E	Young’s Modulus
PR	Poisson’s Ratio (N/A)
Tenmax	Nmax
Gcten	GIC
Shrmax	Smax
Gcshr	GIIC
Pwrt	“power” parameter on *Damage Evolution
Pwrs	“power” parameter on *Damage Evolution
Shrp	N/A
sht_sl	N/A
edot0/edot2	Rate; “rate dependent” parameter on *Damage Initiation
Thkdir	“stack direction” parameter on *Cohesive Section

Extra	N/A
Sdfac	N/A
Sgfac	N/A
Sdefac	N/A
Sgefac	N/A
Bthk	Constitutive thickness; “thickness” parameter on *Cohesive Section; use thickness = specified
Outfail	N/A
Fsip	N/A

Table 3.2: LS-Dyna material parameters

3.2.1. Keyword Interface

LS-Dyna parameters are shown below in < > brackets where they are used with Abaqus keywords for cohesive material definition for failure. Refer to Abaqus Keywords Users Guide [5] for more details.

```
*COHESIVE SECTION, elset=cohesion, material=adhesive,
response=traction separation, thickness=<specified>,
stack direction=<thkdir>, controls=cont
<bthk>

*SECTION CONTROLS, name=cont, element deletion=yes

*MATERIAL, name=adhesive

*ELASTIC, type=traction
<e>, <e>, <e>

*DAMAGE INITIATION, criterion=quads, rate dependent
<tenmax>, <shrmax>, <shrmax>, <edot2>

*DAMAGE EVOLUTION, type=energy, mixed mode behavior=power law,
power=<pwrt>
<gcten>, <gcshr>, <gcshr>

*DENSITY
<ro>
```

4. References

[1] LS-Dyna User’s Keyword Manual, Vol II, R11.

[2] Chen, G., Huang, L., Link, T.M., Tyan, T. et al., “Calibration and Validation of GISSMO Damage Model for A 780-MPa Third Generation Advanced High Strength Steel,” SAE Technical Paper 2020-01-0198, 2020, doi:10.4271/2020-01-0198.

[3] SIMULIA User Assistance – 2021 | Abaqus | Materials

[4] SIMULIA User Assistance – 2021 | Abaqus | User Subroutines

[5] SIMULIA User Assistance – 2021 | Abaqus | Keywords

5. Appendix 1

5.1. GISSMO Material Model Keywords

The keywords used for material definition are shown below. The highlighted keywords and data lines should not be changed.

```
*Material, name=Gissmo
*Elastic
<E>, <ν>
*Density
<ρ>
*Plastic
<σ^y>, <ε^p>
*Damage Initiation, criterion=DUCTILE, accumulation power=2
<ε^eq pl>, <η>
*Damage Initiation, criterion=HC, accumulation power=2
<a>, <b>, <c>, <n>, <d>, <strain rate>
*Damage Initiation, criterion=MSFLD, definition=MSFLD
<ε^eq pl>, <α>
*Damage Initiation, criterion=SHEAR
 1e-06
*Damage Evolution, type=DISPLACEMENT, dependencies=1,
softening=TABULAR
0.0,   0.00, ,0.0
0.0, 1.0e-6, ,0.0
**
0.0,   0.00, ,1.0
1.0, 1.0e-6, ,1.0
*User Defined Field, Properties=176
<DCRIT>, <ECRIT>, <FADEXP>, <STRESS TRIAXIALITY>, <Field Variable#>,
<Flag_dmg>, <Flag_inst>
<Elem_size>, <Regularization factor>
*Depvar
6,
1, DAMAGE, "Damage variable, d"
2, DUCTCRT, "Ductile criterion"
3, INSTCRT, "Instability criterion"
4, DCRIT, "DCRIT"
5, REG, "Regularization factor"
6, TRIAX, "Stress-triaxiality"
**
```

5.1.1. Case 1: With Ductile and MSFLD Criteria

In this case, “Ductile” damage initiation criterion is used to specify the failure curve, and the “MSFLD” damage initiation criterion is used to define the instability curve. Note that for solid elements, the MSFLD criterion is ignored.

```
*Material, name=Gissmo
*Elastic
62800., 0.29
*Density
8.0e-9
*Plastic
 628.,   0.
 750., 0.05
 860.,  0.1
 950., 0.15
1010.,  0.2
1080., 0.25
1130.,  0.3
1175., 0.35
1210.,  0.4
1250., 0.45
1290.,  0.5
1320., 0.55
1345.,  0.6
1360., 0.65
1380.,  0.7
1400., 0.75
1430.,  0.8
1450., 0.85
1475.,  0.9
1500., 0.95
1520.,   1.
*Damage Initiation, criterion=DUCTILE, accumulation power=2
 1.275,  -0.3,    0.
   1.1, -0.25,    0.
  0.93,  -0.2,    0.
  0.85, -0.15,    0.
  0.81,  -0.1,    0.
   0.8, -0.05,    0.
  0.79,    0.,    0.
   0.8,  0.05,    0.
  0.81,   0.1,    0.
  0.85,  0.15,    0.
  0.93,   0.2,    0.
   1.1,  0.25,    0.
 1.275,   0.3,    0.
  1.35, 0.325,    0.
   1.3,  0.35,    0.
  1.13,   0.4,    0.
  0.95,  0.45,    0.
  0.85,   0.5,    0.
   0.8,  0.55,    0.
  0.85,   0.6,    0.
    1.,  0.65,    0.
  1.35,  0.67,    0.
*Damage Initiation, criterion=MSFLD, definition=MSFLD
```

```
 1.68, -0.900828,        0.
 1.37, -0.877288,        0.
  0.9, -0.838922,        0.
  0.6, -0.764723,        0.
 0.45, -0.692666,        0.
 0.36, -0.621337,        0.
 0.29, -0.549272,        0.
 0.25, -0.474777,        0.
 0.25, -0.395661,        0.
 0.25, -0.308734,        0.
 0.25, -0.208712,        0.
  0.3, -0.085284,        0.
 0.39, 0.0876242,        0.
 0.55,   0.43396,        0.
 0.67,  0.604102,        0.
*Damage Initiation, criterion=SHEAR
 1e-06
*Damage Evolution, type=DISPLACEMENT, dependencies=1, softening=TABULAR
0.0,   0.00, ,0.0
0.0, 1.0e-6, ,0.0
**
0.0,   0.00, ,1.0
1.0, 1.0e-6, ,1.0
*User Defined Field, Properties=176
**DCRIT, ECRIT, FADEXP, STRESS TRIAXIALITY, FV#, Flag_dmg, Flag_inst
0,0,2,0.33,1,0,0
** Regularization factors (elem size, factor; based on triaxiality 0.33)
0.1,   2.0
0.5,   1.075
1.0,   0.87
1.5,   0.78
2.0,   0.73
2.5,   0.69
3.0,   0.66
3.5,   0.64
4.0,   0.62
4.5,   0.60
5.0,   0.585
5.5,   0.57
6.0,   0.56
6.5,   0.55
7.0,   0.54
7.5,   0.53
8.0,   0.52
8.5,   0.51
9.0,   0.50
9.5,   0.49
10.0,  0.485
*Depvar
6,
1, DAMAGE, "Damage variable, d"
2, DUCTCRT, "Ductile criterion"
3, INSTCRT, "Instability criterion"
4, DCRIT, "DCRIT"
5, REG, "Regularization factor"
6, TRIAX, "Stress-triaxiality"
**
```

5.1.2. Case 2: With HC and Ductile Criteria

In this case, “HC” damage initiation criterion is used to specify the failure curve, and the “MSFLD” damage initiation criterion is used to specify the instability curve.

```
*Material, name=Gissmo
*Elastic
62800., 0.29
*Density
8.0e-9
*Plastic
 628.,   0.
 750., 0.05
 860.,  0.1
 950., 0.15
1010.,  0.2
1080., 0.25
1130.,  0.3
1175., 0.35
1210.,  0.4
1250., 0.45
1290.,  0.5
1320., 0.55
1345.,  0.6
1360., 0.65
1380.,  0.7
1400., 0.75
1430.,  0.8
1450., 0.85
1475.,  0.9
1500., 0.95
1520.,   1.
*Damage Initiation, criterion=HC, accumulation power=2
 1.03, 1.4, 0.0016, 0.232, 0., 1000.
*Damage Initiation, criterion=DUCTILE, accumulation power=2
 1.68, 0.06,  0.
 1.37, 0.075, 0.
  0.9, 0.1,   0.
  0.6, 0.15,  0.
 0.45, 0.2,   0.
 0.36, 0.25,  0.
 0.29, 0.3,   0.
 0.25, 0.35,  0.
 0.25, 0.4,   0.
 0.25, 0.45,  0.
 0.25, 0.5,   0.
  0.3, 0.55,  0.
 0.39, 0.6,   0.
 0.55, 0.65,  0.
 0.67, 0.66,  0.
*Damage Initiation, criterion=SHEAR
 1e-06
*Damage Evolution, type=DISPLACEMENT, dependencies=1, softening=TABULAR
0.0,   0.00, ,0.0
0.0, 1.0e-6, ,0.0
**
0.0,   0.00, ,1.0
```

```
1.0, 1.0e-6, ,1.0
*User Defined Field, Properties=176
**DCRIT, ECRIT, FADEXP, STRESS TRIAXIALITY, FV#, Flag_dmg, Flag_inst
0,0,2,0.33,1,0,0
** Regularization factors (elem size, factor; based on triaxiality 0.33)
0.1,   2.0
0.5,   1.075
1.0,   0.87
1.5,   0.78
2.0,   0.73
2.5,   0.69
3.0,   0.66
3.5,   0.64
4.0,   0.62
4.5,   0.60
5.0,   0.585
5.5,   0.57
6.0,   0.56
6.5,   0.55
7.0,   0.54
7.5,   0.53
8.0,   0.52
8.5,   0.51
9.0,   0.50
9.5,   0.49
10.0,  0.485
*Depvar
6,
1, DAMAGE, "Damage variable, d"
2, DUCTCRT, "Ductile criterion"
3, INSTCRT, "Instability criterion"
4, DCRIT, "DCRIT"
5, REG, "Regularization factor"
6, TRIAX, "Stress-triaxiality"
**
```

6. Document History

Document Revision	Date	Revised By	Changes/Notes
1.0	05/11/2021	Shashwat Sinha	Original