Tutorial 1
Simple tension test in 2D with two elements

Gergely Molnár[1], Anthony Gravouil

Laboratoire de Mécanique des Contacts et des Structures, Institut National des Sciences Appliquées de Lyon

18-20, rue des Sciences, 69621, Villeurbanne Cedex, France

Abstract

This tutorial is first in the line. It shows how to construct a simple finite element model using phase-fields to model diffused cracks. After the problem definition, several images will show how to transform an input file generated by ABAQUS to use the UEL containing a degradation degree of freedom. At the end a few post-processing tips are going to be provided to simplify the evaluation of the results.

More about the method can be found in our paper: *G. Molnár, A. Gravouil, 2D and 3D Abaqus implementation of a robust staggered phase-field solution for modeling brittle fracture, Finite Elements in Analysis and Design, 130 pp. 27-38, 2017.*
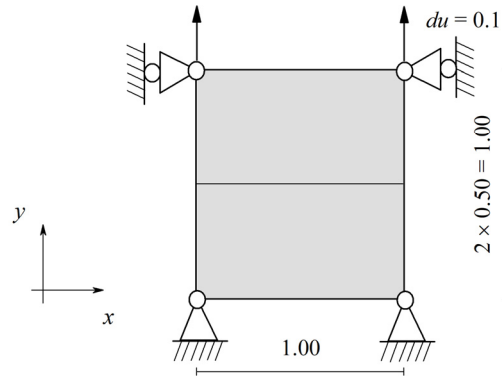
1   Problem definition

Figure 1 Geometry and boundary conditions of a simple tension test with two finite elements.

To demonstrate the basic process, which should always be used to develop a finite element model with the current UEL implementation is demonstrated through the simple example depicted in Figure 1. A 2D plate is extended on top in direction *y*.

The tutorial has four parts: 1) ABAQUS model preparation; 2) Input file modification to include UEL; 3) Running the simulation; 4) Post-processing.

2   ABAQUS model

This tutorial assumes that all the necessary software packages (ABAQUS, FORTRAN compiler, etc.) are already installed and calibrated on the computer. Current tutorial was tested using ABAQUS version 6.13. We assume that the user read the article mentioned in the abstract. Furthermore, we strongly encourage everyone to clearly understand the fundamental theory of the used diffused damage/phase-field model.

---

[1] *e-mail: gmolnar.work@gmail.com*

The tutorial contains screenshots to aid the model development. Every step is highlighted with red numbering. Let's start a new model in ABAQUS and save it as "*SimpleTension.cae"*. Open the *Part* module and click on the *Create Part* icon. Set the modeling space to *2D Planar* and leave the base feature on *Shell*. Set the approximate size to *5* and finally click *Continue* (see Figure 2 for step-by-step details).
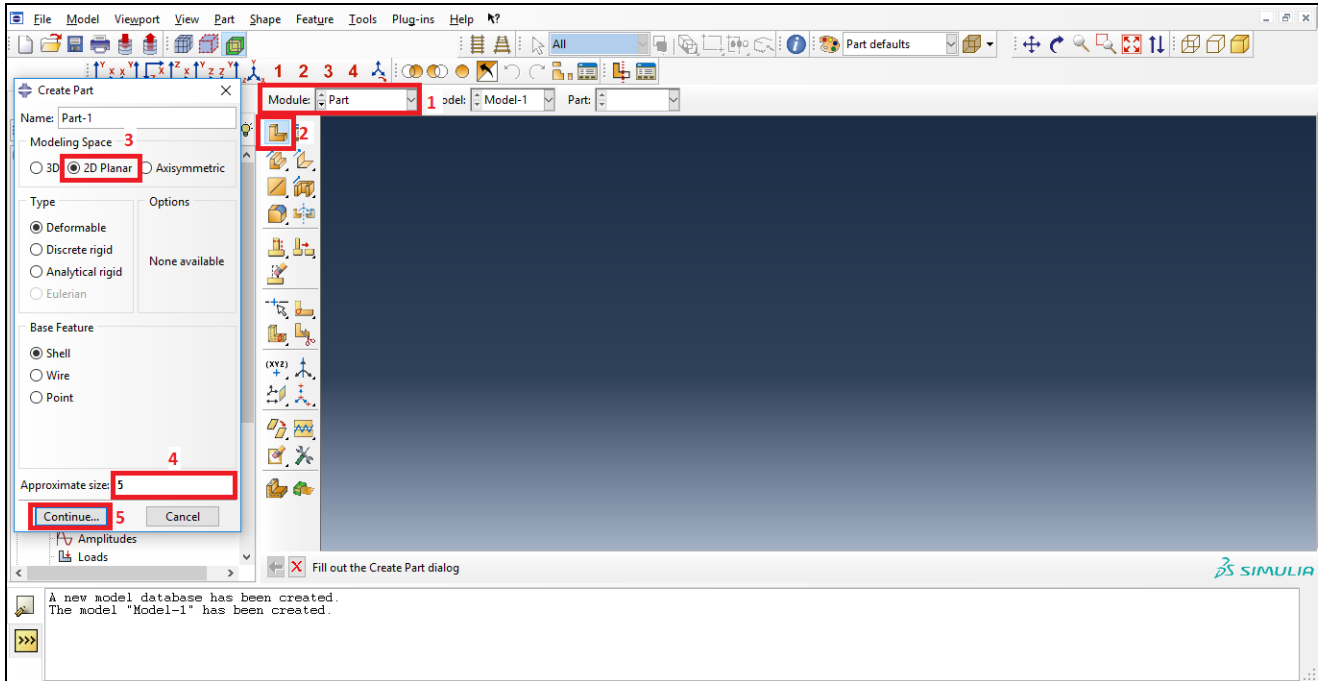

Figure 2 Creating a new part.

Draw a rectangular plate with the dimensions indicated in Figure 3. Use the *Create Lines: Rectangle (4 lines)* tool. Place the first point to (-0.5,-0.5), and the second to (0.5,0.5). Press the escape button and finally hit *Done* (see Figure 3 for step-by-step details).
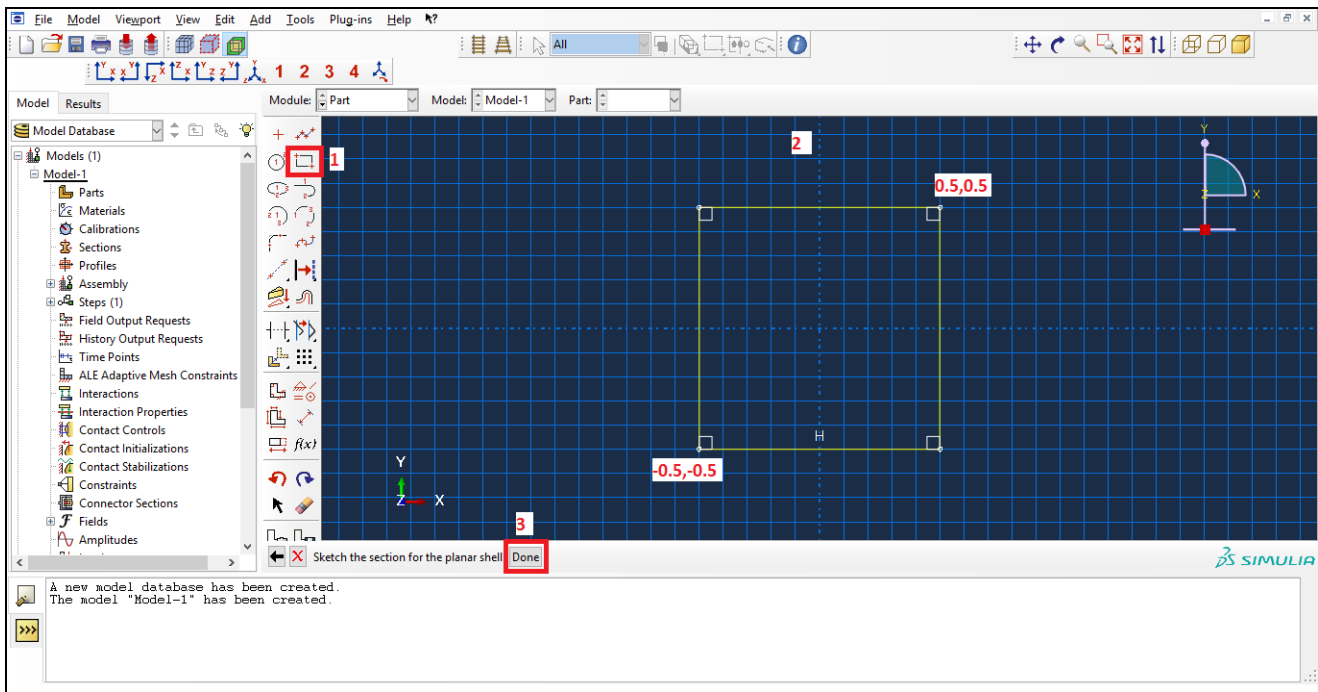

Figure 3 Drawing a part.

2

To create a new material change to *Property* module. Click on the *Create Material* button, name it *umat* and add a *User Material* behavior from the *General* tab (similarly as above, Figure 4 shows the process step-by step).
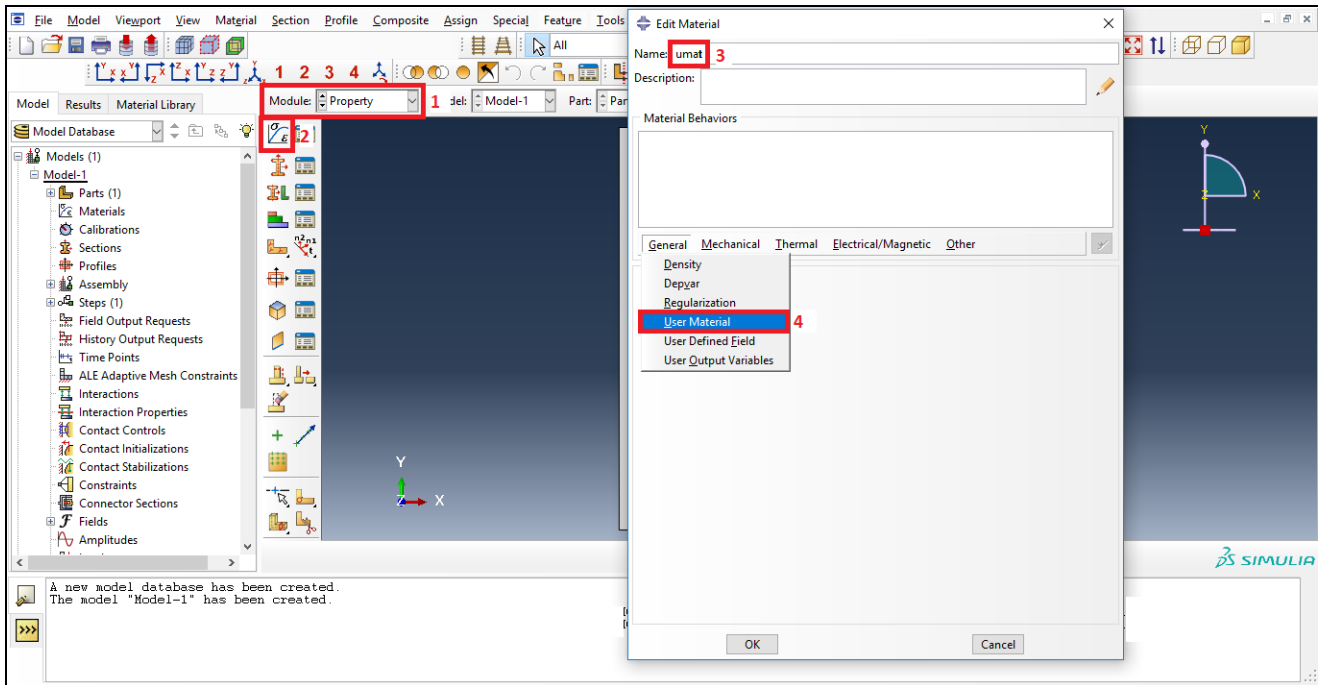

Figure 4 Creating new material property.

As mechanical constants enter *1e-11* and *0.3* (as shown in Figure 5) then add another property from the *General* menu called *Depvar*. Write *16* as the *number of the solution dependent variables* and click *OK*. This will allow us to transfer data from the UELs to the post-processing space by a UMAT layer.
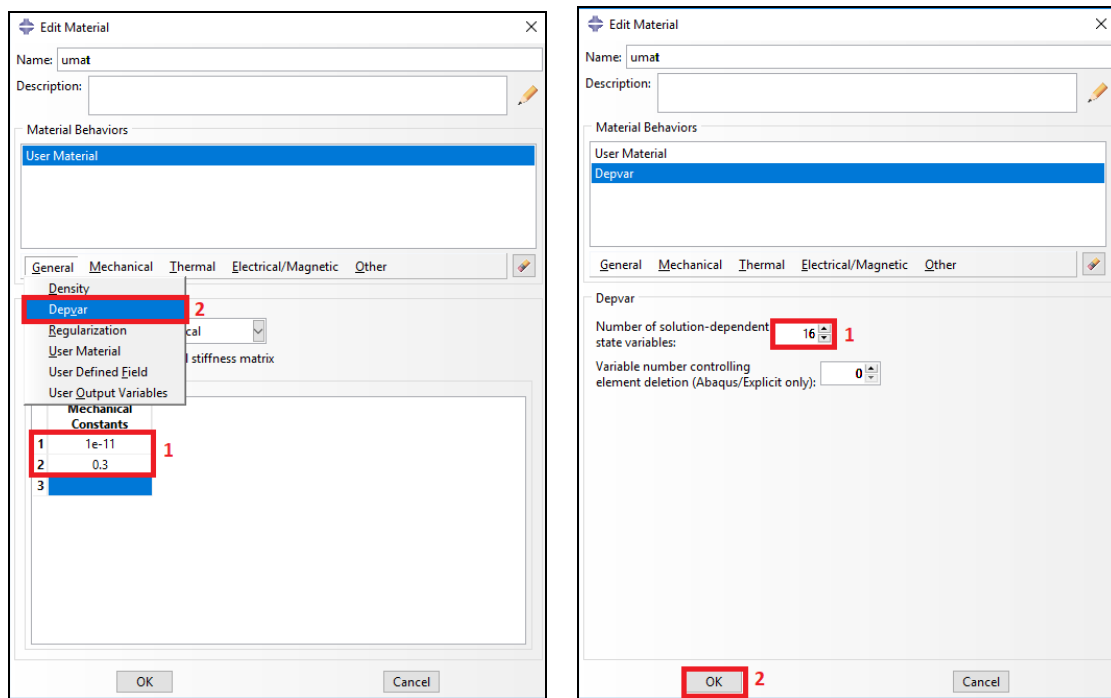

Figure 5 Material coefficients for UEL (left) and number of solution dependent variables (right).

3

Now we have to connect this material to a so-called section. In order to do this, use the *Create Section* menu, leave everything unchanged and *Continue*. Finally click *OK* (Figure 6).
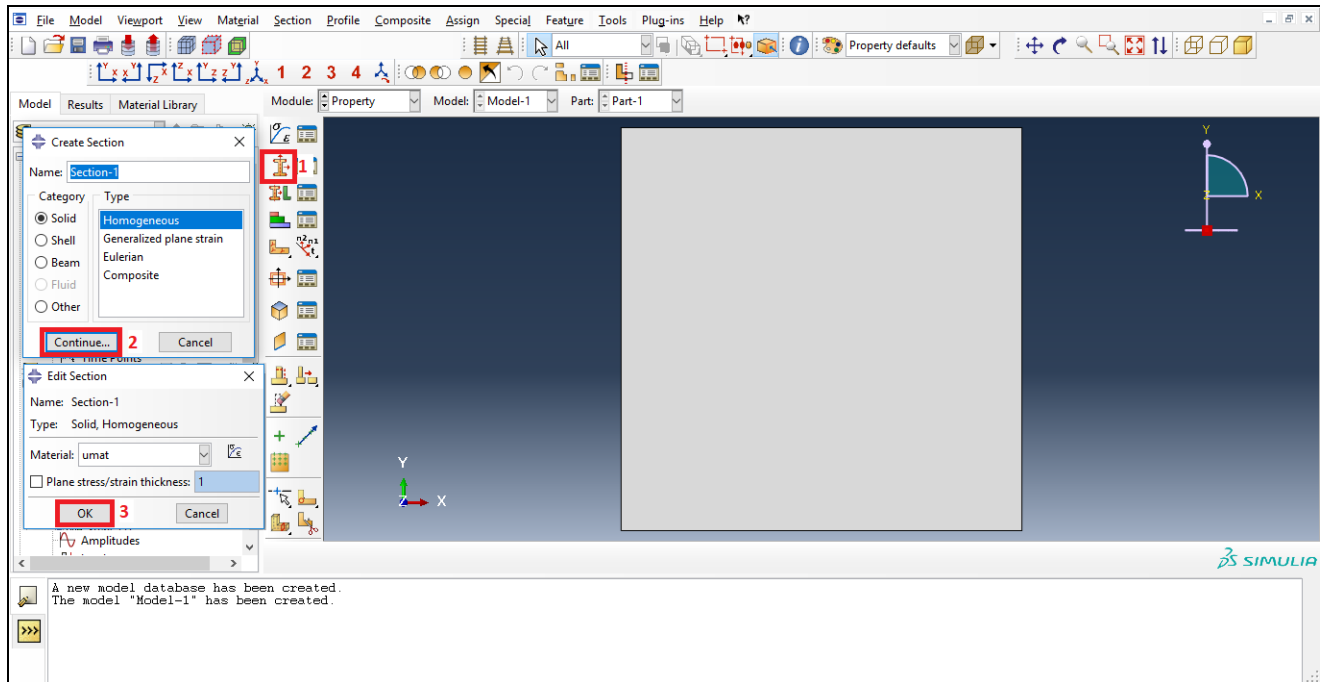

Figure 6 Creating section.

Now to assign the created section to the formerly drew part, click *Assign Section*. Select the plate by clicking on it and hit *Done*. Make sure you create a set named *Set-1*. In the upcoming menu make sure that the selected section is *Section-1* and finally click *OK* (see Figure 7).
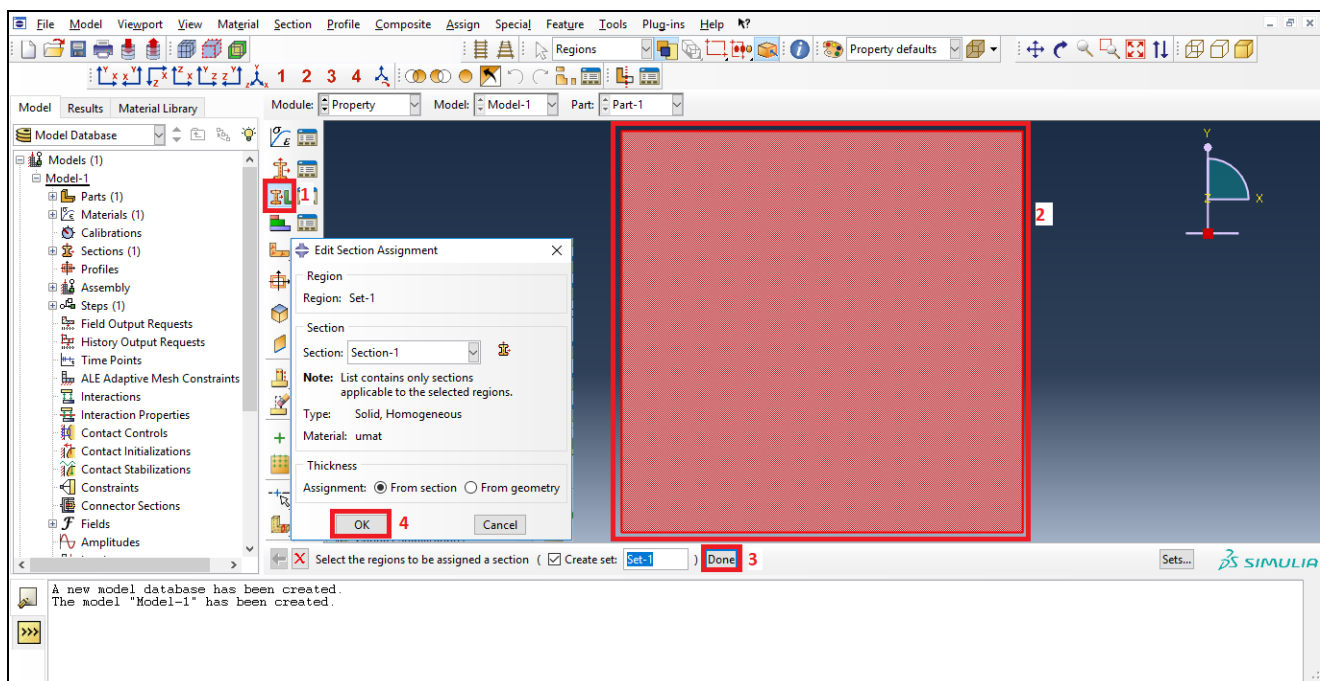

Figure 7 Assigning section to part.

Change the module again to *Assembly*. Click on the *Create Instance* button and then *OK*. Now *Part-1* is added to the simulation with an assigned section and material property. A blue rectangle should appear in the simulation space as it is shown in Figure 8.
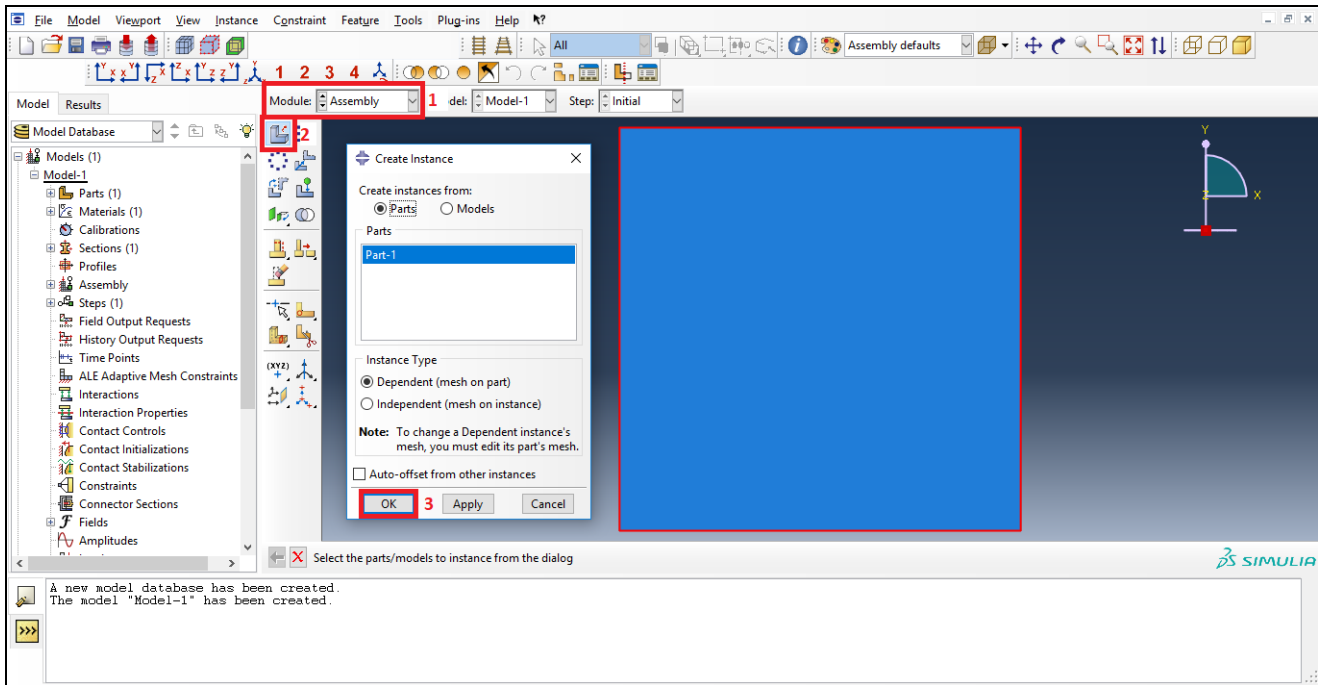

Figure 8 Assembly.

To define the iteration, switch to *Step* module and click *Create Step*. Leave the selection on *Static, General* and hit *Continue*. In the *Incrementation* tab choose a *Fixed* type method. Type *1000* to the maximum number of increments, and set the step size to *0.001*. Finish by clicking *OK* (see Figure 9).
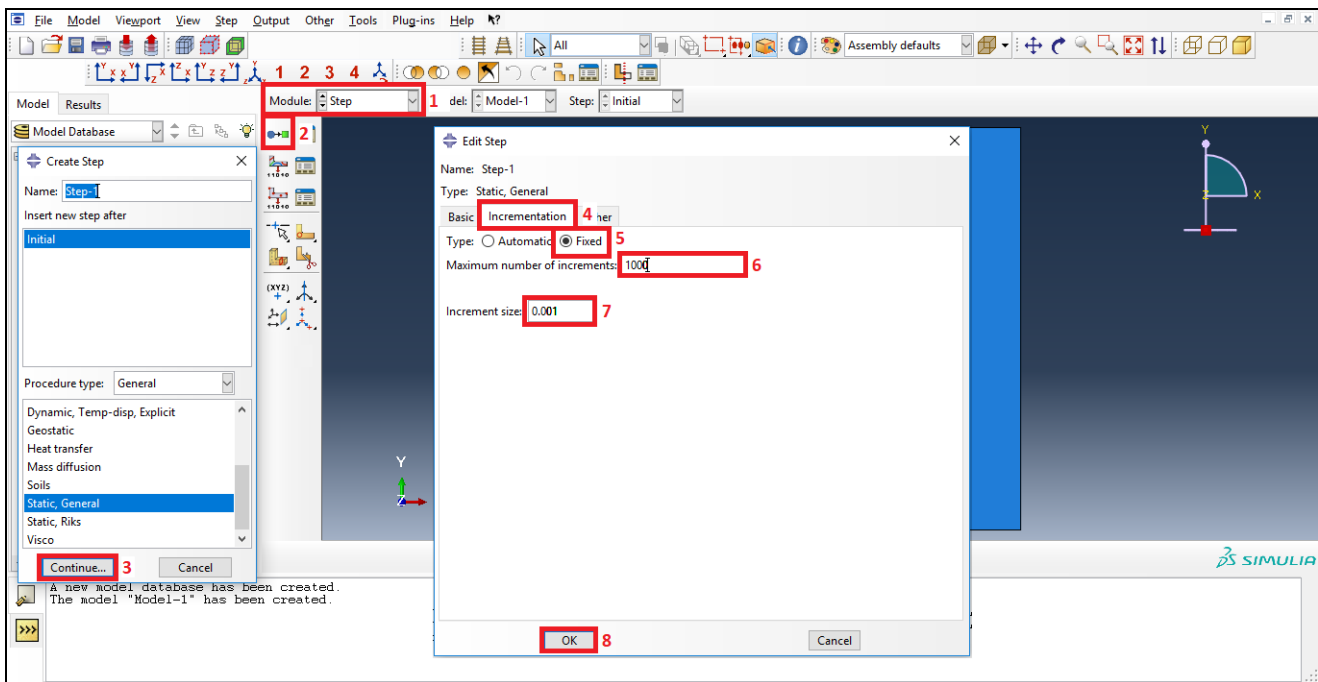

Figure 9 Defining iteration properties.

5

Skip the *Interaction* module and go directly to *Load*. First we will create the fixed boundary conditions on the bottom of the plate. Use the *Create Boundary Condition* button and change the name to *BOTTOM*. Select a *Displacement/Rotation* type condition and press *Continue*. Make sure that the *Step-1* is selected as the active simulation period. Now select the bottom edge of the plate, name the new set to *BTM* and hit *Done*. In the pop-up window tick the first two boxes and put *0* after them. This way, we told the program that on the nodes, which are on the bottom edge, the displacement in both *x* and *y* direction is zero. Click *OK* as shown in Figure 10.
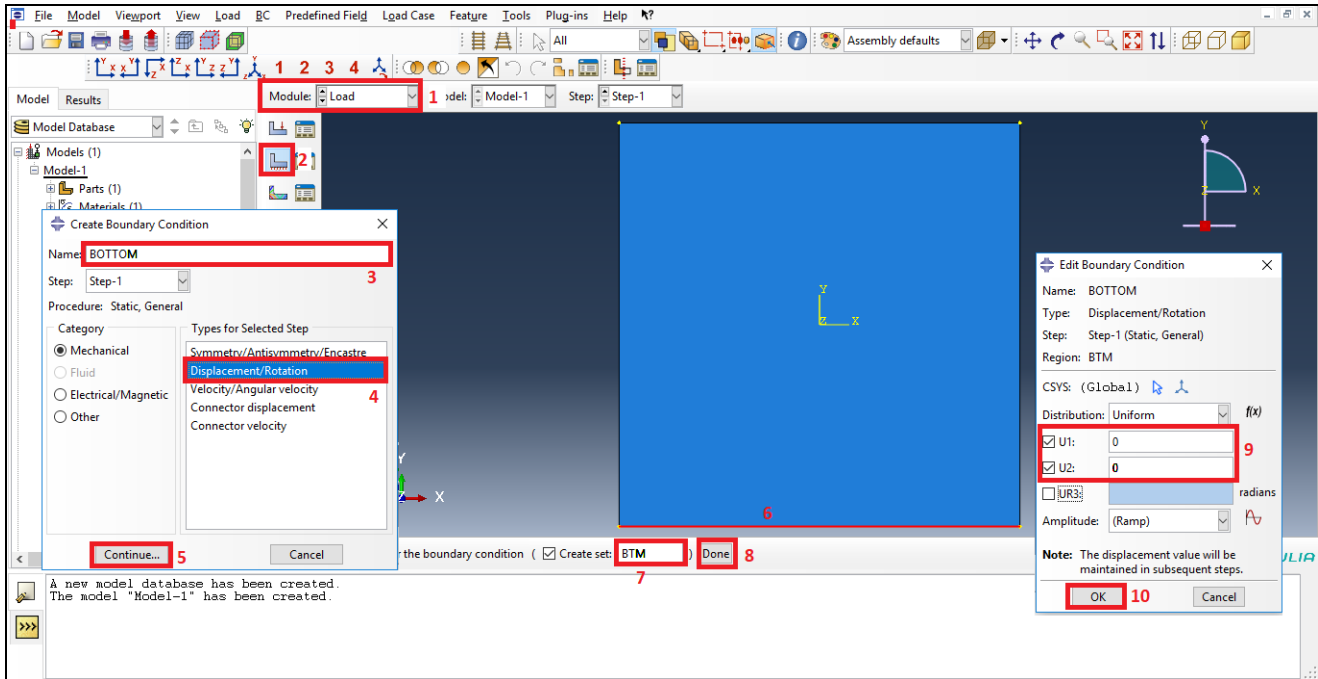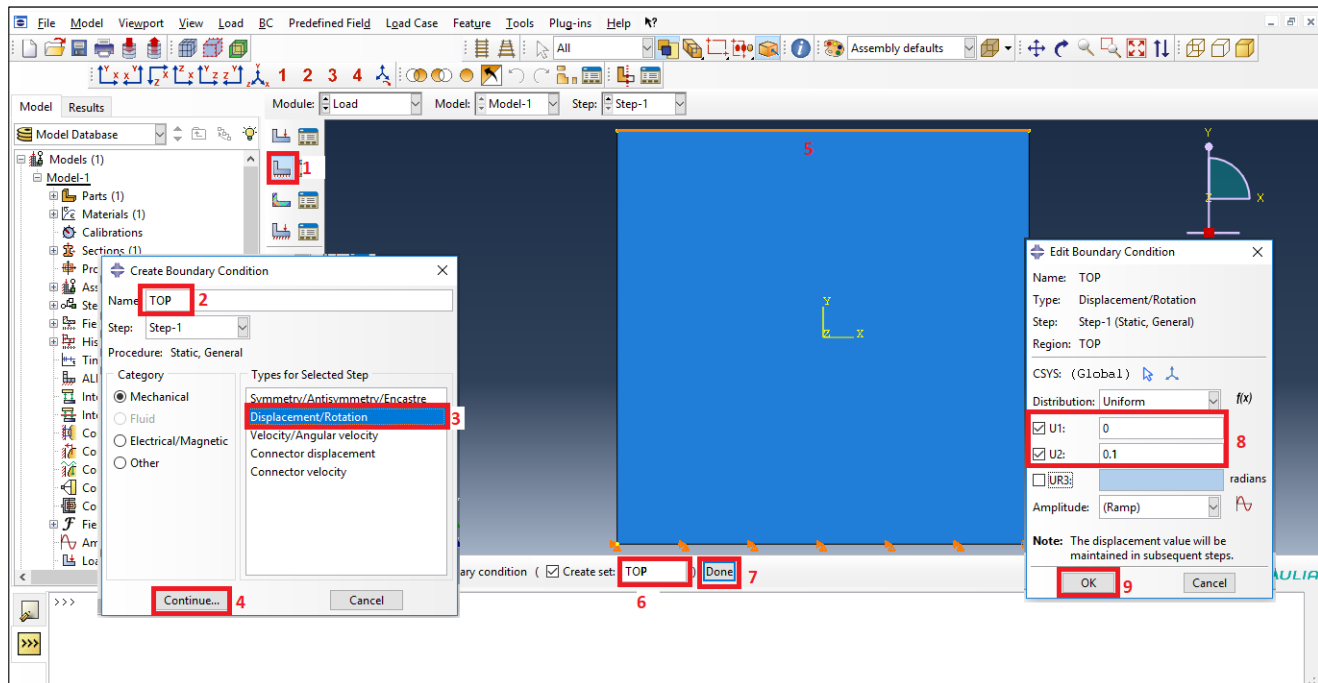

Figure 10 Fixed boundary conditions.

Figure 11 Creating constant displacement step on the top edge.

6

On the top boundary we will apply the displacement step similarly as we did on the bottom. Use the same *Create Boundary Condition* button and change the name to *TOP*. Select a *Displacement/Rotation* type condition and press *Continue*. Make sure similarly, that *Step-1* is selected as the active simulation period. Now select the top edge of the plate, name the new set to *TOP* and hit *Done*. In the pop-up window tick the first two boxes. In the first (*x* direction) put *0*, while in the second (*y* direction) put *0.1*. This means that while the displacement is going to remain zero in the *x* direction, during *Step-1* the displacement in *y* direction is going to increase linearly until 0.1.Finalize the load definition by clicking *OK* as shown in Figure 11.

Go straight to the *Mesh* module. Change the *Object* from *Assembly* to *Part*. This will allow us to define the finite element mesh for each part. Set an average mesh size by clicking onto *Seed Part Instance*. In the pop-up window set the *Approximate global size* to *1* (bottom right pop-up window in Figure 12) and click *OK*. To divide the side edges into two pieces, click on *Seed Edges* (shown with number 6 in Figure 12). In the pop-up window select *By number* method, make sure that there is no bias, and set the *Number of elements* to *2*. After clicking *OK*, set the meshing algorithm using the *Assign Mesh Control* button. Make sure that a *Quad* meshing algorithm is selected, and hit *OK*. To generate the mesh, click on *Mesh Part Instance* (shown with number 16 in Figure 12) and *Yes*. At the end of this procedure, two equal size horizontal elements should appear.

Figure 12 Meshing the part.

7

There is nothing else left, go to *Job* module. Click on *Create Job*. Name the new job to "*Tens*" and click *Continue*. In the pop-up window, simply click *OK*. Go to the *Job Manager* (shown with number 5 in Figure 13) and ask the program to write an input file by clicking on *Write Input*.
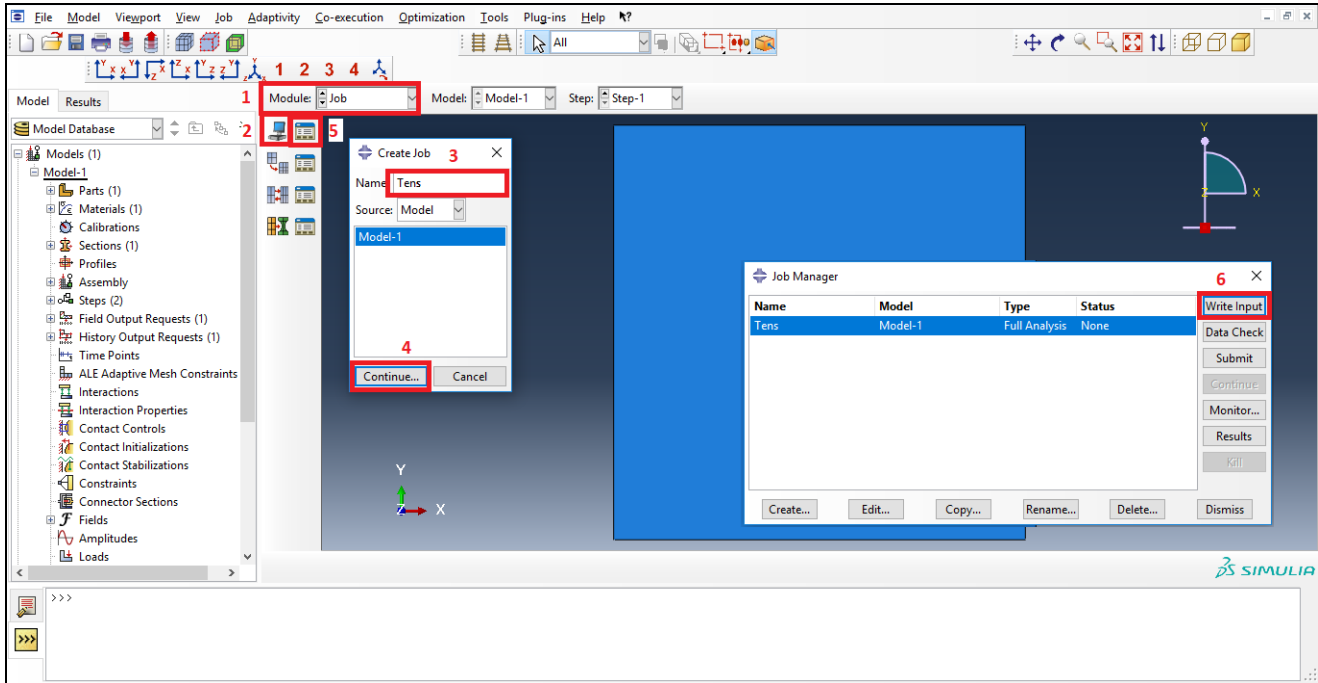

Figure 13 Job definition.

At this stage we have a functional ABAQUS model, however, the phase-field UEL is still not included.

3   Input file modification

In this next step, line-by-line we will modify the original input file and create a new one, which uses the UEL phase-field element. This file will then be used by the solver to create and run the simulation.

Locate and open the input file named *"Tens.inp"*, which was created by ABAQUS or downloaded from our research page[2]. Open it with a text editor (e.g. Notepad++, Vim, gedit, etc.) and save it as *"Tens_PH.inp"*.

We are going to modify certain lines and add some information concerning the UEL by hand. This tutorial is created to demonstrate the use of the 4-node phase-field finite element, which was published in *G. Molnár, A. Gravouil, 2D and 3D Abaqus implementation of a robust staggered phase-field solution for modeling brittle fracture, Finite Elements in Analysis and Design, 130 pp. 27-38, 2017.*

From our website, the already modified input file can be downloaded, as well and this essential step can be skipped.

---

[2] http://www.molnar-research.com/tutorials/1_PH_2element/Tutorial_1_Simple2D.rar

After the headings, the input file contains the Part section where, firstly the nodes are shown. We will modify the following section first:

```
*Element, type=CPS4R
1, 1, 2, 4, 3
2, 3, 4, 6, 5
*Nset, nset=Set-1, generate
 1,  6,  1
*Elset, elset=Set-1
 1, 2
** Section: Section-1
*Solid Section, elset=Set-1, material=umat
,
```

Remove this part to a separate file, we will going to use the element definitions later. Start by inserting the following line just after the coordinates of the nodes:

```
*User element, nodes=4, type=U1, properties=3, coordinates=2, VARIABLES=8
3
```

This command defines the type of the first layer (U1) with only the 3rd degree of freedom (phase-field). Create the first layer of FEs by typing:

```
*Element, type=U1
1, 1, 2, 4, 3
2, 3, 4, 6, 5
```

Tow two latter lines contain a sequence, which should always be followed. The first number is the number of the finite element, the following four are the node numbers which are used to define the element. The nodes must be given in a counterclockwise manner.

Define the first element set, which will contain all phase-field elements (1st and 2nd elements).

```
*Elset, elset=AllPhase, generate
1, 2, 1
```

If the model is much bigger, the original set removed from the original input file can help (in this case located at line 21 and 22), as it is automatically constructed by ABAQUS. Only the name of the set must be changed to *AllPhase*. Element sets can be defined several ways. For more details, please visit the following site[3]. This allows us to assign material properties to the UEL by typing:

```
*Uel property, elset=AllPhase
0.1, 5e-3, 1.0
```

Where the first value corresponds to the length scale parameter ($l_c = 0.1$), the second one to the fracture surface energy ($g_c = 5\times10^{-3}$) and finally the thickness of the plate ($h = 1.0$).

---

[3] http://web.mit.edu/calculix_v2.7/CalculiX/ccx_2.7/doc/ccx/node198.html

Similarly to the first one, the second element is initiated by the following lines:

```
*User element, nodes=4, type=U2, properties=4, coordinates=2, VARIABLES=56
1,2
```

This means, that the second element (U2) has two translational degrees of freedom. Create the elements by typing the same element definition, as was done for the first layer, but increase the initial numbering by 2, as we have 2 elements in each layer:

```
*Element, type=U2
3, 1, 2, 4, 3
4, 3, 4, 6, 5
```

Create the element set for the displacement problem:

```
*Elset, elset=AllStress, generate
3, 4, 1
```

Finally assign the material properties:

```
*Uel property, elset=AllStress
210.0, 0.3, 1.0, 1.0e-7
```

Where the numbers represent the Young's modulus (E), then Poisson's ratio (v), the thickness (h) and finally a numerical stability parameter (k).

As a third layer the UMAT elements should be created. Pay attention to shift again the numbering of the element list by 2.

```
*Element, TYPE=CPS4, elset=umatelem
5, 1, 2, 4, 3
6, 3, 4, 6, 5
*Solid Section, elset=umatelem, material=umat
1.0
```

The already in place *End Part command terminates this section.

Scroll down and search for the end of the Assembly definition: *End Assembly. Just before this line add the following two commands:

```
*Nset, nset=AllP, instance=Part-1-1, generate
1, 6, 1
*Elset, elset=umatelem, instance=Part-1-1, generate
5, 6, 1
```

These will help as in post processing. The first command creates an element set (AllP) with all the nodes in the model. While the second selects all the UMAT elements (umatelem).

There if nothing else to do, but to request the desired output variables. Search for the lines:

```
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, variable=PRESELECT
```

This should be changed to:

```
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
*Output, field
*Node Output, nset=AllP
RF, U
*element output, elset=umatelem
SDV
**
```

These commands request displacement and reaction force values for all nodes, and solution dependent variables (SDV) for the UMAT elements.

The final line should be: `*End Step`, which we left unchanged.

4   Running the simulation

Download or copy a version of the UEL FORTRAN file into the directory, next to *"Tens_PH.inp"*. Open it with a text editor and search for the variable definition `N_ELEM`. There should be two occurrence, at line 44 and at line 525. Make sure that `N_ELEM=2` in this case. This number should always correspond to the number of elements in one layer. Save the FORTRAN file as *"Tens_PH.for"* for PCs, or if it is used on a UNIX computer *"Tens_PH.f"*.

Open the previous ABAQUS file ("*SimpleTension.cae*") and go to the *Job* module. Open the *Job Manager* and *Create* a new job. Name the job as *"Tens_PH"* change the source to *Input file* and search for the hand modified version *"Tens_PH.inp"*. Press *Continue* (see Figure 14).
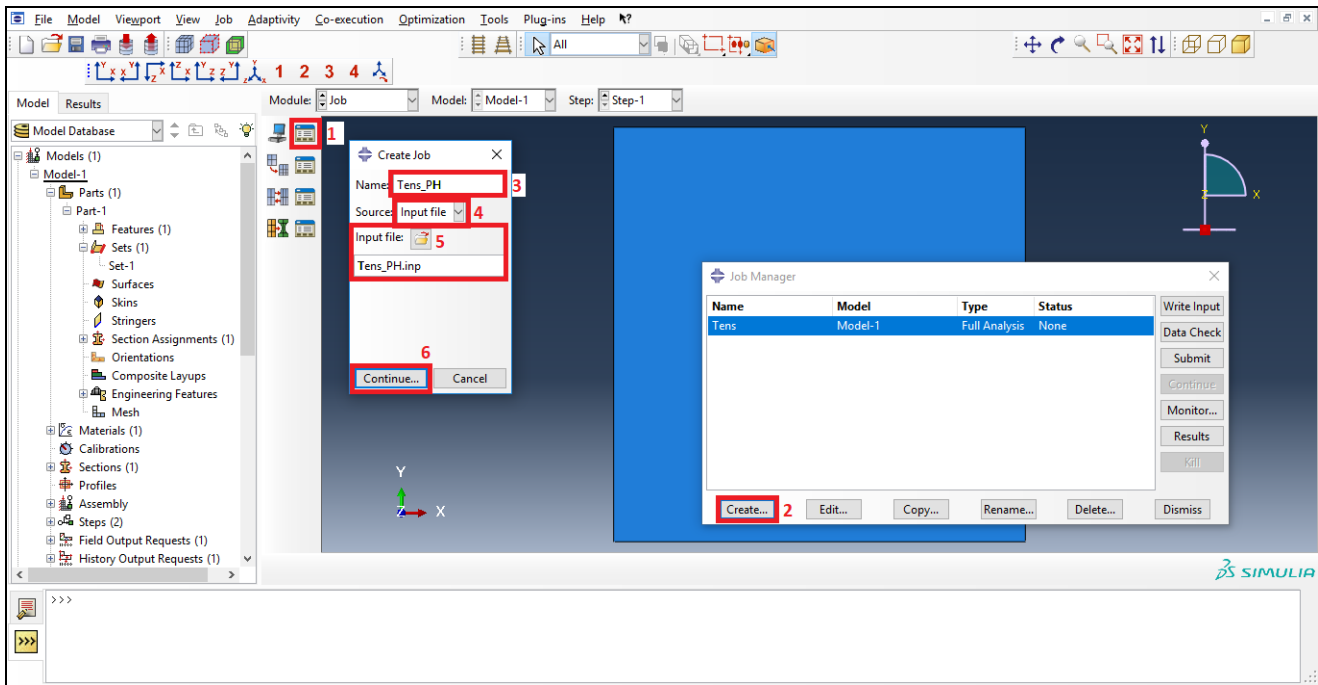
Figure 14 Creating a new job using a handwritten input file.

In the pop-up window go to the *General* tab and add the FORTRAN file by clicking on the folder icon. Finish the job by hitting *OK*.
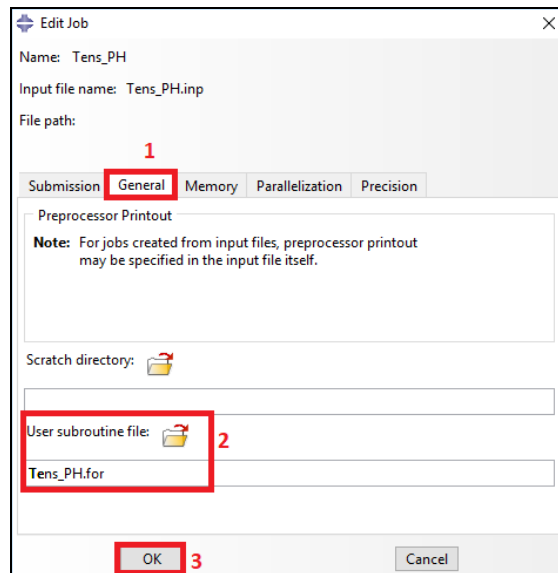

Figure 15 Connecting the job with the UEL.

Press *Submit* to start the simulation in the job manager. When the job is ready, the status will change from *Running* to *Completed*. To open the ODB file click on the *Results* button.

5   Post-processing

After clicking on the *Results* button the program automatically jumps to the *Visualization* module. First we are going to clean up our results. These steps are probably not necessary for a simulation with two elements,

but imagine all the green X's for thousands of nodes. To remove them, first go to Tools/Display Group/Create. In the pop-up menu select the *Element* items, then the *UMATELEM* set. Finally click on *Replace* and then *Dismiss*.
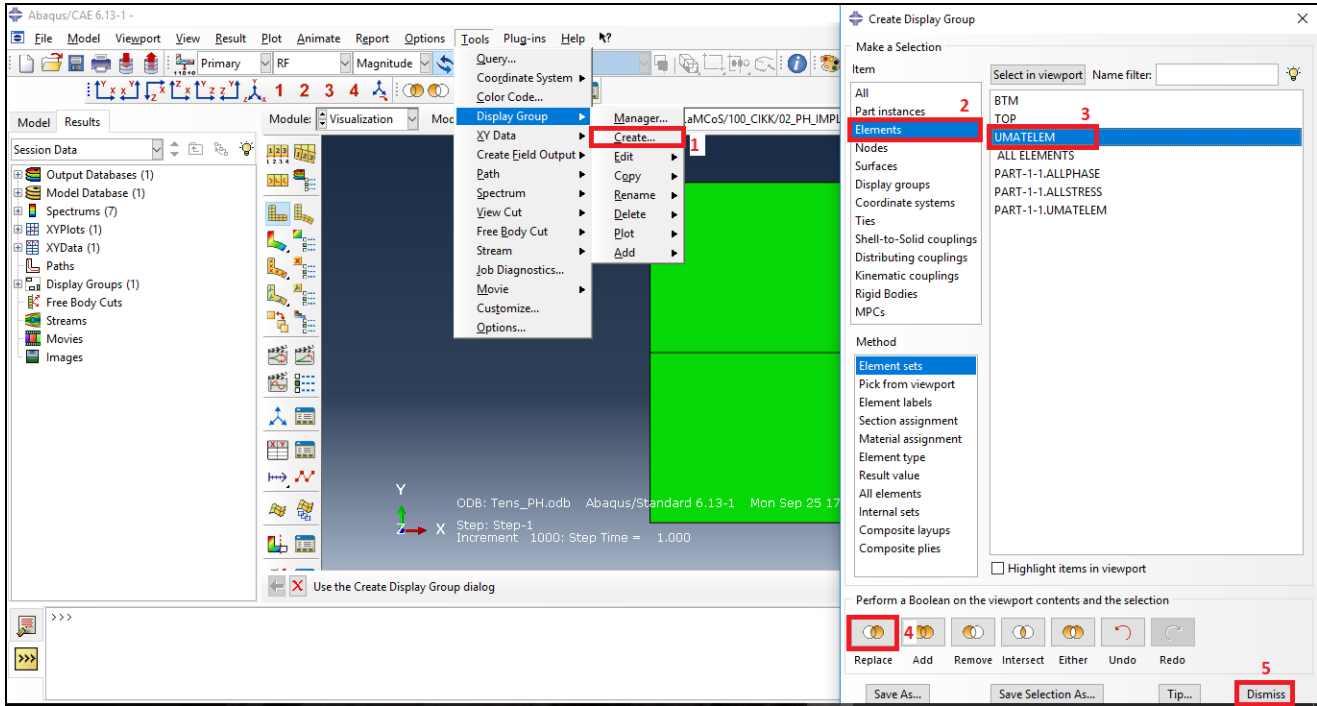

Figure 16 Removing green X's.

To remove the internal mesh click on Common Options and change the *Visible Edges* option to *Feature Edges*. The results now can be displayed as usually done in ABAQUS. The corresponding mechanical variables which we transferred from the UEL through solution dependent variables (stress, strain, phase-field damage, etc.) are summarized in Table 1.

Displacement (stress-strain) element	
Displacement – u_x, u_y	SDV1-SDV2
Axial strains – ϵ_x, ϵ_y	SDV3-SDV4
Engineering shear strain – γ_{xy}	SDV5
Axial stress – σ_x, σ_y	SDV6-SDV7
Shear stress – τ_{xy}	SDV8
Elastic axial stress – $\sigma_{x,0}$, $\sigma_{y,0}$	SDV9-SDV10
Elastic shear stress – $\tau_{xy,0}$	SDV11
Strain energy – ψ	SDV12
Elastic strain energy – ψ_0	SDV13
Phase-field – d	SDV14
Phase-field element	
Phase-field – d	SDV15
History field – H	SDV16

Table 1 Solution dependent variables and corresponding mechanical quantities.

Usually the reaction force measured on the node, where the displacement is applied gives valuable information about the behavior of the model. To visualize the reaction force on one of the top nodes click on *Create XY Data* (still in the Visualization module). Select *ODB field output* and click *Continue*. In the pop-up window, in the *Variables* tab, select *Unique Nodal* as Position. Type: `RF.RF2,U.U2` in the Edit line.
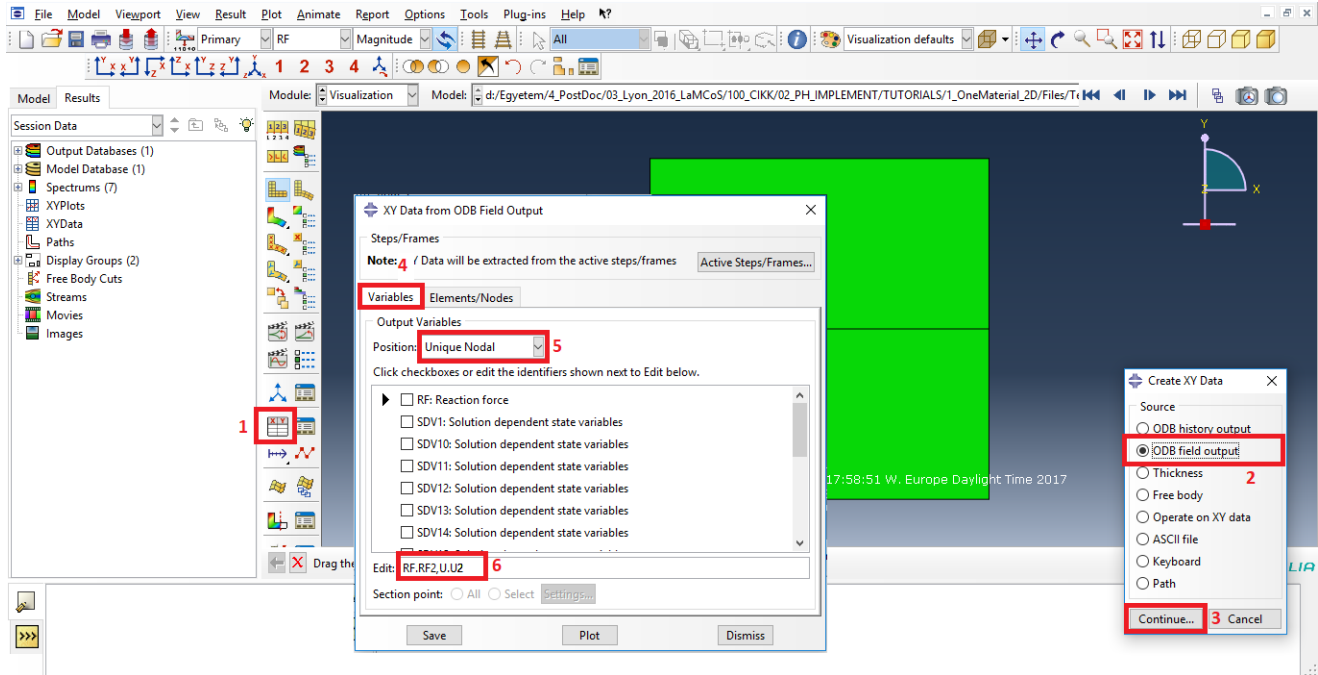

Figure 17 Asking the program to display reaction force and displacement in y direction.

Change the tab to *Elements/Nodes*. Go to *Pick from viewport* and start editing the selection by clicking on *Edit Selection*. Select the tob/left node by clicking on it, then hit *Done*, *Plot* and *Dismiss*.
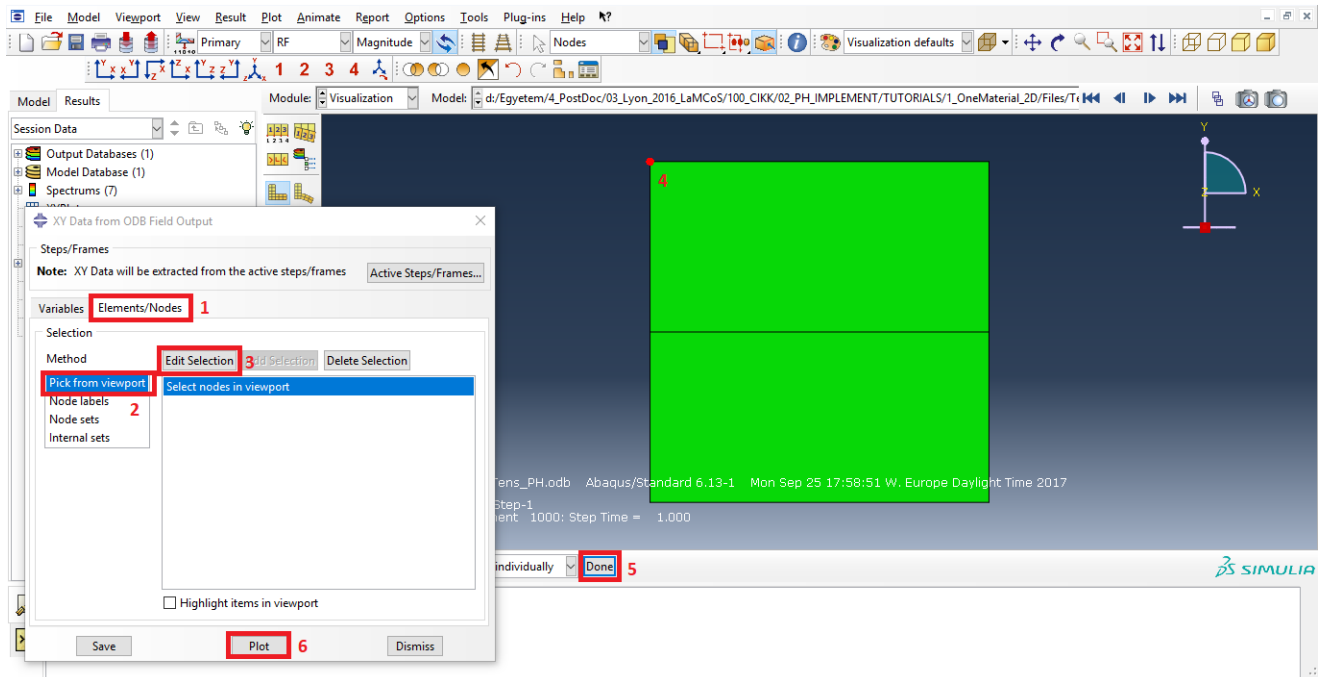

Figure 18 Specifying from which node do we want the reaction force and node results.

14

To display the reaction force as a function of displacement click again *Create XY Data*. Then select *Operate on XY data* and click *Continue*.

Type in: `combine("_U:U2 PI: PART-1-1 N: 2","_RF:RF2 PI: PART-1-1 N: 2")` and click *Plot Expression* and close the window.
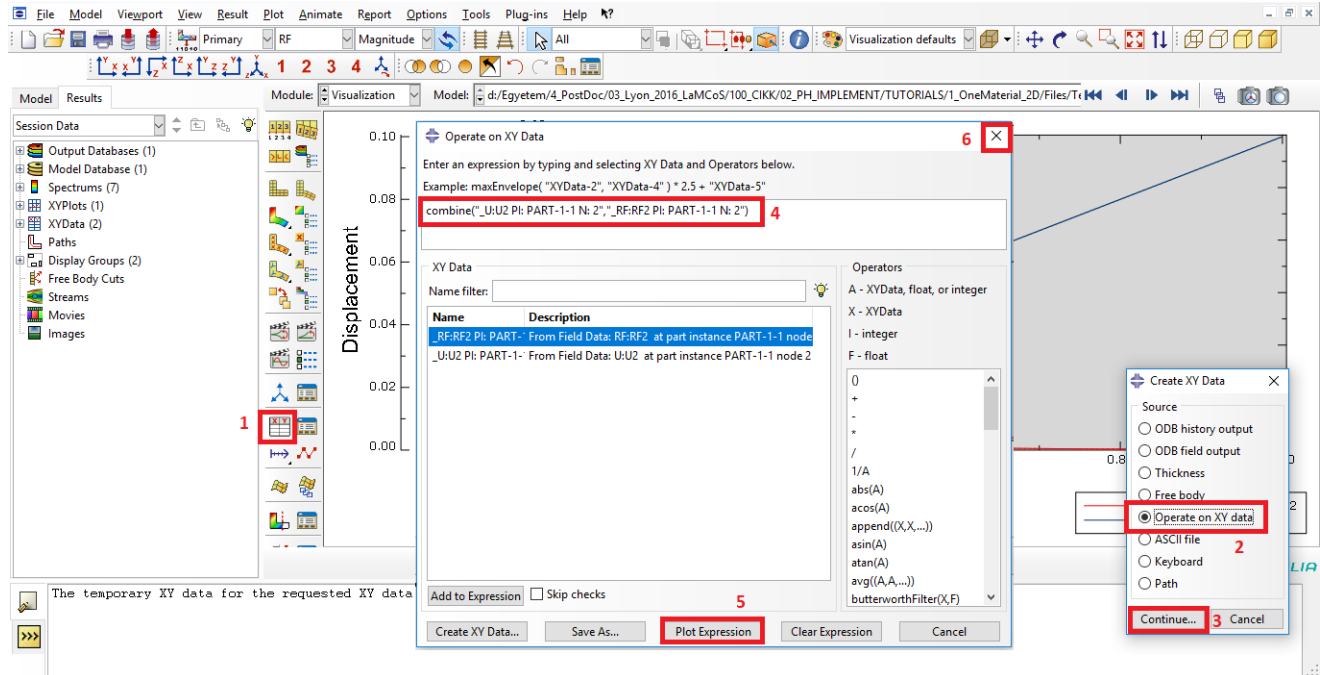

Figure 19 Displaying reaction force as a function of displacement.

Finally the reaction force results can be exported to file with the *Report/XY* menu in the main menu tab.
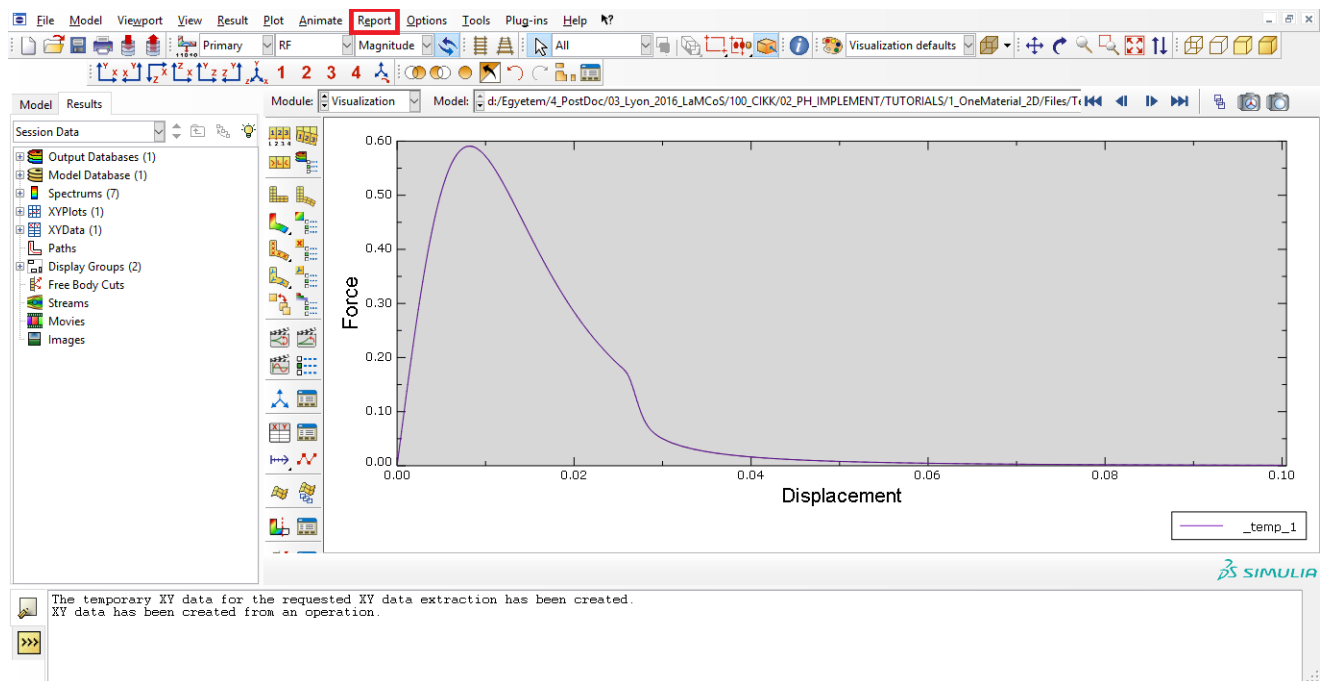

Figure 20 Final result: reaction force as a function of displacement.