



## Start Writing a Subroutine in Abaqus: Basics and Recommendations

The user subroutines in Abaqus allow the program to be customized for particular applications, which are not available through main ... Continue reading



3

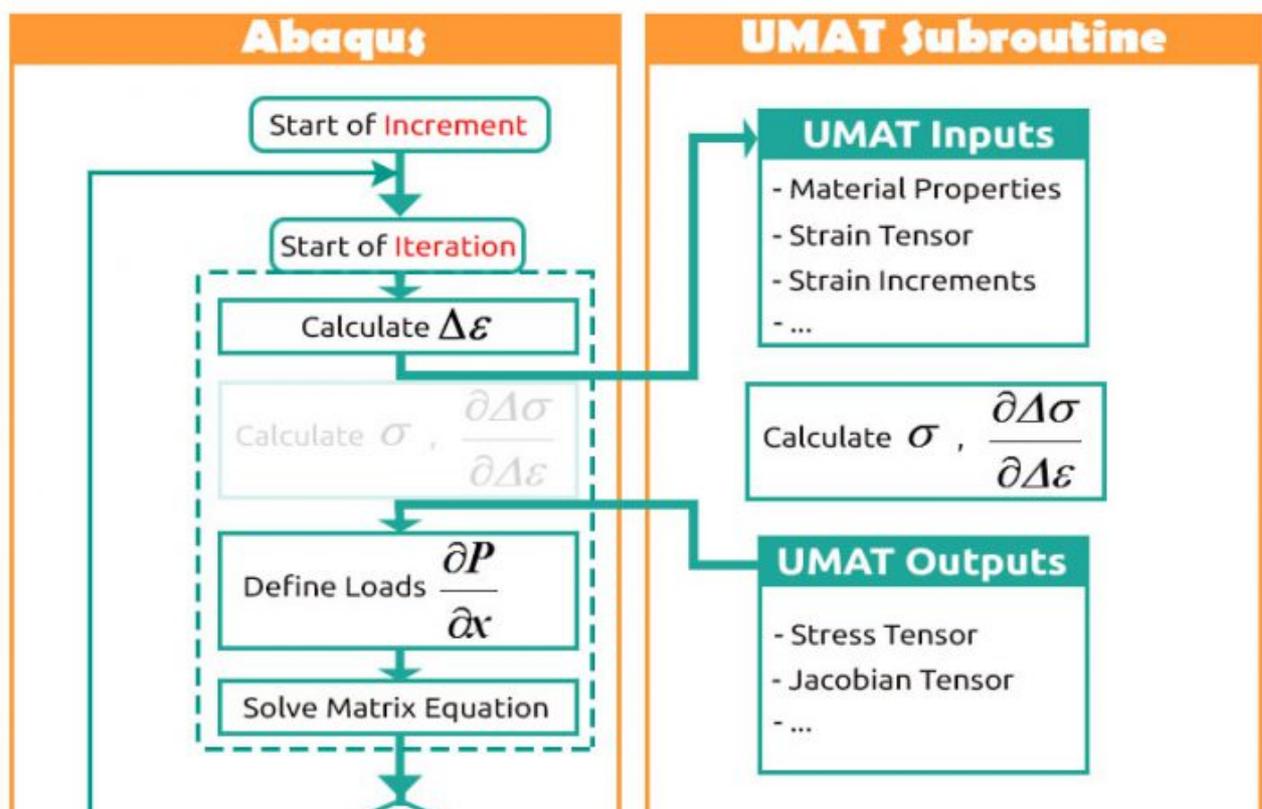


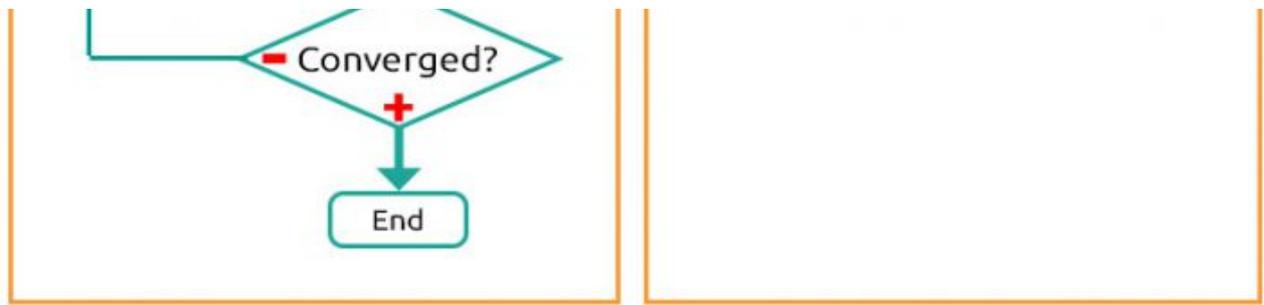
UMAT stands for User MATERIAL. Remember, we use UMAT when none of the existing material models included in the ABAQUS material library accurately represents the behavior of the material to be modeled.

### What does the UMAT Abaqus do?

Obviously, the philosophy behind the UMAT Abaqus subroutine is that the material model developer should be concerned only with the development of the material model and not the development and maintenance of the FE model.

Referring to the article How to start writing a subroutine (Section 3), you can see we use a UMAT in the stress calculation step.





CAEassistant.com

So, the main output of a UMAT subroutine is stress tensor. In the process, Abaqus also need defining of the (consistent) Jacobian to continue its incremental scheme.

Generally, to write a newly developed UMAT, you should follow these steps:

(Since we assumed this is the first time you start writing a UMAT, we start from the simplest one and present our step-by-step example in a pure elastic material, precisely called an isotropic linear elastic material model. For more advanced material models (plasticity, failure, element removal, brittle materials, etc. I refer you to UMAT Subroutine (VUMAT Subroutine) introduction package)



## 1. Knowing well the material model theory for our UMAT Abaqus

You should master all the aspects of the constitutive model for your material. In this step, the material model is represented in concise mathematical equations.

### EXAMPLE: Isotropic Linear Elastic Material (Theory)

**Isotropic** simply means the properties of the material are same in all directions. And, **Elastic** simply means that when the material is unloaded, it will return to its original (undeformed) state. In addition, **Linear** means that the relation between mechanical load and deformation of a material can be described by **Hooke's law**. In a stress-strain literature we can state Hook's law as:

$$\text{Stress} = [\text{Constant}] \times \text{Strain}$$

For our case (isotropic linear elastic material), we need two independent material parameters to define the state:

1. Constant that measures the resistance in the direction of load » **Young's modulus (E)**
2. Constant measures how much the material is dragged sideways for a vertical (perpendicular) load » **Poisson's ratio (ν)**.

Eventually, Hooke's law in the form of these two elastic constants is,

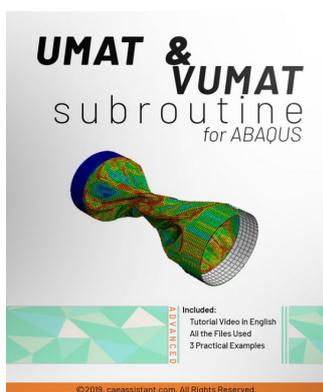
$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{31} \\ \sigma_{12} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & (1-2\nu)/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & (1-2\nu)/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & (1-2\nu)/2 \end{pmatrix} \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{23} \\ 2\epsilon_{31} \\ 2\epsilon_{12} \end{pmatrix}$$

$\sigma =$ 
 $C$ 
 $\epsilon$

And that is enough for writing our simple subroutine. The matrix **C** is the proportional factor that convert strain to stress. It is called stiffness matrix, i.e. how the material resists deformation.

You can see more details about the theory and equations in Workshop 1: Writing UMAT Subroutine for Isotropic Isothermal Elasticity of our UMAT training package:

## UMAT Subroutine (VUMAT Subroutine) introduction



€ 190,00

This package is usable when the material model is not available in ABAQUS software. If you follow this tutorial package, including standard and explicit solver, you will have the ability to write, debug and verify your subroutine based on customized material to use this in complex structures. These lectures are an introduction to write advanced UMAT and VUMAT subroutines in hyperelastic

Martials, Composites and Metal and so on.

**Watch Demo**

Buy now

Read more



## 2. Getting Familiar with UMAT Abaqus parameters (Inputs/Outputs)

We should restate our material model in UMAT language. So, you need to understand parameters name and their structures (are they scalar, vector or matrix? their dimension?).

As we talked in the article How to start writing a subroutine, the first step in writing any subroutine will be referring to the Abaqus Documentation. Therefore, you should find the UMAT section and review the guide to have a know-how of UMAT parameters.

Here, in a most general way, I sort all parameters for a UMAT in 4 distinguished groups:

### A. Parameters that are available by Abaqus to our UMAT

You can see all the parameters passed into UMAT in the documentation. The most important and practical ones are:

#### Reminder .....

In FORTRAN, we define **scalars** as single names, like ALPHA, SDV, etc. **Vectors** are defined by a name and their dimensions, like DMGQ(3), STRE(5), etc. For 2D **matrices**, the No. of rows and columns should be clarified when defining these variables: STRM(6,6), DDSDDL(4,4), ...

#### 1) NTENS

This is the size of the stress or strain component array and reflects the overall dimension of the Abaqus model into UMAT. For example, that is 6 for a general 3D and 4 for a simplified plane strain (2D) model.

In the stress and strain arrays and in the DDSDD (discuss soon), direct components are stored first, followed by shear components. There are NDI direct and NSHR engineering shear components. So,

$$NTENS = NDI + NSHR$$

## 2) STRESS(NTENS)

A vector form of stress tensor at the start of increment.

It has NTENS elements. For a general 3D model, we need 6 values for stress (3 normal and 3 shear) so this vector will have 6 elements.

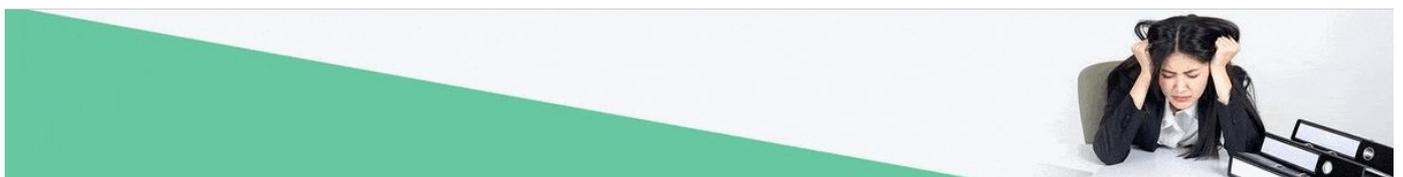
## 3) STRAN(NTENS), DSTRAN(NTENS)

STRAN is a vector form of strain tensor at the start of increment. DSTRAN is the increment (variation) of strain ( ).

## 4) TIME(2), DTIME

Respectively, total and incremental values of time. TIME is a 2-element vector; the first element, i.e. TIME(1) states the value of step time and the second, i.e. TIME(2), is the value of total time, both at the beginning of the current increment.

**Note.** There are many other variables that Abaqus pass into UMAT and were not discuss here, like **COORDS** (An array containing the coordinates of the point of calculation). According to your material, you may need to use them either. For example, if you are working on a FGM (Functionally Graded Material: material whose properties change gradually with respect to its dimensions) you need to use COORDS(3) parameter too.



## B. Parameters must be calculated and updated by UMAT Abaqus (outputs)

In a simple UMAT, these parameters are

### 1) STRESS(NTENS)

A vector form of stress tensor with **NTENS** elements. The dimension (NTENS) comes from the dimension of the model. For a general 3D model, we need 6 values for stress (3 normal and 3 shear) so this vector will have 6 elements.

At the first, **STRESS** is the stress tensor at the beginning of the increment. Then, we update it as the stress tensor at the end of the increment.

## 2) DDSDDDE(NTENS,NTENS)

A 2D format of **Jacobian** matrix with **NTENS** row and **NTENS** column.

For small-deformation problems (e.g., linear elasticity) or large-deformation problems with small volume changes (e.g., metal plasticity), the consistent Jacobian can be calculated as:

$$\mathbf{J} = \frac{\partial \Delta \boldsymbol{\sigma}}{\partial \Delta \boldsymbol{\varepsilon}}$$

Where,  $\Delta \boldsymbol{\sigma}$  is the increment in (Cauchy) stress and  $\Delta \boldsymbol{\varepsilon}$  is the increment in strain.

As you may guess, calculation of the consistent Jacobian can be the main challenge in writing a UMAT.

**Note.** There are other variables that Abaqus can get from a UMAT and we did not discuss here, like **RPL** (Volumetric heat generation caused by mechanical working in a thermo-mechanical material model). According to your material you may need to use them either.

## C. Parameters entered through Abaqus model definition & sent to UMAT

### PROPS (NPROPS)

User-specified array of material constants associated with this user material with **NPROPS** (number of material constants) elements.

When defining our model through Abaqus/CAE, we enter these parameters in Property module (we will discuss more, later).

## D. Parameters defined by user that is needed in next increments or we want Abaqus to save their values and later can be seen in results

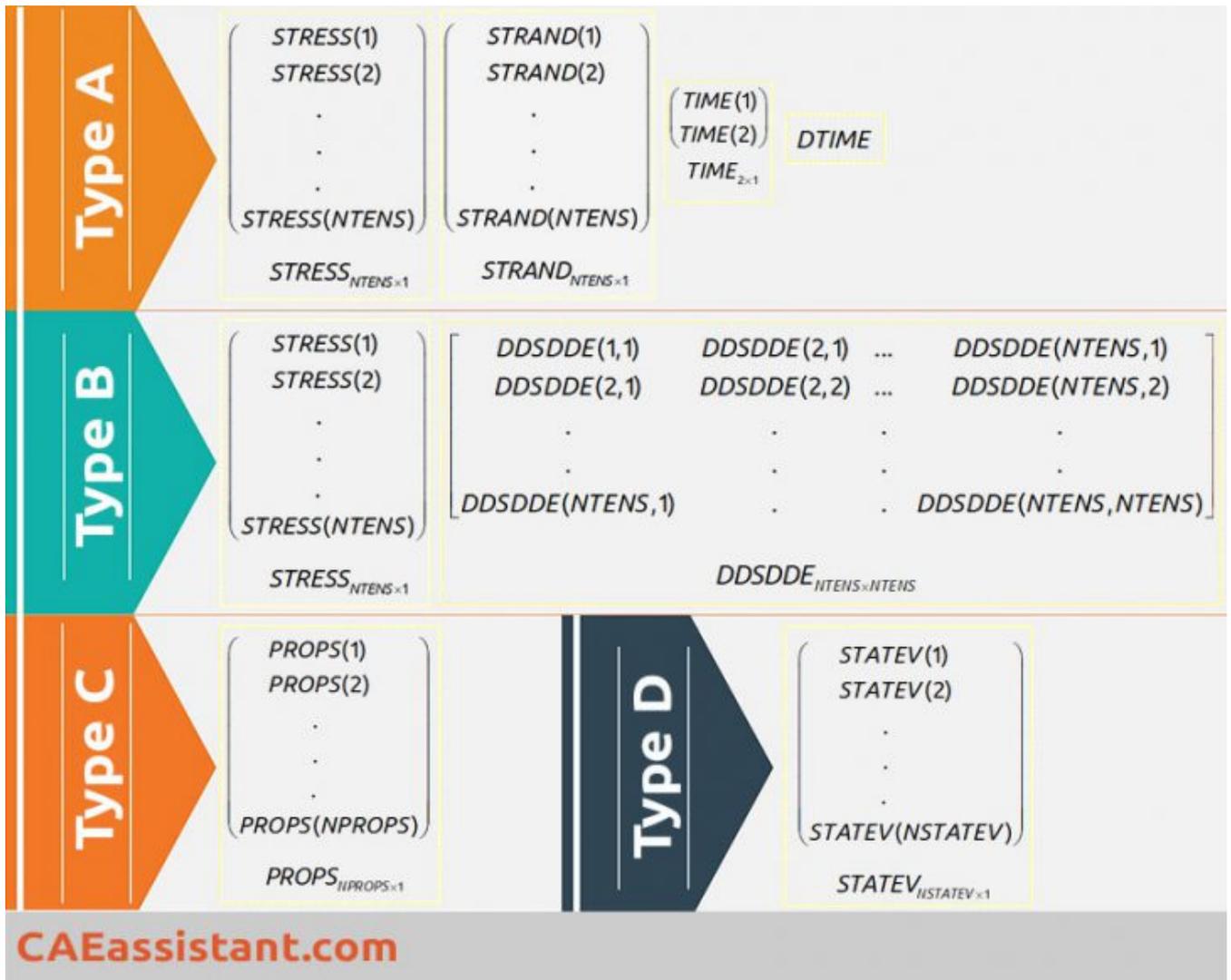
### STATEV(NSTATV)

An array containing the **Solution-Dependent state Variables (SDV)**. Abaqus pass in these parameters as the values at the beginning of the increment and at the end, return them as the values at the end of the increment. **NSTATV** (number of solution-dependent state variables associating with this material type) define the size of the array.

We should specify the number of needed state variable (if any) when defining our model

through Abaqus/CAE in Property module (we will discuss more, later).

After completing the analysis, Abaqus can visualize the values of state variables (named as SDV1, SDV2, ... in Abaqus/CAE) calculated during increments.



## EXAMPLE: Isotropic Linear Elastic Material (Required Outputs/inputs)

Here we will only use **DDSDE**, **STRESS** and **STRAN**. The STRAN matrix is an input to the UMAT. Then, UMAT updates the STRESS matrix (using DDSDE and STRAN matrices) and returns it as an output.

In this simple problem, we can easily find Jacobian from elastic stress-strain relation:

So, here we can use a single notation and use Jacobian or DDSDE matrix to find STRESS too.

In addition, we use PROPS(1) and PROPS(2) as Young's modulus  $E$  and Poisson's ratio ( $\nu$ ), respectively. For that, we will enter the values when defining the material in Abaqus/CAE later. See the figure in **4. Implementation in Abaqus & Compilation** of Part B.

**[Follow the rest of steps in the PART B of this article]**



## Start Writing Your 1st UMAT for Abaqus (PART B: Construction)

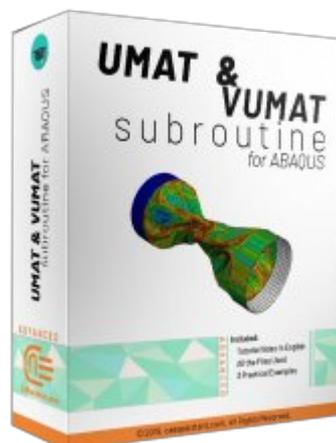
[First you should read the prerequisite article] Start Writing Your 1st UMAT for Abaqus (PART A: Foundation) 3. Developing FORTRAN ... Continue reading



0



If you are looking for more information about UMAT or you need to write a VUMAT, take a look at our comprehensive training package for UMAT/VUMAT:



**References:**