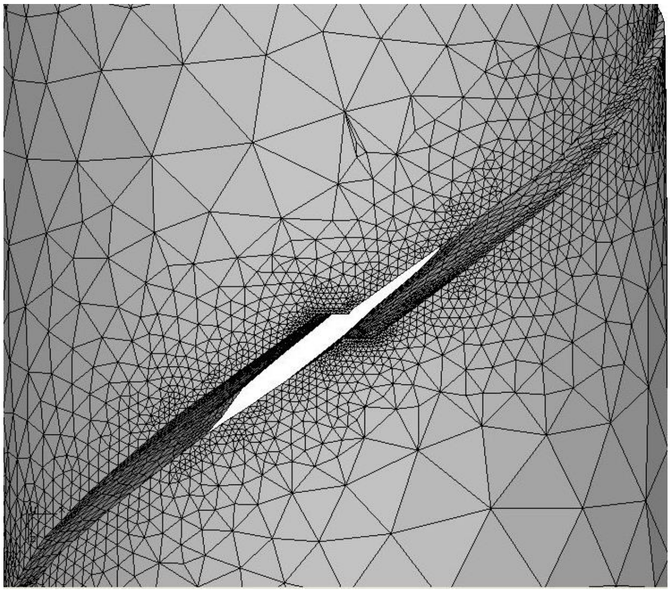
FRANC3D

User's Guide



Version 8.4

Fracture Analysis Consultants, Inc
www.fracanalysis.com

Revised: December 2023

Table of Contents:

1. Introduction

This document is a supplement to the Reference and Tutorial documents. It describes underlying features of the software and is intended to help more experienced users.

2. FRANC3D Startup Initialization

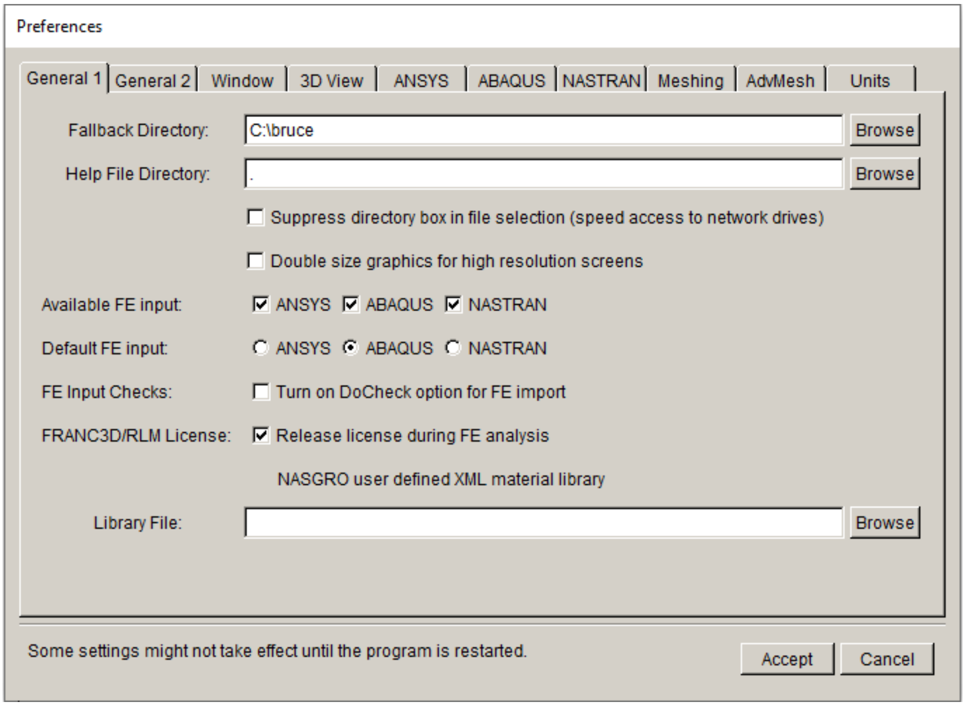
When FRANC3D is started, it reads a file to initialize internal data and preferences. The file can be edited with any text editor.

For Linux, the file is: *franc3d.rc*
It is in the user's home folder (*i.e.*, /home/user-name/.franc3d.rc)

For MS Windows, the file is: *franc3d.ini*
It is in the user's home folder (*i.e.*, C:\Users\user-name\franc3d.ini).

The content of the file consists of blocks of data associated with specific settings. For example, the ANSYS analysis settings are identified by the label: `[f3d_ansys]`.
ANSYS specific settings follow the label.

The settings should match what is in the FRANC3D Preferences dialog, Fig 1.1; see Section 5.4 of the Reference document for a description of this dialog.



Fallback Directory: C:\bruce

Help File Directory: .

☐ Suppress directory box in file selection (speed access to network drives)

☐ Double size graphics for high resolution screens

Available FE input: ☑ ANSYS ☑ ABAQUS ☑ NASTRAN

Default FE input: ○ ANSYS ● ABAQUS ○ NASTRAN

FE Input Checks: ☐ Turn on DoCheck option for FE import

FRANC3D/RLM License: ☑ Release license during FE analysis

NASGRO user defined XML material library

Library File:

Some settings might not take effect until the program is restarted.

Figure 1.1 Preferences dialog with General tab displayed.

For example, the MS Windows version of the file might contain the following information:

```
[prog_defs]
help_dir="F:\\current\\FRANC3D_Docs "
def_dir="F:\\tmp"
fe_input=ABAQUS
suppress_dir_box=false

[avail_codes]
codes=ANSYS|ABAQUS|NASTRAN
```

The “prog_defs” and “avail_codes” correspond to settings under the **General** tabs in Fig 1.1.

```
[SETTINGS]
iconview=12582912
doublesize=false
selforecolor=#ffffff
forecolor=#000000
userheader=50
nameheader=200
width=810
directory="F:\\current\\F3D_v7_models\\extra\\Lug"
dateheader=150
sorting=0
normalfont=helvetica,90,bold
typeheader=100
backcolor=#ffffff
hilitecolor=#ffffff
itemspace=104
shadowcolor=#8b8984
bordercolor=#000000
basecolor=#d4d0c8
sizeheader=60
height=500
attrheader=60
dirwidth=300
selbackcolor=#0a246a

[f3d_linewidth]
linewidth=1

[f3d_3d_color]
Background Color=#ffffff
3rd Highlight=#cccc00
Text Color=#000000
```

```
Polygon Color=#b2b2b2
Vector Color=#000000
2nd Highlight=#00cccc
1st Highlight=#cc0000
Marker Color=#000000

[f3d_3d_view_func]
Recenter=29
Zoom=4
Zoom/Spin=3
Group Select=49
Select=17
Rotate=1
Back Clip=20
Spin=4
Front Clip=18
Pan=2

[File Dialog]
style=12582912
width=500
height=300
showhidden=0

[f3d_startup_size]
width=1155
height=864
def_width=900
def_height=675
```

The “settings” and other options above correspond to settings in the **Window**, **3DView** and **Display** tabs in Fig 1.1.

```
[Visited Directories]
0="F:\\current\\F3D_v7_models\\extra\\Lug"
1="F:\\current\\F3D_v7_models"
2="F:\\current"
3="G:\\"
visiting=0


[f3d_ansys]
executable=C:\\Program Files\\ANSYS Inc\\v222\\ansys\\bin\\win64\\ANSYS222.exe
db_mem=0
num_procs=1
license=ansys
```

```
solver=0
output_nset=1
total_mem=0
include_path=0
delete_file_list=bcs,emat,esav
delete_files=1

[f3d_nastran]
executable="C:\\Program Files\\Siemens\\SimcenterNastran_2206\\bin\\Nastran.exe"
pyramids=1
front_elems=0
include_path=0
delete_file_list=master,dball
delete_files=1

[f3d_abaqus]
executable="C:\\SIMULIA\\Commands\\abq2022.bat"
ask_delete=0
num_procs=2
output_nset=1
output_frequency=0
include_path=0
delete_file_list=cid,com,stt
delete_files=1
```

The “f3d_ansys”, “f3d_nastran” and “f3d_abaqus” blocks correspond to settings in the respective analysis code tabs in Fig 1.1.

```
[f3d_meshing]
coarsen_crack_mouth=1
write_files_only=0
max_backtrack_restarts=4
local_surface_refinement=0
max_volume_elements=500000
volume_mesher=franc3d

[f3d_advanced_meshing]
surface_refinement_factor=1.2
volume_octree_refine_factor=2.6
volume_optimal_sphere_factor=0.75
volume_optimal_size_factor=1.375
surface_boundary_factor=0.3
uniform_refine=0
```

The “f3d_meshing” and “f3d_advanced_meshing” blocks correspond to settings in the respective meshing tabs in in Fig 1.1.

Most data-blocks listed above correspond with tabs in the Preferences dialog. For some data-blocks, such as [f3d_3d_view_func], it is easier to change the settings using the FRANC3D GUI.

For some settings, it might be necessary to change the settings by editing the file. For example, if FRANC3D does not list the user’s ANSYS license string, the string can be set in the resource file.

Settings from older versions of FRANC3D will be ignored if they are no longer used.

3. FRANC3D Files and Archiving Simulations

Many files can be generated during a crack growth simulation. For each step of crack growth, FRANC3D will save a restart file along with the analysis finite element (FE) files. FRANC3D automatically names (with numbers) the crack growth step files using the user-supplied base-name plus: *_STEP_###*. The numbering typically starts with '000' for the initial crack front.

The analysis codes create many files that do not need to be retained. There is a setting (delete_files=1) in the "f3d_ansys/abaqus/nastran" resource block (see Section 2) that allows FRANC3D to delete (most of) the unneeded analysis files. FRANC3D will not delete all the analysis files; for instance, for ANSYS the *.db* and *.rst* file are not deleted, as these can be read into ANSYS to display the crack model and corresponding results.

The various file types are described in Section 3.1. Section 3.2 gives options for archiving the simulation while reducing disk storage requirements.

3.1 File Extensions

A list of the simulation file extensions is given here:

1. *.fdb* – FRANC3D restart file
   a) stores crack geometry, growth model, stress intensity factor (SIF) history,
   b) references other files described below, which are read upon restart also

2. *.cdb*/*.inp*/*.nas* – original uncracked FE model and cracked FE model files
   a) cdb is for ANSYS
   b) inp is for ABAQUS
   c) nas (or bdf or dat) is for NASTRAN

3. *.dtp/.pch* – results file with displacements, temperatures, contact pressures
   a) ANSYS and ABAQUS results are in a .dtp file
   b) NASTRAN results are in a .pch file

4. *.crk* – crack (flaw) geometry

5. *.log* (*.f3d*) – GUI session log file contains FRANC3D commands

6. *.sif* – SIF data
   a) for a single crack growth step, a single crack front, and a single load step
   b) for SIF along a path
   c) for SIFs for all crack growth steps for a crack front and load step

7. *.fcg* – fatigue crack growth data (SIF history and crack growth model)

8. *.cgp* – crack growth parameters, including crack growth rate data and load schedule

9. *.db* & *.rst* are the ANSYS FE analysis files that are not deleted by FRANC3D; these files can be read into ANSYS to view results

10. *.odb* is the ABAQUS FE analysis file that is not deleted by FRANC3D; this file can be read into ABAQUS CAE to view results
    a) the ABAQUS *.dat* file contains ABAQUS contour integral results and is not deleted

11. *.op2*/*.fno* are NASTRAN FE results files that can be used to view results

12. *.py* – Python script

13. *.txt* – several ASCII text files are created and some of these are important when archiving a simulation. For instance, the *RETAINED* text files created when importing a FE model should be saved.

3.2 Archiving Files

Users might save all their simulation files. However, even if the files are compressed, this easily can lead to TB's worth of data. The notes below provide ideas for reducing the amount of data that is archived.

You can retain the FRANC3D session log file along with the original uncracked FE input (*.inp*, *.cdb,* or *.nas*) file. If you have multiple session log files for a simulation, you can combine these into a single session.log (*.f3d* is an alternate extension that can be used instead of the generic *.log* extension).

If you are doing a local+global subdivision inside FRANC3D, you will have a *_RETAINED_ELEMS.txt*. Also, you might have a *_RETAINED.txt* file for the mesh facets/nodes that are retained. You should keep these two *.txt* files along with the LOCAL and GLOBAL *.inp* (or *.cdb* or *.nas*) files.

You can save the initial crack as a *.crk* file. Even though the initial crack might be described in the session log, saving this file is recommended especially if you do not save the initial crack restart *.fdb* file (discussed below). Note that if you save the initial crack restart file, you can extract the initial crack geometry information from the *.fdb* file (see Section 2.10 of the base Tutorial documents).

With the above files, you can repeat everything using the FRANC3D playback feature; note that fatigue cycle integration (and corresponding display) cannot be replayed.

The initial uncracked FE files and the FRANC3D session log represent the minimum for archiving, but this will require re-running all the crack growth steps.

For bigger more complex models that take significant runtime, you could also keep the initial and final crack step files. This includes the *.fdb*, *.inp* (or *.cdb* or *.nas*), and *.dtp* files. Sometimes it is helpful to keep a few intermediate steps as well.

The final crack-step *.fdb* file contains all the crack growth history data. With this, you can re-extract the SIF history or re-compute cycles without having to rerun the entire simulation.

Another alternative is to keep all crack step files except for the FE analysis files; the FE analysis results files (*.rst*, *.odb* or *.op2*) are usually the largest. You could also keep the initial and final crack step FE files but compress them.

The analysis results files (*.odb* for ABAQUS or *.rst* /*.db* for ANSYS or *.op2/.fno* for NASTRAN) can be moved into a subfolder and saved (and compressed) until you are finished with the simulation. Those files usually take the most disk space, but they are useful for extracting results such as color contours of displacement or stress for the full model, which might be required for reports or presentations.

For example, consider the basic tutorial using ABAQUS files and using the local+global extraction. The minimum archive consists of:

1) *Abaqus-Cube.inp,*
2) *Abaqus-Cube_RETAINED_ELEMS.txt* and
3) *session.log*.

The ‘RETAINED’ .txt file contains the list of elements in the local portion. The first command in the session log will import and divide the model using this file. The rest of the simulation will proceed without any additional files. Edit the *session.log* file as needed, if you want to change the initial crack, for example.

4. Crack and Model Surface Geometry

This chapter describes how FRANC3D manages the underlying geometry of the model and crack surfaces. Crack insertion is based on geometric intersections of the crack surface, crack front template and model surfaces.

The first section describes how model surface geometry is created from the imported FE model.

4.1. Creating Model Geometry

Starting from an uncracked 3D FE volume mesh, the exterior surface facets of the volume mesh are determined, and the model surface geometry is approximated from this faceted-surface mesh.

Extracting a local submodel from a full model reduces and simplifies the mesh-to-geometry process. Fig 4.1 shows the full model of a simple “test specimen” along with the highlighted collection of elements that comprise the local model. The local submodel is converted to FRANC3D geometry while the rest of the full model remains “as-is” in a separate “global” FE model file.

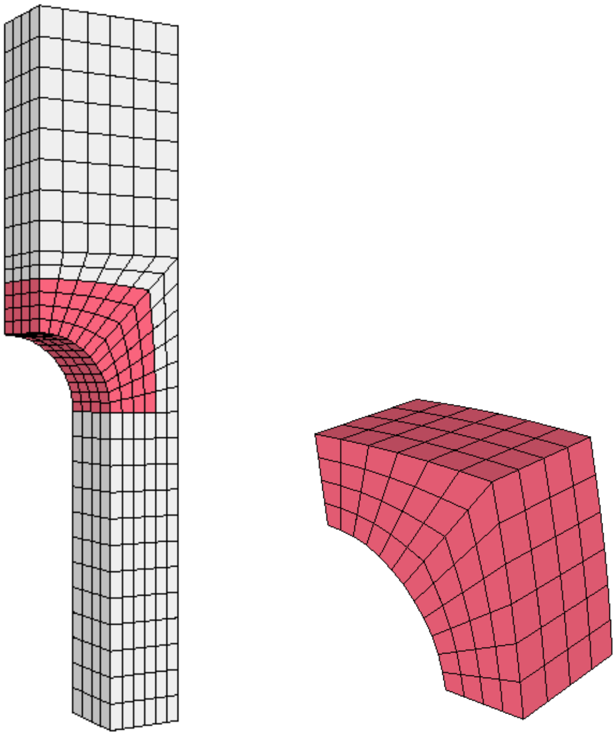


Figure 4.1 Full 3D model with highlighted local submodel.

Converting the 3D volume elements to FRANC3D geometry follows these steps:

1) compute the weighted average surface-facet normal at all FE nodes,
2) define one or two triangular Bezier patches for each FE facet,
3) identify "topological" edges, and group together patches that form logical faces.

Fig 4.2 shows the local submodel surface triangular patches. Each quadrilateral facet of the original brick mesh is split into two patches. If the original mesh is comprised of tetrahedral elements, the surface mesh would consist of triangles, and these would be used directly. Fig 4.3 shows the underlying Bezier triangle geometry with 10 control points; planar and curved surfaces can be represented.

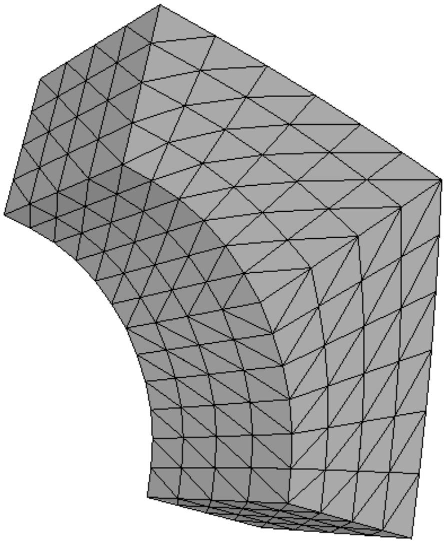


Figure 4.2 Triangular patches defined from the surface facets of 3D elements.

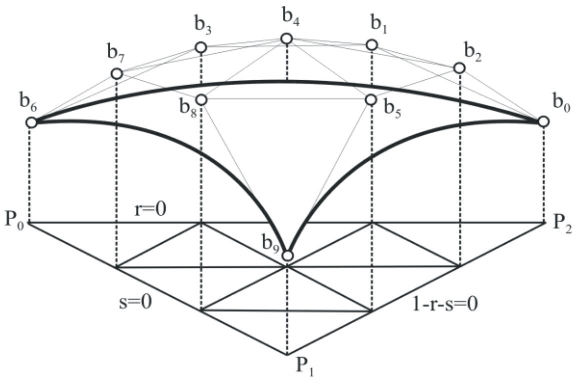


Figure 4.3 Triangular Bezier patch with control points.

The angle between adjacent patches can be used to define the topological boundaries of collections of patches. Fig 4.4 shows the “logical faces” for the submodel from Fig 4.2. Section 4.4 describes how one can modify the ‘kink’ angle to improve the topological surfaces for some models. The lower right corner of Fig 4.4 shows an extra edge; it is a display artifact that is shown because the surface mesh facets, on the two adjacent cut-surfaces, have been retained.

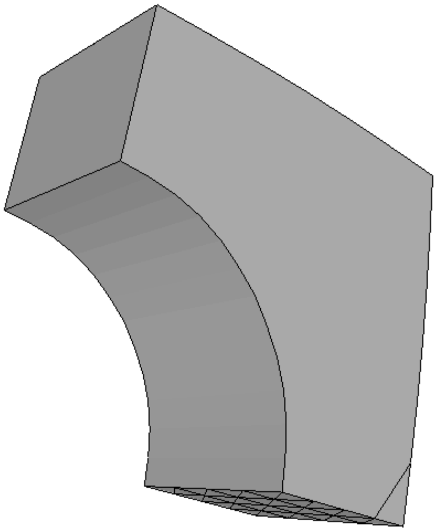


Figure 4.4 Topological faces of the submodel.

In FRANC3D, the local submodel is merged with the “global” FE model after crack insertion and remeshing. The best method for merging is to ‘glue’ the nodes and element faces on the cut-surfaces. Consequently, the mesh facets on the cut-surface are normally retained, Fig 4.5, and FRANC3D uses these facets for the surface mesh when remeshing.

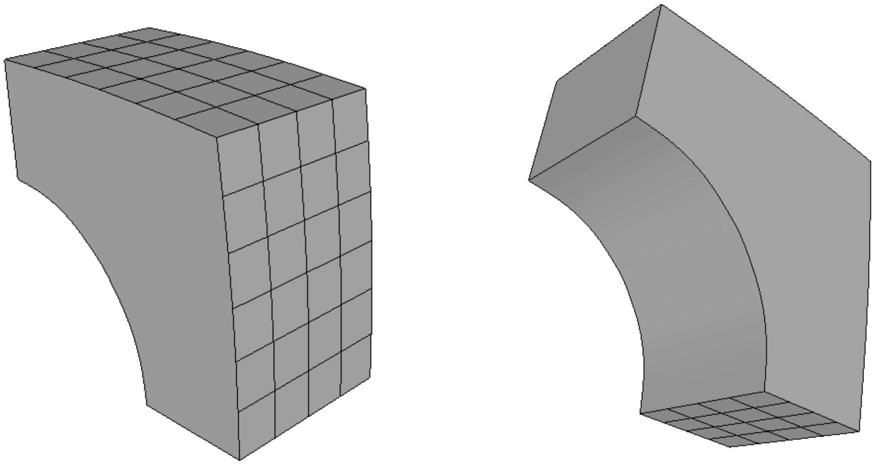


Figure 4.5 Topological faces of the submodel with cut-surface mesh facets retained.

4.2 Crack Insertion

The model shown in Fig 4.5 includes both geometry and topology information. A crack or flaw, which has its own geometry and topology, is inserted into this model.

Flaw surfaces are defined using a collection of Bezier patches. Crack front edges can be curves (cubic splines) or straight lines, and the flaw surface can be planar or non-planar. Fig 4.6 shows a penny-shaped crack consisting of a collection of triangular Bezier patches being inserted into the local submodel; the triangular Bezier patches of both the local model and crack surface are shown.

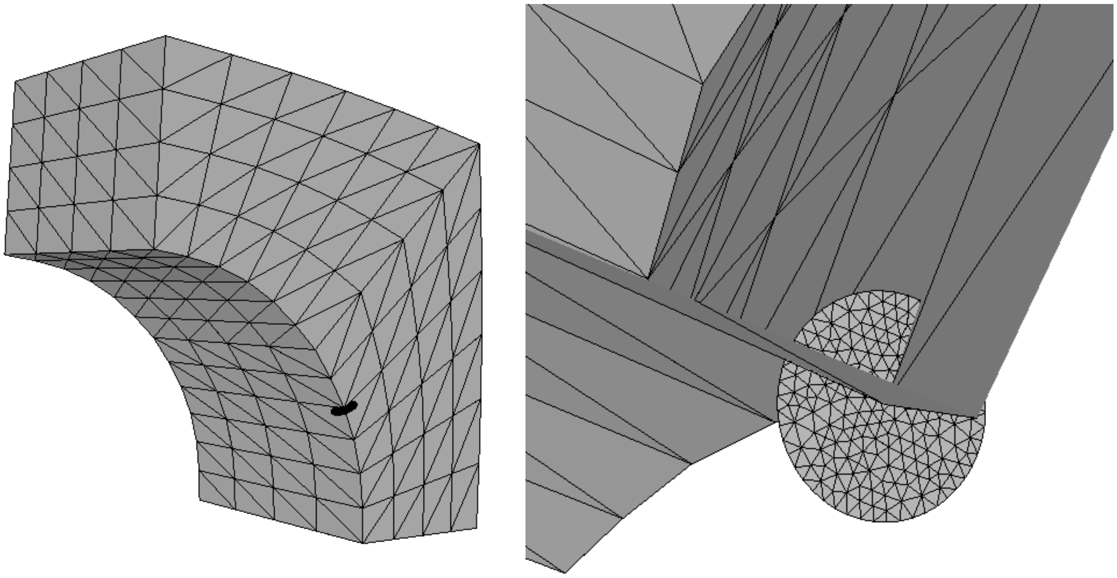


Figure 4.6 Flaw geometry inserted into model geometry.

Surface-surface intersections are determined so that only the portion of the crack geometry that falls inside the model is retained when building the composite geometry for remeshing.

It is noted here that cracks are extended by re-inserting the “extended” crack geometry into the original uncracked model. Fig 4.7 shows the original penny-shaped crack plus two steps of crack growth. The new extended geometry is added to the previous geometry; the crack surface is typically non-planar.

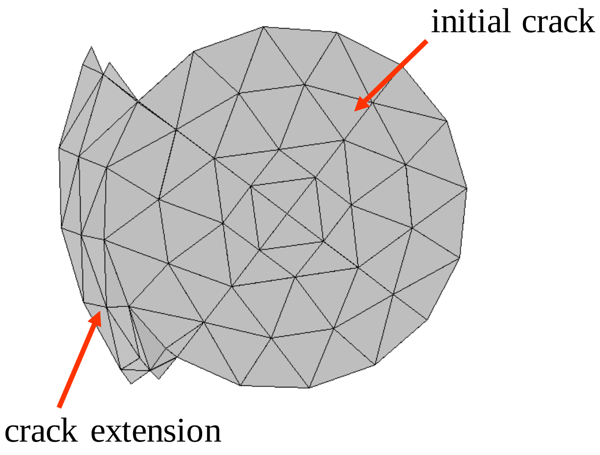


Figure 4.7 Extended flaw geometry.

4.2.1 Crack Front Template

A crack normally has a template mesh along the crack front. The template is used to place well-shaped elements, a combination of brick and quarter-point-wedge elements, along the crack fronts. The template geometry, Fig 4.8, is tied to the flaw surface geometry, and also must intersect the model surface geoemtry for surface-breaking cracks.

The original flaw surface patches are split into as many pieces as needed to produce watertight connections between all geometric patches on the flaw and the template, Fig 4.8.

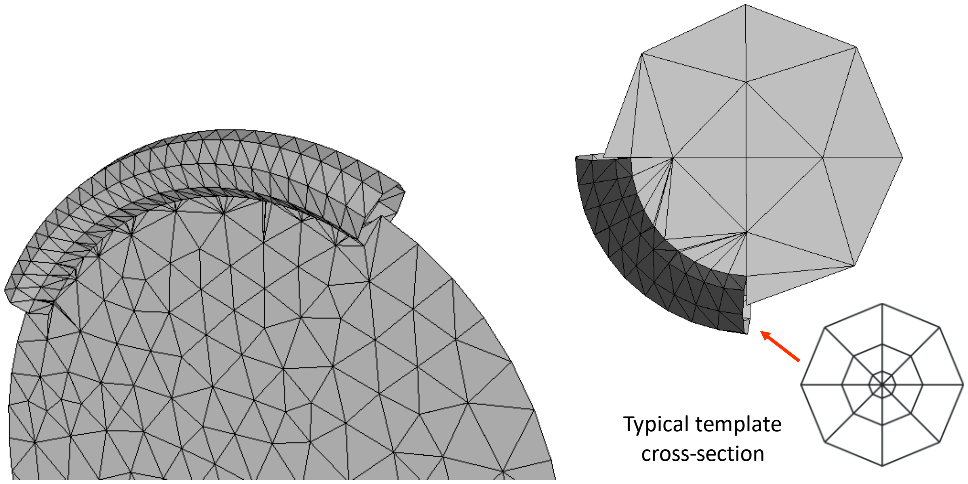


Figure 4.8 Template geometry tied to the flaw geometry.

The combined flaw and template geometry is added to the model geometry. First, surface-surface intersections are computed for all model and flaw patches. Patches are trimmed, so that only the geometry inside the model is retained, and then these are combined into composite objects. The trimmed patches are divided into triangular sub-patches to keep the model watertight. Fig 4.9 shows the trimmed penny-shaped crack inserted into the model, with patches divided to produce a watertight geometry. Note that in Fig 4.9, the template is “hollow”. The inside of the template is pre-defined and does not need to be included at this stage.

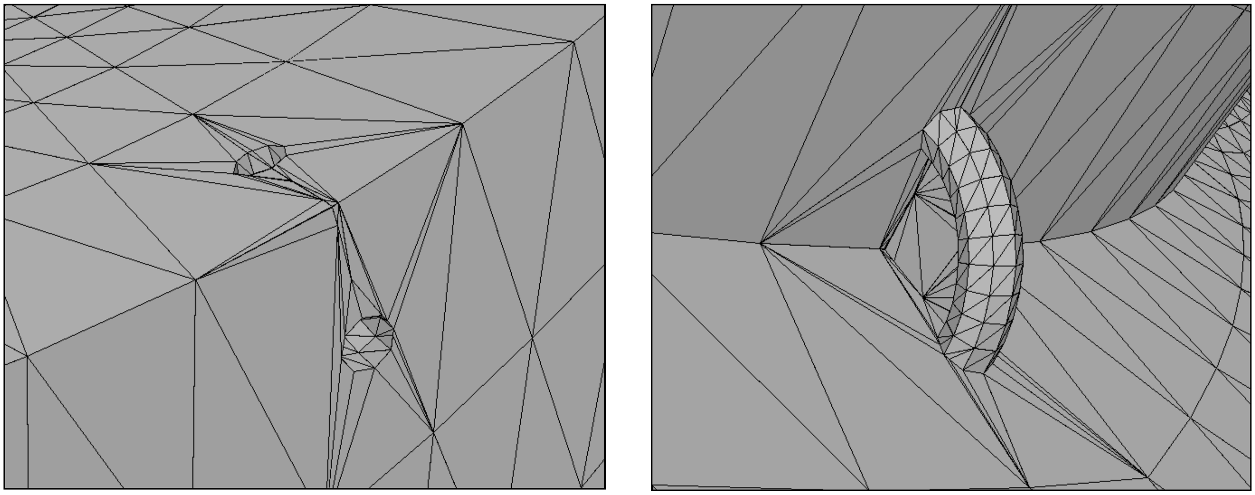


Figure 4.9 Trimmed flaw geometry added to the model geometry.

The collection of triangular patches into logical faces, for this corner crack model, is shown in Fig 4.10.

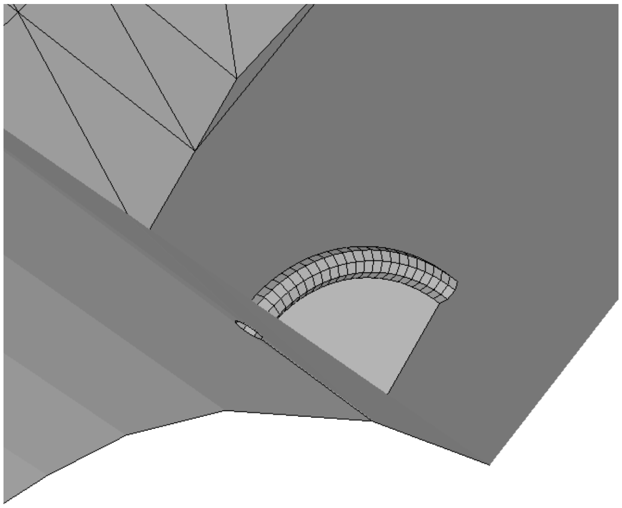


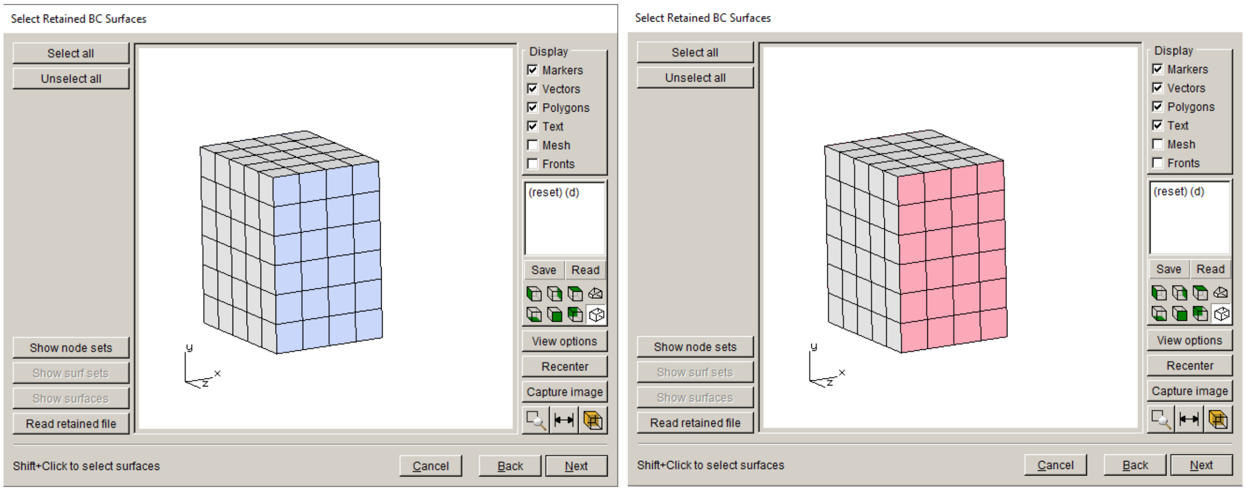
Figure 4.10 Topological faces of the flaw, template, and model.

4.2.2 Crack Insertion into Model Surfaces with Boundary Conditions

In Fig 4.5, the local model was displayed with the cut-surface mesh facets retained. A crack cannot be inserted (or propagated) into surfaces where the mesh is retained, so the local model should be large enough to allow for crack insertion and volume meshing around it. Local submodel extraction is discussed further in Chapter 5.

If a model surface has boundary conditions attached to it, FRANC3D allows the user to retain the mesh facets on those surfaces. However, a crack cannot be inserted into such a surface; FRANC3D will give a FLAW_IN_RETAINED_ERR error message.

If a crack is to be inserted into a model surface that has boundary conditions, then the surface mesh must not be retained. In such a case, the boundary condition data is mapped (rather than directly transferred) to the remeshed surface. Fig 4.11 shows the Select Retained BC Surfaces dialog; the blue color indicates that a boundary condition is attached to the surface. If the user elects to retain this surface, then the surface color is switched to red.



Select Retained BC Surfaces

Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh ☐ Fronts

(reset) (d)

Shift+Click to select surfaces

Select Retained BC Surfaces

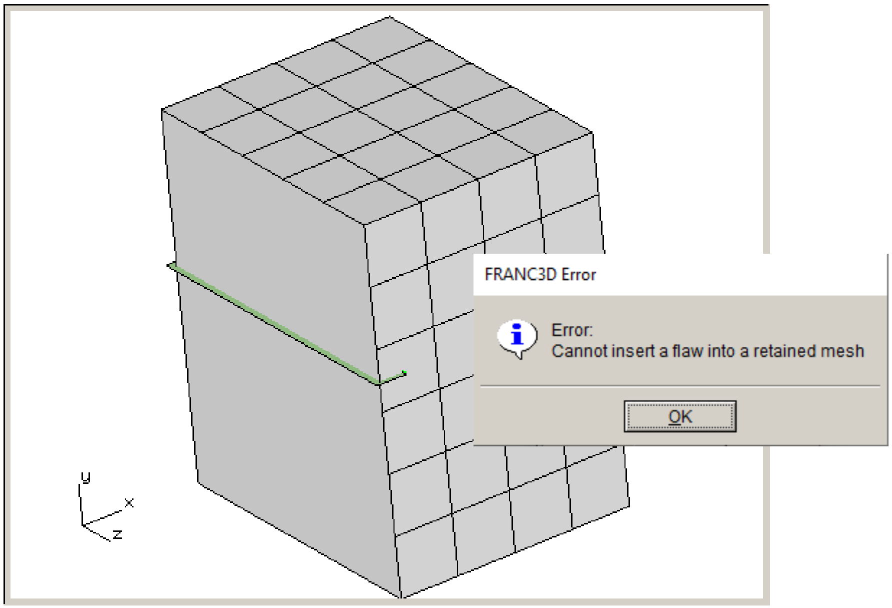
Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh ☐ Fronts

(reset) (d)

Shift+Click to select surfaces

Figure 4.11 Surfaces with boundary conditions – selected to be retained.

If one tries to insert an edge-crack, Fig 4.12, an error message is displayed and the crack insertion is aborted. Note the the upper suface mesh facets are also retained as these are part of the cut-surface. To get an edge-crack inserted at this location, do not retain the "blue" surface in Fig 4.11. The local model will appear as shown in the left panel of Fig 4.13; the mesh facets on the 'front' surface are not retained but the cut-surface facets on the 'top' surface are retained.



FRANC3D Error

Error:
Cannot insert a flaw into a retained mesh

Figure 4.12 Edge crack cannot be inserted due to the retained mesh.

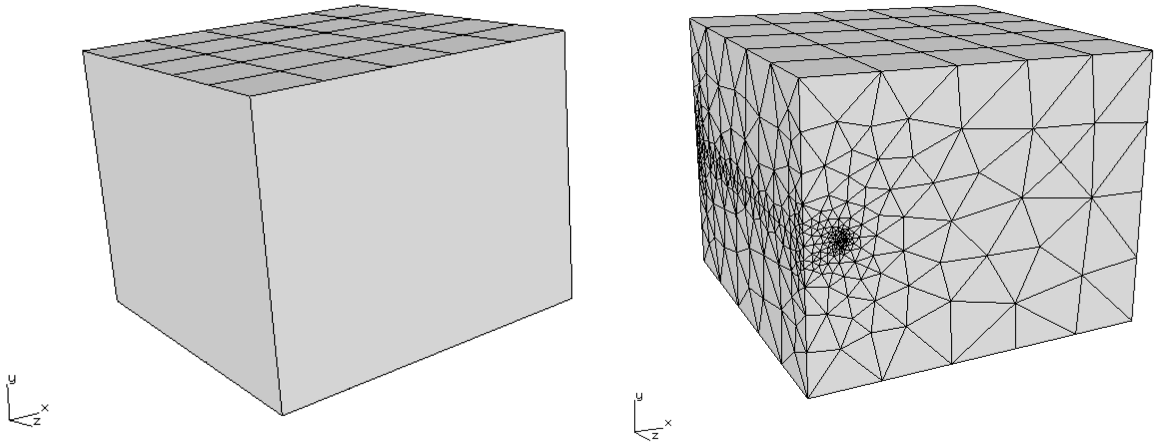


Figure 4.13 Edge crack inserted and remeshed.

FRANC3D will insert the edge-crack in this model and remesh (right panel of Fig 4.13), and automatically map the boundary condition data from the original mesh to the remeshed surface, Fig 4.14.

Note that transfer of boundary condition data is simpler and more precise if the surface mesh can be retained, but FRANC3D will automatically map the boundary condition data to the new mesh regardless. This model corresponds with Chapter 5 of the Benchmark document, where symmetry boundary conditions and plane-strain constraints are required.

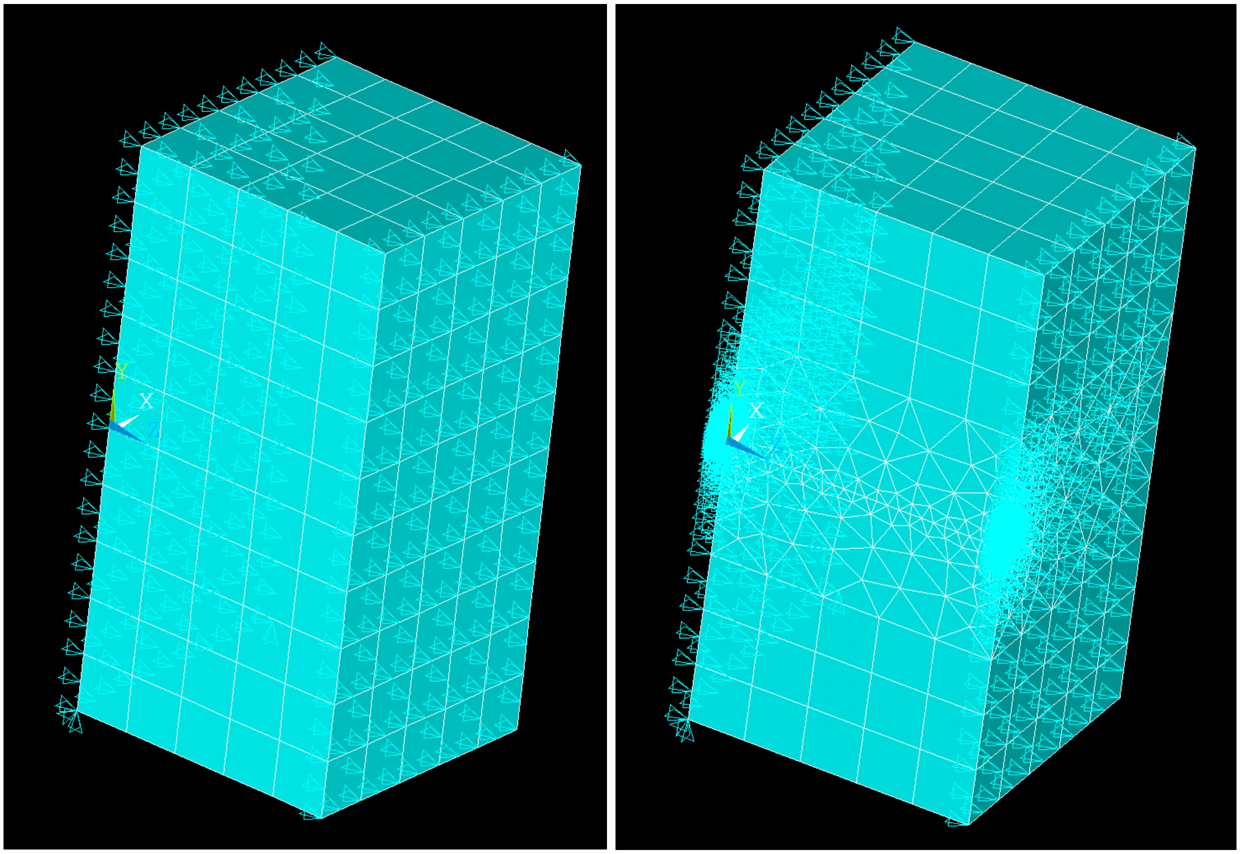


Figure 4.14 Constraint boundary conditions mapped from the original mesh (left) to the cracked-remeshed model (right).

4.3 Meshing Cracked Geometry

Once the geometric intersection (of the crack, template and model surfaces) is complete and the patches are organized into logical surfaces, meshing can be performed. First, surface meshes are generated for all logical surfaces using triangular facets (note that the template outer surface is already meshed). The surface mesh must conform to any retained cut-surface mesh, as well as all un-retained surface geometry. Fig 4.15 shows the surface mesh for the corner crack in the model (see Fig 4.10), minus the template mesh ends; Fig 4.16 shows additional views.

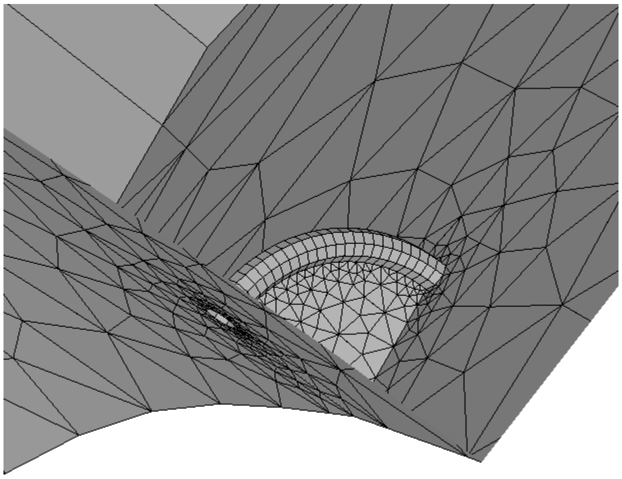


Figure 4.15 Surface mesh on the flaw and model surfaces.

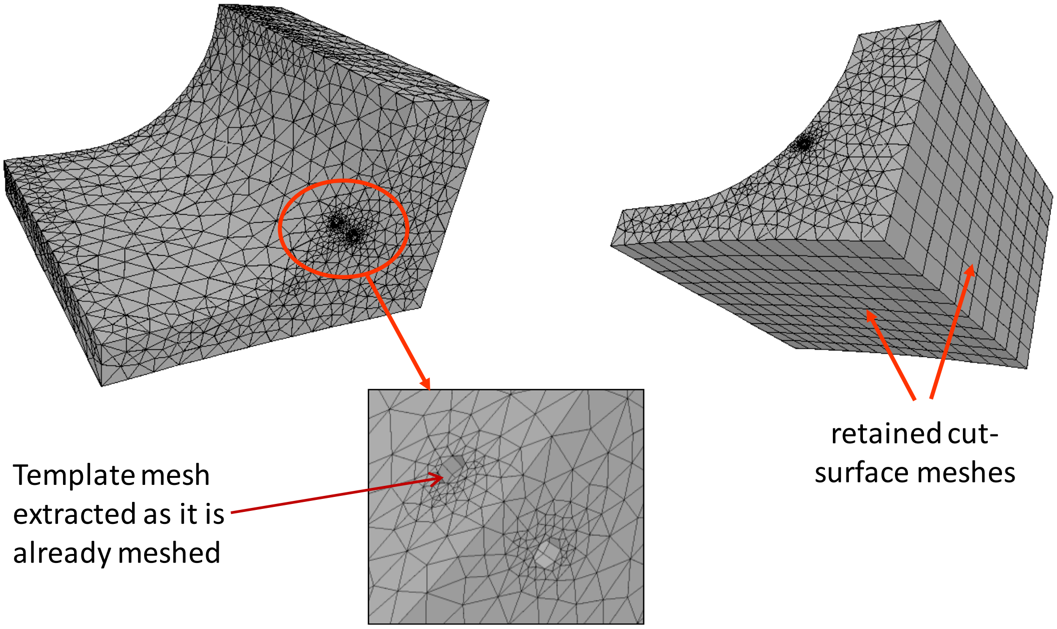

Template mesh extracted as it is already meshed
retained cut-surface meshes

Figure 4.16 Surface mesh for a corner crack.

The surface mesh is the starting point for the 3D volume meshing. Quadrilateral faces will get a pyramid element added first, which leaves an all-triangular surface mesh for the subsequent tetrahedral meshing. Pyramid elements are generated for compatibility between quadrilateral facets on template or retained cut-surface facets, as shown in Fig 4.17. It is noted that pyramid elements will have a finite volume (FRANC3D does not produce flat pyramids).

Additional information on meshing and the ‘advanced’ settings for meshing are discussed in Chapter 7.

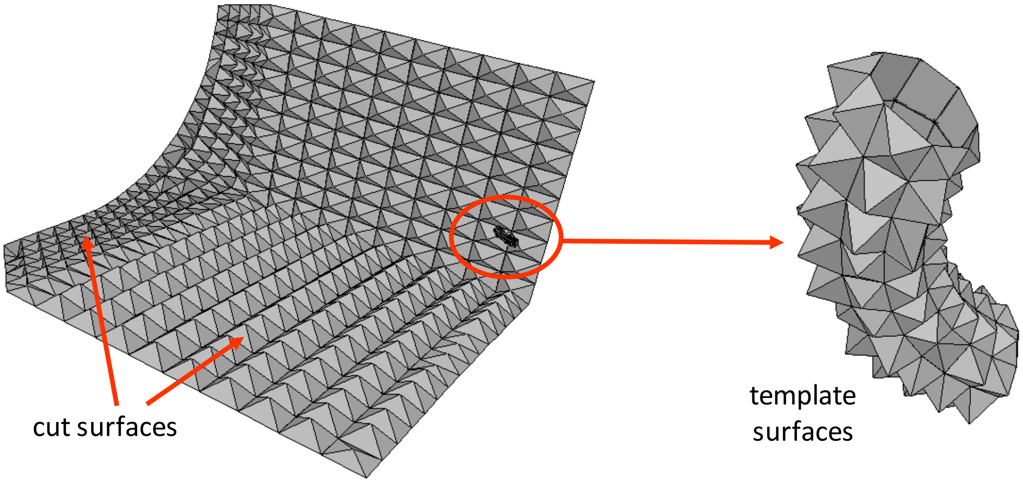


Figure 4.17 Pyramid elements attached to any quadrilateral mesh facet.

The crack front template volume mesh is pre-defined and will be added to the final volume mesh. The template mesh consists of wedge and brick elements, Fig 4.18; this is described in Section 6.1.15 of the Reference document.

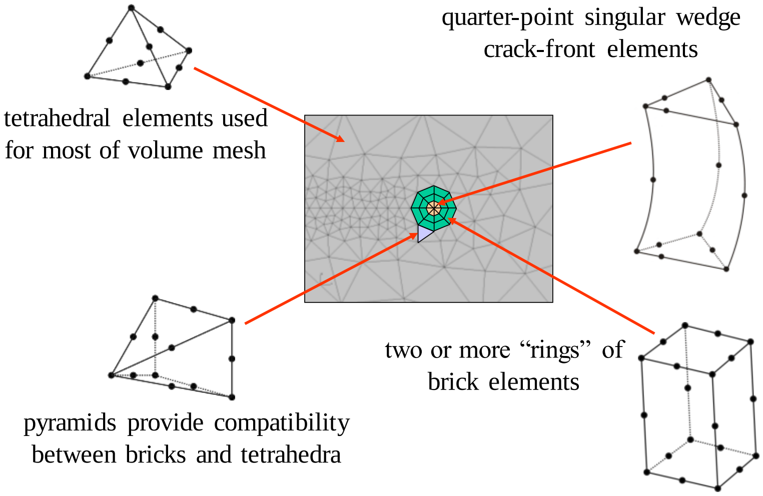

quarter-point singular wedge
crack-front elements
tetrahedral elements used
for most of volume mesh
two or more “rings” of
brick elements
pyramids provide compatibility
between bricks and tetrahedra

Figure 4.18 Crack front template elements.

4.4 Edge Wizard

The FRANC3D **Advanced** menu includes the **Edges Wizard…** option. The edge extraction dialog is often required for models with no clearly defined geometry boundaries. For example, an airfoil, Fig 4.19, typically requires that the user break the geometry into manageable pieces. The leading (and trailing) edge of the airfoil and the transition to the platform might not have edges that meet the default kink angle threshold. The result is a large complex surface that is more difficult to remesh. Quite often, there are small pieces of surfaces and dangling edges, Fig 4.20, that lead to meshing failures (typically the software will get stuck at surface meshing).

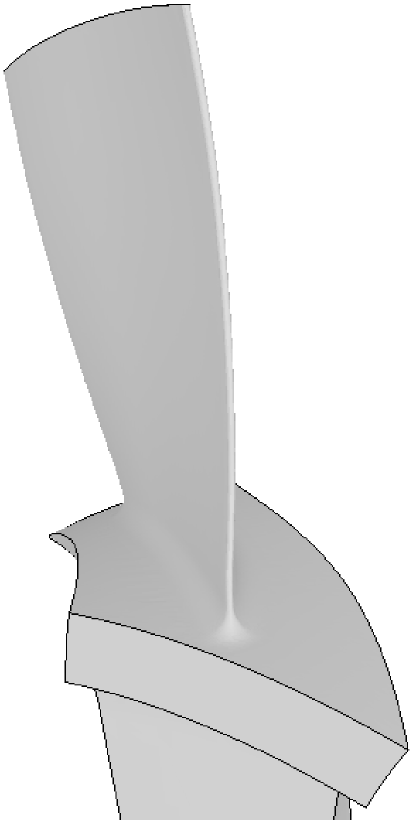


Figure 4.19 Airfoil imported – prior to using edge extraction dialog.

By adjusting the kink angle threshold, Fig 4.21, one can create additional boundary edges that break the surface into more manageable pieces and remove dangling surface-edges. Note that you will have to use the up/down arrow to get the angle threshold to change correctly. The only guideline is that you should use an angle that does not create dangling surface edges; you might have to carefully inspect the bounding edges that are displayed.

Note that you should rarely need to use the Edge extraction dialog. If FRANC3D is having difficulty creating the surface mesh during crack insertion, then you can try adjusting the angle threshold (or you can send us the *debug.tst* file to examine). Section 17.3 describes potential error messages that you might encounter in these cases.

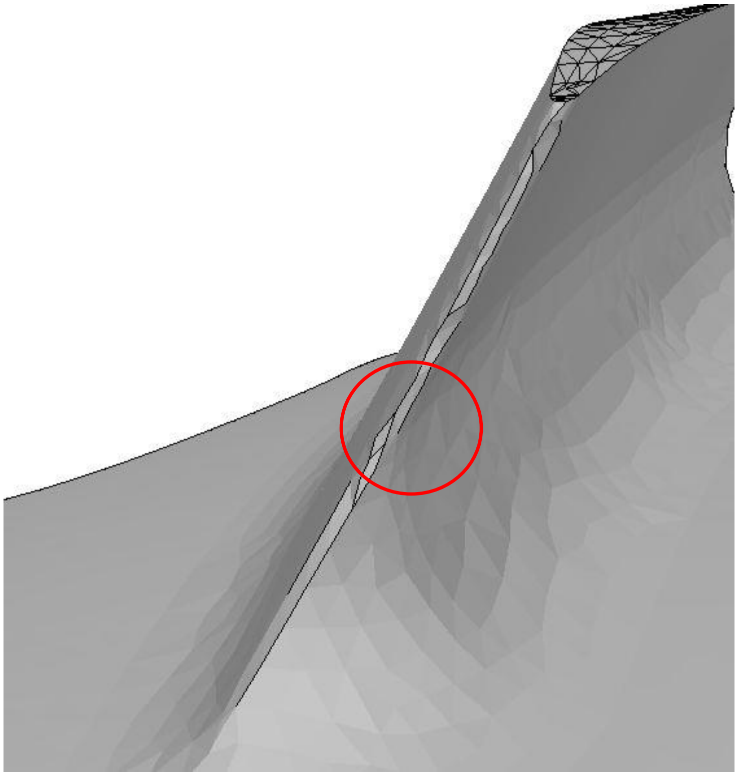
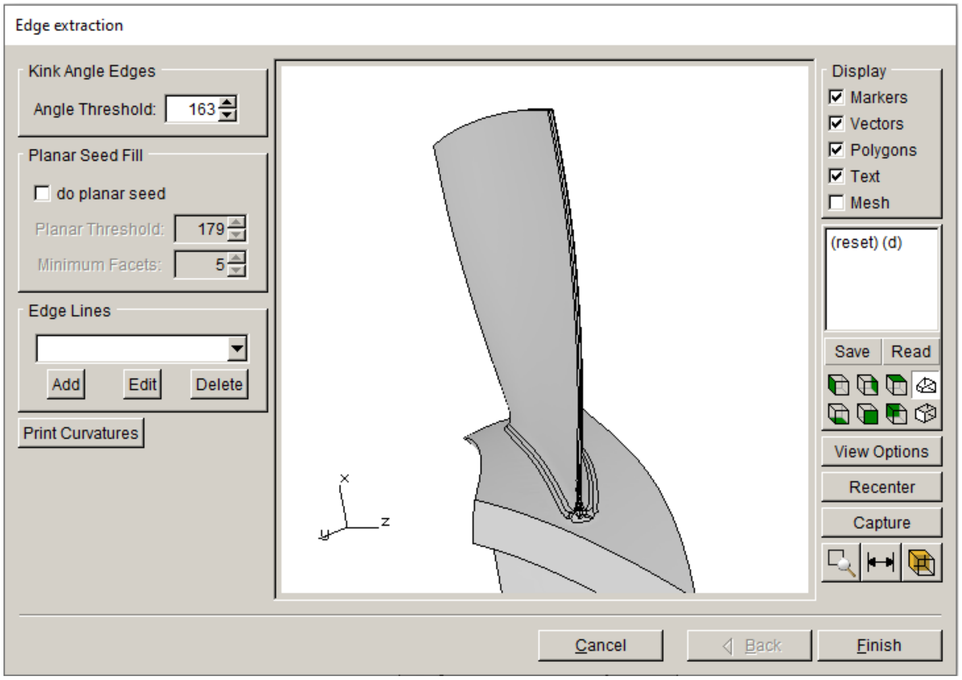


Figure 4.20 Red circle highlights a dangling edge on the surface.



Edge extraction

Kink Angle Edges

Angle Threshold: 163

Planar Seed Fill

☐ do planar seed

Planar Threshold: 179

Minimum Facets: 5

Edge Lines

Display

☑ Markers

☑ Vectors

☑ Polygons

☑ Text

☐ Mesh

(reset) (d)

Figure 4.21 Edge extraction kink angle adjusted to break the surface into multiple surfaces.

5. Local Model Extraction

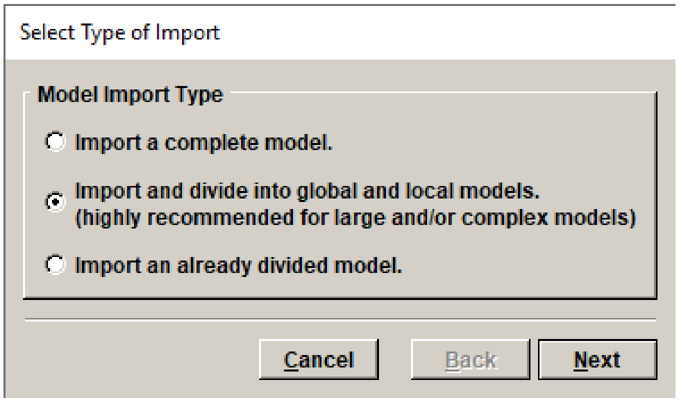
In Section 4.1, it was stated that a local submodel, which is extracted from the full FE model, simplifies the geometry re-construction. It also simplifies the crack insertion process by limiting the number of patches that must be checked for intersection. Plus, it speeds up remeshing and simplifies the transfer/mapping of boundary condition data.

Unless you are working with a simple model (such as the tutorial cube model), then you should always extract a local submodel for crack insertion.

5.1 FE Model Import and Extraction

Crack insertion and growth is simplified by restricting the model size that is read into FRANC3D. However, the local submodel should be large enough to capture the desired crack growth (although you can choose a larger submodel later if the crack becomes too large).

There are three options for importing the original uncracked FE model: 1) import the complete model, 2) import and divide the model into local and global using FRANC3D tools, and 3) import an already divided model, Fig 5.1.



Select Type of Import

Model Import Type

○ Import a complete model.

● Import and divide into global and local models.
(highly recommended for large and/or complex models)

○ Import an already divided model.

Figure 5.1 FRANC3D model import dialog.

If the model is simple and the crack growth will be large, then the first option might be best. We use this option in the tutorials, where we have a relatively large surface crack in a cube. However, for most models, you will want a smaller local model.

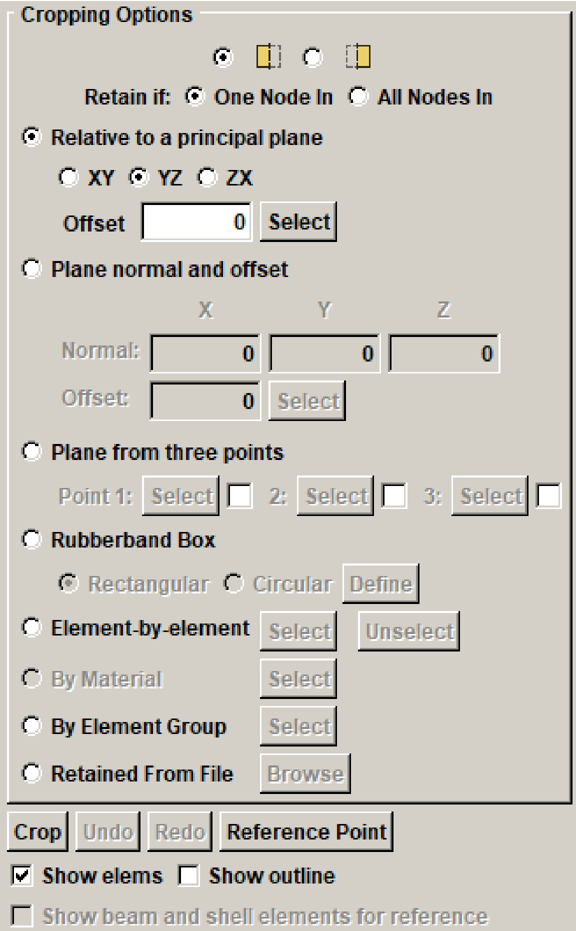
The FRANC3D tools for extracting portion(s) of the full model, Fig 5.2, are described in Section 4.5.2 of the Reference document. The By Material and By Element Group options require that the user define sets (or components) of elements in the analysis code, but this is often the case anyway.

One thing to note is that FRANC3D will condense materials; if there are two (or more) materials with different material IDs but identical properties, FRANC3D will condense this to a single material ID. You can alter the material property data slightly for one of the materials to prevent the condensing.

Once an element set has been selected and cropped, the resulting local and global portions are saved along with a file with the base file name followed by *_RETAINED_ELEMS.txt*. This file contains the element IDs for the local model. This file can be used with the Retained From File option, Fig 5.2, and it can be re-used during playback of a session log.

You can keep just the original FE model and the *_RETAINED_ELEMS.txt* file to recreate the local and global model extraction, as described in Section 3.2, to archive a simulation.

The third option in Fig 5.1 can be used once _LOCAL and _GLOBAL FE files are saved (using the second option), or it can be used if the FE model is divided using other tools/software. In the second case, it is important to remember to collect the node sets (components) that define the cut-surface between the local and global portions so that these can be selected later when merging the cracked-remeshed local portion with the global portion.



Cropping Options

● ○

Retain if: ● One Node In ○ All Nodes In

● Relative to a principal plane

○ XY ● YZ ○ ZX

Offset 0 Select

○ Plane normal and offset

X Y Z

Normal: 0 0 0

Offset: 0 Select

○ Plane from three points

Point 1: Select □ 2: Select □ 3: Select □

○ Rubberband Box

● Rectangular ○ Circular Define

○ Element-by-element Select Unselect

○ By Material Select

○ By Element Group Select

○ Retained From File Browse

Crop Undo Redo Reference Point

☑ Show elems □ Show outline

□ Show beam and shell elements for reference

Figure 5.2 FRANC3D model cropping dialog.

The FRANC3D tools automatically create an AUTO_CUT_SURF for the local portion and a GLOBAL_CONNECT_SURF for the global portion. It is not necessary to use these same names when dividing the model using other tools; it is best to use two distinct names for the local and global, even if the node IDs are the same.

The size of the local model should be based on the extent of crack growth and the geometry of the model. Also, it might be advantageous to adjust the local model based on the boundary conditions – including contact, constraint, cyclic symmetry, *etc*. Keeping the boundary condition data in the global portion limits how much data needs to be transferred or mapped to the cracked-remeshed portion.

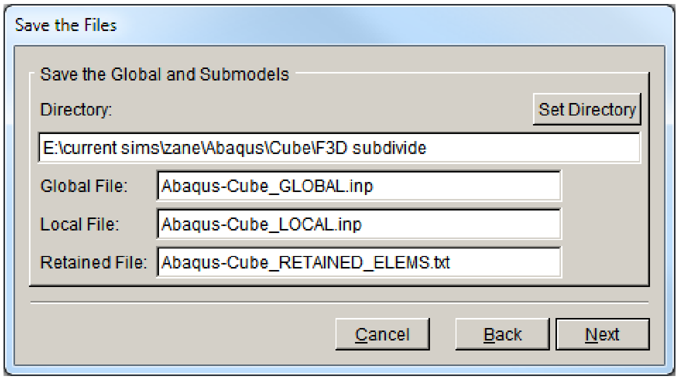
5.2 Extracting a Larger Local Model

If the local model is too small to contain the crack growth that is needed, a larger local model can be extracted, and the current crack can be inserted into the larger portion. This is described in Section 2.10 of the base Tutorial documents. In that tutorial, the larger submodel consisted of the full cube. However, in practice, one would usually use the import and divide option discussed in the previous section to extract a larger local model.

One thing to note, when saving the new files associated with the larger local model, Fig 5.3, do not overwrite the initial local, global, and retained file names. You will have to edit the file names in the dialog before selecting **Next**. The existing crack steps will refer to the original local and global file names; you can edit any *.fdb* file to find this type of information:

```
ORIG_MODEL_NAME: test_Local.cdb
ORIG_MODEL_TYPE: ANSYS
…
STATICMETA
(
VERSION: 1
ANAL_TYPE: ANSYS
MESH_FILE: test_surf_crk_STEP_049.cdb
RSLT_FILE: test_surf_crk_STEP_049_full.dtp
GLOBAL_FILE: test_Global.cdb
)
```

The _Local and _Global are not necessary when reading the .fdb file back into FRANC3D if you are only going to compute SIFs. However, if you are going to grow the crack or rerun the FE analysis, then these files are required.



Save the Files

Save the Global and Submodels

Directory: E:\current sims\zane\Abaqus\Cube\F3D subdivide

Global File: Abaqus-Cube_GLOBAL.inp

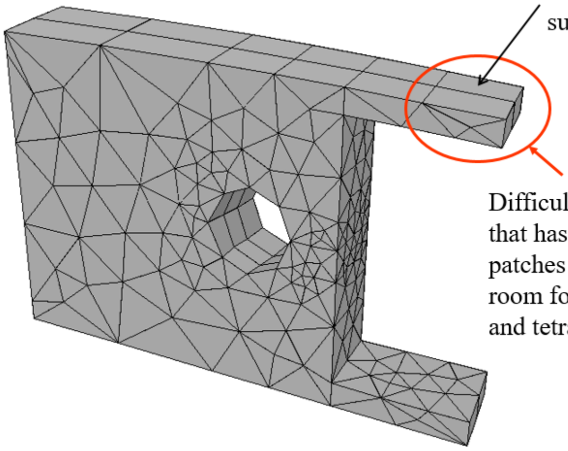
Local File: Abaqus-Cube_LOCAL.inp

Retained File: Abaqus-Cube_RETAINED_ELEMS.txt

Figure 5.3: Local/Global model save window

5.3 Extracting a Local Model that can be Remeshed

It is important to remember that there are limitations for volume meshing when extracting a local model. Fig 5.4 shows a local portion of a model that might cause the volume meshing to fail. In this case, the cut-surface quadrilateral facets are retained (on the top surface). Thus, pyramid elements are pasted onto these quad-facets. In the region circled in red, due to the size of the quadrilateral facets and the thin volume to be meshed, adding a reasonable pyramid might fail. If a pyramid is added successfully, the remaining region surrounding the pyramid might not be meshable.



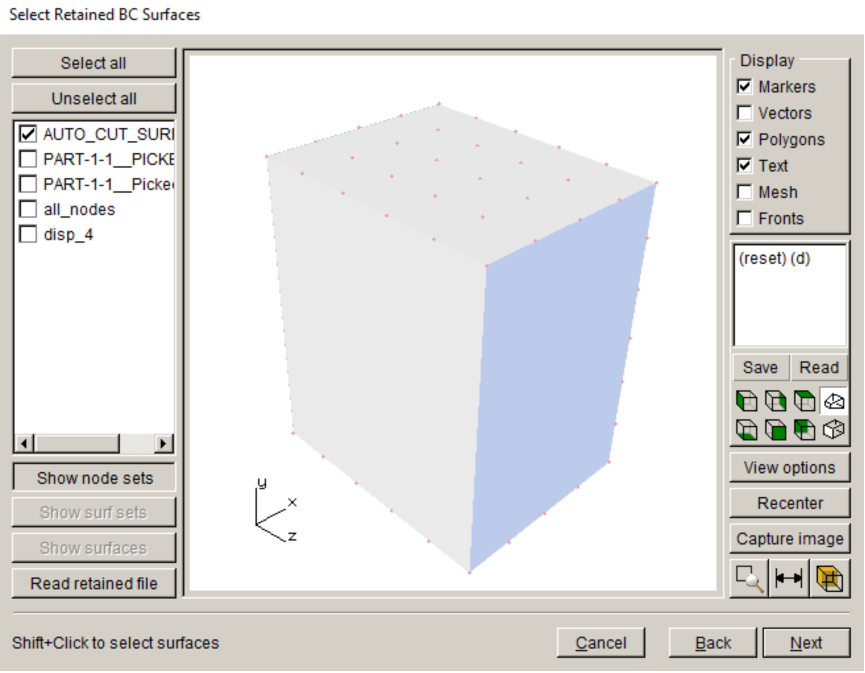
Retained cut-surface facet

Difficult to mesh thin volume that has large quadrilateral patches retained as there is no room for well-shaped pyramid and tetrahedral elements.

Figure 5.4 A local sub-model that might cause volume meshing to fail.

There are a couple of options in this case, extract a different local model or do not retain the cut-surface facets. In the Select Retained dialog, click on the **Show Node Sets** button to display the node sets in the local model, Fig 5.5. This will include the AUTO_CUT_SURF set (assuming

FRANC3D extracted the local portion). This node set can be un-checked (it is checked by default) and the mesh facets on these surfaces will not be retained. After crack insertion and remeshing, the FE analysis will require Tie constraints between the local and global portions as node-merging will not be possible. Tutorial #3 (in the Tutorials 2-12 document) describes the local+global connection options.


Select Retained BC Surfaces

☑ AUTO_CUT_SURI
□ PART-1-1__PICKE
□ PART-1-1__Picke
□ all_nodes
□ disp_4

Display
☑ Markers
□ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

(reset) (d)

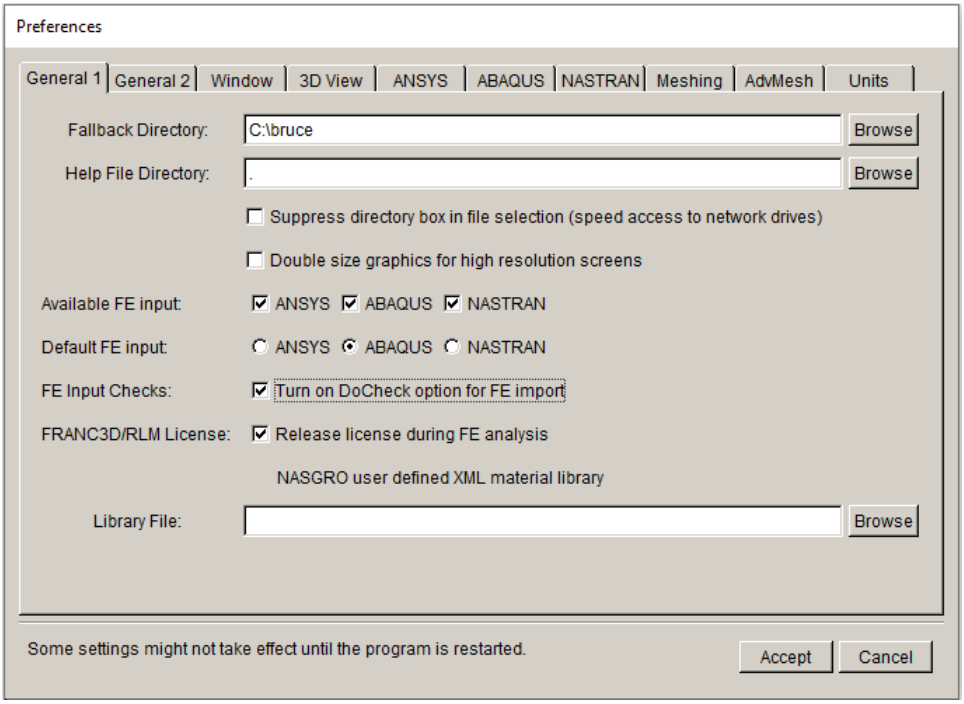
Shift+Click to select surfaces

Figure 5.5 AUTO_CUT_SURF node set.

5.4 Potential Sources of Error in FE Input

We have encountered cases where an issue with the original uncracked FE model causes errors during crack insertion and remeshing. One can turn on the FRANC3D internal checks by editing the Preferences, Fig 5.6. The Turn on Do Check option will enable an extra check box and dialog in the FE Import dialog, Fig 5.7.

The user can choose which checks are enabled by selecting the **Choose Checks** button in Fig 5.7. The dialog in Fig 5.8 lists the available checks. The checks are briefly described in the subsections below.



Fallback Directory: C:\bruce

Help File Directory: .

□ Suppress directory box in file selection (speed access to network drives)

□ Double size graphics for high resolution screens

Available FE input: ☑ ANSYS ☑ ABAQUS ☑ NASTRAN

Default FE input: ○ ANSYS ● ABAQUS ○ NASTRAN

FE Input Checks: ☑ Turn on DoCheck option for FE import

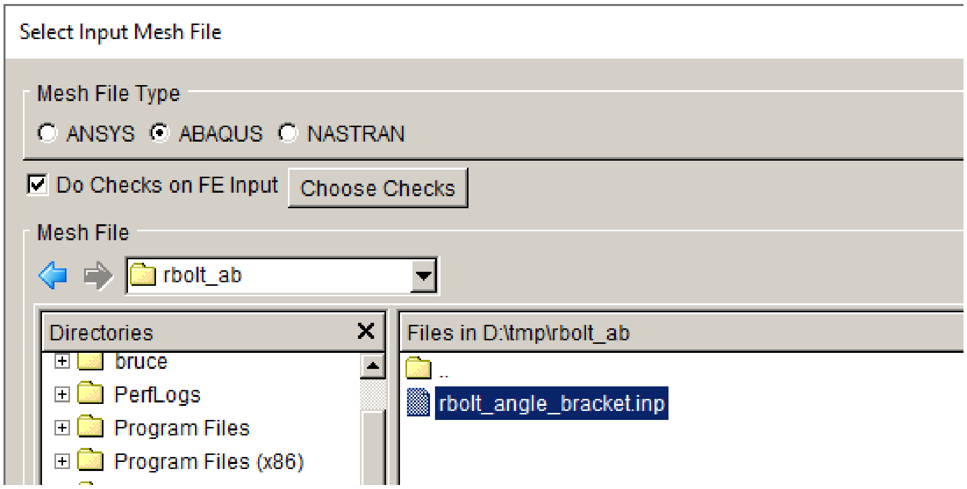
FRANC3D/RLM License: ☑ Release license during FE analysis

NASGRO user defined XML material library

Library File:

Some settings might not take effect until the program is restarted.

Figure 5.6 Preference settings for turning on the FE import DoCheck option.



○ ANSYS ● ABAQUS ○ NASTRAN

☑ Do Checks on FE Input

Files in D:\tmp\rbolt_ab

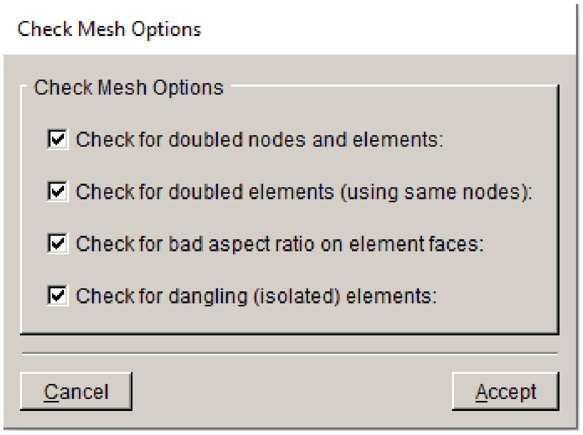
rbolt_angle_bracket.inp

Figure 5.7 FE import dialog with the DoCheck option enabled.

5.4.1 Duplicate Mesh

It is possible to create a 3D FE model with duplicate elements and nodes. For instance, ANSYS cyclic symmetry models might be doubled when archived to the *.cdb* file; you use the “cyclic,undouble” command prior to the “cdwrite” command to avoid this.

FRANC3D will import the doubled mesh, but crack insertion will fail, usually with an error message that states that the crack does not fall inside the model region.



☑ Check for doubled nodes and elements:

☑ Check for doubled elements (using same nodes):

☑ Check for bad aspect ratio on element faces:

☑ Check for dangling (isolated) elements:

Figure 5.8 Check mesh options.

5.4.2 Badly Shaped Elements

It is possible to create a 3D FE model with highly distorted or poorly shaped elements. If these elements are part of the model surface in the local model, they can cause crack insertion and remeshing to fail. For example, we have encountered an uncracked 3D FE mesh where one element was distorted and penetrated adjacent elements. FRANC3D was not able to create a valid watertight geometry of the model surface in this case.

A simple solution was to exclude the poor element and the adjacent elements from the local model. The poor element is then part of the global model and still might cause issues when solving the combined cracked and remeshed model.

It is difficult to imagine or illustrate all the potential errors that such meshes can create. We have tools to check the quality and integrity of the input mesh. If you encounter errors when importing or when inserting/remeshing a crack, you can send us the files to examine and debug. Section 17.3 describes potential error messages that you might see in these cases.

5.4.3 Unreferenced Nodes Far from the Local Model

It is possible to create a 3D FE model with unreferenced nodes – nodes that do not belong to any elements – that are located far from the local elements. FRANC3D determines tolerances based on the model’s bounding box and facet sizes. If there are nodes far from the elements, the bounding box dimensions and tolerances might be highly distorted; this can cause the crack insertion to fail.

Any node that is not tied to an element should be automatically removed prior to the crack insertion and remeshing, which should prevent this error from occurring. Thus, there is no specific check for this error.

5.4.4 Dangling or Isolated Elements

When creating the local submodel, users should try to make sure there are no dangling or isolated elements, especially brick elements. An isolated element is a single element that is not attached to any of the other elements. A dangling element is an element that is connected to other elements at a single node or an edge. A dangling tetrahedral (tet) element is usually not an issue as a single tet element can be added during remeshing. A dangling brick element, however, must be meshed with pyramid and tet elements, which might not be possible.

5.5 Surface Mesh NOT Retained

As described in Section 4.2.2, the surface mesh facets do not have to be retained on the local model, and if a crack will be inserted into a surface, then the surface mesh must not be retained on that surface. In such a case, the boundary condition data is mapped (rather than directly transferred) to the remeshed surface.

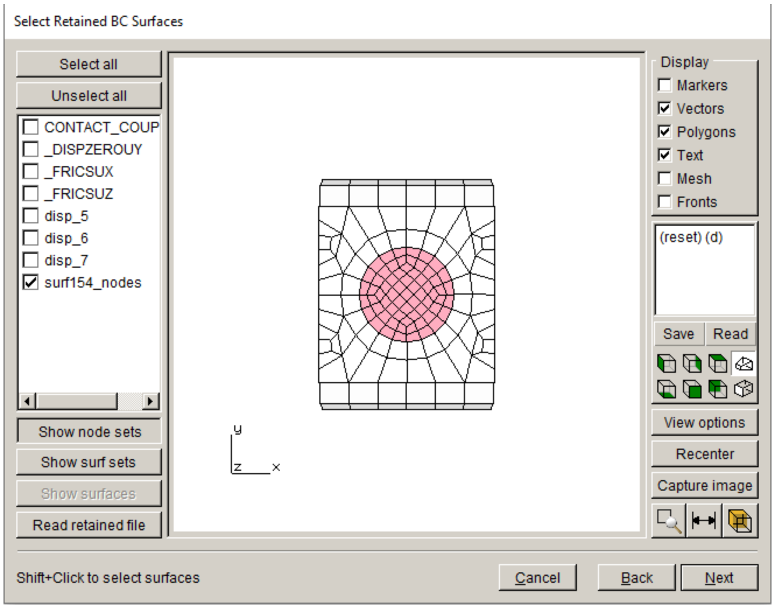
In practice, the transfer of boundary condition data is simpler and more precise if the surface mesh is retained, but sometimes this is not possible. FRANC3D will automatically transfer or map the boundary condition data to the new mesh regardless.

If a model has boundary conditions, contact, constraint, *etc.* applied to a surface and the crack will be inserted into this surface, it is best to have a 'named' surface. The original surface information – node IDs and associated element facets – will be saved as part of the named surface. Upon crack insertion and remeshing, the remeshed node IDs and associated element facets are mapped to the named surface to define the interpolated boundary conditions.

Depending on the topological/geometrical features that define the boundary of the surface, the mapped remeshed surface might be slightly different in terms of shape and size. For example, Fig 5.9 shows the mesh facets and nodes for a contact-surface, which is part of a larger flat surface of the model; Fig 5.9 shows the surface as being selected.

Without selecting the surface, and after inserting the crack and remeshing (Fig 5.10 – left image), the mapped surface is different, as shown in the right image of Fig 5.10.

If we retained the outer ring of element facets of the initial surface, the mapping is more constrained. For example, Fig 5.11 (left image) shows the model in FRANC3D where the outer mesh facets that surround the contact-surface are retained, which leads to a better reproduction of the contact surface (right panel of Fig 5.11).


Select Retained BC Surfaces

- ☐ CONTACT_COUP
- ☐ _DISPZEROUY
- ☐ _FRICSUX
- ☐ _FRICSUZ
- ☐ disp_5
- ☐ disp_6
- ☐ disp_7
- ☑ surf154_nodes

Display
- ☐ Markers
- ☑ Vectors
- ☑ Polygons
- ☑ Text
- ☐ Mesh
- ☐ Fronts

(reset) (d)

Shift+Click to select surfaces

Figure 5.9 A portion of a larger flat surface.

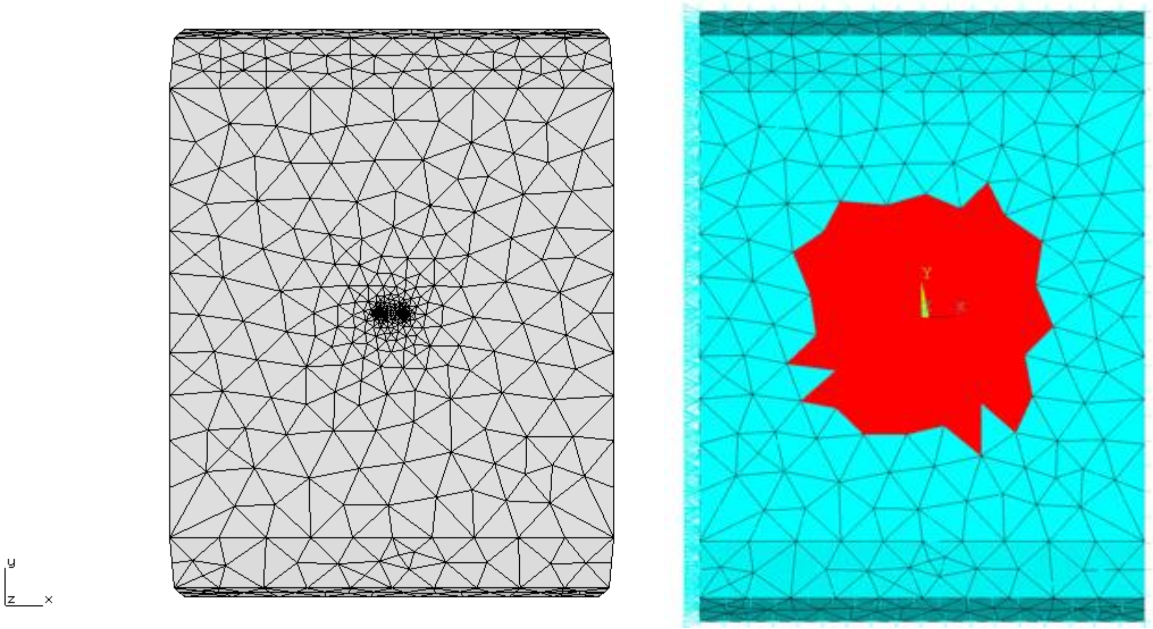


Figure 5.10 Mapped remeshed surface portion.

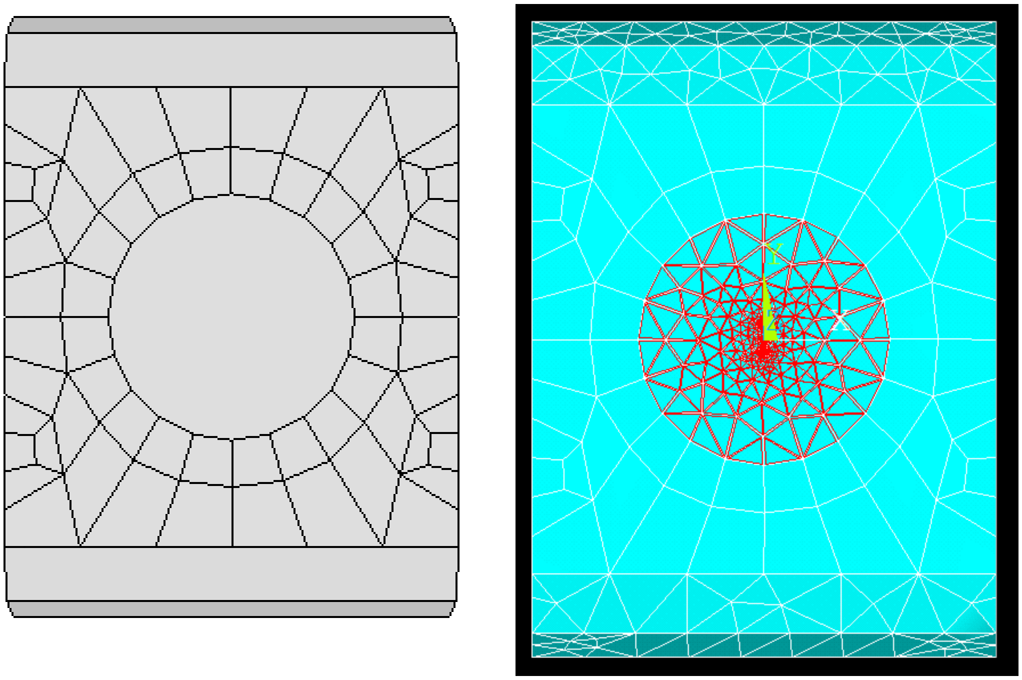


Figure 5.11 Mapped remeshed surface portion constrained by the retained surface mesh.

5.6 Edit Retained Nodes

This section describes how to edit the retained nodes. The model, Fig 5.12, is a 10x5x5 plate, and the boundary conditions consist of uniform traction and displacement constraints. The elastic modulus is 3.0e7 and Poisson's ratio is 0.30. The top and bottom surfaces of the plate have uniform unit traction. There are constraints on the bottom surface to prevent rigid body motion and the +z and -z surfaces are constrained to produce plane strain Fig 5.13.

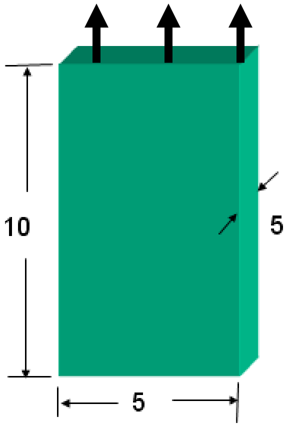


Figure 5.12 Plate model.

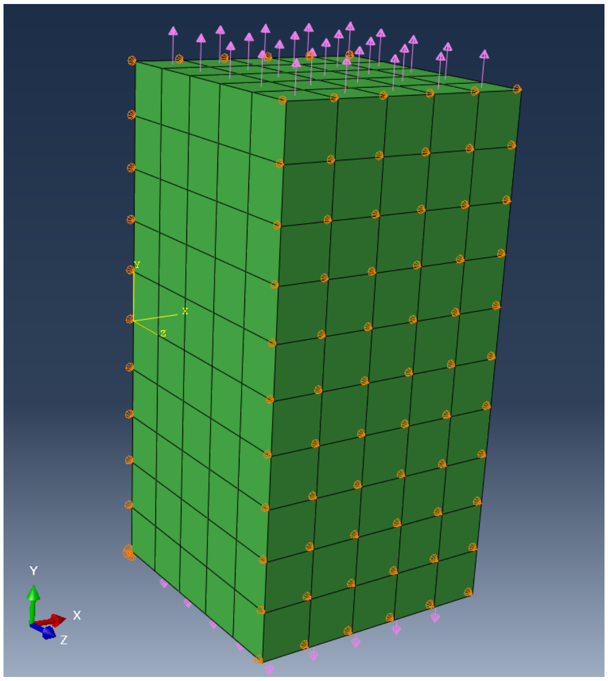


Figure 5.13 Traction and constraint applied.

Using the FRANC3D submodel dialog, we cut out the selection of elements shown in Fig 5.14 and save the Local and Global model files. FRANC3D imports the Local portion and displays the surfaces/sets that can be retained, Fig 5.15. The +z and -z faces are selected to be retained, Fig 5.15, and the cut-surface faces are automatically retained.

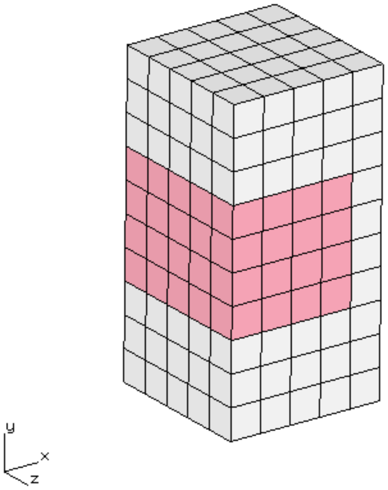
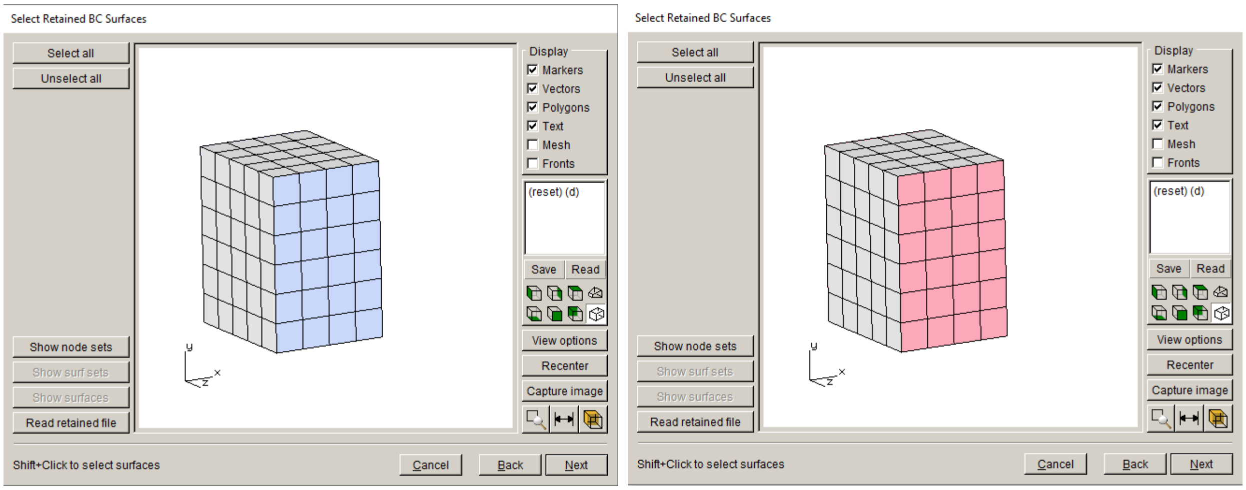


Figure 5.14 Local selection.


Select Retained BC Surfaces

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Fronts

(reset) (d)

Shift+Click to select surfaces

Select Retained BC Surfaces

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Fronts

(reset) (d)

Shift+Click to select surfaces

Figure 5.15 Surfaces with constraints; surfaces retained.

The result is the model shown in Fig 5.16. A half-penny surface crack is inserted into the -x face, Fig 5.17. This crack is then propagated automatically for a number of steps. At the step where the crack reaches the retained z-faces, the growth stops with an error message, Fig 5.18.

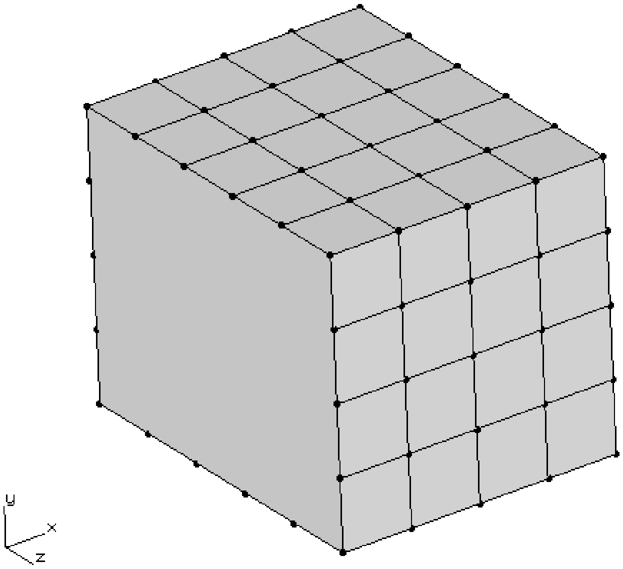


Figure 5.16 FRANC3D local model.

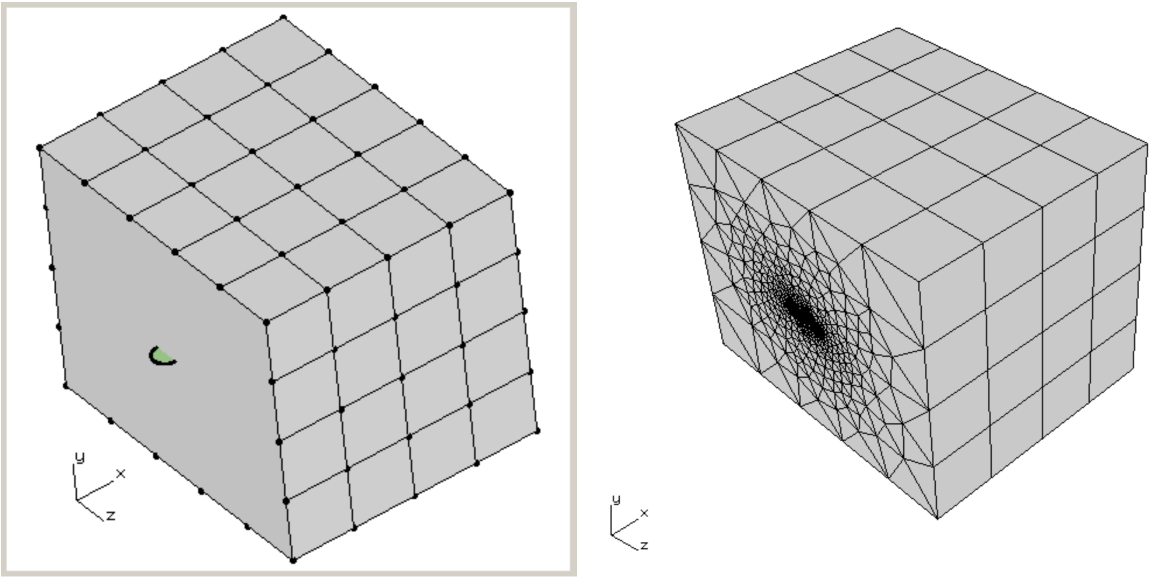
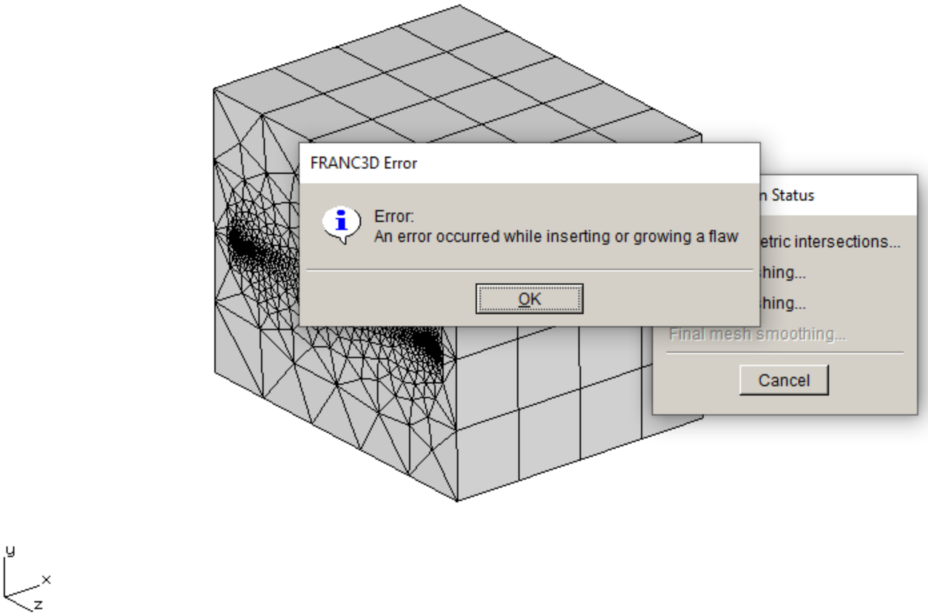


Figure 5.17 Surface half-penny crack inserted into local model.


FRANC3D Error

Error:
An error occurred while inserting or growing a flaw

Figure 5.18 Error when trying to grow the crack into the retained faces.

At this point, we use the **Edit Retained Nodes** option in the **Advanced** menu, Fig 5.19, to remove the nodes on the z-faces from the retained nodes list. Fig 5.20 shows the original set of retained nodes (left panel); click the **Unselect all** button to clear the selection (right panel).

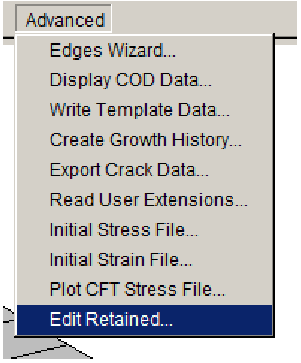


Figure 5.19 Advanced menu option to edit the retained faces.

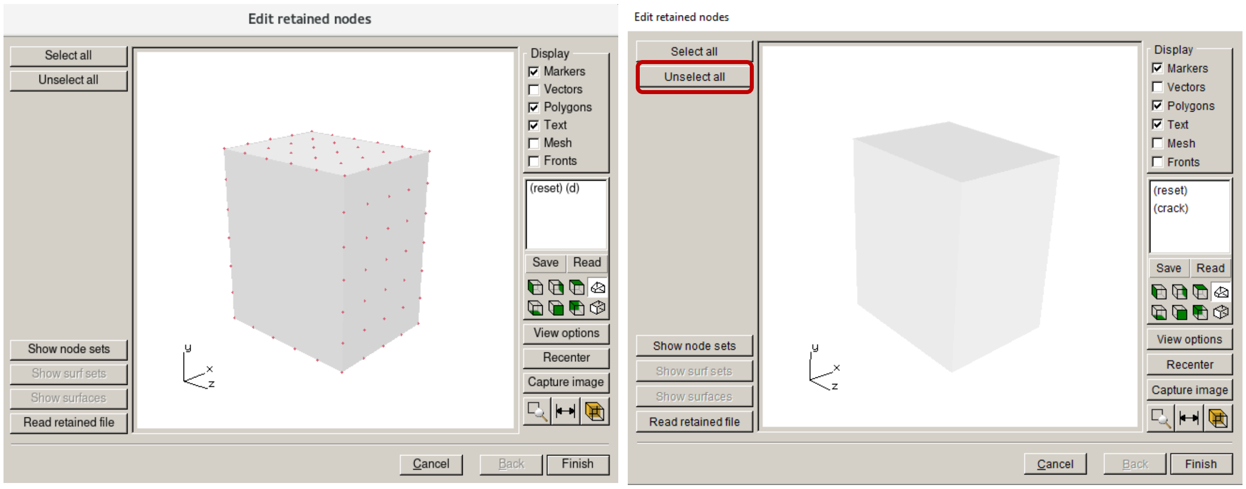
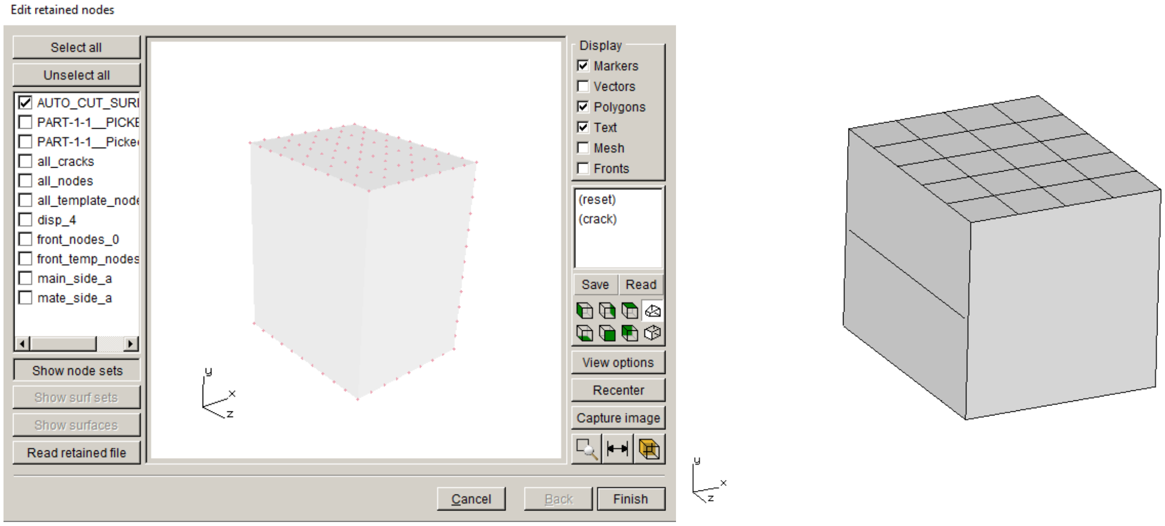


Figure 5.20 Original retained nodes cleared by the Unselect all button.

Next click on the **Show Node Sets** button; the AUTO_CUT_SURF is automatically checked, Fig 5.21. Do not select any other sets or surfaces, and click **Finish**. The model in the FRANC3D main window will be updated (see right panel of Fig 5.21).

Crack growth can now be continued Fig 5.22, and the SIF history for all crack fronts can be obtained, Fig 5.23. FRANC3D transfers the boundary conditons directly from the original uncracked mesh if the nodes are retained, Fig 5.24 – left panel. If the nodes are not retained, then the boundary conditions are mapped to the remeshed surface, Fig 5.24 – right panel.



☑ AUTO_CUT_SURI
□ PART-1-1__PICKE
□ PART-1-1__Picke
□ all_cracks
□ all_nodes
□ all_template_node
□ disp_4
□ front_nodes_0
□ front_temp_nodes
□ main_side_a
□ mate_side_a

☑ Markers
□ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

Figure 5.21 Reselect only the AUTO_CUT_SURF nodes.

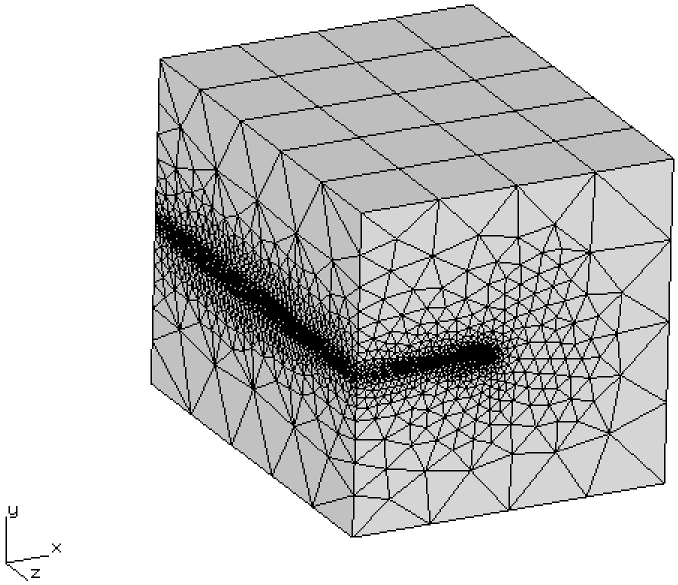
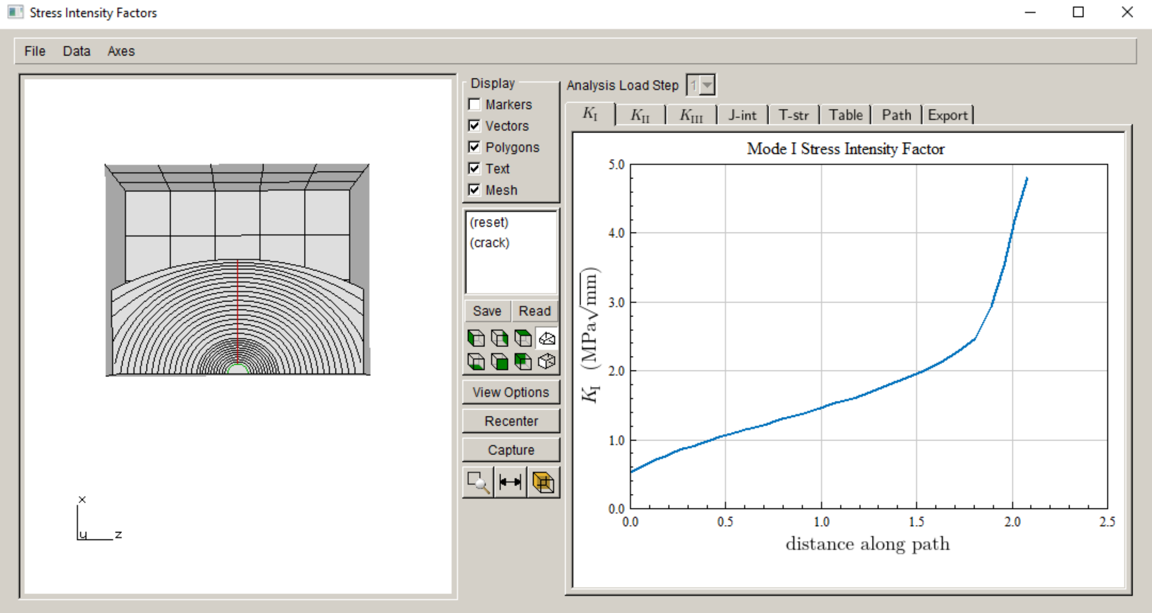


Figure 5.22 Further crack growth allowed after editing the retained nodes.



Stress Intensity Factors

File Data Axes

Display

☐ Markers

☑ Vectors

☑ Polygons

☑ Text

☑ Mesh

(reset)
(crack)

Analysis Load Step 1

Figure 5.23 Complete SIF history.

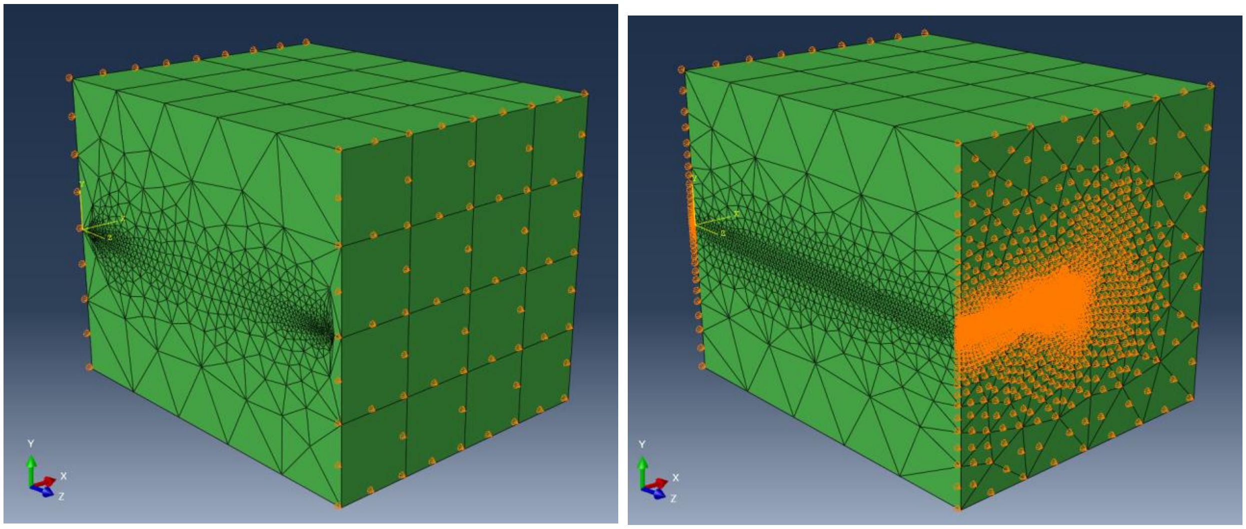


Figure 5.24 Constraints on retained mesh prior to editing, and constraints on the same surface after editing and further crack growth.

6. Defining Initial Crack Geometry

FRANC3D provides a set of parameterized crack shapes plus two options for user-defined cracks, Fig 6.1. The two user-defined crack types are discussed below.

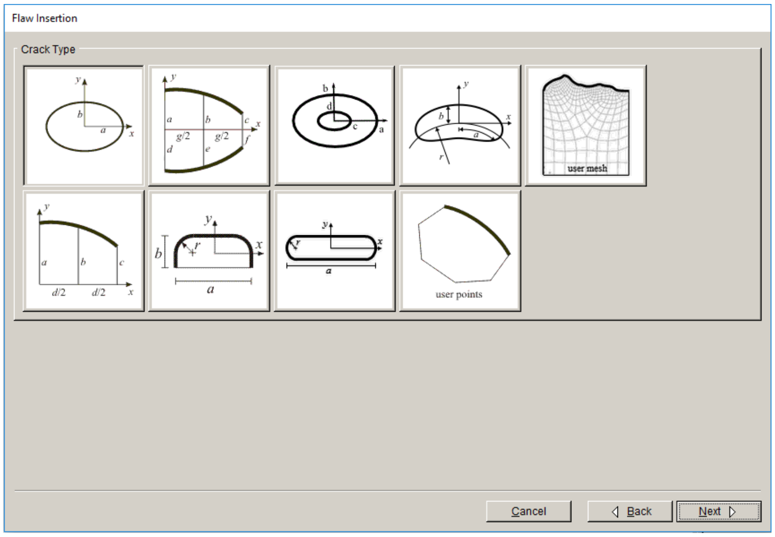


Figure 6.1 Crack shape library.

6.1 User-Points Crack

If you have an essentially planar crack surface with an arbitrary front shape, you can use the user-points crack. Fig 6.2 shows such a user-defined crack shape.

Section 6.1.10 of the Reference document describes the boundary point data that must be entered to define the crack. Once the boundary points are entered, FRANC3D triangulates the region inside the boundary to create the Bezier surface geometry. To successfully triangulate the region, there are a couple of things to note:

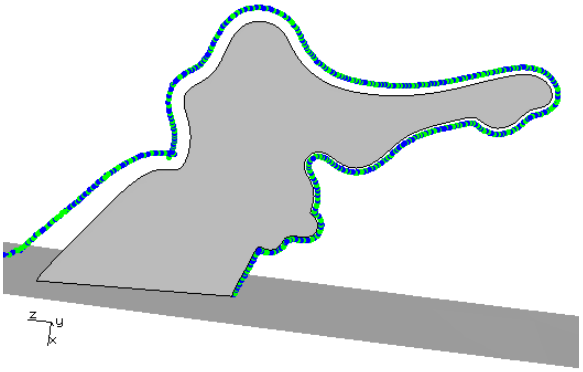
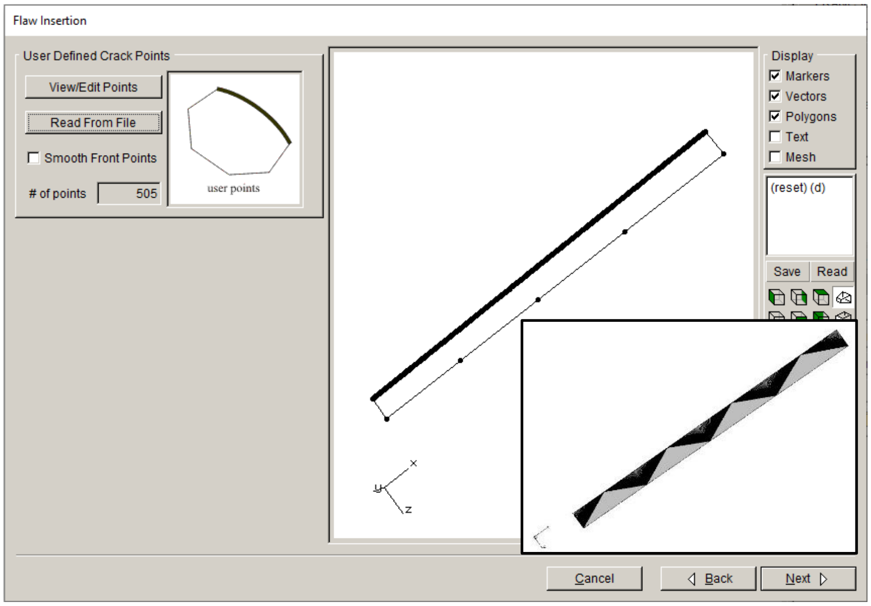


Figure 6.2 A user-defined crack shape.

1) There should be a reasonable number of points on the boundary to create a reasonable number of Bezier patches.
2) There should be enough non-front points on the boundary to create a reasonable transition from the front points so that triangular Bezier patches can be defined.

Fig 6.3 shows a case where the number of points on the front boundary is excessive; the resulting triangulated region is shown in the bottom-left inset-image. FRANC3D provides options for smoothing and reducing the number of front points. If we reduce the front points to 10, then the number of triangles and the shape of the triangles is reasonable, Fig 6.4. The amount of time required to insert the crack is reduced with fewer triangles because the number of intersections will be reduced.

One should use enough points to capture the shape of the front. The number of points along the front might determine the number of non-front points that are required. Fig 6.5 shows a case where there are insufficient non-front boundary points. The user should add additional non-front points to create better triangles, Fig 6.6.


Flaw Insertion

User Defined Crack Points

□ Smooth Front Points

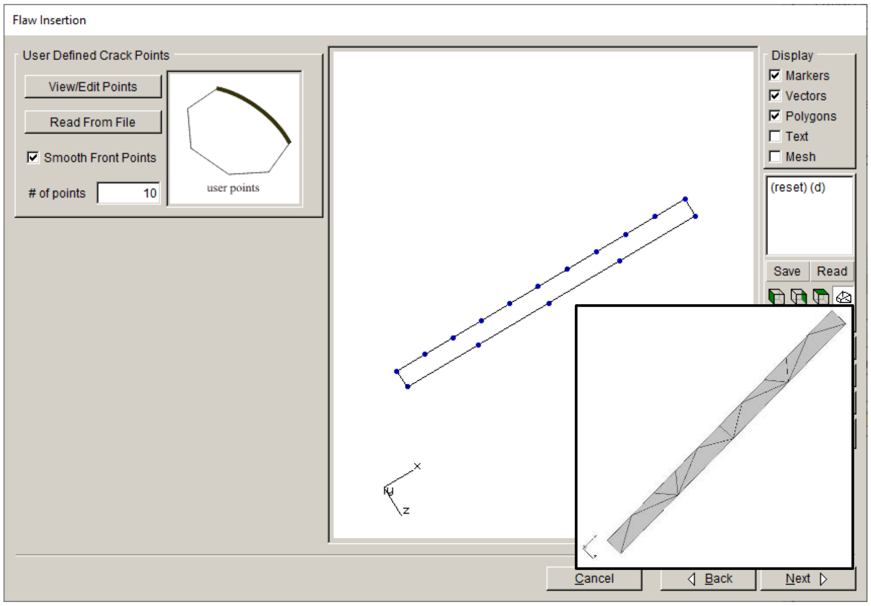
# of points 505

Display

☑ Markers
☑ Vectors
☑ Polygons
□ Text
□ Mesh

(reset) (d)

Figure 6.3 Excessive number of front points.


Flaw Insertion

User Defined Crack Points

☑ Smooth Front Points

# of points 10

Display

☑ Markers

☑ Vectors

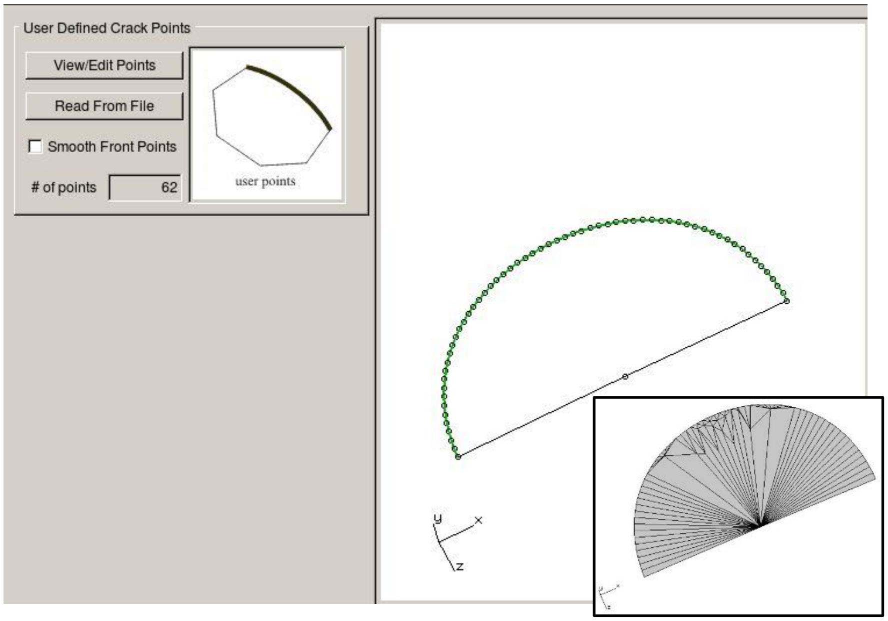
☑ Polygons

☐ Text

☐ Mesh

(reset) (d)

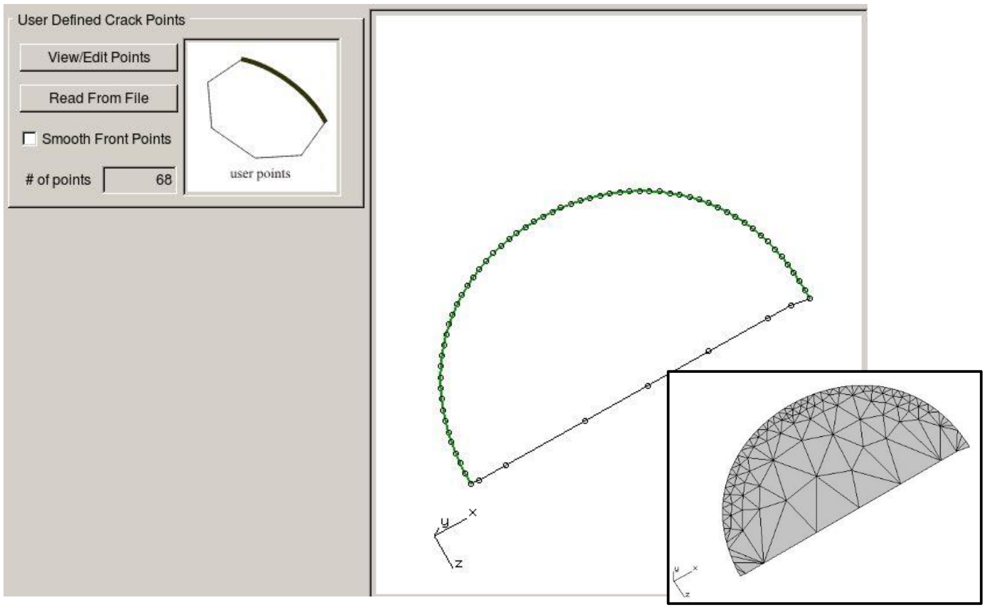
Figure 6.4 Reasonable number of front points.


User Defined Crack Points

☐ Smooth Front Points

# of points 62

Figure 6.5 Single non-front point.



☐ Smooth Front Points

# of points 68

Figure 6.6 Additional non-front points produce a better triangulated surface.

For a surface-breaking crack, the non-front boundary points should fall outside the model surface to ensure that intersections between the crack and model can be computed, and the first and last front boundary points should fall outside the model surface to ensure that the crack front template intersects the model surface too.

6.2 User-Mesh Crack

If the initial crack surface is non-planar, the user-mesh crack is the only option. FRANC3D can read a surface mesh and convert it to a crack. The surface mesh can be from ANSYS, ABAQUS or NASTRAN. A generic *.stl* file can also be used, but it requires an extra text file for the crack front points.

Section 6.1.11 of the Reference document describes the FRANC3D interface to add this crack type to a model. The surface mesh file should include a collection of surface elements and nodes, and a node set (or component) that defines the crack front. The front node IDs should be given in geometric order along the front (although FRANC3D should resolve the ordering). For the *.stl* file, an extra *.txt* file is required that gives the Cartesian coordinates of the front points; the coordinates should match those in the *.stl* file.

An example ABAQUS surface mesh is shown in Fig 6.7. The element surface normal vectors should all be oriented consistently. One can use a standard 2D-planar or shell-surface element type; shell elements are best for non-planar surfaces.

The surface geometry and mesh must be contiguous and water-tight; there should not be doubled nodes along any boundaries.

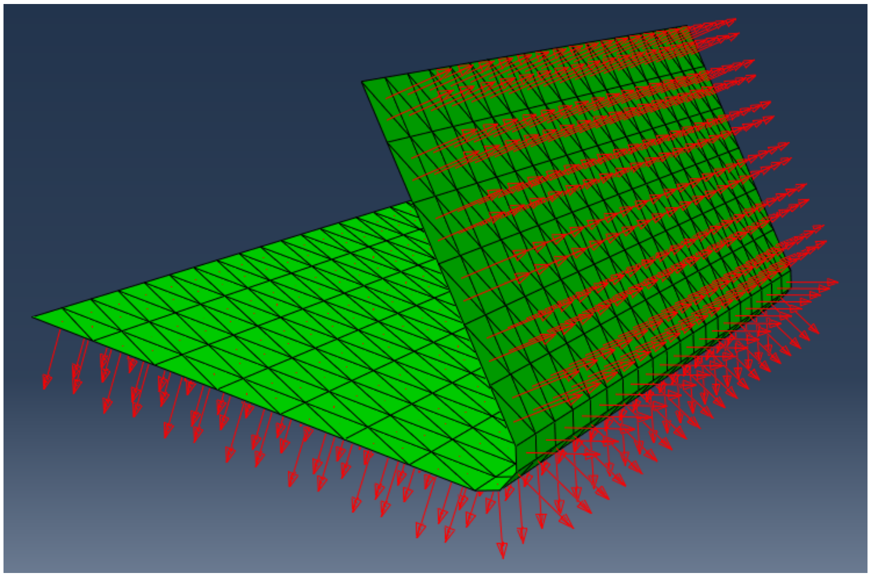


Figure 6.7 ABAQUS surface mesh with element-normals shown.

The mesh is read into FRANC3D, and the user selects the crack front node sets, which should be defined in the mesh file (.*inp* for ABAQUS). FRANC3D displays this as a crack surface with the front edges highlighted in green, Fig 6.8.

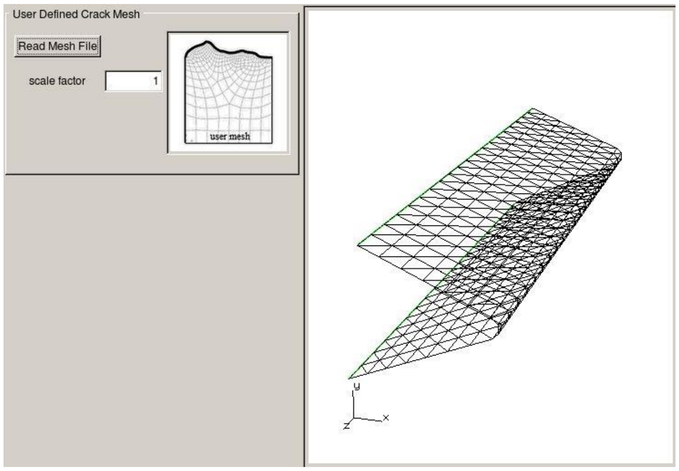
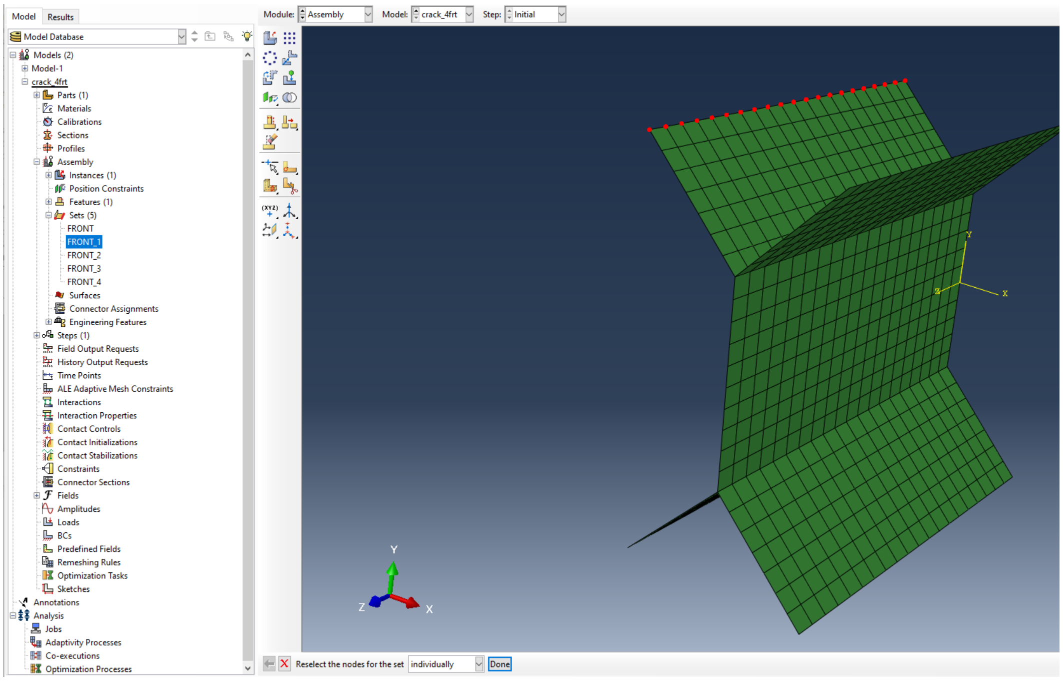


Figure 6.8 ABAQUS surface mesh as a FRANC3D crack surface – front edges colored green.

The surface mesh should be defined such that the crack geometry clearly intersects the model surface geometry – for a surface-breaking crack. This might require that the user add extra elements and nodes (*i.e.*, crack geometry) to the surface mesh.

6.2.1 Non-Manifold Crack

For a non-manifold crack, Fig 6.9, one must designate the front nodes explicitly for each crack front. Fig 6.9 shows the FRONT_1 node set; there are four crack fronts with four corresponding node sets in the model.

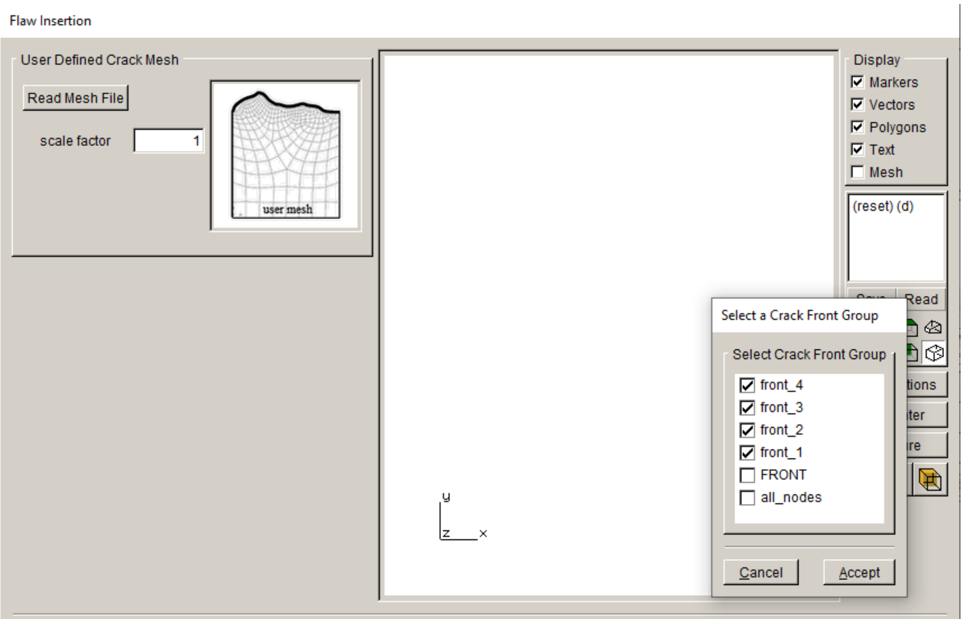


Module: Assembly Model: crack_4frt Step: Initial

Reselect the nodes for the set individually

Figure 6.9 Non-manifold ABAQUS surface mesh for a FRANC3D crack surface.

When importing the mesh surface into FRANC3D as a crack surface, the front node sets will be listed, Fig 6.10, and the user should select the node sets that define the crack fronts. One should see the front edges highlighted and templates along each of the crack fronts, Fig 6.11.



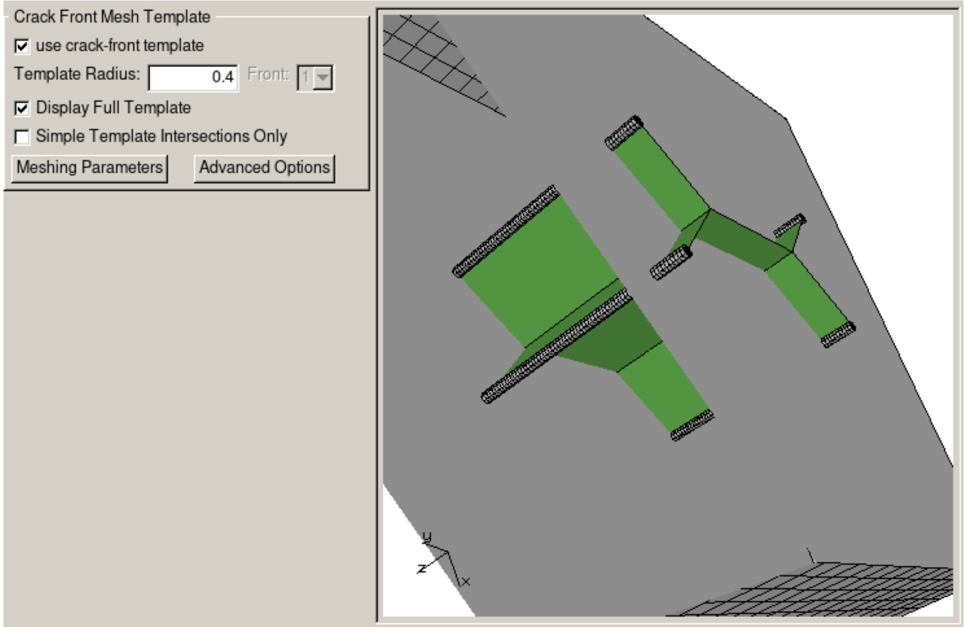
scale factor 1

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Select Crack Front Group

☑ front_4
☑ front_3
☑ front_2
☑ front_1
☐ FRONT
☐ all_nodes

Figure 6.10 Crack front node sets listed.



☑ use crack-front template

Template Radius: 0.4 Front: 1

☑ Display Full Template

☐ Simple Template Intersections Only

Figure 6.11 Crack front templates along the four fronts.

6.3 Multiple Cracks

When inserting multiple cracks with significant differences in crack size, Fig 6.12, you should make sure the geometry patches (triangles) are similar in size for all cracks. You will need to use the **Advanced Geometry** option when defining the cracks; see Section 6.1.2 of the Reference document.

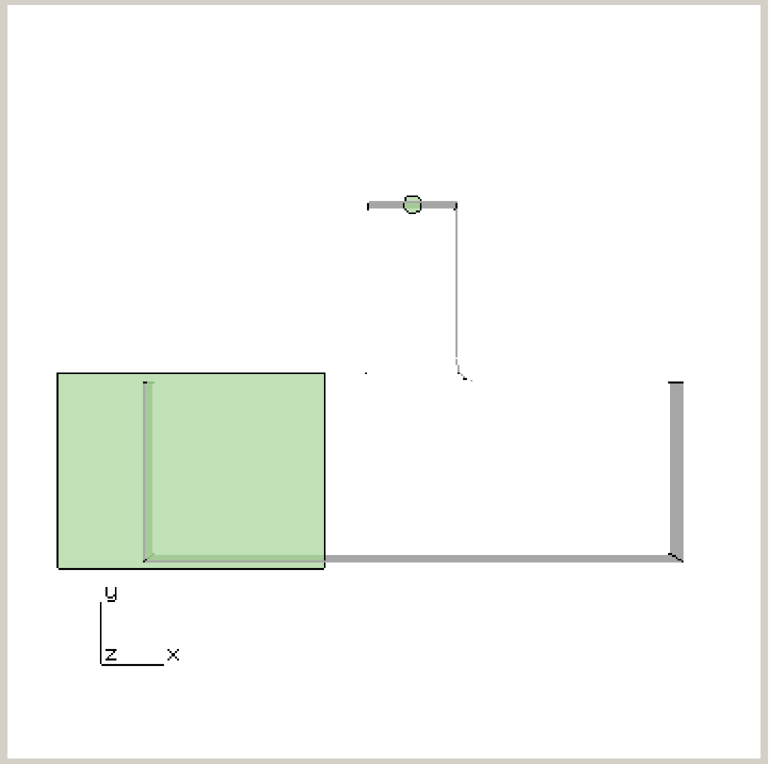
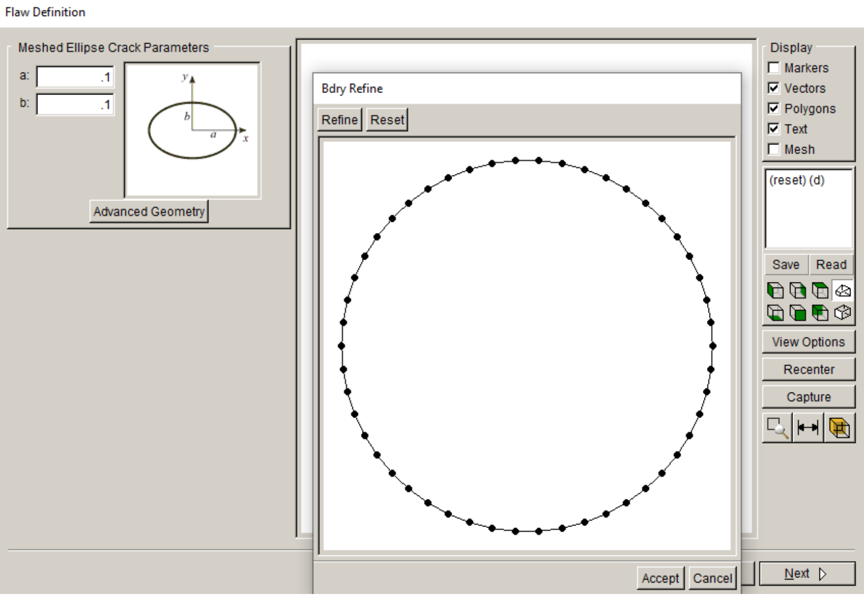


Figure 6.12 Multiple cracks of differing size inserted into a T-shape model.

Fig 6.13 shows the geometric boundary for a penny-shaped crack with radius of 0.1 units; the triangulation algorithm fills the interior space with comparable size triangles. We will not use the Refine option here as this is the smaller crack; the **Refine** button increases the number of points on the boundary, which will increase the number of triangles created inside.

Fig 6.14 shows the original geometry triangles for an edge-crack with a length of 3 units. The triangles are an order of magnitude bigger than the triangles for the penny-shaped crack in Fig 6.13. The **Refine** button in the Advanced Geometry dialog should be used to divide the triangles into smaller triangles, Fig 6.15.

FRANC3D should automatically refine the crack geometry to ensure comparable size triangles, but the user can check to make sure the refinement is sufficient.


Flaw Definition

Meshed Ellipse Crack Parameters

a: .1

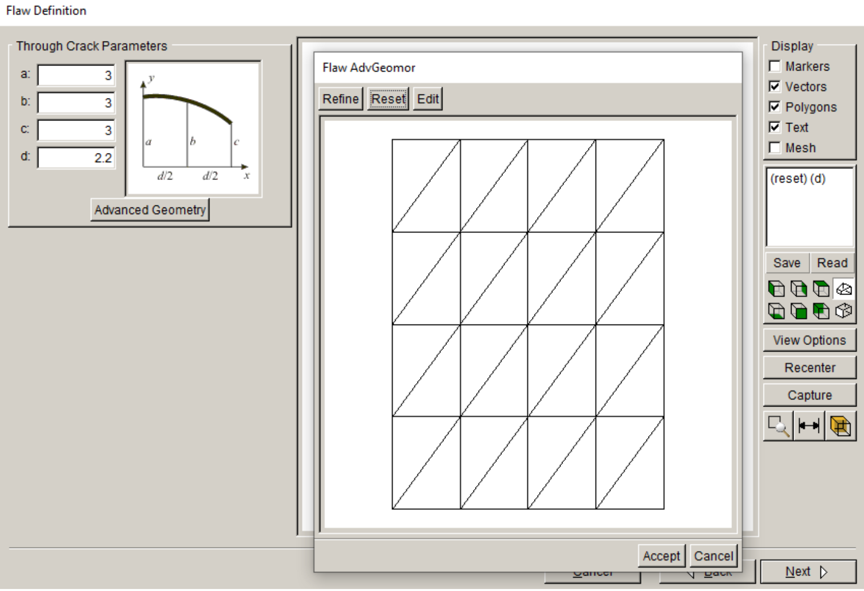
b: .1

Display

☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)

Figure 6.13 Boundary geometry for penny-shaped crack with radius of 0.1 units.


Flaw Definition

Through Crack Parameters

a: 3

b: 3

c: 3

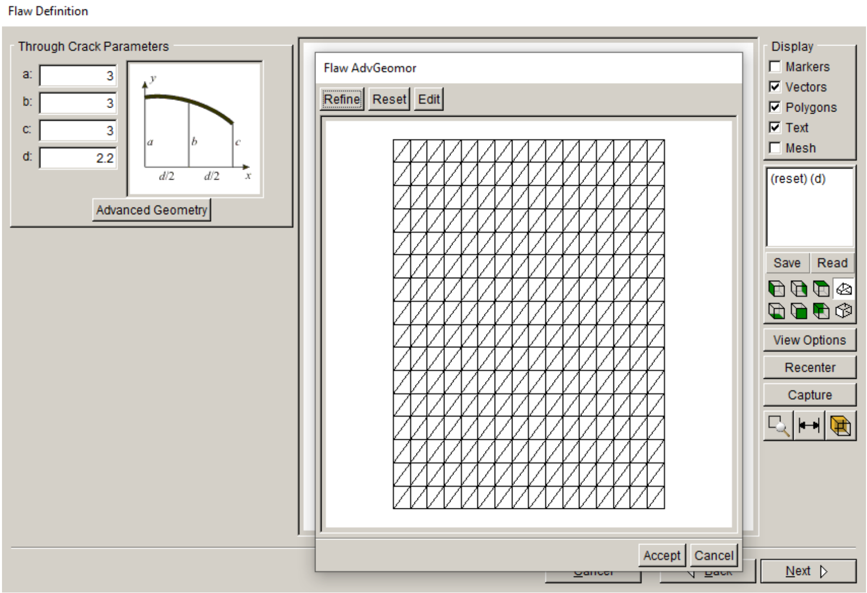
d: 2.2

Display

☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)

Figure 6.14 Geometry for an edge (through) crack with length of 3 units.


Flaw Definition

Through Crack Parameters

a: 3

b: 3

c: 3

d: 2.2

Display

☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)

Figure 6.15 Refined geometry for an edge (through) crack with length of 3 units.

6.4 Potential Problems for Crack Insertion

Potential problems for crack insertion are discussed in this section.

6.4.1 Crack at a Kink in the Model Surface Geometry

Cracks can be inserted at kinks in the model surface geometry, Fig 6.16, but one might need to turn on “simple intersections” for the crack front template. Fig 6.17 shows the template radius reduced by half and the Simple Template Intersections Only option checked. It usually helps to reduce the template radius in this case so that you get more elements along the crack front and thus more nodes closer to the model surface. Fig 6.18 shows the resulting surface mesh on the crack, where the template is pulled back from the model surface.

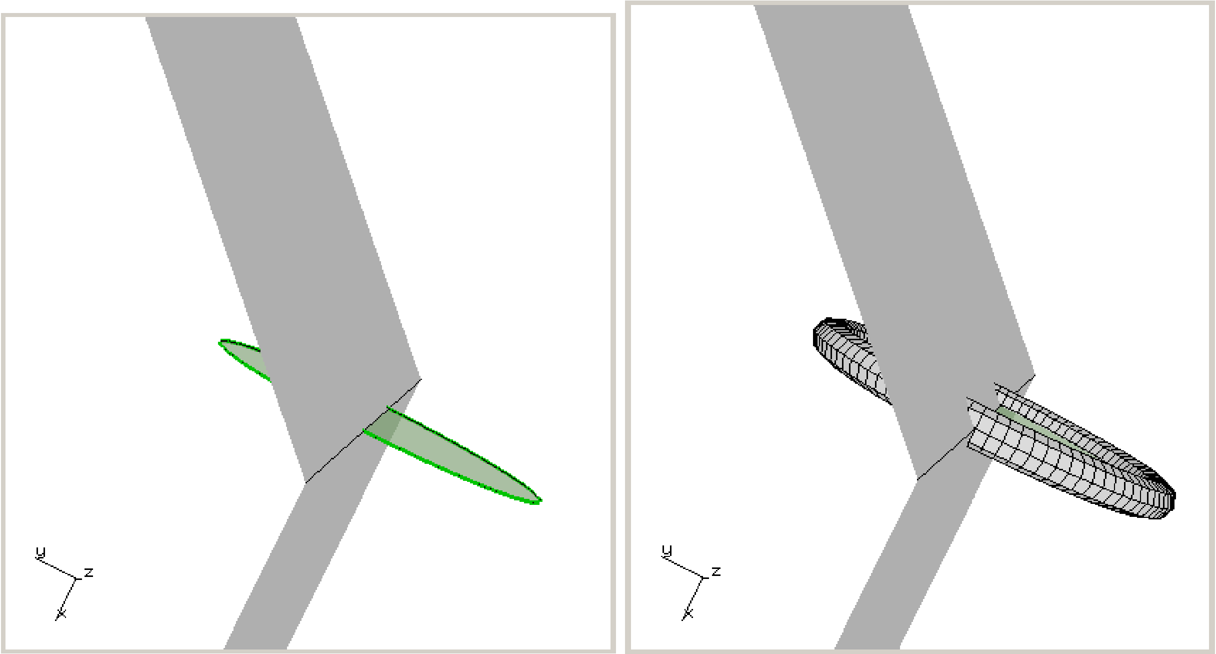
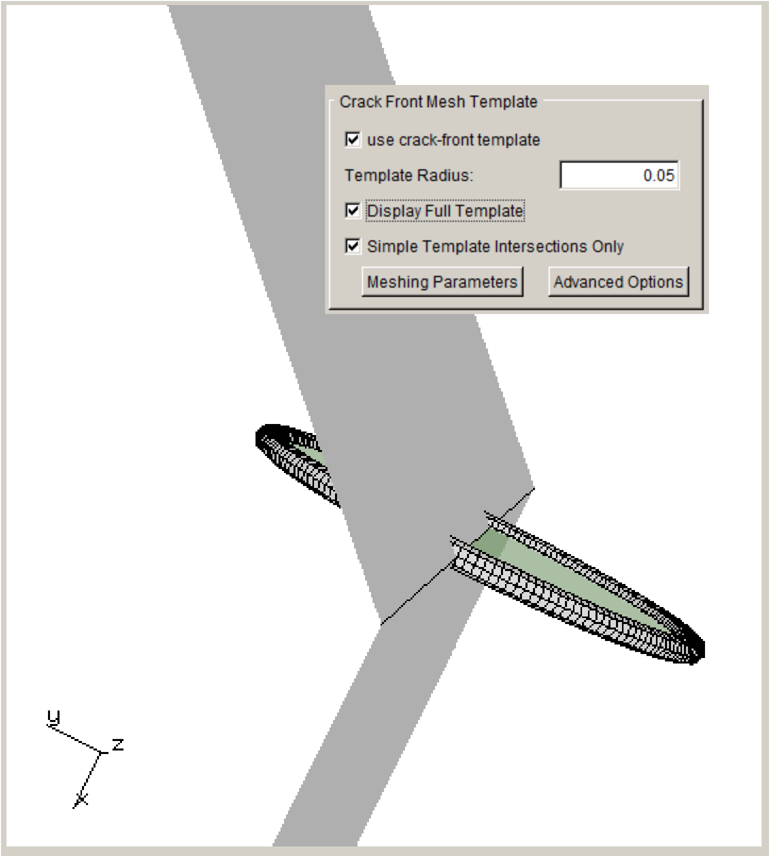


Figure 6.16 Crack inserted at a kink in the model surface – crack (left) and template (right).


Crack Front Mesh Template

☑ use crack-front template

Template Radius: 0.05

☑ Display Full Template

☑ Simple Template Intersections Only

Figure 6.17 Template simple intersections turned on and template radius reduced.

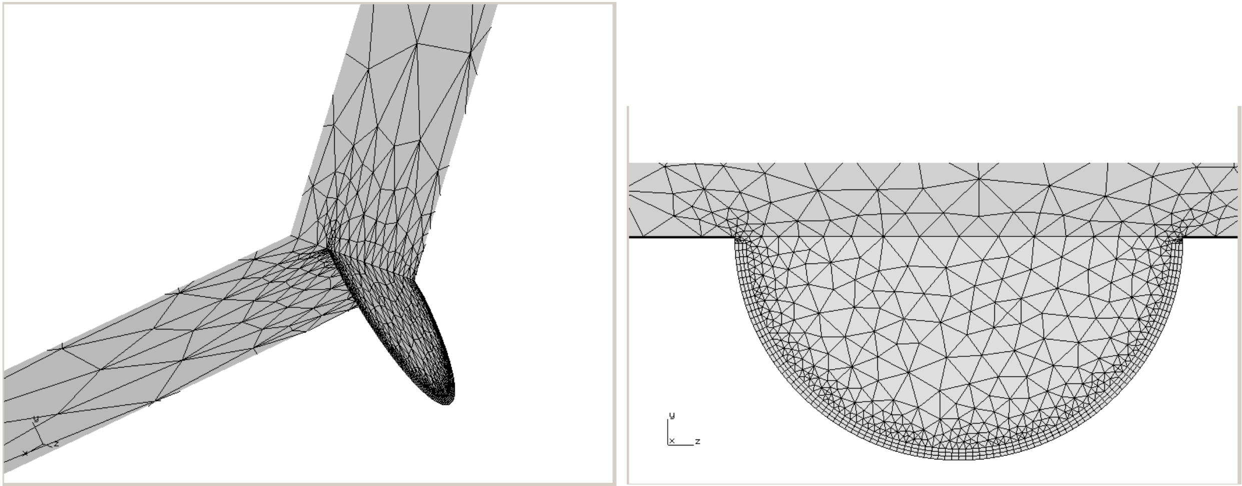


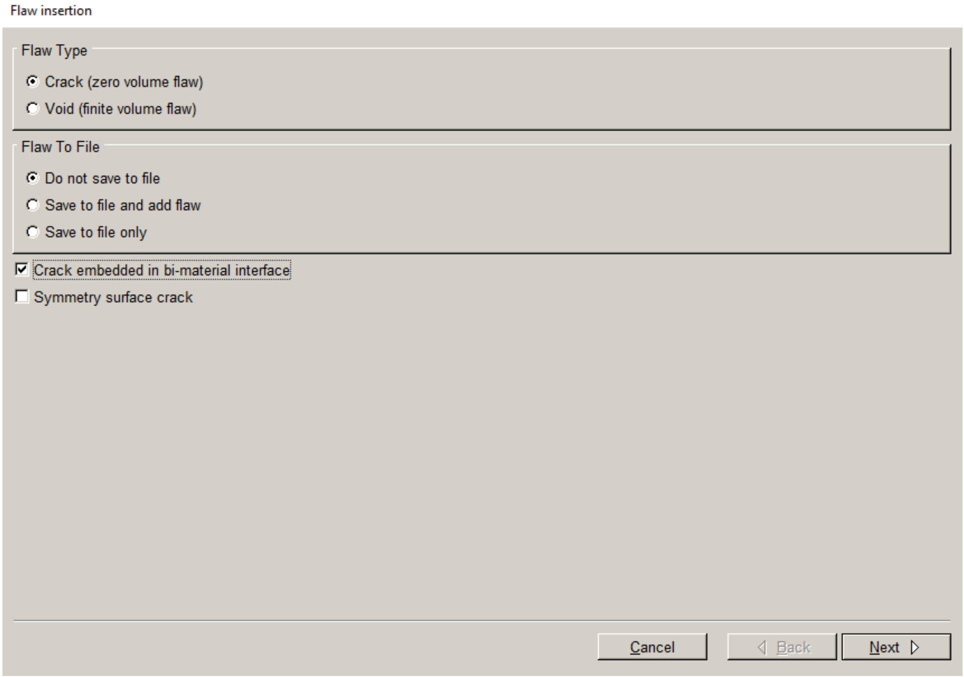
Figure 6.18 Crack inserted and meshed with template simple intersections turned on.

The crack insertion library should limit the need for simple intersections and should turn on simple intersections automatically when needed. The option to manually turn this feature on is still there, however.

6.4.2 Cracks Embedded in Interfaces

FE models might contain interior surfaces; this can result from material boundaries or can be the product of separate parts that are meshed independently and then constrained. There is an optional setting for embedding a crack into a material interface, Fig 6.19. Fig 6.20 shows a penny-shape crack inserted into a material interface between two halves of a cube. The resulting crack surface mesh is shown in Fig 6.21 and the deformed shape after the FE analysis is shown in Fig 6.22.

This only works for interfaces with a single surface; for a doubled interface see Chapter 15.



Flaw insertion

Flaw Type

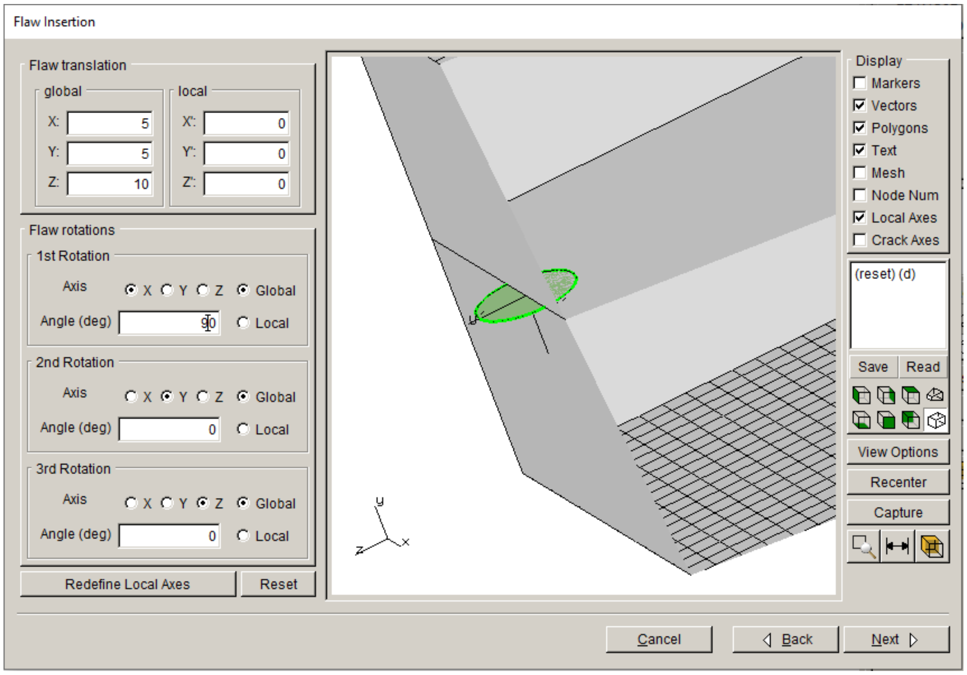
● Crack (zero volume flaw)
○ Void (finite volume flaw)

Flaw To File

● Do not save to file
○ Save to file and add flaw
○ Save to file only

☑ Crack embedded in bi-material interface
□ Symmetry surface crack

Figure 6.19 Flaw insertion wizard with bi-material interface option checked.



Flaw Insertion

Flaw translation

global		local	
X:	5	X':	0
Y:	5	Y':	0
Z:	10	Z':	0

Flaw rotations

1st Rotation
Axis ● X ○ Y ○ Z ● Global
Angle (deg) 90 ○ Local

2nd Rotation
Axis ○ X ● Y ○ Z ● Global
Angle (deg) 0 ○ Local

3rd Rotation
Axis ○ X ○ Y ● Z ● Global
Angle (deg) 0 ○ Local

Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Node Num
☑ Local Axes
□ Crack Axes

(reset) (d)

Figure 6.20 Crack inserted in a material interface.

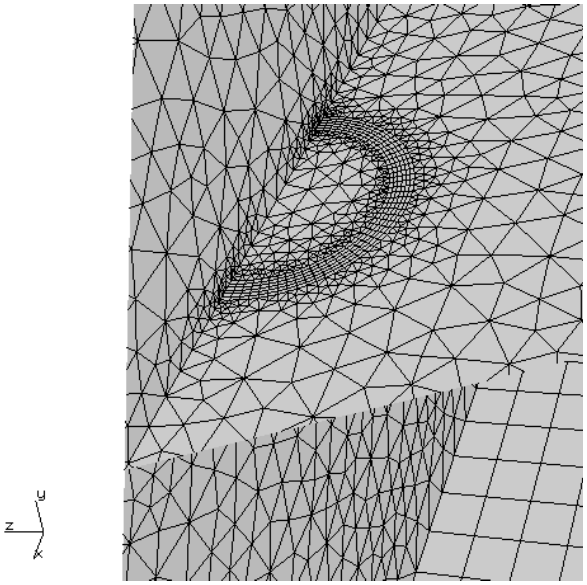
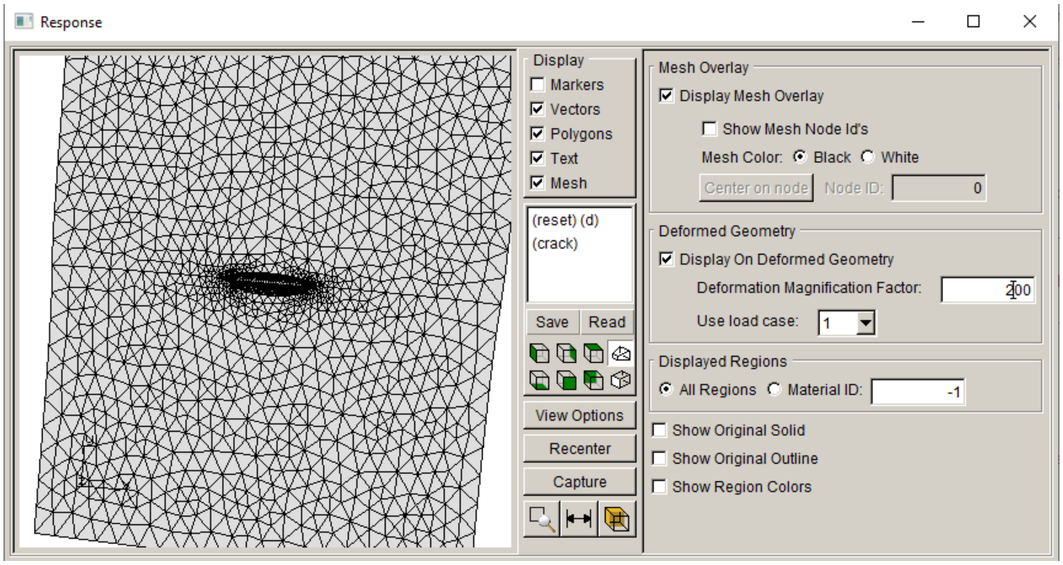


Figure 6.21 Surface mesh for a crack embedded in a material interface.



Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

(reset) (d)
(crack)

Mesh Overlay
☑ Display Mesh Overlay
☐ Show Mesh Node Id's
Mesh Color: ● Black ○ White
Node ID: 0

Deformed Geometry
☑ Display On Deformed Geometry
Deformation Magnification Factor: 200
Use load case: 1

Displayed Regions
● All Regions ○ Material ID: -1

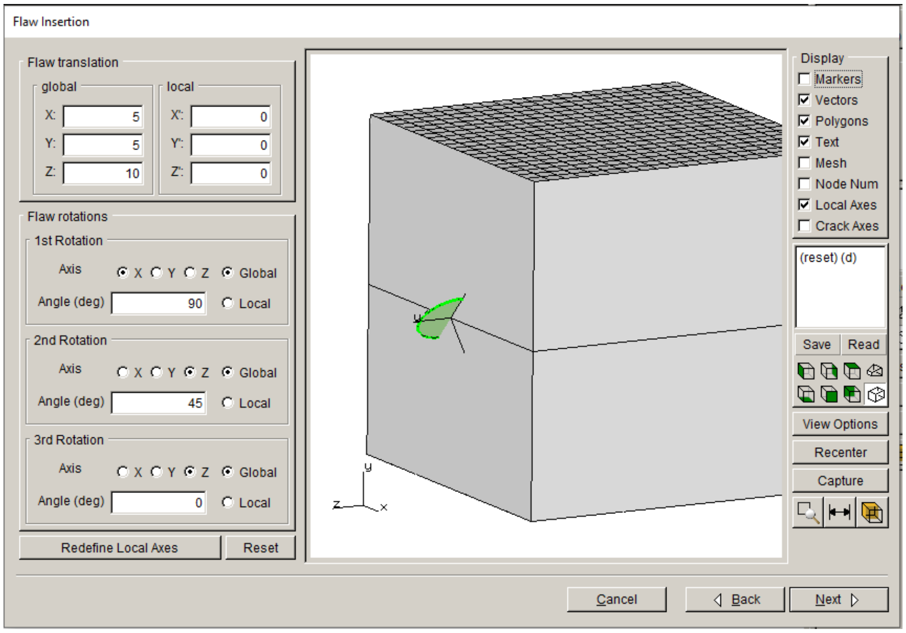
☐ Show Original Solid
☐ Show Original Outline
☐ Show Region Colors

Figure 6.22 Deformed shape of crack embedded in a material interface.

6.4.3 Cracks Crossing Interfaces

Using the same bi-material cube model from the previous section, a penny-shape crack is defined, Fig 6.23, such that it crosses the material interface (at a 45-degree angle). The resulting crack surface mesh is shown in Fig 6.24 and the deformed shape after the FE analysis is shown in Fig 6.25. The deformed shape shows a difference in crack opening for the two materials.

This only works for interfaces with a single surface; for a doubled interface see Chapter 15.



Flaw Insertion

Flaw translation

global		local	
X:	5	X':	0
Y:	5	Y':	0
Z:	10	Z':	0

Flaw rotations

1st Rotation
Axis ● X ○ Y ○ Z ● Global
Angle (deg) 90 ○ Local

2nd Rotation
Axis ○ X ○ Y ● Z ● Global
Angle (deg) 45 ○ Local

3rd Rotation
Axis ○ X ○ Y ● Z ● Global
Angle (deg) 0 ○ Local

Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Node Num
☑ Local Axes
□ Crack Axes

(reset) (d)

Figure 6.23 Crack crossing a material interface.

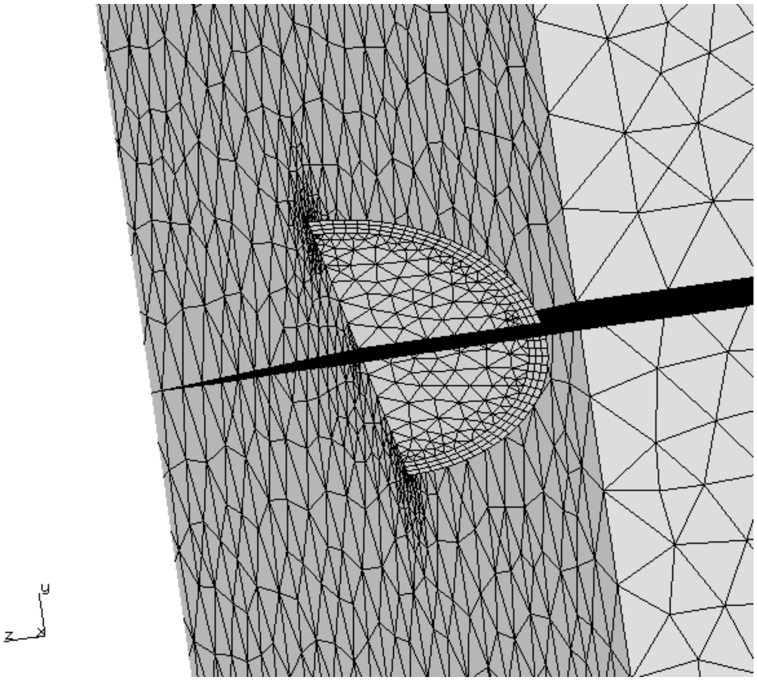
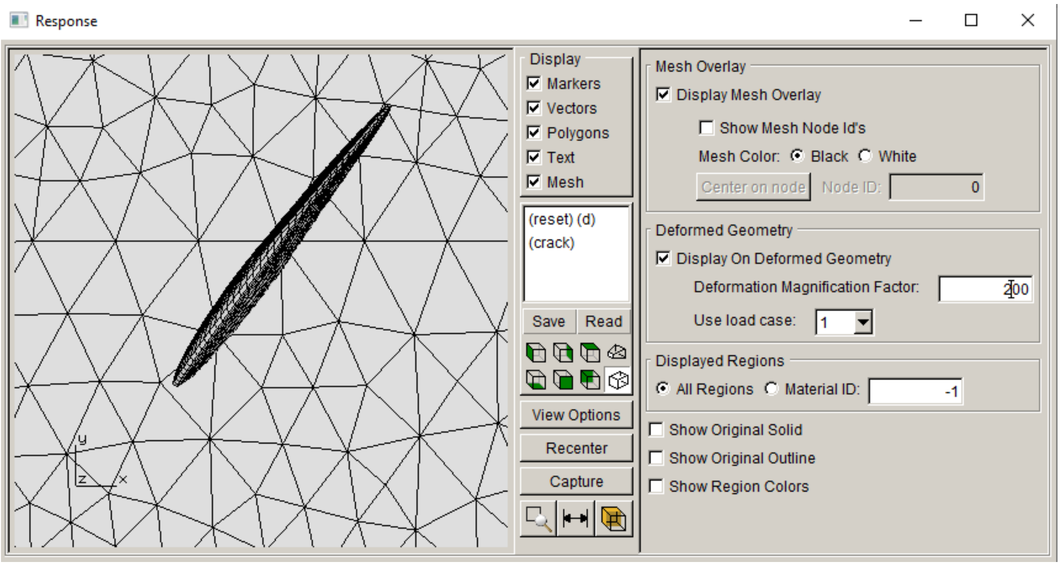


Figure 6.24 Surface mesh for a crack crossing a material interface.


Response

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

(reset) (d)
(crack)

Mesh Overlay
☑ Display Mesh Overlay
□ Show Mesh Node Id's
Mesh Color: ● Black ○ White
Node ID: 0

Deformed Geometry
☑ Display On Deformed Geometry
Deformation Magnification Factor: 200
Use load case: 1

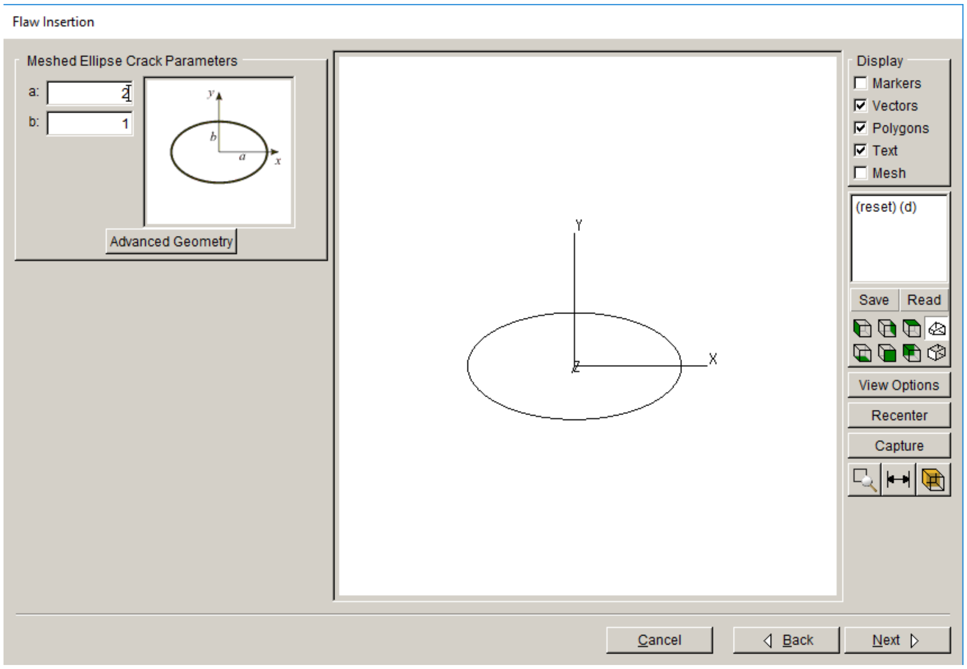
Displayed Regions
● All Regions ○ Material ID: -1

□ Show Original Solid
□ Show Original Outline
□ Show Region Colors

Figure 6.25 Deformed shape for a crack crossing a material interface.

6.5 Local Coordinate System for Positioning a Crack

FRANC3D uses three coordinate systems when orienting a flaw. There is the global Cartesian system, a local user-definable coordinate system, and an intrinsic crack system that is attached to the flaw. The intrinsic crack system is shown when the parameters for the flaw are defined. This consists of the x, y, and z axes shown in Fig 6.26.


Flaw Insertion

Meshed Ellipse Crack Parameters
a: 2
b: 1

Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset) (d)

Figure 6.26 Flaw insertion dialog.

6.5.1 Crack Translations and Rotations

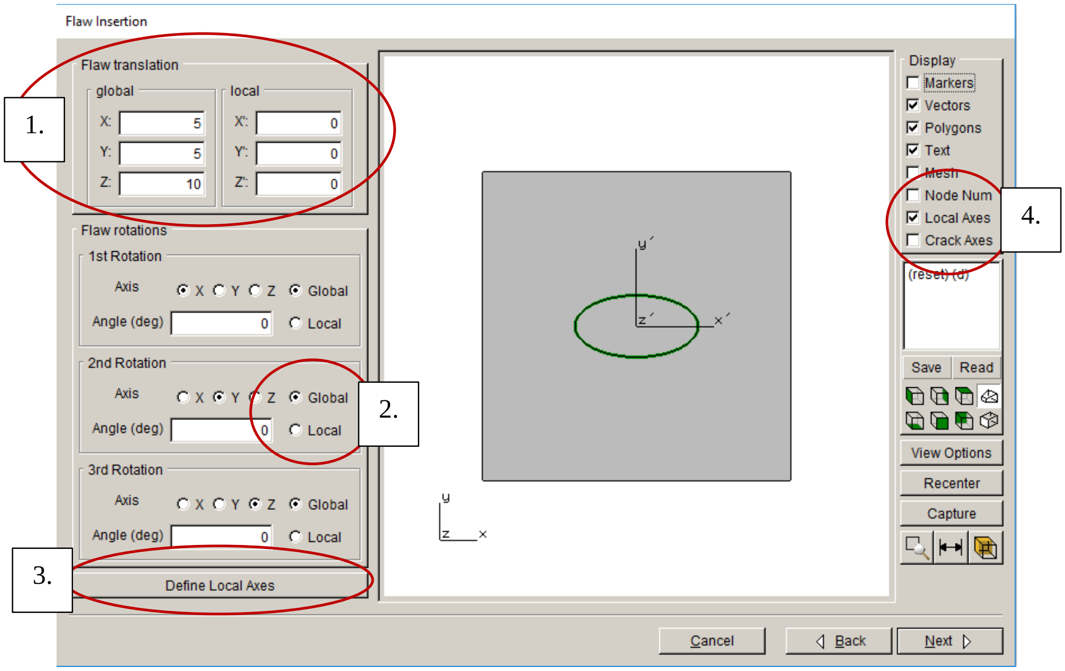
Fig 6.27 shows the crack oriented with model. The crack can be translated and rotated into the correct position by manually editing the data in the dialog.

1. There are two types of translations, global and local. Global translations translate the origin of the local coordinate system relative to the global Cartesian system. By default, FRANC3D creates a local coordinate system and translates it so that the flaw sits just in front of the model. Local translations will translate the flaw relative to the local coordinate system.

2. There are three rotation options; these rotations can be about the global Cartesian axes or about the local coordinate system.

3. There is a button to define the local coordinate system.

4. There are additional display options to turn on/off node numbers, the local (single prime) coordinate system axes, and the crack (double prime) coordinate system axes.



Flaw Insertion

Flaw translation

global: X: 5, Y: 5, Z: 10

local: X': 0, Y': 0, Z': 0

Flaw rotations

1st Rotation: Axis ● X ○ Y ○ Z; ● Global ○ Local; Angle (deg) 0

2nd Rotation: Axis ○ X ● Y ○ Z; ● Global ○ Local; Angle (deg) 0

3rd Rotation: Axis ○ X ○ Y ● Z; ● Global ○ Local; Angle (deg) 0

Display: □ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh □ Node Num ☑ Local Axes □ Crack Axes

(reset) (d)

Figure 6.27 Flaw placement dialog.

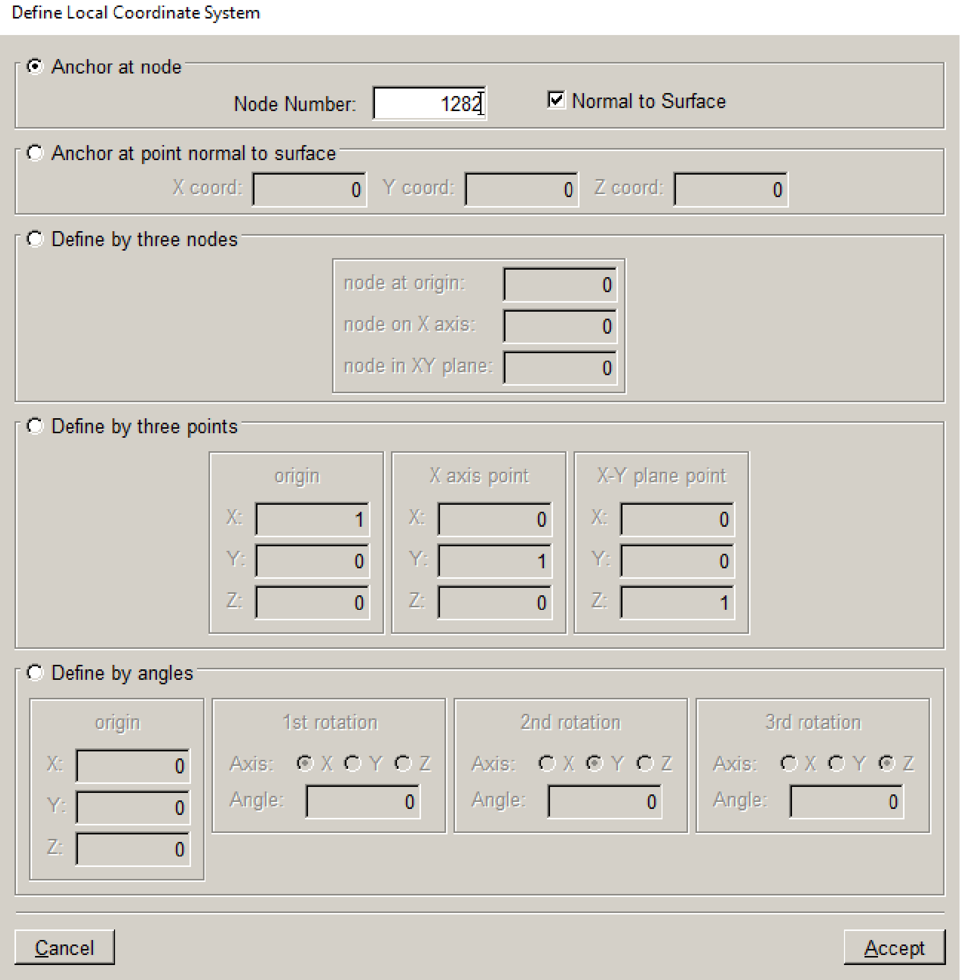
6.5.2 Define Local Axes

Selecting the **Define Local Axes** button in Fig 6.27 will bring up the dialog box in Fig 6.28, where there are options to define the local coordinate system. Some options will orient the crack so that it is normal to the model surface, but it might require additional rotations to get the correct orientation. Each option is described below.

For the first option, the user just specifies a node number on the model surface. A local coordinate system is defined where the y-axis of the flaw is oriented along the direction of the local normal to the surface at the node.

The second option is like the first, but you specify the coordinates of a point on the model surface. FRANC3D finds the point on the surface of the model closest to the specified point and orients the flaw normal to the local surface normal at this point.

The third option defines a local coordinate system by giving three node numbers. The coordinates of the first node become the origin of the local system. The coordinates of the second node define a point on the local x axis, and the coordinates of the third node define a point that lies in the local x-y plane.



Define Local Coordinate System

● Anchor at node

Node Number: 1282 ☑ Normal to Surface

○ Anchor at point normal to surface

X coord: 0 Y coord: 0 Z coord: 0

○ Define by three nodes

node at origin: 0
node on X axis: 0
node in XY plane: 0

○ Define by three points

	origin	X axis point	X-Y plane point
X:	1	0	0
Y:	0	1	0
Z:	0	0	1

○ Define by angles

origin X: 0 Y: 0 Z: 0

1st rotation Axis: ● X ○ Y ○ Z Angle: 0

2nd rotation Axis: ○ X ● Y ○ Z Angle: 0

3rd rotation Axis: ○ X ○ Y ● Z Angle: 0

Figure 6.28 Define local coordinate system dialog.

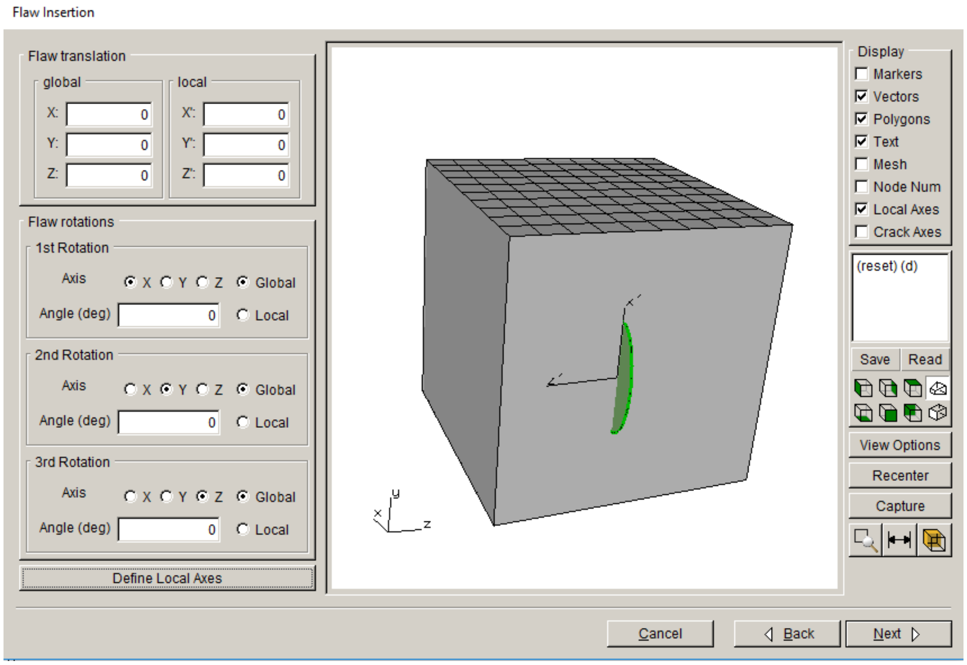
The fourth option is like the previous three-node option, but the user must specify the actual coordinates of the three points.

The fifth option uses the location of an origin and three rotations. The rotation axes can be specified in any order, and the rotations are defined about the global axes.

The flaw from Fig 6.27, oriented in a local coordinate system normal to the surface at a node, is shown in Fig 6.29.

The local y axis is now oriented into the body. The directions of the local x and z axis are defined so that the x is parallel to one of the x-y, y-z, or z-x planes, whichever gives the most numerically stable axis rotations.

Notice that the global translations have been reset to zero. Any changes to these will move the origin of the local coordinate system relative to the coordinates of the specified surface point.



Flaw Insertion

Flaw translation

global: X: 0, Y: 0, Z: 0

local: X': 0, Y': 0, Z': 0

Flaw rotations

1st Rotation — Axis: ● X ○ Y ○ Z; ● Global ○ Local; Angle (deg): 0

2nd Rotation — Axis: ○ X ● Y ○ Z; ● Global ○ Local; Angle (deg): 0

3rd Rotation — Axis: ○ X ○ Y ● Z; ● Global ○ Local; Angle (deg): 0

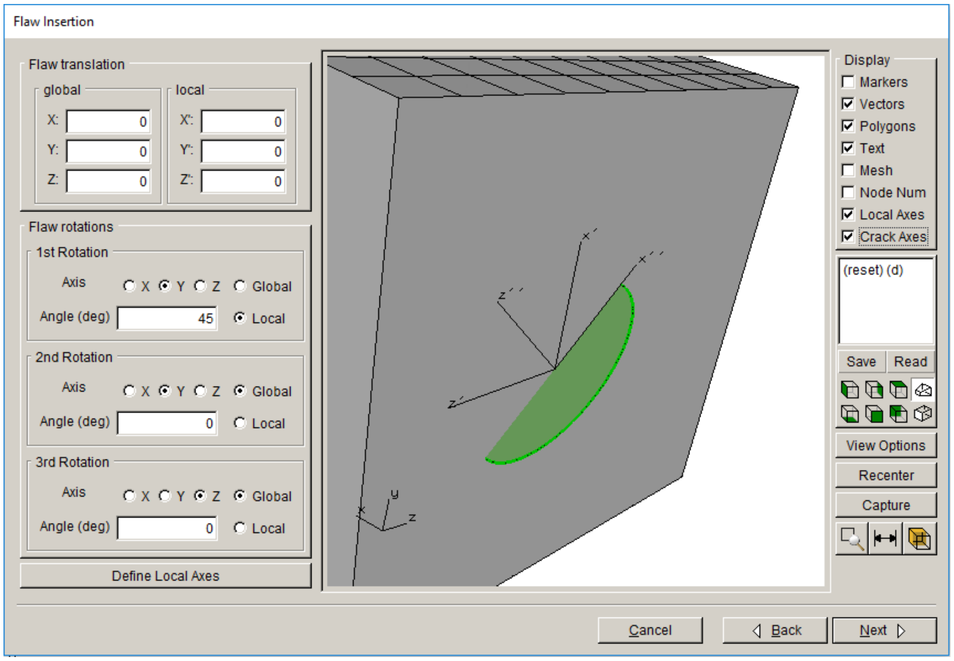
Display

□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Node Num
☑ Local Axes
□ Crack Axes

(reset) (d)

Figure 6.29 Crack oriented in a local coordinate system.

To orient the flaw normal to the local maximum principal tensile stress, one can define a local rotation about the y axis, Fig 6.30. The intrinsic crack axes (double primes) have been turned on to illustrate the relationships among the coordinate systems. Fig 6.31 shows how a translation in the local y direction can be used to modify the portion of the flaw that is inserted into the body.



Flaw Insertion

Flaw translation

global

X: 0

Y: 0

Z: 0

local

X': 0

Y': 0

Z': 0

Flaw rotations

1st Rotation

Axis ○ X ● Y ○ Z ○ Global

Angle (deg) 45 ● Local

2nd Rotation

Axis ○ X ● Y ○ Z ● Global

Angle (deg) 0 ○ Local

3rd Rotation

Axis ○ X ○ Y ● Z ● Global

Angle (deg) 0 ○ Local

Display

☐ Markers

☑ Vectors

☑ Polygons

☑ Text

☐ Mesh

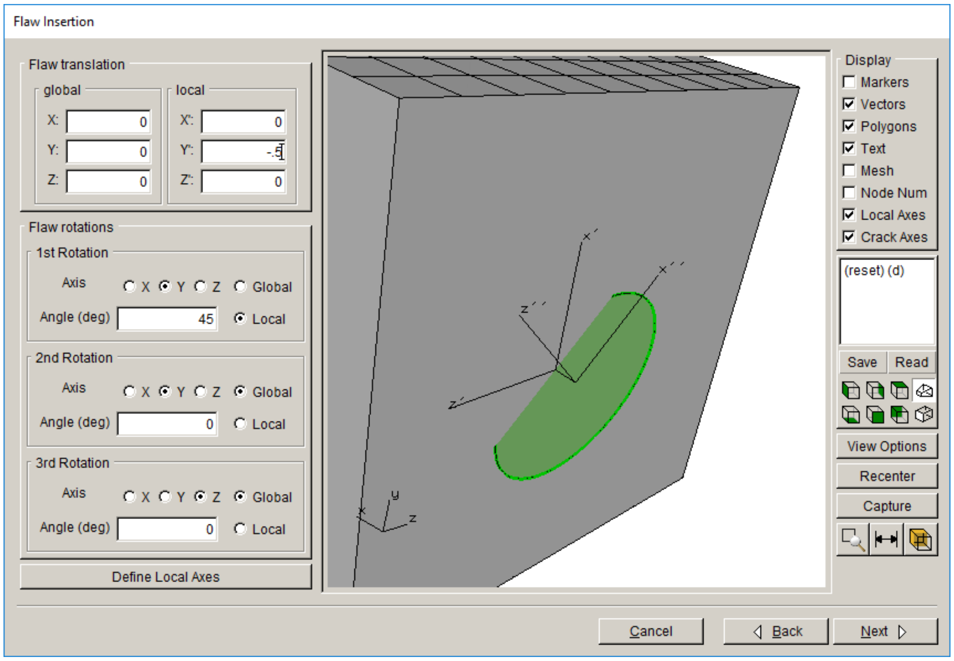
☐ Node Num

☑ Local Axes

☑ Crack Axes

(reset) (d)

Figure 6.30 Crack oriented in a local coordinate system and rotated about the local y-axis.



Flaw Insertion

Flaw translation

global

X: 0

Y: 0

Z: 0

local

X': 0

Y': -5

Z': 0

Flaw rotations

1st Rotation

Axis ○ X ● Y ○ Z ○ Global

Angle (deg) 45 ● Local

2nd Rotation

Axis ○ X ● Y ○ Z ● Global

Angle (deg) 0 ○ Local

3rd Rotation

Axis ○ X ○ Y ● Z ● Global

Angle (deg) 0 ○ Local

Display

☐ Markers

☑ Vectors

☑ Polygons

☑ Text

☐ Mesh

☐ Node Num

☑ Local Axes

☑ Crack Axes

(reset) (d)

Figure 6.31 Crack oriented in a local coordinate system and rotated about the local y-axis and translated along the local y-axis.

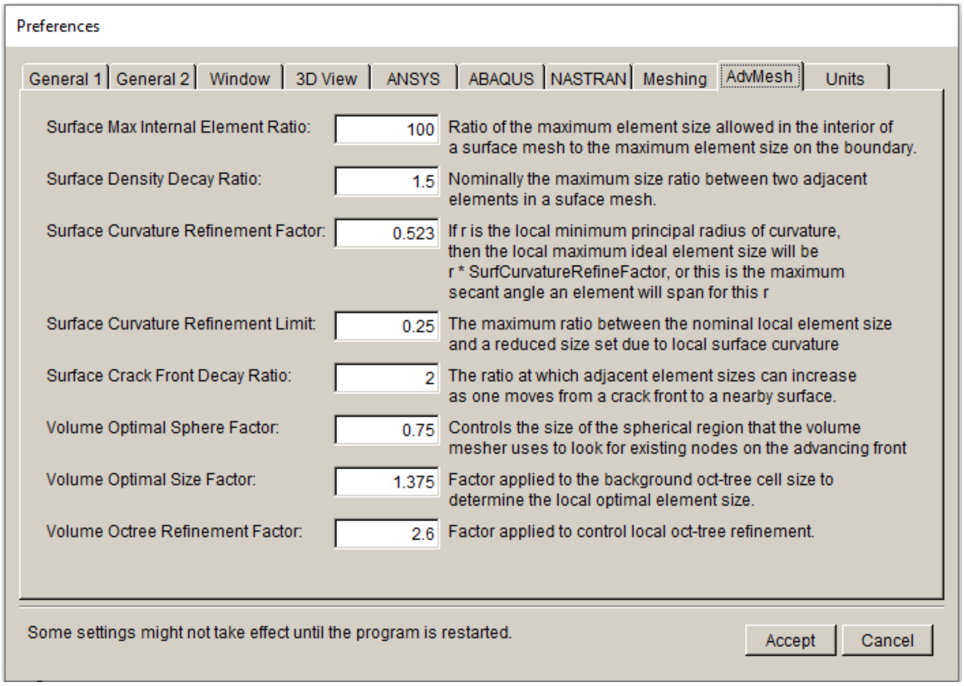
7. Meshing

This chapter discusses "advanced" settings for meshing and for the crack front template. Usually the default settings are sufficient, but in some cases, the user might want to study the effects of mesh density on SIFs. The advanced settings allow more user-control.

7.1 Surface and Volume Mesh Settings

The *Advanced Meshing* tab in the FRANC3D Preferences dialog, Fig 7.1, allows the user to alter the default advanced meshing parameters.

Meshing is done in two phases. Surfaces are meshed first, and the first five parameters in the dialog 'roughly' control the surface mesh refinement. Once the surfaces are meshed, then the volume is meshed. Note that the volume can be meshed using FRANC3D's internal advancing-front algorithm, or it can be meshed using either ABAQUS or ANSYS volume meshing capabilities. The last three parameters in the dialog 'roughly' control the volume mesh refinement when using FRANC3D's volume meshing capability.



Surface Max Internal Element Ratio:	100	Ratio of the maximum element size allowed in the interior of a surface mesh to the maximum element size on the boundary.
Surface Density Decay Ratio:	1.5	Nominally the maximum size ratio between two adjacent elements in a suface mesh.
Surface Curvature Refinement Factor:	0.523	If r is the local minimum principal radius of curvature, then the local maximum ideal element size will be r * SurfCurvatureRefineFactor, or this is the maximum secant angle an element will span for this r
Surface Curvature Refinement Limit:	0.25	The maximum ratio between the nominal local element size and a reduced size set due to local surface curvature
Surface Crack Front Decay Ratio:	2	The ratio at which adjacent element sizes can increase as one moves from a crack front to a nearby surface.
Volume Optimal Sphere Factor:	0.75	Controls the size of the spherical region that the volume mesher uses to look for existing nodes on the advancing front
Volume Optimal Size Factor:	1.375	Factor applied to the background oct-tree cell size to determine the local optimal element size.
Volume Octree Refinement Factor:	2.6	Factor applied to control local oct-tree refinement.

Some settings might not take effect until the program is restarted.

Figure 7.1 Advanced Meshing tab of FRANC3D Preferences dialog.

Surface meshing is most easily explained by mapping a 3D surface to a 2D plane. The 2D surface mesh is based on a quadtree structure that divides the region. Internal nodes are generated within the boundary of the region on the 2D plane. This is done by decomposing the region using a quadtree recursive spatial decomposition (RSD) procedure. RSD algorithms act upon a region of space and subdivide the region into smaller regions of a similar shape. The process is repeated an arbitrary number of times for each of the smaller regions.

Fig 7.2 shows two examples of RSD procedures. In the first, a triangular region is subdivided into four similar triangular regions recursively. In the second, a rectangular region is subdivided into four similar rectangular regions recursively. RSD procedures only work for a small number of region shapes, in two-dimensions these include triangles, rectangles, and hexagons. For a rectangular region, every subregion is either undivided or divided into four similar regions. This information can be stored conveniently in a tree data structure, where each node in the tree has exactly zero or four children. Such a data structure is called a *quadtree*.

Fig 7.3 shows a simple example of a divided region and its corresponding quadtree. The undivided regions, which correspond to leaf nodes in the tree, are called terminal quadrants. The size of a terminal quadrant can be determined from its level in the tree.

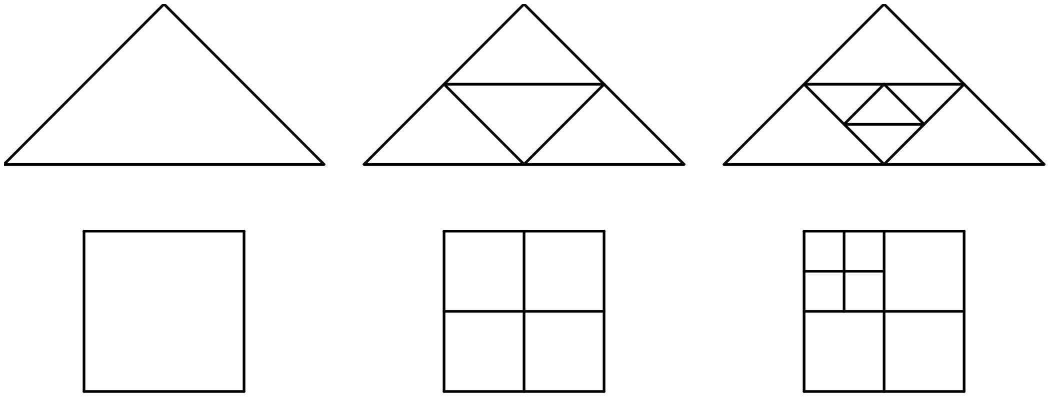


Figure 7.2 A triangular and a rectangular recursive subdivision procedure.

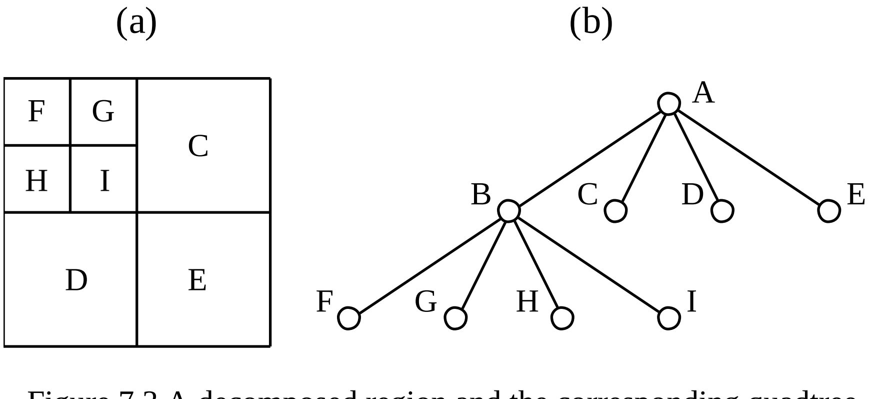


Figure 7.3 A decomposed region and the corresponding quadtree.
Terminal quadrants are labeled in (a).

In the meshing algorithm, the level of decomposition is a function of the nodal spacing on the nearby boundaries. The surface refinement factor then controls how quickly the level of decomposition transitions towards the interior of the region. A refinement factor of one indicates that the decomposition can only change by one level for adjacent quadtree cells. The surface boundary factor controls the placement of internal points near the boundaries. A small value allows points to be placed close to the boundary. Internal nodes can be generated either at the center or the corners of the terminal quadtree cells if the location is sufficiently far from the boundary, Fig 7.4.

A mesh comprised of triangles is generated using a boundary contraction scheme, Fig 7.5. A list of all the boundary edges is created. Starting with the first edge in the list, all the internal and boundary nodes are assessed to determine which node produces the largest included angle when edges are extended from the two ends of the edge to the node. The best vertex is then used to create a triangular element. The list of boundary edges is updated, removing the edge that was started from and adding the new edges if they are not part of an existing element. The region, therefore, contracts by extracting elements one at a time.

The mesh is smoothed by repositioning each internal node to lie at the centroid of its surrounding polygon. The smoothing uses an iterative approach in which each internal node is repositioned based on the current nodal positions of its surrounding polygon. This process is repeated until there is no change of nodal positions between iterations. .

In FRANC3D, the surface meshing algorithm allows a face of arbitrary shape to be meshed using all triangular elements. There are many subtle aspects of surface meshing that must be considered to produce a good mesh for volume meshing. For example, the surface curvature is considered in this paper:

Surface mesh regeneration considering curvatures.
Miranda et al., Engineering with Computers (2009) 25:207–219

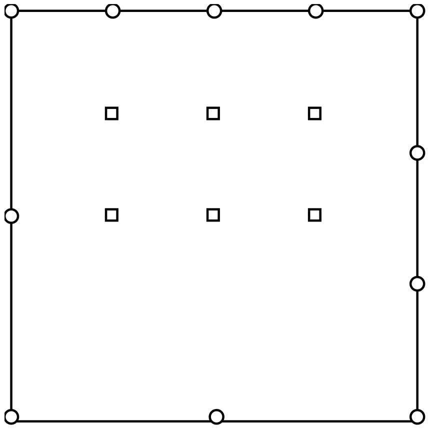


Figure 7.4 Generated interior points using a quadtree procedure.

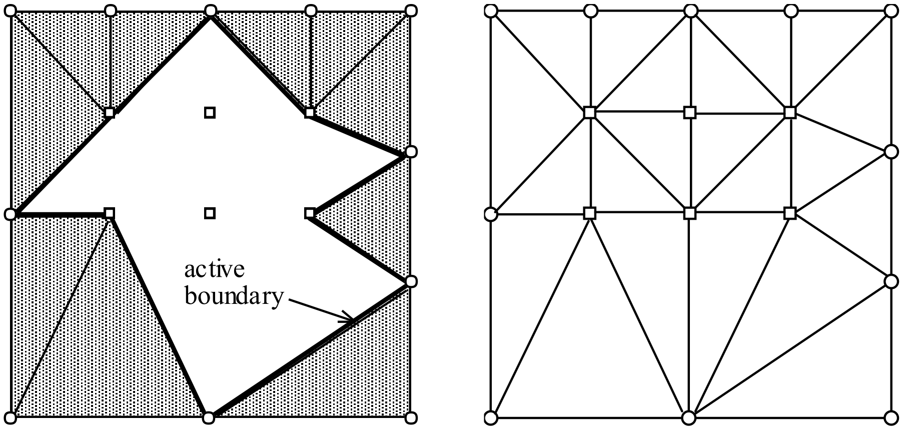


Figure 7.5 Generated triangular element mesh using a boundary contraction procedure.

We will discuss additional FRANC3D settings below in relation to cracks and accurate stress intensity factors (SIFs).

The basis for the 3D volume meshing algorithm in FRANC3D is described in journal articles. For example:

> *An Algorithm for Three-Dimensional Mesh Generation for Arbitrary Regions with Cracks, Cavalcante et al., Engineering with Computers (2001) 17: 75–91.*

The 2D quadtree data structure is replaced with a 3D octree. The triangular surface mesh facets created during surface meshing comprise the boundary of the 3D region. The octree is defined such that it encompasses the region volume and octree cells contain information about the bounding triangles. The advancing front algorithm uses up the boundary triangles to form tetrahedral elements, and the front is updated by removing used and adding any new triangular faces.

The volume octree refinement factor (see Fig 7.1) is similar to the quadtree surface refinement factor. It controls how quickly the octree cell subdivision occurs when moving from the boundary to the interior. The optimal size and sphere factors (see Fig 7.1) are described in Cavalcante *et al*; Fig 7.6, which is extracted from the paper, shows a sphere in relation to the current triangular boundary face being examined. If the optimal size is decreased, this allows for smaller elements and slower transitions to larger elements. The factors do not allow for real fine-tuning of the mesh density, however.

In some cases, the advancing front algorithm can get stuck, and the algorithm must backtrack. FRANC3D allows the user to control the number of backtracking attempts as well as the maximum number of volume elements allowed, Fig 7.7. The backtracking process is described in:

> *A back-tracking procedure for optimization of simplex meshes.*
> *Cavalcante et al., Commun. Numer. Meth. Engng., 2005; 21:711–722.*

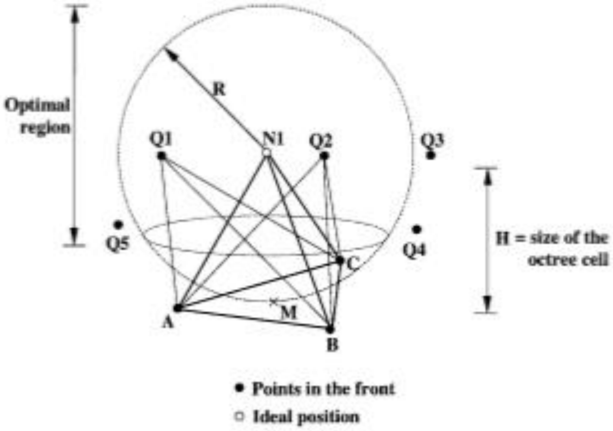
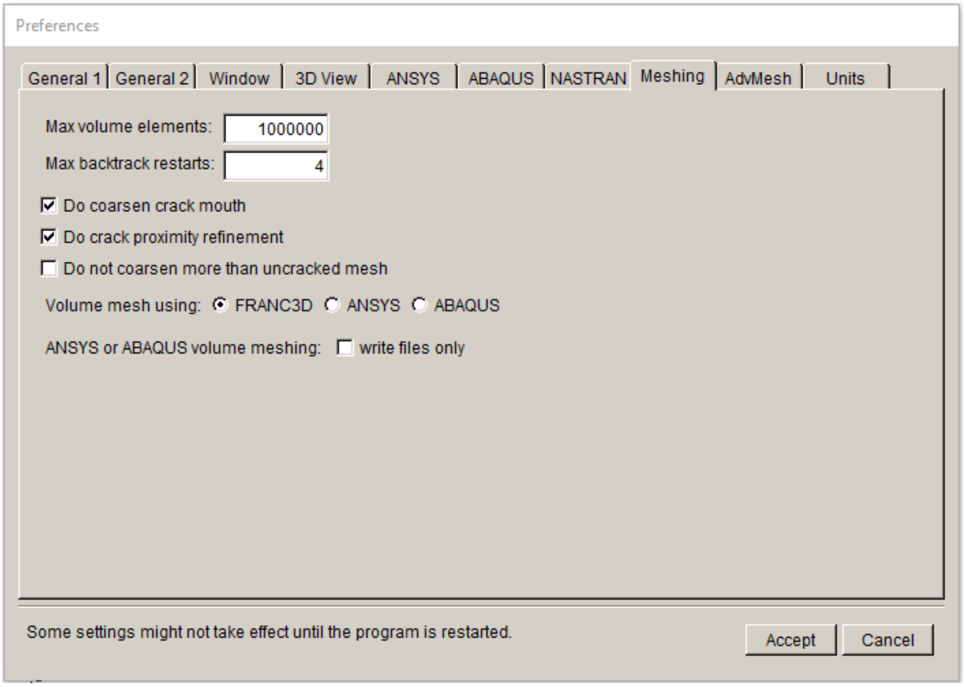


Figure 7.6 Generating optimal tetrahedral element using a boundary contraction procedure.



Max volume elements: 1000000

Max backtrack restarts: 4

☑ Do coarsen crack mouth

☑ Do crack proximity refinement

☐ Do not coarsen more than uncracked mesh

Volume mesh using: ● FRANC3D ○ ANSYS ○ ABAQUS

ANSYS or ABAQUS volume meshing: ☐ write files only

Some settings might not take effect until the program is restarted.

Figure 7.7 Meshing tab of FRANC3D Preferences dialog.

Fig 7.7 shows the option for choosing FRANC3D or ABAQUS or ANSYS volume meshing. The Max volume elements and Max backtrack events only apply to the FRANC3D volume meshing.

Fig 7.7 also shows three additional user options:

1) Do coarsen crack mouth
2) Do crack proximity refinement

3) Do not coarsen more than the uncracked mesh

These options are discussed in Section 6.1.15 of the Reference document. We will repeat some of that information here with additional images and details.

The Do coarsen crack… option allows the user to 'roughly' control the surface mesh density on the crack surface. Note that this option sometimes gets overruled by the underlying geometry and the advancing front algorithm. In general, the default is to coarsen the crack surface mesh as it transitions from the template radius (crack front) towards the crack mouth (intersection of the model free surface). This reduces the overall number of elements while not significantly affecting the SIF accuracy in most cases. Fig 7.8 shows the same crack with the option on (left) and off (right).

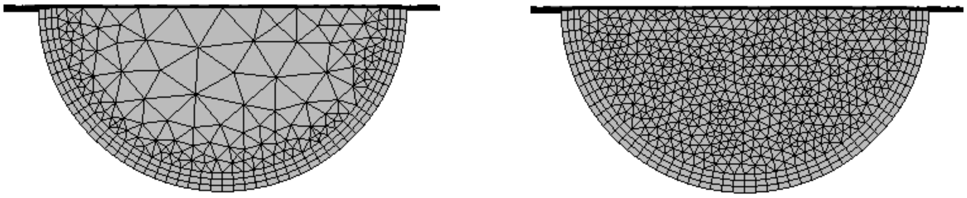


Figure 7.8 Do coarsen crack… option: on – left image and off – right image.

This option can be important for shallow edge cracks in thin plates. Fig 7.9 shows a shallow surface crack in a thin plate. To produce enough elements through the thickness of the plate and along the crack front, to compute accurate (SIFs), the template radius must be relatively small. The meshing parameters can be set when defining the crack front template, Fig 7.10. Fig 7.11 (top image) shows the crack surface mesh with the Do coarsen crack… turned on. The bottom image in Fig 7.11 shows the crack surface mesh when this option is turned off.

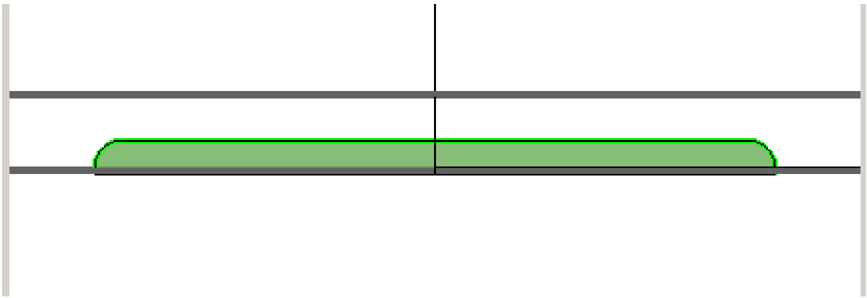
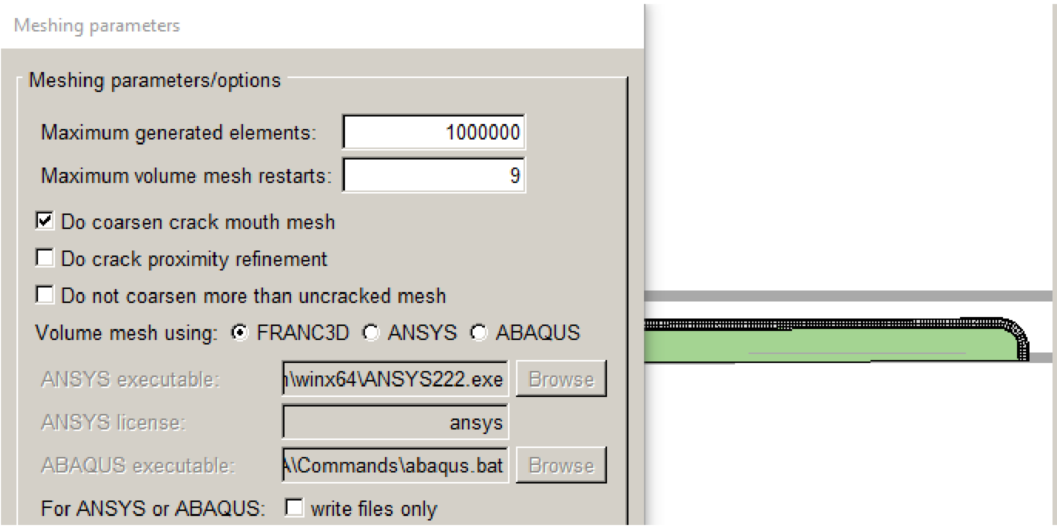


Figure 7.9 Shallow surface crack in a thin plate.



Meshing parameters

Meshing parameters/options

Maximum generated elements: 1000000

Maximum volume mesh restarts: 9

☑ Do coarsen crack mouth mesh

☐ Do crack proximity refinement

☐ Do not coarsen more than uncracked mesh

Volume mesh using: ● FRANC3D ○ ANSYS ○ ABAQUS

ANSYS executable: \winx64\ANSYS222.exe Browse

ANSYS license: ansys

ABAQUS executable: \Commands\abaqus.bat Browse

For ANSYS or ABAQUS: ☐ write files only

Figure 7.10 Meshing parameters dialog.

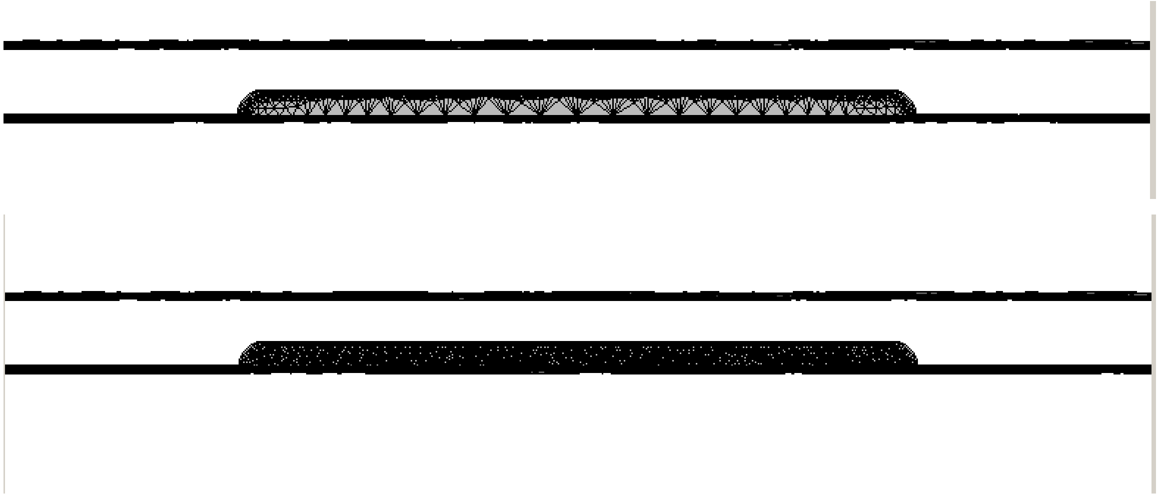


Figure 7.11 Surface mesh on shallow surface crack with Do coarsen crack… turned on (top) and turned off (bottom).

The Do crack proximity… option is useful when a crack front is approaching a model surface that is not intersected by the crack. Fig 7.12 shows a surface crack front that is approaching the back surface of the plate. If the Do crack proximity… is off, the resulting surface mesh on the back surface of the plate is shown in Fig 7.13. Fig 7.14 shows the same model with the Do crack proximity… turned on. The surface mesh near the crack front is refined. In some cases, this is the only way a volume mesh can be created in the space between the crack front and the plate surface.

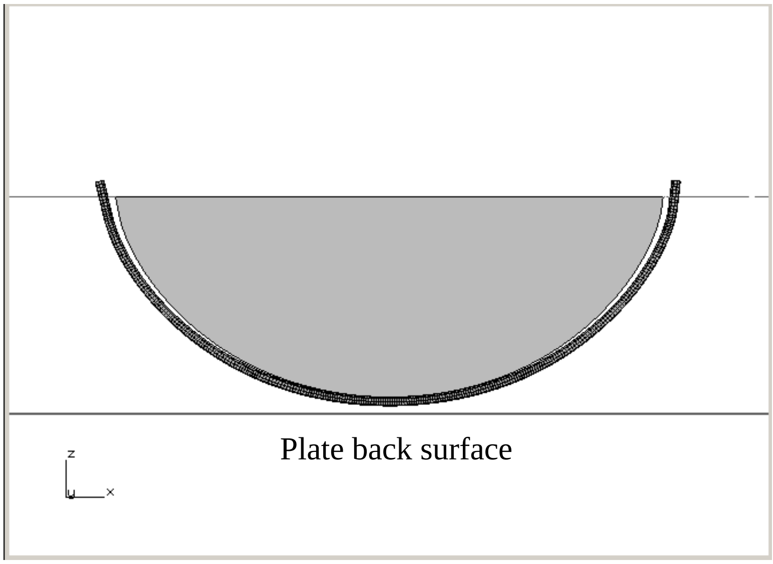


Figure 7.12 Surface crack front approaching back surface.

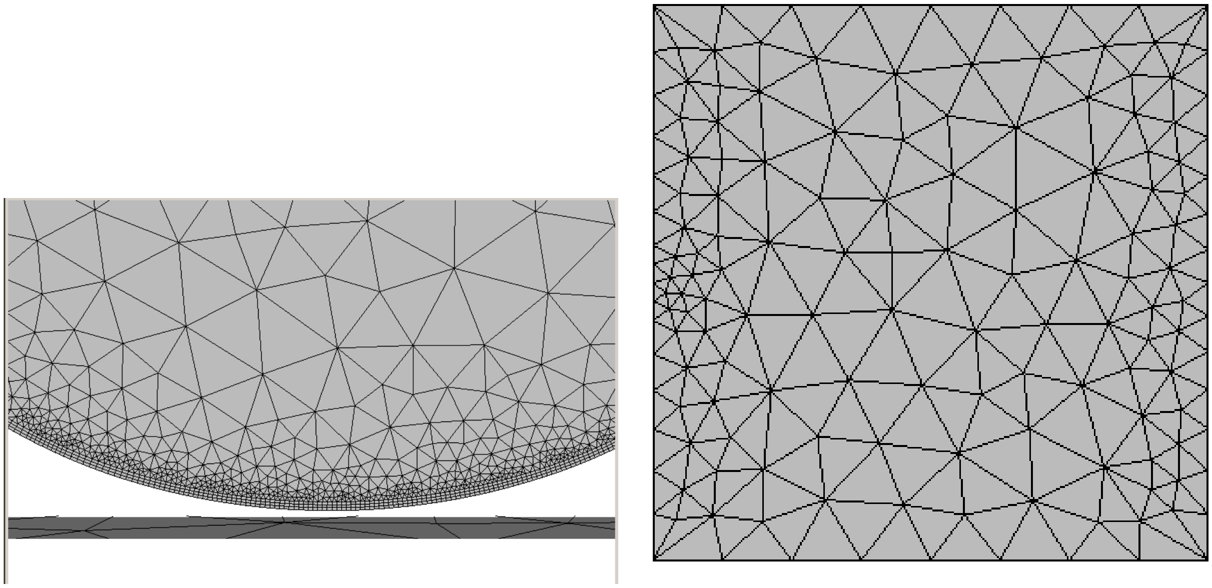


Figure 7.13 Surface mesh on back surface of plate with Do crack proximity… off.

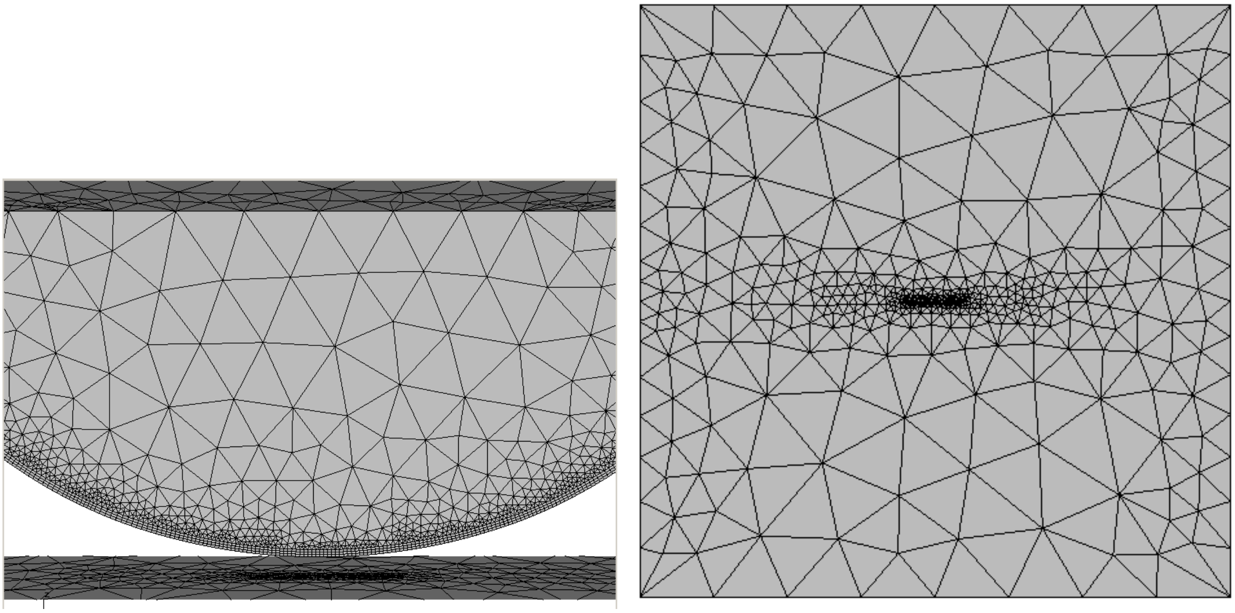


Figure 7.14 Surface mesh on back surface of plate with Do crack proximity… on.

The Do not coarsen more than the uncracked mesh aims to retain the original uncracked mesh density when inserting a crack and remeshing. Fig 7.15 shows the same model and crack with the option off (left image) and on; Fig 7.16 shows the original uncracked mesh. Note that the original mesh is composed of bricks with quadrilateral facets on the surface. The crack requires mesh refinement, but the extra option produces a surface mesh that is more like the original density away from the crack. Turning this option on will usually create more volume elements.

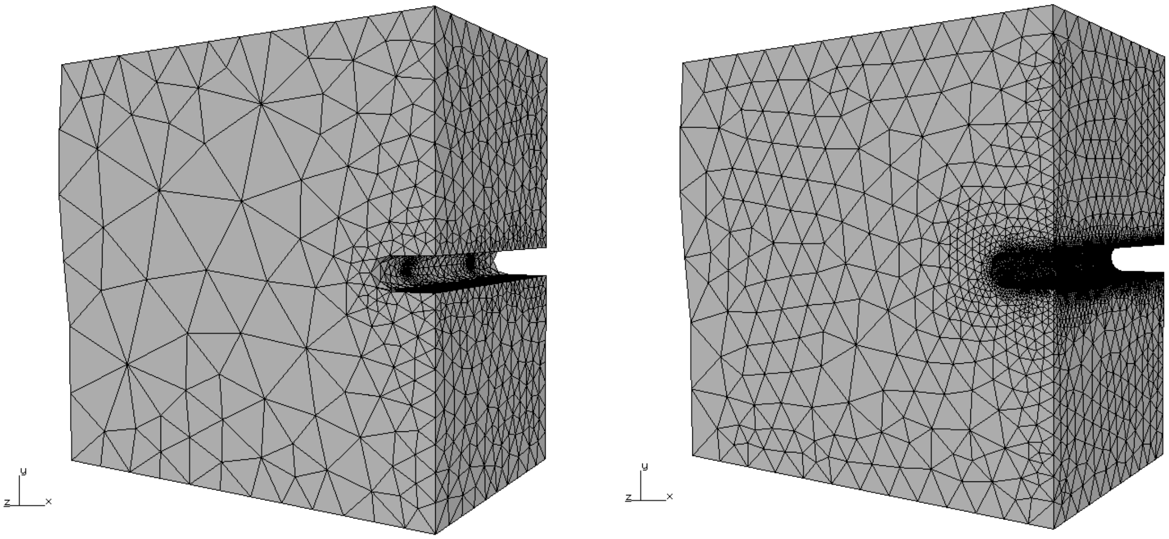


Figure 7.15 Do not coarsen more… option: off – left and on – right.

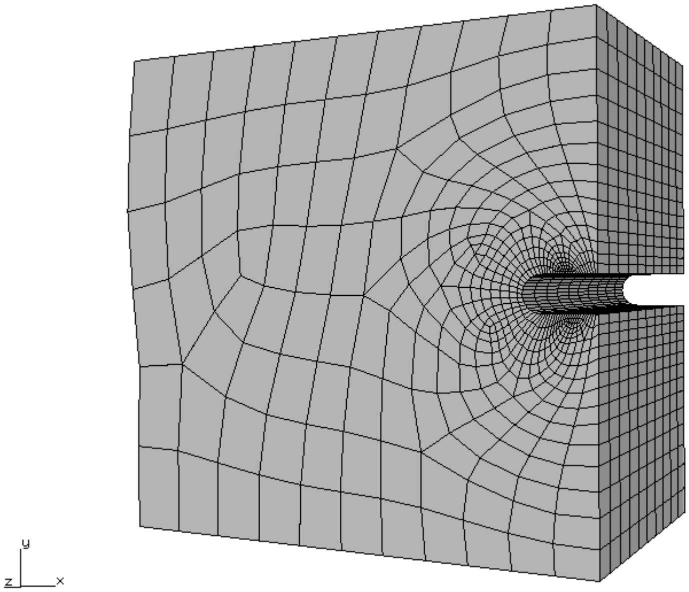
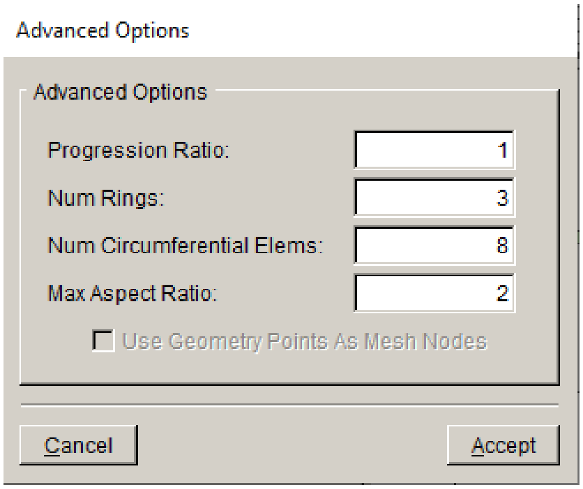


Figure 7.16 Original uncracked mesh.

7.2 Crack Front Template

Crack front template options are described in Section 6.1.15 of the Reference document. We will repeat some of that information here with additional images and details. The Advanced Options for the template, Fig 7.17, has default values that are based on studies from:

> *Methods for calculating stress intensity factors in anisotropic materials: Part II—Arbitrary geometry, Banks-Sills et al, EFM 74, 1293-1307.*



Advanced Options

Advanced Options

Progression Ratio: 1

Num Rings: 3

Num Circumferential Elems: 8

Max Aspect Ratio: 2

☐ Use Geometry Points As Mesh Nodes

Figure 7.17 Template advance options.

The options are depicted in Fig 7.18. In the Benchmark document, the default template settings were modified to study the accuracy of the SIFs; those studies were limited to one or two modifications, changing the number of rings from 3 to 5 for example. It was shown that the default settings generally produce SIFs within about 1-2% of the analytical or handbook values. However, modifying the template (as well as the mesh settings) could give us SIFs that are even more accurate – at the expense of significantly more elements and longer FE solution times.

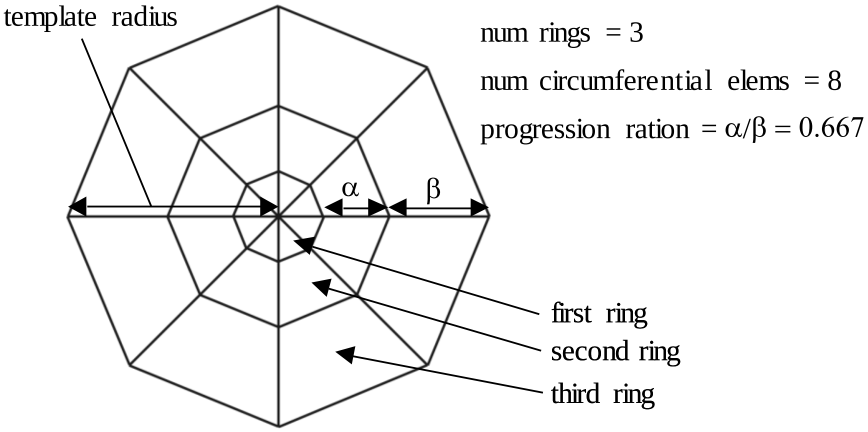

template radius
num rings = 3
num circumferential elems = 8
progression ration = α/β = 0.667
α
β
first ring
second ring
third ring

Figure 7.18 Template options.

The number of circumferential elements defaults to 8. This is sufficient for a crack in a single (elastic) material and provides accurate SIFs. For elasto-plastic J-integral computations, adding circumferential elements generally improves the accuracy. Also, for cracks embedded in bi-material interfaces, additional circumferential elements might be helpful.

The aspect ratio is not shown in Fig 7.18. It controls the "length" of the elements along the crack front in relation to the circumferential/radial dimension. Fig 7.19 shows three different aspect ratios with the other template settings held constant; the left is the default, the middle image has an aspect ratio of 1, and the right image has an aspect ratio of 4. For a curved crack front, the aspect ratio should not be more than 2.

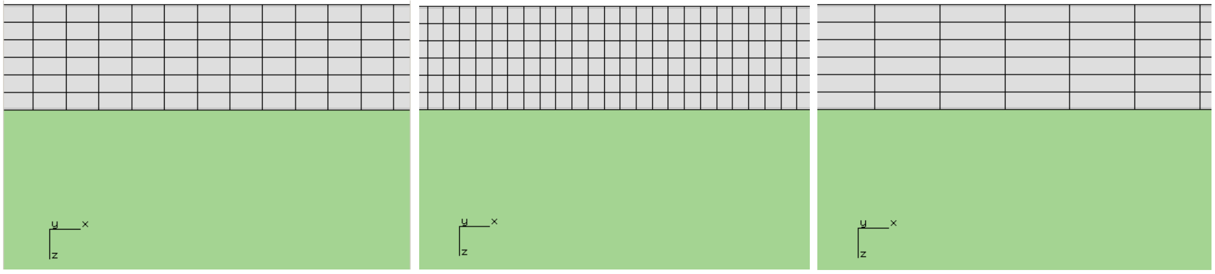


Figure 7.19 Template aspect ratio.

8. Potential Difficulties with Crack Growth

Crack growth in FRANC3D involves adding new crack surface geometry to the existing crack geometry and re-inserting the combined crack geometry into the uncracked (local) model. As the growth continues, the crack might encounter geometric features in the model geometry that cause growth to fail or stop with an error/information message. Section 5.8 of the Tutorials 2-12 document describes steps for transitioning through geometry in a simple plate model.

A couple more example situations are described below along with descriptions of how to continue the crack growth if it stops.

8.1 Crack Growth at Geometric Corners

This section describes crack growth around and through a geometric corner. A simple T-shaped ABAQUS model is shown in Fig 8.1. The model is constrained on the back (-z) side, and a fixed displacement is applied to the front (+z) side. The dimensions, properties and displacement values are not important as this example is intended only to give guidelines on crack growth procedures.

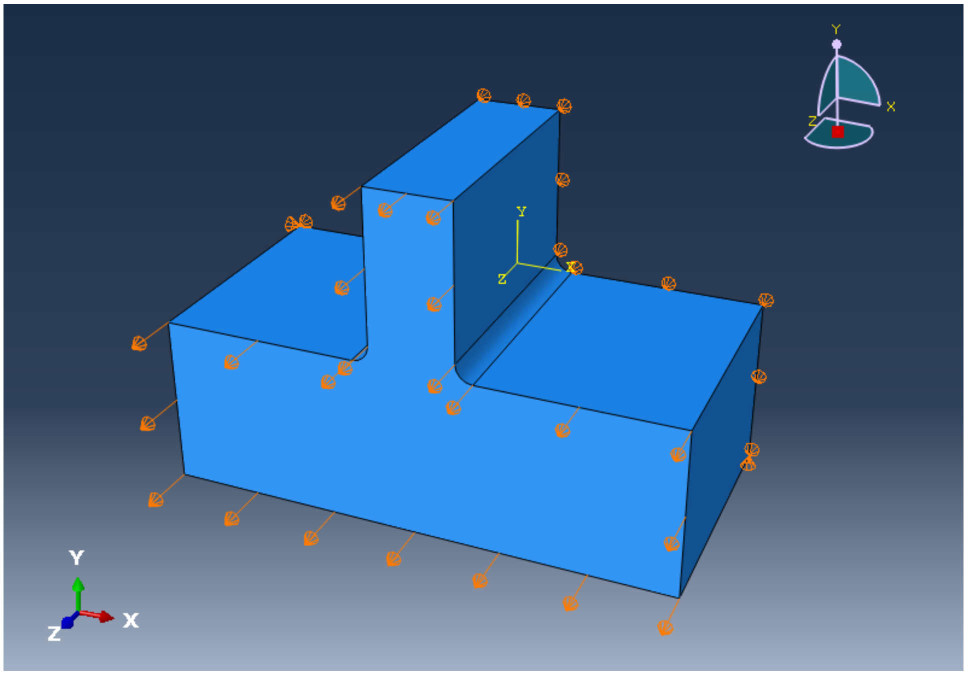
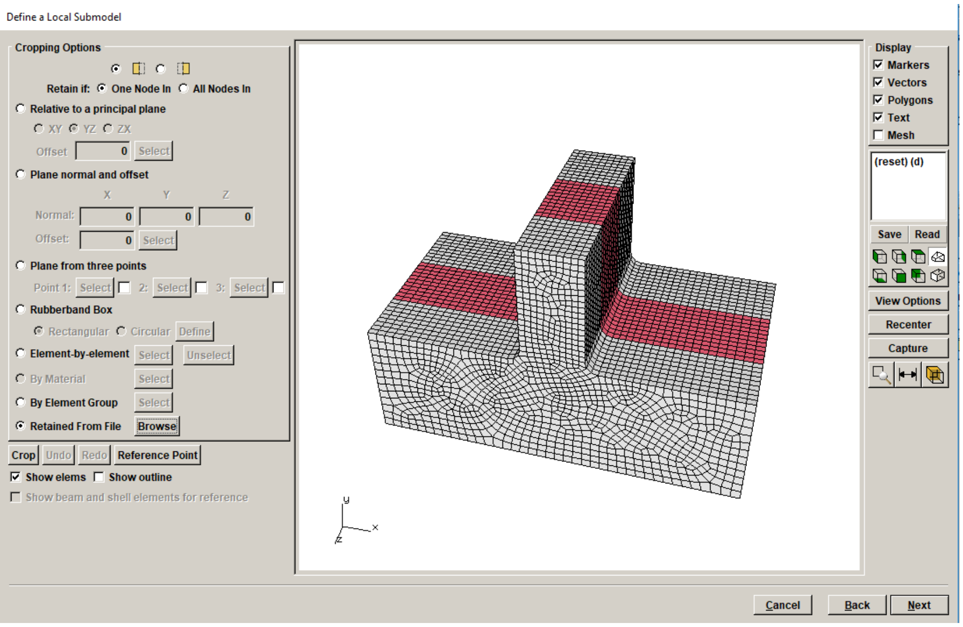


Figure 8.1 T-shape model.

The ABAQUS model is imported into FRANC3D and divided into local and global portions – the local portion is highlighted in Fig 8.2. An initial penny-shape crack, Fig 8.3, is inserted into the model to create an initial corner quarter-circular crack. The cracked and remeshed local portion is recombined with the global portion and analyzed in ABAQUS. The resulting Mode I SIF is shown in Fig 8.4 (the values and units are not important).

To simplify the crack growth process, planar crack growth is used with a quasi-static growth rule with a median crack growth step of 0.1. Automatic crack growth is then started, specifying 40 additional steps of growth. The default crack front fitting options – fixed 3rd order polynomial – are used throughout. The automatic growth stops at step #21 with a "fitting error".



Define a Local Submodel

Cropping Options

● ○

Retain if: ● One Node In ○ All Nodes In

○ Relative to a principal plane

○ XY ● YZ ○ ZX

Offset: 0

○ Plane normal and offset

Normal: X 0, Y 0, Z 0

Offset: 0

○ Plane from three points

Point 1: □ 2: □ 3: □

○ Rubberband Box

● Rectangular ○ Circular

○ Element-by-element

○ By Material

○ By Element Group

● Retained From File

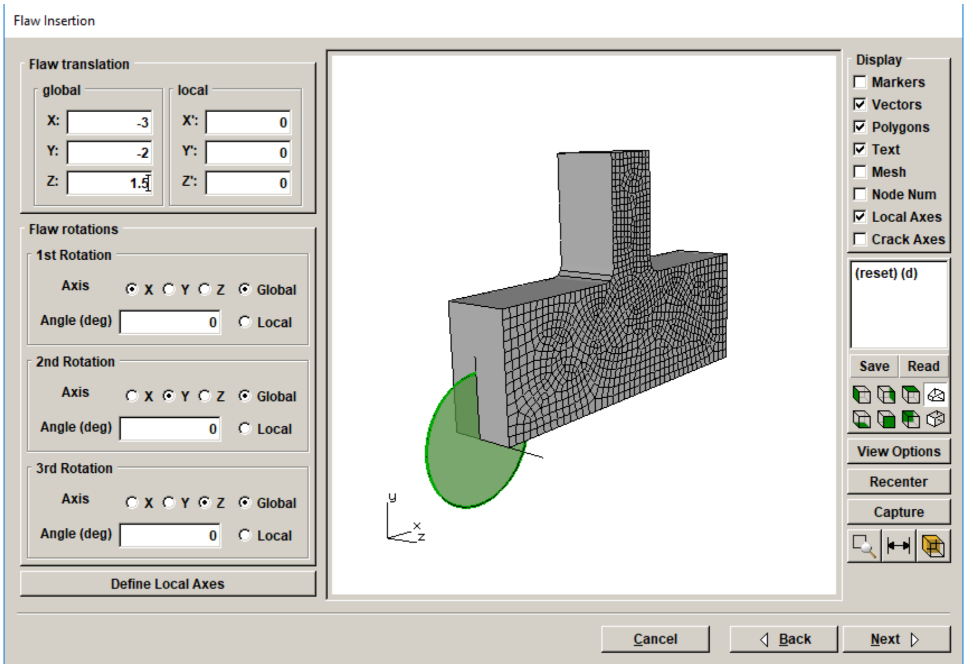
☑ Show elems □ Show outline

□ Show beam and shell elements for reference

Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh

(reset) (d)

Figure 8.2 T-shape model with local FRANC3D portion highlighted.



Flaw Insertion

Flaw translation

global: X: -3, Y: -2, Z: 1.5

local: X': 0, Y': 0, Z': 0

Flaw rotations

1st Rotation — Axis: ● X ○ Y ○ Z; ● Global ○ Local; Angle (deg): 0

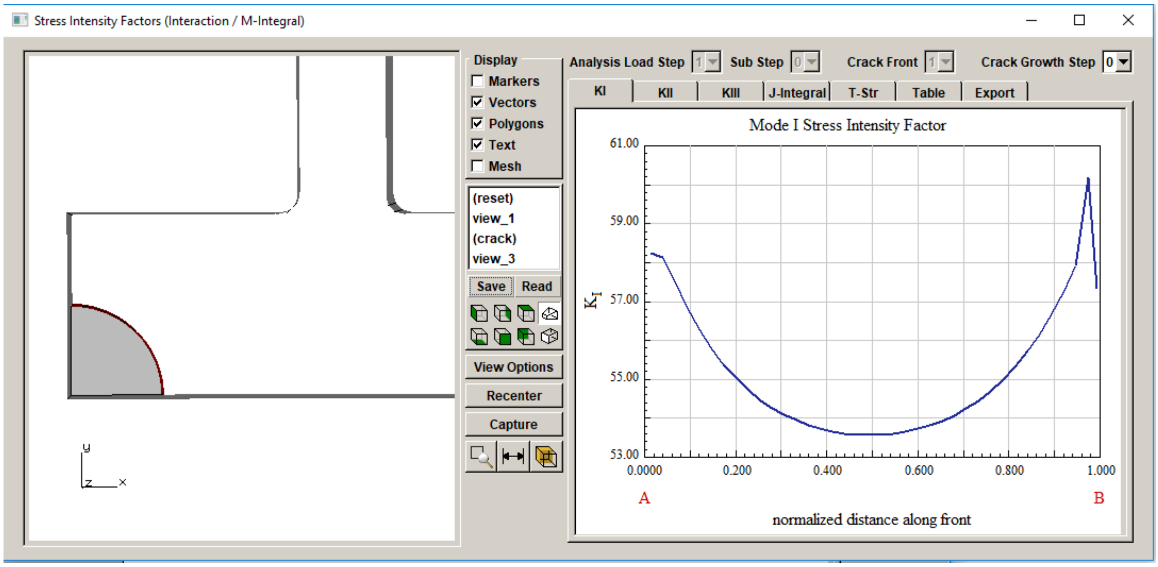
2nd Rotation — Axis: ○ X ● Y ○ Z; ● Global ○ Local; Angle (deg): 0

3rd Rotation — Axis: ○ X ○ Y ● Z; ● Global ○ Local; Angle (deg): 0

Display: □ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh □ Node Num ☑ Local Axes □ Crack Axes

(reset) (d)

Figure 8.3 Initial corner crack.


Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1
Sub Step 0
Crack Front 1
Crack Growth Step 0

Figure 8.4 Mode I SIFs for the initial corner crack.

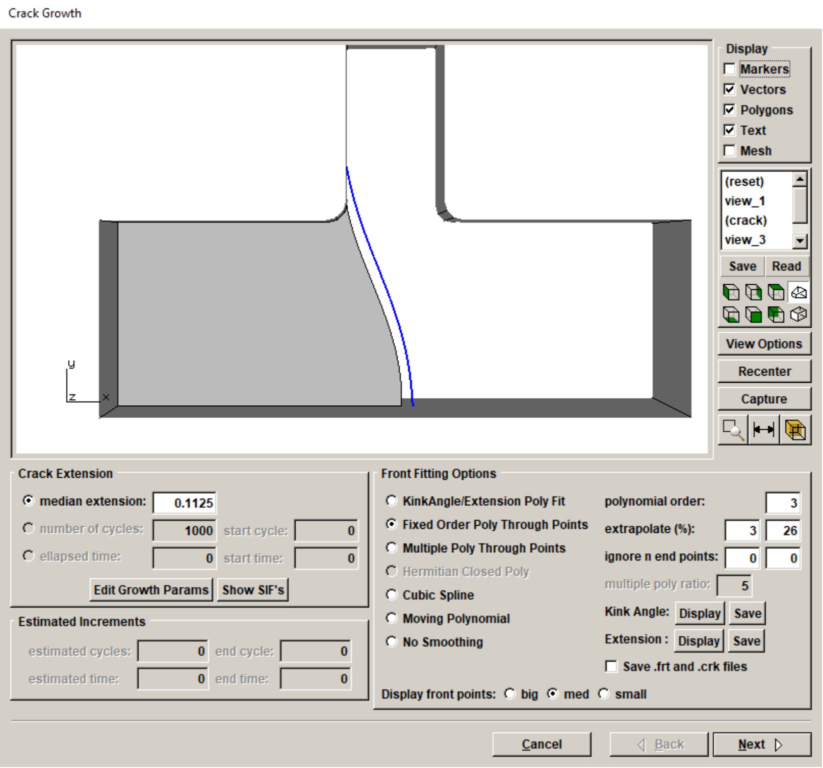
The crack front at step #21 is shown in Fig 8.5. Note that the default fitting options require excessive extrapolation at this stage to ensure that the curve intersects the model surface. If the fitting option fails to produce a good fit during automatic crack growth – growth will stop, and a message will be displayed indicating that the fitting failed.

To improve the fit, we switch to a moving polynomial with order=1 and cut down the extrapolation, Fig 8.6. The moving polynomial is a useful option, but it should be used with care as it will tend to capture more of the numerical noise in the SIFs (predicted points) and can lead to ever-increasing oscillations in the predicted fronts.

The crack is manually propagated to step #22 and analyzed. One more step of manual crack growth is performed at this stage to revise the fitting options again; Fig 8.7 shows the front fitted with a 2nd order polynomial with just enough extrapolation to ensure that the ends of the curve are outside of the model.

Note that the shallow intersection angle requires that we use "simple intersections" for the crack front template, Fig 8.8. To ensure that there are enough predicted new front points near the ends of the crack front, the template radius should be reasonably small, Fig 8.9.

The combination of numerical noise in the SIFs and simple intersections can cause problems with predicting good end points for the next crack front. To prevent additional numerical noise, the fitting options can be simplified to produce reasonable crack front shapes as the crack grows. Automatic crack growth using a fixed polynomial of 2nd order is set for an additional 20 steps with a median increment of 0.125.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Crack Extension

● median extension: 0.1125
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

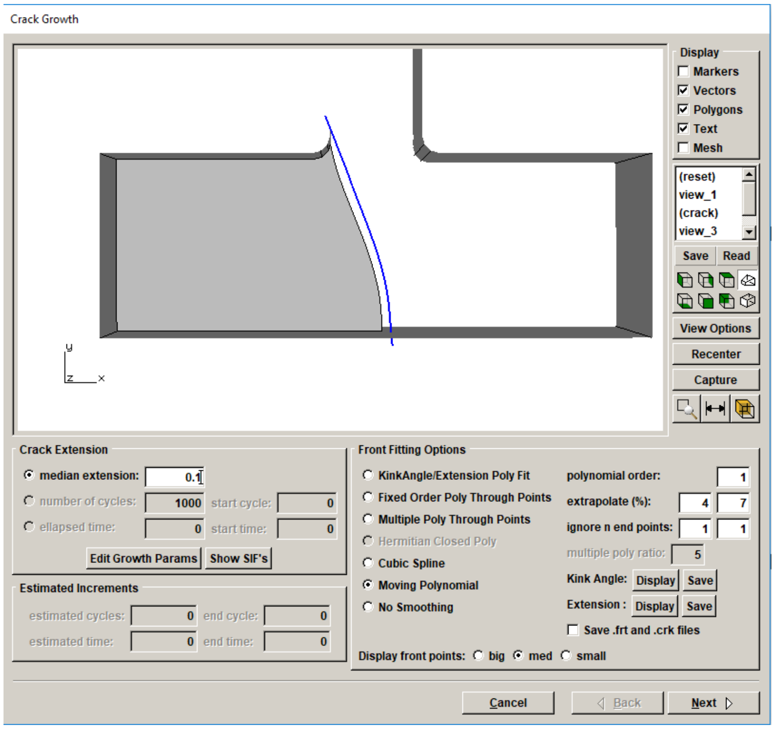
Front Fitting Options

○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 3
extrapolate (%): 3 26
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 8.5 Crack growth at step #21.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Crack Extension

● median extension: 0.1
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

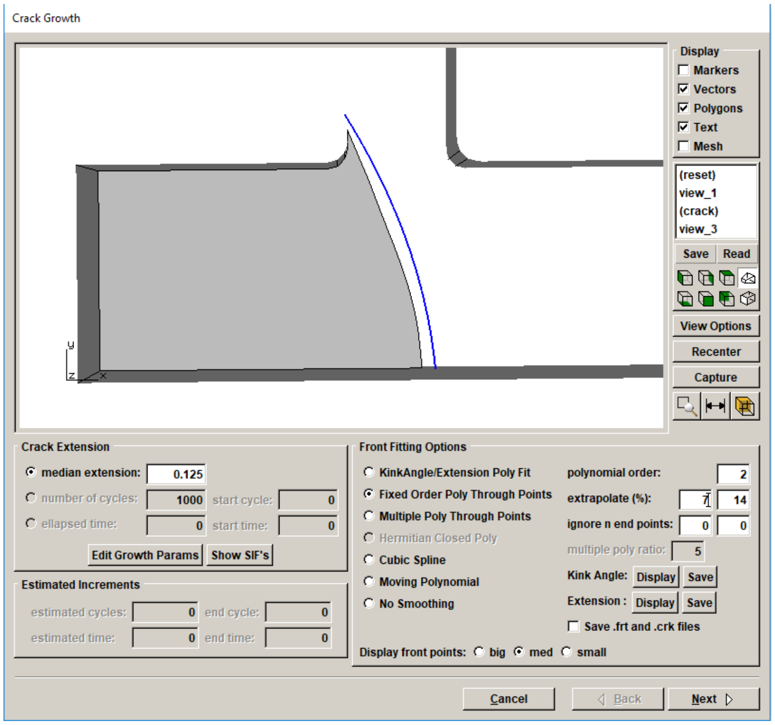
Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 4 7
ignore n end points: 1 1
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 8.6 Crack growth at step #21 with revised fitting options.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
(crack)
view_3

Crack Extension

● median extension: 0.125
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

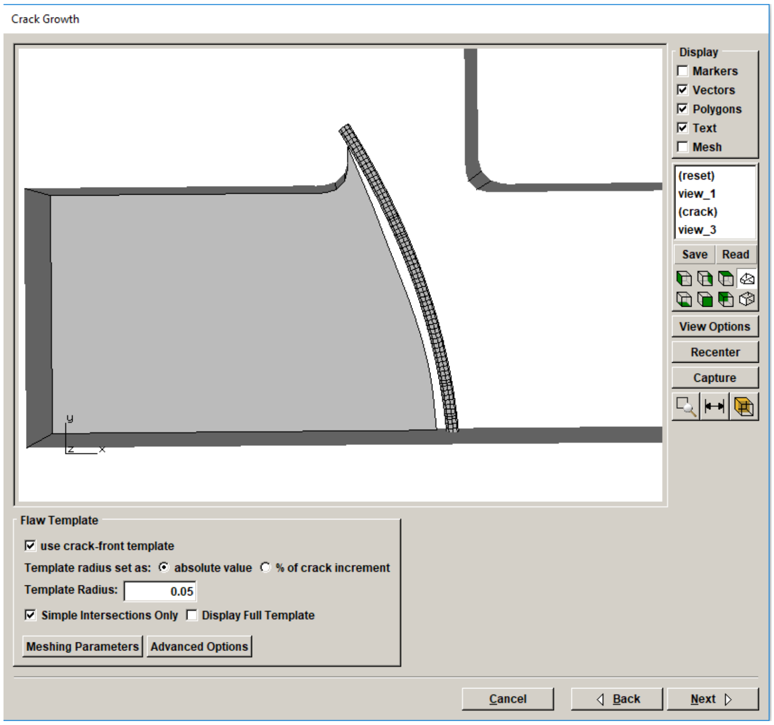
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options

○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 2
extrapolate (%): 7 14
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 8.7 Crack growth at step #22 with a 2nd order poly.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
(crack)
view_3

Flaw Template

☑ use crack-front template
Template radius set as: ● absolute value ○ % of crack increment
Template Radius: 0.05
☑ Simple Intersections Only ☐ Display Full Template

Figure 8.8 Crack growth at step #22 showing crack front template with simple intersections turned on.

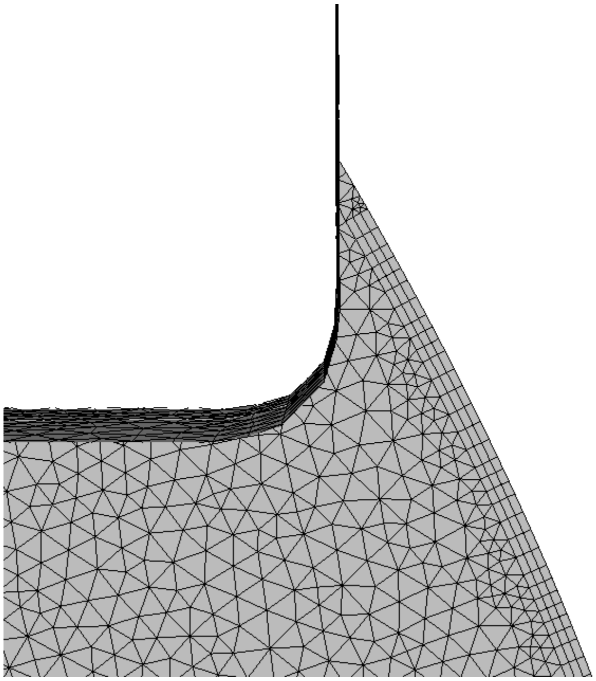


Figure 8.9 Crack front mesh at step #23 – simple intersections turned on.

The automatic crack growth continues from step #23 to #29. It fails to grow further because step #29 is not valid; the crack front reaches the corner of the T-shape, Fig 8.10, but it does not break through the corner correctly. FRANC3D needs to ensure that the crack surface geometry, the crack front template geometry, and the model geometry intersect and create “water-tight” regions that can be meshed. Step #29 leads to subsequent errors on the next step of growth due to current limitations when breaking the crack surface geometry to get two crack fronts instead of one front.

To continue crack growth, step #28 is read into FRANC3D, and the crack is re-grown with a larger median increment (0.25 instead of 0.125). This produces the crack shown in Fig 8.11. The crack front has clearly broken through the corner, and there is enough geometry outside of the model corner to allow FRANC3D to clearly separate the two crack fronts.

From step #29, automatic crack growth can be continued using the prior increment of 0.125 and a 2nd order polynomial fit. The automatic growth will continue until the crack front reaches the top of the T shape. Depending on the increment the growth might stop, or it might cut through the top completely – so that there is only a single front again, Fig 8.12. Crack growth might stop prior to cutting through the top of the T if the model cannot be meshed because the template is too close to the top surface. If it stops, restart from the previous step and take a bigger step of growth.

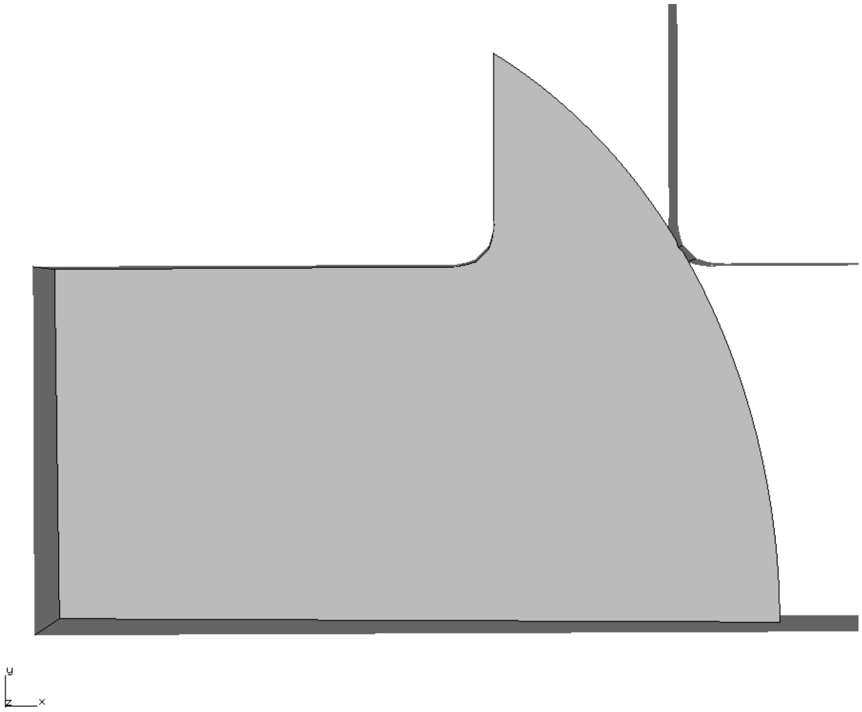


Figure 8.10 Crack front at step #29.

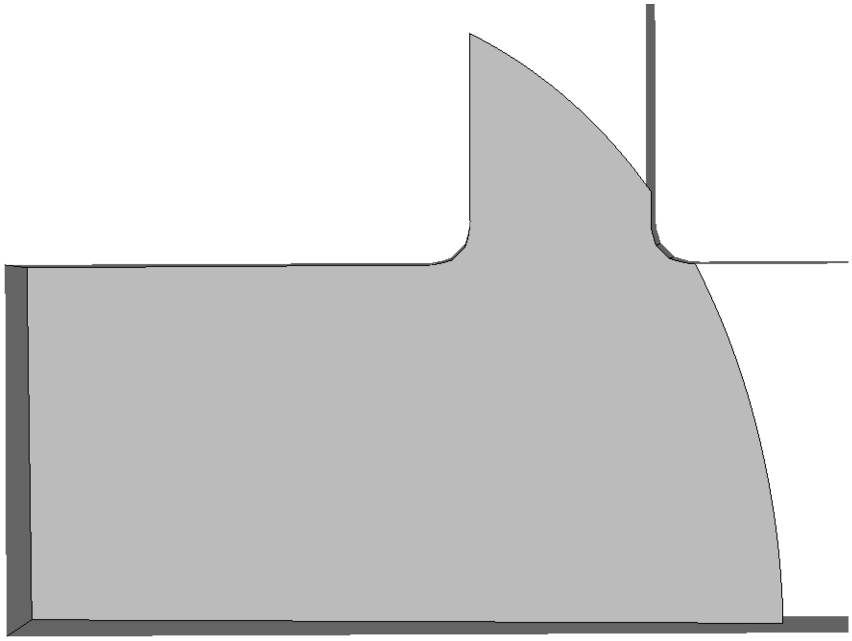


Figure 8.11 Revised crack front at step #29.

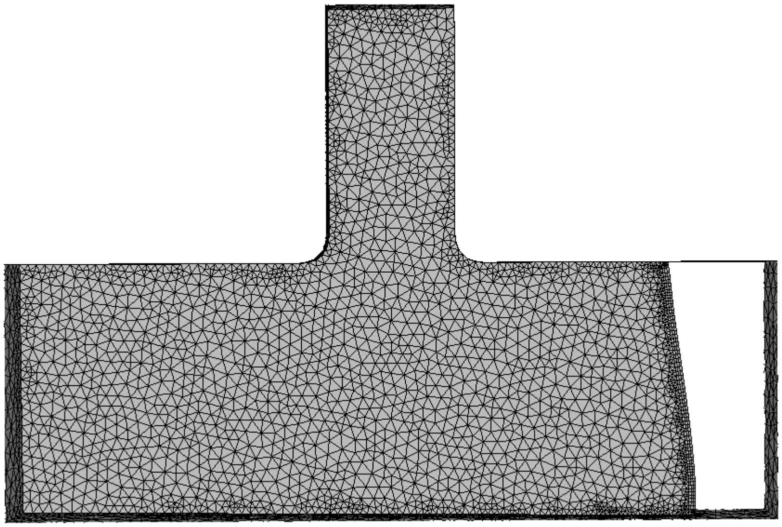
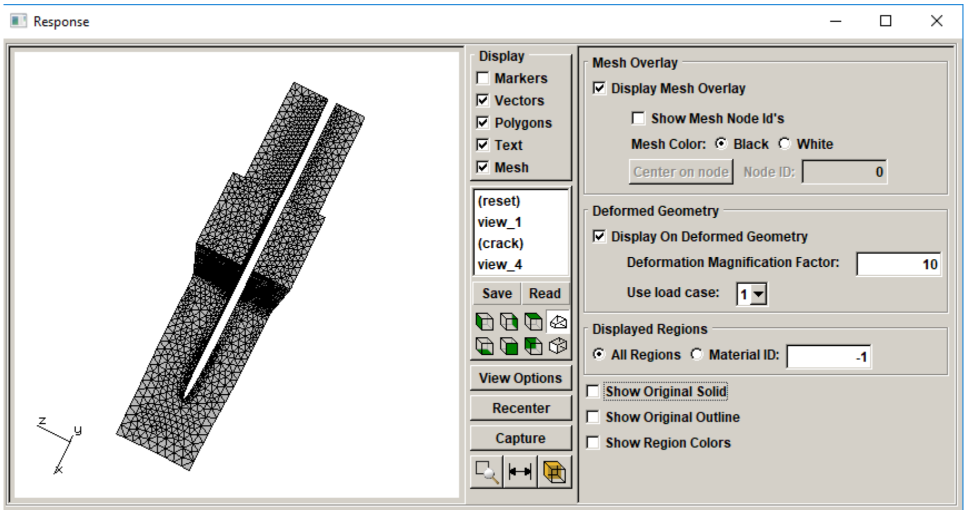


Figure 8.12 Crack shape at step #36.

The crack should always cut cleanly through the model. The FRANC3D Display - View Response menu option/dialog, Fig 8.13, can be used to quickly verify that the deformed shape is valid.



Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

Mesh Overlay
☑ Display Mesh Overlay
☐ Show Mesh Node Id's
Mesh Color: ● Black ○ White
Node ID: 0

Deformed Geometry
☑ Display On Deformed Geometry
Deformation Magnification Factor: 10
Use load case: 1

Displayed Regions
● All Regions ○ Material ID: -1

☐ Show Original Solid
☐ Show Original Outline
☐ Show Region Colors

Figure 8.13 Crack shape at step #36.

Fig 8.14 shows a case where the crack did not cleanly cut through the model; there is a node that is shared by both sides of the crack surface. FRANC3D tries to prevent this from happening, but it is possible. You can periodically inspect the deformed shape, which can be done using the View Response dialog.

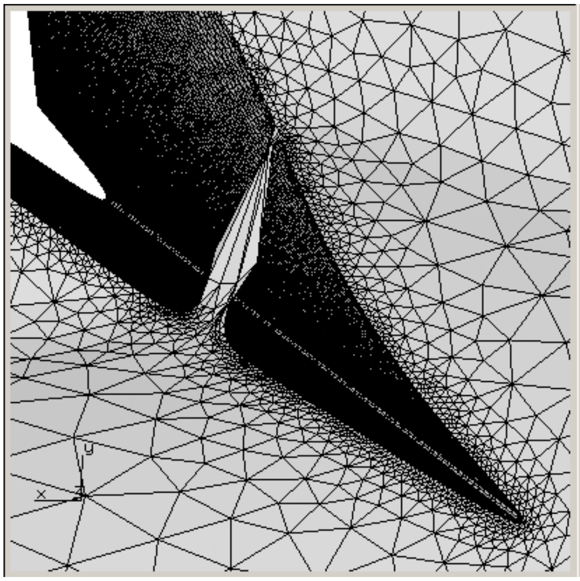


Figure 8.14 Crack shape at step #36.

8.1.1 Crack Growth around Two Corners

The same model can be used to illustrate potential problems when growing a crack front around two corners, Fig 8.15. Starting from a small penny-shape crack at the top of the T-bracket, the crack propagates down, transitions around the two 90-degree corners at the top, and then eventually transitions around the two fillet corners. If the growth is similar at both corners, the transition should be straight-forward, as seen in Fig 8.15.

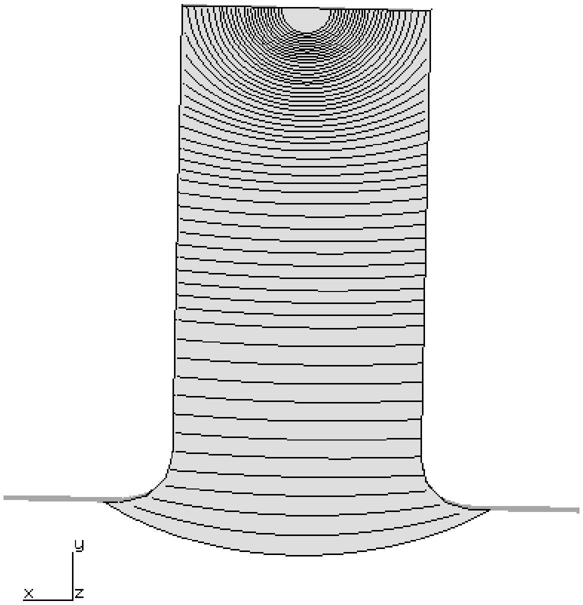
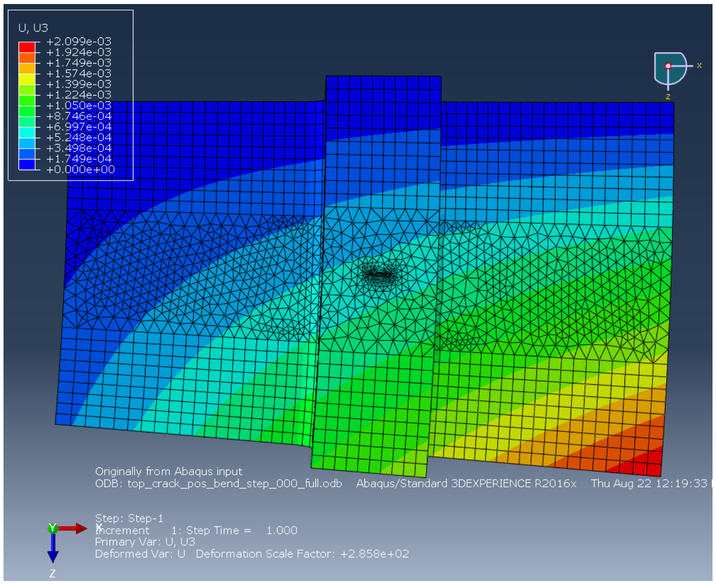


Figure 8.15 Crack front approaching model surface.

If the growth is not similar at both corners, the user might have to perform several ‘manual’ steps of crack growth to get the crack front to transition properly. To illustrate this, the model boundary conditions were altered to produce more tension on the right side compared to the left side, Fig 8.16. This produces slightly different crack extension on the left and right ends of the crack front, Fig 8.17.



```
Originally from Abaqus input
ODB: top_crack_pos_bend_step_000_full.odb    Abaqus/Standard 3DEXPERIENCE R2016x    Thu Aug 22 12:19:33

Step: Step-1
Increment      1: Step Time =    1.000
Primary Var: U, U3
Deformed Var: U   Deformation Scale Factor: +2.858e+02
```

Figure 8.16 Deformed shape for initial top-crack model.

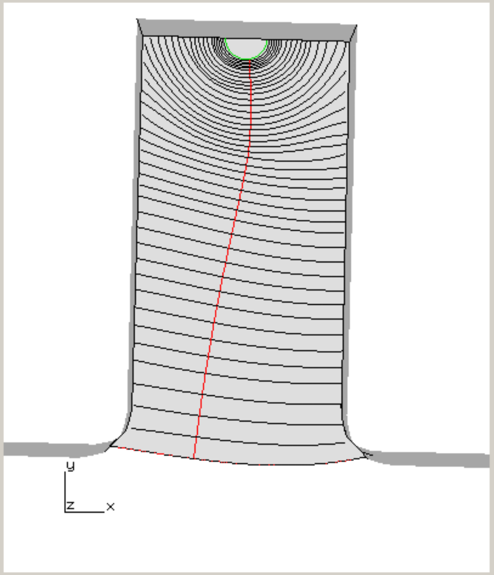
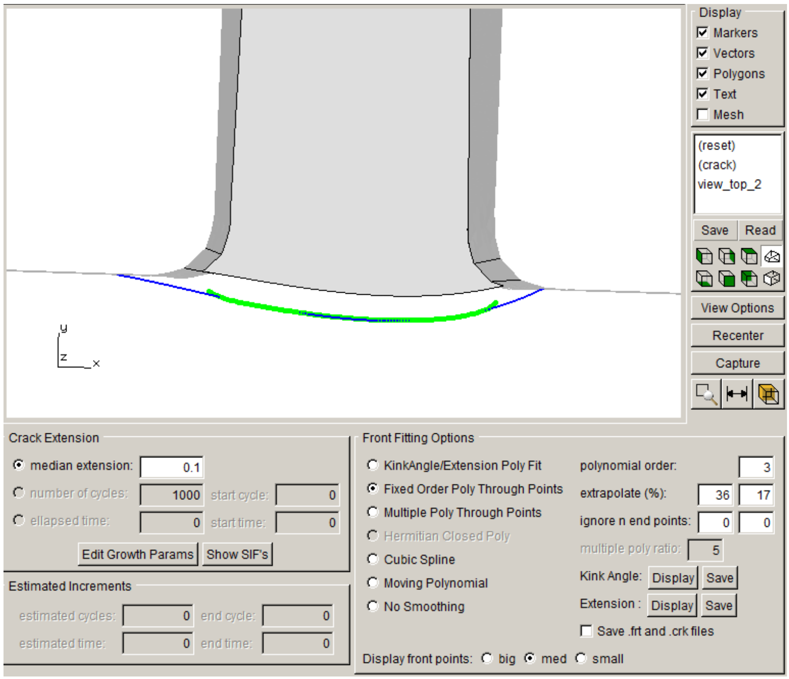


Figure 8.17 Crack fronts as the crack grows down from the top to the fillets.

The crack transition around the left and right fillets is only slightly different in this case, Fig 8.18, but it still might require user-intervention/assistance to make sure the curve fit is reasonable and that it intersects the model surface. The relative extension along the front is governed by the crack growth rate for the computed SIFs; in this case the relative extension is not significantly different.

If we change the crack growth model slightly (changing the dKth), we can get crack extension as seen in the left panel of Fig 8.19. In this case, we can simply ignore a few points at the one end and fit a polynomial curve through the remaining points. This ignores the fact that the crack front should not advance at that end – based on K < dKth; however, it will allow the simulation to continue. Partial extension of a crack front is possible, but with just one or two end points, partial extension might not be valid anyway. Partial extension is discussed in Section 9.5.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)
view_top_2

Crack Extension
● median extension: 0.1
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

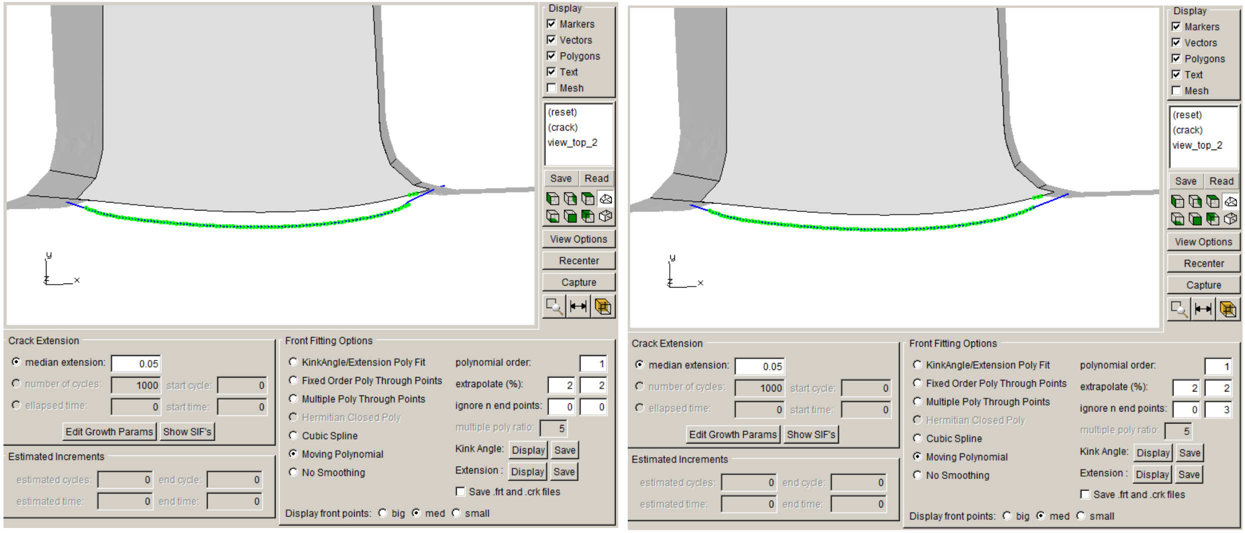
Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 3
extrapolate (%): 36 17
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 8.18 The specified extension requires significant extrapolation to ensure intersection of the curve with the model surface.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)
view_top_2

Crack Extension
● median extension: 0.05
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 2 2
ignore n end points: 0 0
multiple poly ratio: 5
☐ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)
view_top_2

Crack Extension
● median extension: 0.05
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 2 2
ignore n end points: 0 3
multiple poly ratio: 5
☐ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 8.19 Ignoring end points can assist with growth.

8.1.2 Corner Left Uncracked

A simple cube model is used to show how one can end up with an uncracked segment (see Fig 8.14) as the crack transitions around a corner. The left panel in Fig 8.20 shows a series of crack fronts starting from a half-penny surface crack. Propagation from step #7 to step #8 is shown in the right panel. While the (blue) curve-fit through the front points extends outside the model surface, it does not extend far (highlighted by the red circle).

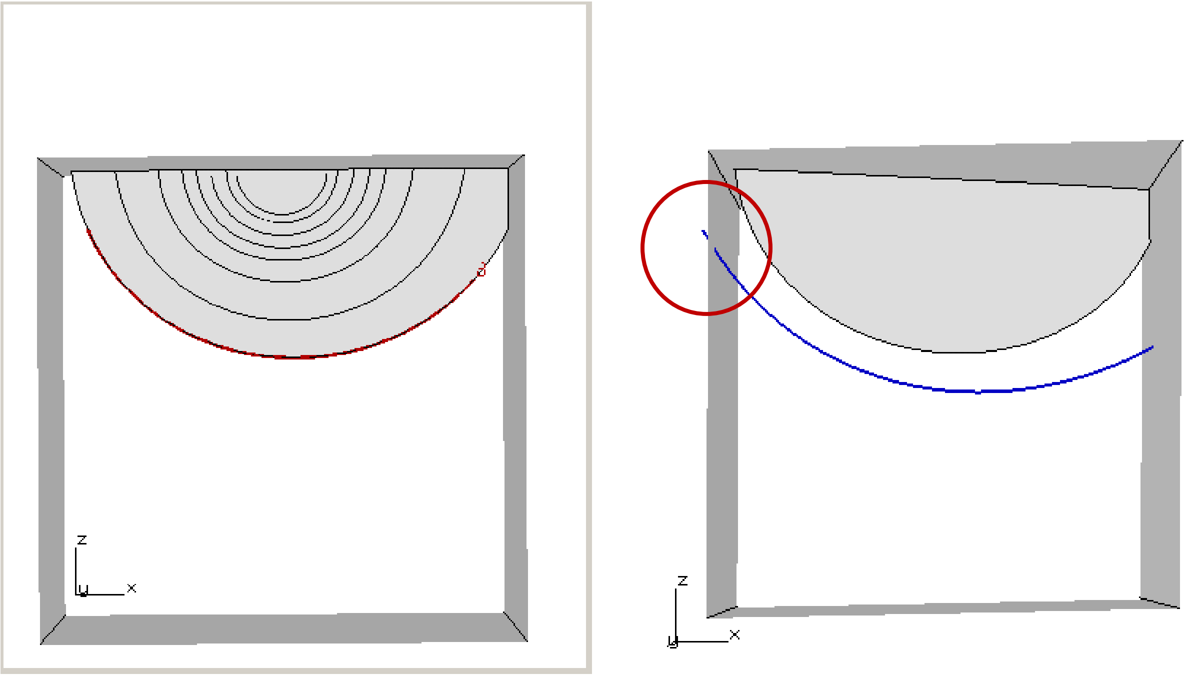


Figure 8.20 Ignoring end points can assist with growth.

FRANC3D defines new crack surface that is limited by the ends of the curve fit. The resulting crack surface is shown in Fig 8.21. The region highlighted by the red-circle in the left-side panel is not cracked. The deformed shape for the resulting FE model is shown in the right-side panel.

One can easily avoid this situation by decreasing the crack growth step and/or increasing the amount of extrapolation for the curve fit. The end of the (blue) curve should extend beyond the +Z model surface to connect with the previous crack front curve fit (and crack geometry). FRANC3D tries to avoid creating these geometric issues, but it is important that users check their results in such cases; showing the deformed shape is a straightforward way to do this.

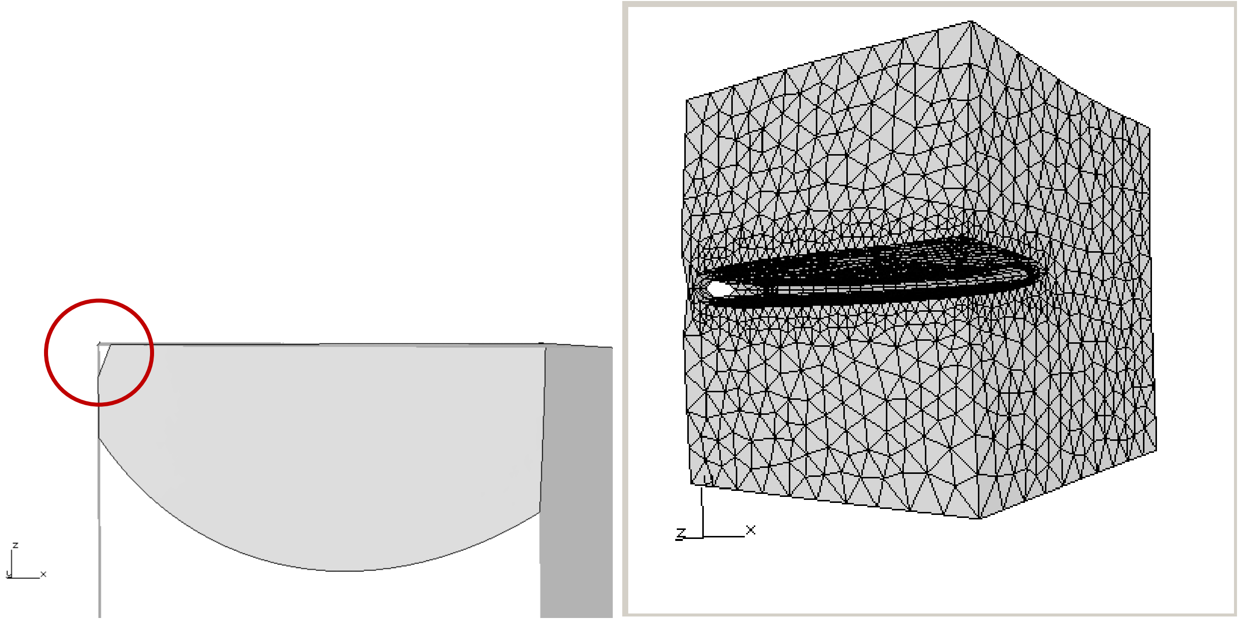


Figure 8.21 Ignoring end points can assist with growth.

8.2 Crack Growth Through Surfaces

Another common situation that can cause crack growth failure is shown in Fig 8.22, where the crack front is approaching a model surface. Errors usually occurs because the template cannot be added correctly or the volume between the template and the back model surface cannot be meshed.

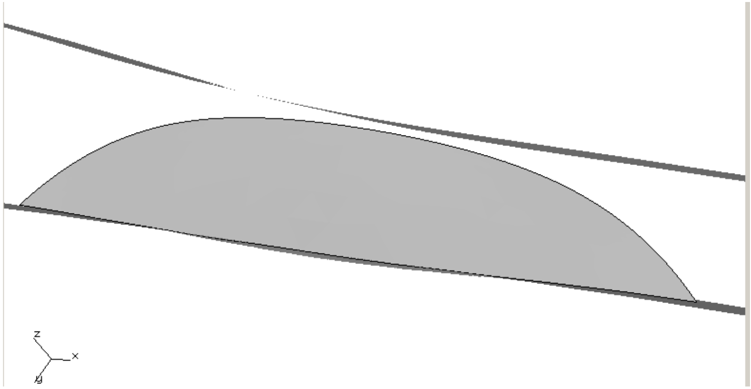
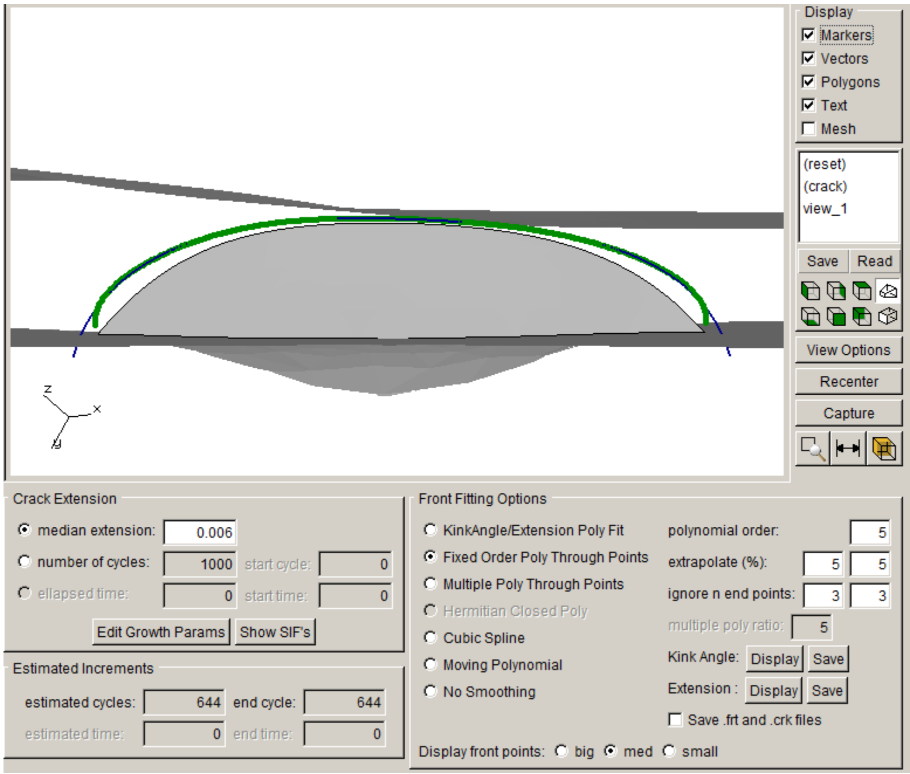


Figure 8.22 Crack front approaching model surface.

If the new crack front is close to the back surface, Fig 8.23, we can change the growth settings to get the front close while keeping the template mesh inside. We can also decrease the template radius and adjust the meshing parameters, Fig 8.24, to ensure that we get a volume mesh.

In the Meshing Parameters dialog (accessed by clicking the **Meshing Parameters** button in Fig 8.24), we turn on Do crack proximity refinement. This ensures that the back model surface gets a more refined mesh, which helps with the volume meshing between the template and the back surface. The resulting mesh on the crack surface, near the back surface, is shown in Fig 8.25.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Crack Extension
● median extension: 0.006
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

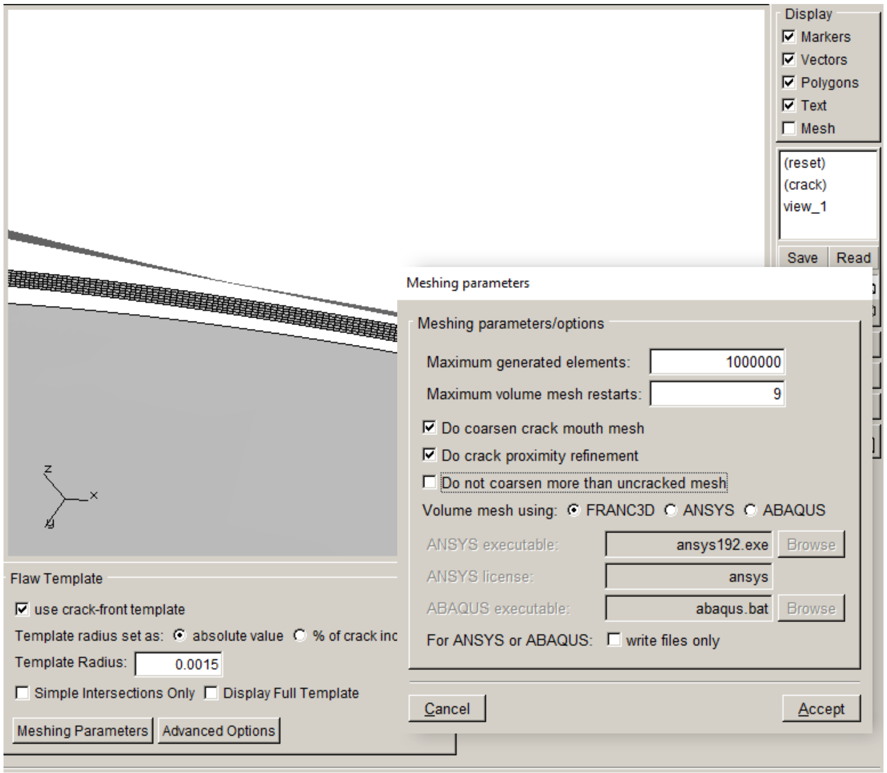
Estimated Increments
estimated cycles: 644 end cycle: 644
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 5
extrapolate (%): 5 5
ignore n end points: 3 3
multiple poly ratio: 5
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 8.23 Crack extension decreased, and fitting adjusted to keep the template inside.



Display

☑ Markers

☑ Vectors

☑ Polygons

☑ Text

□ Mesh

(reset)
(crack)
view_1

Save Read

Meshing parameters

Meshing parameters/options

Maximum generated elements: 1000000

Maximum volume mesh restarts: 9

☑ Do coarsen crack mouth mesh

☑ Do crack proximity refinement

□ Do not coarsen more than uncracked mesh

Volume mesh using: ● FRANC3D ○ ANSYS ○ ABAQUS

ANSYS executable: ansys192.exe Browse

ANSYS license: ansys

ABAQUS executable: abaqus.bat Browse

For ANSYS or ABAQUS: □ write files only

Cancel Accept

Flaw Template

☑ use crack-front template

Template radius set as: ● absolute value ○ % of crack inc

Template Radius: 0.0015

□ Simple Intersections Only □ Display Full Template

Meshing Parameters Advanced Options

Figure 8.24 Template radius decreased, and meshing parameters adjusted.

Once this crack growth step is analyzed, the crack front can be propagated through the model back surface. We can increase the extension and adjust fitting parameters if needed, Fig 8.26. It is important to check that the front fitting options push the curve-fit outside the model surface, Fig 8.27. Template radius and meshing parameters might need to be adjusted again, Fig 8.28.

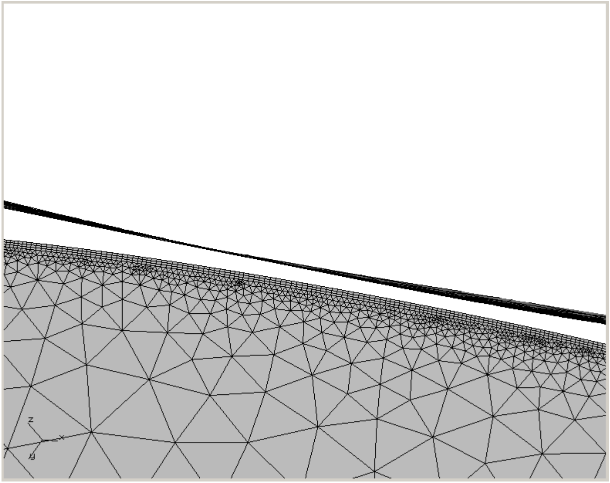
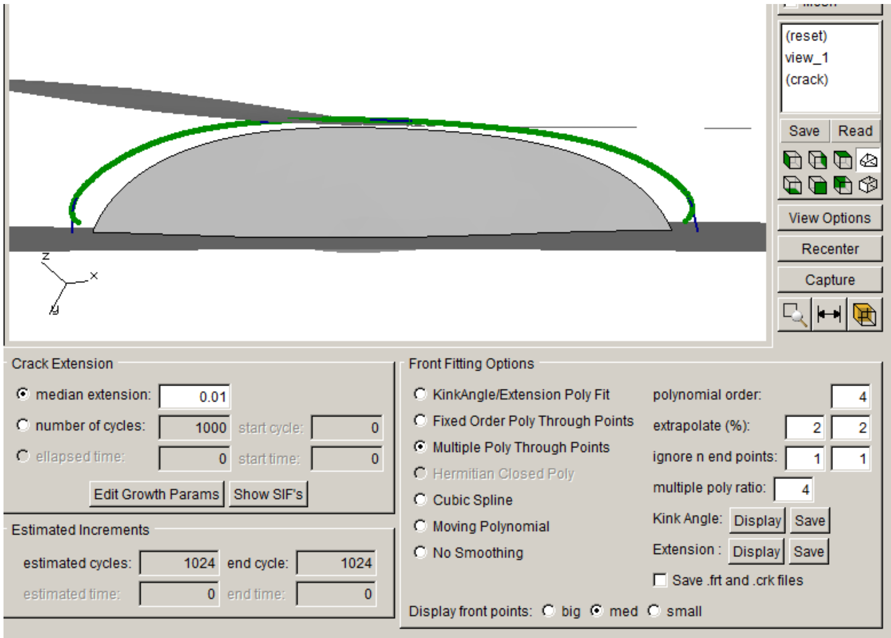


Figure 8.25 Crack surface mesh near the model back surface.



Crack Extension

● median extension: 0.01
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

estimated cycles: 1024 end cycle: 1024
estimated time: 0 end time: 0

Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
● Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 4
extrapolate (%): 2 2
ignore n end points: 1 1
multiple poly ratio: 4
Kink Angle:
Extension :
□ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 8.26 Crack growth through the model back surface.

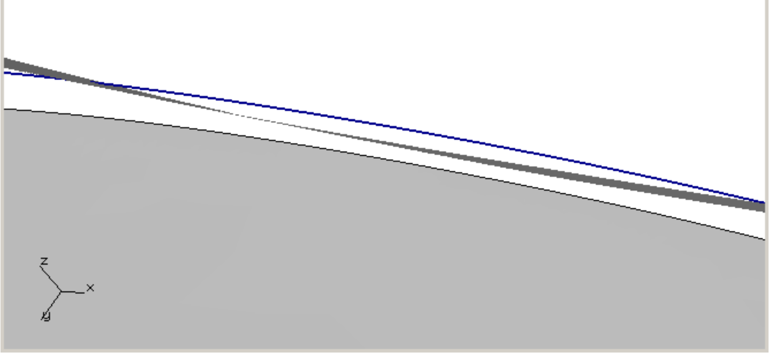
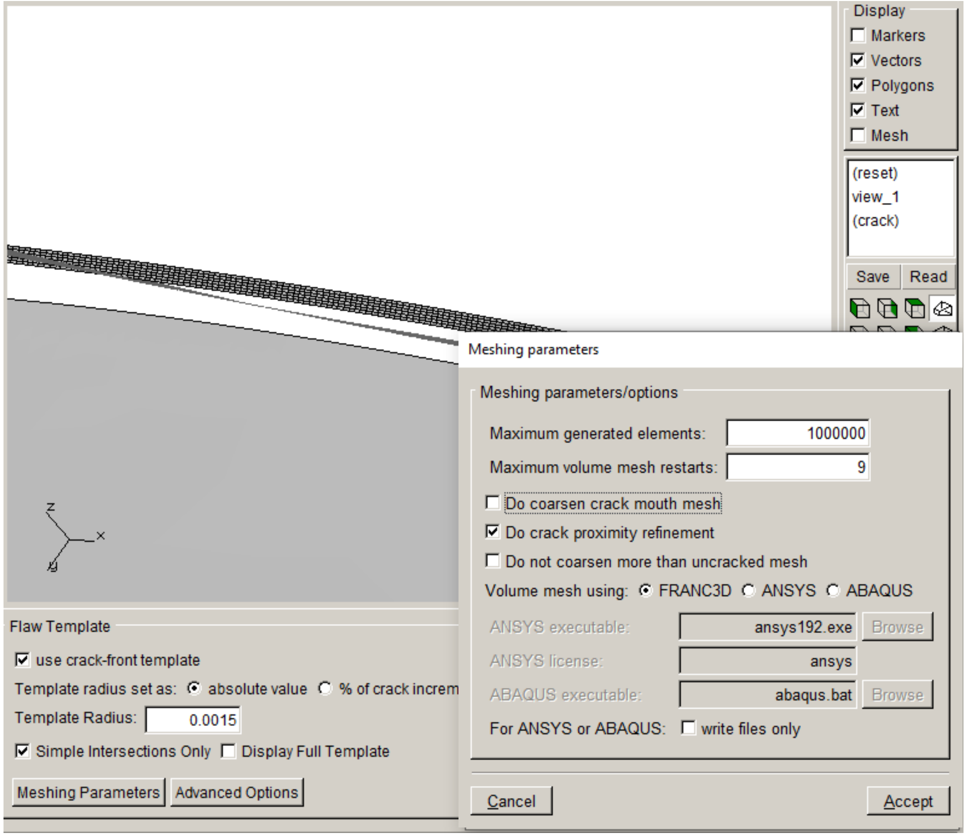


Figure 8.27 Crack front fit outside the back surface.

Fig 8.28 shows that Simple Intersections are turned on. While FRANC3D will automatically turn on simple intersections (see Section 6.1.15 of the Reference document), we can turn this on to ensure that the template ends are pulled back from the model surface.

The template radius is kept small, the Do coarsen crack… is turned off, and we increase the maximum number of volume elements (applicable for FRANC3D volume meshing only). The resulting surface mesh on the crack is relatively dense, Fig 8.29, but we need the smaller elements to fill the volume with tetrahedral elements.



Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset)
view_1
(crack)

Save | Read

Meshing parameters

Meshing parameters/options

Maximum generated elements: 1000000
Maximum volume mesh restarts: 9
□ Do coarsen crack mouth mesh
☑ Do crack proximity refinement
□ Do not coarsen more than uncracked mesh
Volume mesh using: ● FRANC3D ○ ANSYS ○ ABAQUS
ANSYS executable: ansys192.exe Browse
ANSYS license: ansys
ABAQUS executable: abaqus.bat Browse
For ANSYS or ABAQUS: □ write files only

Cancel | Accept

Flaw Template
☑ use crack-front template
Template radius set as: ● absolute value ○ % of crack increm
Template Radius: 0.0015
☑ Simple Intersections Only □ Display Full Template

Meshing Parameters | Advanced Options

Figure 8.28 Meshing parameters adjusted for the front-breaking the back surface.

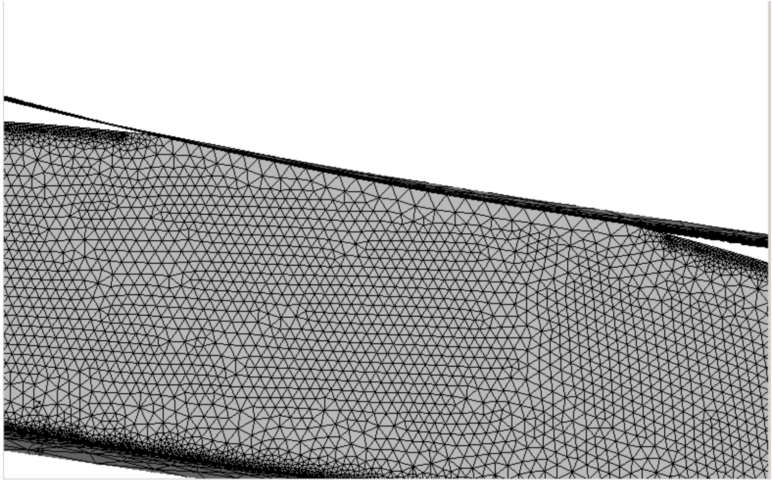
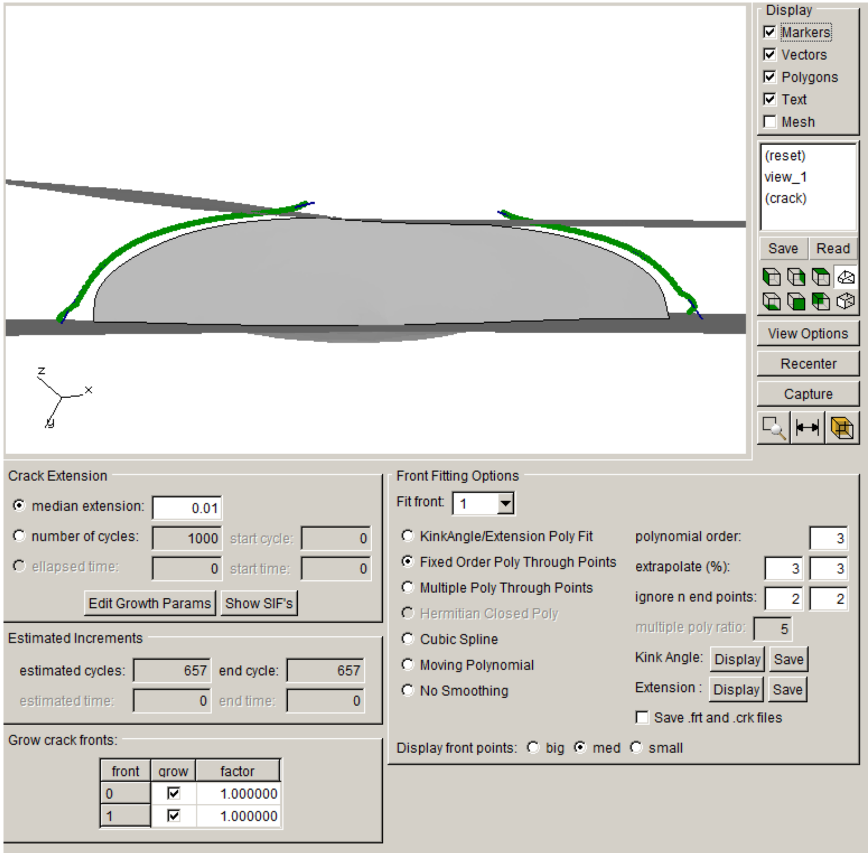


Figure 8.29 Crack surface mesh near the model back surface.

Once this step of crack growth has been analyzed, we can continue to grow the crack, but there will now be two crack fronts, Fig 8.30. Each front will have its own set of fitting parameters. We can discard end points to simplify the fit. The ends of the front should be smoothed-out as the fronts continue to advance.

Simple intersections might produce questionable results at the ends but are needed to get a valid mesh. Once the fronts have advanced, Fig 8.31, you can turn off simple intersections to get the template to extend to the model surface.



Display

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset)
view_1
(crack)

Save Read

View Options

Recenter

Capture

Crack Extension

● median extension: 0.01
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Edit Growth Params Show SIF's

Estimated Increments

estimated cycles: 657 end cycle: 657
estimated time: 0 end time: 0

Grow crack fronts:

front	grow	factor
0	☑	1.000000
1	☑	1.000000

Front Fitting Options

Fit front: 1

○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 3
extrapolate (%): 3 3
ignore n end points: 2 2
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
□ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 8.30 Two crack fronts exist after breaking through the back surface.

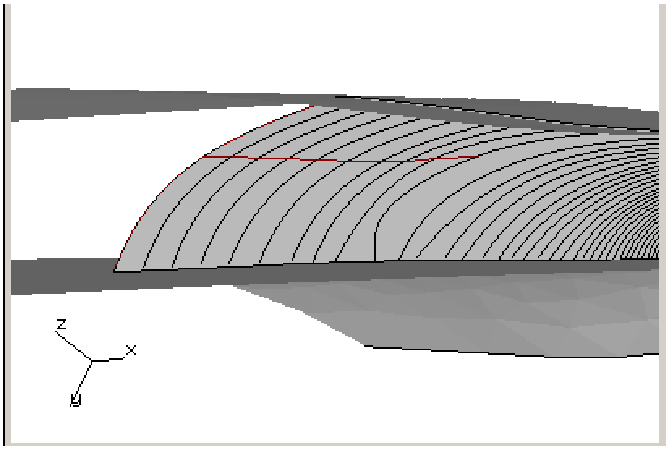


Figure 8.31 Two crack fronts continue to advance.

8.3 Crack Growth with Twisting/Turning

FRANC3D allows for out-of-plane crack-turning and twisting. A sketch of a model and crack is shown in Fig 8.32. The initial crack will turn/twist to align with the principal stress caused by the loading and constraint. FRANC3D can simulate this – the actual factory-roof crack growth is "smoothed", but the final crack shape resembles the observed crack shape.

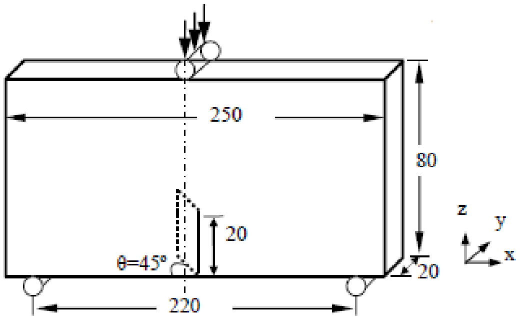


Figure 8.32 Model with a slanted crack;
from Maitireyimu et al, Journal of Solid Mech and Mat Engng, Vol 3, 2009.

Twisting/turning of the crack front might require a bit of manual intervention to get the crack to grow from its initial configuration. Fig 8.33 shows the FRANC3D default fit-curve passing through the predicted new front points (blue curve and green points). The FRANC3D default for the curve fit is a 3rd order polynomial. This fits the points but also produces a difficult geometry for the template mesh (see Sections 4.2 and 4.3). The user can simplify the geometry for this step of growth by switching to a 1st order polynomial fit, Fig 8.34.

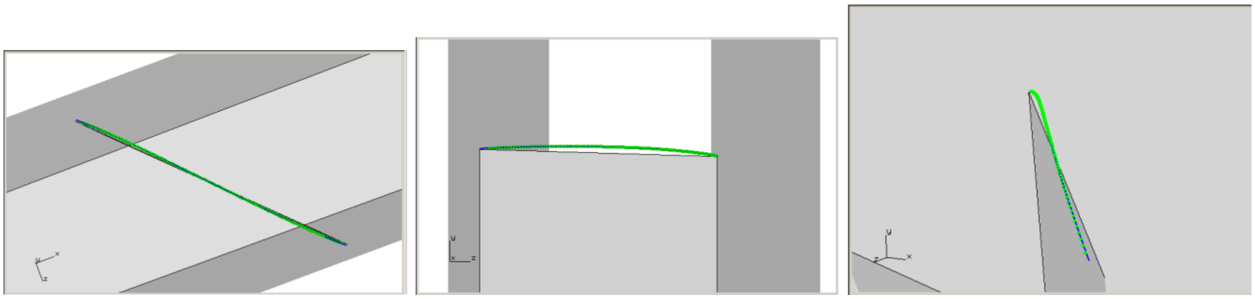
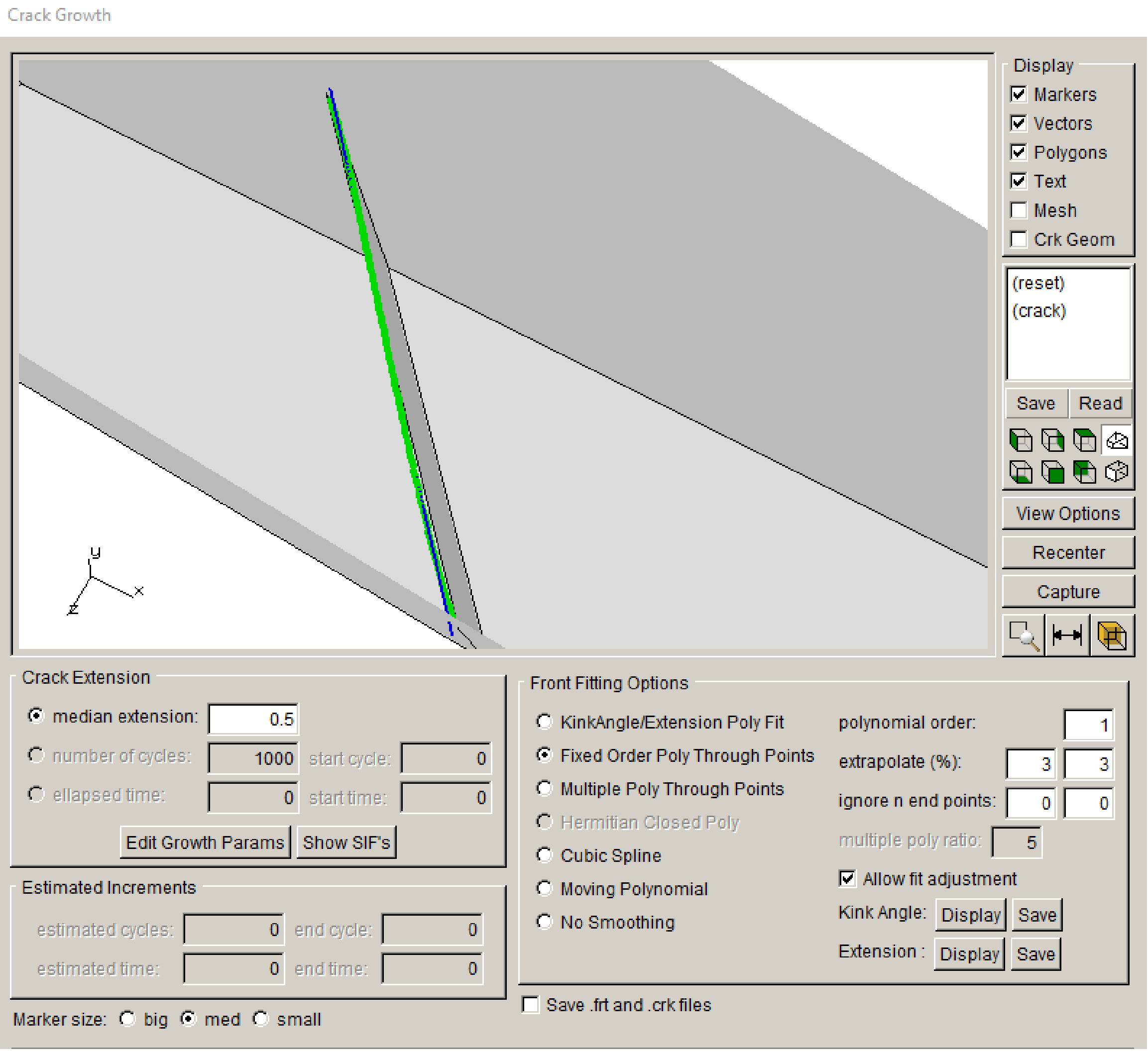


Figure 8.33 First step of predicted crack growth from the initial slanted crack.



Crack Growth

Display

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Crk Geom

(reset)
(crack)

Crack Extension

● median extension: 0.5
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Marker size: ○ big ● med ○ small

Front Fitting Options

○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 3 3
ignore n end points: 0 0
multiple poly ratio: 5
☑ Allow fit adjustment
Kink Angle: Display Save
Extension : Display Save

☐ Save .frt and .crk files

Figure 8.34 First step of growth with a 1st order polynomial fit.

This simplified curve-fit allows FRANC3D to create a template mesh at the crack front with well-shaped elements, Fig 8.35. The template should not encompass large kinks in the geometry surface. In this case, the template radius is less than the amount of growth so the template uses only the new crack surface geoemtry. If the template is larger, it will generate template elements that are highly distorted, Fig 8.36, and potentially might not allow for a FE solution.

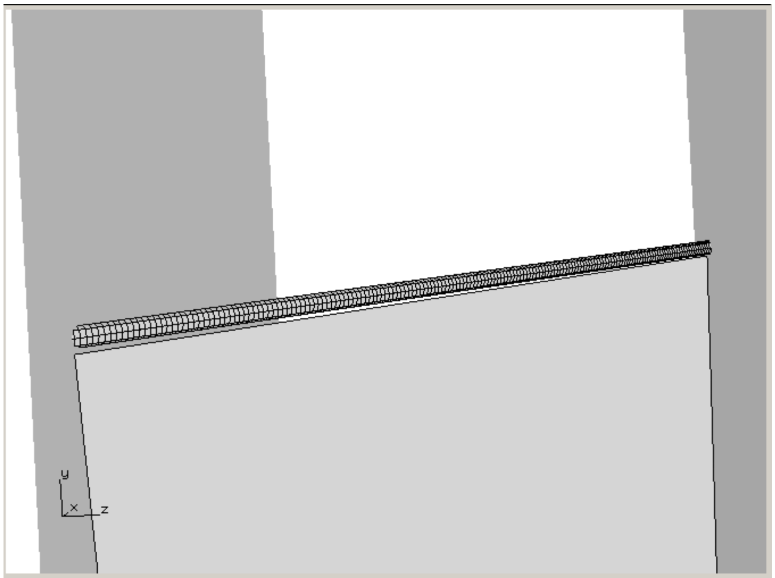


Figure 8.35 Template mesh for the first step of growth.

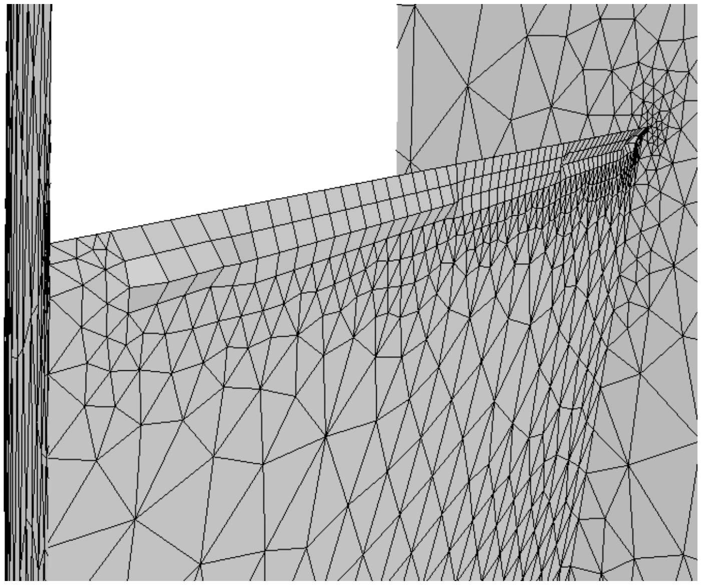


Figure 8.36 Crack template for a relatively large template radius that spans the old and new crack surface geometry.

After the first few steps of growth, the shape of the crack surface and the new surface growth do not include any large kinks, and the default 3rd order polynomial can be used to simulate the rest of the growth steps. Fig 8.37 shows the crack at 33 mm of growth.

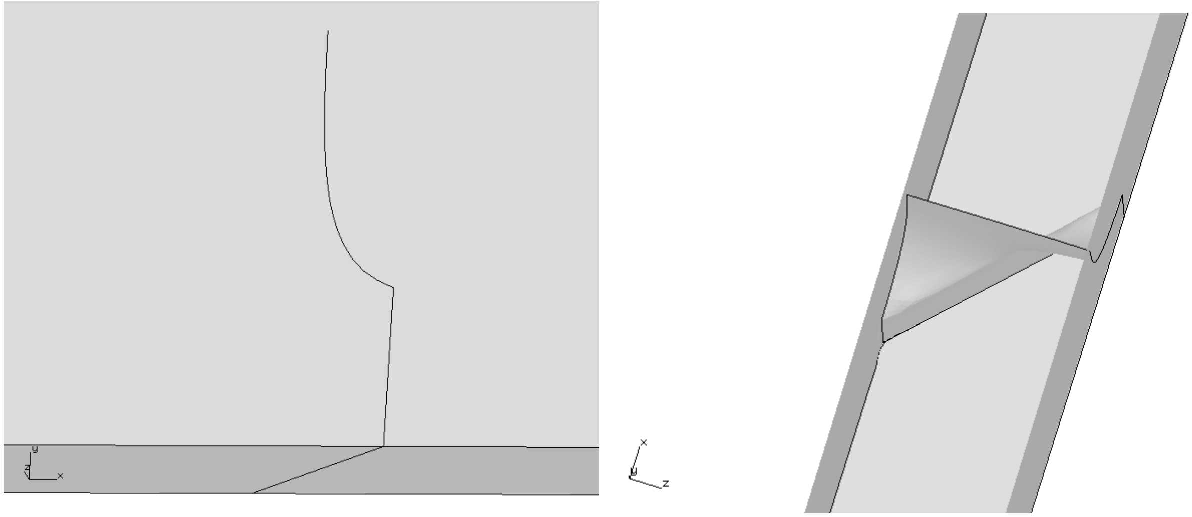


Figure 8.37 Crack after 33 mm of growth.

8.4 Crack Front Merging

It is possible to merge semi-coplanar crack fronts in FRANC3D; the grow/merge wizard is described in Section 6.7 of the Reference document.. There are limitations to this capability. The first is that the crack surfaces should be mostly coplanar. Other limitations will be described in more detail here.

8.4.1 Cannot Merge Initial Cracks

If a user starts by inserting two penny-shaped coplanar surface cracks, Fig 8.38, these cannot be merged immediately.

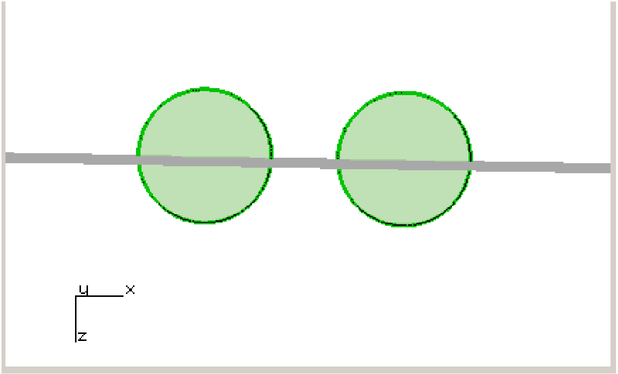
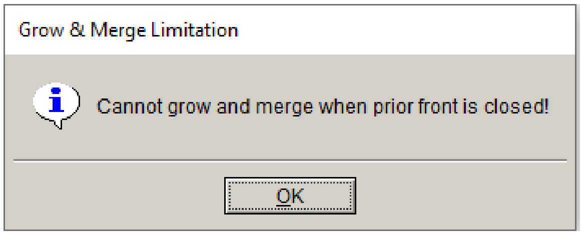


Figure 8.38 Two penny-shape library flaws inserted as half-penny surface cracks.

Once the two cracks shown in Fig 8.38 are inserted, meshed and analyzed, user might be tempted to use the Grow/Merge option to merge the fronts into a single front. Currently, this is not possible; FRANC3D will display the warning shown in Fig 8.39. Even though the cracks in the mesh are seen as half-penny cracks, the underlying geoemtry is as shown in Fig 8.38, where the crack fronts are "closed".


Grow & Merge Limitation

Cannot grow and merge when prior front is closed!

Figure 8.39 Grow/Merge limitation warning.

To get these two crack fronts to merge, one must first grow the cracks at least one step. This will produce an "open" crack front (see Fig 4.7 which shows two steps of growth added to the original crack geometry). Fig 8.40 shows the geometry for these two cracks after one step of growth, and Fig 8.41 shows the merged crack front curve for the specified amount of crack extension.

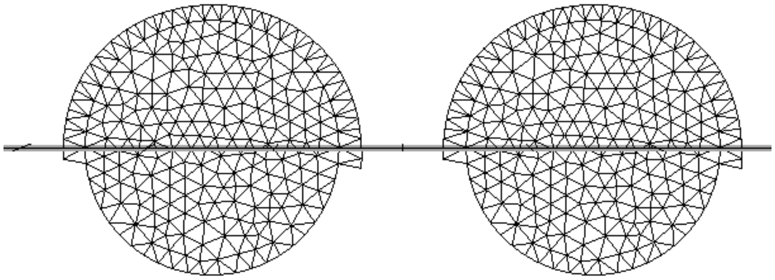


Figure 8.40 Two penny-shape library flaws after one step of growth.

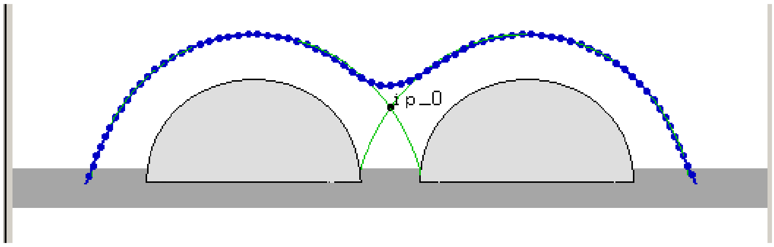


Figure 8.41 Two penny-shape library flaws after one step of growth.

8.4.2 Cannot Grow One Crack While Merging Other Cracks

If a user has multiple cracks in a model, Fig 8.42, and two of the cracks will be merged, the process of growing and merging currently requires a couple of steps.

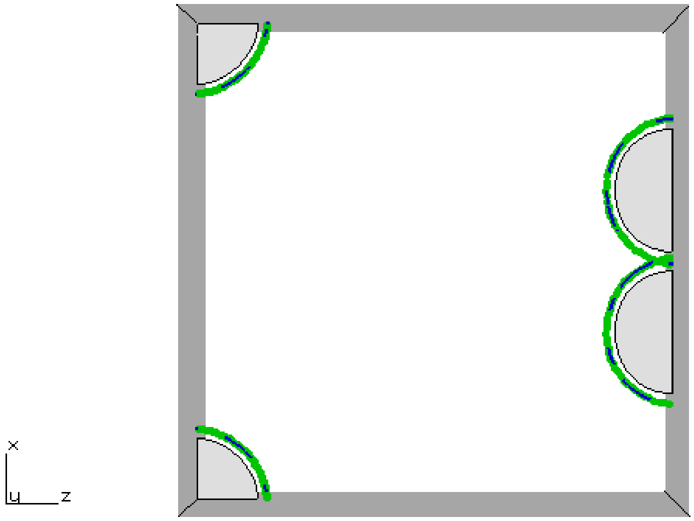
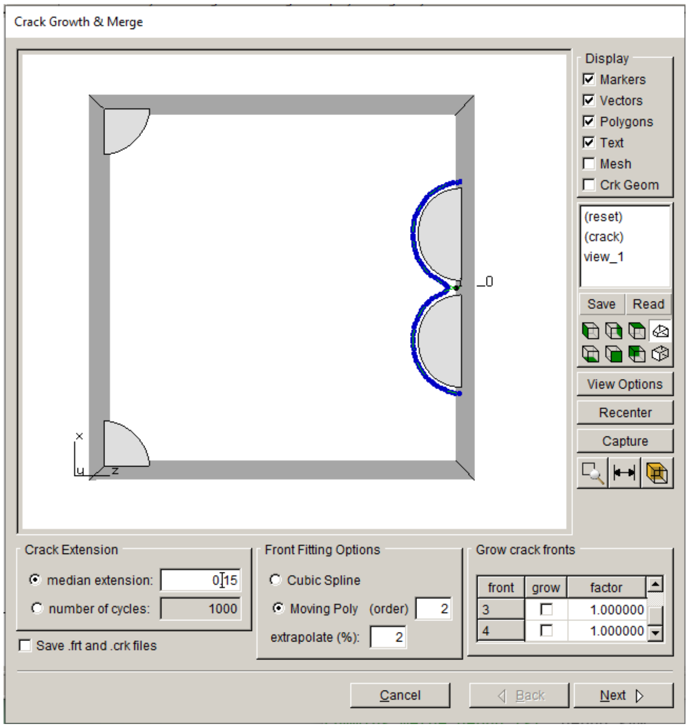


Figure 8.42 Four cracks with two cracks about to merge.

The first step is to use the Grow/Merge dialog to grow and merge the crack fronts for the two cracks that are merging, Fig 8.43. The growth is turned off for crack fronts 3 and 4, while the fronts for 1 and 2 are being merged. This configuration is analyzed, and then crack fronts 3 and 4 are propagated while the merged crack front is held.

Once the merged crack step is analyzed, one can use the regular Crack Growth dialog to grow the other two crack fronts while turning off growth of the merged front, Fig 8.44. This will allow the non-merged crack fronts to catch up. Once this configuration is analyzed, all crack fronts can be propagated normally, Fig 8.45.

This process is not ideal, but it should produce reasonable results for reasonable crack growth increments. The process of growing and merging will be revised in the future to allow growth and merging in a single step.



Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh □ Crk Geom

Crack Extension

● median extension: 0.15

○ number of cycles: 1000

□ Save .frt and .crk files

Front Fitting Options

○ Cubic Spline

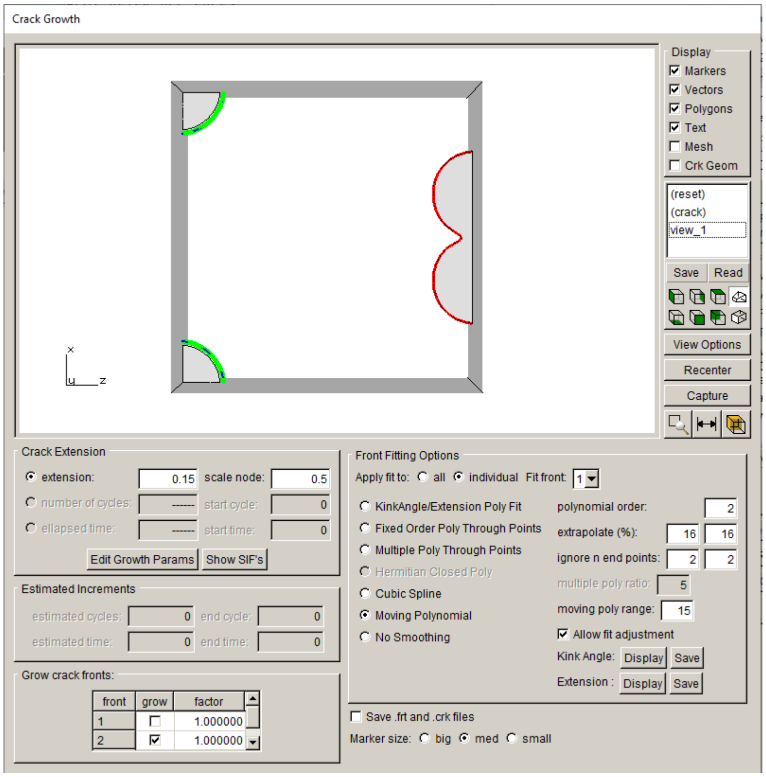
● Moving Poly (order) 2

extrapolate (%): 2

Grow crack fronts

front	grow	factor
3	□	1.000000
4	□	1.000000

Figure 8.43 Crack fronts 1 and 2 are merged.



Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh □ Crk Geom

Crack Extension

● extension: 0.15 scale node: 0.5

○ number of cycles: ------ start cycle: 0

○ ellapsed time: ------ start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0

estimated time: 0 end time: 0

Grow crack fronts:

front	grow	factor
1	□	1.000000
2	☑	1.000000

Front Fitting Options

Apply fit to: ○ all ● individual Fit front: 1

○ KinkAngle/Extension Poly Fit

○ Fixed Order Poly Through Points

○ Multiple Poly Through Points

○ Hermitian Closed Poly

○ Cubic Spline

● Moving Polynomial

○ No Smoothing

polynomial order: 2

extrapolate (%): 16 16

ignore n end points: 2 2

multiple poly ratio: 5

moving poly range: 15

☑ Allow fit adjustment

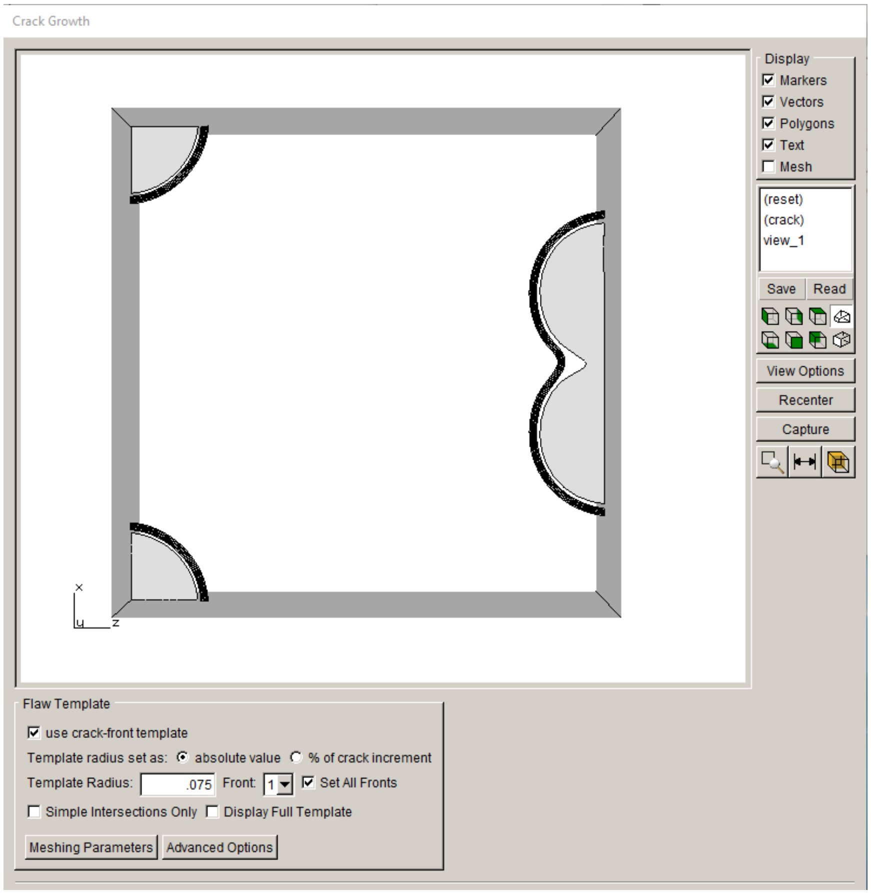
Kink Angle: Display Save

Extension : Display Save

□ Save .frt and .crk files

Marker size: ○ big ● med ○ small

Figure 8.44 Growth for merged crack front turned off.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset)
(crack)
view_1

Flaw Template
☑ use crack-front template
Template radius set as: ● absolute value ○ % of crack increment
Template Radius: .075 Front: 1 ☑ Set All Fronts
□ Simple Intersections Only □ Display Full Template

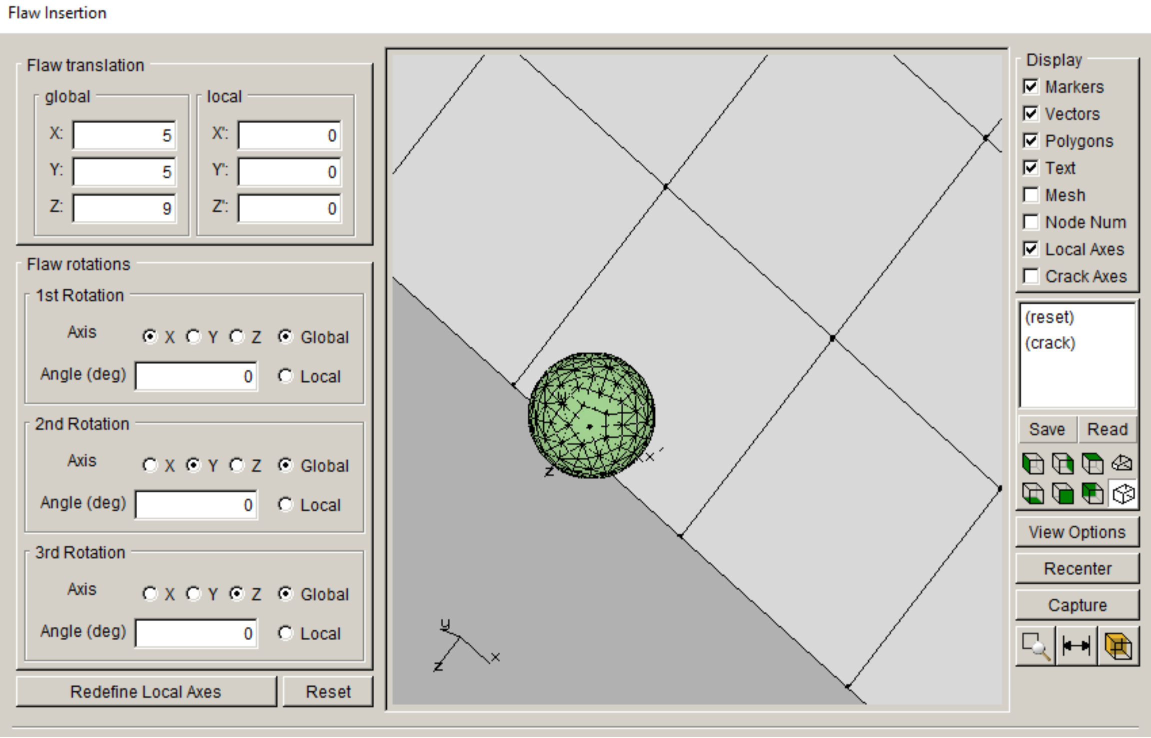
Figure 8.45 Normal growth of all crack fronts.

8.4.3 Merging the Ends of a Single Crack Front

If a crack starts from the surface of a void (or cylindrical hole), the crack might wrap around the void such that the ends of the crack front meet on the opposite side. If the crack front ends are mostly coplanar, the ends can be merged. This requires some user-control in the GUI, which is described here.

An interior void is first inserted into a cube, Fig 8.46; this is the Tutorial #1 ABAQUS cube model. We import the full cube model into FRANC3D for simplicity and retain all the boundary conditions. The void is defined as a sphere, with a diameter of 0.5 units, and is located at coordinates (5,5,9). Once the void is inserted, the files are saved, which provides an ABAQUS *.inp* file that can then be re-imported into FRANC3D.

The *void.inp* file can be imported and divided in FRANC3D so that we do not have to remesh the full cube while inserting and growing the crack. A crack is inserted into the surface of the void, Fig 8.47. A circular crack with diameter of 0.2 units is inserted at coordinates (5,5,9.25). The resulting mesh is shown in Fig 8.48.



Flaw Insertion

Flaw translation

global

X: 5

Y: 5

Z: 9

local

X': 0

Y': 0

Z': 0

Flaw rotations

1st Rotation

Axis ● X ○ Y ○ Z ● Global

Angle (deg) 0 ○ Local

2nd Rotation

Axis ○ X ● Y ○ Z ● Global

Angle (deg) 0 ○ Local

3rd Rotation

Axis ○ X ○ Y ● Z ● Global

Angle (deg) 0 ○ Local

Display

☑ Markers

☑ Vectors

☑ Polygons

☑ Text

□ Mesh

□ Node Num

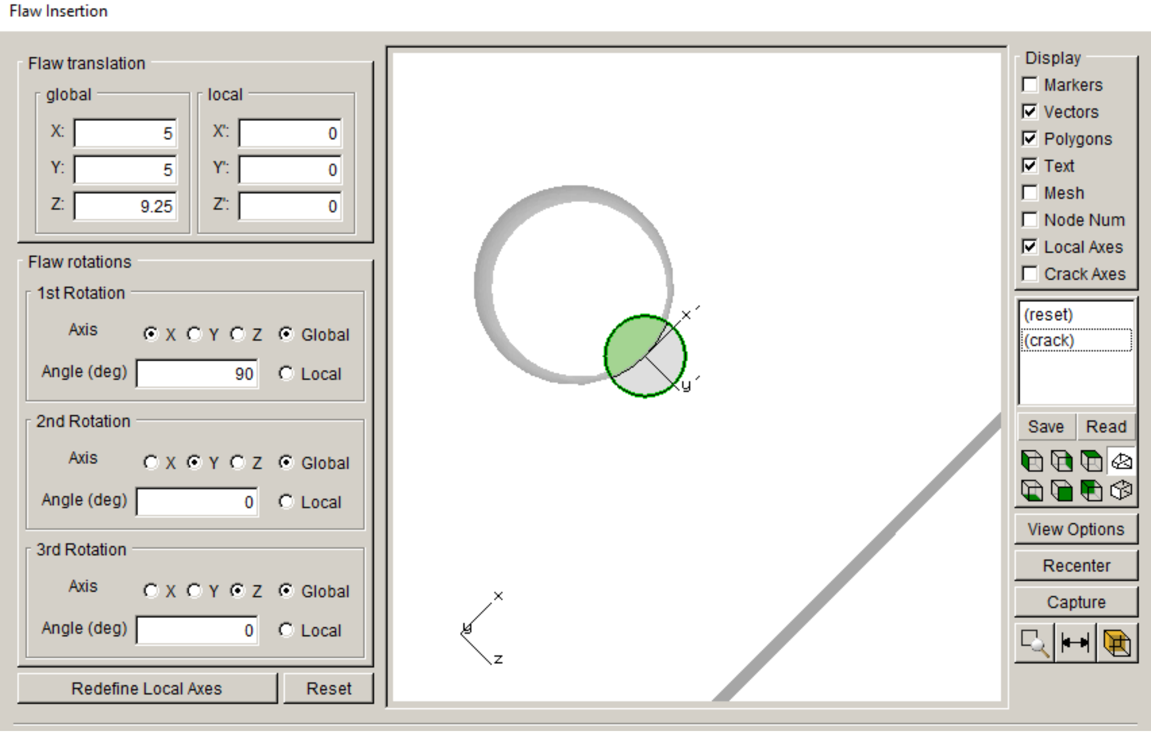
☑ Local Axes

□ Crack Axes

(reset)

(crack)

Figure 8.46 Interior void inserted into a cube.



Flaw Insertion

Flaw translation

global

X: 5

Y: 5

Z: 9.25

local

X': 0

Y': 0

Z': 0

Flaw rotations

1st Rotation

Axis ● X ○ Y ○ Z ● Global

Angle (deg) 90 ○ Local

2nd Rotation

Axis ○ X ● Y ○ Z ● Global

Angle (deg) 0 ○ Local

3rd Rotation

Axis ○ X ○ Y ● Z ● Global

Angle (deg) 0 ○ Local

Display

□ Markers

☑ Vectors

☑ Polygons

☑ Text

□ Mesh

□ Node Num

☑ Local Axes

□ Crack Axes

(reset)

(crack)

Figure 8.47 Surface penny crack inserted into a void.

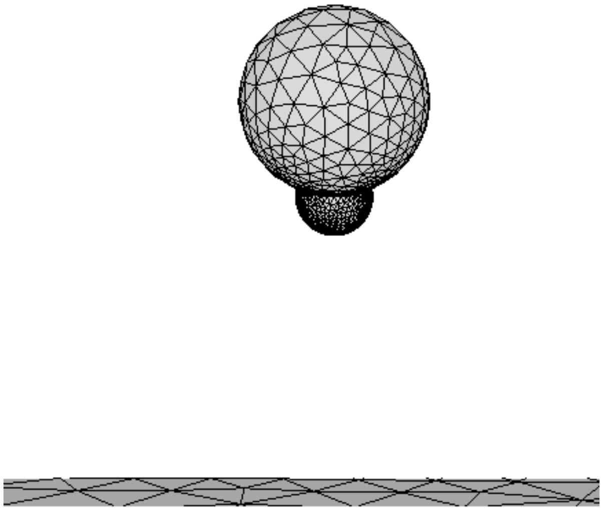


Figure 8.48 Surface mesh for a penny crack inserted into a void.

The crack propagates around the void, staying approximately planar, as the cube is subjected to uniform tension. When the ends of the crack front approach each other on the opposite side of the void, Fig 8.49, the user must use the **Grow/Merge Cracks** menu option to merge the front ends together. Fig 8.50 shows the Grow/Merge dialog as the ends are merged; note that a relatively large growth step is used to prevent a “sharp” kink in the merged front template.

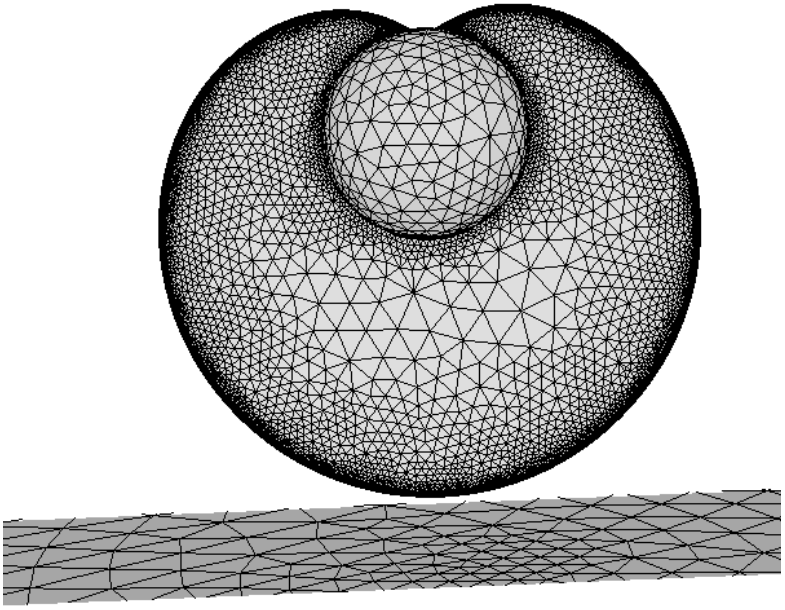
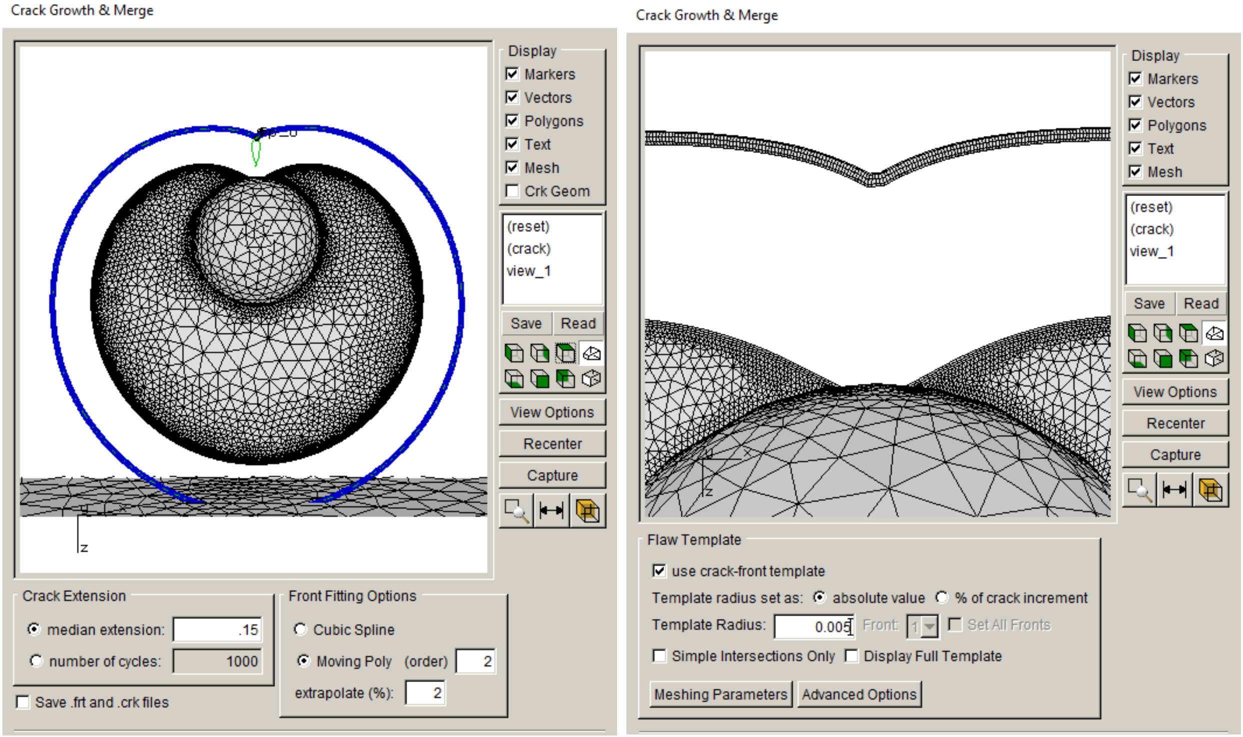


Figure 8.49 Ends of the crack front are ready to merge.



☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh
□ Crk Geom

Crack Extension
● median extension: .15
○ number of cycles: 1000
□ Save .frt and .crk files

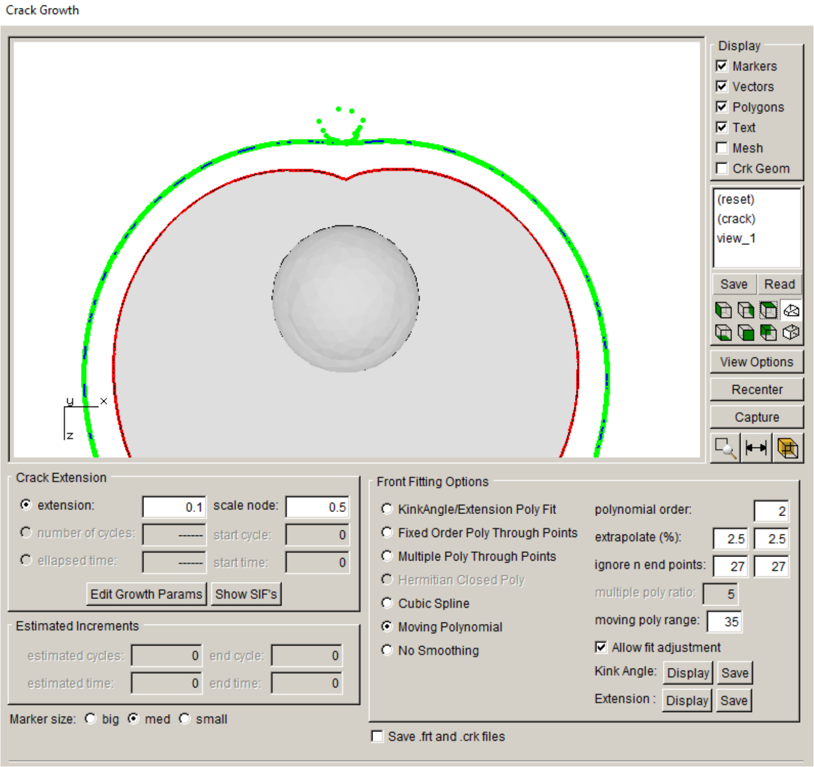
Front Fitting Options
○ Cubic Spline
● Moving Poly (order) 2
extrapolate (%): 2

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

Flaw Template
☑ use crack-front template
Template radius set as: ● absolute value ○ % of crack increment
Template Radius: 0.005 Front: 1 □ Set All Fronts
□ Simple Intersections Only □ Display Full Template

Figure 8.50 Grow/Merge dialog.

Once the front has been merged (and remeshed) successfully, it might become a “closed” front, usually with a concave portion that can lead to overlaping new front points for the subsequent step of growth, Fig 8.51. This subsequent step of growth is done using the standard **Grow Crack** menu option. Fig 8.51 shows the fitting options that will discard the overlapping new front points. The resulting front is less concave, Fig 8.52, but could still lead to overlapping new front points, so another manual step of growth might be required to control the front fitting. Once the front has reached a fully-convex shape, automated growth should be possible.



Display

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Crk Geom

Crack Extension

● extension: 0.1 scale node: 0.5
○ number of cycles: ------ start cycle: 0
○ ellapsed time: ------ start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Marker size: ○ big ● med ○ small

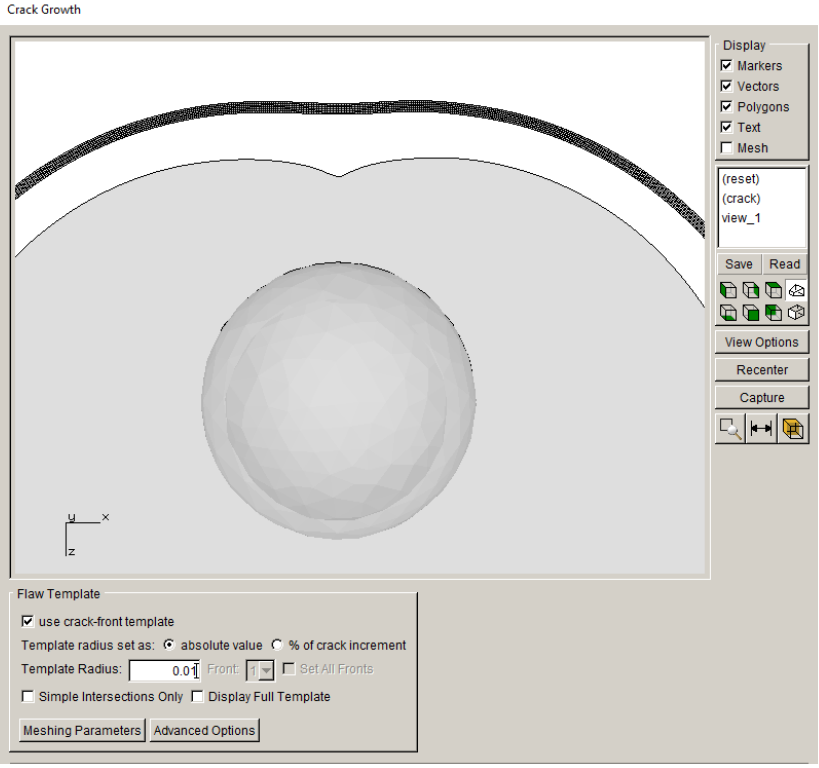
Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 2
extrapolate (%): 2.5 2.5
ignore n end points: 27 27
multiple poly ratio: 5
moving poly range: 35
☑ Allow fit adjustment
Kink Angle: Display Save
Extension : Display Save

□ Save .frt and .crk files

Figure 8.51 Grow Cracks - fitting dialog.



Display

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Flaw Template

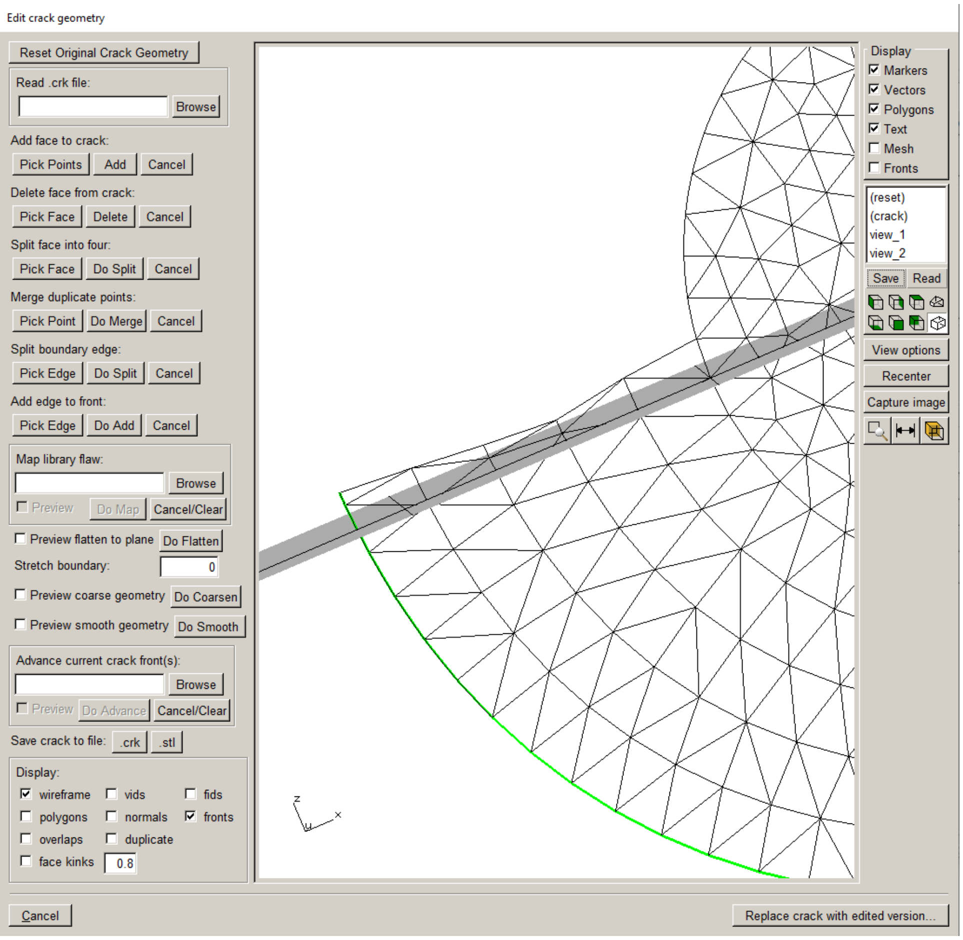
☑ use crack-front template
Template radius set as: ● absolute value ○ % of crack increment
Template Radius: 0.01 Front: 1 □ Set All Fronts
□ Simple Intersections Only □ Display Full Template

Figure 8.52 Grow Cracks - template dialog.

8.5 Edit Crack Geometry

The Edit Crack Geometry dialog allows you to edit the crack geometry. The edited geometry can be saved to a *.crk* file, or you can replace the current model crack geometry with the edited geometry. The latter option allows you fix issues in the geometry prior to growing the crack.

For example, a surface penny-crack in a cube (Tutorial #1) is shown in the Edit Crack dialog, Fig 8.53. The crack front is identified by the green curve; this is turned on by checking the fronts box at the bottom left under Display. The crack surface normals should be oriented consistently, and there should not be any edge overlaps or duplicate points. The Display check boxes allow you to check for these conditions.



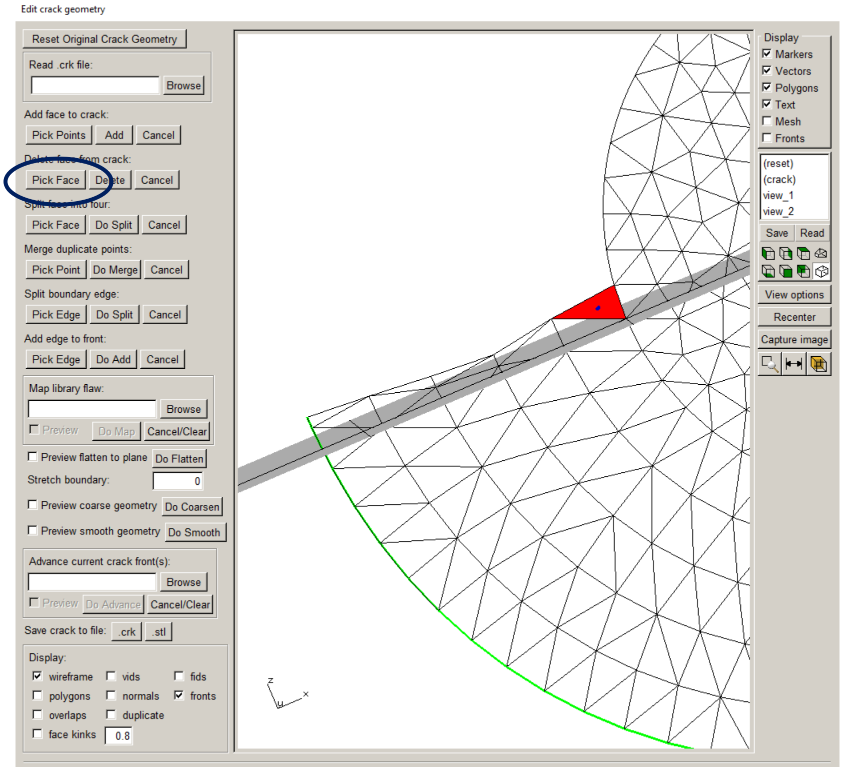
☐ Preview flatten to plane
Stretch boundary: 0
☐ Preview coarse geometry
☐ Preview smooth geometry

Display:
☑ wireframe ☐ vids ☐ fids
☐ polygons ☐ normals ☑ fronts
☐ overlaps ☐ duplicate
☐ face kinks 0.8

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Fronts

Figure 8.53 Edit Crack Geometry dialog.

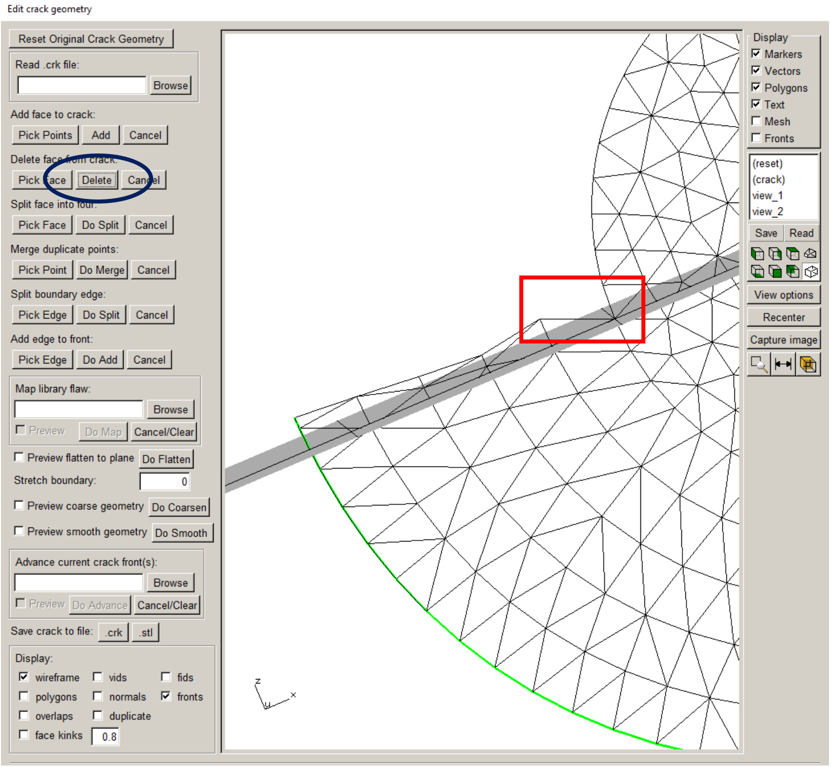
The dialog allows you to add or delete a geometry patch to the crack. To delete a triangle from the boundary of the crack, click **Pick Face** and then pick a point on a boundary face, Fig 8.54a. If the edge is a valid boundary edge, the adjacent face is colored red. Click the **Delete** button to finish removing the triangle, Fig 8.54b. You can repeat this process as many times as needed to remove overlapping (or badly oriented) triangles.



☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

☑ wireframe □ vids □ fids
□ polygons □ normals ☑ fronts
□ overlaps □ duplicate
□ face kinks 0.8

Figure 8.54a Delete face from crack.



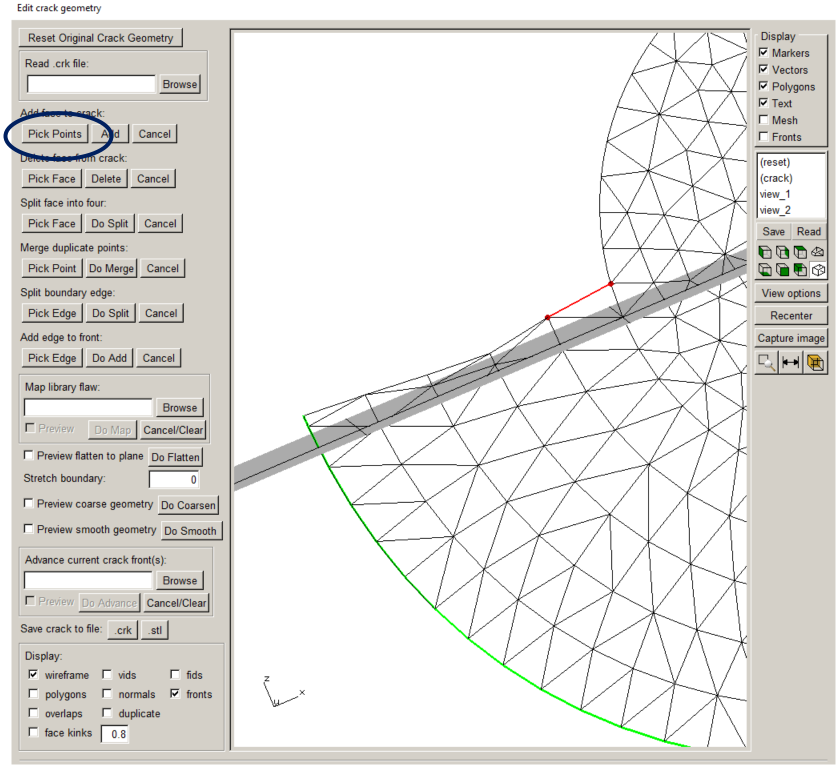
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

☑ wireframe □ vids □ fids
□ polygons □ normals ☑ fronts
□ overlaps □ duplicate
□ face kinks 0.8

Figure 8.54b Face is deleted.

The Add face to crack allows you to pick two points that already exist on the outer boundary of the crack to create a new triangle. For example, Fig 8.55a shows two points that were picked after clicking the **Pick Points** button. If the points are valid, an edge is shown between them. Click the **Add** button to finish adding the face, Fig 8.55b; the new face should use the existing edge geometry and be oriented with a normal direction that is consistent with the rest of the crack surface.

Once the crack has been edited, the **Replace crack with edited version** button can be used to replace the crack geometry in the current model. This will not affect the current mesh, but the new geometry will be used when creating the propagated crack geometry.

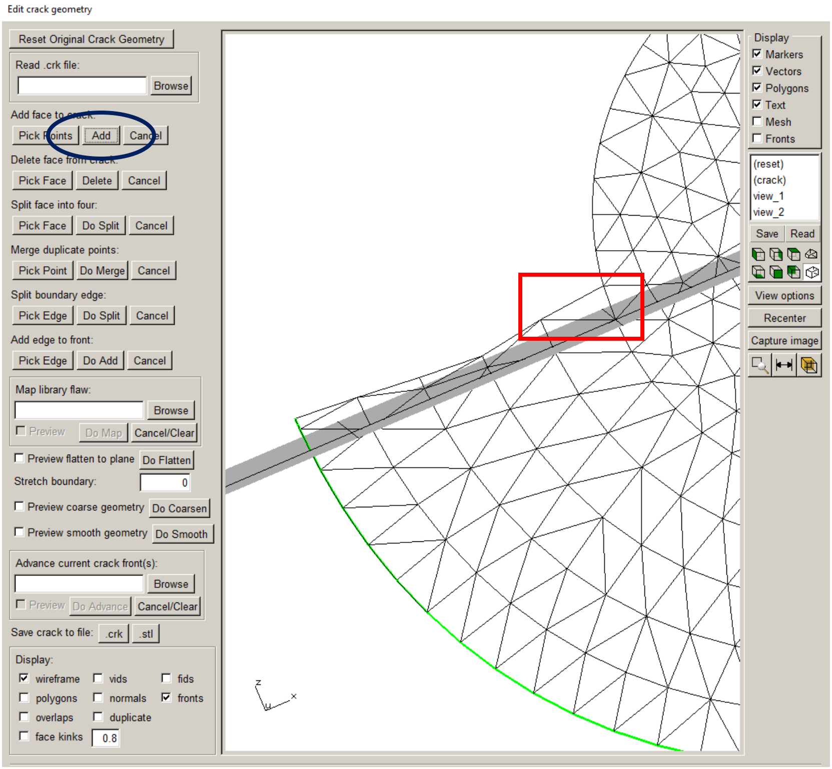


☐ Preview flatten to plane
Stretch boundary: 0
☐ Preview coarse geometry
☐ Preview smooth geometry

Display:
☑ wireframe ☐ vids ☐ fids
☐ polygons ☐ normals ☑ fronts
☐ overlaps ☐ duplicate
☐ face kinks 0.8

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Fronts

Figure 8.55a Pick two points to define a new boundary edge.



□ Preview flatten to plane
Stretch boundary: 0
□ Preview coarse geometry
□ Preview smooth geometry

Display:
☑ wireframe □ vids □ fids
□ polygons □ normals ☑ fronts
□ overlaps □ duplicate
□ face kinks 0.8

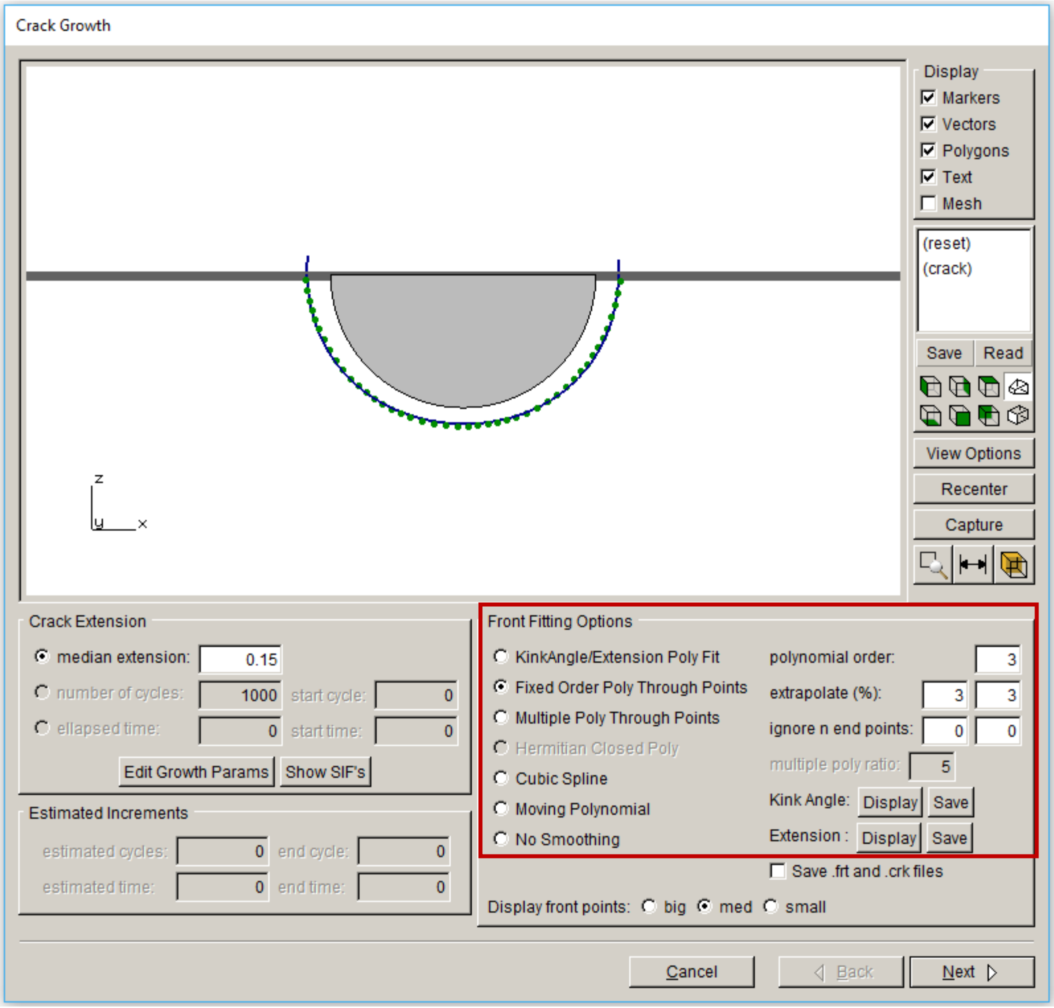
Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

Figure 8.55b Face is added.

9. Crack Front Fitting

FRANC3D has options for fitting the predicted new crack front points, Fig 9.1. There are two reasons to fit a curve through the new front points: 1) smooth out numerical noise and 2) extrapolate the ends outside of the model surface.

The default fit is a 3rd-order polynomial with extrapolation set to 2-3%. This gives a reasonable fit in many cases, but it is not always appropriate. Section 6.5.5.2: of the Reference document describes each fitting option. Additional examples and guidelines are provided here.


Crack Growth

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)

Crack Extension
● median extension: 0.15
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 3
extrapolate (%): 3 3
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save

☐ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 9.1 Front fitting options.

9.1 Edge Crack in a Plate – When a 3rd Order Single Polynomial Fails

In most cases, the default fitting works well. However, there are cases where the 3rd order polynomial fitting fails for no obvious reason. For example, Fig 9.2 shows an edge crack in a plate. The front has been propagated for 18 steps, using a constant median extension and the default fitting options, without any issues. However, when growing to the next step, the 3rd order polynomial fit and extrapolation fails, and the error message shown in Fig 9.3 is displayed in the terminal or CMD window. FRANC3D automatically switches the fit to a 1st order moving polynomial fit, Fig 9.4, and this fit is reasonable and crack growth could continue. However, we will highlight why the original fit fails

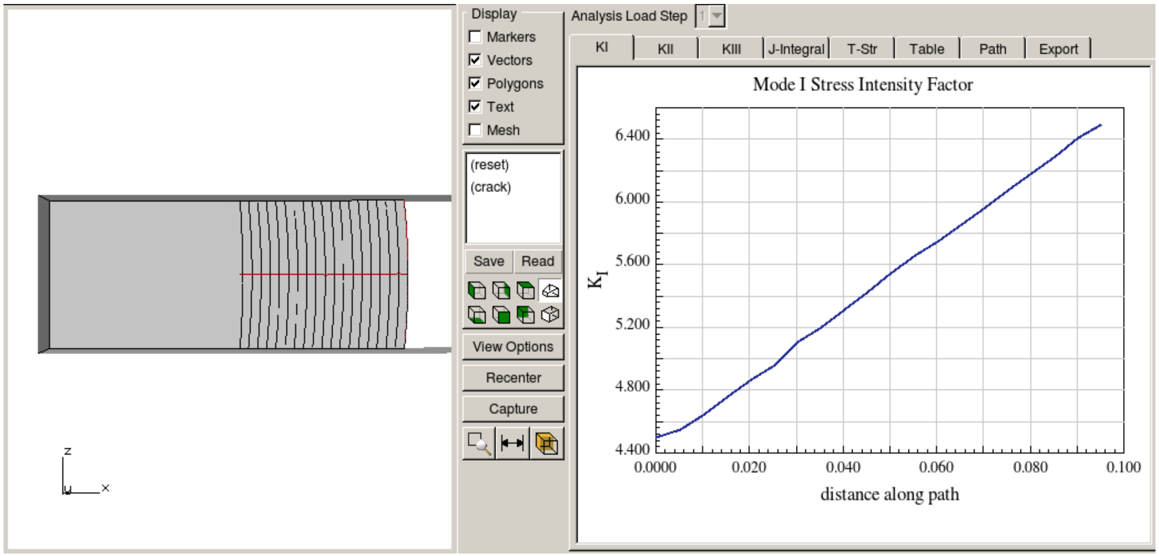
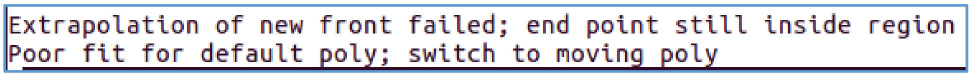


Figure 9.2 Edge crack in a plate propagated for 18 steps.

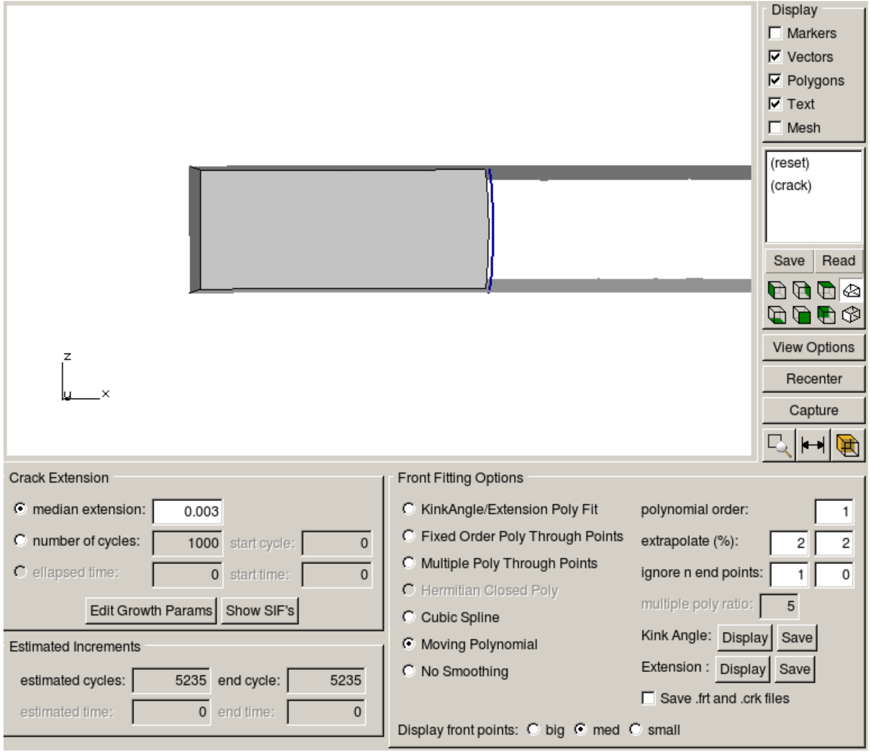


```
Extrapolation of new front failed; end point still inside region
Poor fit for default poly; switch to moving poly
```

Figure 9.3 Error message during front fitting for step 19.

It might not always be obvious why the fitting fails, especially when it works well for a number of steps prior to the failure. Fig 9.5 shows the computed new front points for the entire crack front, and Fig 9.6 shows the points at the two ends. There are a couple of things to point out in Fig 9.6. First, the elements at the two ends are distorted with different lengths compared to the adjacent elements. The distortion produces some numerical error in the computed SIFs.

Sometimes there is not much that FRANC3D can do to avoid this distortion and size variation while extending the template to the model surface.



Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)

Crack Extension
● median extension: 0.003
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

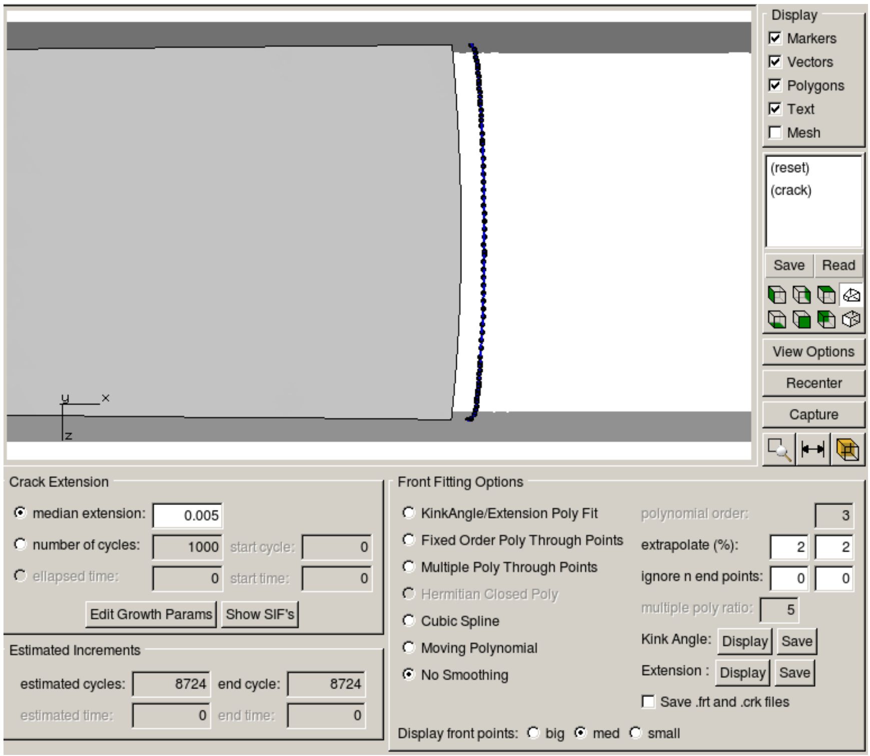
Estimated Increments
estimated cycles: 5235 end cycle: 5235
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 2 2
ignore n end points: 1 0
multiple poly ratio: 5
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.4 Step 19 front using a 1st order moving polynomial.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)

Crack Extension
● median extension: 0.005
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 8724 end cycle: 8724
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
● No Smoothing

polynomial order: 3
extrapolate (%): 2 2
ignore n end points: 0 0
multiple poly ratio: 5
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.5 Computed new front points – no smoothing.

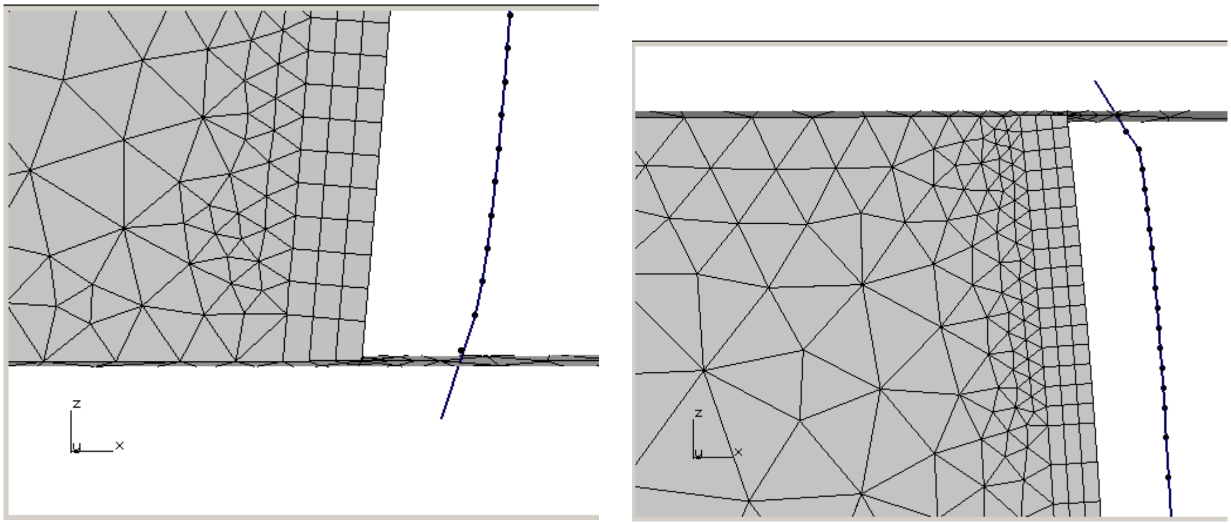


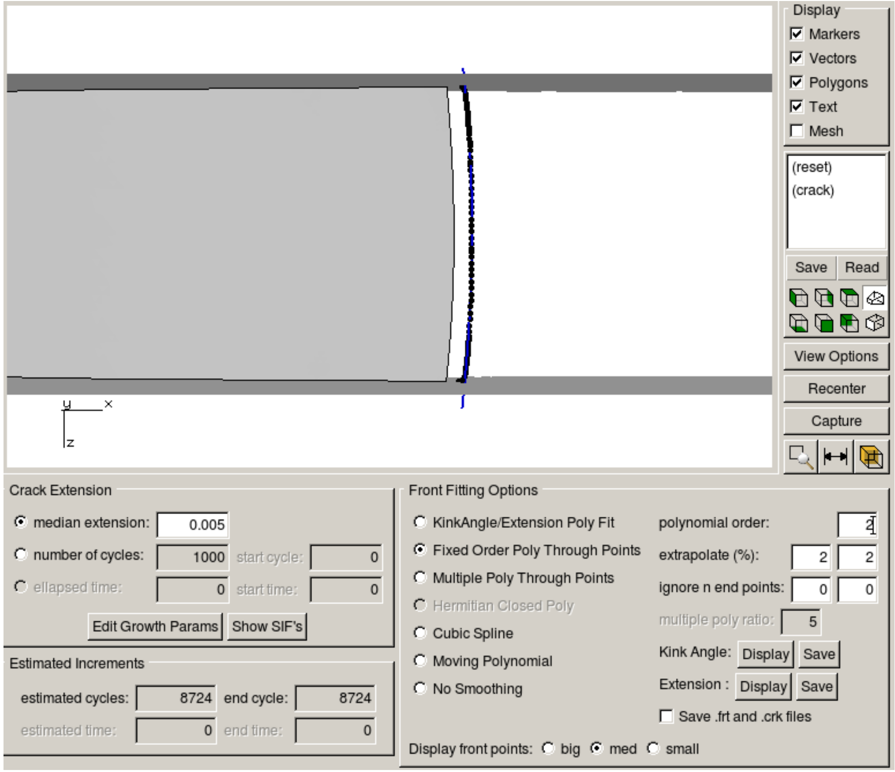
Figure 9.6 Computed new front points at the two ends of the front.

The distortion and variation in the element sizes leads to non-uniform spacing of the computed front points. The distortion in the element shape can lead to numerical noise in the SIFs, in particular the Mode II / Mode I ratio can change drastically at the model surface. The fixed-order single polynomial is fit through the set of 3D points using a least-squares method. The extrapolation of a polynomial, especially as the order increases, can be prone to undesired curvature. In this case, the combination of these things causes the 3rd order polynomial fit to fail as the end curls back inside the model and no amount of extrapolation will push the end outside of the model surface.

A simple fix is to use a lower order single polynomial for all steps of growth. Fig 9.7 shows a 2nd order polynomial fit to the step 19 front points. The fit captures the curvature of the front and the ends can easily be extrapolated outside of the model.

An additional example is shown in Figs 9.8 - 9.9. Fig 9.8 shows 33 steps of crack growth for an edge crack in a thin plate when using a 3rd order polynomial fit. Fig 9.9 shows 21 steps for the same crack in the same plate but using a linear polynomial fit.

The 3rd order fit requires ‘simple intersections’ for the crack front template (the template end is pulled back from the model surface – see Section 6.1.15 of the Reference document). The 3rd order fit produces convex/concave oscillations in the front shape, which leads to oscillations in the SIF history.



Crack Extension

● median extension: 0.005

○ number of cycles: 1000 start cycle: 0

○ ellapsed time: 0 start time: 0

Edit Growth Params | Show SIF's

Estimated Increments

estimated cycles: 8724 end cycle: 8724

estimated time: 0 end time: 0

Front Fitting Options

○ KinkAngle/Extension Poly Fit

● Fixed Order Poly Through Points

○ Multiple Poly Through Points

○ Hermitian Closed Poly

○ Cubic Spline

○ Moving Polynomial

○ No Smoothing

polynomial order: 2

extrapolate (%): 2 2

ignore n end points: 0 0

multiple poly ratio: 5

Kink Angle: Display Save

Extension : Display Save

□ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.7 Crack front fit using 2nd order polynomial.

The linear fit is not ideal and there are oscillations in the angles for the fronts due to the large steps in growth (and possible FE numerical errors), but the SIF history curve is smooth. A larger number of smaller growth steps, using the linear fit, would avoid some issues but at the expense of additional computing time.

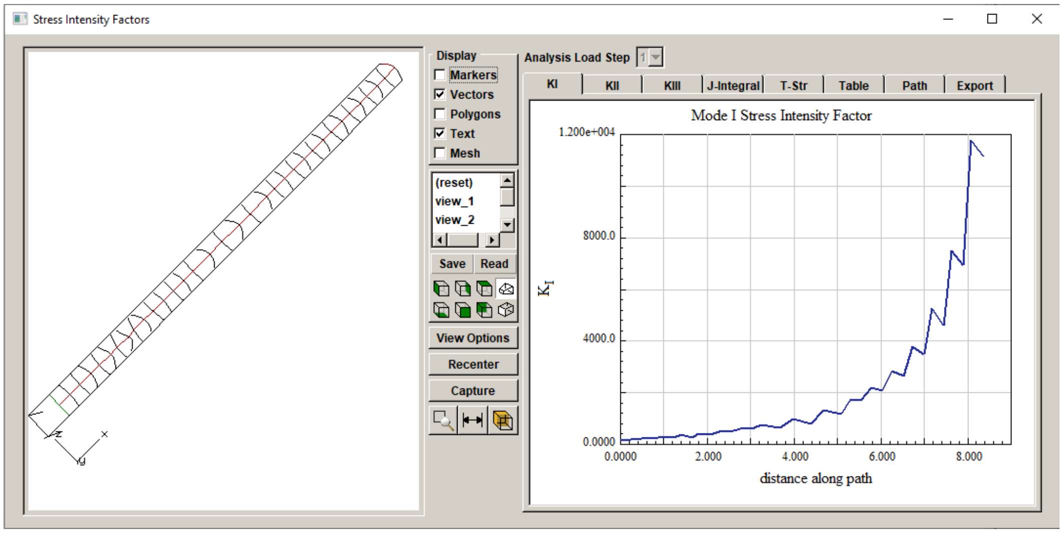
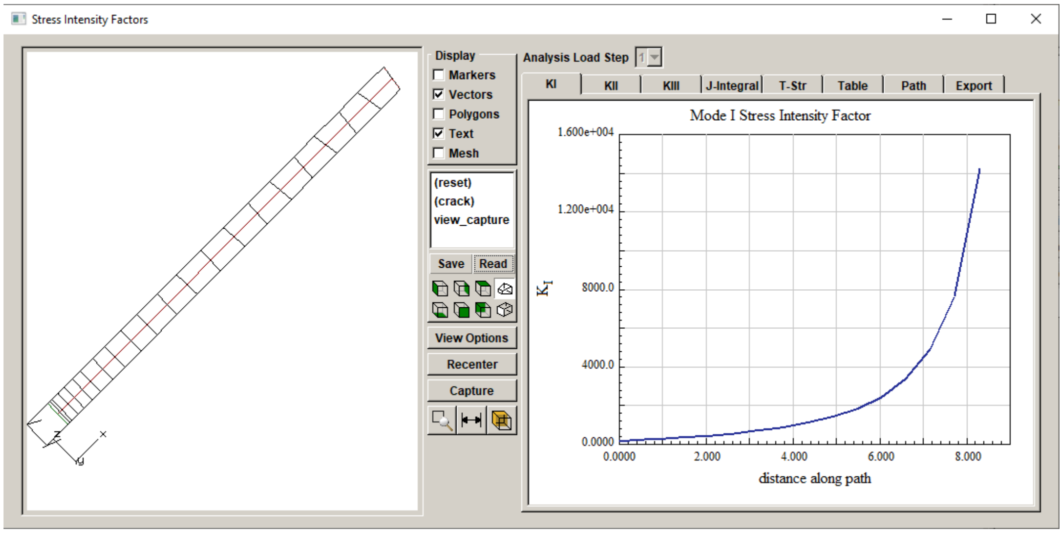


Figure 9.8 Crack front fit using 3rd order polynomial.



Stress Intensity Factors

Display
☐ Markers
☑ Vectors
☐ Polygons
☑ Text
☐ Mesh

(reset)
(crack)
view_capture

Save Read

View Options

Recenter

Capture

Analysis Load Step 1

KI KII KIII J-Integral T-Str Table Path Export

Figure 9.9 Crack front fit using 1st order polynomial.

9.2 Long Shallow Cracks fit with Multiple Polynomials

A long shallow surface crack is another shape that fails with a 3rd order single polynomial fit. Sometimes an initial half-penny-shape surface crack will evolve into a relatively long shallow surface crack, and the fit will have to change as the crack propagates. Fig 9.10 shows an initial long shallow surface crack in a simple cube subjected to uniform tension. Fig 9.11 shows the resulting computed new front points; clearly a single 3rd order polynomial is not capable of fitting this set of points.

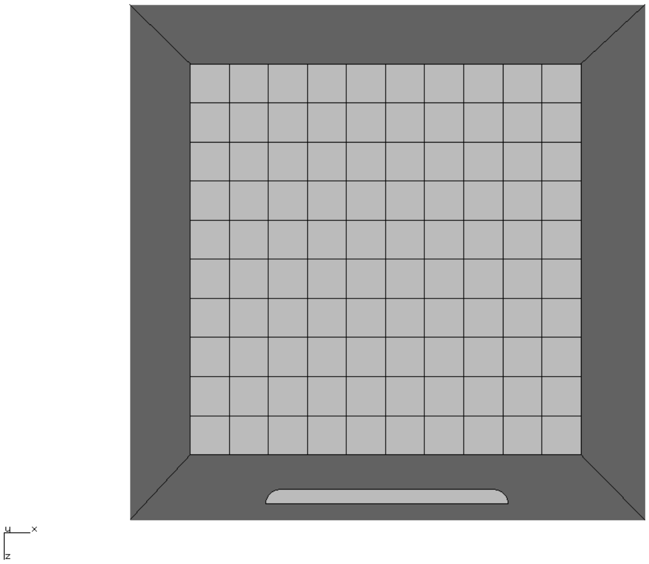


Figure 9.10 Initial long shallow crack.

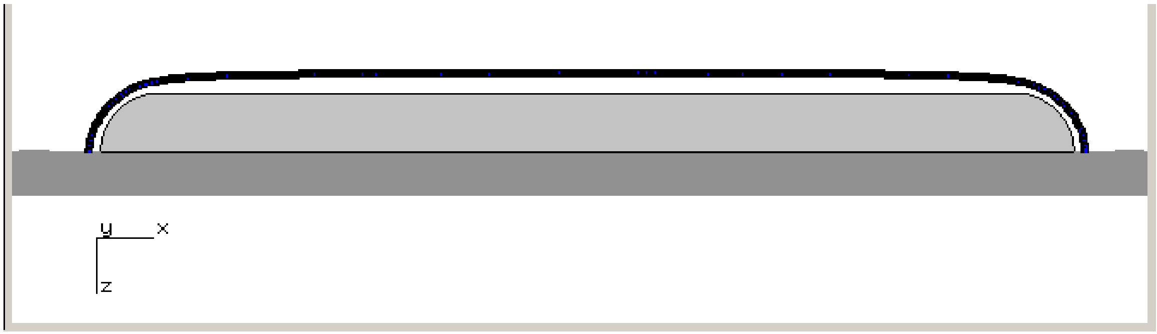
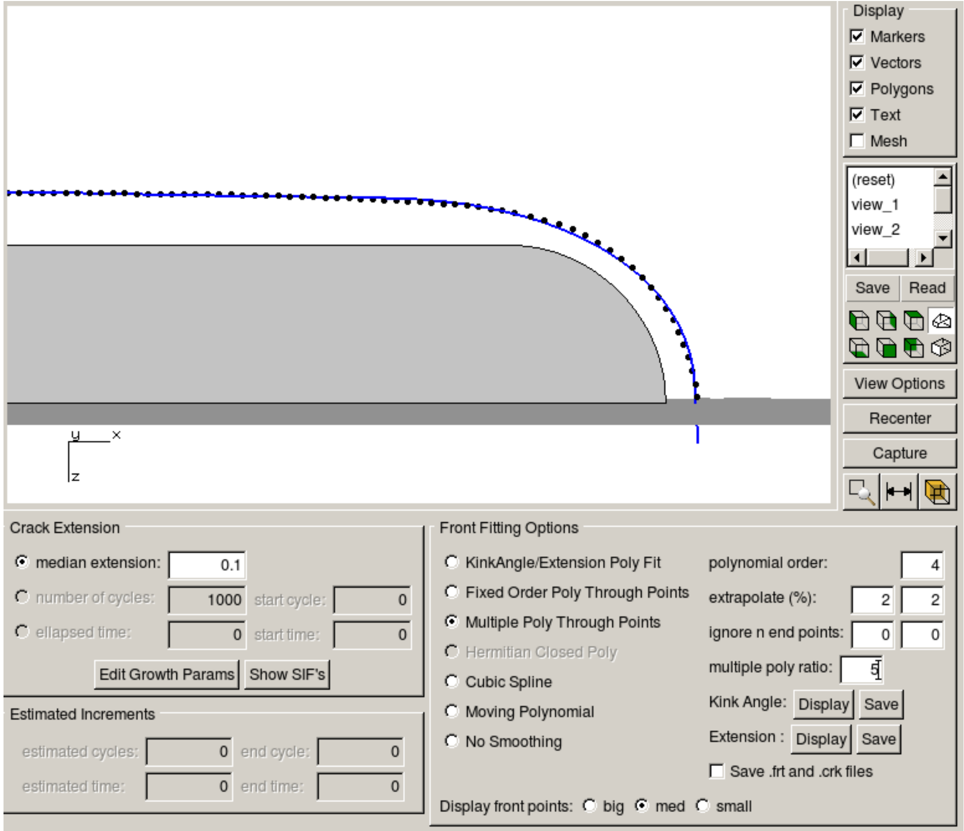


Figure 9.11 Computed new front points for initial long shallow crack.

The multiple polynomial option was added specifically for this crack shape. Three polynomials are used – one for the middle portion and one for each of the two ends. The polynomial order can be set, but usually a 4th order gives a reasonable fit for all three segments. The multiple poly ratio setting allows the user to adjust how much of the crack front is fit by each segment. The default ratio is 5, so at each end, the segment consists of the total number of front points divided by 5. Each end is fit separately from the middle segment. The ends of the fit segments are blended after fitting. However, it is possible to create kinks in the crack front geometry that will lead to numerical noise/error and potential oscillations in the subsequent front(s).

Fig 9.12 shows the fit at one end of the front using the typical settings. Fig 9.13 shows the effect of increasing the multiple poly ratio value from 5 to 25, which means fewer points in each end segment. While the end segment fit is okay, the middle segment fit is poor, and a kink is created when trying to blend the curves together. Fig 9.14 shows the effect of lowering the polynomial order to 2 while using the default poly ratio. In this case, a 2nd order polynomial cannot capture the curvature at the end, and a poor connection between the end and middle segments is created. In practice, once the appropriate settings are determined for a crack front, those settings should work for several steps of propagation.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
view_2

Save Read

View Options

Recenter

Capture

Crack Extension

● median extension: 0.1
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Edit Growth Params Show SIF's

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

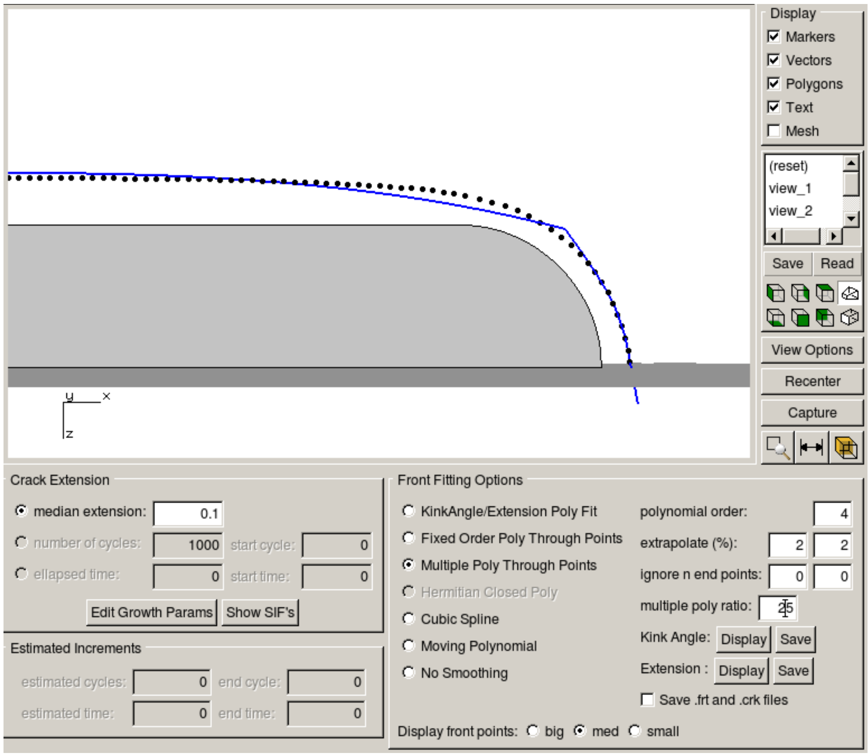
Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
● Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 4
extrapolate (%): 2 2
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.12 Fit for long shallow crack using typical settings.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
view_2

Save Read

View Options

Recenter

Capture

Crack Extension

● median extension: 0.1
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Edit Growth Params Show SIF's

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

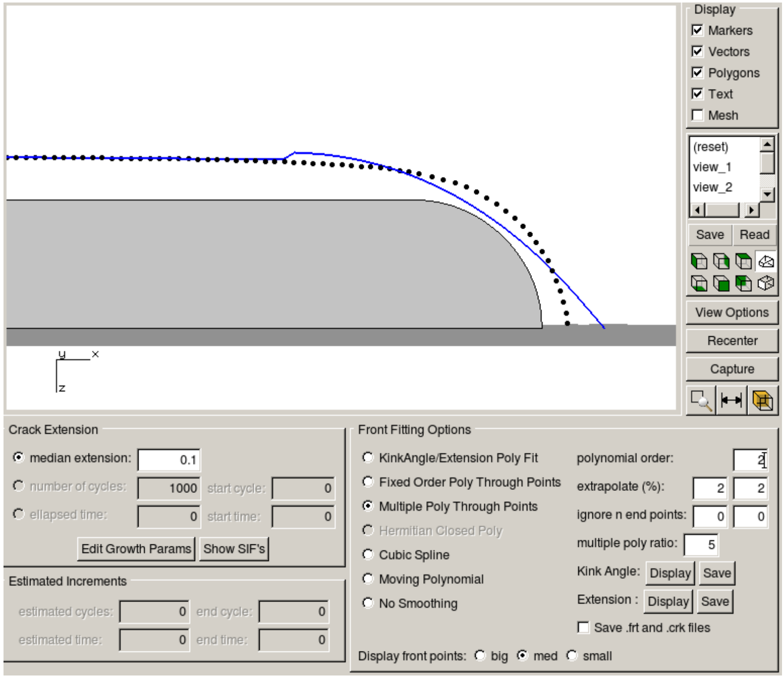
Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
● Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 4
extrapolate (%): 2 2
ignore n end points: 0 0
multiple poly ratio: 25
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.13 Fit for long shallow crack using a higher ratio.



Display

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Crack Extension

● median extension: 0.1
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
● Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 2
extrapolate (%): 2 2
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle:
Extension :
□ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.14 Fit for long shallow crack using a lower order polynomial.

9.3 Moving Polynomial

If the multiple polynomial fit is not acceptable, one can try using the moving polynomial fit. This fit was originally added for extremely complicated crack front shapes. For example, Fig 9.15 shows an observed flaw shape with a 12th order single polynomial (left image) and a low-order moving-polynomial fit (right image). The single polynomial cannot capture the shape. The multiple polynomial fit described above does not work well either. A cubic spline captures the shape; however, the moving polynomial has more flexibility and captures the predicted new shape.

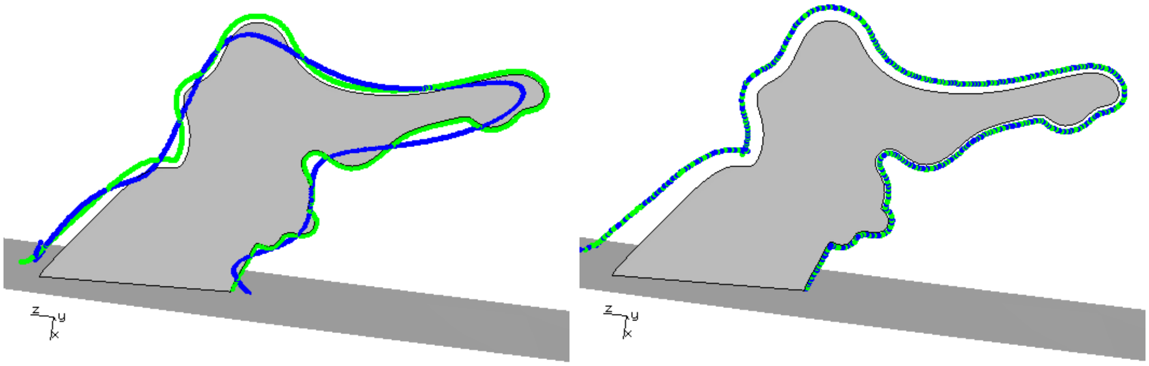


Figure 9.15 Observed flaw shape with a 12th order polynomial (left) and a moving polynomial (right) fit to the new front points.

We note that for crack fronts with concave segments, the new crack front points can overlap depending on the amount of extension. The moving polynomial (and the cubic spline) fitting option looks for reversals in the crack front points, and points are discarded if this occurs.

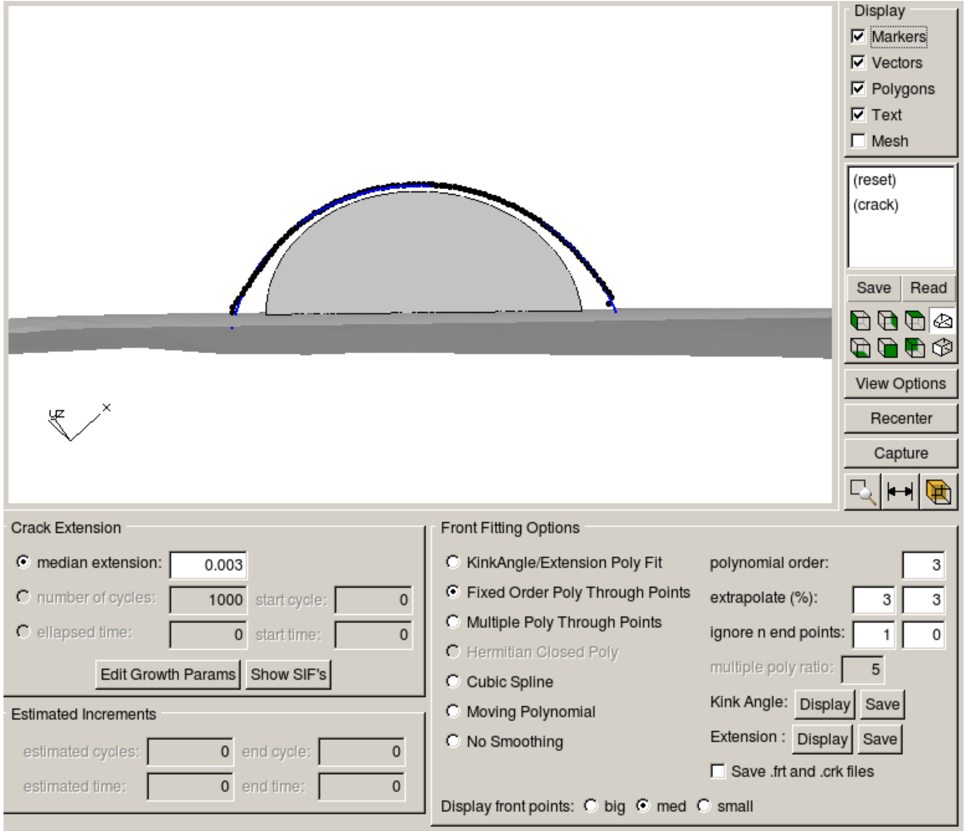
A moving polynomial smooths the front points by replacing each point with the fit-point based on neighboring points defined within the span and the specified polynomial order. The default span is 5, but this is adjusted for the polynomial order. The ends are extrapolated using a linear fit through the last few points, thus the required extrapolation % is usually less than for the typical single polynomial fit.

Typically, a 2nd order moving polynomial provides a reasonable fit. Very high-order polynomials are not recommended as they will tend to capture more of the minor variations due to numerical noise/error in the SIFs.

The moving polynomial often can be used for relatively long shallow almost-ellipse shaped cracks. Quite often users will insert a half-penny surface crack into a model and the crack will propagate further at the model surface than at the middle (deepest part of the crack), which will lead to a more ellipse-like shape.

Fig 9.16 shows predicted growth from initial surface crack with aspect ratio (0.05/0.04=) 1.25. The surface SIFs are higher, Fig 9.17, which leads to more growth at the surface. The relative extension along the crack front is a function of the SIFs and the growth model (see Section 6.5 of the Reference document).

Fig 9.16 shows a single 3rd order polynomial fit through the new front points. In this case, the polynomial fits. We can ignore one or more end points on either end; as seen earlier, the SIFs at the free surface are subject to more numerical noise/error so ignoring the end points is sometimes the best option. Fig 9.18 shows the Mode II SIFs; the first point shows a significant drop from the curve trend, and FRANC3D automatically ignores that one end point. Crack growth involves both extension and kink angle, and noise/error can exist in either or both, leading to badly predicted new front points.



Crack Extension

● median extension: 0.003

○ number of cycles: 1000 start cycle: 0

○ ellapsed time: 0 start time: 0

Edit Growth Params | Show SIF's

Estimated Increments

estimated cycles: 0 end cycle: 0

estimated time: 0 end time: 0

Front Fitting Options

○ KinkAngle/Extension Poly Fit

● Fixed Order Poly Through Points

○ Multiple Poly Through Points

○ Hermitian Closed Poly

○ Cubic Spline

○ Moving Polynomial

○ No Smoothing

polynomial order: 3

extrapolate (%): 3 3

ignore n end points: 1 0

multiple poly ratio: 5

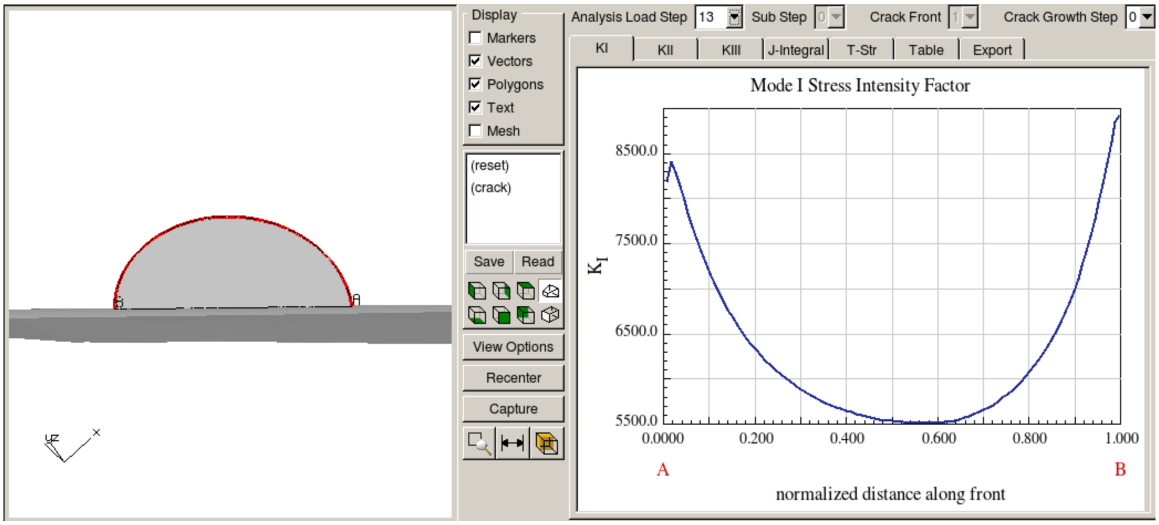
Kink Angle: Display Save

Extension : Display Save

□ Save .frt and .crk files

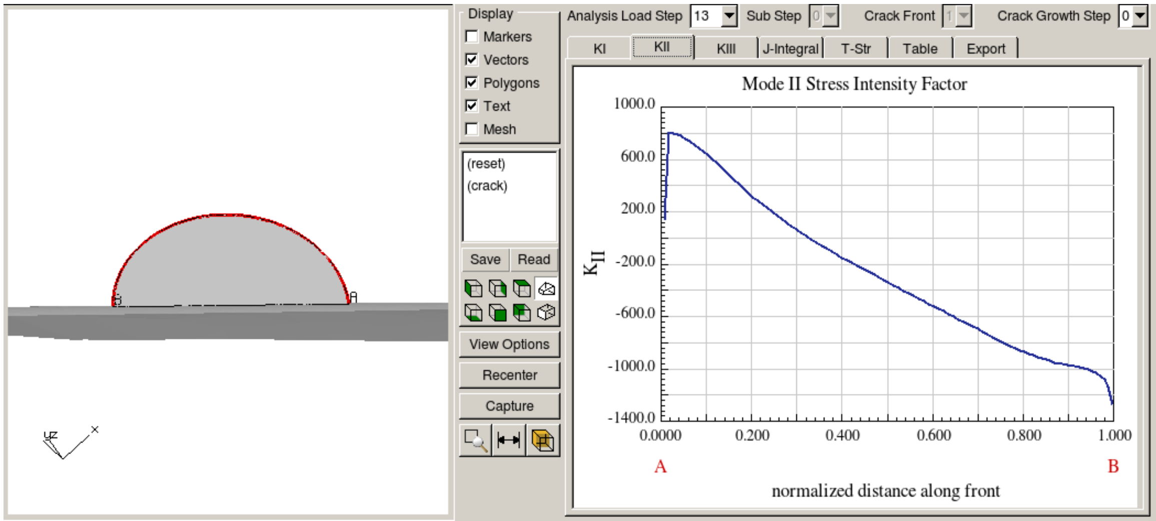
Display front points: ○ big ● med ○ small

Figure 9.16 Predicted new front points for an almost-half-penny surface crack.



Analysis Load Step 13 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 9.17 Mode I SIFs for initial almost-half-penny surface crack.



Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 13 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 9.18 Mode II SIFs for initial almost-half-penny surface crack.

As this crack continues to propagate, it continues to grow more at the surface than at the deepest point. This leads to a higher aspect ratio shape, Fig 9.19. Eventually the 3rd order single polynomial fit might fail to capture the shape correctly, Fig 9.20. In this case, FRANC3D should issue a message to the terminal/cmd window and automatically switch to a moving polynomial, Fig 9.21. The multiple-polynomial fit also works okay for this crack, Fig 9.22.

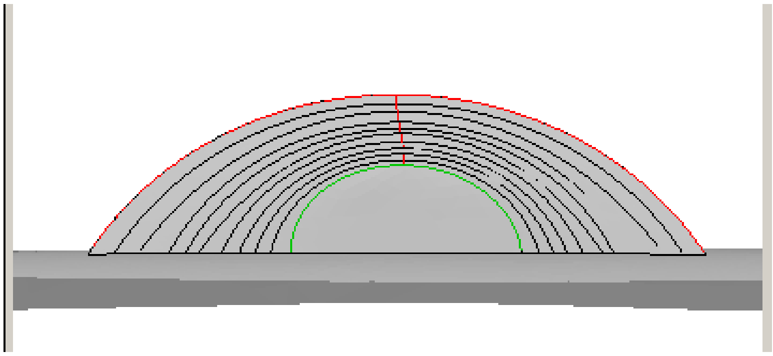
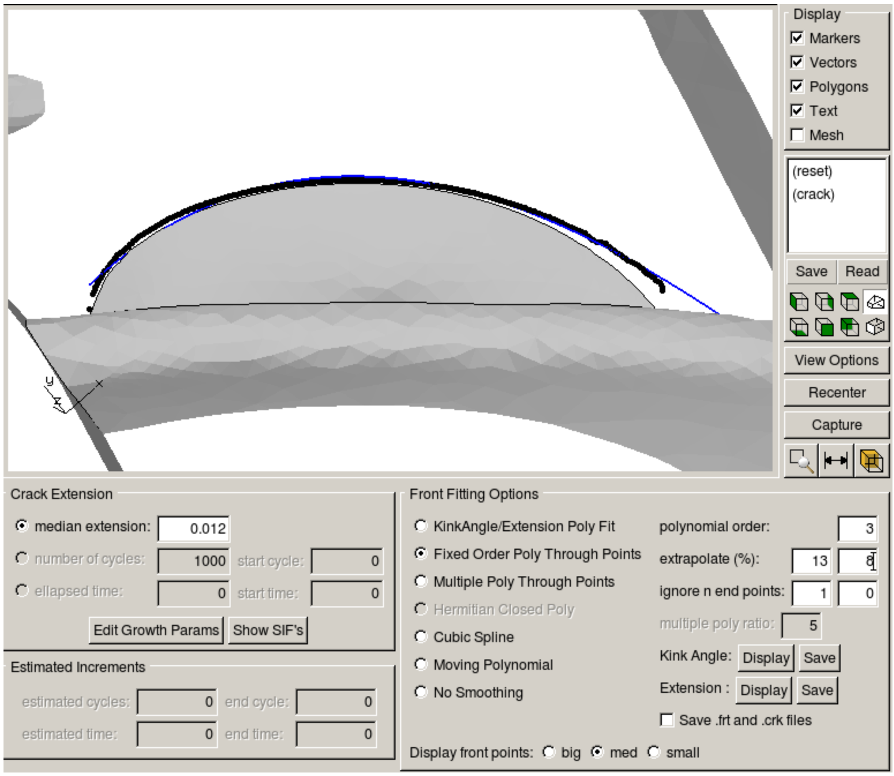


Figure 9.19 After 10 steps of propagation.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset)
(crack)

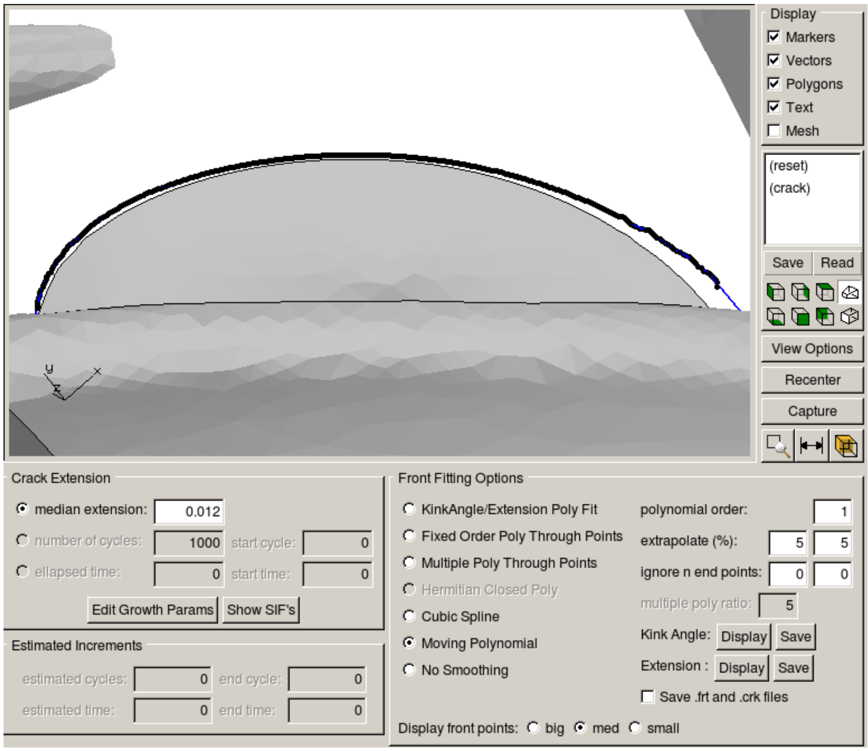
Crack Extension
● median extension: 0.012
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
● Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 3
extrapolate (%): 13 8
ignore n end points: 1 0
multiple poly ratio: 5
□ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 9.20 A 3rd order single polynomial fails after a number of steps of propagation.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset)
(crack)

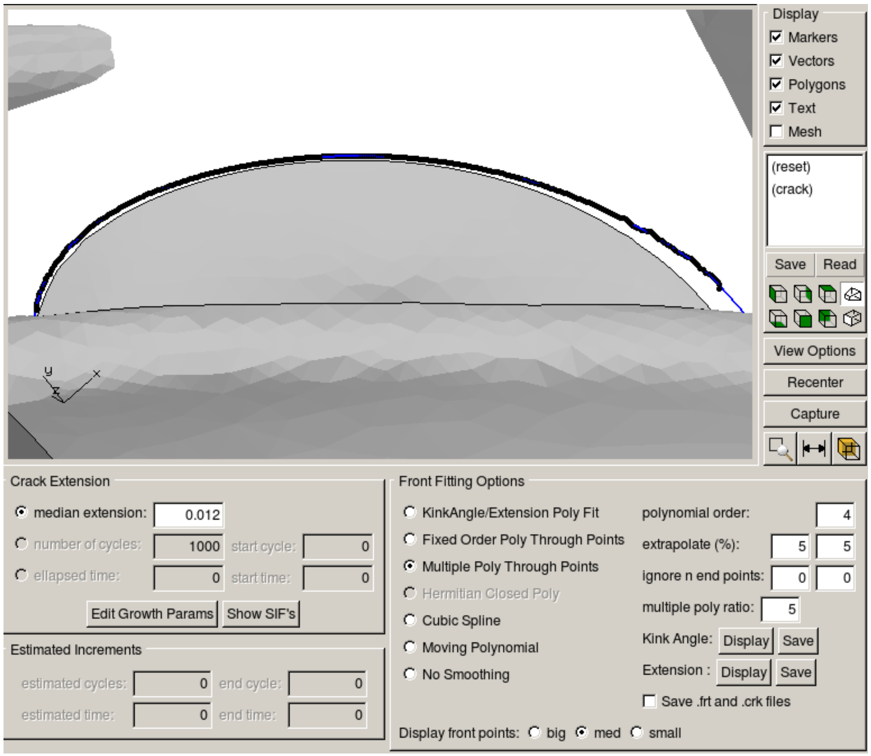
Crack Extension
● median extension: 0.012
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 5 5
ignore n end points: 0 0
multiple poly ratio: 5
□ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 9.21 1st order moving polynomial fit.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Crack Extension
● median extension: 0.012
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

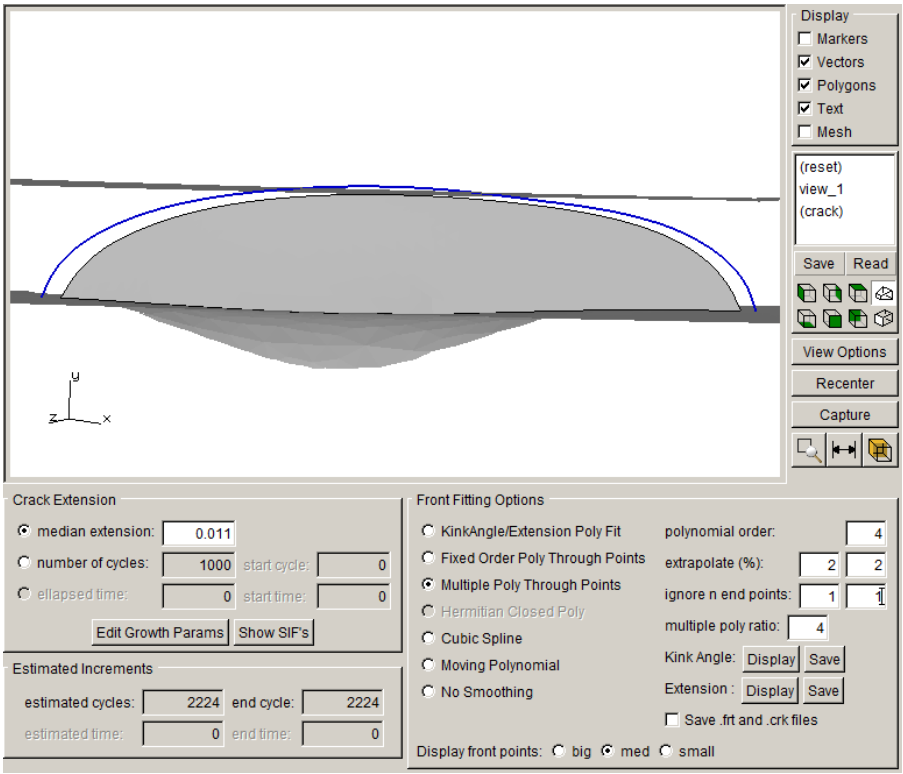
Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
● Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 4
extrapolate (%): 5 5
ignore n end points: 0 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
□ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.22 4th order multiple polynomial fit.

In some cases, the multiple-polynomial fit might work better than the moving polynomial. Figs 9.23 - 9.24 show a more elongated surface crack that is breaking through the back surface on the model. The 4th order multiple-polynomial provides a good fit to the front points. The moving polynomial provides a good fit but captures the reduced growth on the front model surface. While this reduction might be valid, it can lead to difficulties creating a good template mesh and can lead to crack geometry overlap outside of the model, which will cause the growth to fail.



Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
(crack)

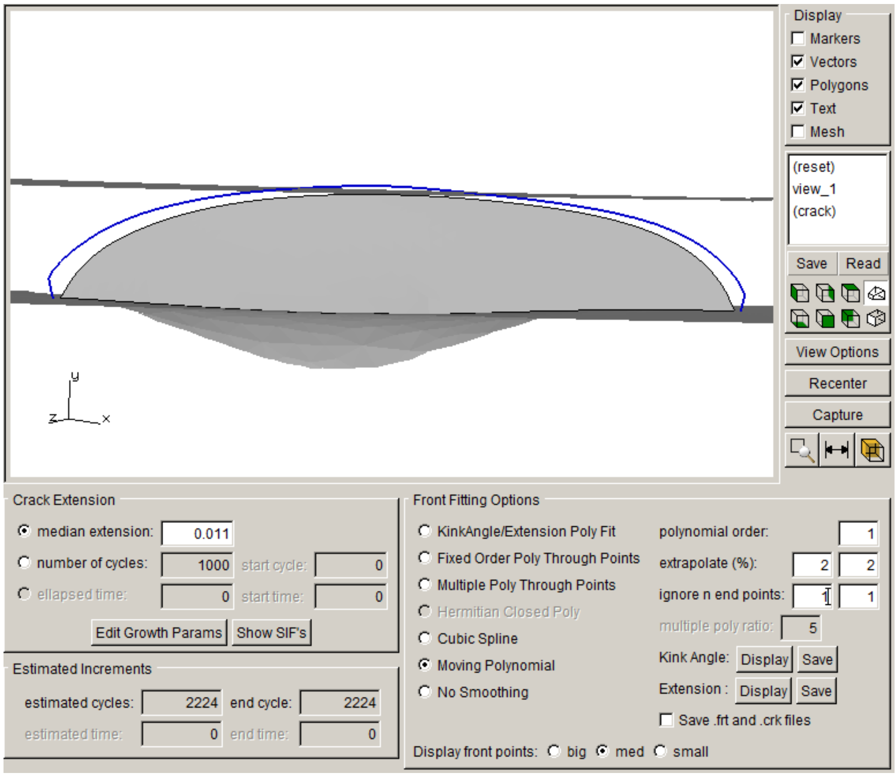
Crack Extension
● median extension: 0.011
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 2224 end cycle: 2224
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
● Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
○ Moving Polynomial
○ No Smoothing

polynomial order: 4
extrapolate (%): 2 2
ignore n end points: 1 1
multiple poly ratio: 4
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 9.23 4th order multiple polynomial fit.



Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
(crack)

Crack Extension
● median extension: 0.011
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments
estimated cycles: 2224 end cycle: 2224
estimated time: 0 end time: 0

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 2 2
ignore n end points: 1 1
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files
Display front points: ○ big ● med ○ small

Figure 9.24 1st order moving polynomial fit.

9.4 Concave Crack Fronts

While most crack fronts have a convex shape, it is possible to have concave portions along the front, Fig 9.25. This can produce new crack front points that overlap, Fig 9.26, depending on the amount of extension requested by the user and the number of nodes/elements along the existing crack front. A user should ensure that the curve-fit, through these points, does not also have overlap, Fig 9.27.

The moving polynomial fitting option will check for overlap and remove points to prevent the curve-fit from overlapping, Fig 9.28. The user might need to adjust the amount of extension to get a smooth curve; small kinks or jumps in the crack front geometry usually leads to bumps in the SIFs along the front.

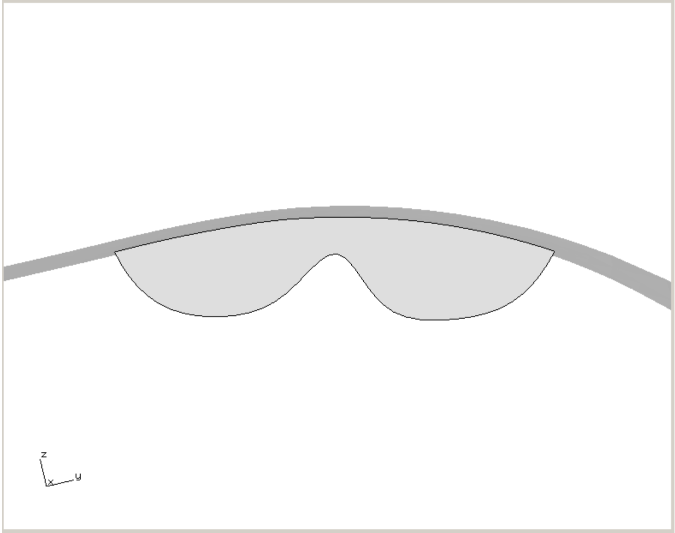


Figure 9.25 Concave portion of a crack front.

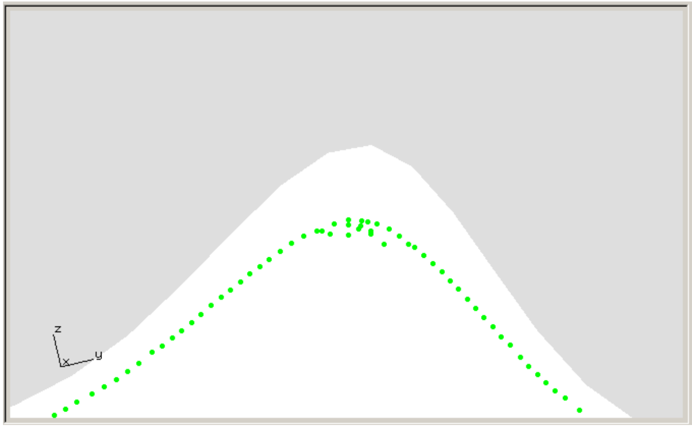


Figure 9.26 Overlap of predicted new front points.

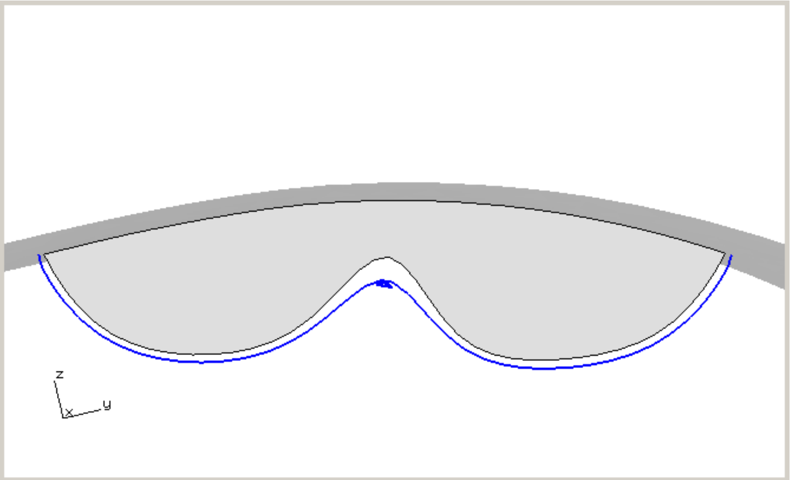
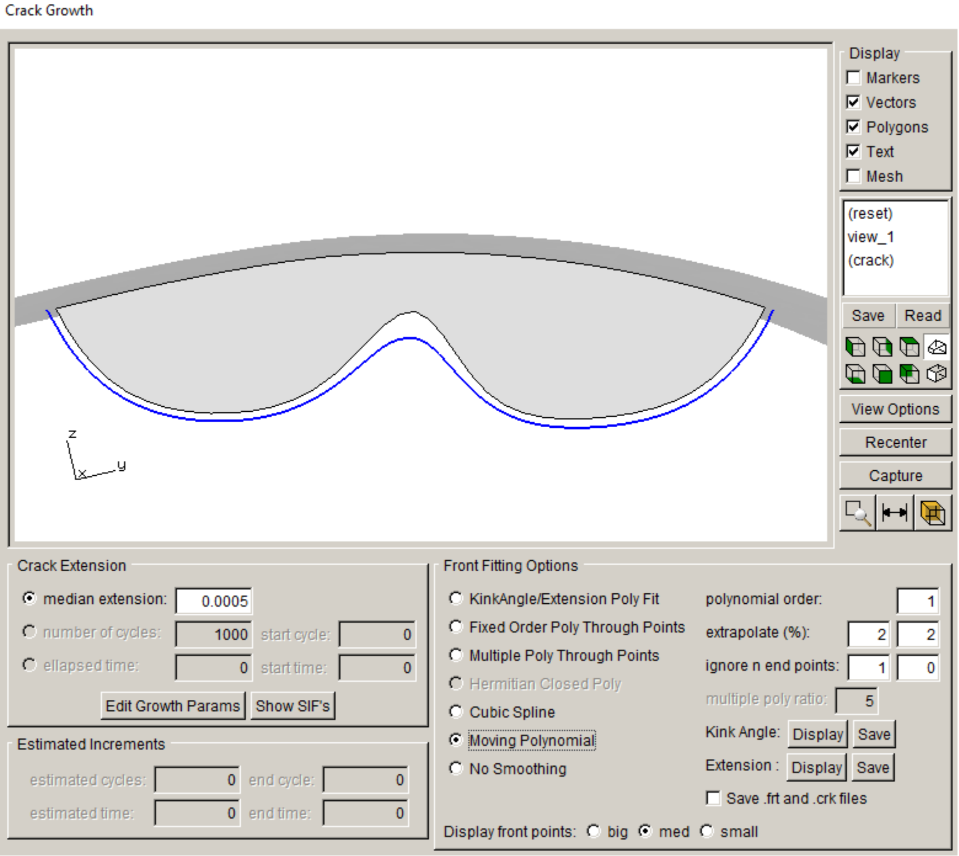


Figure 9.27 Overlap of new front curve – will fail to produce crack growth.



Crack Growth

Display

☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
view_1
(crack)

Crack Extension

● median extension: 0.0005
○ number of cycles: 1000 start cycle: 0
○ ellapsed time: 0 start time: 0

Estimated Increments

estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

Front Fitting Options

○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 1
extrapolate (%): 2 2
ignore n end points: 1 0
multiple poly ratio: 5
Kink Angle: Display Save
Extension : Display Save
☐ Save .frt and .crk files

Display front points: ○ big ● med ○ small

Figure 9.28 Overlap points removed during smoothing using the moving polynomial.

9.5 Partial Crack Front Extension

Most models have cracks that grow by extending the entire crack front by a discrete amount. However, there are cases where partial front extension occurs. Consider a plate with an edge crack that is subjected to pure bending. In this case, only half of the crack front advances (Fig 9.29). FRANC3D can simulate this growth.

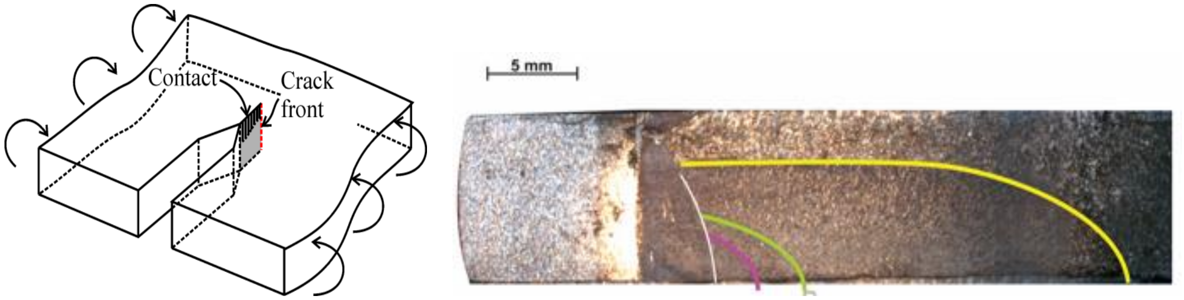
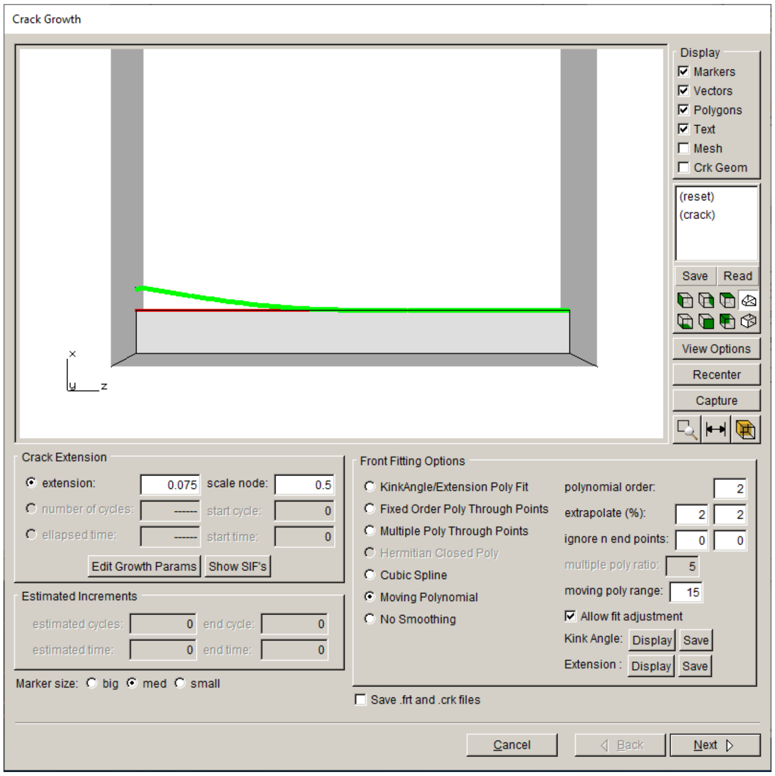


Figure 9.29 Plate with an edge crack subjected to bending; from:
Corbani *et al*, Crack shape evolution under bending-induced partial closure, EFM, 188, p493-508.

A simple brick model subject to pure bending is used to illustrate this growth. An edge crack is inserted in the side of the brick and analyzed. The predicted new front is shown in Fig 9.30. FRANC3D recognizes that part of the front has no growth; there is a comparison between the computed extension at each point along to crack front with a tolerance, where the tolerance is model and crack dependent. To fit a curve through the new front points for this type of growth, FRANC3D automatically switches to a moving polynomial fit (see Fig 9.30).

Fig 9.31 shows a series of crack front shapes and the corresponding SIF history along a path through the fronts where there is non-zero growth.



Crack Growth

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Crk Geom

(reset)
(crack)

Crack Extension
● extension: 0.075 scale node: 0.5
○ number of cycles: ------ start cycle: 0
○ ellapsed time: ------ start time: 0

Estimated Increments
estimated cycles: 0 end cycle: 0
estimated time: 0 end time: 0

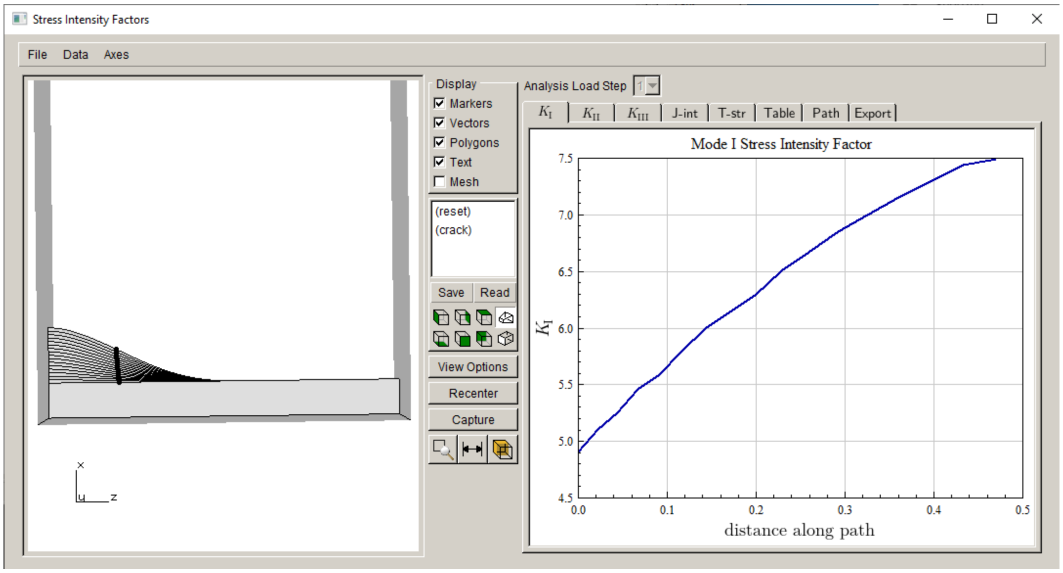
Marker size: ○ big ● med ○ small

Front Fitting Options
○ KinkAngle/Extension Poly Fit
○ Fixed Order Poly Through Points
○ Multiple Poly Through Points
○ Hermitian Closed Poly
○ Cubic Spline
● Moving Polynomial
○ No Smoothing

polynomial order: 2
extrapolate (%): 2 2
ignore n end points: 0 0
multiple poly ratio: 5
moving poly range: 15
☑ Allow fit adjustment

□ Save .frt and .crk files

Figure 9.30 Edge crack subjected to bending – showing partial crack front extension.



Stress Intensity Factors

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset)
(crack)

Analysis Load Step 1

Figure 9.31 Edge crack subjected to bending – showing partial crack front extension.

9.6 Crack Front Template Issues

If you found an acceptable fit to the crack front points, the next issue might be the crack front template. Problems can arise if the template spans geometric kinks in the crack surface. Fig 9.32 shows the crack extension for an initially inclined crack that wants to turn; the left image shows the left end of the crack. Fig 9.33 shows the template with the radius set such that template spans the kink in the geometry, and the resulting mesh is shown in the right image. In this case, the template was added and meshed successfully, although SIFs at the left end will usually be poor due to the distorted elements. The FE analysis code might complain about inverted elements or negative Jacobians and not produce a solution.

In many cases, the template will not be added successfully, and a generic error message will be displayed, Fig 9.34. One solution is to make the template radius smaller.

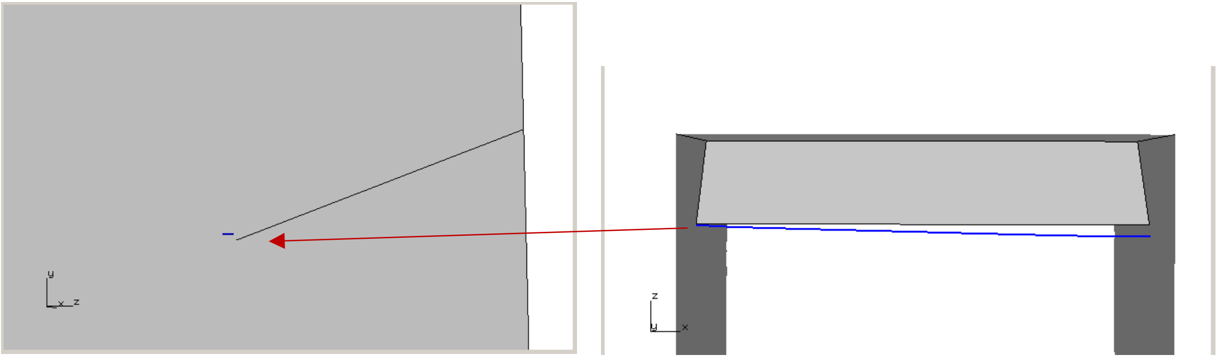


Figure 9.32 Crack front turns (kinks).

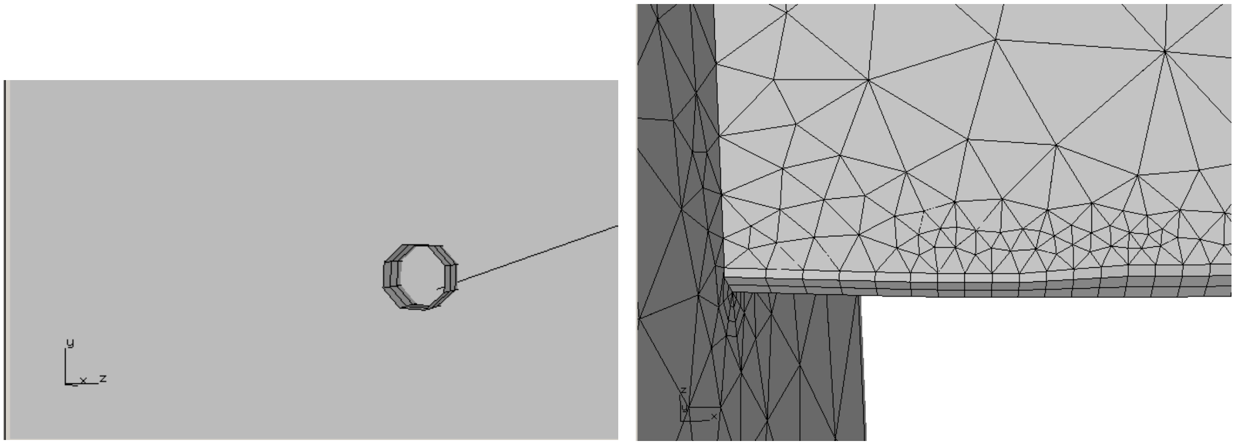
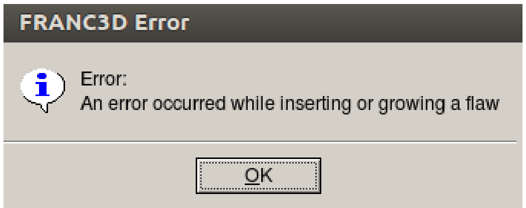


Figure 9.33 Template spans the kink in the geometry.


FRANC3D Error

Error:
An error occurred while inserting or growing a flaw

Figure 9.34 Error message displayed if the template cannot be formed

Sometimes a user might see a template that appears twisted. This is usually just a display issue; FRANC3D uses a simplified algorithm to display the template in the GUI compared to the algorithm that is used to do the actual insertion and remeshing. If there are kinks in the crack front/surface geometry, as in Fig 9.35, the template displayed to the user might appear to be twisted (right image). The user can usually ignore this and proceed with the insertion and meshing.

For any case where crack insertion and meshing fail, if possible, the user should send us the *debug.tst* file so that we can determine the problem and fix any bugs in the code.

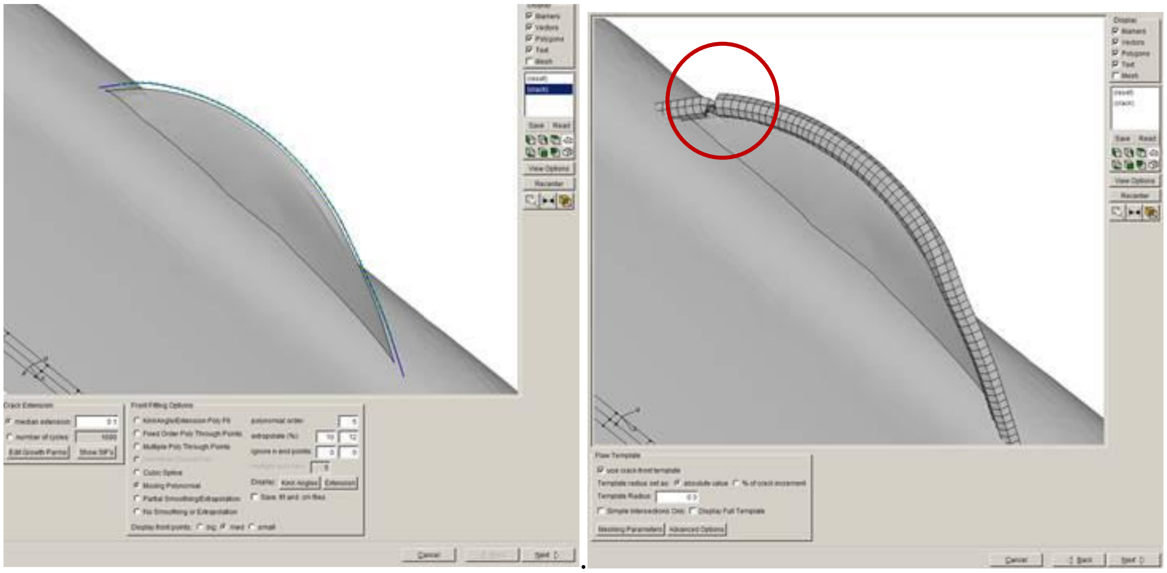


Figure 9.35 Twist in displayed template.

9.7 Unseen Errors

Crack front fitting and extrapolation can produce crack geometry that is poor or fails when inserting and remeshing. As described above, one potential failure is due to the template spanning kinks in the crack surface geometry.

9.7.1 Oscillating Crack Front Shapes

Another potential error occurs due to excessive extrapolation of polynomial fits. This is especially true for the case where the crack front oscillates between convex and concave. Fig 9.36 shows the crack geometry for an edge crack in a thin plate (see Fig 9.8). The fronts between steps #5 and 9 show significant oscillation from convex to concave. The concave fronts also have excessive amounts of extrapolation. The right image shows the geometry in relation to the plate. In this case, the ends of the fronts and geometry (triangular Bezier patches) do not actually overlap so the crack can be inserted and meshed successfully. For cases where there is overlap, FRANC3D will output an error message “failed to triangulate”.

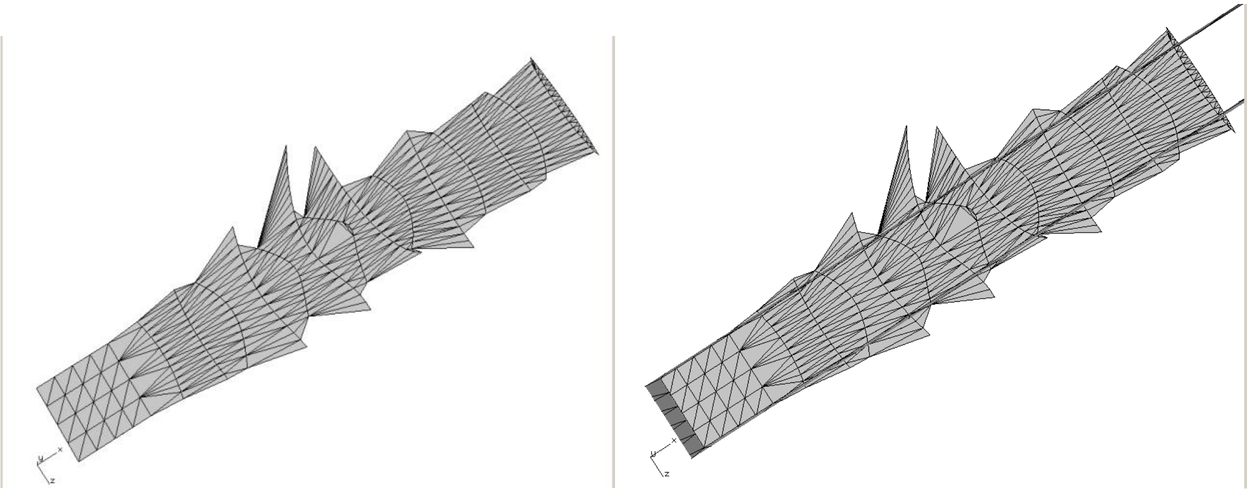


Figure 9.36 Edge crack geometry in a thin plate at step 15.

Over-extrapolation can lead to overlapping of the crack front geometry that will lead to a “failure to triangulate” error message and no crack growth will occur.

9.7.2 Overlapping Crack Geometry

An additional limitation of the crack growth algorithm occurs for the case of highly non-planar crack growth, Fig 9.37. The bounding points of the current and new crack front are collapsed to a least-squares plane, and a 2D triangulation algorithm is used to create the triangular facet geometry representing the new crack surface. For the set of points in Fig 9.37, the corresponding least-squares boundary is shown in Fig 9.38. The bounding edges cross, which causes the triangulation to fail, so no crack growth occurs.

When this occurs, FRANC3D reverts to a simple triangulation scheme that connects points on the current front with the new front points. In practice, this should give a reasonable new crack surface geometry if there are sufficient points on the front.

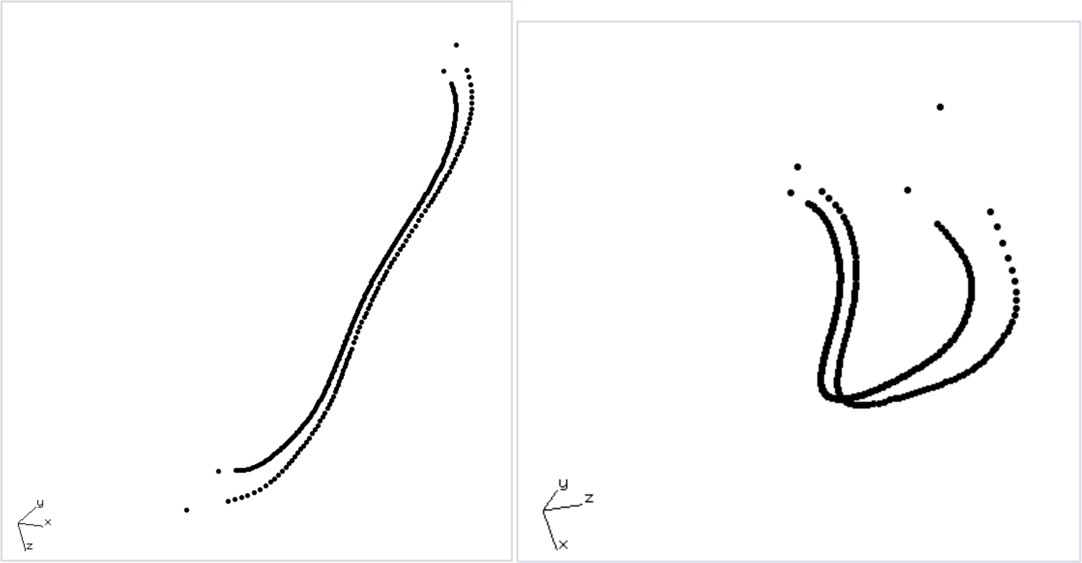


Figure 9.37 Current and new crack front points in 3D space.

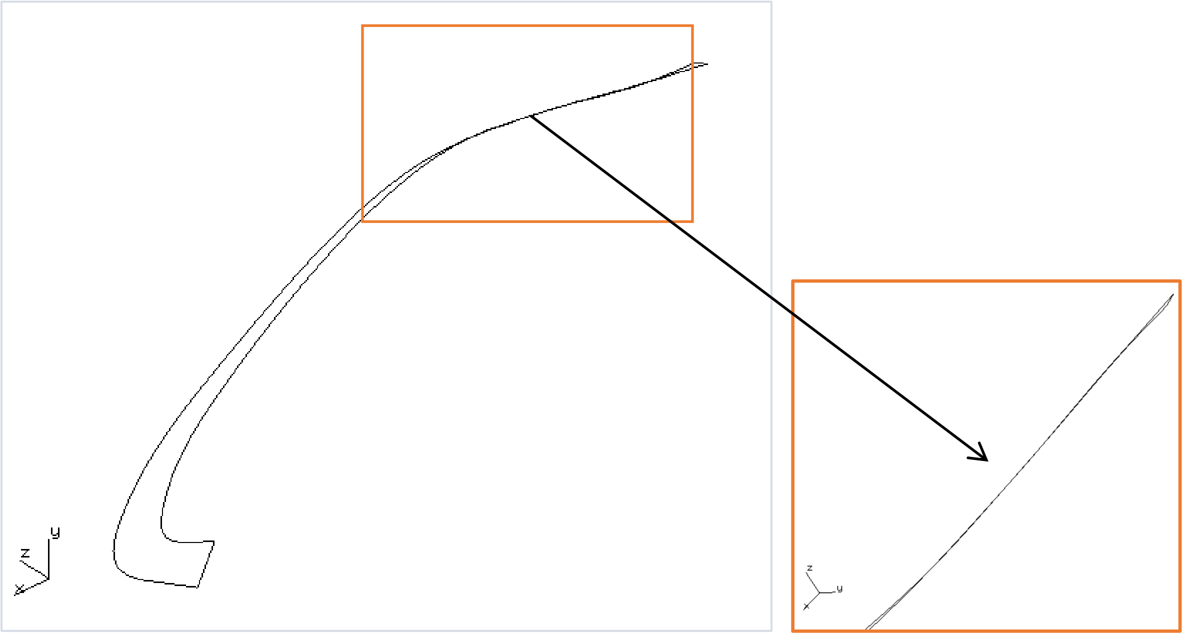


Figure 9.38 Current and new crack front boundary in least-squares planar space; the bounding edges cross leading to a “no crack growth occurred” error.

9.7.3 Partial Growth

Partial crack front extension is managed a little differently than a typical growth step where there is finite growth along the entire crack front. Fig 4.7 showed an image of a crack with two steps of growth where new geometry is added along the entire front. For partial extension, the geometry is modified by deleting some existing crack geometry surface and then re-triangulating the old and new surface region. This usually works well, but there are limitations. For example, if the initial crack geometry is too coarse compared to the region of partial extension, re-triangulation can be difficult. If the user expects partial front extension, it is best to start with a refined geometry.

For example, partial growth is shown in Fig 9.39. While the mesh is refined enough to capture the growth, the underlying geometry is not, Fig 9.40. Removing geometry where there is partial growth means we must remove geometry where there is no growth. Forming new triangulated geometry to include the partial growth while tying the new and old geometry together leads to poor triangles.

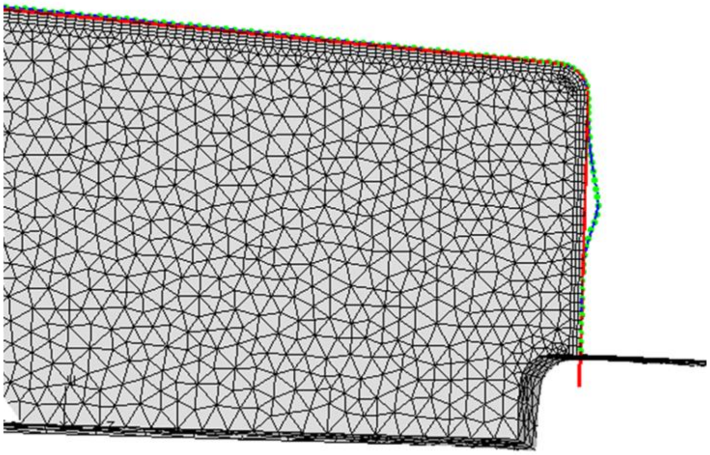


Figure 9.39 Partial crack front extension.

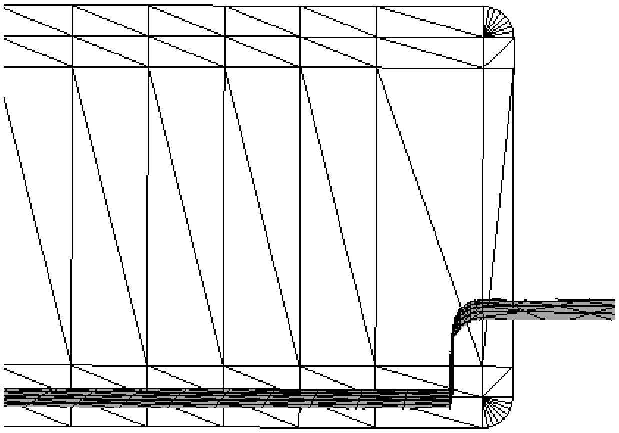


Figure 9.40 Crack geometry cannot easily be modified for partial growth.

10. Crack Growth with Multiple Load Cases

Tutorial #2 describes some aspects of multiple load steps but does not include crack growth. Tutorial #10 describes how one can combine two load steps with a hold time. Tutorial #14 describes an example of combined multiple low and high cycle fatigue load steps.

This chapter is used to illustrate how multiple load steps can be combined for crack growth and to illustrate some issues that occur with transient load steps or load steps with multiple substeps.

10.1 Combining (static) Load Steps (without substeps)

This section describes how multiple load steps combine for crack growth. We start with a simple cube model in ABAQUS where the first load step is tension with a gradient in the x-direction and the second load step is uniform tension on the top surface. The bottom is constrained to prevent rigid body motion, Fig 10.1.

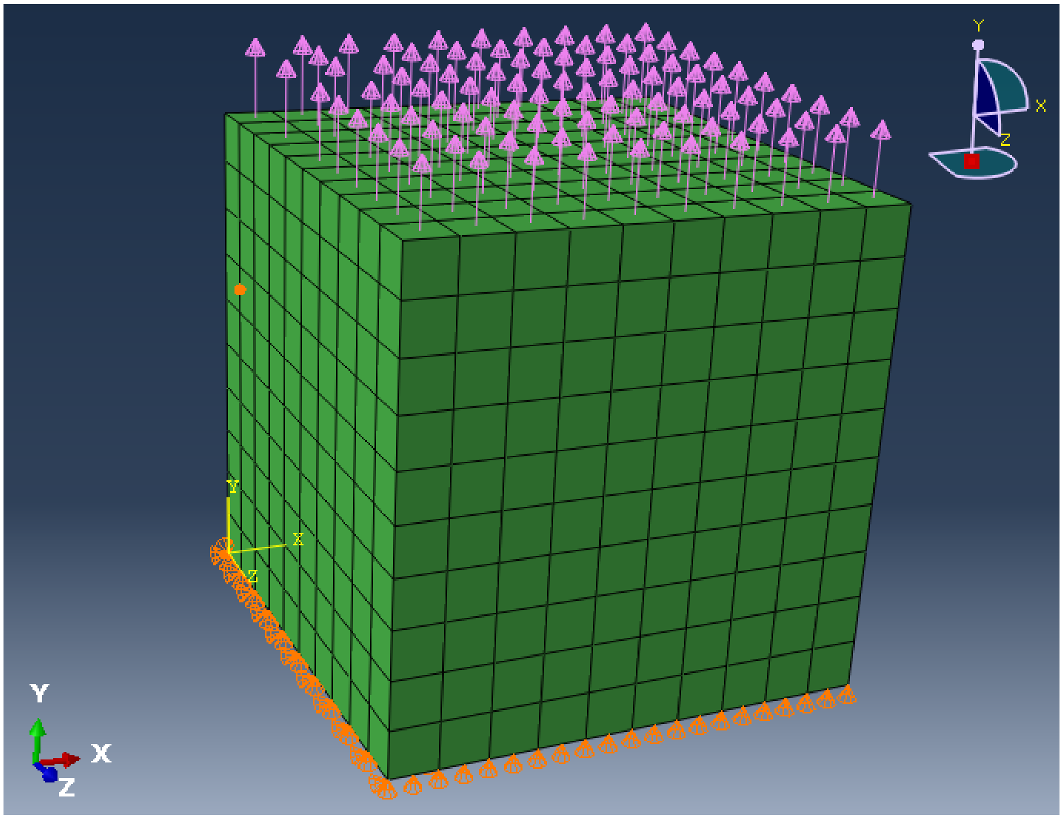


Figure 10.1 ABAQUS cube with tension on top surface.

The model is imported and divided in FRANC3D. An edge-crack is defined on the front face that is inclined 20 degrees from normal, Fig 10.2. This will produce a mix of Mode I and II SIFs for both load steps.

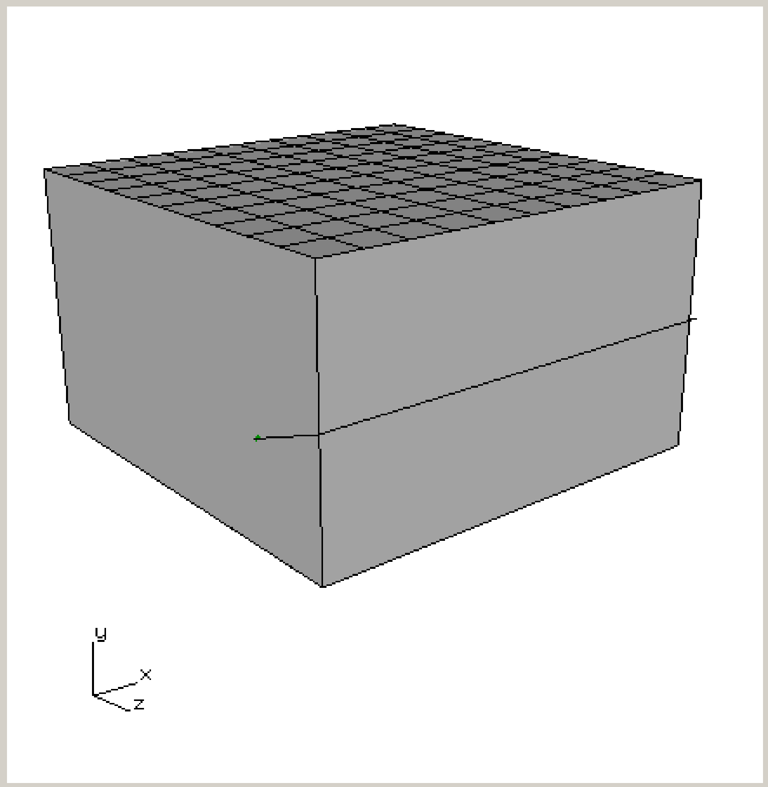
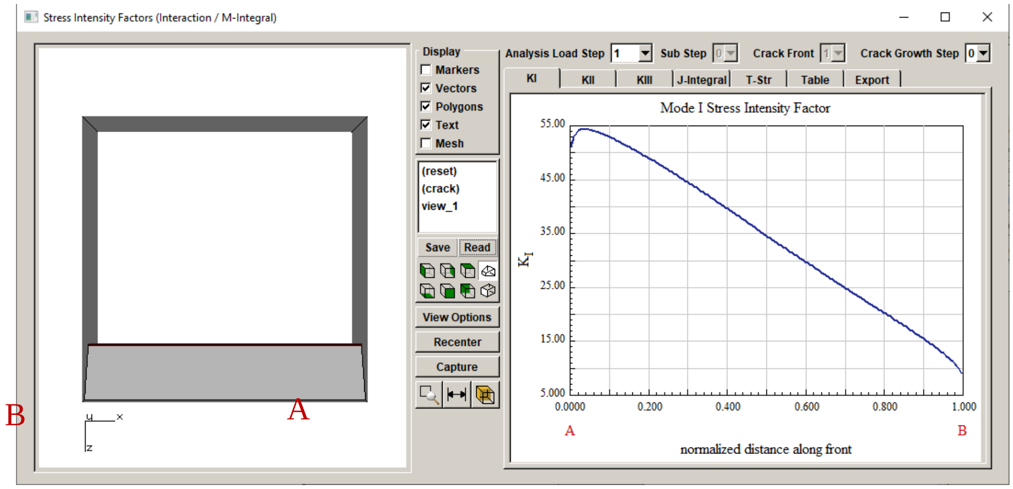


Figure 10.2 Inclined edge crack.

Once the ABAQUS analysis has finished, we examine the SIFs for the initial crack front. Mode I and II SIFs for load steps 1 and 2 are shown in Figs 10.3 – 10.6.

A fatigue load schedule is defined using simple cyclic events. We will examine three different cases: 1) load step 2 only, 2) load step 1 only, and 3) both load steps; Fig 10.7 shows the schedule for case 3). The schedule is set to repeat forever.

The Kmin is set to 0 (R=0) but could be based on another load step in which case dK would be Kmax-Kmin rather than simply Kmax.

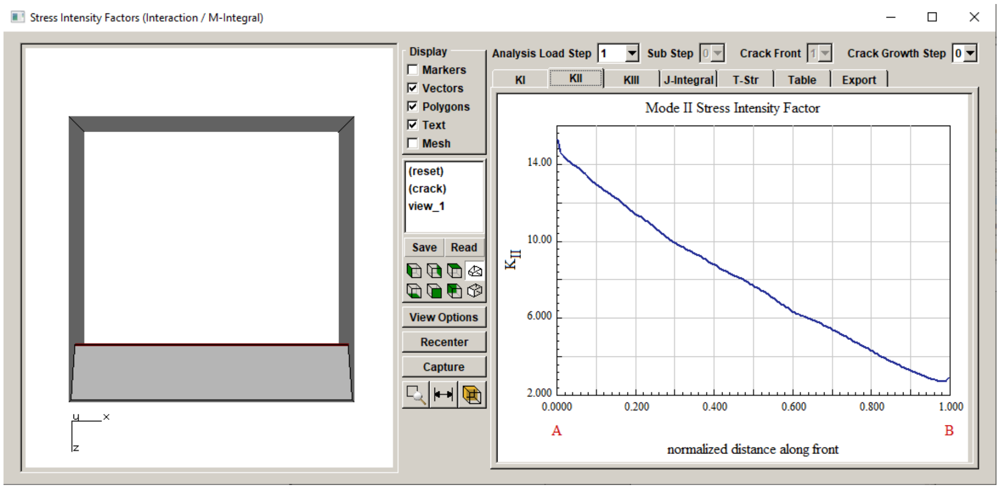


Stress Intensity Factors (Interaction / M-Integral)

Analysis Load Step 1 · Sub Step 0 · Crack Front 1 · Crack Growth Step 0

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Figure 10.3 Mode I for load step 1.

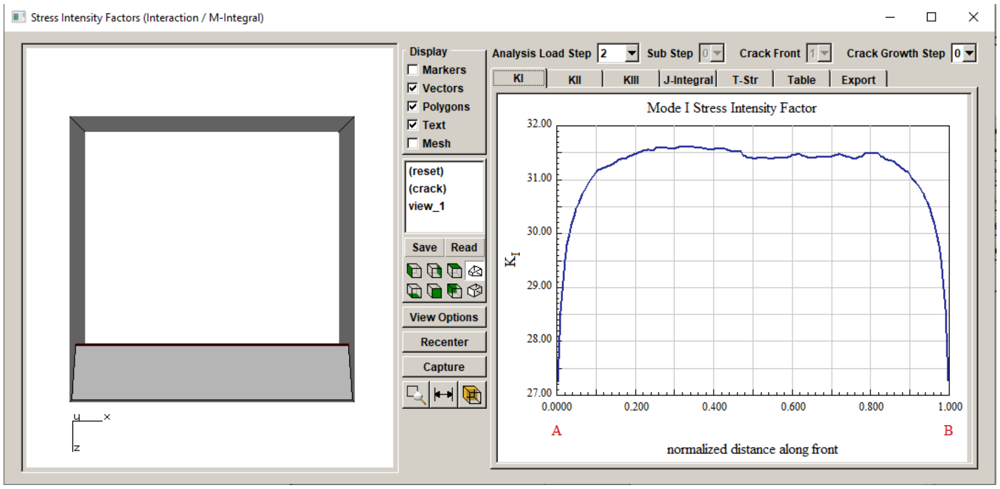


Stress Intensity Factors (Interaction / M-Integral)

Analysis Load Step 1 · Sub Step 0 · Crack Front 1 · Crack Growth Step 0

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Figure 10.4 Mode II for load step 1.

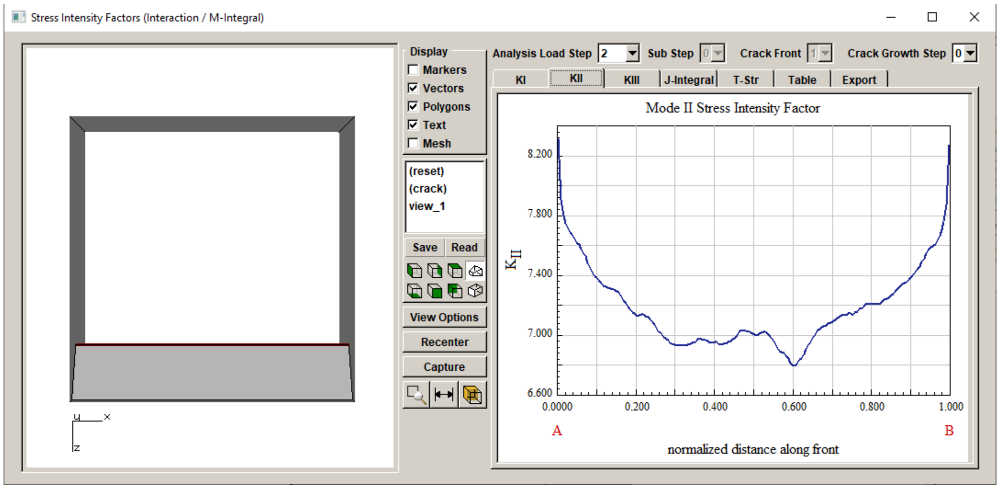


Stress Intensity Factors (Interaction / M-Integral)

Analysis Load Step 2 · Sub Step 0 · Crack Front 1 · Crack Growth Step 0

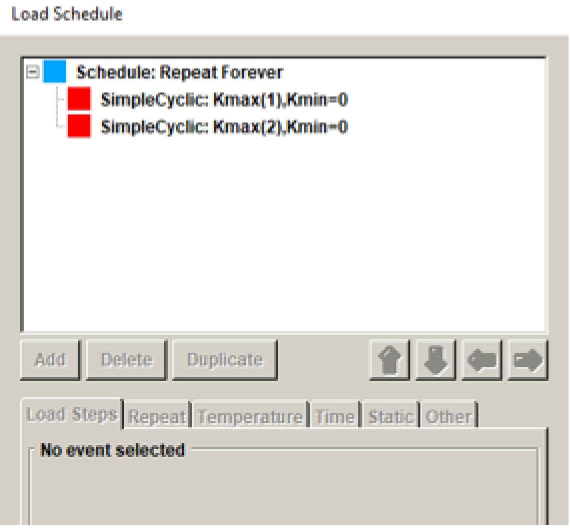
Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Figure 10.5 Mode I for load step 2.



Analysis Load Step 2
Sub Step 0
Crack Front 1
Crack Growth Step 0

Figure 10.6 Mode II for load step 2.



Schedule: Repeat Forever
SimpleCyclic: Kmax(1),Kmin=0
SimpleCyclic: Kmax(2),Kmin=0

No event selected

Figure 10.7 Load schedule for the case where both load steps 1 and 2 are used.

Load step 2 is a uniform tension of 10 MPa on the top surface. It produces relatively uniform values for both Mode I and II SIFs along the front (see Figs 10.5 and 10.6; ignoring the free-surface effects). The MTS criterion is used to compute kink angle and a median extension of 0.15 mm is specified. The resulting kink angles along the crack front are shown in Fig 10.8.

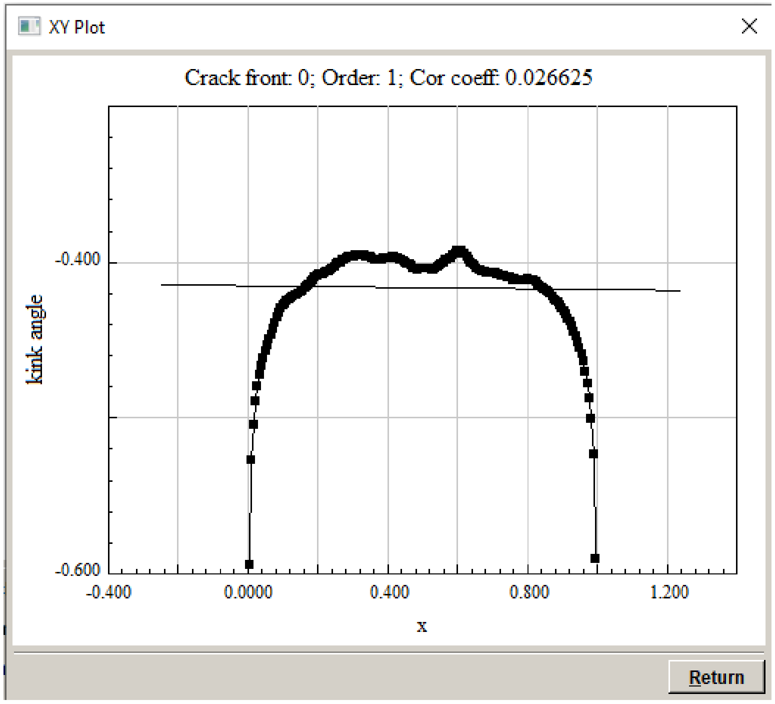
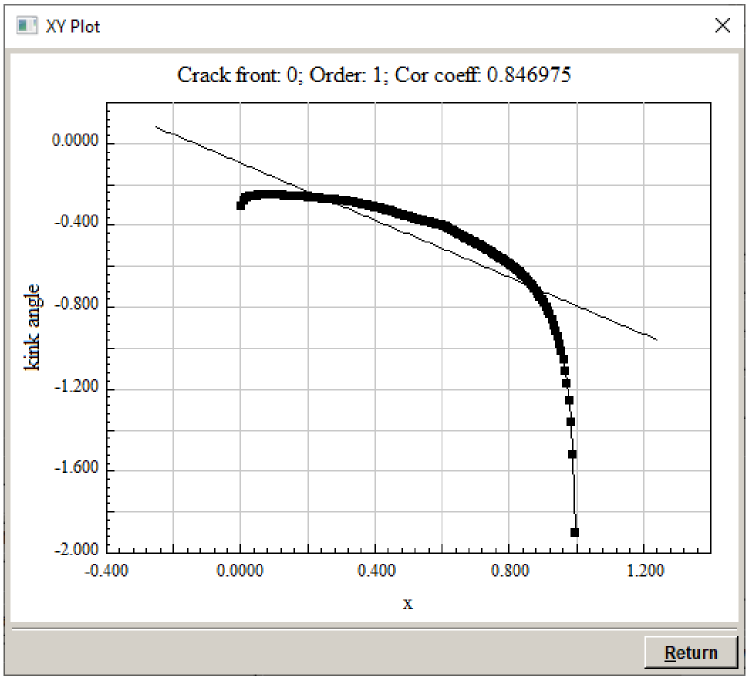


Figure 10.8 Kink angles (in radians) along the normalized crack front for load step 2.

Load step 1 has a gradient in the x-direction, varying from 2 MPa at x=0 to 20 MPa at x=10 mm. The Mode I and II SIFs reflect this gradient. The kink angle is a function of the KII / KI ratio and thus varies along the crack front, Fig 10.9. The value at the B-end is significantly higher than it was for load step 2 (103 deg vs 34 deg). The relatively large kink angle combined with the low extension can make crack growth difficult; this will be described later.

For the load schedule that combines both load steps, the kink angle along the crack front is a combination of the two previous cases, Fig 10.10.

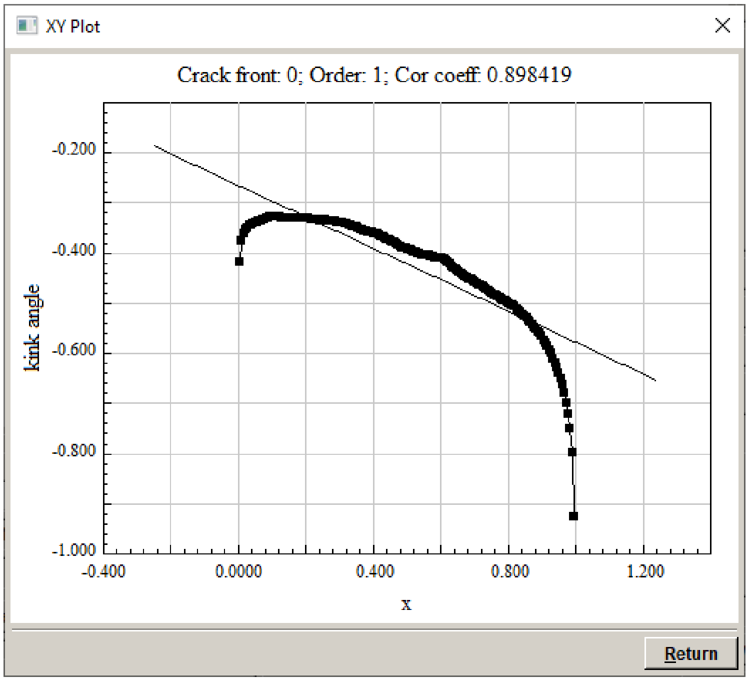
For example, if 1000 cycles are required to advance the crack to the median specified extension of 0.15 mm, 500 of those cycles will be based on load step 1 and 500 will be based on load step 2. The dK for each load step is computed at each (mid-side node) point along the crack front. The dK along with the growth rate data (da/dN vs dK) gives us the da and dN. We accumulate enough cycles (dN) to achieve da=0.15 mm at the point along the front where the median dK exists.



Crack front: 0; Order: 1; Cor coeff: 0.846975

Figure 10.9 Kink angles (in radians) along the normalized crack front for load step 1.

The kink angle is treated like extension; at each point along the front, the KII / KI ratio from each load step is computed. As we sum the extension and cycles at the point, we also sum the kink angle. Fig 10.10 shows that at the B-end, the kink angle is an intermediate value compared to Figs 10.8 and 10.9.



Crack front: 0; Order: 1; Cor coeff: 0.898419

Figure 10.10 Kink angles (in radians) along the normalized crack front for both load steps.

The extension along the front is shown in Figs 10.11 – 10.13. The trend is similar to the kink angle. The combined load schedule produces intermediate values as expected.

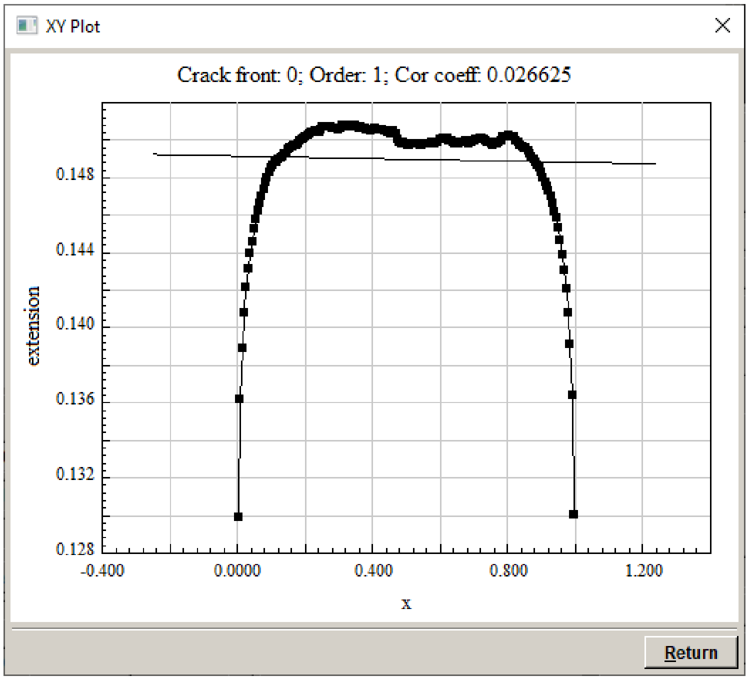


Figure 10.11 Extension (in mm) along the normalized crack front for load step 2.

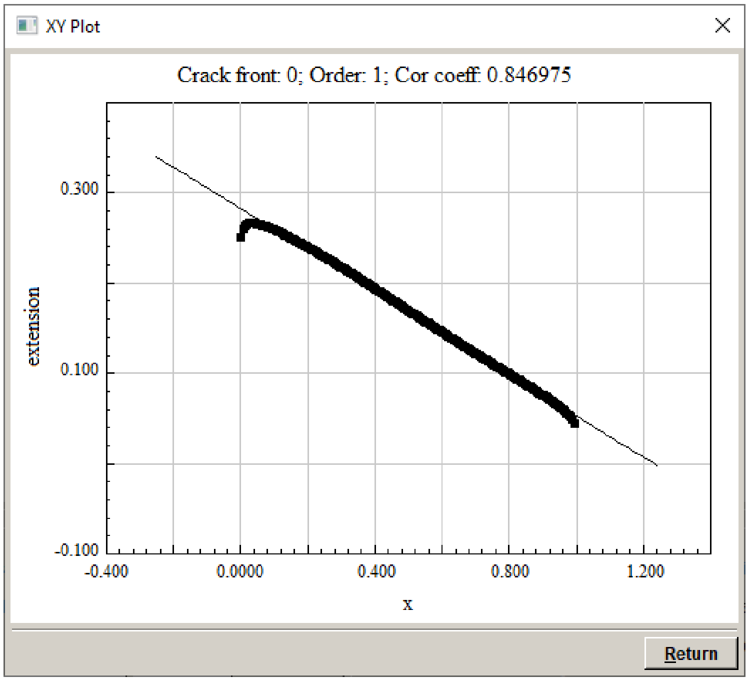


Figure 10.12 Extension (in mm) along the normalized crack front for load step 1.

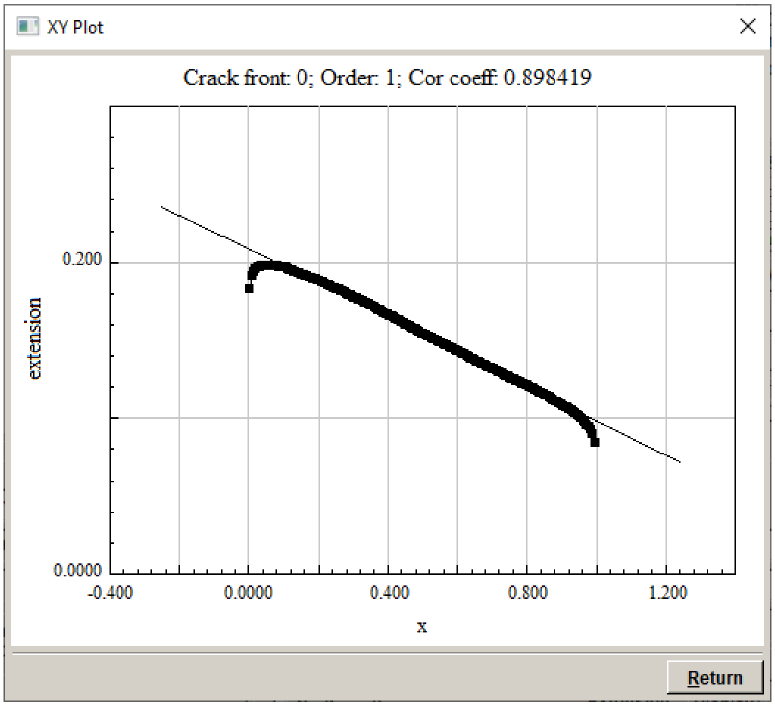


Figure 10.13 Extension (in mm) along the normalized crack front for both load steps.

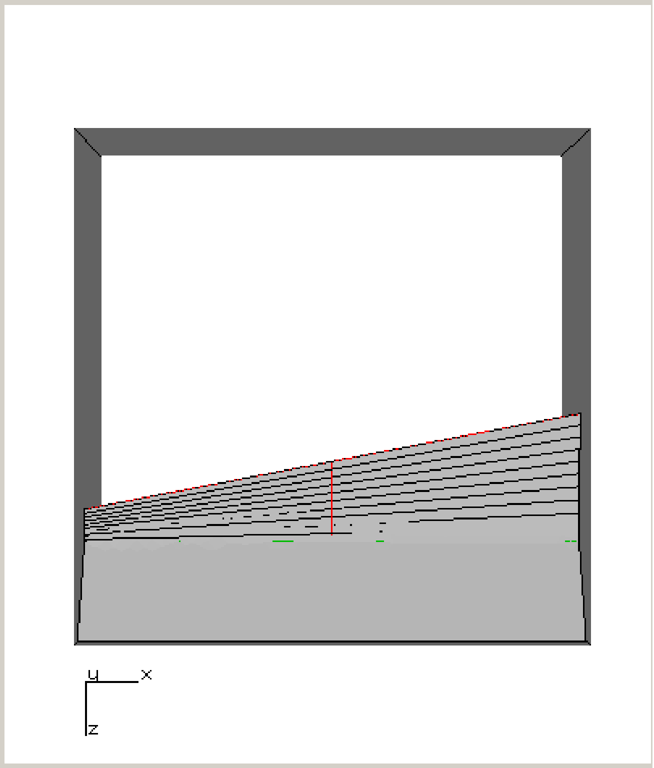


Figure 10.14 Crack fronts for 10 steps of growth using only load step 1.

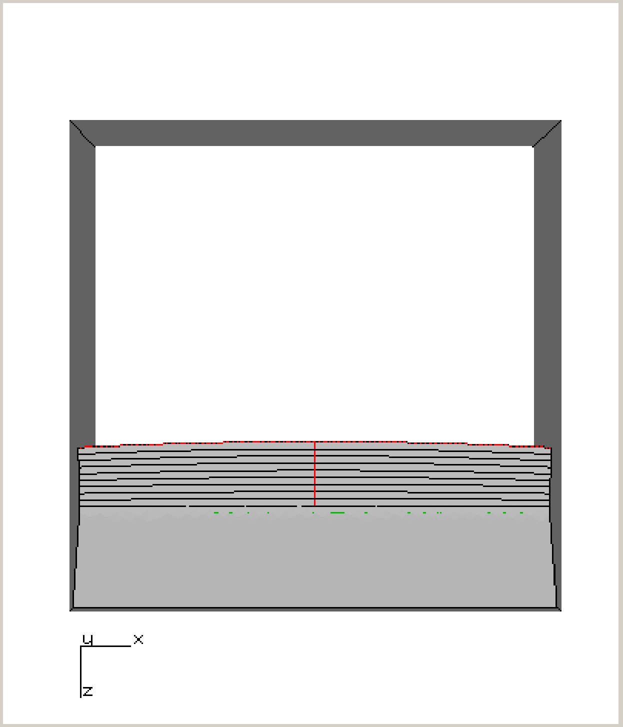


Figure 10.15 Crack fronts for 10 steps of growth using only load step 2.

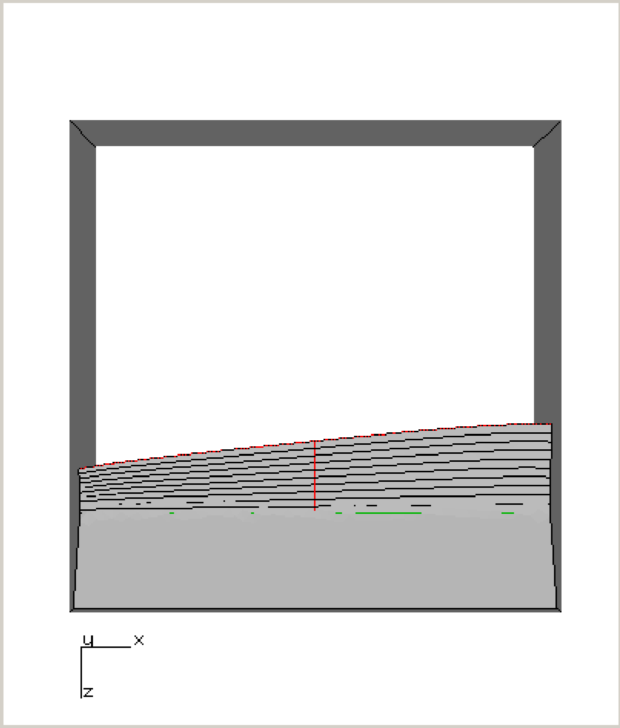


Figure 10.16 Crack fronts for 10 steps of growth using both load steps.

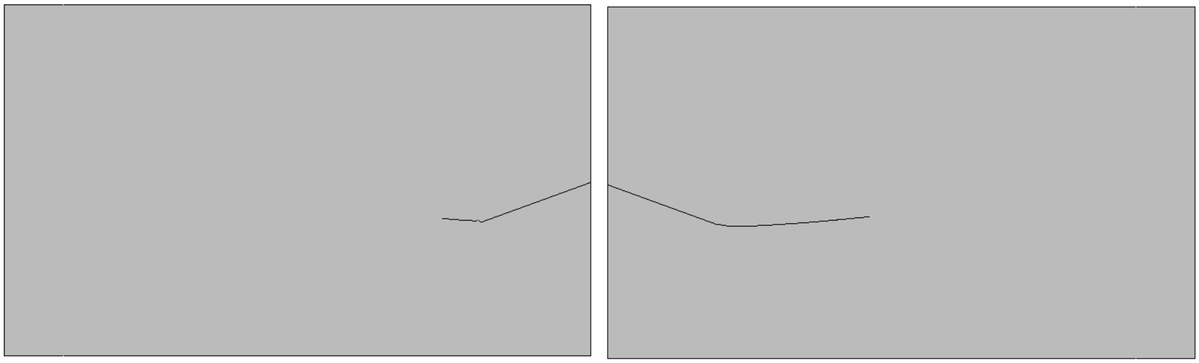


Figure 10.17 Left and right views for 10 steps of growth using only load step 1.

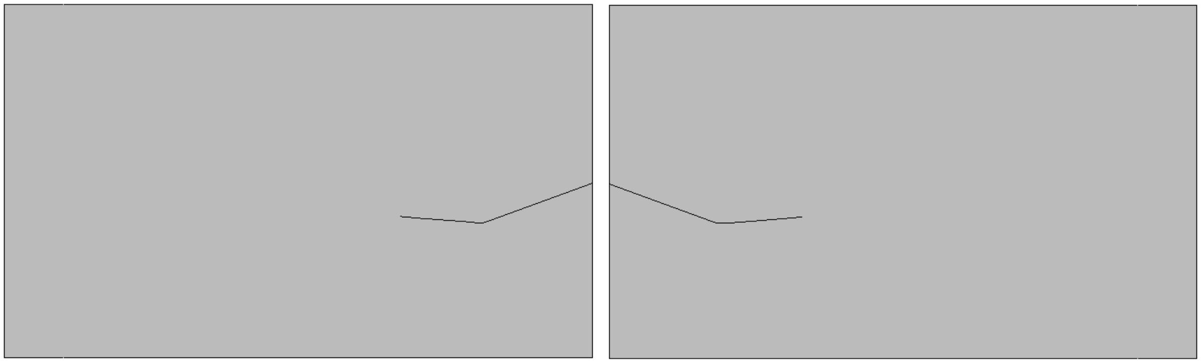


Figure 10.18 Left and right views for 10 steps of growth using only load step 2.

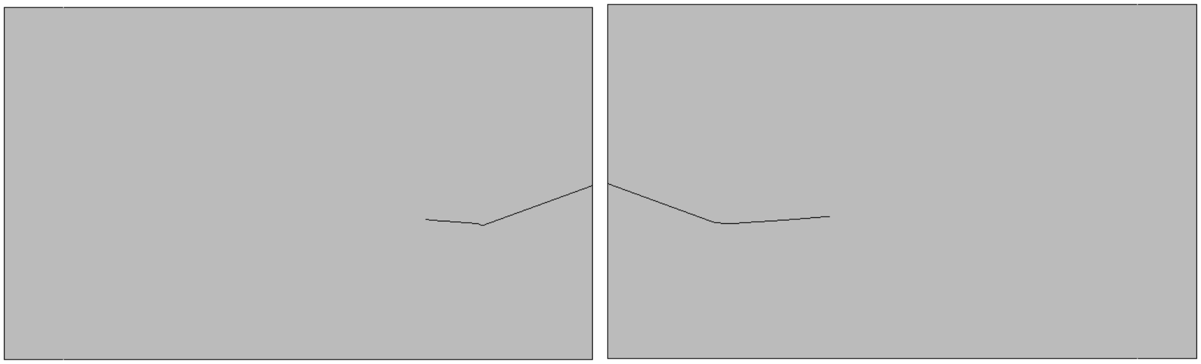
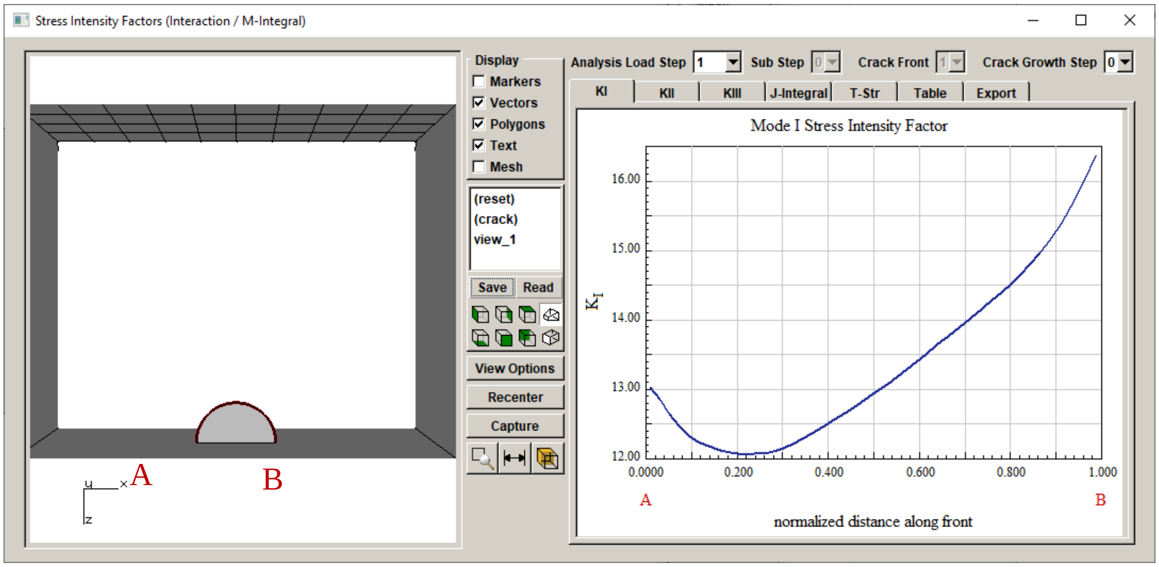


Figure 10.19 Left and right views for 10 steps of growth using both load steps.

In the above example, the extension and kink angle followed similar trends along the crack front. If we introduced a crack that was normal to the front surface, the Mode II SIFs would be essentially zero for both load cases. In this case, only the extension along the front would be affected by the choice of load steps in the schedule.

A penny-shaped surface crack (radius=1mm) is inserted into the front face of the cube and analyzed. The Mode I and II SIFs for load steps 1 and 2 are shown in Figs 10.20 – 10.23. Mode II is close to zero for both load steps.

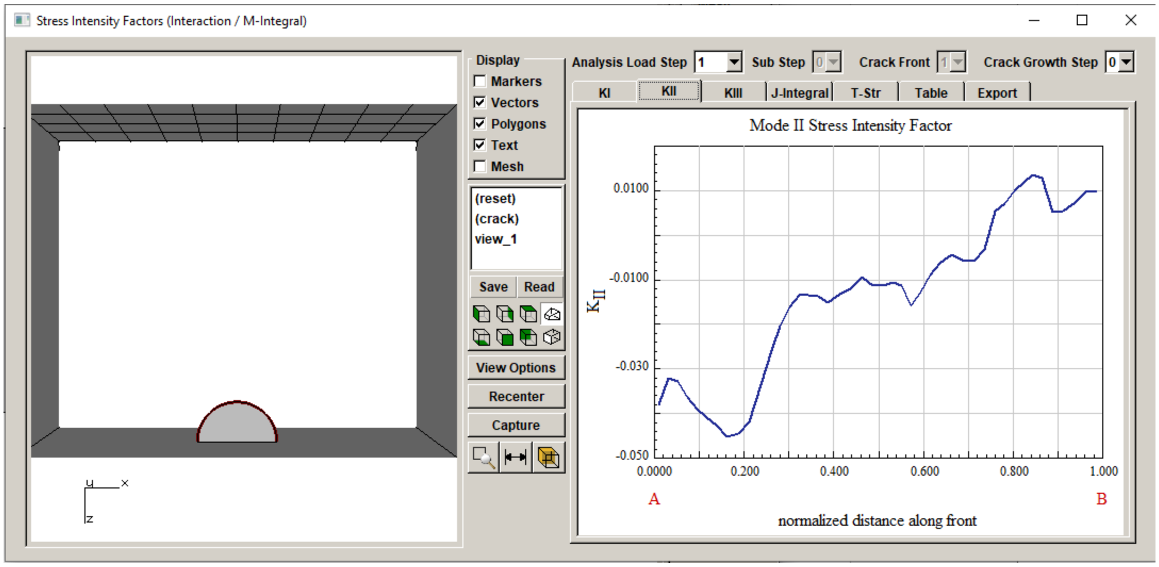


Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 10.20 Mode I for load step 1.

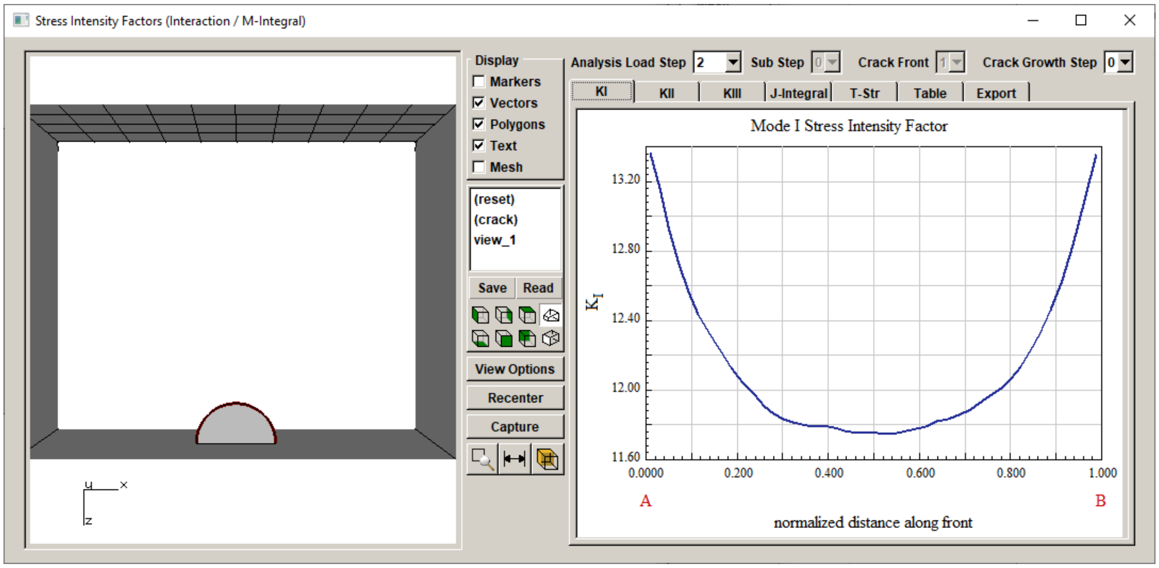


Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

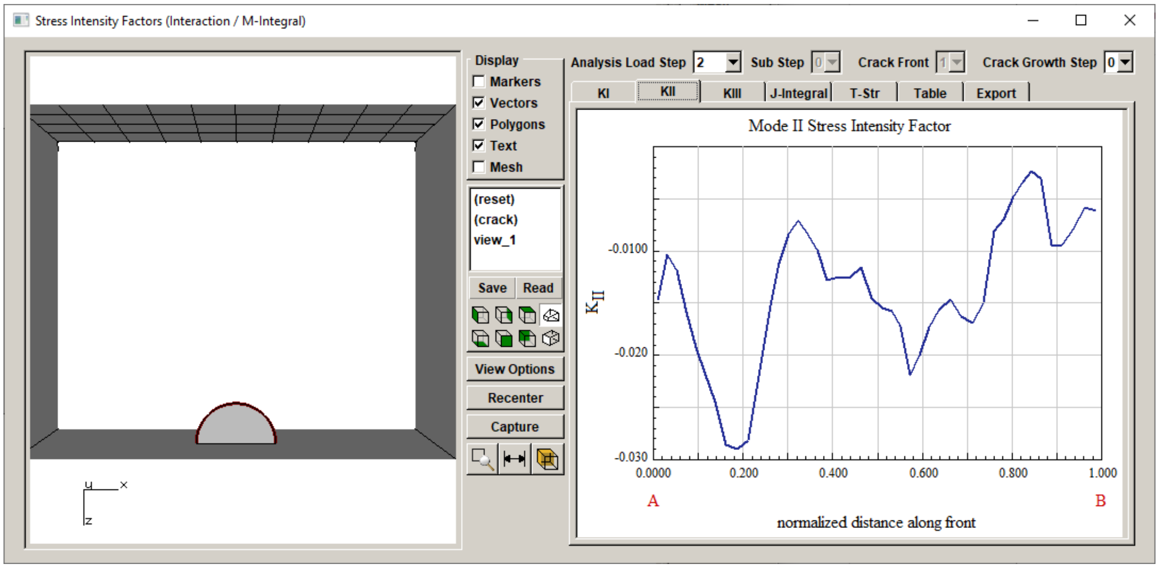
Figure 10.21 Mode II for load step 1.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 2 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 10.22 Mode I for load step 2.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 2 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 10.23 Mode II for load step 2.

If we grow the crack using just load step 2, the extension is relatively uniform along the crack front, Fig 10.24. Growth using only load step 1 is shown in Fig 10.25 and growth using both load steps is shown in Fig 10.26.

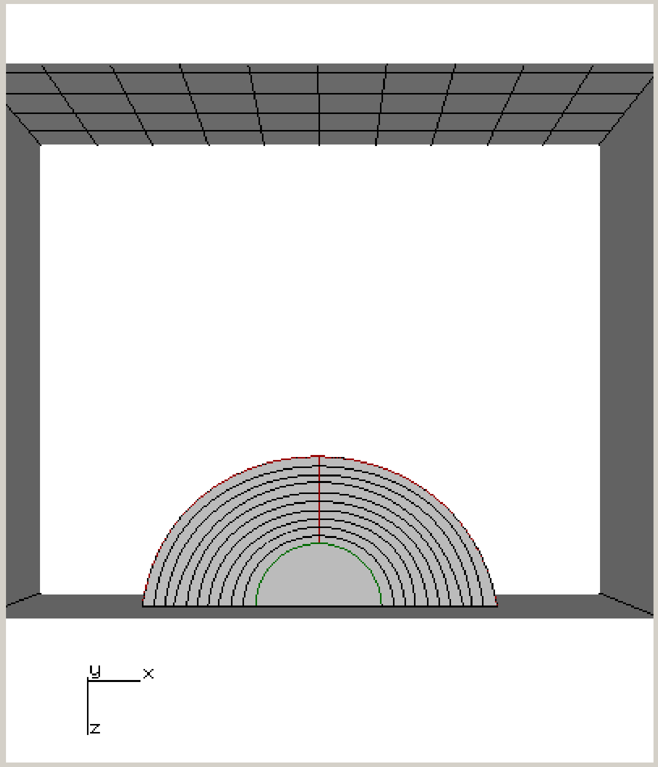


Figure 10.24 Growth using only load step 2.

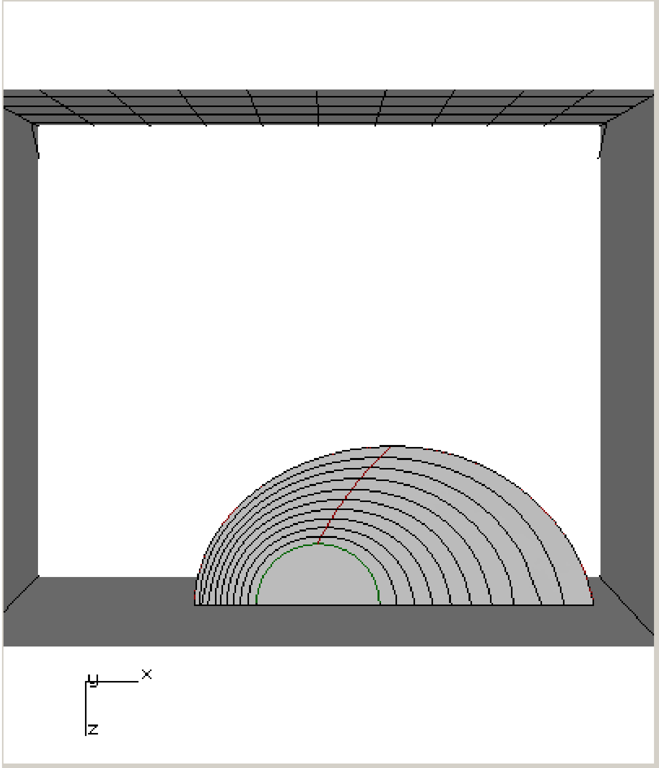


Figure 10.25 Growth using only load step 1.

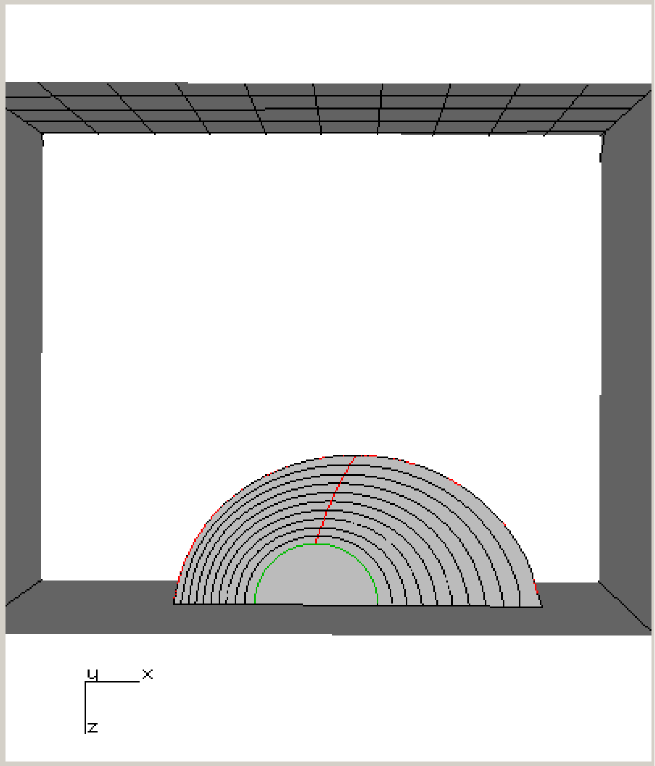
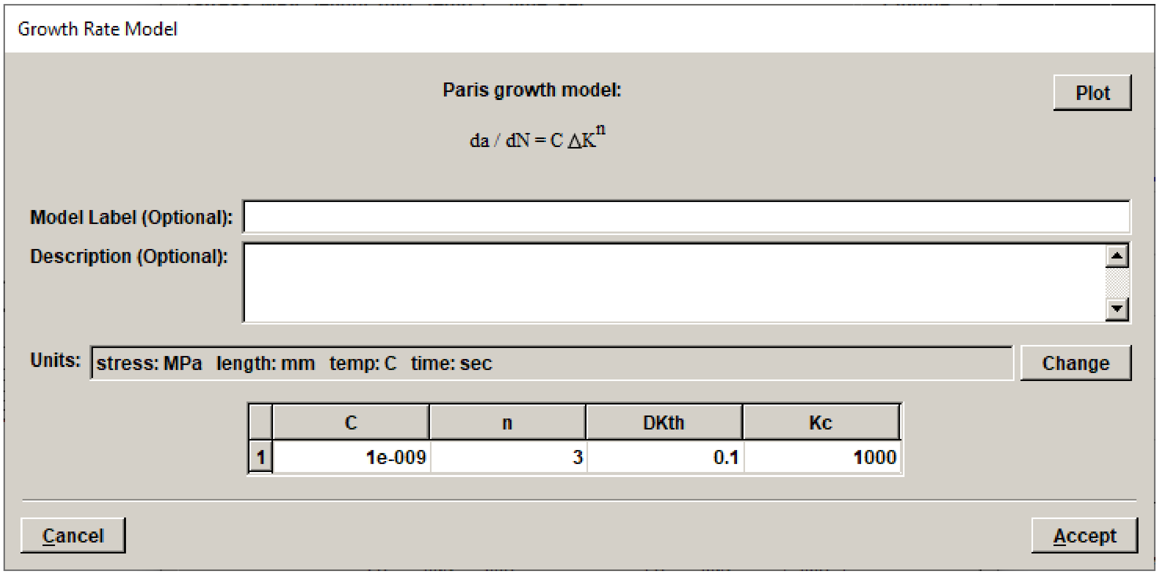


Figure 10.26 Growth using both load steps.

The computed fatigue cycles are a function of the number of load steps as well as a function of the dK and growth rate model. A simple Paris fatigue model is used with the parameters shown in Fig 10.27.

The fatigue cycles using only load step 2 are shown in Fig 10.28. The fatigue cycles using only load step 1 are shown in Fig 10.29, and the fatigue cycles using both load steps are shown in Fig 10.30. In Fig 10.30, the cycle count is the total number of cycles of both load steps. Some people prefer to count "missions" where the two load steps are part of one mission; in this case, the cycles should be divided by 2. FRANC3D refers to these as "passes".



Growth Rate Model

Paris growth model:

$$da / dN = C \, \Delta K^{n}$$

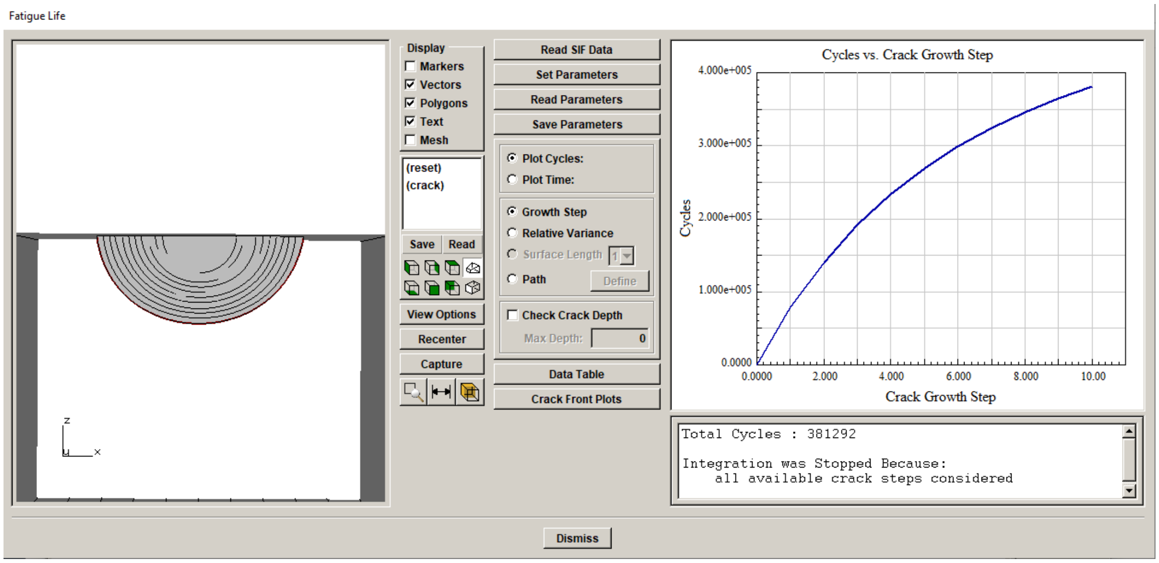
Model Label (Optional):

Description (Optional):

Units: stress: MPa length: mm temp: C time: sec

	C	n	DKth	Kc
1	1e-009	3	0.1	1000

Figure 10.27 Paris growth model parameters.



Fatigue Life

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset)
(crack)

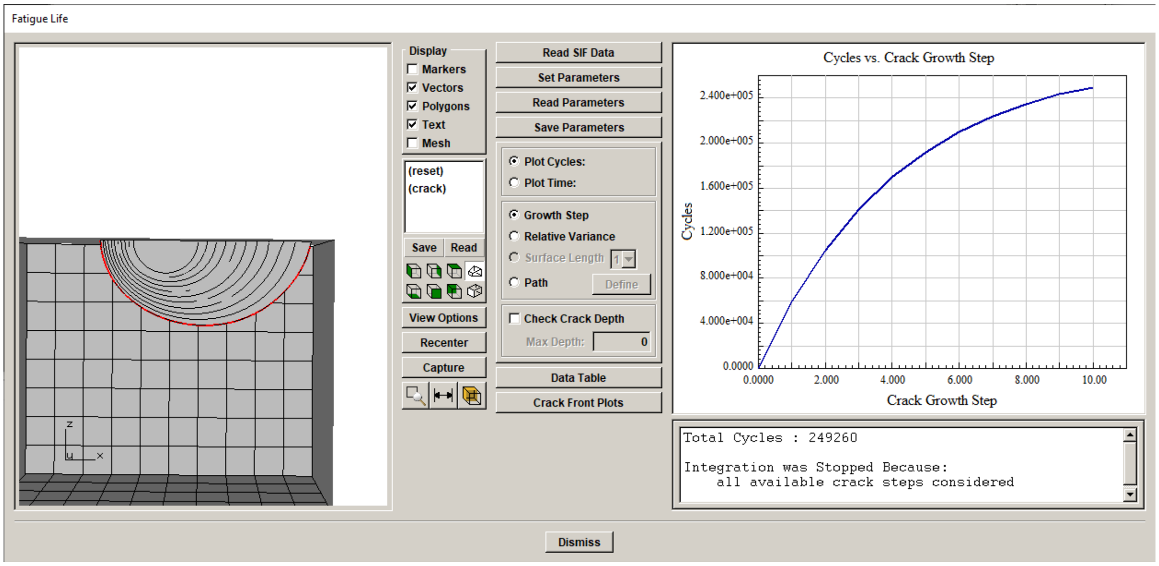
● Plot Cycles:
○ Plot Time:

● Growth Step
○ Relative Variance
○ Surface Length 1
○ Path

☐ Check Crack Depth
Max Depth: 0

```
Total Cycles : 381292

Integration was Stopped Because:
    all available crack steps considered
```

Figure 10.28 Fatigue cycles using only load step 2.



Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

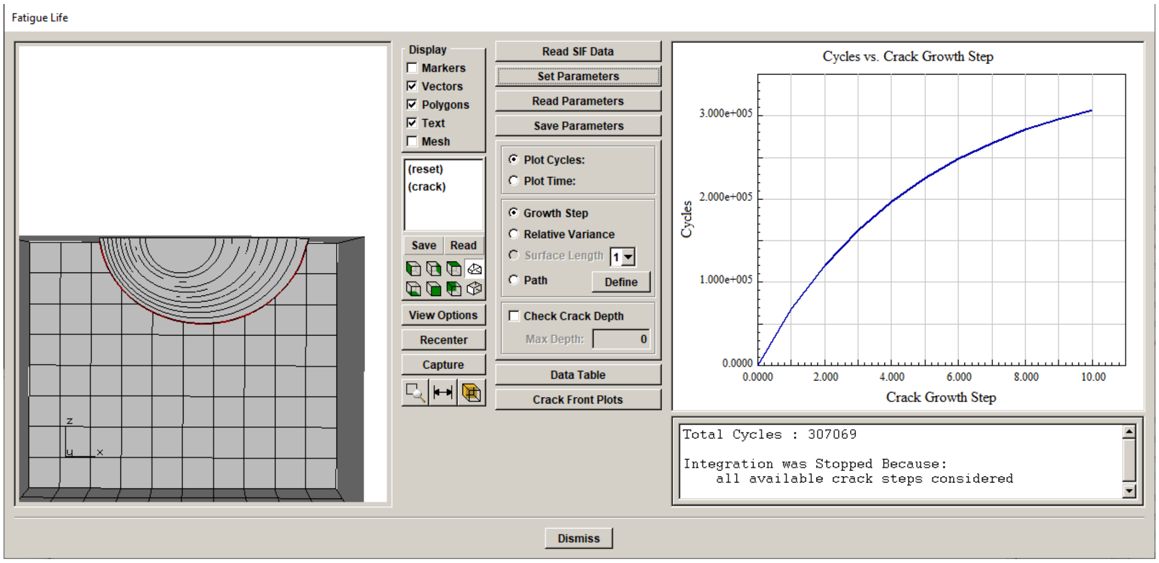
● Plot Cycles:
○ Plot Time:

● Growth Step
○ Relative Variance
○ Surface Length 1
○ Path

□ Check Crack Depth
Max Depth: 0

```
Total Cycles : 249260

Integration was Stopped Because:
    all available crack steps considered
```

Figure 10.29 Fatigue cycles using only load step 1.



Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

● Plot Cycles:
○ Plot Time:

● Growth Step
○ Relative Variance
○ Surface Length 1
○ Path

□ Check Crack Depth
Max Depth: 0

```
Total Cycles : 307069

Integration was Stopped Because:
    all available crack steps considered
```

Figure 10.30 Fatigue cycles using both load steps.

In the example above, there are only two load pairs in the schedule. If a user has a more complicated mission profile, Fig 10.31, the load schedule might include multiple load pairs, Fig 10.32. For example, the user might identify a subset of three important load pairs from the full mission. The cycle count computed by FRANC3D would be divided by 3 to get “passes”.

As for the cycle count, if there is “hold” time included, FRANC3D displays the total hold-time for all load steps. It is up to the user to decide on “units” for the time and cycles; for example, a single flight (mission) might consist of a specified number of load and time events.

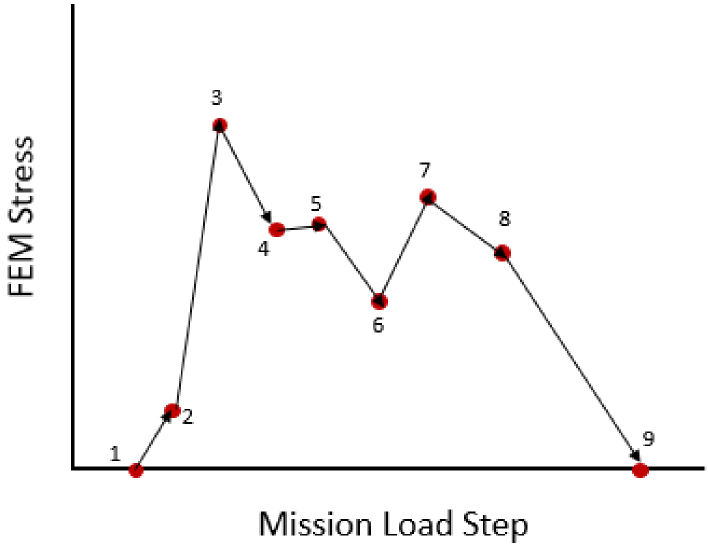
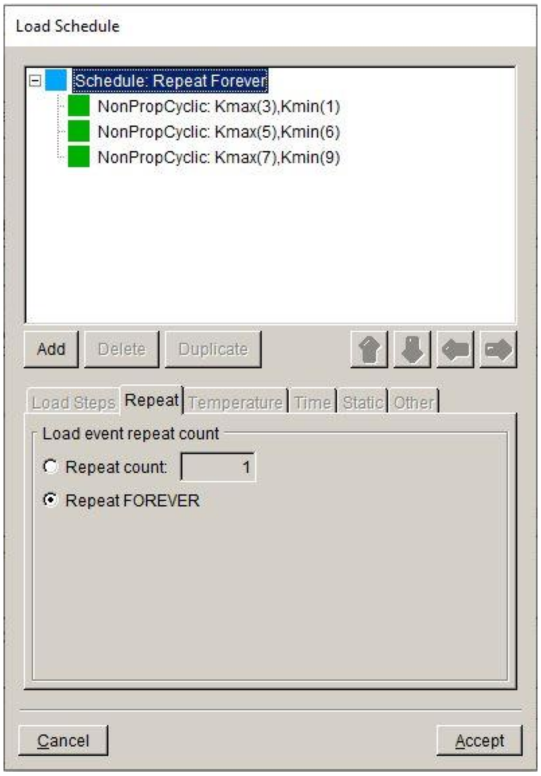


Figure 10.31. Example mission profile.


Load Schedule

Schedule: Repeat Forever

- 🟩 NonPropCyclic: Kmax(3),Kmin(1)
- 🟩 NonPropCyclic: Kmax(5),Kmin(6)
- 🟩 NonPropCyclic: Kmax(7),Kmin(9)

Load event repeat count

○ Repeat count: 1

● Repeat FOREVER

Figure 10.32. Three load pairs from the example mission profile included in the schedule.

10.2 Load Steps with Substeps

In this section, we describe how FRANC3D handles load steps that might have substeps. We start with a simple cube model in ABAQUS with three load steps. The bottom is constrained to prevent rigid body motion, Fig 10.33. The first load step includes a uniform tension on the top surface as shown in Fig 10.33. The second and third load steps include a strip of tension on the top left side, Fig 10.34, and a point force at the center of the top surface, Fig 10.34, respectively.

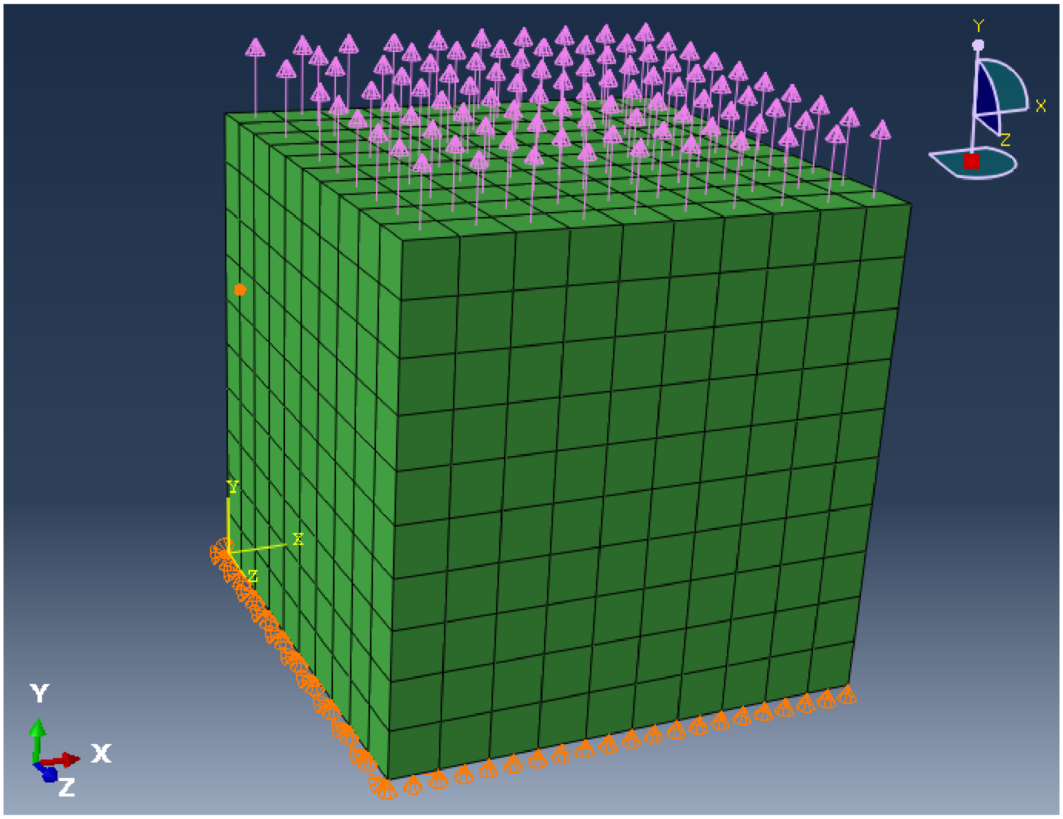


Figure 10.33 ABAQUS cube with tension on top surface.

The ABAQUS (static) load steps are defined to generate substeps during the analysis. This could be considered a simplified form of a transient analysis. The ABAQUS results will include displacements for each substep of each load step, Fig 10.35.

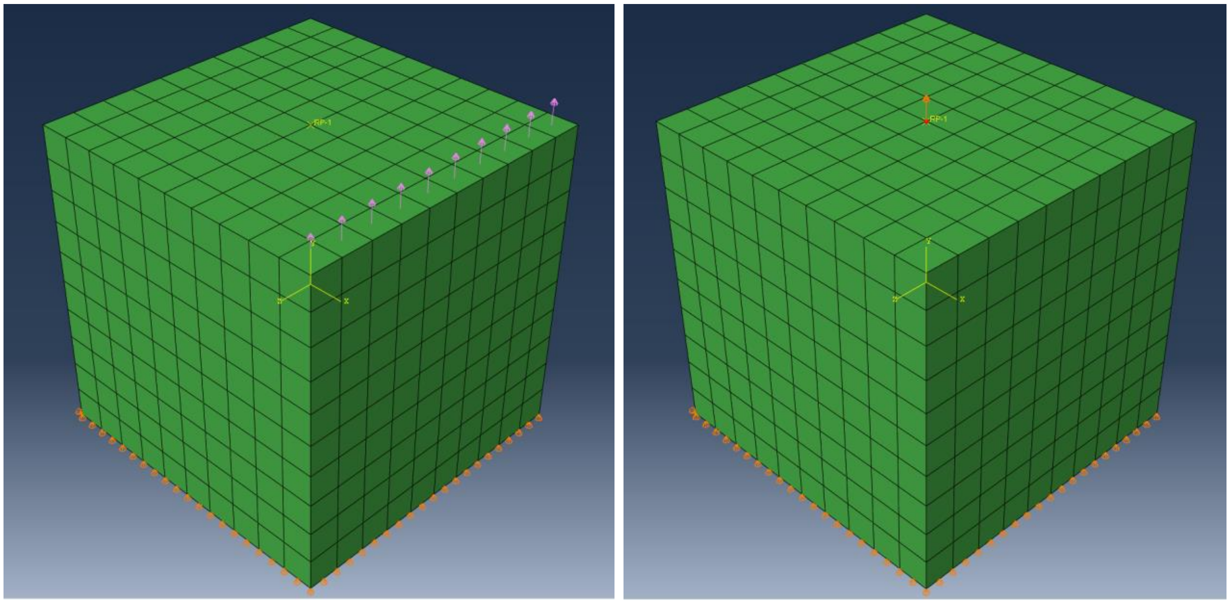
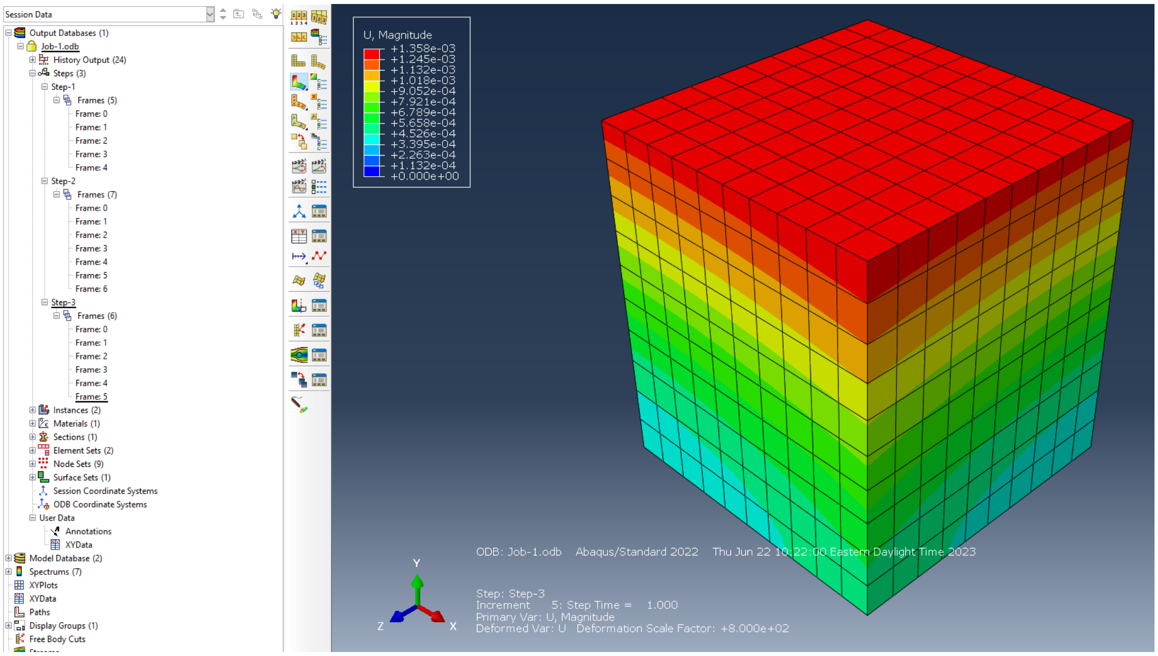


Figure 10.34 ABAQUS cube with load steps 2 and 3.



ODB: Job-1.odb   Abaqus/Standard 2022   Thu Jun 22 10:22:00 Eastern Daylight Time 2023

```
Step: Step-3
Increment      5: Step Time =    1.000
Primary Var: U, Magnitude
Deformed Var: U   Deformation Scale Factor: +8.000e+02
```

Figure 10.35 ABAQUS cube results ‘frames’ - multiple load steps and substeps.

The ABAQUS model is imported into FRANC3D, and a half-penny surface crack is defined on the front face, Fig 10.36.

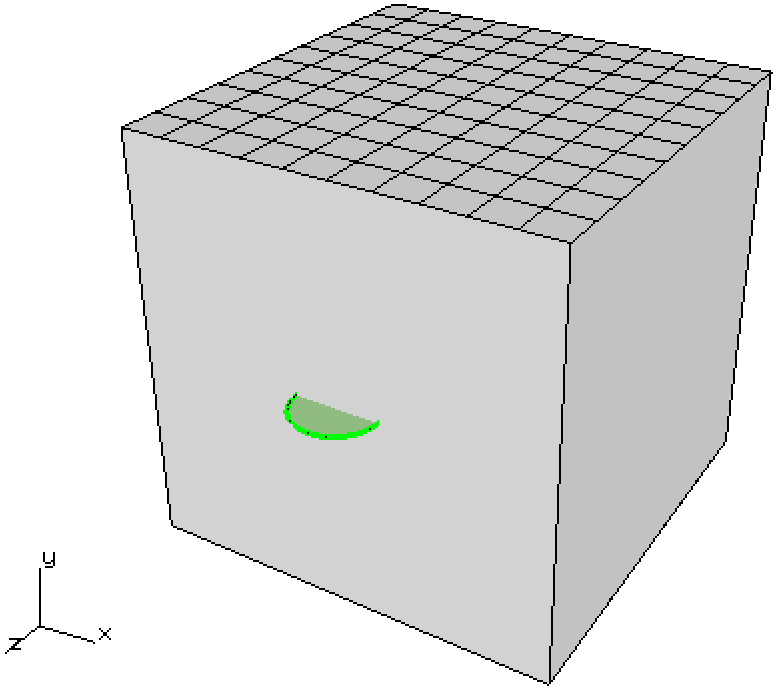
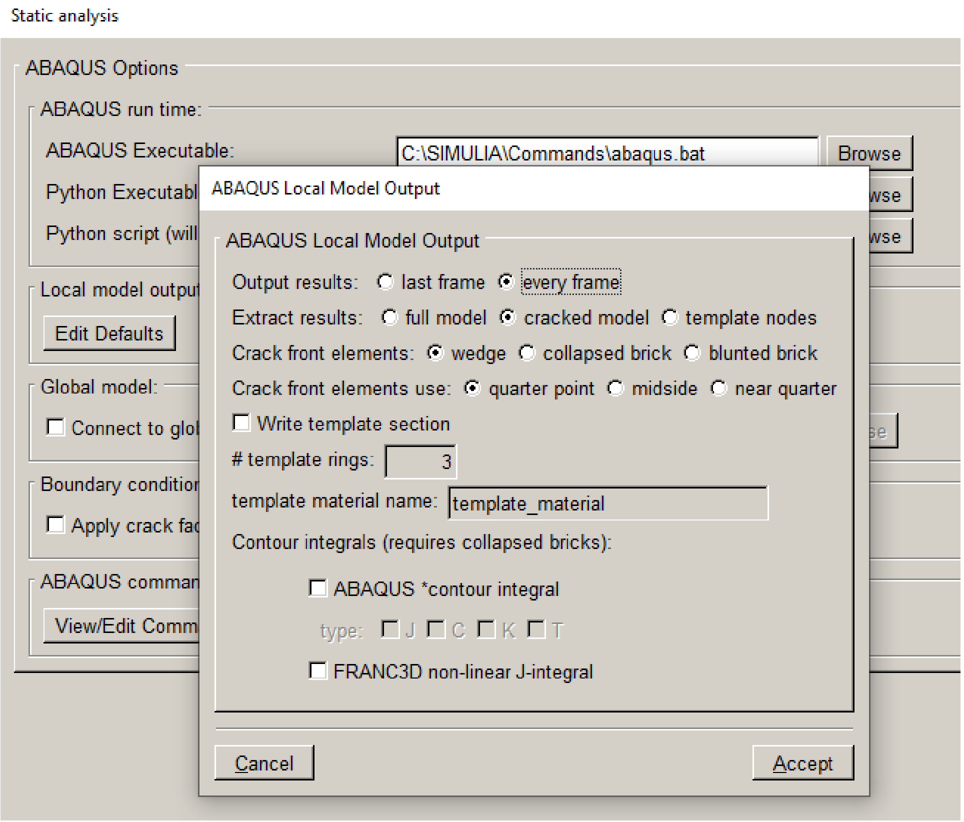


Figure 10.36 Half-penny surface crack in front face of a cube.

Once the crack is inserted, an ABAQUS static crack analysis can be performed. The FRANC3D/ABAQUS analysis defaults are edited so that results are output for every frame, Fig 10.37. The results (*.dtp*) file that is generated by ABAQUS for FRANC3D will have displacements for all substeps of all load steps. The *.dtp* file can be opened in an editor; it should have data like:

```
NUM STEP 3
LOADSTEP 1
SUBSTEP 1
TIME 0.20000000298
DISPLACEMENT
1 -0.000592763477471 0.00202108267695 -0.000624222855549
11 -0.000592737516854 -9.88858435164e-38 -0.000608731235843
…
SUBSTEP 2
TIME 0.40000000596
DISPLACEMENT
1 -0.00118552695494 0.00404216535389 -0.0012484457111
…
```



Static analysis

ABAQUS Local Model Output

Output results: ○ last frame ● every frame

Extract results: ○ full model ● cracked model ○ template nodes

Crack front elements: ● wedge ○ collapsed brick ○ blunted brick

Crack front elements use: ● quarter point ○ midside ○ near quarter

□ Write template section

template rings: 3

template material name: template_material

Contour integrals (requires collapsed bricks):

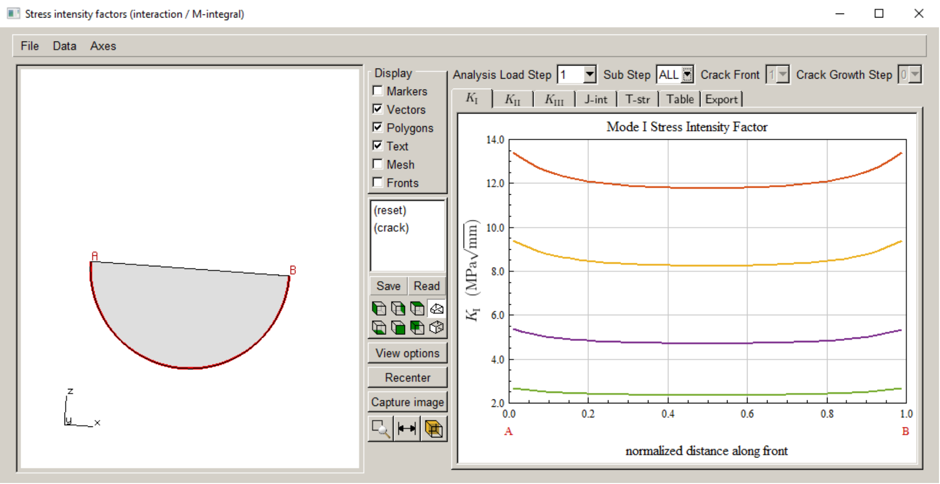
□ ABAQUS *contour integral

type: □ J □ C □ K □ T

□ FRANC3D non-linear J-integral

Figure 10.37 ABAQUS analysis defaults with results output set to every frame.

Once the analysis is finished and the *.dtp* file is imported into FRANC3D, the SIFs can be computed. FRANC3D computes SIFs for each substep of each load step. Fig 10.38 shows the Mode I SIFs for load step #1 for ALL substeps. Similar plots can be displayed for load steps 2 and 3, Figs 10.39-40.

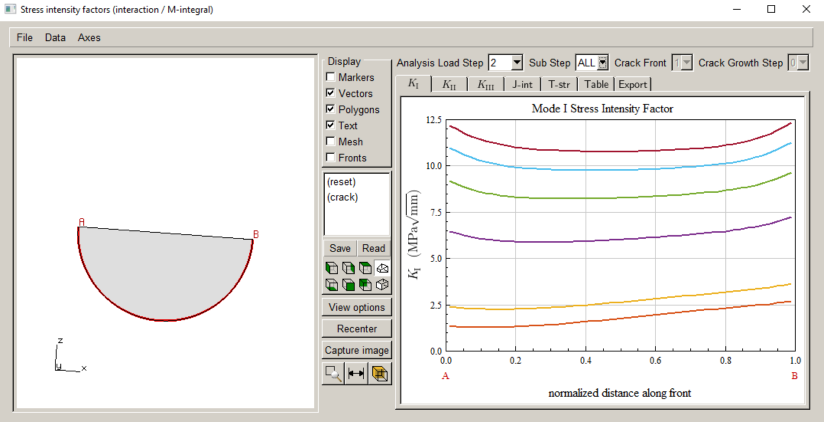


Stress intensity factors (interaction / M-integral)

Display: □ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh □ Fronts

Analysis Load Step 1 Sub Step ALL Crack Front 1 Crack Growth Step 0

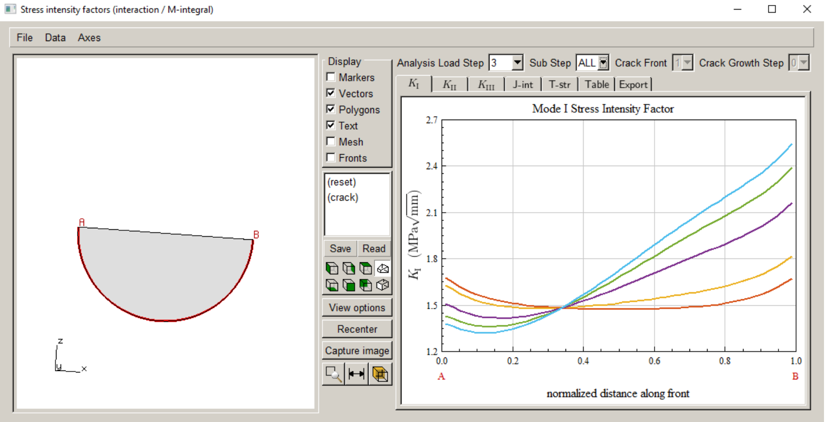
Figure 10.38 Mode I SIFs for all substeps of load step 1.


Stress intensity factors (interaction / M-integral)

Analysis Load Step 2 Sub Step ALL Crack Front 1 Crack Growth Step 0

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh ☐ Fronts

Figure 10.39 Mode I SIFs for all substeps of load step 2.


Stress intensity factors (interaction / M-integral)

Analysis Load Step 3 Sub Step ALL Crack Front 1 Crack Growth Step 0

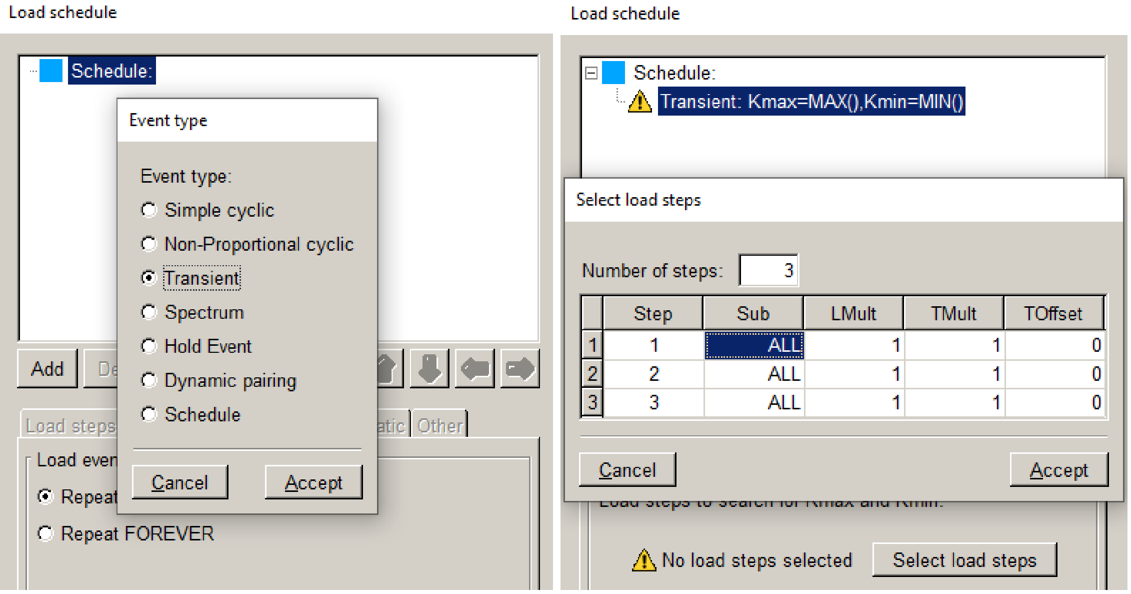
Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh ☐ Fronts

Figure 10.40 Mode I SIFs for all substeps of load step 3.

Once the SIFs are computed, they can be used to define the crack growth. A load schedule must be defined, and it might use some or all the SIFs. For example, one might choose the transient event type and then select all load steps and all substeps, Fig 10.41. In this case, FRANC3D will look through all SIFs to find K_{max} and K_{min}.

As the crack grows, the ABAQUS analysis could potentially produce different numbers of substeps for a load step. This can happen, for example, if one uses ABAQUS automatic time-stepping.

A transient event schedule that uses ALL substeps will not be affected. However, if a specific substep is chosen, Fig 10.42, and this substep does not exist for each crack step, the crack growth and/or the fatigue cycle counting can be affected; Section 10.3 describes an example of this.



Event type

Event type:
○ Simple cyclic
○ Non-Proportional cyclic
● Transient
○ Spectrum
○ Hold Event
○ Dynamic pairing
○ Schedule

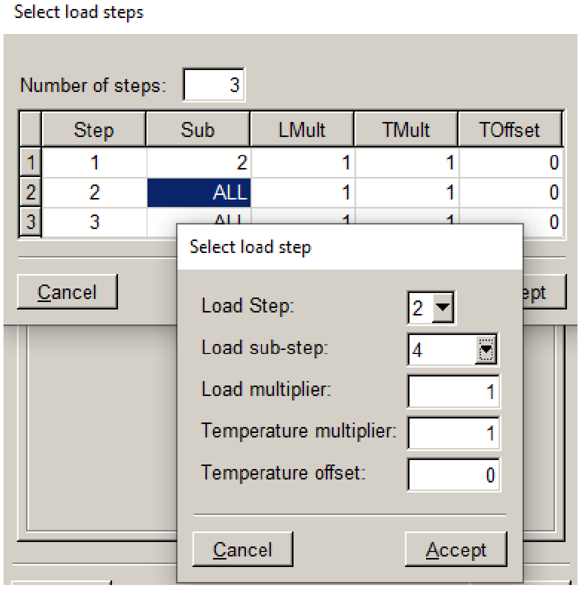
Schedule:
Transient: Kmax=MAX(),Kmin=MIN()

Select load steps

Number of steps: 3

	Step	Sub	LMult	TMult	TOffset
1	1	ALL	1	1	0
2	2	ALL	1	1	0
3	3	ALL	1	1	0

Figure 10.41 Transient load schedule event using all load steps and substeps.



Select load steps

Number of steps: 3

	Step	Sub	LMult	TMult	TOffset
1	1	2	1	1	0
2	2	ALL	1	1	0
3	3	ALL	1	1	0

Select load step

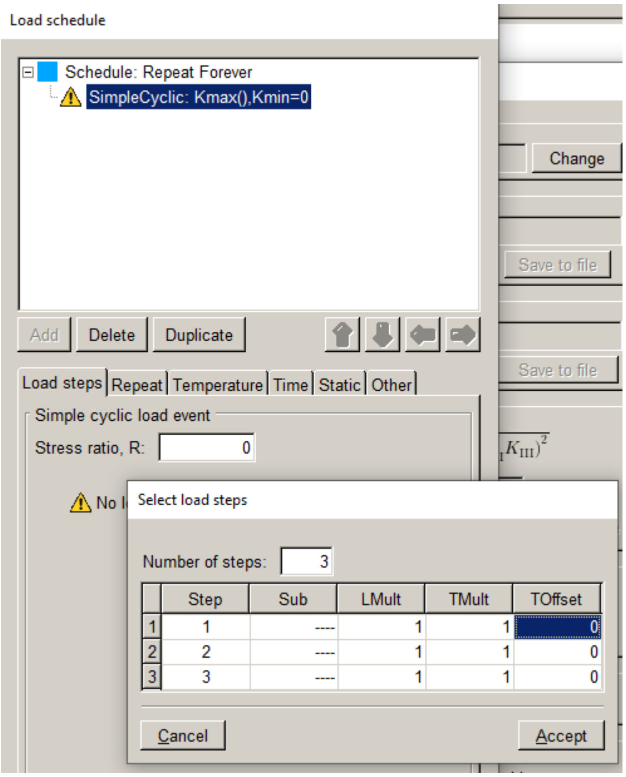
Load Step: 2
Load sub-step: 4
Load multiplier: 1
Temperature multiplier: 1
Temperature offset: 0

Figure 10.42 Choosing specific substeps for a load step.

10.2.1 Static Crack Analysis versus Crack Growth Analysis

In the above example, a static crack analysis was performed on the initial crack prior to defining the load schedule. Thus, FRANC3D has read the results from the *.dtp* file and knows about the substeps.

If the user does not do a static analysis, FRANC3D will not have any knowledge of the substeps. If we try to do an automatic crack growth analysis immediately after inserting the crack, FRANC3D will present the fatigue load schedule dialogs with only the load step identified, Fig 10.43. The substep is represented as "---" indicating that substeps are undefined.



Schedule: Repeat Forever
SimpleCyclic: Kmax(),Kmin=0

Simple cyclic load event
Stress ratio, R: 0

Number of steps: 3

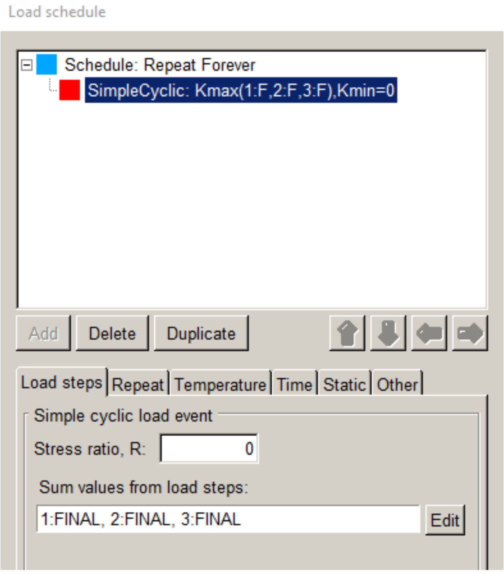
	Step	Sub	LMult	TMult	TOffset
1	1	----	1	1	0
2	2	----	1	1	0
3	3	----	1	1	0

Figure 10.43 Select load steps dialog – Sub Step ID is undefined.

It is generally recommended that one does a static crack analysis after inserting the crack as this will allow the user to ensure that the cracked model results are consistent with the uncracked model results as well as providing the load step substep SIFs. Tutorial #1 describes how to run both static crack and automatic crack growth analyses.

10.3 Load Steps with Missing Substeps

Using the model from the previous section, six steps of crack growth are completed. A simple load schedule is defined that uses the sum of the final frame of all three load steps, Fig 10.44. The cycle count for this schedule is shown in Fig 10.45.



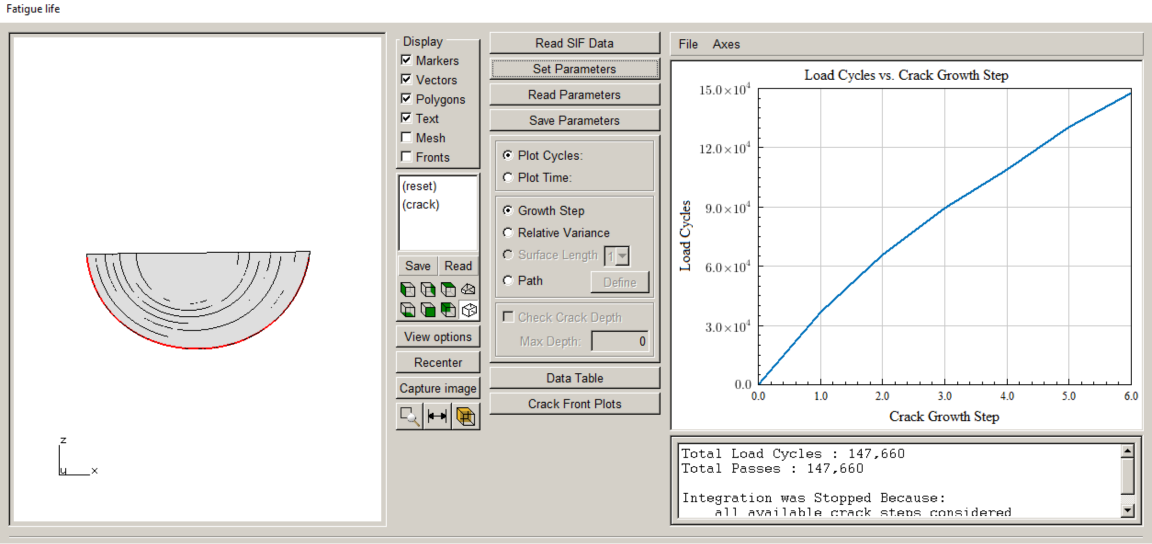
Schedule: Repeat Forever
SimpleCyclic: Kmax(1:F,2:F,3:F),Kmin=0

Simple cyclic load event

Stress ratio, R: 0

Sum values from load steps: 1:FINAL, 2:FINAL, 3:FINAL

Figure 10.44 Simple cyclic load event using the sum of the final frames.



Display: ☑ Markers, ☑ Vectors, ☑ Polygons, ☑ Text, □ Mesh, □ Fronts

● Plot Cycles: ○ Plot Time:

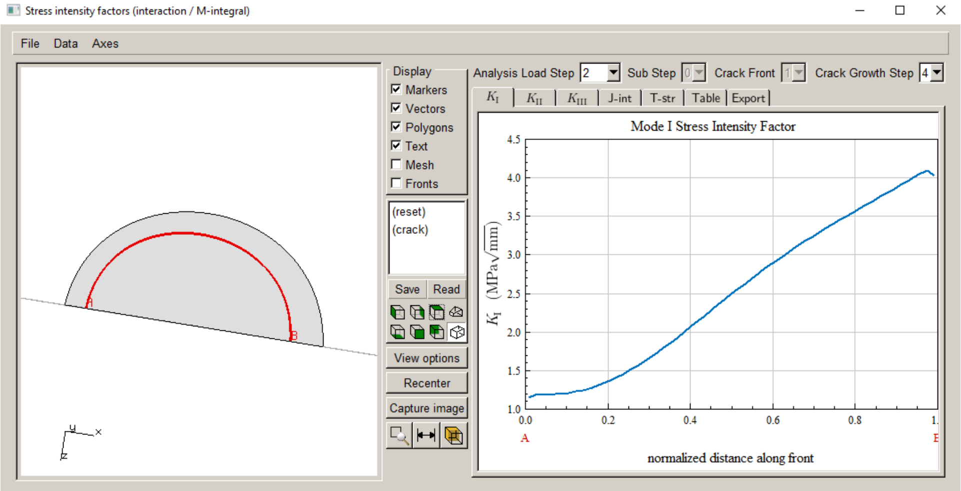
● Growth Step ○ Relative Variance ○ Surface Length ○ Path

□ Check Crack Depth Max Depth: 0

```
Total Load Cycles : 147,660
Total Passes : 147,660

Integration was Stopped Because:
    all available crack steps considered
```

Figure 10.45 Cycle count for simple cyclic load event using the sum of the final frames.

For this same model, at crack step #4, if the user accidentally turns off the option to output all frames from ABAQUS (see Fig 10.37), the cycle counting will be affected. Fig 10.46 shows the SIF plot for crack step 4 and load step 2; note that the Sub Step dropdown is greyed-out as there are only SIFs for the final substep. Fig 10.47 shows the same plot for crack step 3, where the SIFs for all substeps are displayed.

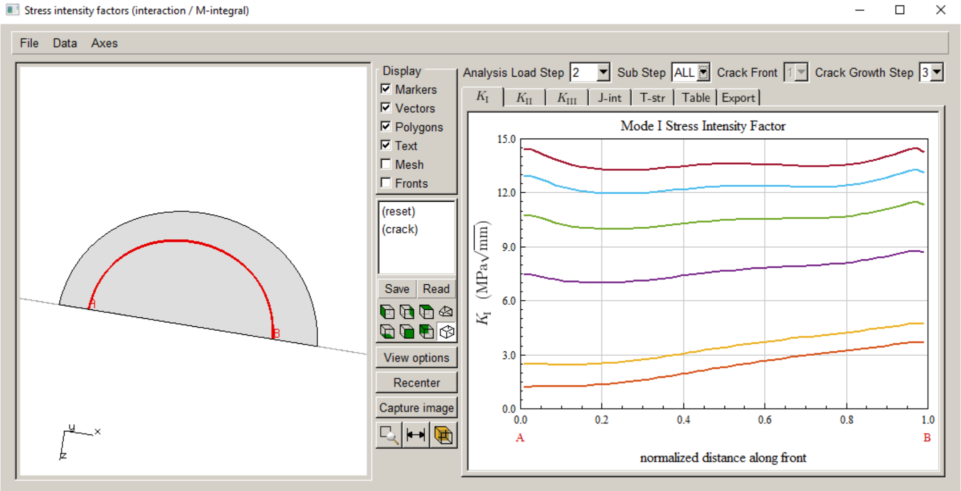


Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

(reset)
(crack)

Analysis Load Step 2
Sub Step 0
Crack Front 1
Crack Growth Step 4

Figure 10.46 SIF plot for crack step 4 and load step 2.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh
□ Fronts

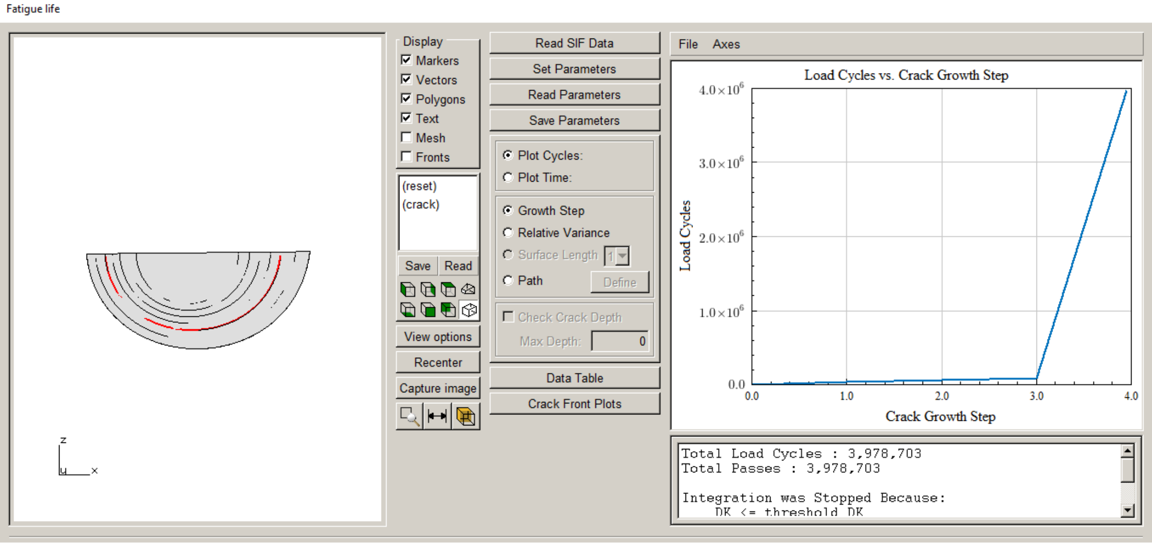
(reset)
(crack)

Analysis Load Step 2
Sub Step ALL
Crack Front 1
Crack Growth Step 3

Figure 10.47 SIF plot for crack step 3 and load step 2 for all substeps.

Using the same schedule as before (see Fig 10.44), the cycle count is shown in Fig 10.48. The cycle counting stops at crack step 4 and indicates that the ΔK is below threshold. This is not true, but FRANC3D recognizes that the load steps and substeps are inconsistent at step 4 and stops the cycle counting.

This example is constructed to demonstrate potential issues if one is not careful. This type of inconsistent cycle counting might only show up for some load schedules and events. Even though the load schedule uses only the FINAL substep SIFs, the cycle counting stops due to the inconsistency in the substeps.

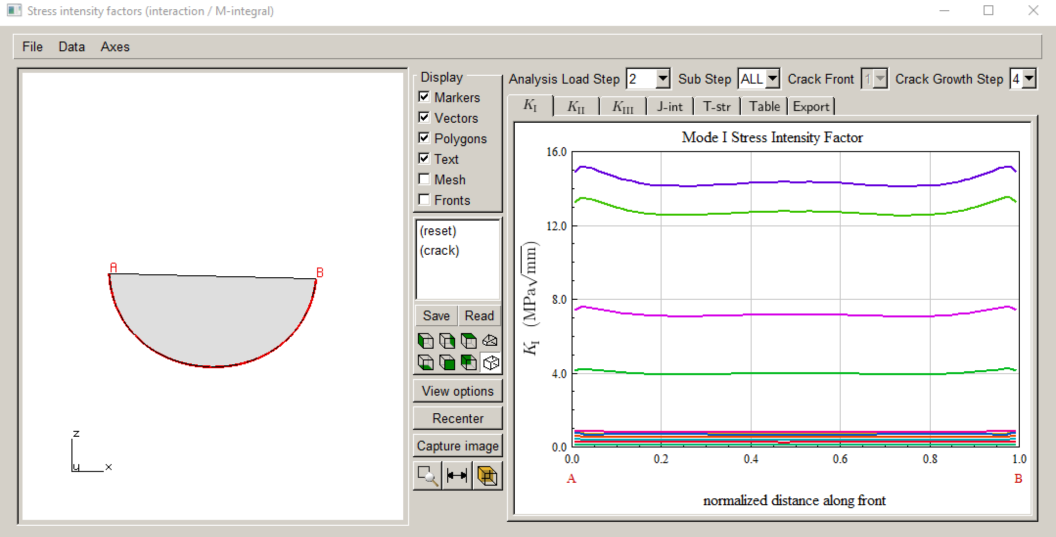


```
Total Load Cycles : 3,978,703
Total Passes : 3,978,703

Integration was Stopped Because:
   DK <= threshold DK
```

Figure 10.47 Cycle count for simple cyclic load event with missing substeps at crack step 4.

10.3.1 Missing Substep Results

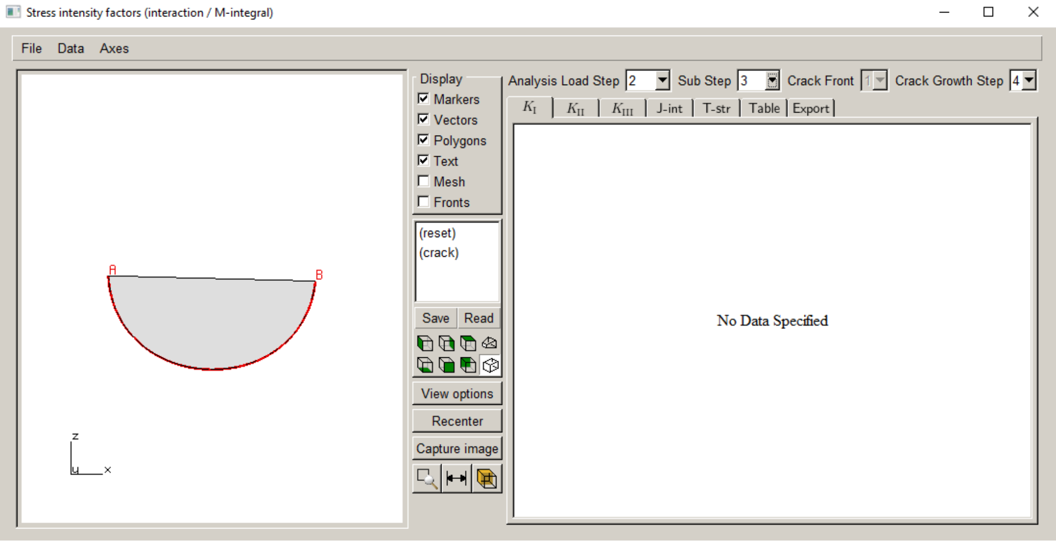
In some cases, ABAQUS might not write the results for a substep to the *.dtp* file. If FRANC3D finds that a substep is missing, the SIFs for that substep will be displayed as 'NoValue'. For example, Fig 10.48 shows the SIFs for load step 2 for ALL substeps. It is not obvious in this plot, but the substep #3 results are missing.



Analysis Load Step 2
Sub Step ALL
Crack Growth Step 4

Figure 10.48 SIF plot for load step 2 for ALL substeps.

For any substep where the results are missing, the SIF plot will appear as in Fig 10.49. The table display for all substeps will use the 'NoValue' string for this substep, Fig 10.50.



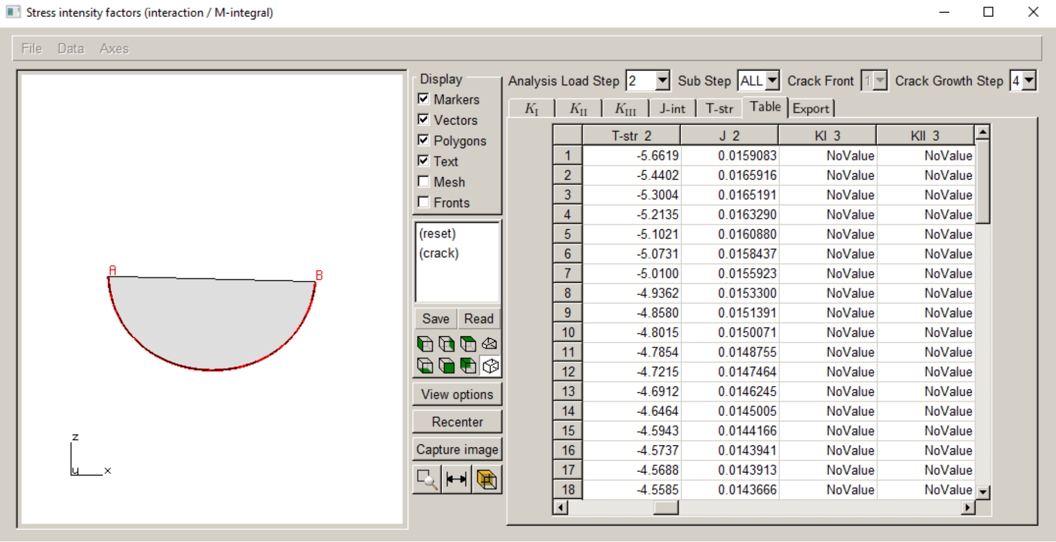
Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Fronts

(reset)
(crack)

Analysis Load Step 2 Sub Step 3 Crack Front 1 Crack Growth Step 4

No Data Specified

Figure 10.49 SIF plot for load step 2 for substep 3 with missing results.



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh
☐ Fronts

(reset)
(crack)

Analysis Load Step 2 Sub Step ALL Crack Front 1 Crack Growth Step 4

	T-str 2	J 2	KI 3	KII 3
1	-5.6619	0.0159083	NoValue	NoValue
2	-5.4402	0.0165916	NoValue	NoValue
3	-5.3004	0.0165191	NoValue	NoValue
4	-5.2135	0.0163290	NoValue	NoValue
5	-5.1021	0.0160880	NoValue	NoValue
6	-5.0731	0.0158437	NoValue	NoValue
7	-5.0100	0.0155923	NoValue	NoValue
8	-4.9362	0.0153300	NoValue	NoValue
9	-4.8580	0.0151391	NoValue	NoValue
10	-4.8015	0.0150071	NoValue	NoValue
11	-4.7854	0.0148755	NoValue	NoValue
12	-4.7215	0.0147464	NoValue	NoValue
13	-4.6912	0.0146245	NoValue	NoValue
14	-4.6464	0.0145005	NoValue	NoValue
15	-4.5943	0.0144166	NoValue	NoValue
16	-4.5737	0.0143941	NoValue	NoValue
17	-4.5688	0.0143913	NoValue	NoValue
18	-4.5585	0.0143666	NoValue	NoValue

Figure 10.50 SIF table for load step 2 showing NoValue for substeps with missing results.

If the NoValue is encountered, for example while looking for Kmax and Kmin, it is skipped. FRANC3D transient load events are not affected by this, but load events that specifically reference a particular substep will be affected especially if the substep results are missing.

11. Local+Global Connections

This chapter describes the local+global connections for several model scenarios. The recommended practice for local+global connections is node-merging using second-order elements for both the local and global model portions. The local portion is remeshed after crack insertion and second-order elements are required for SIF extraction. If the global portion has linear elements, one-to-one node merging is not possible as the global elements do not have midside nodes. This chapter describes the connections and gives recommendations to produce accurate SIFs. Tutorial #3 includes a section on different local+global connections also.

11.1 Uncracked ANSYS Model

We compare SIFs for a center-through crack in a thick plate under both uniform tension and pure shear for models with linear and quadratic elements. SIFs are computed using the three different extraction methods: M-integral (M), displacement correlation (DC), and virtual crack closure (VCCT).

A local+global approach is used, which means that the local cracked portion of the model will always have quadratic elements. For a linear-element global mesh, complete node merging on the cut-surfaces is not possible, so we discuss merging versus constraint-connections and compare the resulting SIFs.

The uncracked model is the same thick plate that is described in Section 6 of the Benchmark document. Two different sets of boundary conditions are used: 1) simple uniform unit tension, Fig 11.1, and 2) simple unit shear, Fig 11.2. Simple supports (constraints) are used in both cases. The original plate is 30x30x15, and the original mesh uses elements that are 1x1x1. Meshes with both linear elements and quadratic elements are created.

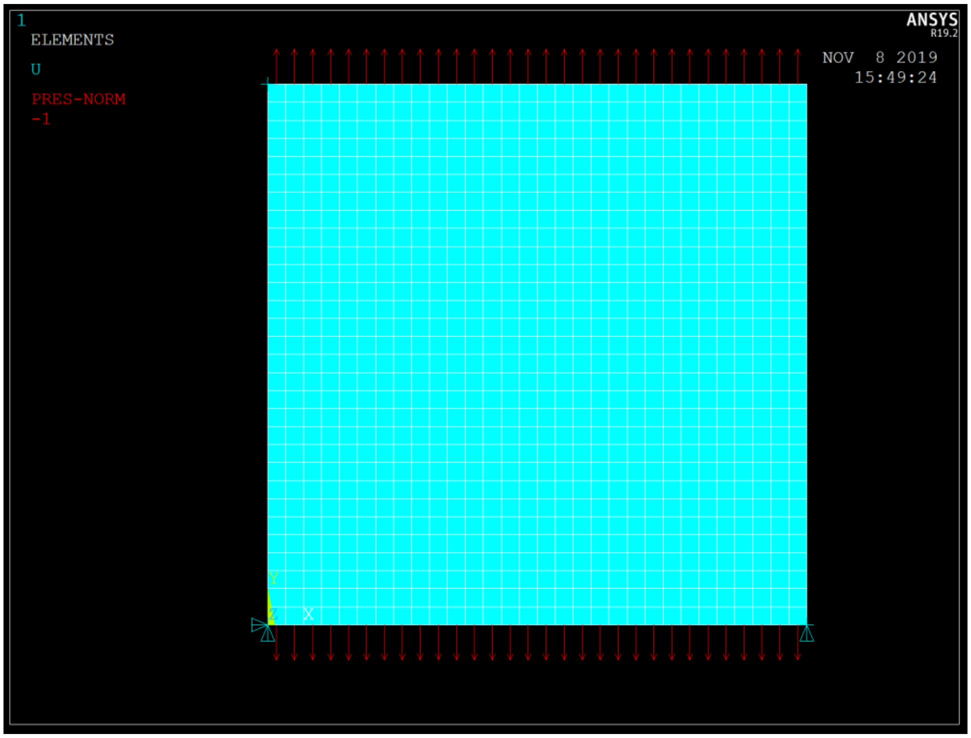


Figure 11.1 Initial ANSYS model with simple tension.

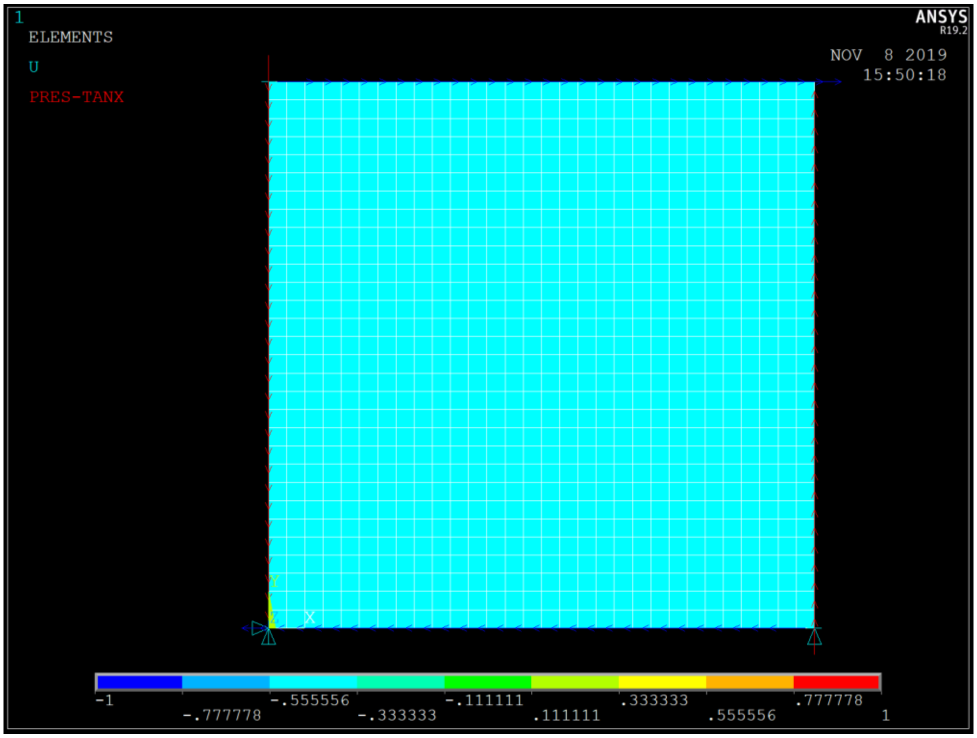


Figure 11.2 Initial ANSYS model with simple shear.

11.2 Cracked Model

A local+global approach is used, Fig 11.3; the global portion contains all the constraints and loads. The global portion of the mesh has either linear or quadratic elements depending on the original mesh. A center through crack is created, Fig 11.4; it is 2 units wide and extends through the plate thickness and is inserted into the local model.

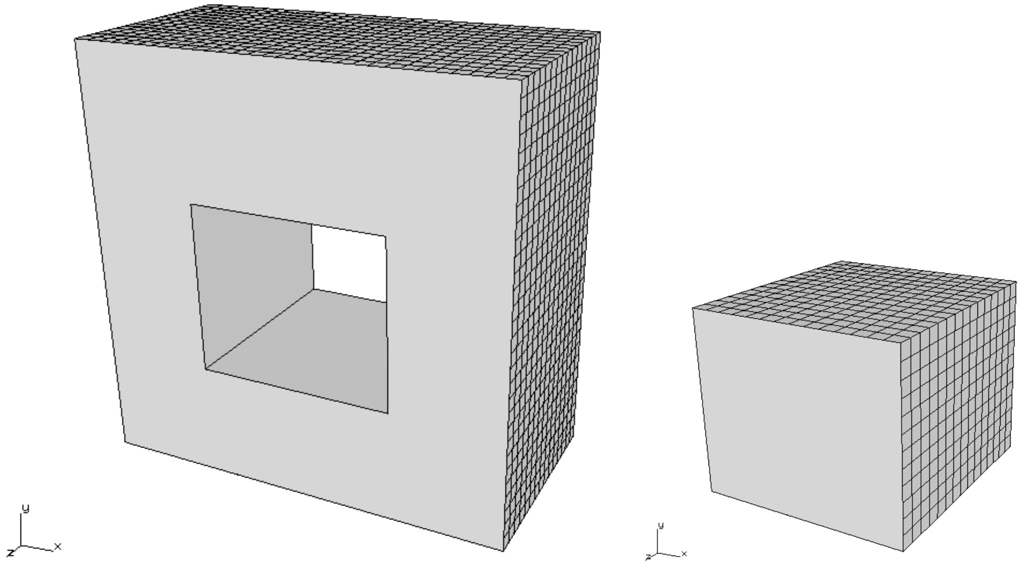


Figure 11.3 Global and local portions.

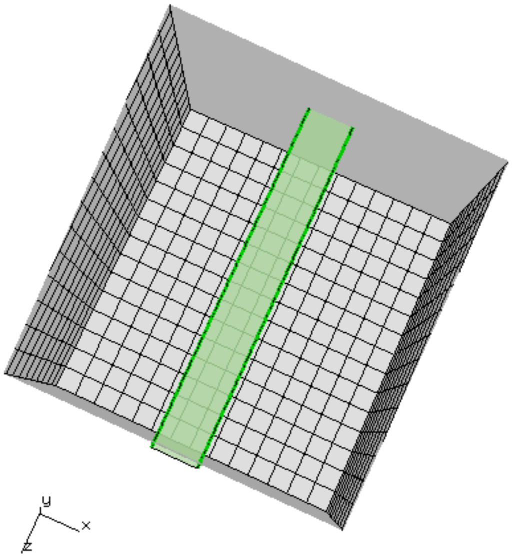


Figure 11.4 Center through crack.

11.3 Comparing SIFs for ANSYS Global Quadratic vs Linear Elements

Only one of the two crack fronts is examined. The first set of SIFs, Fig 11.5, is for the quadratic global mesh and uniform tension. The three methods of SIF extraction: M-integral (M), Virtual Crack Closure (VCCT) and Displacement Correlation (DC) give comparable results. The VCCT SIFs are less than 1% below the M SIFs, and the DC SIFs are about 1% above the M SIFs.

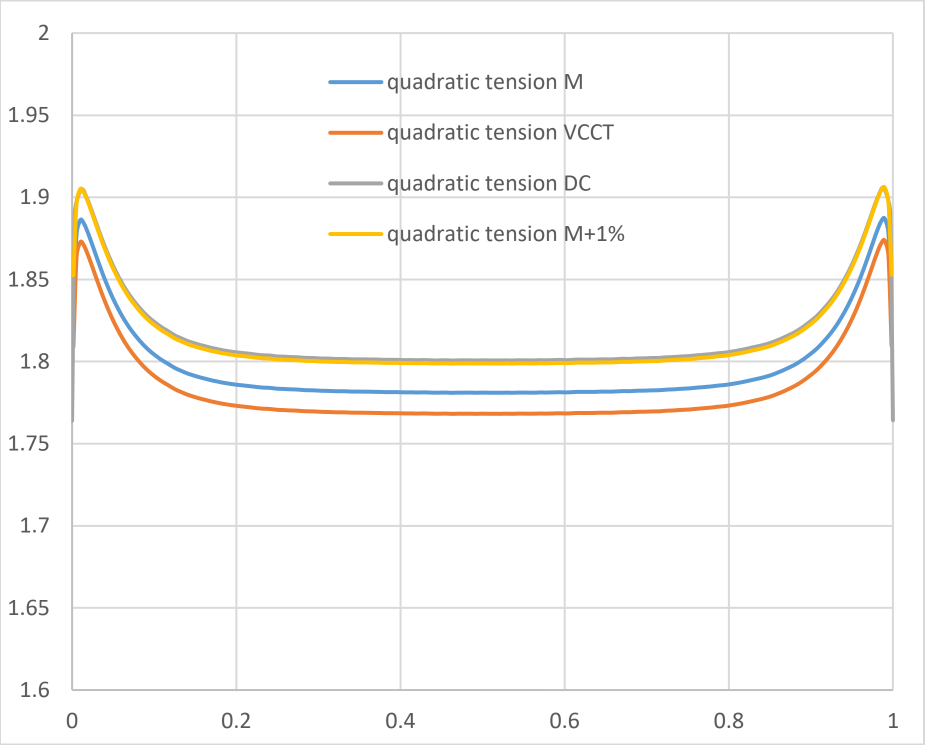
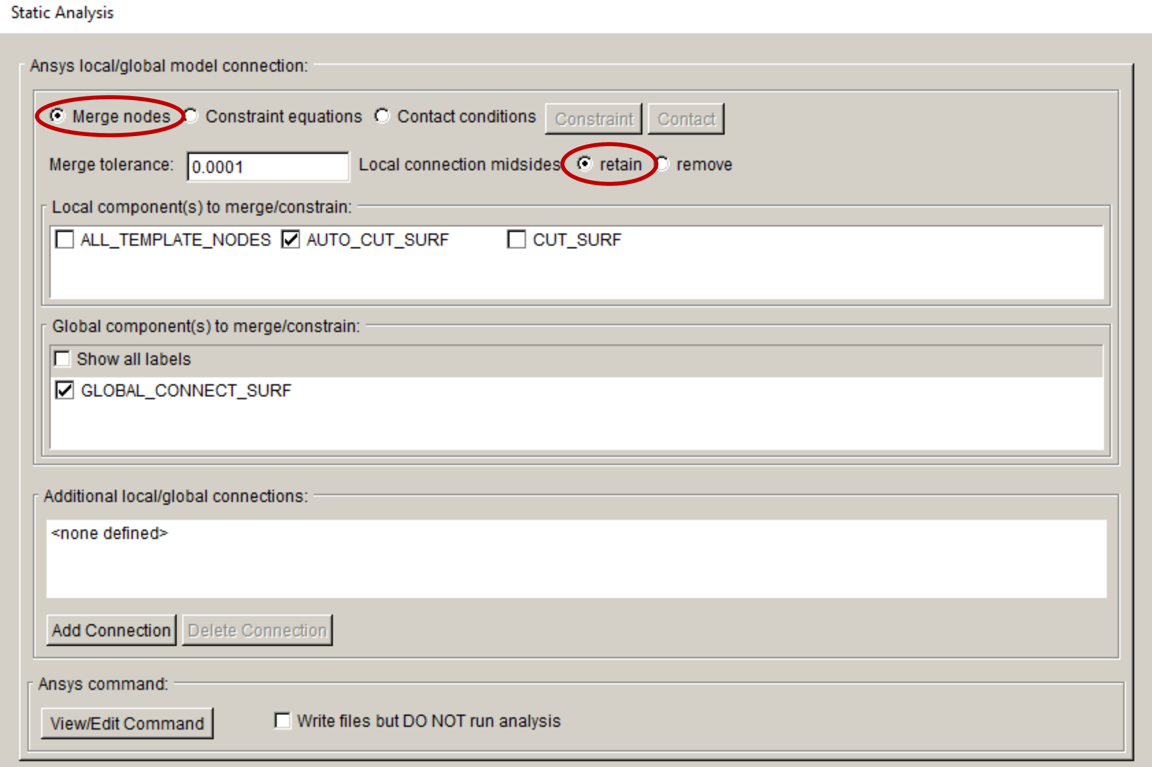


Figure 11.5 SIFs for quadratic mesh and uniform tension.

Fig 11.6 shows the FRANC3D local/global model connection dialog. The Merge nodes option is the default with the Local midside nodes retained. If the local model is extracted using the FRANC3D tools, the AUTO_CUT_SURF and GLOBAL_CONNECT_SURF components (or sets) are selected automatically. If the user subdivided the model using other tools, one component (or set) from the Local and one component from the Global must be selected to continue with the FE analysis.



Ansys local/global model connection:

● Merge nodes ○ Constraint equations ○ Contact conditions

Merge tolerance: 0.0001 Local connection midsides ● retain ○ remove

Local component(s) to merge/constrain:

☐ ALL_TEMPLATE_NODES ☑ AUTO_CUT_SURF ☐ CUT_SURF

Global component(s) to merge/constrain:

☐ Show all labels

☑ GLOBAL_CONNECT_SURF

Additional local/global connections:

<none defined>

Ansys command:

☐ Write files but DO NOT run analysis

Figure 11.6 Local – global connection for a mesh with all quadratic elements.

Fig 11.7 shows the corresponding results for the linear global mesh. The local+global connection uses node merging, but the midside nodes on the cut-surface of the local model are removed. Fig 11.8 shows the local/global model connection dialog with the local midside nodes set to be removed. Note that this option is available for ANSYS only. FRANC3D generates ANSYS APDL commands within the *_full.cdb* file to remove the midside nodes from the AUTO_CUT_SURF component of the local model.

The AUTO_CUT_SURF and GLOBAL_CONNECT_SURF are the only components that are selected.

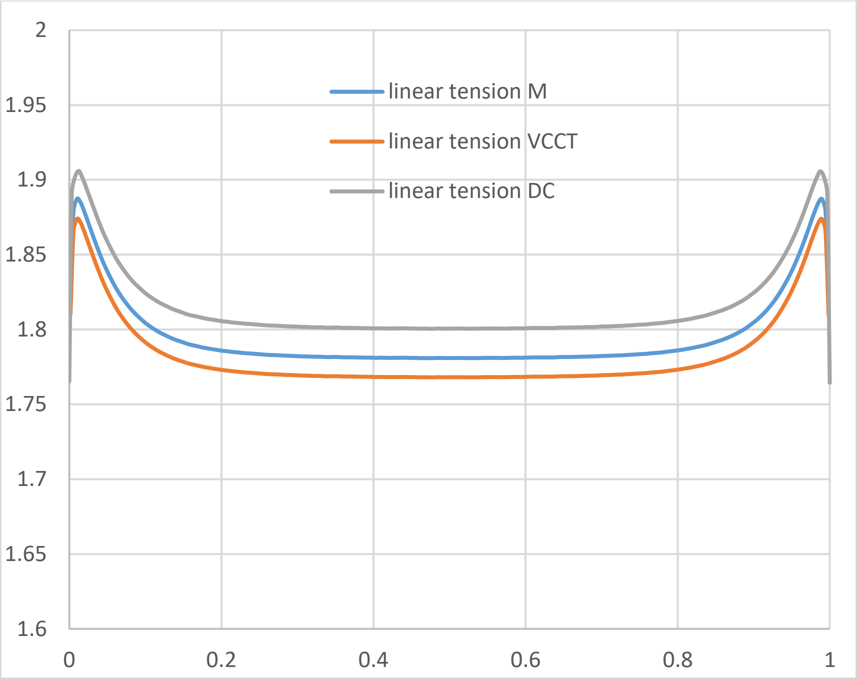
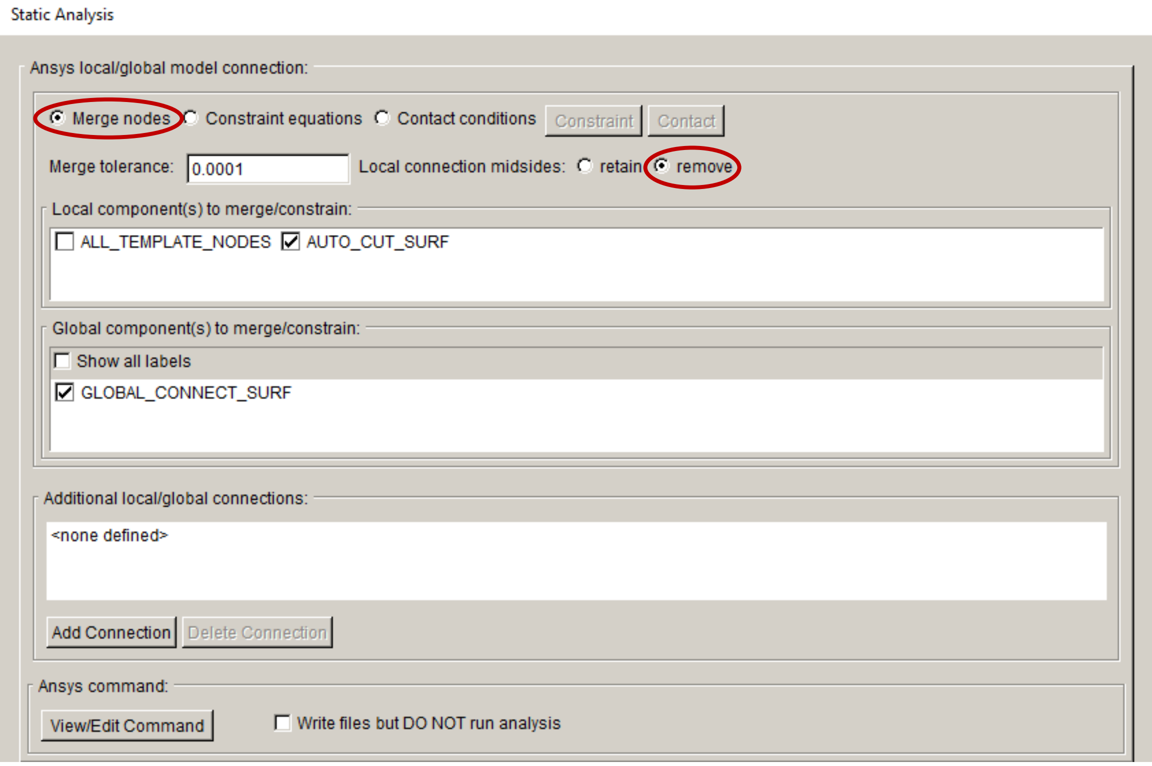


Figure 11.7 SIFs for linear global mesh and uniform tension.



Static Analysis

Ansys local/global model connection:

● Merge nodes ○ Constraint equations ○ Contact conditions

Merge tolerance: 0.0001 Local connection midsides: ○ retain ● remove

Local component(s) to merge/constrain:

□ ALL_TEMPLATE_NODES ☑ AUTO_CUT_SURF

Global component(s) to merge/constrain:

□ Show all labels

☑ GLOBAL_CONNECT_SURF

Additional local/global connections:

<none defined>

Ansys command:

□ Write files but DO NOT run analysis

Figure 11.8 Local – global connection for a mesh with global linear elements.

Figs 11.9 and 11.10 show the SIFs for the uniform shear model. In this case, we plot KII rather than KI. The DC SIFs are about 3% below the M SIFs, while the VCCT SIFs are close to the M SIFs. The linear global mesh produces comparable results, Fig 11.10. The local/global connections for the two models are the same as for the tension models.

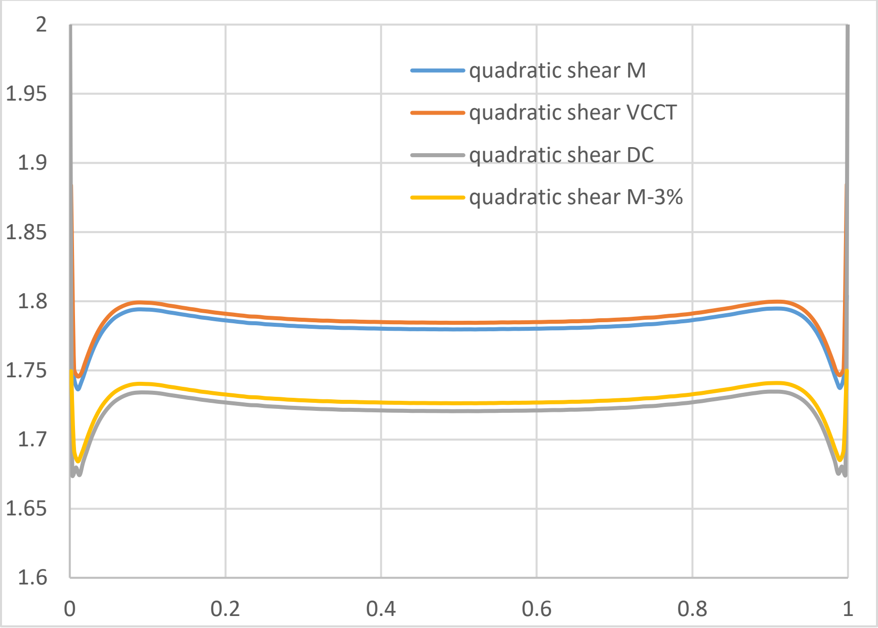


Figure 11.9 SIFs for quadratic mesh and uniform shear.

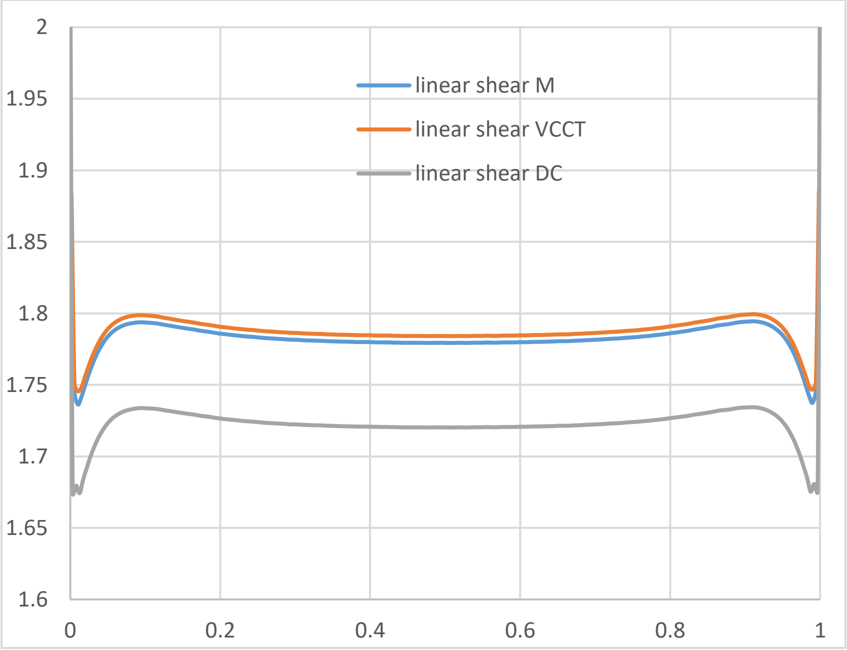


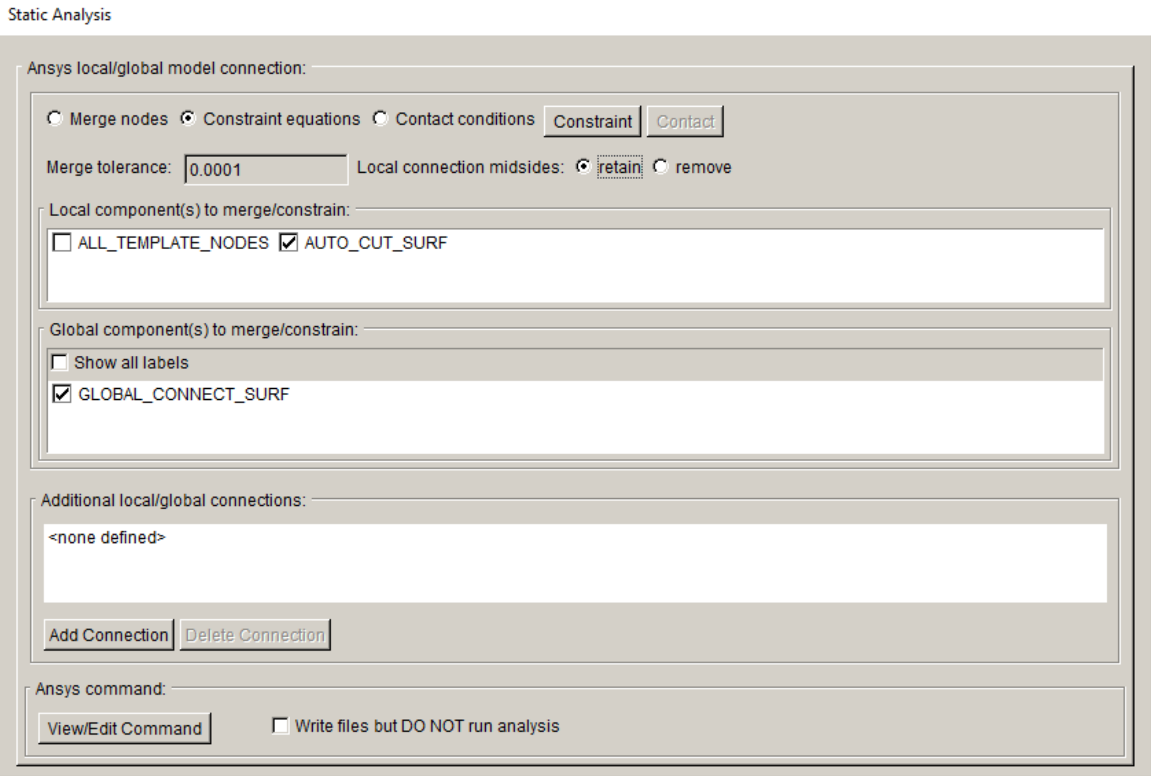
Figure 11.10 SIFs for linear global mesh and shear tension.

11.5 Local+Global Connection Options

FRANC3D provides three ways to connect the remeshed cracked local portion with the global portion: 1) merging nodes, 2) constraint equations, and 3) contact. The best connection is achieved by merging nodes.

For a quadratic global mesh, merging all corner and mid-side nodes is possible. For a linear global mesh, merging of only corner nodes is possible. For an ANSYS linear global mesh, FRAN3D suggests that the extra midside node be removed. For ABAQUS, for a linear global mesh, merging is usually not the best option. Rather, constraint equations typically provide better results.

Fig 11.11 shows the ANSYS local/global model connection with constraint equations selected. The constraint parameters can be set by selecting the **Constraint** button, which will display the dialog shown in Fig 11.12. A single local component and a single global component should be selected along with valid constraint settings.



Static Analysis

Ansys local/global model connection:

○ Merge nodes ● Constraint equations ○ Contact conditions

Merge tolerance: 0.0001

Local connection midsides: ● retain ○ remove

Local component(s) to merge/constrain:

☐ ALL_TEMPLATE_NODES ☑ AUTO_CUT_SURF

Global component(s) to merge/constrain:

☐ Show all labels

☑ GLOBAL_CONNECT_SURF

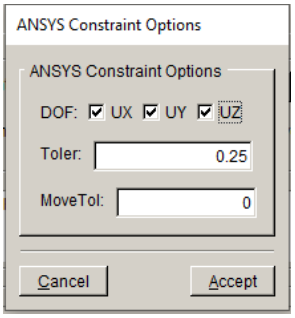
Additional local/global connections:

<none defined>

Ansys command:

☐ Write files but DO NOT run analysis

Figure 11.11 Local – global connection using constraint equations.



ANSYS Constraint Options

ANSYS Constraint Options

DOF: ☑ UX ☑ UY ☑ UZ

Toler: 0.25

MoveTol: 0

Figure 11.12 ANSYS constraint parameters.

Fig 11.13 shows the KI SIFs for different local+global connections for the uniform tension model. The M-integral is used to extract SIFs in all cases. The ANSYS constraint connection produces SIFs that are noticeably low compared with other connections; this might be improved by changing the retained midside node setting or by changing the constraint parameters.

The ABAQUS connection options are similar to what are available for ANSYS. The results for all the options have not been compiled here; just the linear global mesh with constraints is included for the tension model.

Fig 11.14 shows the KII SIFs for the uniform shear model; in this case, the ANSYS constraints produce reasonable SIFs.

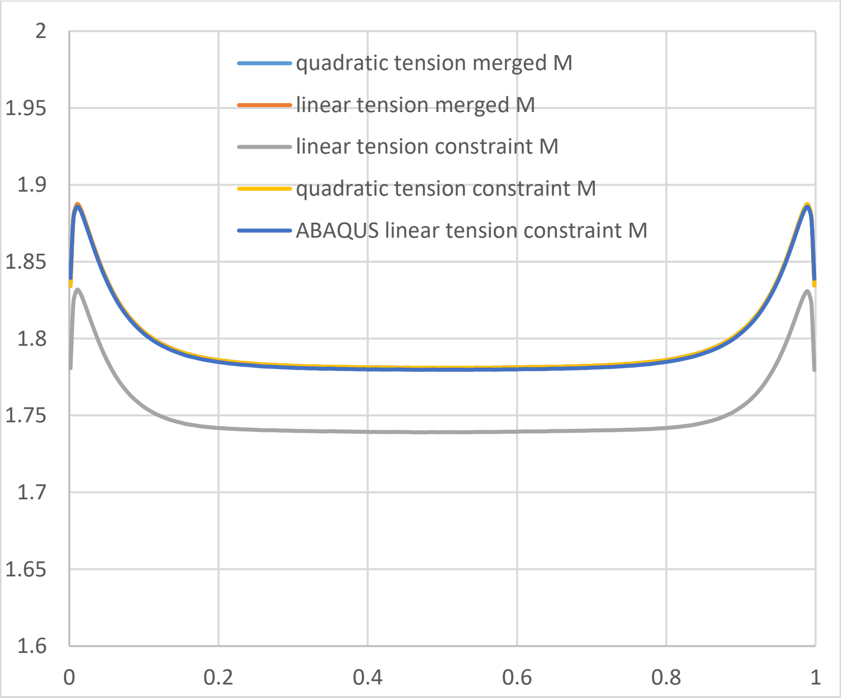


Figure 11.13 SIFs for different local+global connections for uniform tension.

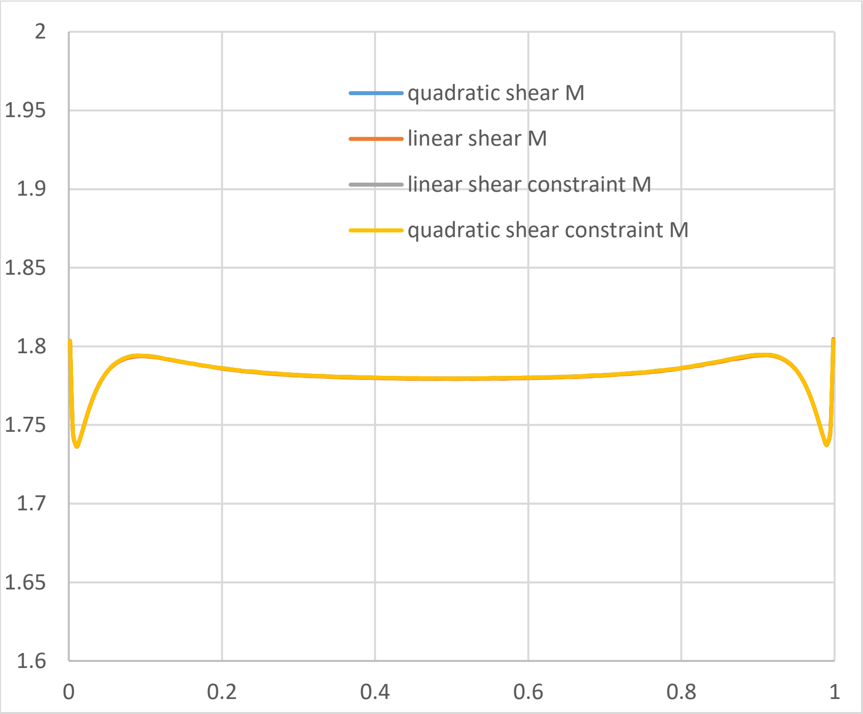
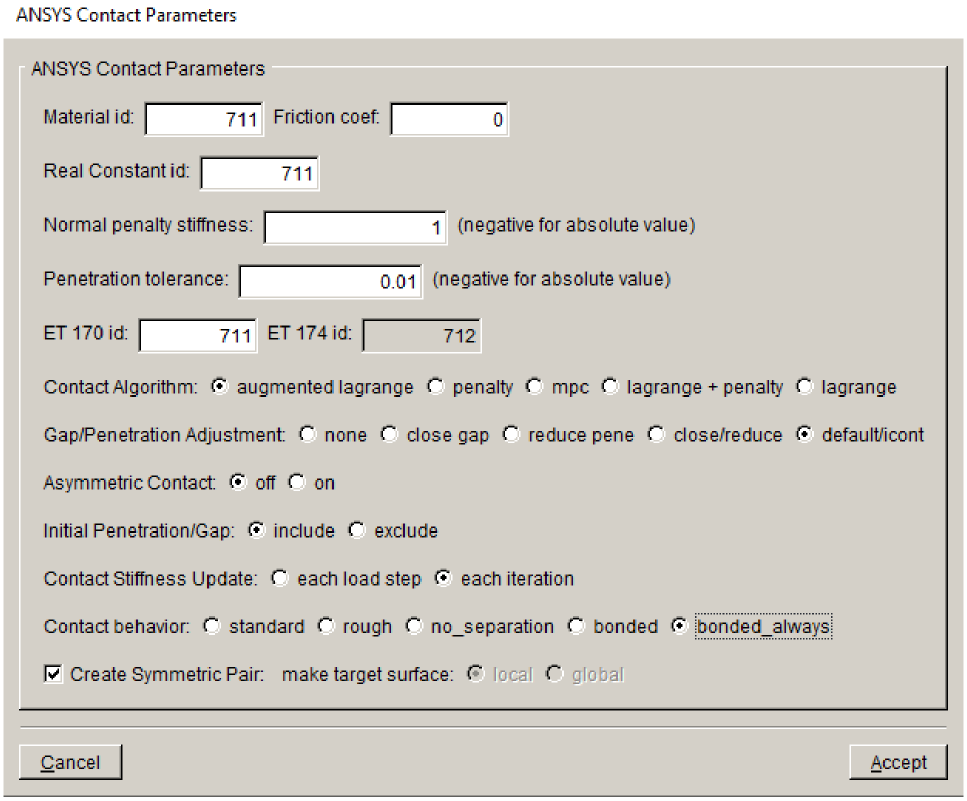


Figure 11.14 SIFs for different local+global connections for uniform shear.

In this model, ANSYS contact connections could not be used easily; the cut-surface probably needs to be separated into four surfaces to get contact to work correctly for ANSYS. However, if one were to use “bonded” contact for the local/global connection, the same rules apply: a single local component and a single global component should be selected along with valid contact settings, Fig 11.15.

For some models, the local/global connection is not sufficient to tie the two model portions together. Tutorial Sections 3.9 and 7.6 show examples where the user needs to use extra connections depending on the selection of the local submodel. Extra connections are like the basic local/global connection.



ANSYS Contact Parameters

ANSYS Contact Parameters

Material id: 711 Friction coef: 0

Real Constant id: 711

Normal penalty stiffness: 1 (negative for absolute value)

Penetration tolerance: 0.01 (negative for absolute value)

ET 170 id: 711 ET 174 id: 712

Contact Algorithm: ● augmented lagrange ○ penalty ○ mpc ○ lagrange + penalty ○ lagrange

Gap/Penetration Adjustment: ○ none ○ close gap ○ reduce pene ○ close/reduce ● default/icont

Asymmetric Contact: ● off ○ on

Initial Penetration/Gap: ● include ○ exclude

Contact Stiffness Update: ○ each load step ● each iteration

Contact behavior: ○ standard ○ rough ○ no_separation ○ bonded ● bonded_always

☑ Create Symmetric Pair: make target surface: ● local ○ global

Figure 11.15 ANSYS contact parameters with bonded_always selected.

12. Crack Face Tractions (CFTs)

Crack face traction loading can be used to include a simple pressure on the crack faces, or a residual or initial stress condition, or it can be used to replace the full boundary conditions. In the latter case, crack face tractions use the principle of linear superposition as depicted in Fig 12.1. FRANC3D has several options for applying crack face tractions as described in Section 7.2 of the Reference document.

Section 12.1 describes how the tractions are applied as nodal forces in the analysis. Section 12.2 describes how to apply CFTs to the analysis load steps. Section 12.3 describes how temperature is included.

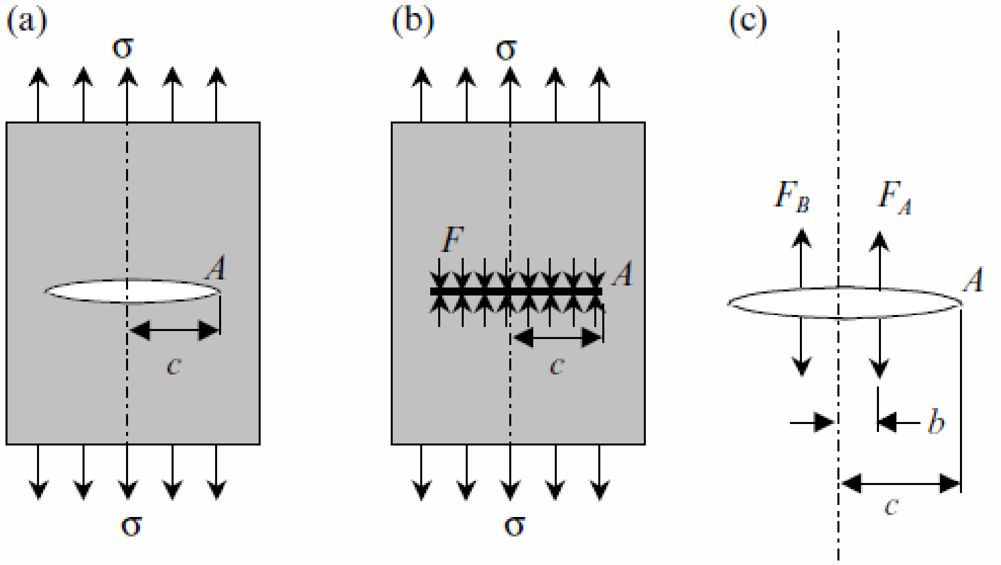


Fig 2.5.2 (a) Internal crack in a solid loaded with an external stress σ. (b) Crack closed by the application of a distribution of surface tractions F. (c) Internal crack loaded with surface tractions F_A and F_B.

Figure 12.1 Principle of linear superposition depicted;
from Introduction to Contact Mechanics, Fischer-Cripps, A.C., 2007, Springer-Verlag.

12.1 CFTs as Nodal Forces

CFTs are applied as nodal forces and include normal and shear components. All types of tractions, even a simple constant pressure, are converted to nodal forces for consistency. Section 2.1 of the Benchmark document compares the stress intensity factors (SIFs) for an internal penny-shape crack subjected to far-field tension versus the equivalent crack face traction (applied as a constant pressure); the CFT SIFs differ from the far-field SIFs by less than 1%.

The CFTs are applied as nodal forces in the analysis codes to make it easier to apply both normal and shear stress. To validate, we use a simple cube model with a surface half-penny crack.

Fig 12.2 shows an ABAQUS cube with the bottom surface constrained and both shear and tension applied to the top surface. The uncracked model is analyzed in ABAQUS to produce a nodal stress listing (both *.fil* and *.rpt* ABAQUS file types can be imported into FRANC3D). Section 2.9 of the Tutorials #2-14 document describes how to extract stress to apply as CFT.

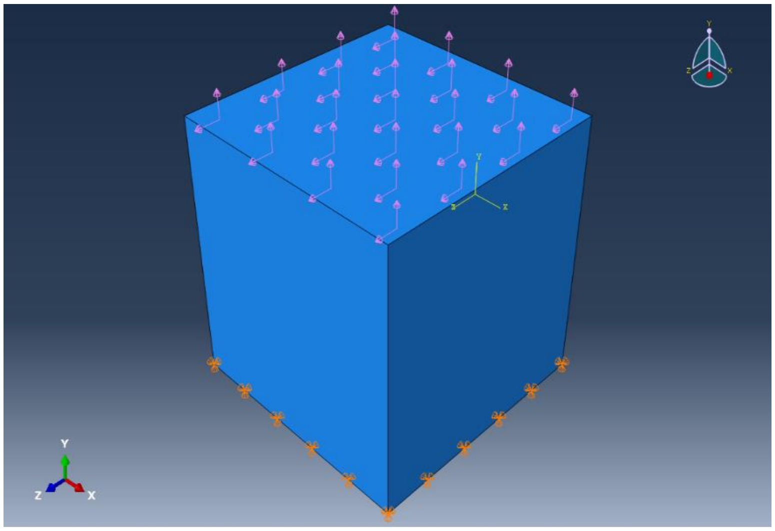


Figure 12.2 Cube model with normal tension and shear load.

12.1.1 Half-Penny Surface Crack (Curved Front)

A crack is inserted in the +z face of the cube, Fig 12.3, and analyzed. Fig 12.4 shows the maximum principal stress contours near the crack on the model surface. Note that crack face contact is not applied.

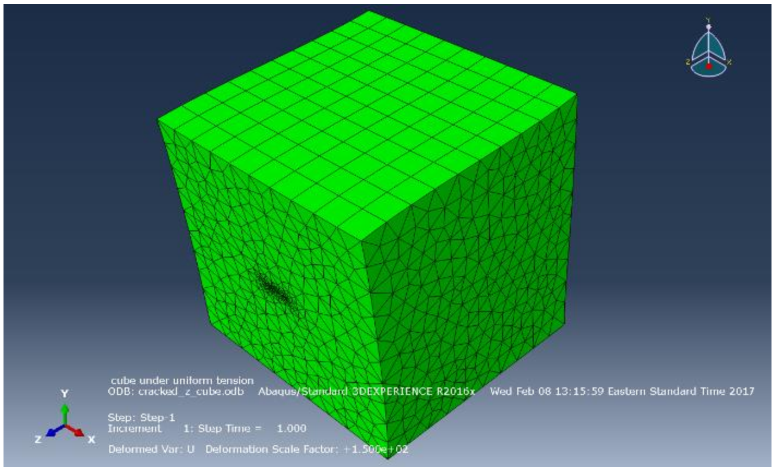

cube under uniform tension
ODB: cracked_z_cube.odb Abaqus/Standard 3DEXPERIENCE R2016x Wed Feb 08 13:15:59 Eastern Standard Time 2017
Step: Step-1
Increment 1: Step Time = 1.000
Deformed Var: U Deformation Scale Factor: +1.500e+02
Y
Z
X

Figure 12.3 Cube model with crack inserted in +z face.

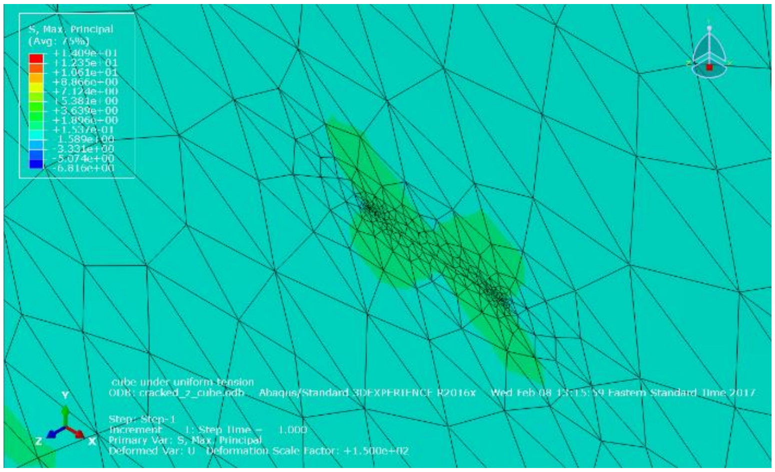

S, Max. Principal
(Avg: 75%)
+1.409e+01
+1.235e+01
+1.061e+01
+8.866e+00
+7.124e+00
+5.381e+00
+3.639e+00
+1.896e+00
+1.537e-01
-1.589e+00
-3.331e+00
-5.074e+00
-6.816e+00
cube under uniform tension
ODB: cracked_z_cube.odb Abaqus/Standard 3DEXPERIENCE R2016x Wed Feb 08 13:15:59 Eastern Standard Time 2017
Step: Step-1
Increment 1: Step Time = 1.000
Primary Var: S, Max. Principal
Deformed Var: U Deformation Scale Factor: +1.500e+02
Y
Z
X

Figure 12.4 Stress contours for +z face crack.

The resulting SIFs are shown in Figs 12.5a-c, using the M-integral (M), displacement correlation (DC) and Virtual Crack Closure (VCCT). The M-integral and VCCT methods include a term for the CFT, and we use the DC SIFs to verify that the CFT term is added correctly; if the CFT was not included the SIFs would differ significantly. In this case, the SIFs are basically the same for the three methods. In addition, the far-field loading and CFT loading produce the same SIFs.

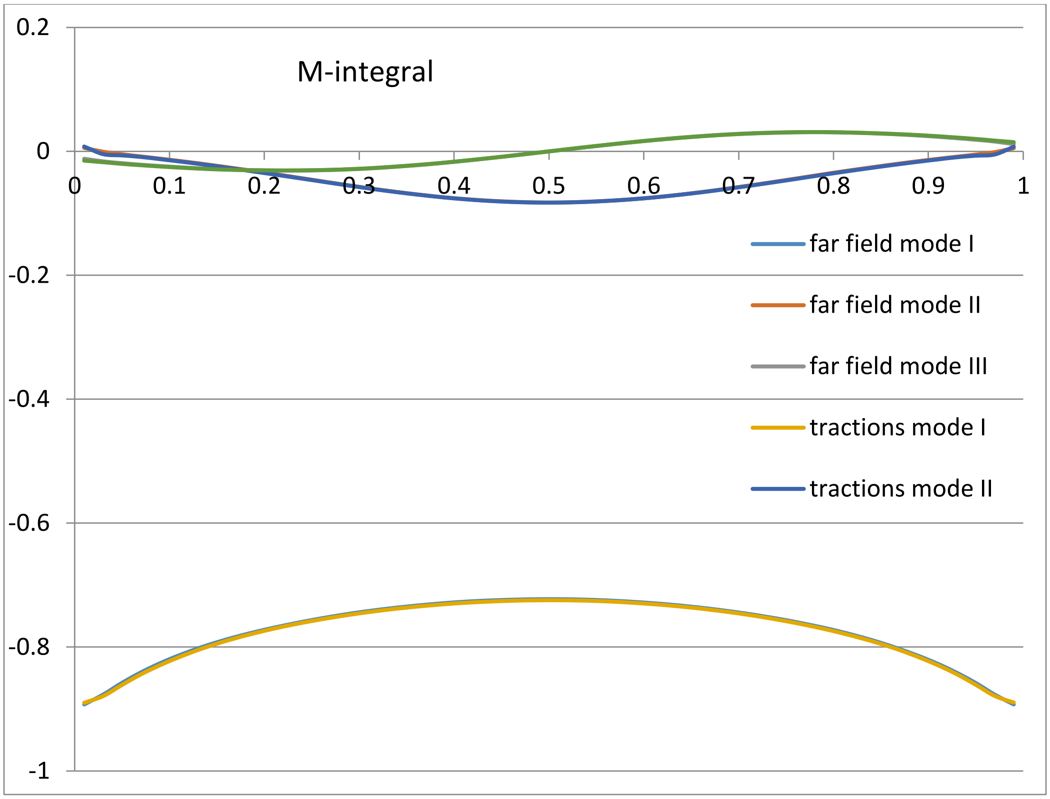


Figure 12.5a M-integral SIFs for +z face crack.

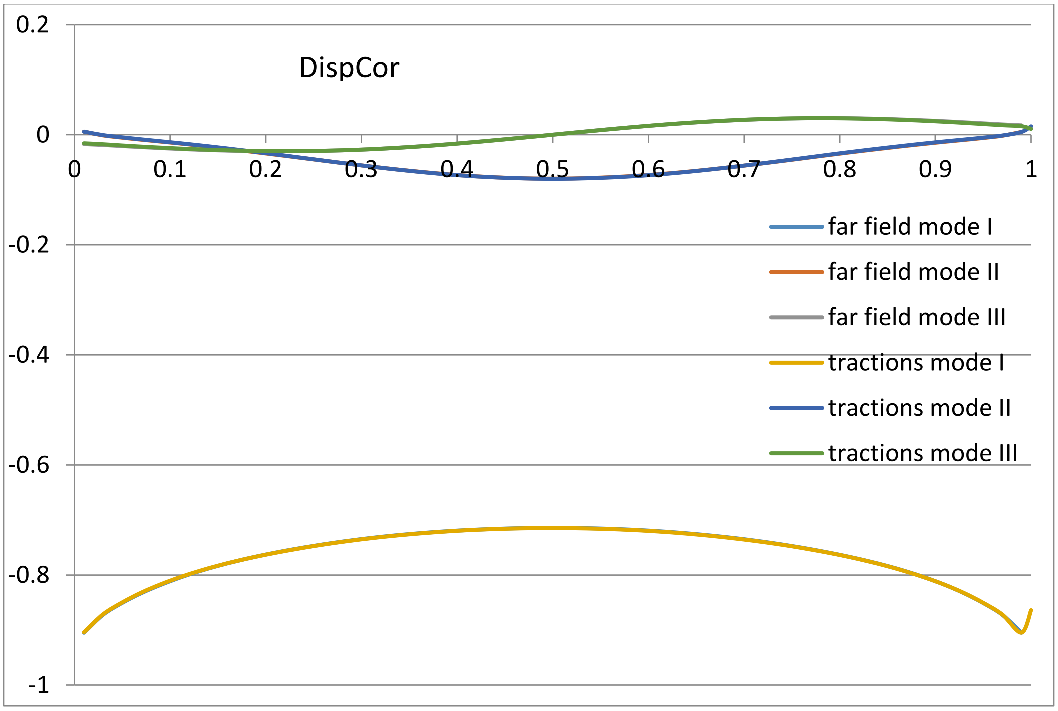


Figure 12.5b DC SIFs for +z face crack.

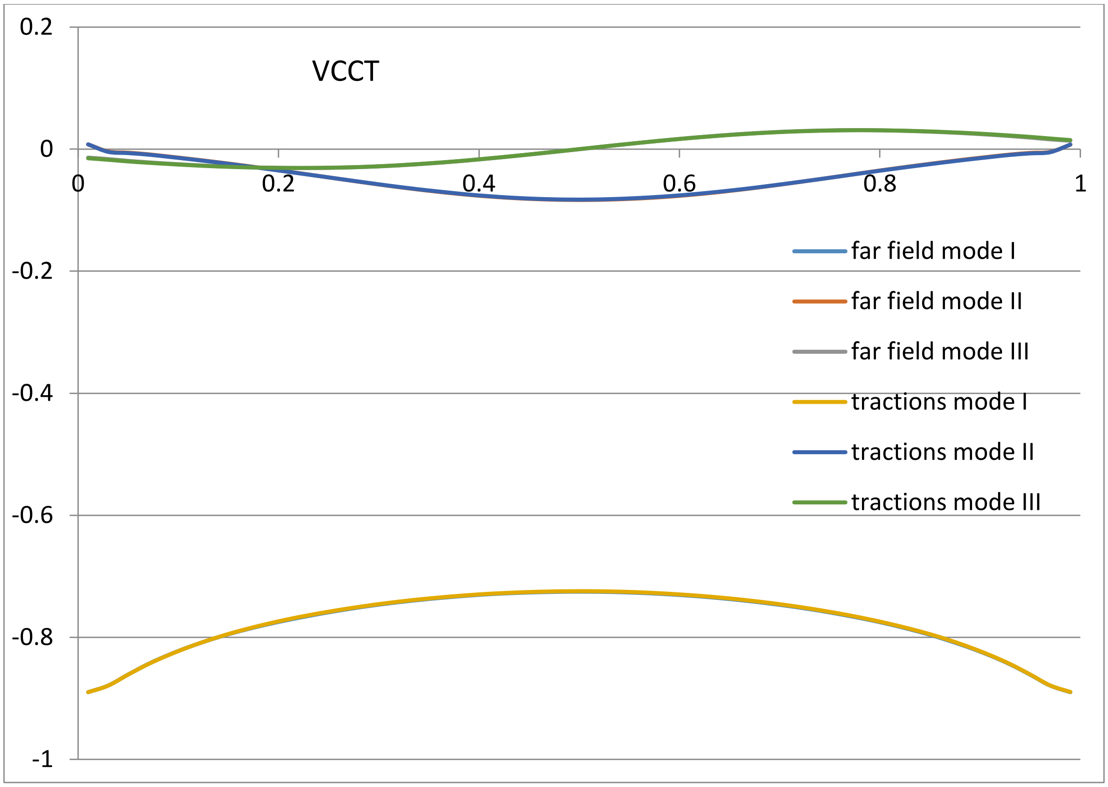


Figure 12.5c VCCT SIFs for +z face crack.

The crack in the +z face is not subject to much variation in shear stress. We insert a crack into the +x face of the cube, Fig 12.6. The stress contours on the model surface near the crack are shown in Fig 12.7. Compared with the +z face crack, the +x face crack has a significantly different local stress field.

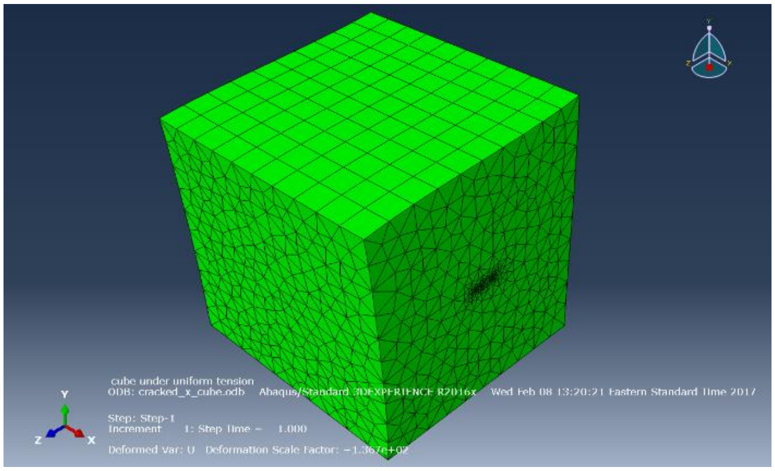

```
cube under uniform tension
ODB: cracked_x_cube.odb    Abaqus/Standard 3DEXPERIENCE R2016x    Wed Feb 08 13:20:21 Eastern Standard Time 2017

Step: Step-1
Increment     1: Step Time =    1.000

Deformed Var: U   Deformation Scale Factor: -1.367e+02
```

Figure 12.6 Cube model with crack inserted in +x face.

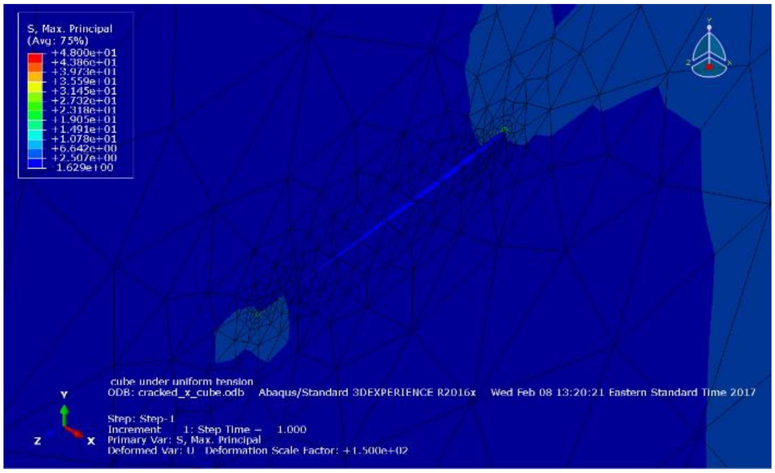

S, Max. Principal
(Avg: 75%)
+4.800e+01
+4.386e+01
+3.973e+01
+3.559e+01
+3.145e+01
+2.732e+01
+2.318e+01
+1.905e+01
+1.491e+01
+1.078e+01
+6.642e+00
+2.507e+00
-1.629e+00
Y
Z
X
cube under uniform tension
ODB: cracked_x_cube.odb Abaqus/Standard 3DEXPERIENCE R2016x Wed Feb 08 13:20:21 Eastern Standard Time 2017
Y
Z
X
Step: Step-1
Increment 1: Step Time = 1.000
Primary Var: S, Max. Principal
Deformed Var: U Deformation Scale Factor: +1.500e+02

Figure 12.7 Stress contours for +x face crack.

The SIFs for the far-field loading versus CFT for M-integral, DC and VCCT are shown in Figs 12.8a-c. The far-field and CFT SIFs are the same, and the M-integral, DC and VCCT SIFs are the same.

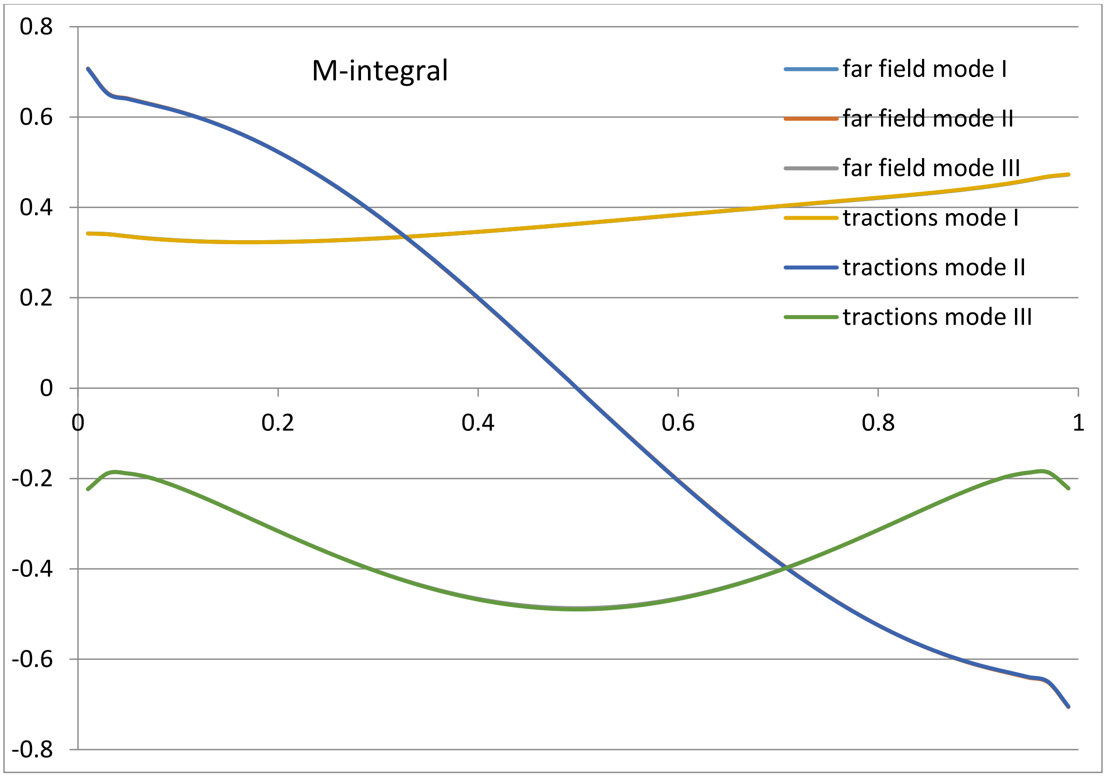


Figure 12.8a M-integral SIFs for +x face crack.

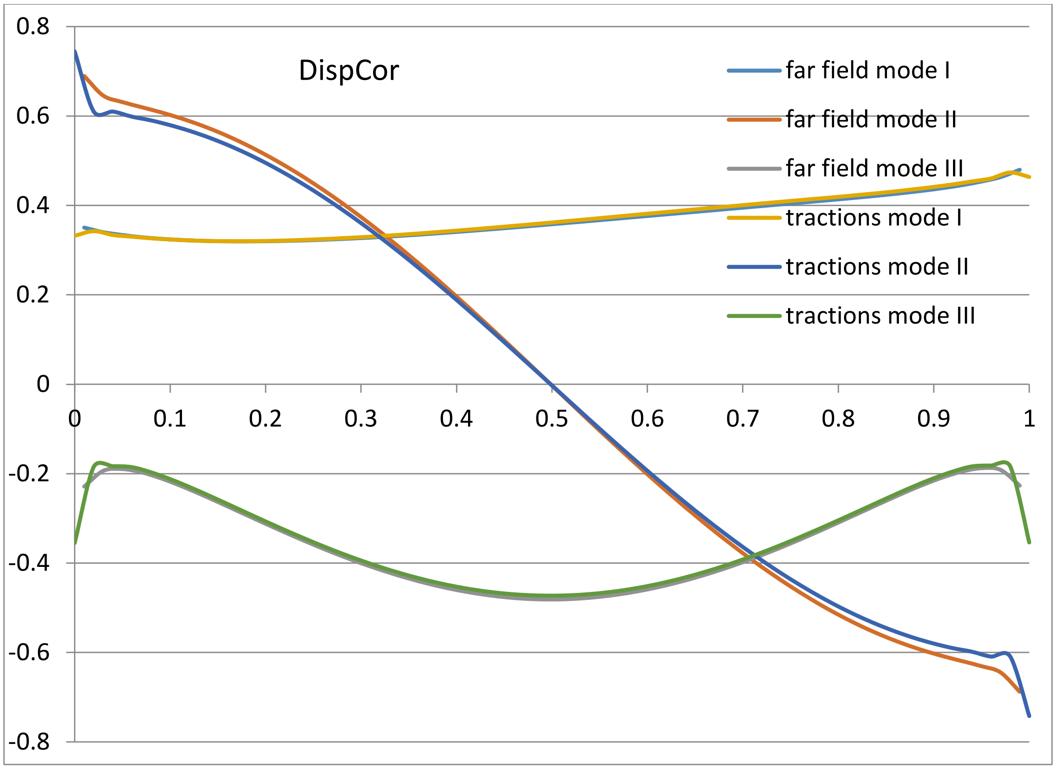


Figure 12.8b DC SIFs for +x face crack.

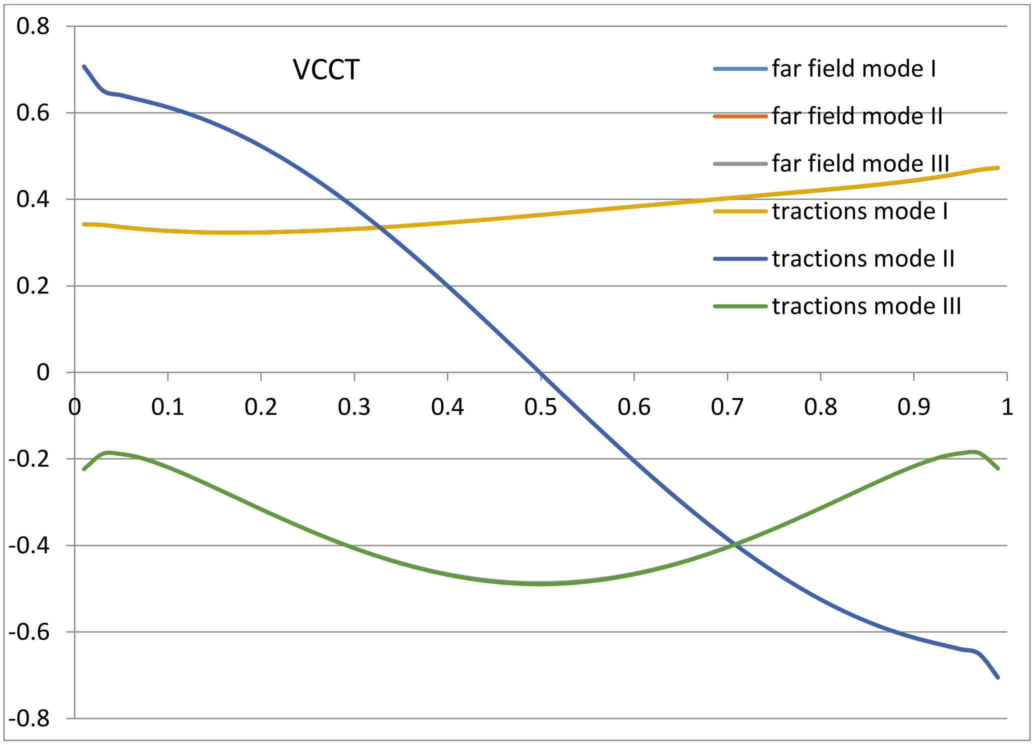


Figure 12.8c VCCT SIFs for +x face crack.

Figs 12.9a-b show the Mode I, II and III SIFs for the CFT loading for all three SIF methods. The values for the DC method differ by a couple of percent from the values for the M and VCCT methods.

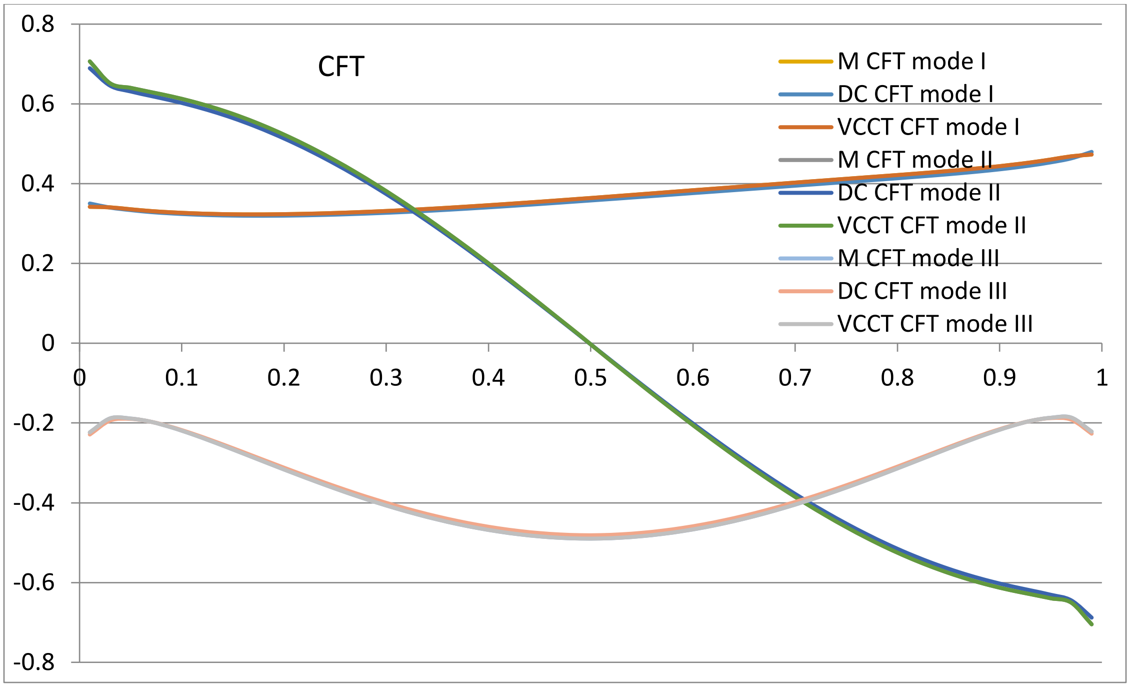


Figure 12.9a CFT SIFs for +x face crack.

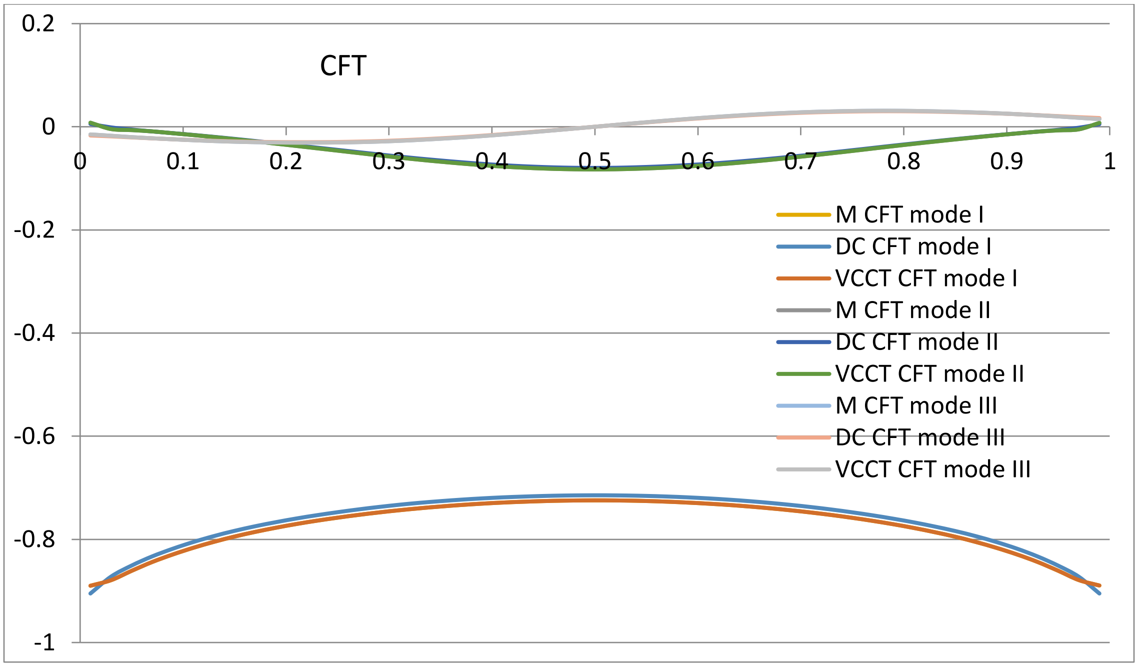


Figure 12.9b CFT SIFs for +z face crack.

12.1.2 Edge-Through Crack (Straight Front)

Straight-front edge cracks were inserted in the same +x and +z cube faces, Fig 12.10. The corresponding SIFs are shown in Figs 12.11-13. The straight crack front does not require any “curvature” correction when computing M-integral SIFs; the crack front local coordinate system is constant along the front.

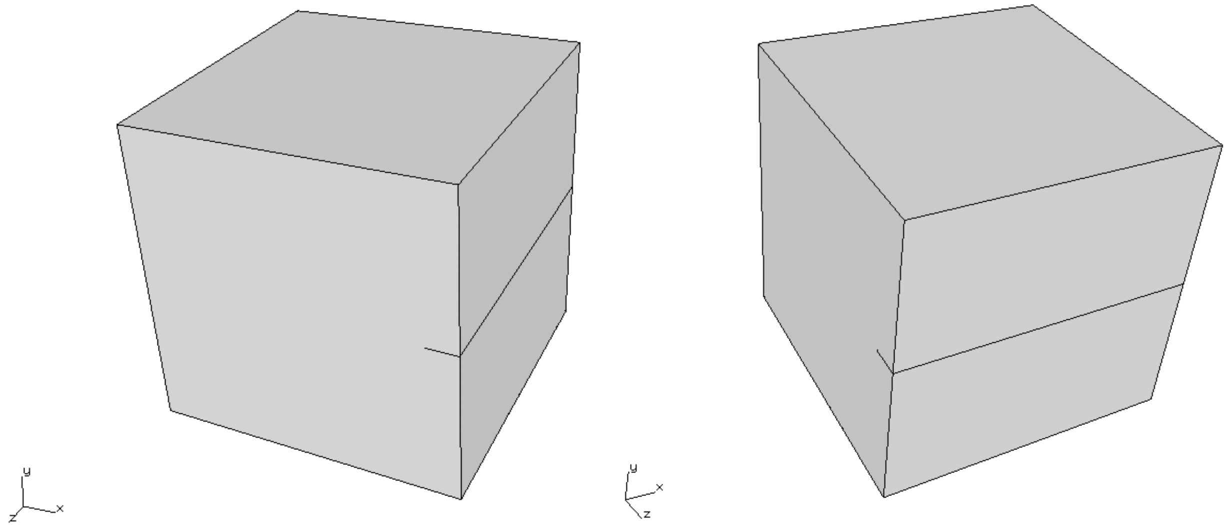


Figure 12.10. +x edge crack (left) and +z edge crack (right).

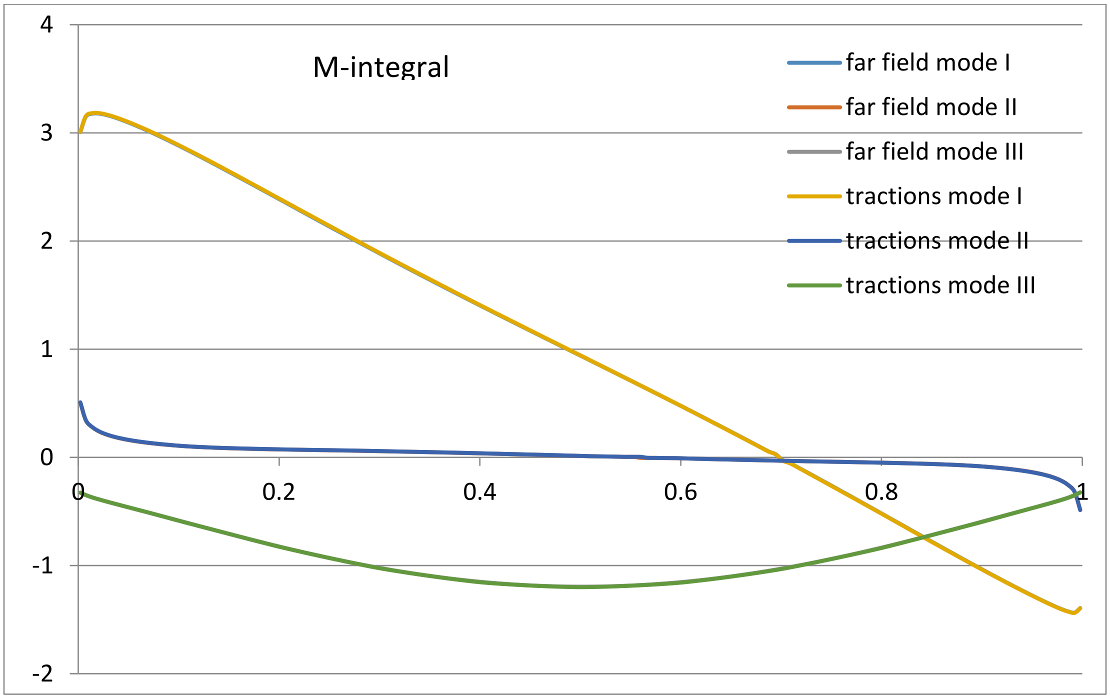


Figure 12.11a M SIFs for +x edge crack.

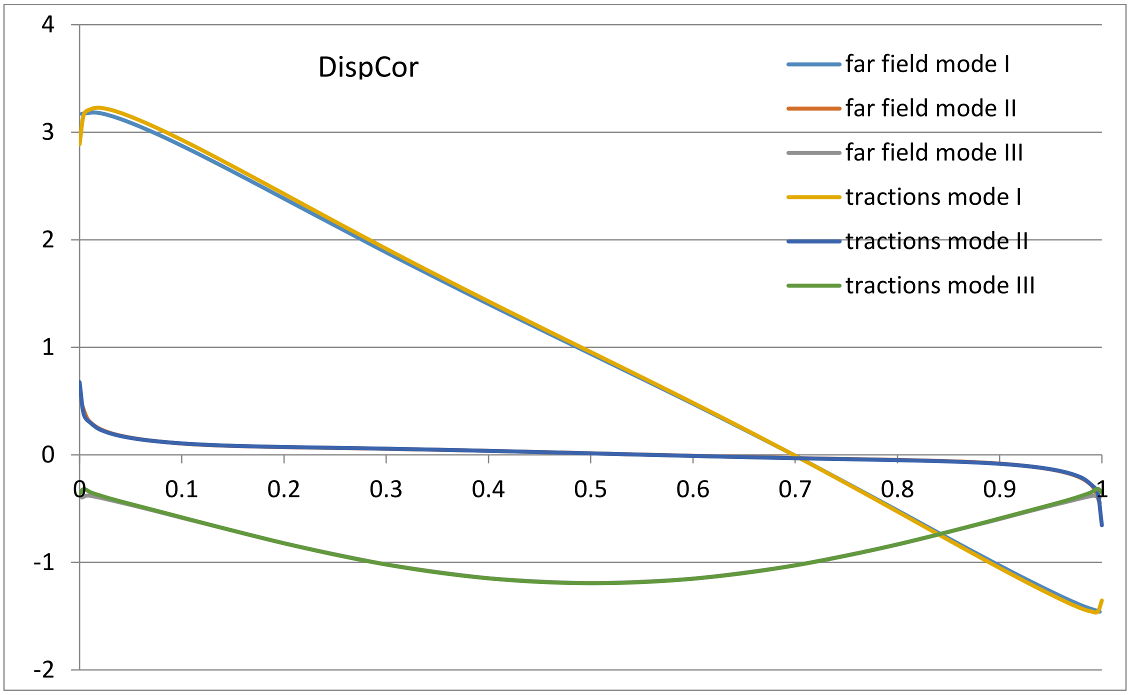


Figure 12.11b DC SIFs for +x edge crack.

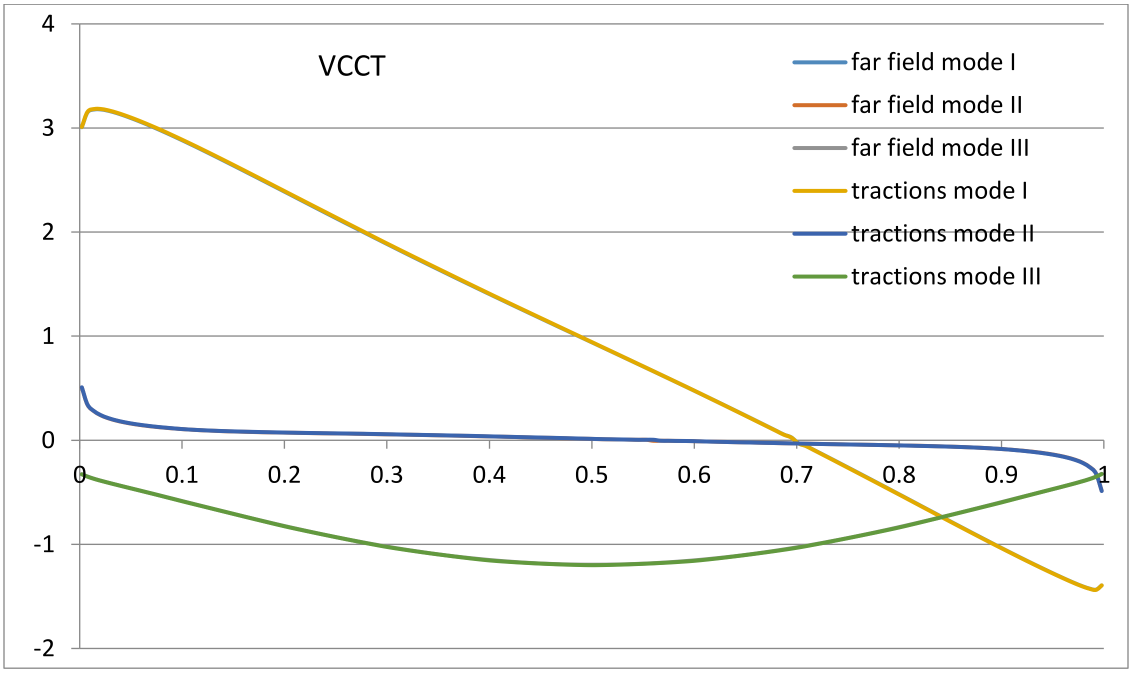


Figure 12.11c VCCT SIFs for +x edge crack.

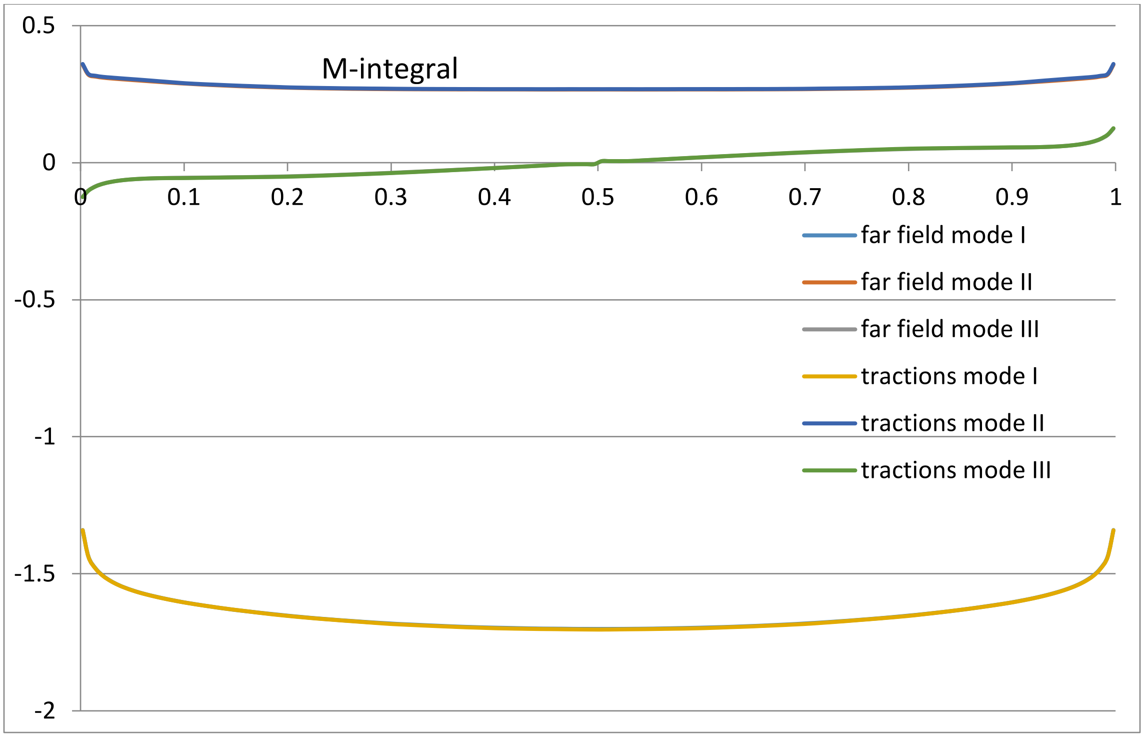


Figure 12.12a M SIFs for +z edge crack.

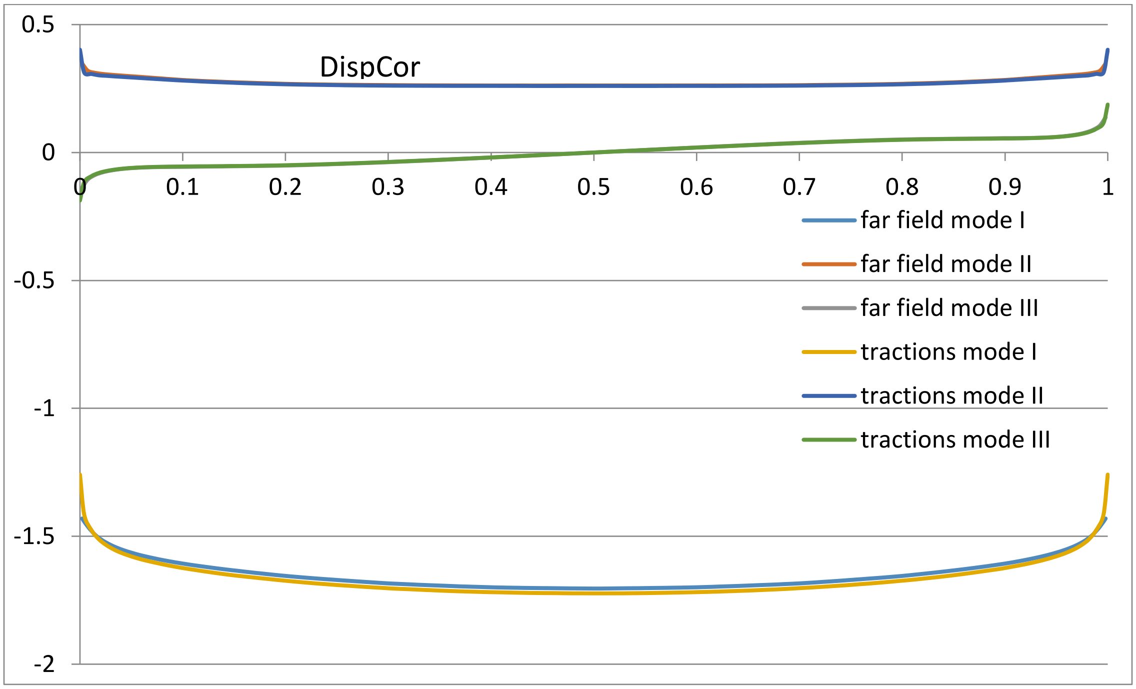


Figure 12.12b DC SIFs for +z edge crack.

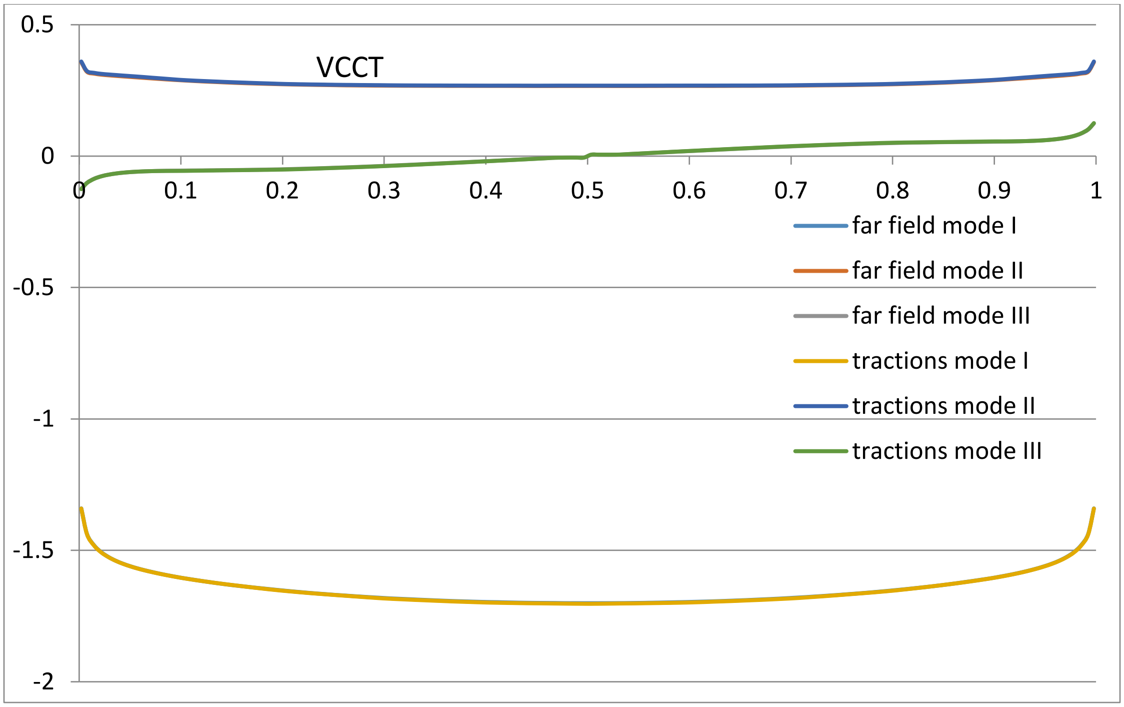


Figure 12.12c VCCT SIFs for +z edge crack.

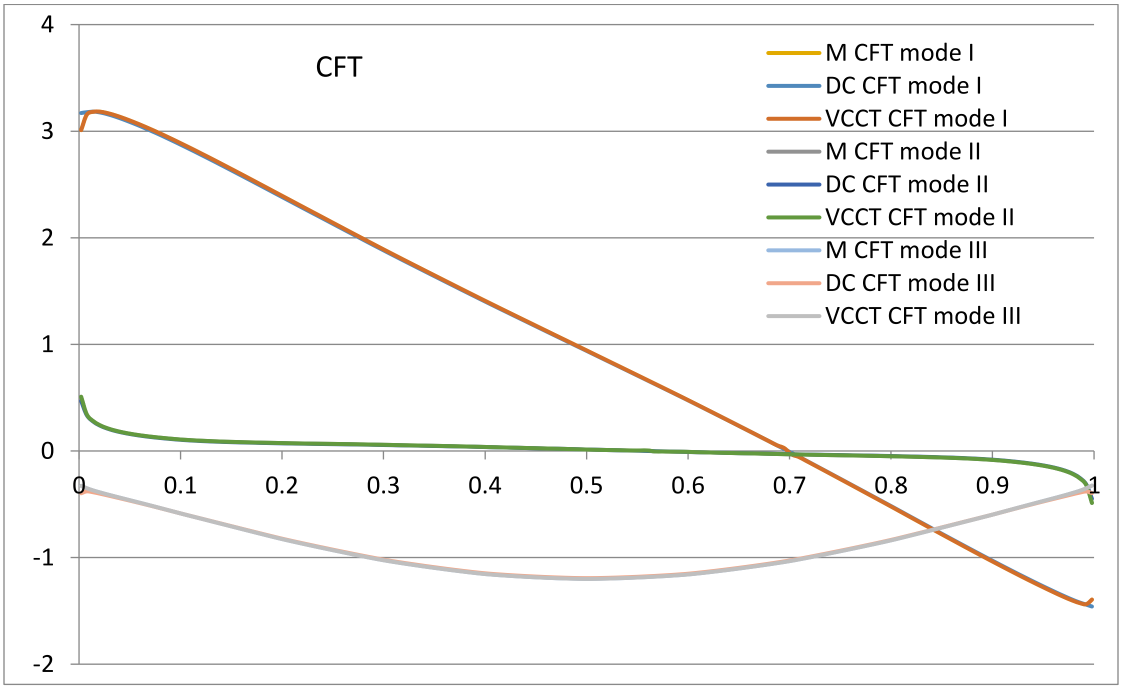


Figure 12.13a CFT SIFs for +x edge crack.

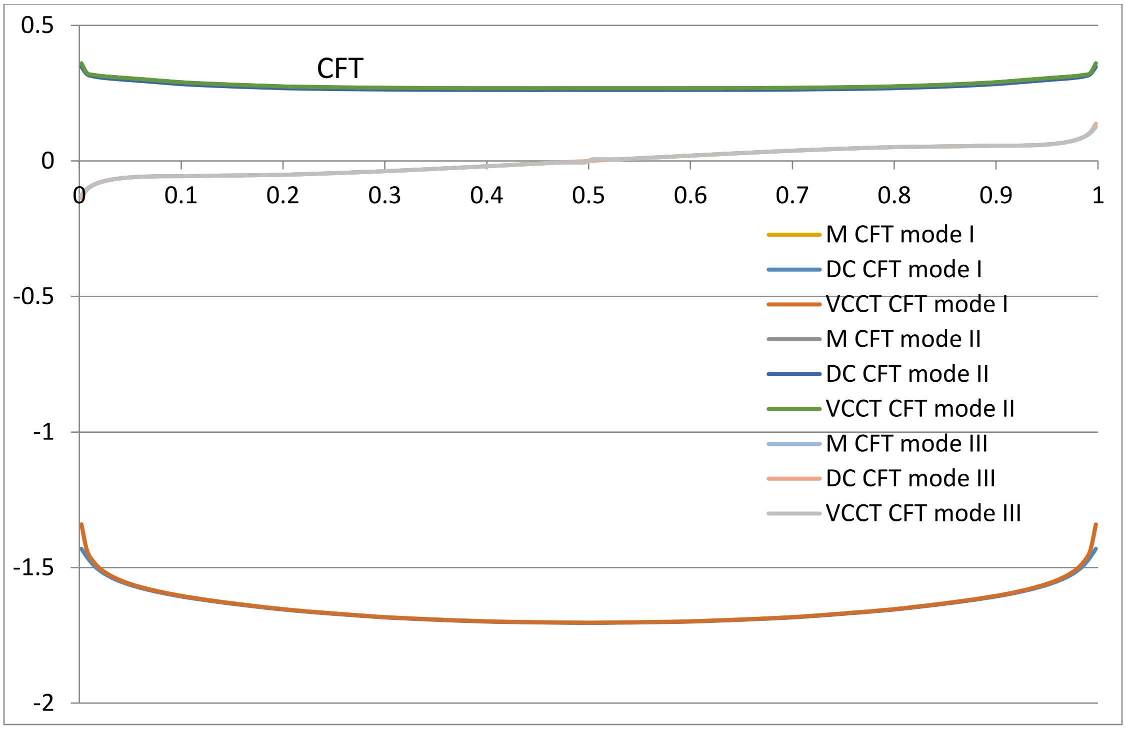


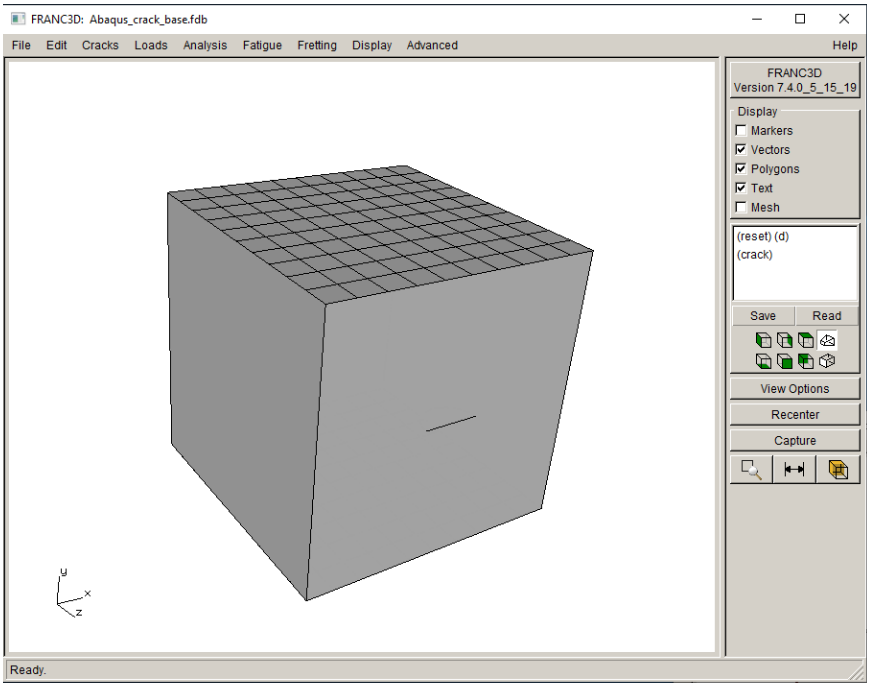
Figure 12.13b CFT SIFs for +z edge crack.

12.2 Add Crack Face Tractions to Existing Load Steps

In older versions of FRANC3D, CFT was added as a separate load step that was solved after all other input FE load steps had been solved. Now one can add a CFT to an existing load step. This allows for situations such as a pressurized pipe with a crack where the pressure on the pipe also acts on the crack surfaces.

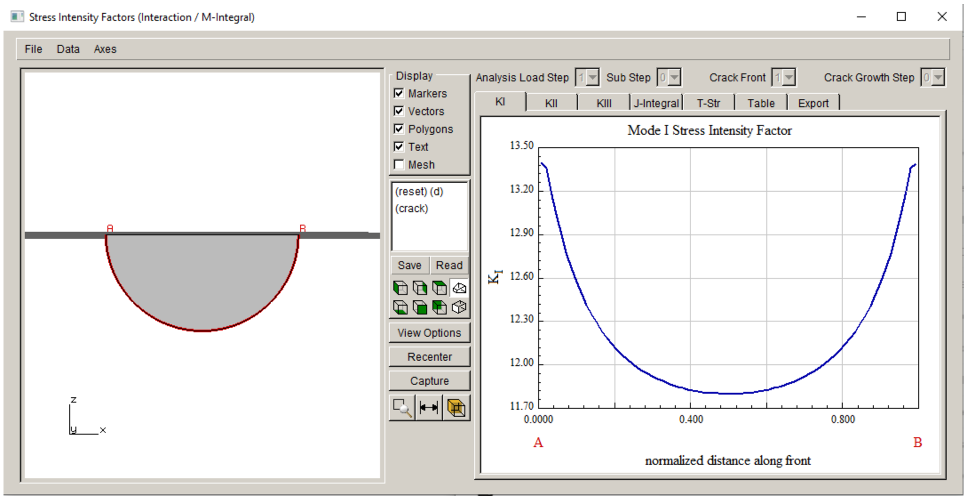
This also will allow for thermal conditions to be included – something that was not permitted previously if a CFT was added as an extra load step (temperature is discussed in Section 12.3).

To demonstrate how to apply a CFT in different load steps, we use with a simple cube with a penny-shaped surface crack (see the base Tutorial document). This ‘base’ model is shown in Fig 12.14 and the ‘base’ SIFs are shown in Fig 12.15. The cube has E=10000 and NU=0.3; the reference temperature is 0; there are no temperatures set in the model. The base loading consists of uniform traction on the top surface and constraint on the bottom surface.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Figure 12.14 Base model - penny-shaped surface crack in a cube under uniform tension.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

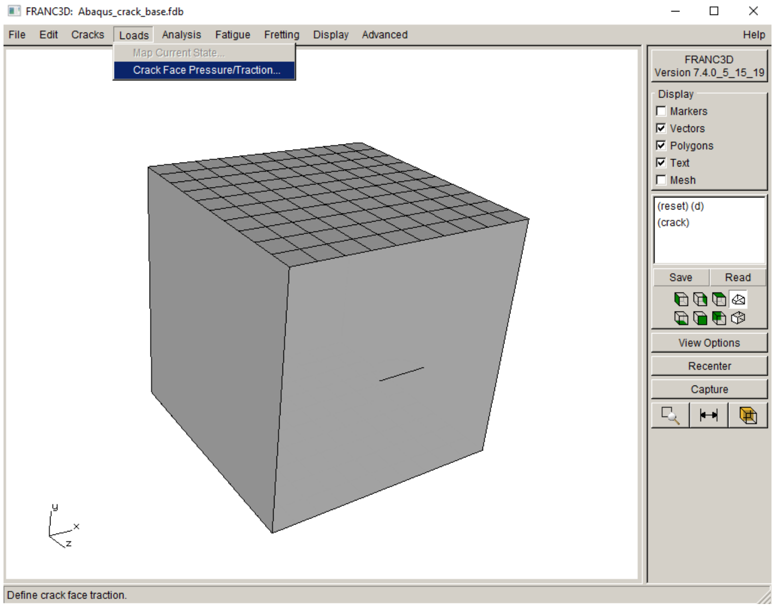
Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 12.15 Base SIFs for penny-shaped surface crack in a cube under uniform tension.

To add CFTs to this model, go to the FRANC3D **Loads** menu and select **Crack Face Pressure/Traction**, Fig 12.16. The CFT dialog is displayed, Fig 12.17a. Click on the **Add** button to add a new CFT to the model.

The default type of CFT is a constant pressure, Fig 12.17b, which is what we will use here. Click on the **Advanced** button to display the dialog in Fig 12.17c. The default is to add a new load step, which is what we will do first; click the **Accept** button. Click the **Next** button on the dialog in Fig 12.17b to display the dialog in Fig 12.17d and set the pressure to 1.0.

Click **Next** to return to the top-level dialog we started with in Fig 12.17a – it will display the new CFT that we just added (Fig 12.17e).

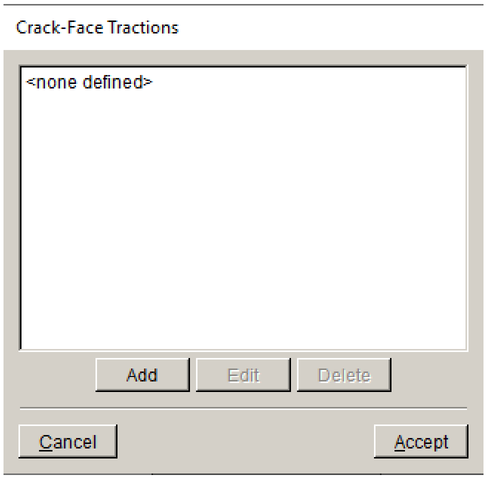

FRANC3D: Abaqus_crack_base.fdb

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

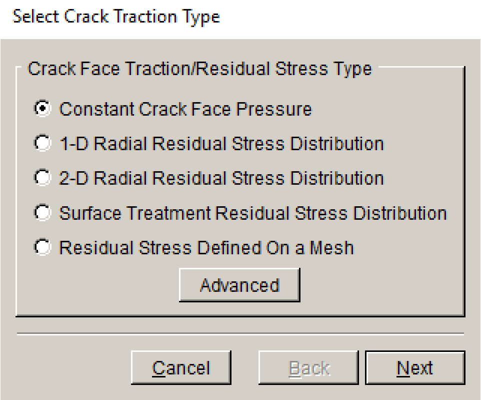
Define crack face traction.

Figure 12.16 FRANC3D Loads menu.


Crack-Face Tractions

<none defined>

Figure 12.17a FRANC3D CFT dialog.

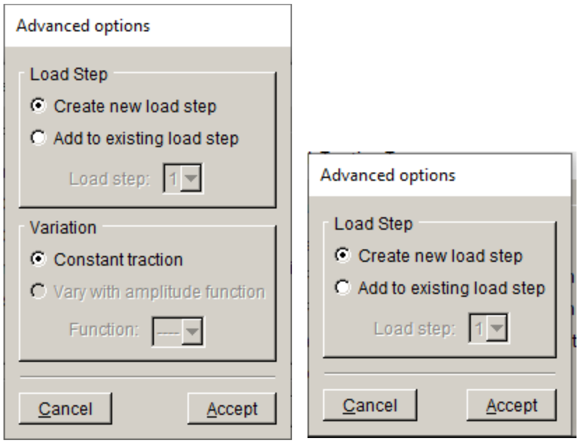


Select Crack Traction Type

Crack Face Traction/Residual Stress Type

- ● Constant Crack Face Pressure
- ○ 1-D Radial Residual Stress Distribution
- ○ 2-D Radial Residual Stress Distribution
- ○ Surface Treatment Residual Stress Distribution
- ○ Residual Stress Defined On a Mesh

Figure 12.17b Select CFT type dialog.



Advanced options

Load Step

- ● Create new load step
- ○ Add to existing load step

Load step: 1

Variation

- ● Constant traction
- ○ Vary with amplitude function

Function: ----

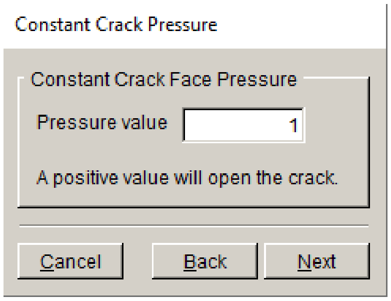
Advanced options

Load Step

- ● Create new load step
- ○ Add to existing load step

Load step: 1

Figure 12.17c CFT Advanced settings dialog; left for ABAQUS, right for ANSYS/NASTRAN.



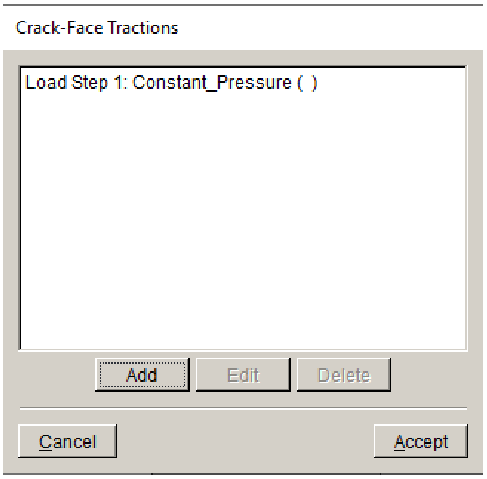
Constant Crack Pressure

Constant Crack Face Pressure

Pressure value 1

A positive value will open the crack.

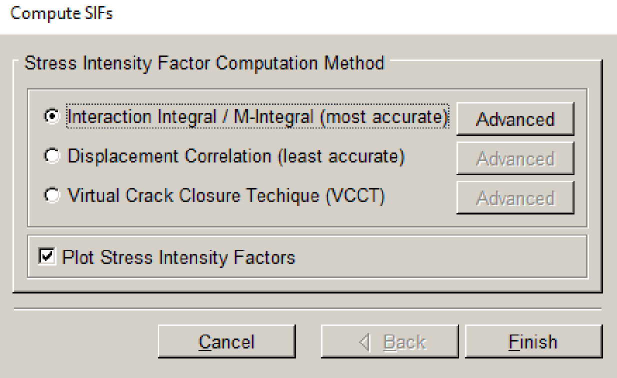
Figure 12.17d CFT constant pressure value dialog.


Crack-Face Tractions

Load Step 1: Constant_Pressure ( )

Figure 12.17e CFT dialog.

Run the static analysis and then compute SIFs. In the Compute SIFs dialog, Fig 12.18a, click on the **Advanced** button beside the M-integral label to display the dialog in Fig 12.18b. The Include Applied Crack Traction box should be checked automatically (as the M-integral includes a term that accounts for the tractions on the crack surface). Click **Accept** and then **Finish** to display the SIFs.


Compute SIFs

Stress Intensity Factor Computation Method

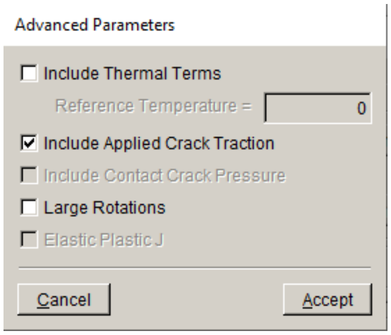
● Interaction Integral / M-Integral (most accurate)

○ Displacement Correlation (least accurate)

○ Virtual Crack Closure Techique (VCCT)

☑ Plot Stress Intensity Factors

Figure 12.18a Compute SIFs dialog.


Advanced Parameters

□ Include Thermal Terms

Reference Temperature = 0

☑ Include Applied Crack Traction

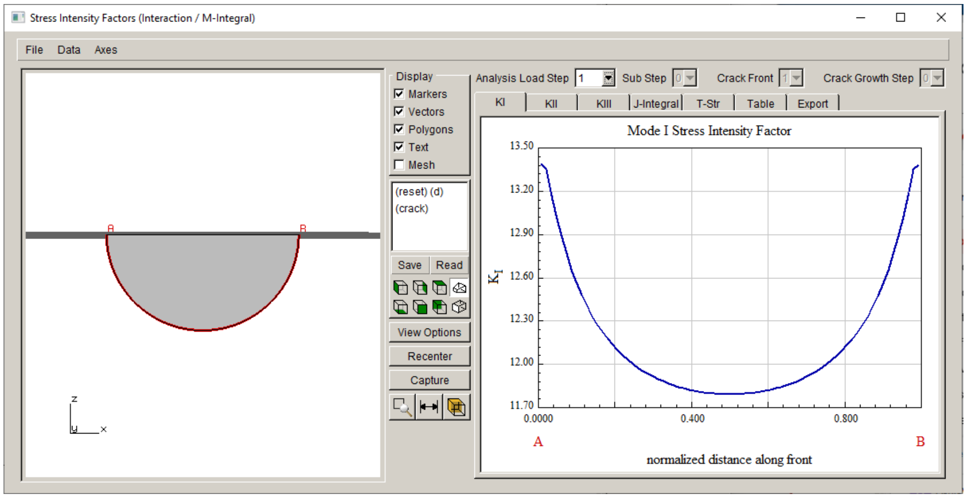
□ Include Contact Crack Pressure

□ Large Rotations

□ Elastic Plastic J

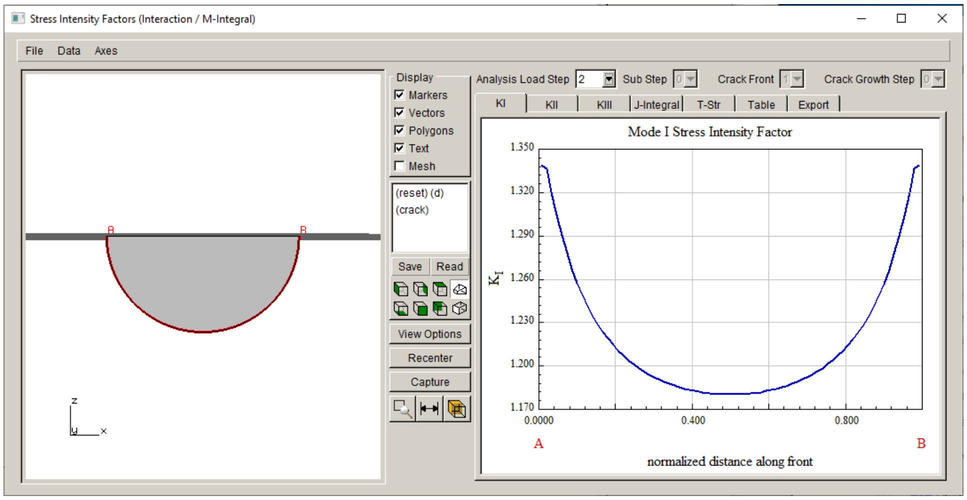
Figure 12.18b Compute SIFs Advanced Parameters dialog.

The Mode I SIFs for the first load step, Fig 12.19a, are the same as the base model (see Fig 12.15). The SIFs for load step 2, which is the CFT load, are shown in Fig 12.19b. The sum of the SIFs for the two load steps are shown in Fig 12.19c.



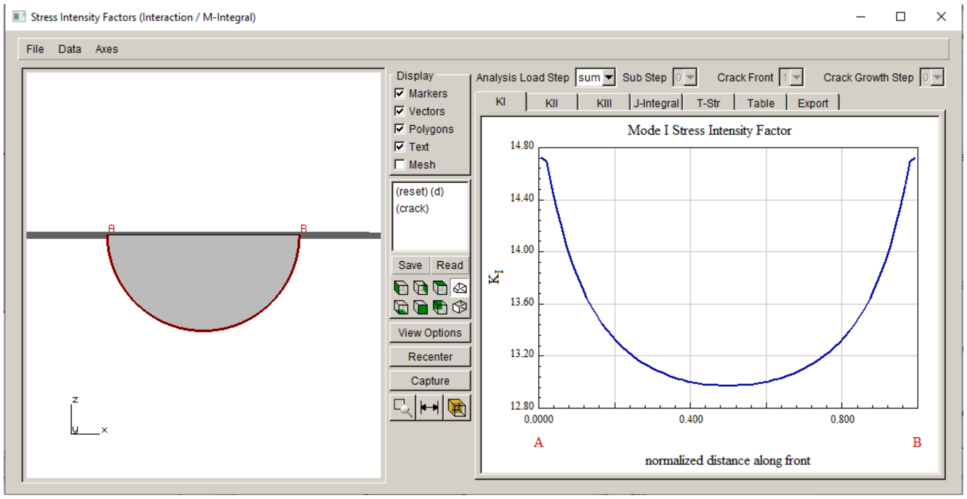
Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 12.19a Mode I SIFs for the first (base) load step.



Analysis Load Step 2 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 12.19b Mode I SIFs for the second extra (CFT) load step.



Stress Intensity Factors (Interaction / M-Integral)

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

(reset) (d)
(crack)

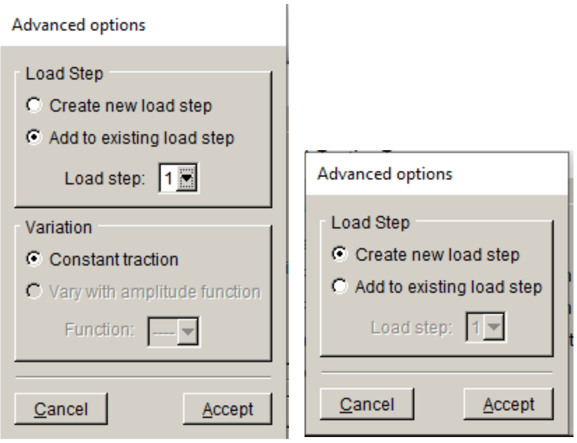
Analysis Load Step: sum
Sub Step: 0
Crack Front: 1
Crack Growth Step: 0

Figure 12.19c Sum of Mode I SIFs for the base plus the extra load steps.

To add the CFT to an existing load step, close the FRANC3D model we just analyzed and restart from the base model (surface crack in a cube with far-field tension in one load step).

Proceed to the **Loads** menu and add a CFT. When we get to the Advanced dialog (see Fig 12.17c), we switch the Load Step setting to <u>Add to existing load step</u>, Fig 12.20; there is only one load step in the model so the CFT will be added to it. Finish adding the CFT (constant pressure = 1) and run the static analysis and compute the SIFs.

Because we added the CFT to the base FE load step, the SIF dialog shows only one load step, Fig 12.21. The Mode I SIFs are the same as those shown in Fig 12.19c.



Advanced options

Load Step
○ Create new load step
● Add to existing load step
Load step: 1

Variation
● Constant traction
○ Vary with amplitude function
Function: ----

Advanced options

Load Step
● Create new load step
○ Add to existing load step
Load step: 1

Figure 12.20 CFT Advanced options dialog with Load Step set to “Add to existing”; left for ABAQUS, right for ANSYS/NASTRAN

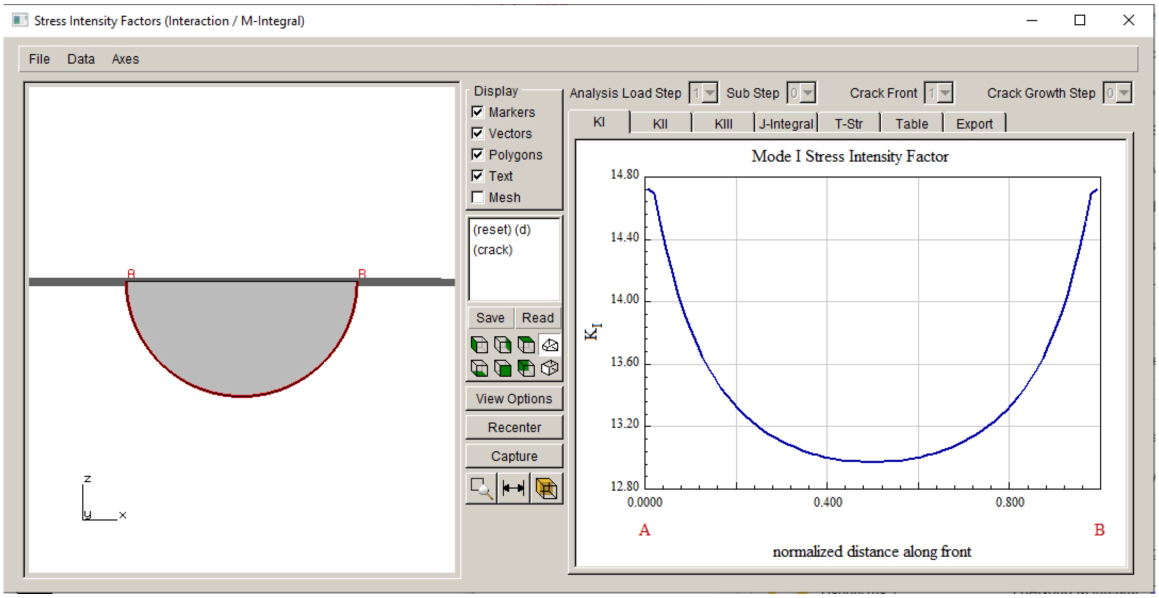
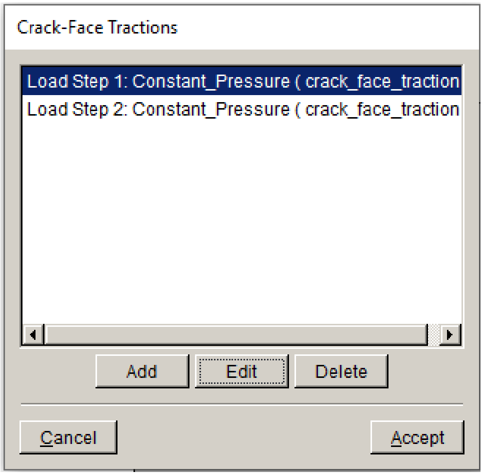


Figure 12.21 Mode I SIFs for the case where the CFT is added to the base load step.

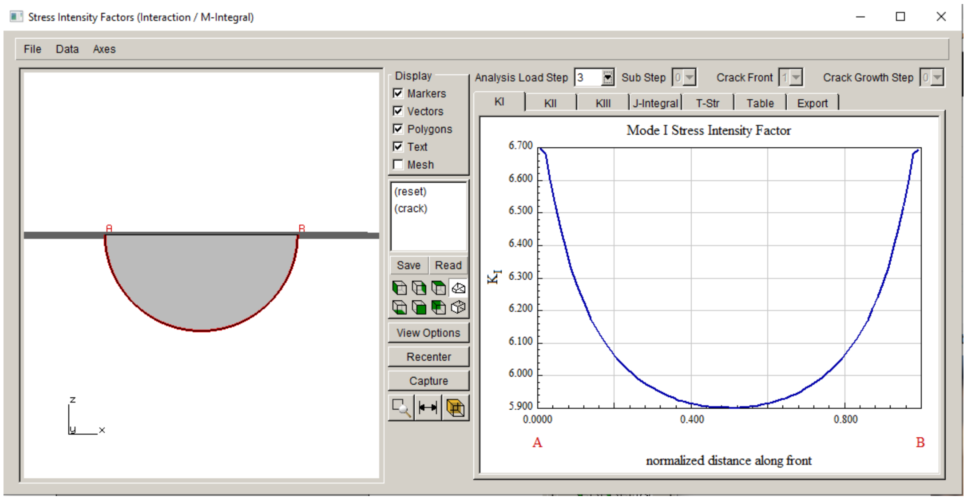
We can repeat the previous steps, but this time we add two CFTs to the base model, Fig 12.22. The first CFT is a constant pressure of 1.0 and the second is a constant pressure of 5.0. We could choose any of the CFT options, but the constant pressure is simple and exercises all the updated sections of the software.

Start by adding the CFTs as extra load steps; there will be two extra load steps after the base FE load step. The Mode I SIFs for the first two load steps are the same as in Figs 12.19a-b. The SIFs for load step 3 and for the sum of all three load steps are shown in Figs 12.23a-b.


Crack-Face Tractions

Load Step 1: Constant_Pressure (crack_face_traction
Load Step 2: Constant_Pressure (crack_face_traction

Figure 12.22 CFT dialog with two CFTs.

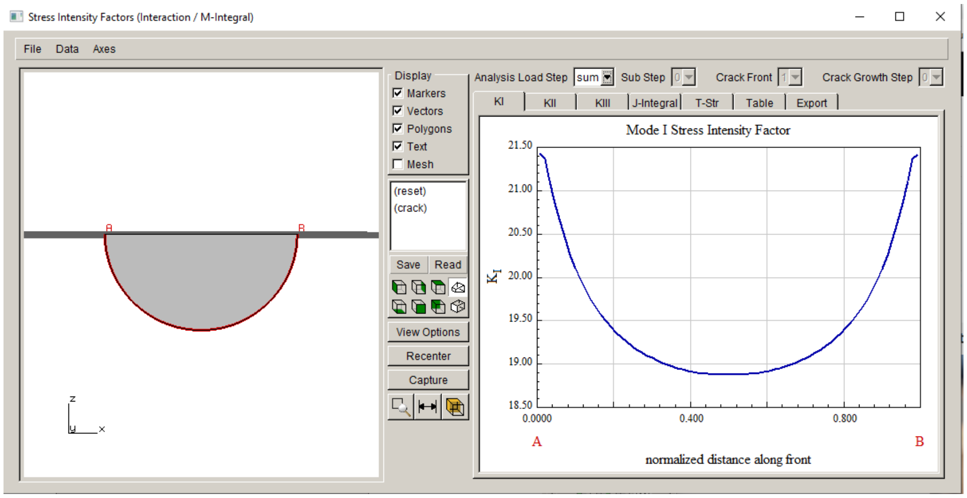


Stress Intensity Factors (Interaction / M-Integral)

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 3
Sub Step 0
Crack Front 1
Crack Growth Step 0

Figure 12.23a Mode I SIFs for third extra load step – CFT with constant pressure=5.



Stress Intensity Factors (Interaction / M-Integral)

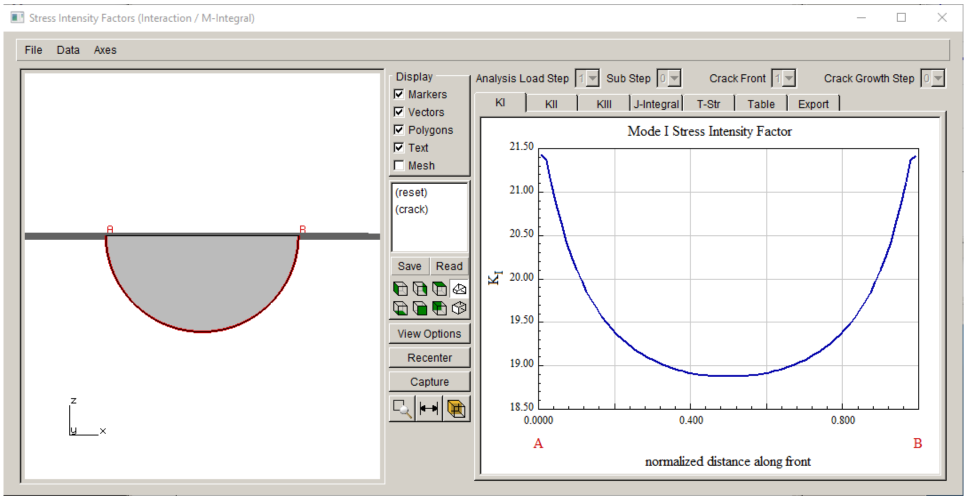
Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step sum
Sub Step 0
Crack Front 1
Crack Growth Step 0

Figure 12.23b Sum of Mode I SIFs for the base plus two extra (CFT) load steps.

Close this model and restart from the base model again. Now we add the same two CFTs, but we add them both to the existing FE load step. The Mode I SIFs for this case are shown in Fig 12.24. The values match those shown in Fig 12.23b.

If there are doubts about the values, one can plot the DC SIFs, Fig 12.25. The DC SIFs depend only on the displacements from the analysis code and should match the M-integral SIFs as shown in Section 12.1.

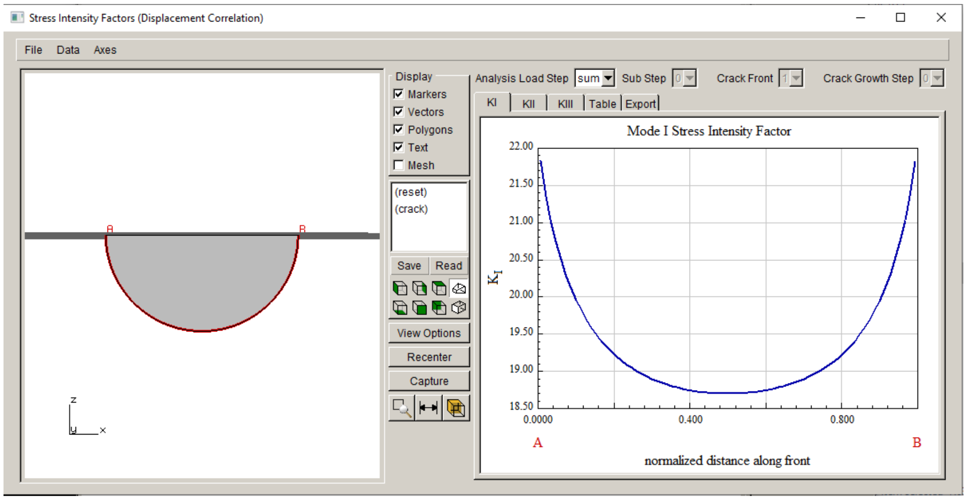


Stress Intensity Factors (Interaction / M-Integral)

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 12.24 Mode I SIFs for the case where the two CFTs are added to the base load step.



Stress Intensity Factors (Displacement Correlation)

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step sum | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

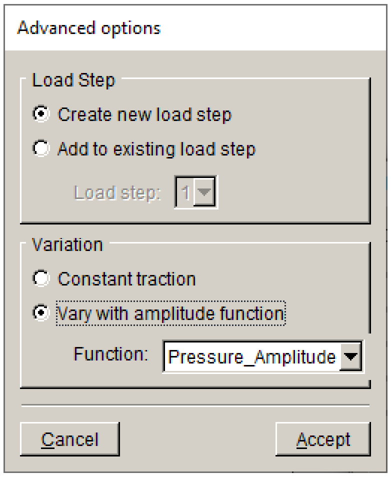
Figure 12.25 Mode I SIFs using displacement correlation for the case where the two CFTs are added to the base load step.

One should be able to perform any combination of adding extra load step CFTs and/or adding CFTs to existing load steps. However, one should not add CFTs to an extra CFT load step.

Note for NASTRAN users: one cannot add multiple CFTs to an existing load step; it is possible to do so, but restarts will not produce the correct SIFs; a warning message is printed to the console/terminal window if this is attempted.

12.3 CFT with ABAQUS Amplitude

For ABAQUS models, if a load step includes an *Amplitude, the amplitude data can be attached the CFT load as well, but there are limitations. First, only amplitudes that are defined using total time with tabular data are supported. FRANC3D will read and store all *Amplitudes from an *.inp* file; any that do not match the above conditions are simply passed through. Amplitudes that match the above conditions will be listed in the CFT Advanced options dialog, Fig 12.26.



Advanced options

Load Step

● Create new load step

○ Add to existing load step

Load step: 1

Variation

○ Constant traction

● Vary with amplitude function

Function: Pressure_Amplitude

Figure 12.26 CFT Advanced options dialog with ABAQUS amplitude selection.

The amplitude time and values are accessed when computing SIFs. The CFT values are scaled based on the amplitude value for a given time. The time is stored in the *.dtp* results file.

12.4 CFT with Temperature

In older versions of FRANC3D, an extra CFT load step was not allowed to have thermal stress/strain included. Now one can specify whether the extra load step should allow a non-zero coefficient of thermal expansion.

Note for NASTRAN users: material properties cannot be changed from one load step to the next.

If there are temperatures in the model and if there are temperature dependent material properties, the extra CFT load step can include temperatures so that the correct material property data is extracted. The analysis of this load step in the analysis code should include the temperatures. The thermal expansion coefficients can be set to zero (for ANSYS and ABAQUS) or can be retained from the previous load step, which will allow thermal stress/strain.

To include thermal stress/strain with a CFT, one can also add the CFT to an existing load step, which was described in the previous section.

The traditional use of the CFT was for linear superposition, where one is taking the stress state from a complex model configuration and applying that as traction to the crack face; the complex stress state is assumed to include all the thermal stress/strain. However, in some cases, users want to use CFTs in non-traditional ways, so the ability to add CFT to an existing load step was added (previous section), along with the ability to turn on/off thermal conditions for extra CFT load steps.

The addition of temperature to a CFT load step can be confusing. In this section, we demonstrate the current FRANC3D capabilities along with the user-interface for including temperature.

We use the same cube model from the first section, but we add temperature to the uncracked model. Fig 12.27 shows the ANSYS version of the model with the ‘base’ loading (uniform traction on the top surface) and with a temperature gradient from left to right, going from 1000 to 100 degrees. The material properties are set such that E@1000 degrees=5000 and E@0 degrees=10000; NU=0.3 and CTE (ALPX) = 3e-7 for all temperatures. The deformation for this model is shown in the left panel of Fig 12.28, which can be compared to the original model (without temperature) deformation, which shown in the right panel of Fig 12.28.

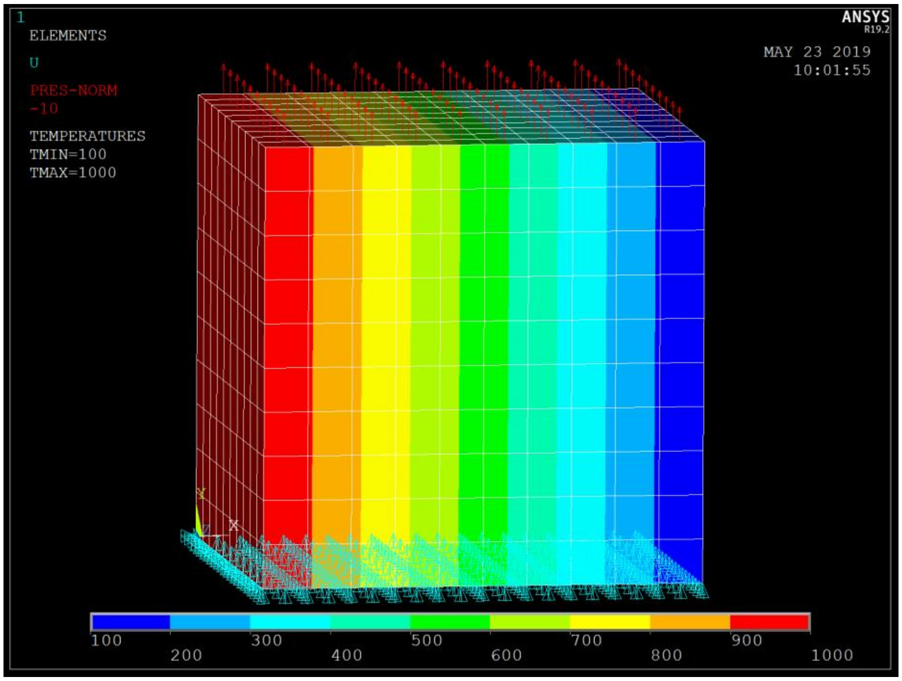


Figure 12.27 Uncracked cube model with base loading plus a temperature gradient.

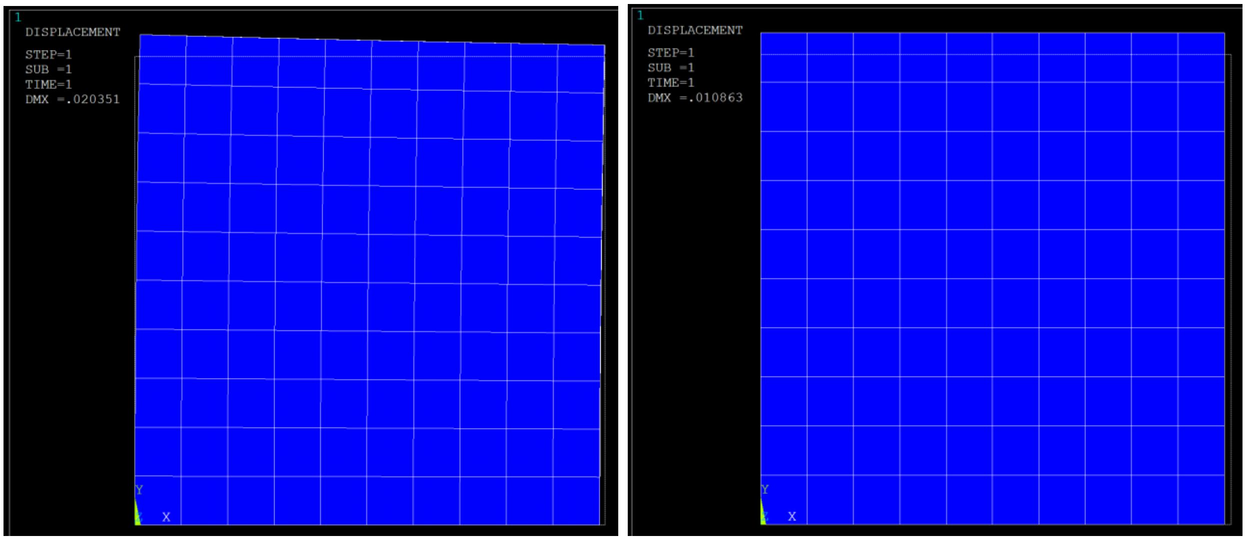
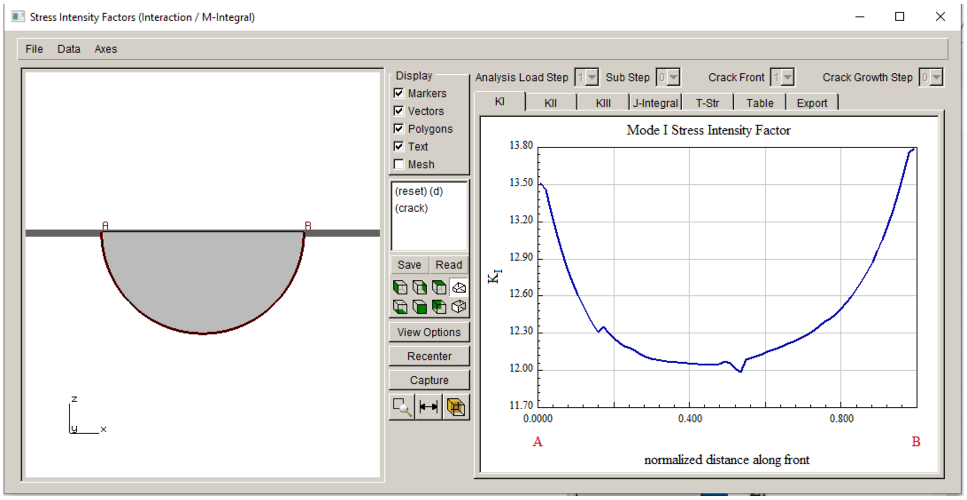


Figure 12.28 Uncracked cube model deformation for the case with temperature gradient (left) and for the original model without temperature (right).

This cube-with-temperature model is imported into FRANC3D. The same penny-shaped surface crack is inserted as in the previous section. The base SIFs are shown in Fig 12.29. The effect of the temperature and temperature dependent modulus produces slightly higher SIFs (compared to Fig 12.15) and an unsymmetric curve. The temperature varies along the front from 640 to 460. (The small 'jumps' in the SIF curve are due to the coarse initial mesh and corresponding jumps in nodal temperatures.)


Stress Intensity Factors (Interaction / M-Integral)

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

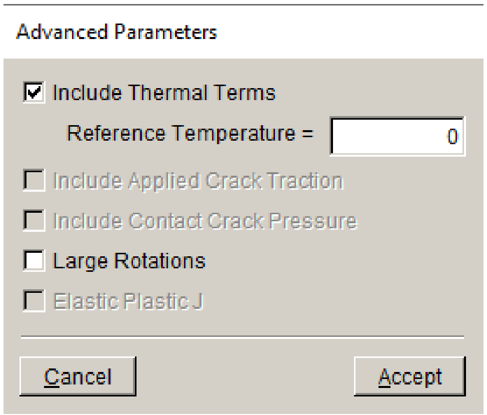
Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

(reset) (d)
(crack)

Figure 12.29 Mode I SIFs for the base-with-temperature model.

Note that the M-integral includes thermal terms, Fig 12.30; the reference temperature is 0 degrees (this is read from the input FE model). To verify that the thermal terms are included, we can plot the DC SIFs, Fig 12.31, which shows that the SIFs match.

We can also plot the M-integral SIFs with thermal terms turned off, by unchecking the box in Fig 12.30. The SIFs in Fig 12.32 are quite different from the DC SIFs, which indicates that they are not correct.



Advanced Parameters

☑ Include Thermal Terms

Reference Temperature = 0

□ Include Applied Crack Traction

□ Include Contact Crack Pressure

□ Large Rotations

□ Elastic Plastic J

Figure 12.30 M-integral advance parameters dialog.

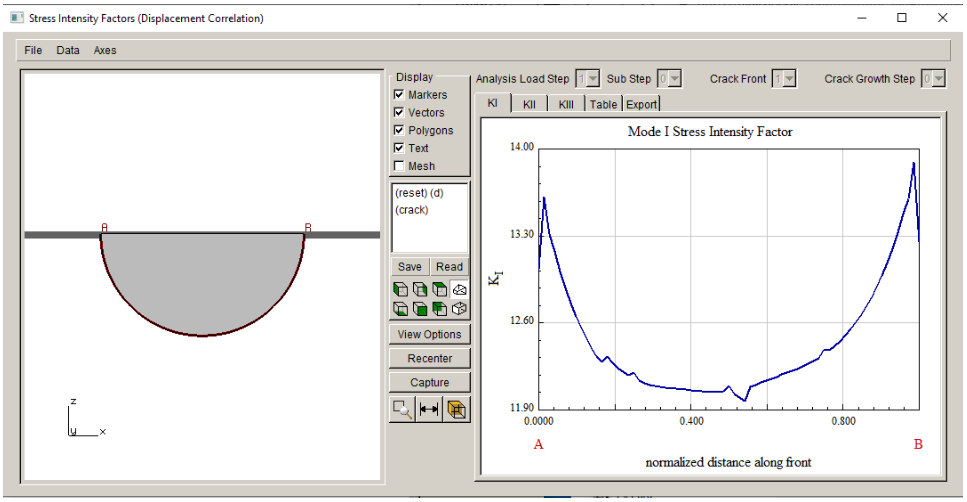
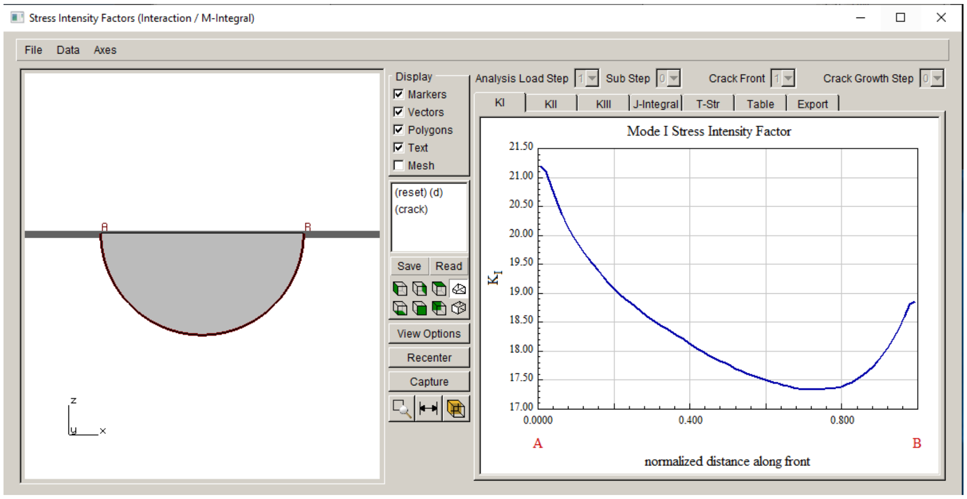


Figure 12.31 Mode I SIFs for the base-with-temperature model using DC.


Stress Intensity Factors (Interaction / M-Integral)
File Data Axes
Display
(reset) (d)
(crack)
Save Read
View Options
Recenter
Capture
KI KII KIII J-Integral T-Str Table Export
Mode I Stress Intensity Factor
21.50
21.00
20.50
20.00
19.50
19.00
18.50
18.00
17.50
17.00
K_I
0.0000
0.400
0.800
A
B
normalized distance along front
z
y
x

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

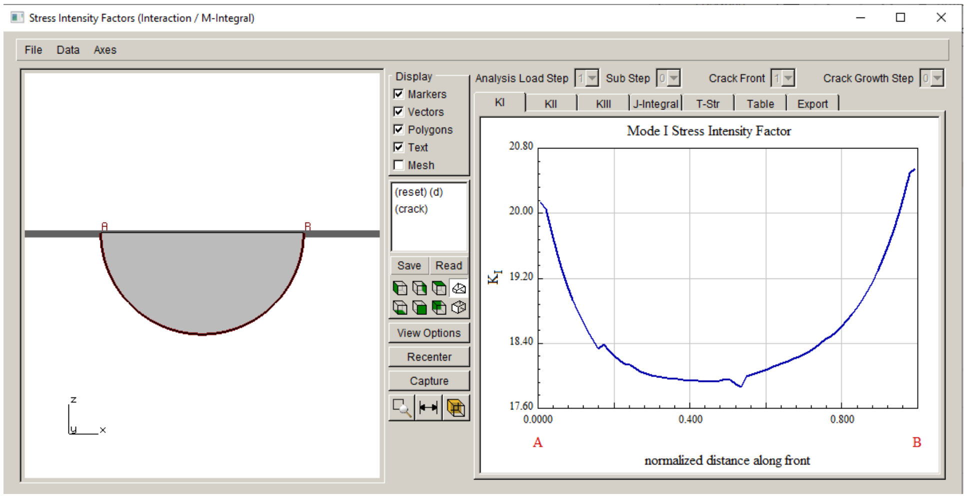
Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 12.32 Mode I SIFs for the base-with-temperature model – M-integral thermal terms off.

12.4.1 Adding CFT to Existing Load Step

Adding a CFT to an existing load step means that the CFT will simply be included with other loads, constraints, temperatures *etc.* for that load step. Thus, one does not have to specify how temperature will be included. To demonstrate, restart with the base-with-temperature model that was just analyzed.

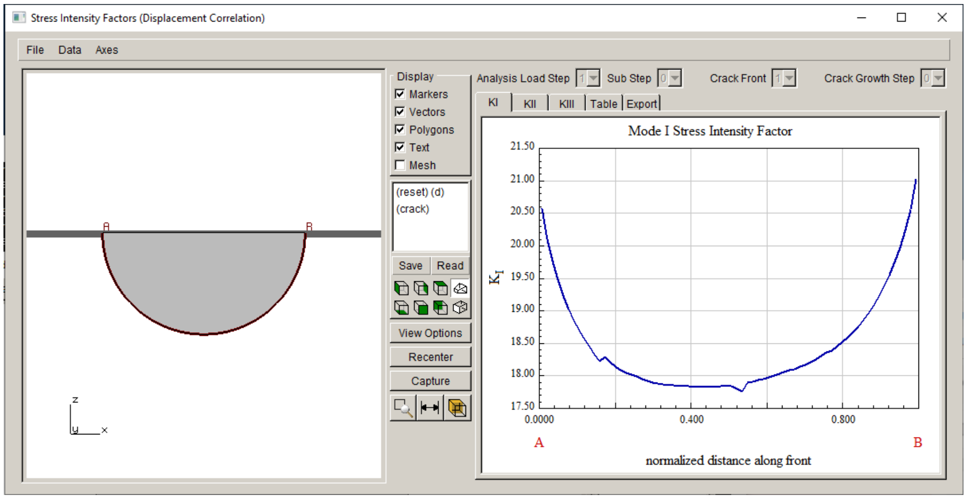
We add a CFT load step, using a constant pressure = 5, to the existing load step (see Fig 12.20); there is only one load step in the model. The resulting SIFs for combined load step are shown in Figs 12.33a-b. Fig 12.33a shows the M-integral SIFs and Fig 12.33b shows the DC SIFs; the values match.


Stress Intensity Factors (Interaction / M-Integral)
File Data Axes
Display
(reset) (d)
(crack)
Save Read
View Options
Recenter
Capture
KI KII KIII J-Integral T-Str Table Export
Mode I Stress Intensity Factor
20.80
20.00
19.20
18.40
17.60
K_I
0.0000
0.400
0.800
A
B
normalized distance along front
z
y
x

☑ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 12.33a Mode I SIFs for the combined base plus CFT load using M-integral.



Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh

(reset) (d)
(crack)

Analysis Load Step 1, Sub Step 0, Crack Front 1, Crack Growth Step 0

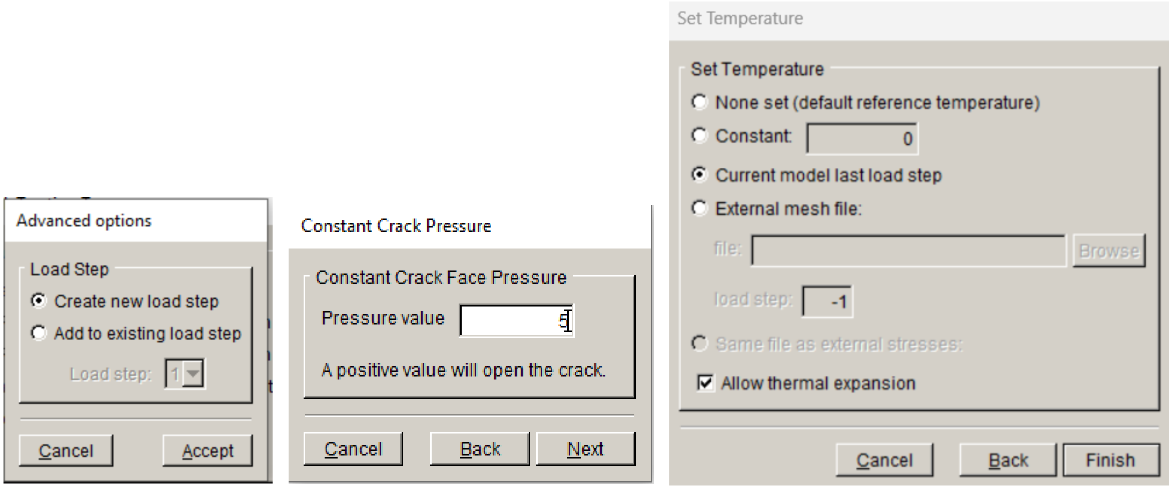
Figure 12.33b Mode I SIFs for the combined base plus CFT load using DC.

12.4.2 Adding CFT to an Extra Load Step

When we add a CFT as an extra load step, we need to define the temperature settings. Restart from the base-with-temperature model and add an extra CFT load. Fig 12.34 shows the dialogs for adding the extra CFT load step, using a constant pressure = 5.

The temperature is set so that nodal temperatures from the previous load step are applied to the CFT load step, and thermal stress/strain is allowed – the ANSYS ALPX values are not set to zero for the CFT load step.

The resulting Mode I SIFs are shown in Figs 12.35a-c. Fig 12.35a shows the sum; the values are reasonably close to the values in Fig 12.33a. Fig 12.35b shows the base load step SIF values, which are the same as in Fig 12.29. Fig 12.35c shows the SIFs for the extra CFT load step.



Advanced options

Load Step
● Create new load step
○ Add to existing load step
Load step: 1

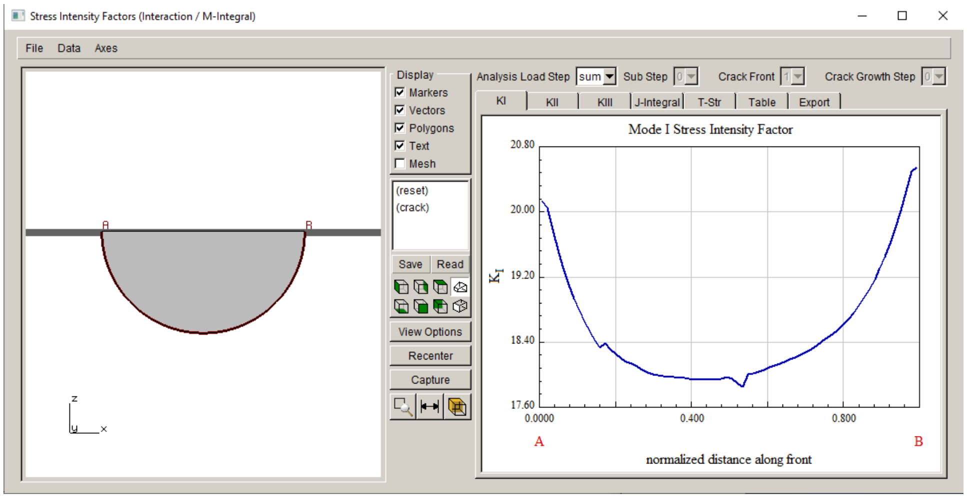
Constant Crack Pressure

Constant Crack Face Pressure
Pressure value 5
A positive value will open the crack.

Set Temperature

Set Temperature
○ None set (default reference temperature)
○ Constant: 0
● Current model last load step
○ External mesh file:
file:
load step: -1
○ Same file as external stresses:
☑ Allow thermal expansion

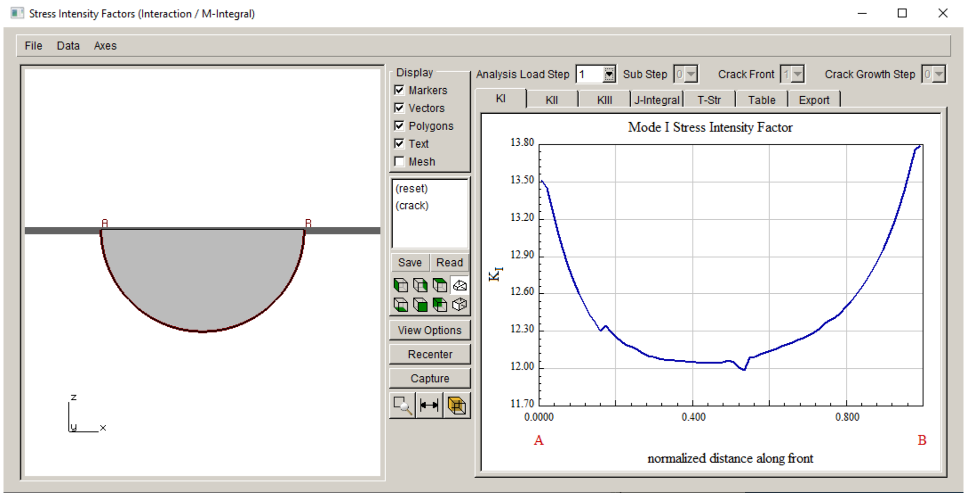
Figure 12.34 Mode I SIFs for the combined base plus CFT load using displacement correlation.



☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step: sum
Sub Step: 0
Crack Front: 1
Crack Growth Step: 0

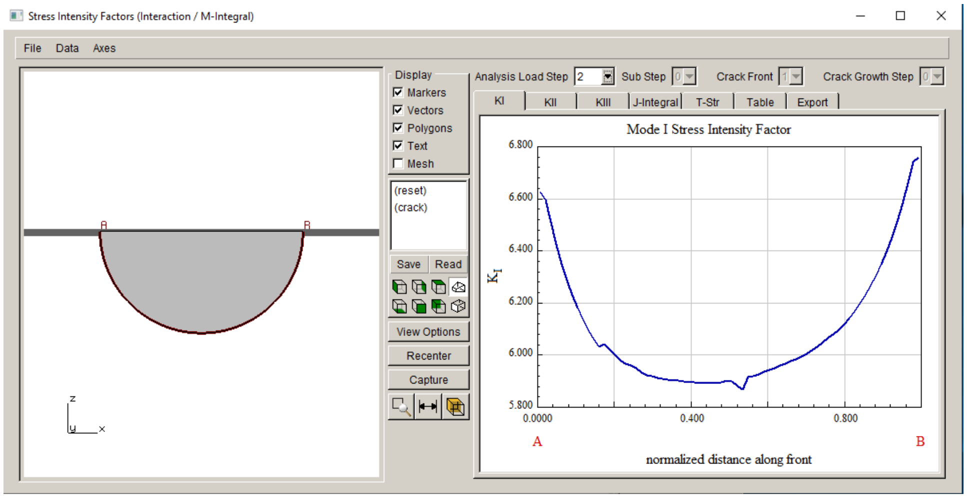
Figure 12.35a Mode I SIFs for the sum of the base and the extra CFT load.



☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step: 1
Sub Step: 0
Crack Front: 1
Crack Growth Step: 0

Figure 12.35b Mode I SIFs for the the base load.



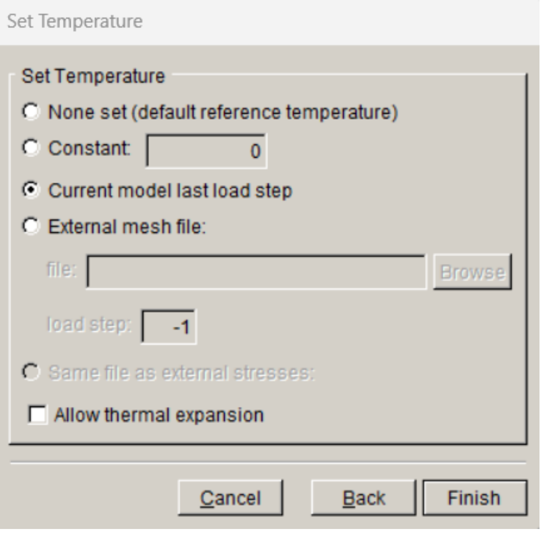
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step: 2
Sub Step: 0
Crack Front: 1
Crack Growth Step: 0

Figure 12.35c Mode I SIFs for the extra CFT load.

12.4.3 Adding CFT to an Extra Load Step – No Thermal Expansion

The next analysis starts from the base-with-temperature model and adds an extra CFT load step, with temperatures set to the previous load step, but the Allow thermal expansion is turned off, Fig 12.36. The resulting Mode I SIF sum curve is shown in Fig 12.37a. The SIFs are the same as in Fig 12.35a. The displacement correlation SIFs are shown in Fig 12.37b; the M-integral and DC SIFs are comparable.



Set Temperature

Set Temperature

○ None set (default reference temperature)

○ Constant: 0

● Current model last load step

○ External mesh file:

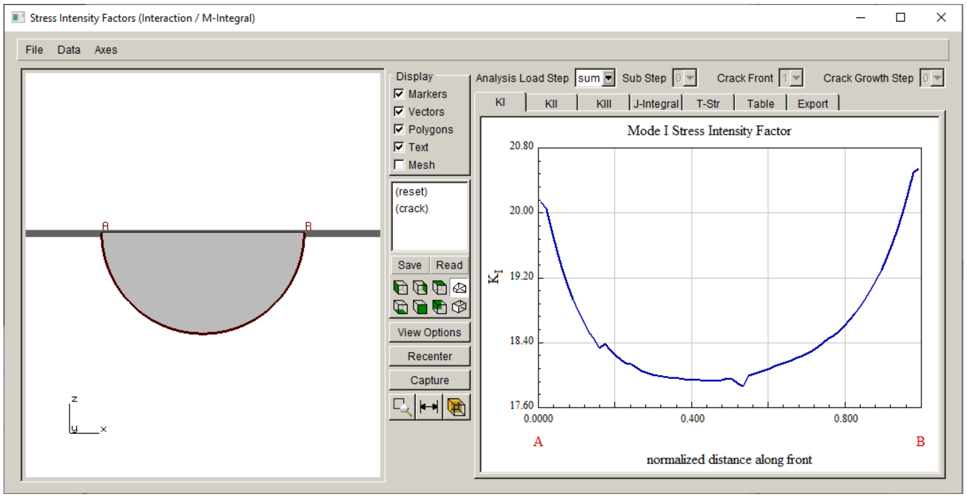
file: Browse

load step: -1

○ Same file as external stresses:

□ Allow thermal expansion

Figure 12.36 CFT Set Temperature dialog with thermal expansion off.



Stress Intensity Factors (Interaction / M-Integral)

Display

☑ Markers

☑ Vectors

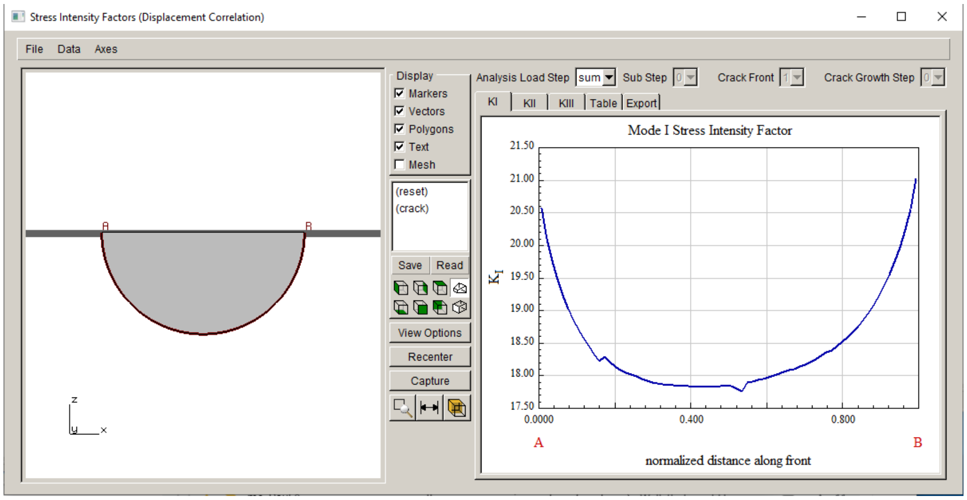
☑ Polygons

☑ Text

□ Mesh

Analysis Load Step: sum Sub Step: 0 Crack Front: 1 Crack Growth Step: 0

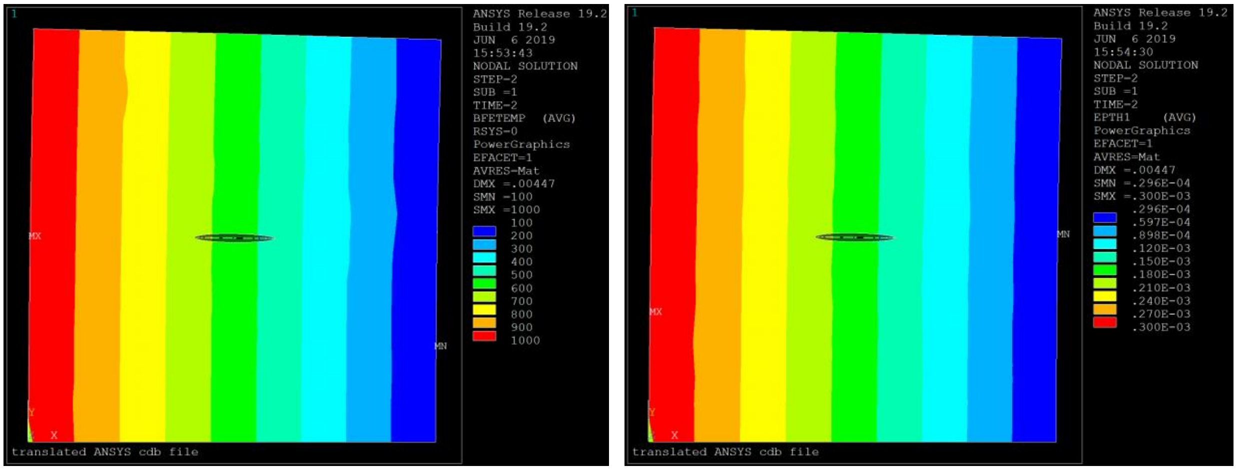
Figure 12.37a Mode I SIFs for the sum of the base and the extra CFT load; thermal expansion is turned off.



Analysis Load Step: sum
Sub Step: 0
Crack Front: 1
Crack Growth Step: 0

Figure 12.37b Mode I SIFs for the sum of the base and the extra CFT load using displacement correlation; thermal expansion is turned off.

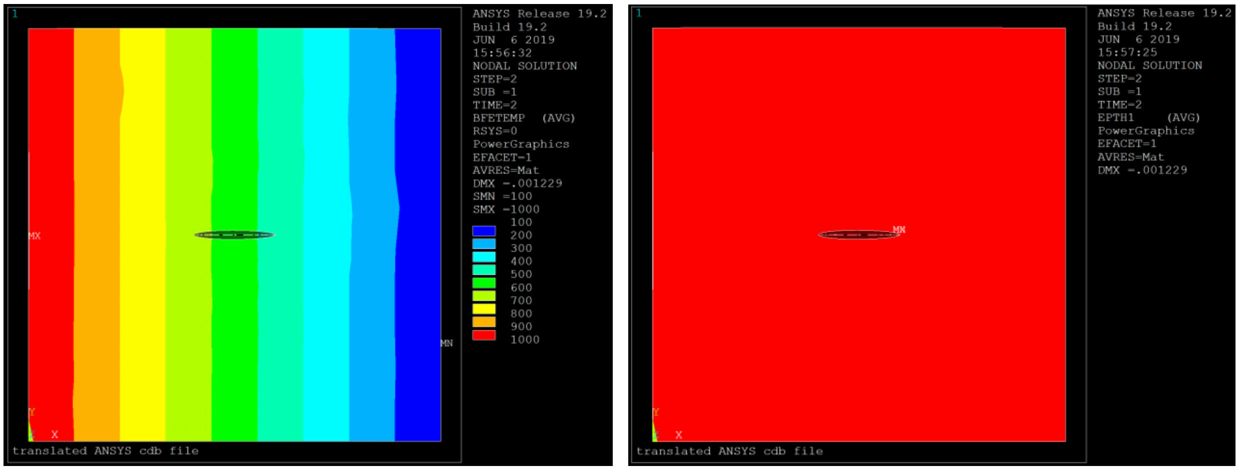
Fig 12.38a shows the nodal temperature and principal thermal strain contours for the case where thermal expansion is turned on. Fig 12.38b shows the nodal temperature and principal thermal strain contours for the case where thermal expansion is turned off.



```
ANSYS Release 19.2
Build 19.2
JUN  6 2019
15:53:43
NODAL SOLUTION
STEP-2
SUB =1
TIME-2
BFETEMP   (AVG)
RSYS-0
PowerGraphics
EFACET=1
AVRES-Mat
DMX =.00447
SMN -100
SMX =1000
     100
     200
     300
     400
     500
     600
     700
     800
     900
     1000
```

```
ANSYS Release 19.2
Build 19.2
JUN  6 2019
15:54:30
NODAL SOLUTION
STEP-2
SUB =1
TIME-2
EPTH1      (AVG)
PowerGraphics
EFACET=1
AVRES=Mat
DMX =.00447
SMN =.296E-04
SMX =.300E-03
      .296E-04
      .597E-04
      .898E-04
      .120E-03
      .150E-03
      .180E-03
      .210E-03
      .240E-03
      .270E-03
      .300E-03
```

Figure 12.38a Temperature and thermal strain contours for non-zero ALPX for CFT load step (displacement magnification = 100).

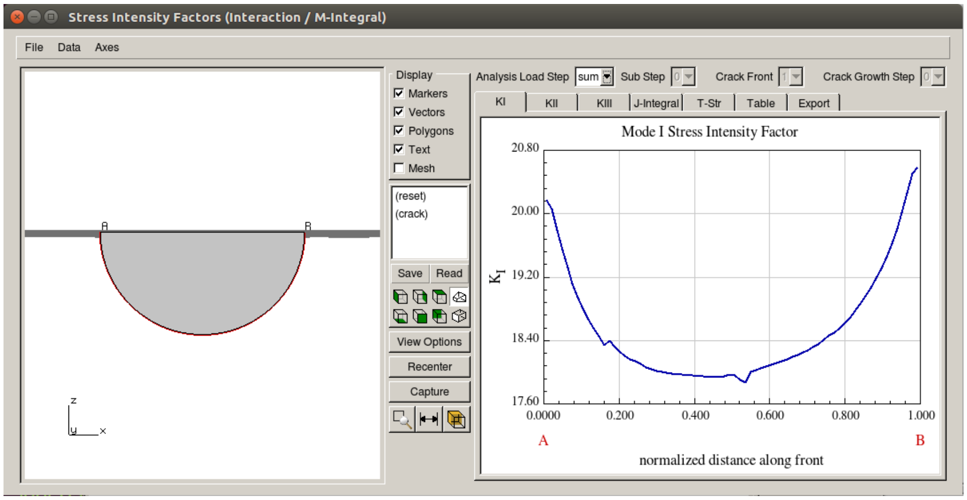


```
ANSYS Release 19.2
Build 19.2
JUN  6 2019
15:56:32
NODAL SOLUTION
STEP=2
SUB =1
TIME=2
BFETEMP  (AVG)
RSYS=0
PowerGraphics
EFACET=1
AVRES=Mat
DMX =.001229
SMN =100
SMX =1000
```

```
ANSYS Release 19.2
Build 19.2
JUN  6 2019
15:57:25
NODAL SOLUTION
STEP=2
SUB =1
TIME=2
EPTH1     (AVG)
PowerGraphics
EFACET=1
AVRES=Mat
DMX =.001229
```

Figure 12.38b Temperature and thermal strain contours for case with zero ALPX for CFT load step (displacement magnification = 100).

Using ABAQUS, the SIFs for the model with CTE set to zero are shown in Fig 12.39; the values match those from the ANSYS analysis. Figs 12.40a-b show the ABAQUS temperature and thermal strain contours for the non-zero and zero CTE analyses for the CFT load step. The ABAQUS contours match the ANSYS contours (see Fig 12.38).



Analysis Load Step: sum
Sub Step: 0
Crack Front: 1
Crack Growth Step: 0

Figure 12.39 ABAQUS analysis Mode I SIFs for the sum of the base and the extra CFT load; thermal expansion is turned off.

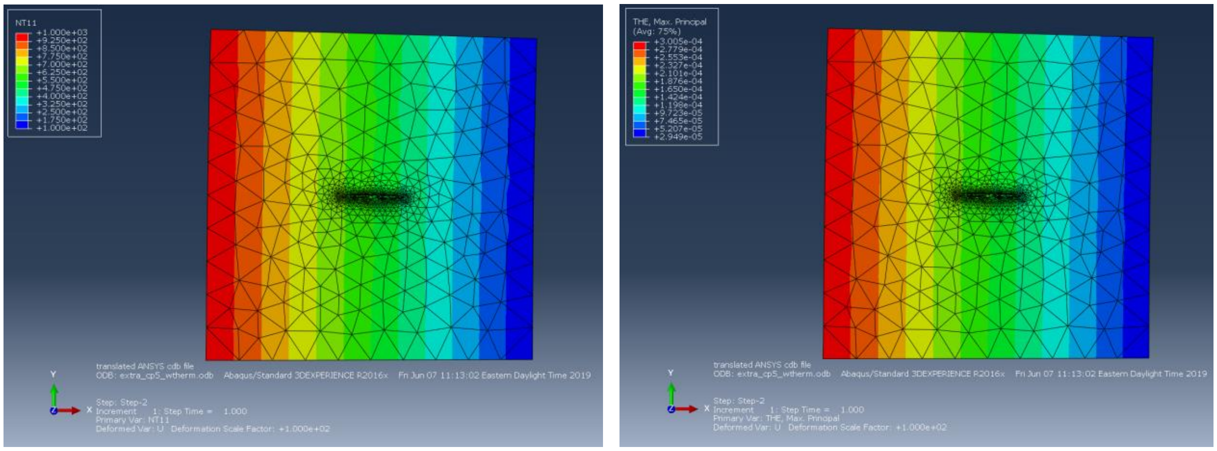

NT11
+1.000e+03
+9.250e+02
+8.500e+02
+7.750e+02
+7.000e+02
+6.250e+02
+5.500e+02
+4.750e+02
+4.000e+02
+3.250e+02
+2.500e+02
+1.750e+02
+1.000e+02
translated ANSYS cdb file
ODB: extra_cp5_wtherm.odb Abaqus/Standard 3DEXPERIENCE R2016x Fri Jun 07 11:13:02 Eastern Daylight Time 2019
Y
X
Step: Step-2
Increment 1: Step Time = 1.000
Primary Var: NT11
Deformed Var: U Deformation Scale Factor: +1.000e+02
THE, Max. Principal
(Avg: 75%)
+3.005e-04
+2.779e-04
+2.553e-04
+2.327e-04
+2.101e-04
+1.876e-04
+1.650e-04
+1.424e-04
+1.198e-04
+9.723e-05
+7.465e-05
+5.207e-05
+2.949e-05
translated ANSYS cdb file
ODB: extra_cp5_wtherm.odb Abaqus/Standard 3DEXPERIENCE R2016x Fri Jun 07 11:13:02 Eastern Daylight Time 2019
Y
X
Step: Step-2
Increment 1: Step Time = 1.000
Primary Var: THE, Max. Principal
Deformed Var: U Deformation Scale Factor: +1.000e+02

Figure 12.40a ABAQUS temperature and thermal strain contours for non-zero CTE for CFT load step (displacement magnification = 100).

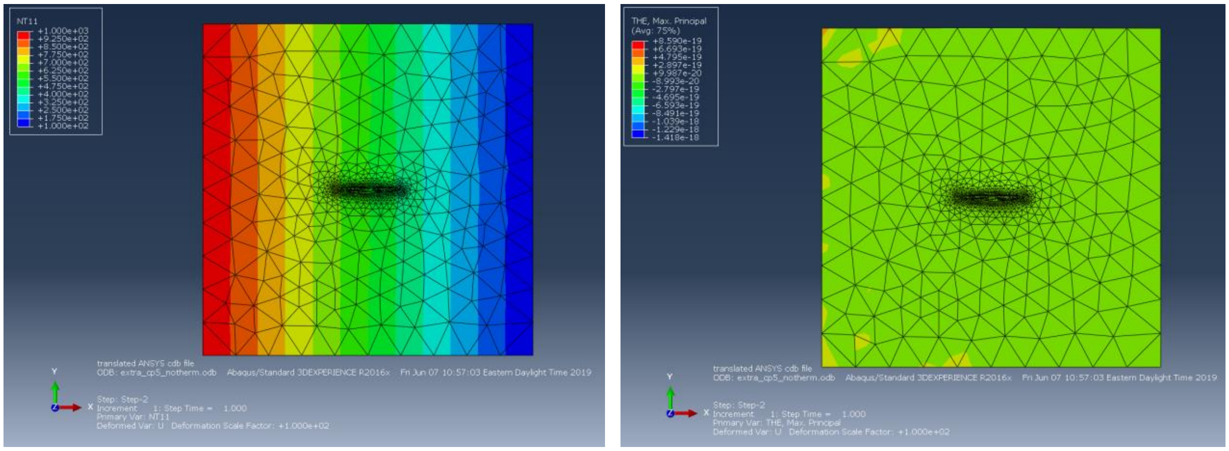

NT11
+1.000e+03
+9.250e+02
+8.500e+02
+7.750e+02
+7.000e+02
+6.250e+02
+5.500e+02
+4.750e+02
+4.000e+02
+3.250e+02
+2.500e+02
+1.750e+02
+1.000e+02
translated ANSYS cdb file
ODB: extra_cp5_notherm.odb Abaqus/Standard 3DEXPERIENCE R2016x Fri Jun 07 10:57:03 Eastern Daylight Time 2019
Y
X
Step: Step-2
Increment 1: Step Time = 1.000
Primary Var: NT11
Deformed Var: U Deformation Scale Factor: +1.000e+02
THE, Max. Principal
(Avg: 75%)
+8.590e-19
+6.693e-19
+4.795e-19
+2.897e-19
+9.987e-20
-8.993e-20
-2.797e-19
-4.695e-19
-6.593e-19
-8.491e-19
-1.039e-18
-1.229e-18
-1.418e-18
translated ANSYS cdb file
ODB: extra_cp5_notherm.odb Abaqus/Standard 3DEXPERIENCE R2016x Fri Jun 07 10:57:03 Eastern Daylight Time 2019
Y
X
Step: Step-2
Increment 1: Step Time = 1.000
Primary Var: THE, Max. Principal
Deformed Var: U Deformation Scale Factor: +1.000e+02

Figure 12.40b ABAQUS temperature and thermal strain contours for case with zero CTE for CFT load step (displacement magnification = 100).

NASTRAN does not allow one to edit material properties from one load step (subcase) to the next. As seen in the ANSYS and ABAQUS analyses, however, setting the CTE to zero does not change the resulting SIFs (at least under these boundary conditions). NASTRAN users must be aware of the fact that unchecking the Allow thermal expansion check box in Fig 12.36 will have no effect.

12.5 Creating a User-Defined Residual Stress

In some cases, a user might have a 2D planar surface with residual stress values at discrete points. FRANC3D cannot use this directly; instead, one can create a 3D mesh and results file from the 2D data. In this section, we provide a simple Python script that imports a 2D grid of stress and writes an ABAQUS *.inp* file and corresponding *.dtp* file with nodal stress.

Fig 12.41 shows a regular grid with x,y coordinates. A table of “normal” stress values corresponding to the grid points can be stored in a simple ASCII text file with the following format:

blnk	0	1	2	3
0	1.1	1.2	1.0	0.9
1	1.	1.1	0.9	0.8
2	.9	1.	0.9	0.75

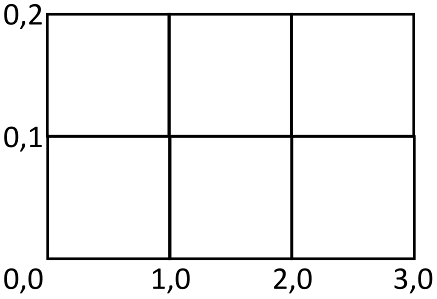


Figure 12.41 Simple rectangular grid.

The Python script imports the 2D grid data and creates a single layer of 3D brick elements, Fig 12.42. The stress at the 2D grid points is constant through the layer. An ABAQUS *.inp* file of the mesh is saved, and the nodal stress is written to a *.dtp* file (this is the ‘neutral’ format that FRANC3D uses for all results).

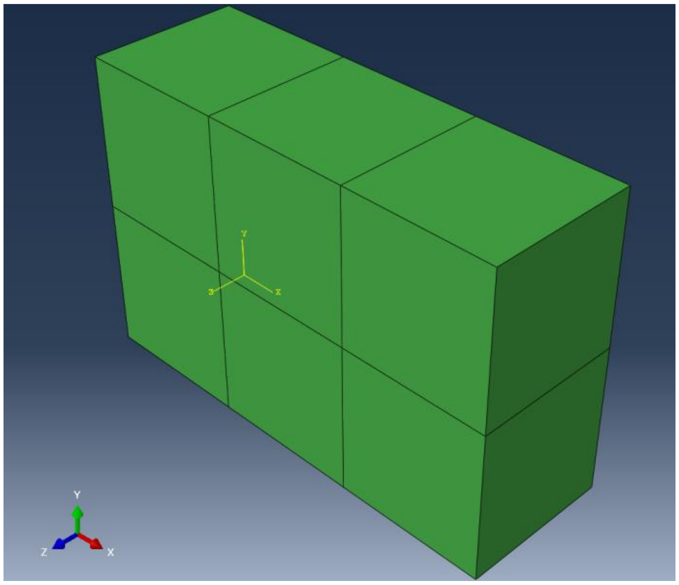


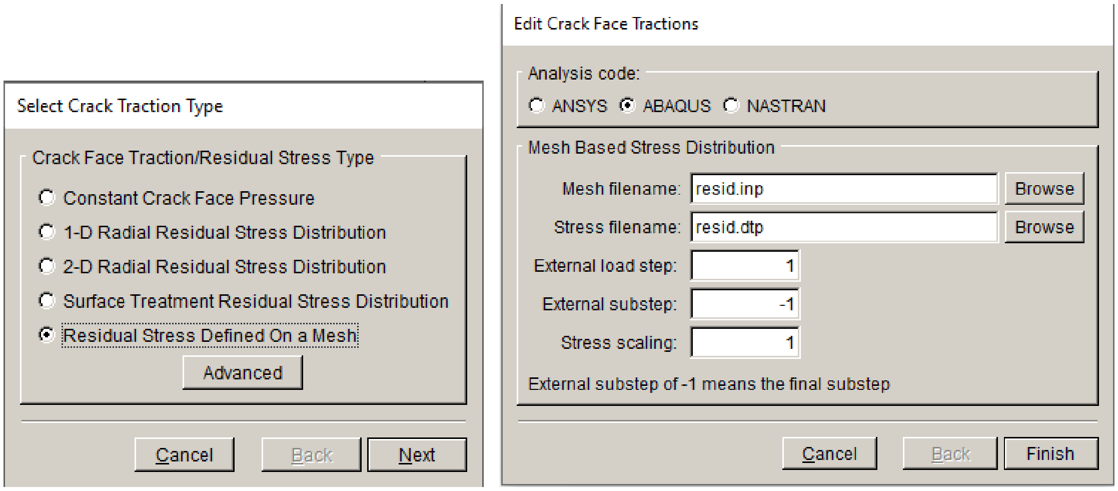
Figure 12.42 3D grid of ABAQUS nodes and elements.

```
#!/usr/bin/python

import sys
import string
import os
import shutil

class Resid:

    # reads file
    def __init__(self,fin):
        """Read the file:
        """

        self.Xc = []
        self.Yc = []
        self.Res = {}

        buff = fin.readline()
        # first line has x coords
        vals = buff.split()
        ncol = len(vals)

        for i in range(1,len(vals)):
            self.Xc.append(float(vals[i]))

        yxi = 0
        while buff != None:
            buff = fin.readline()
            vals = buff.split()
            if len(vals) != ncol:
                break
            self.Yc.append(float(vals[0]))
            for i in range(1,len(vals)):
                self.Res[yxi] = float(vals[i])
                yxi += 1

    def GetXc(self):
        return self.Xc
    def GetYc(self):
        return self.Yc
    def GetRes(self):
        return self.Res
```

```
if __name__ == '__main__':

    verbose = None
    if "-v" in sys.argv: verbose = 1

    # define data

    nodes = {}
    offset = {}
    zdim = 1.0

    # read the input file

    fin = open(sys.argv[1])
    inp = Resid(fin)
    fin.close()

    xc = inp.GetXc()
    yc = inp.GetYc()

    nid = 1
    for y in yc:
        for x in xc:
            nodes[nid] = [x,y,0.0]
            nid += 1
    tot_nd = nid-1

    ofn = tot_nd+1
    for n in nodes:
        offset[ofn] = [nodes[n][0],nodes[n][1],zdim]
        ofn += 1

    fout = open(sys.argv[2],'w')
    fout.write("*Node \n")
    for n in nodes:
        fout.write("%d, %f, %f, %f \n"%(n,nodes[n][0],nodes[n][1],nodes[n][2]))
    for n in offset:
        fout.write("%d, %f, %f, %f \n"%(n,offset[n][0],offset[n][1],offset[n][2]))

    fout.write("*Element, type=C3D8 \n")
    eid = 1
    for i in range(len(yc)-1):
        for j in range(len(xc)-1):
            id1 = i*len(xc) + j
            id2 = (i+1)*len(xc) + j
            id3 = (i+1)*len(xc) + (j+1)
```

```
      id4 = i*len(xc) + (j+1)
      fout.write("%d, %d, %d, %d, %d, %d, %d, %d, %d \n"
%(eid,(id1+1),(id2+1),(id3+1),(id4+1),(id1+tot_nd+1),(id2+tot_nd+1),(id3+tot_nd+1),(id4+tot_
nd+1) ))
      eid += 1
  fout.close()

  fdtp = open(sys.argv[3],'w')
  res = inp.GetRes()
  fdtp.write("NUM STEP 1 \n")
  fdtp.write("LOADSTEP 1 \n")
  fdtp.write("STRESS NODAL\n")
  for r in res:
    fdtp.write("%d, %f, %f, %f, %f, %f, %f \n"%((r+1),0.,0.,res[r],0.,0.,0.))
  for r in res:
    fdtp.write("%d, %f, %f, %f, %f, %f, %f \n"%((r+tot_nd+1),0.,0.,res[r],0.,0.,0.))
  fdtp.close()
```

The resulting *.inp* and *.dtp* file can be used in FRANC3D with the dialogs shown in Fig 12.43.



Select Crack Traction Type

Crack Face Traction/Residual Stress Type

○ Constant Crack Face Pressure
○ 1-D Radial Residual Stress Distribution
○ 2-D Radial Residual Stress Distribution
○ Surface Treatment Residual Stress Distribution
● Residual Stress Defined On a Mesh

Edit Crack Face Tractions

Analysis code:
○ ANSYS ● ABAQUS ○ NASTRAN

Mesh Based Stress Distribution

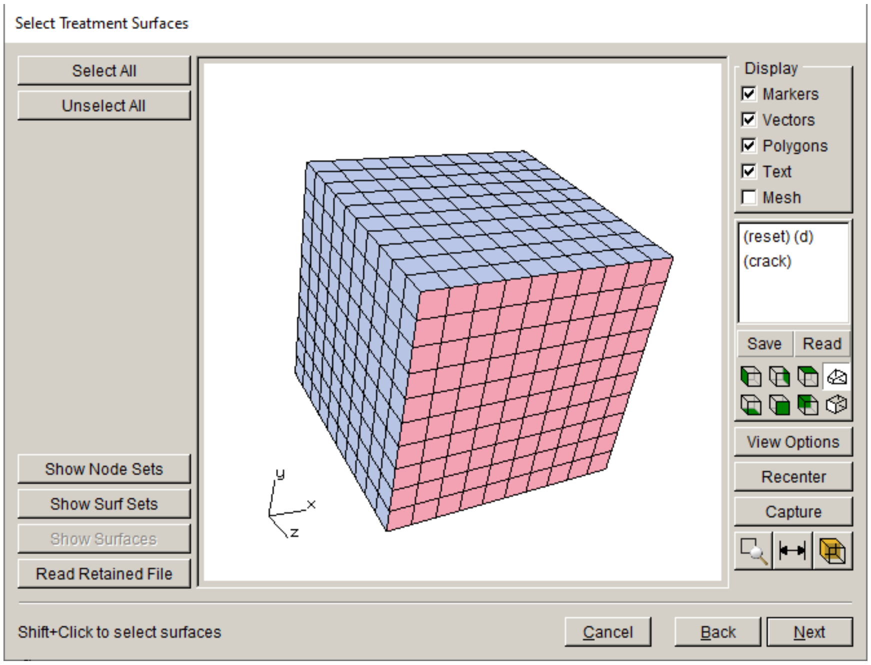
Mesh filename: resid.inp
Stress filename: resid.dtp
External load step: 1
External substep: -1
Stress scaling: 1

External substep of -1 means the final substep

Figure 12.42 FRANC3D dialogs for mesh-based residual stress CFTs.

12.6 Surface Treatment CFT

For surface treatment residual stress, the user must select the treated surface in FRANC3D. Section 7.1.5 of the Reference manual shows the entire front surface of the cube is selected, Fig 12.43.



Select Treatment Surfaces

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Shift+Click to select surfaces

Figure 12.43 Surface treatment residual stress – full surface selected.

To select only a portion of the full surface, a named surface must be defined. For example, in ABAQUS CAE, a portion of the surface is defined as shown in Fig 12.44. In FRANC3D, when the Surface Treatment dialog is displayed, the user can select the Show Surfaces button to display the named surfaces, Fig 12.45.

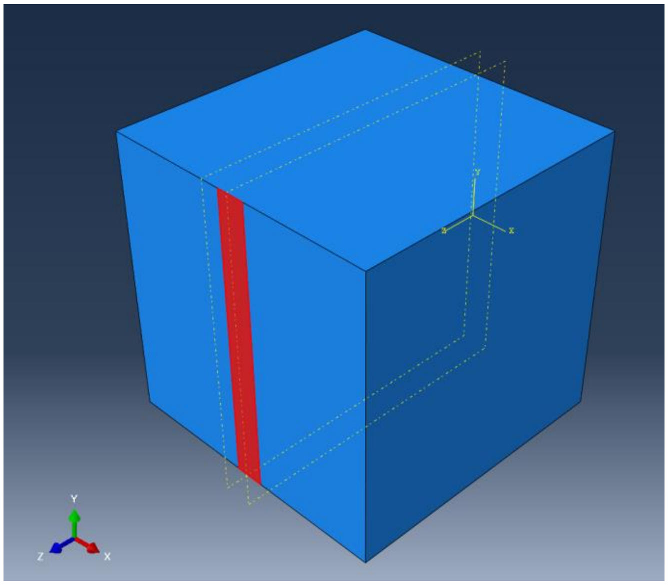


Figure 12.44 ABAQUS CAE surface partitioned.

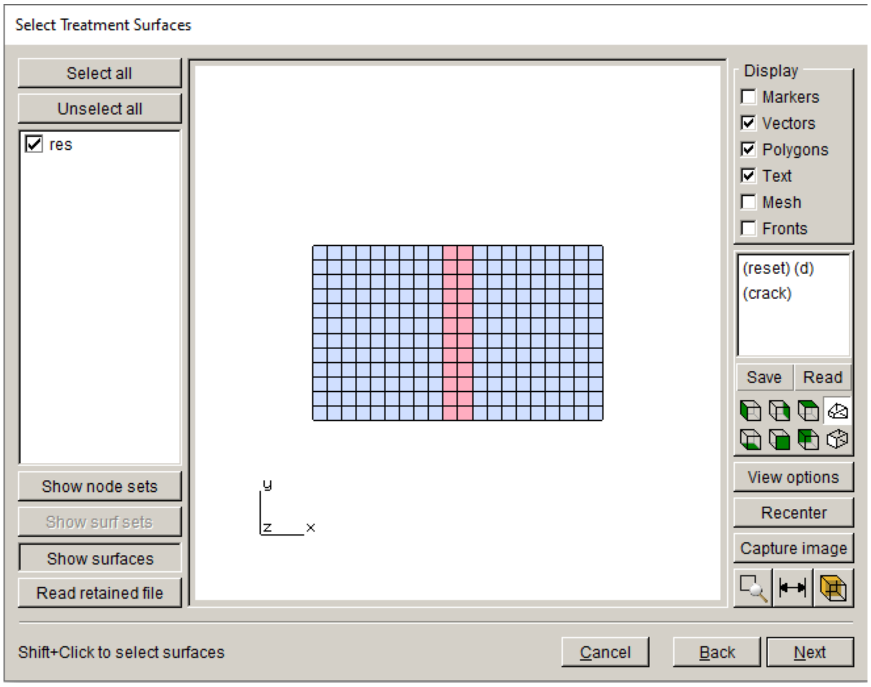


Figure 12.45 Surface treatment residual stress – named surface selected.

If the crack is larger than the treated surface, FRANC3D will define CFT (nodal forces) within a band defined by the size of the treated surface; the boundaries on the crack surface will not be exactly constrained by the boundaries of the treated surface; it depends on the crack surface mesh. For example, Fig 12.46 shows the ABAQUS nodal forces on a crack surface where the full surface is selected, and Fig 12.47 shows the nodal forces where the partial surface is selected. Fig 12.48 shows the stress distribution versus depth from the surface; the crack has a radius of 0.5 units, and the named surface is 0.2 units wide.

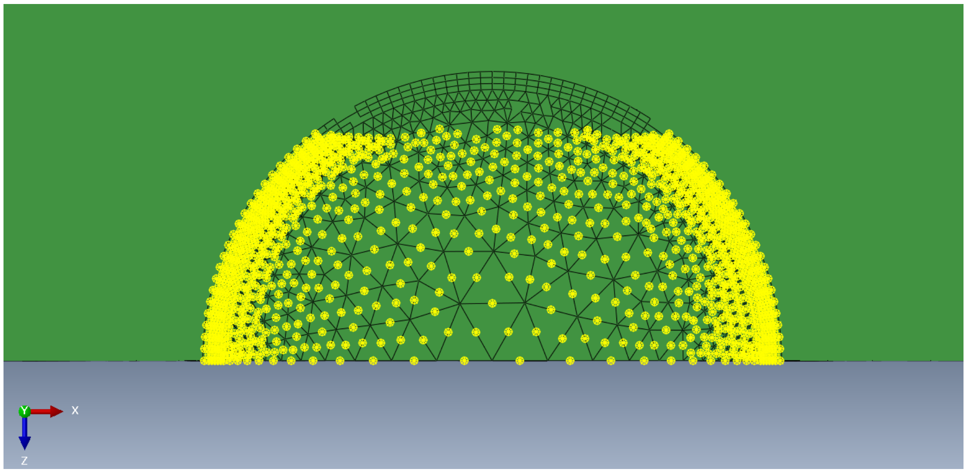


Figure 12.46 ABAQUS nodal forces for full treated surface.

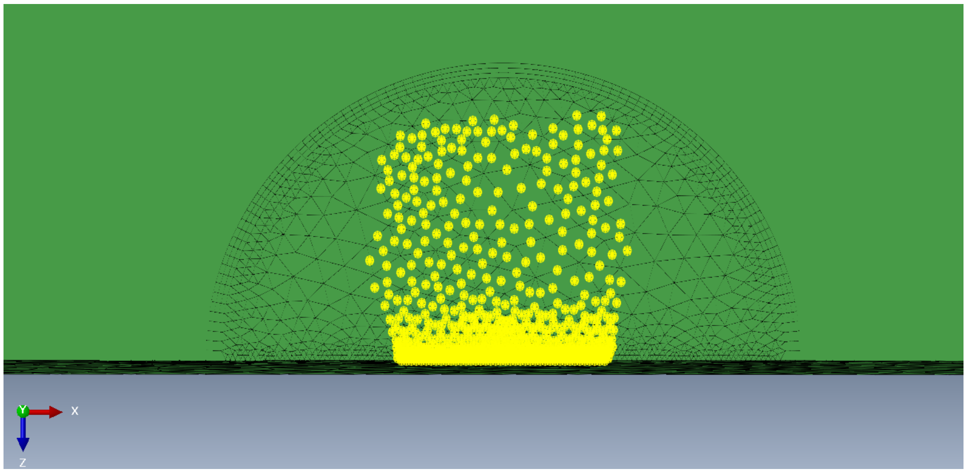
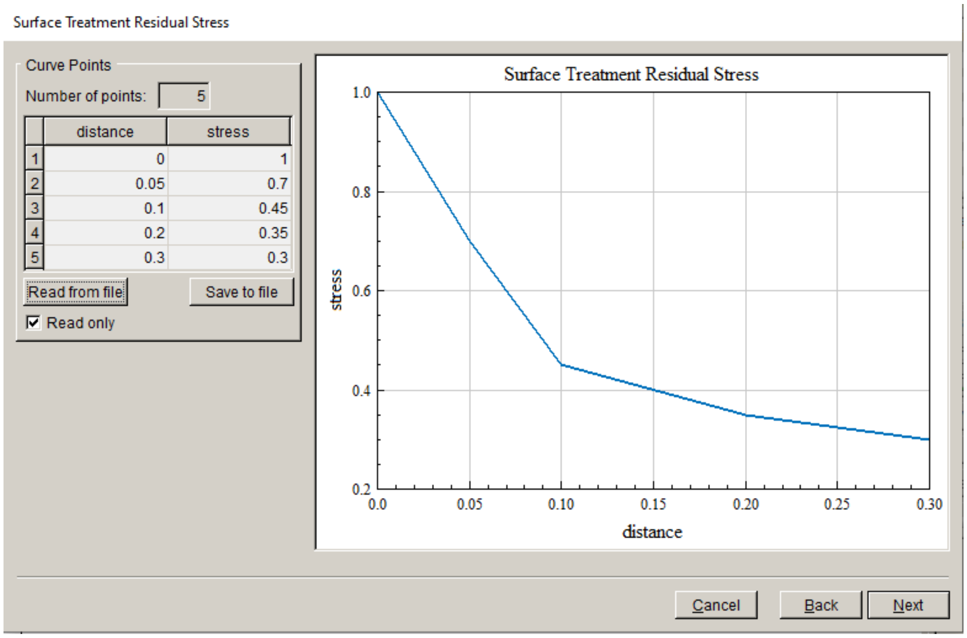


Figure 12.47 ABAQUS nodal forces for partial treated surface.



Surface Treatment Residual Stress

Curve Points

Number of points: 5

	distance	stress
1	0	1
2	0.05	0.7
3	0.1	0.45
4	0.2	0.35
5	0.3	0.3

☑ Read only

Figure 12.48 Surface treatment stress versus depth.

The user should verify the CFT (nodal forces) on the crack surface when using partial surfaces. If this approach is not satisfactory, a mesh based CFT approach can be used instead, which should produce a more accurate mapping of residual stress.

13. Crack Face Contact

This chapter describes the use of crack face contact (CFC) to prevent crack surface overlap, for a model (crack) under compression. Without CFC included, the user should expect negative Mode I SIFs; with CFC, the user should expect zero (or near zero) Mode I SIFs. The analyses below show that FRANC3D computes near-zero SIFs for all three SIF computation methods: M-integral, displacement correlation (DC), and virtual crack closure (VCCT); the only exception is for ABAQUS tied-contact.

We also note that NASTRAN contact pressure is not part of the results in the *.pch* file, so DC should be used for NASTRAN analyses.

13.1 ANSYS Thick Plate

We start with an ANSYS simply supported thick plate model; this is the same model that is used in Section 6 of the Benchmark document. That document shows the SIFs for the case of uniform applied tension. The tension is switched to compression for this study, Fig 13.1.

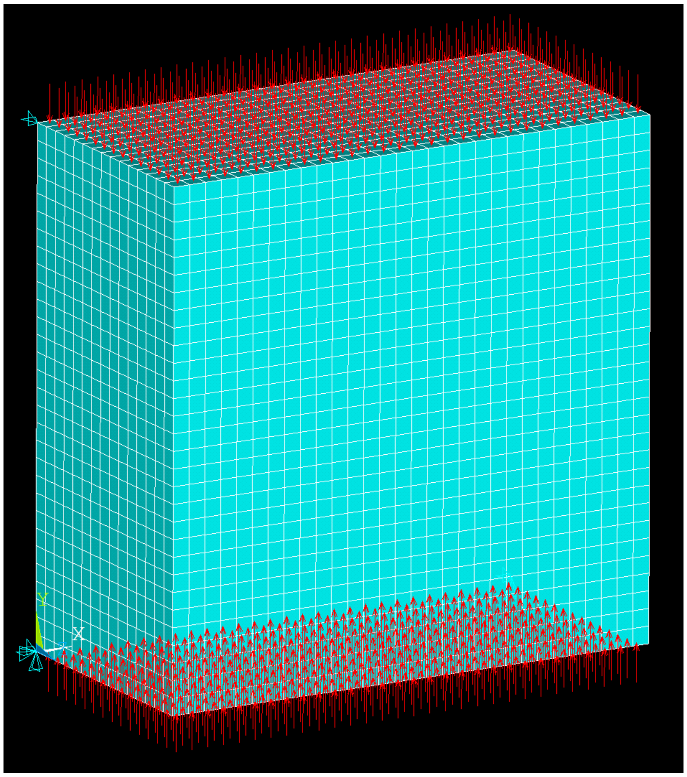


Figure 13.1 ANSYS thick plate with simple supports and uniform compression.

The model can be split into local + global portions, Fig 13.2, as in the Benchmark document. A center-through crack is inserted into the local portion, Fig 13.3. The crack is 2 units wide and extends through the plate thickness; the plate dimensions are 30x30x15.

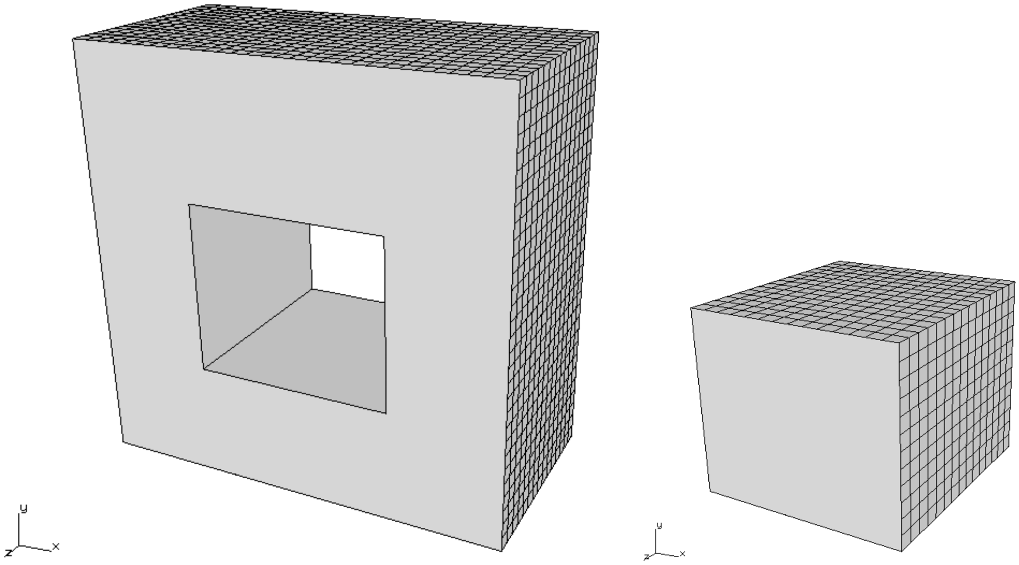


Figure 13.2 Global and local portions of the thick plate.

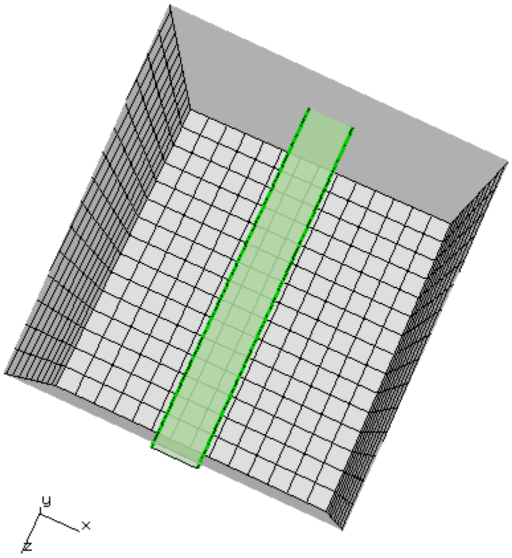
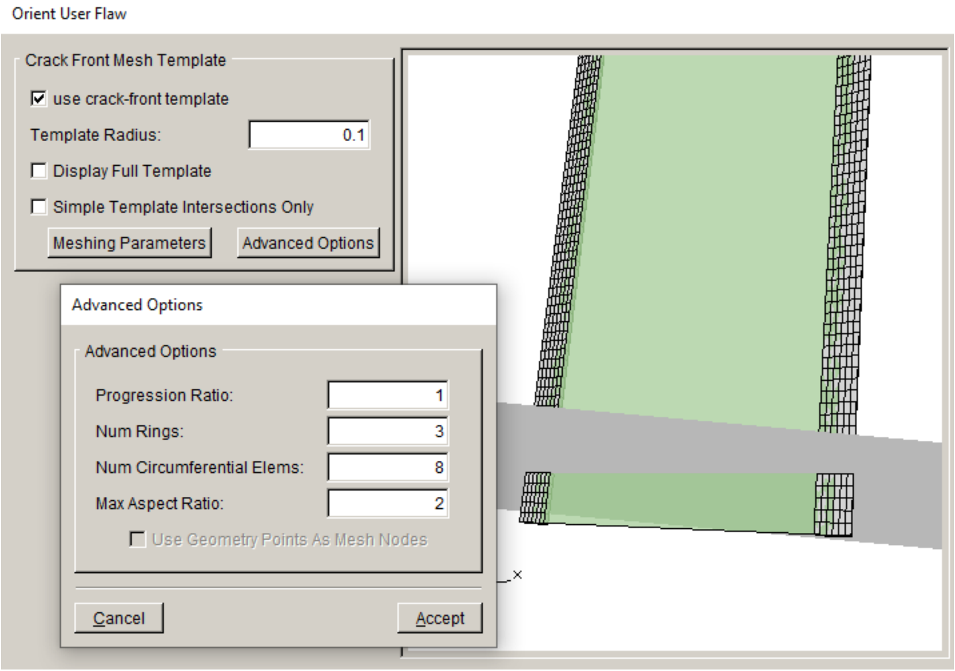


Figure 13.3 Center-through crack inserted into local portion.

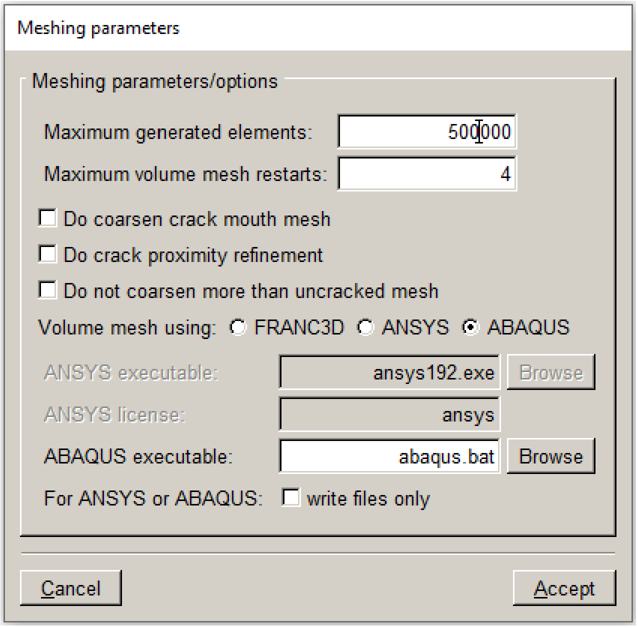
The crack is inserted and meshed using the default template settings, Fig 13.4. The meshing parameters are modified to produce a more refined mesh, Fig 13.5; <u>Do coarsen crack</u>… is turned off. The resulting surface mesh around the crack on the plate surface is shown in Fig 13.6.



☑ use crack-front template
Template Radius: 0.1
☐ Display Full Template
☐ Simple Template Intersections Only

Progression Ratio: 1
Num Rings: 3
Num Circumferential Elems: 8
Max Aspect Ratio: 2
☐ Use Geometry Points As Mesh Nodes

Figure 13.4 Center-through crack with default template mesh parameters.



Maximum generated elements: 500000
Maximum volume mesh restarts: 4
☐ Do coarsen crack mouth mesh
☐ Do crack proximity refinement
☐ Do not coarsen more than uncracked mesh
Volume mesh using: ○ FRANC3D ○ ANSYS ● ABAQUS
ANSYS executable: ansys192.exe
ANSYS license: ansys
ABAQUS executable: abaqus.bat
For ANSYS or ABAQUS: ☐ write files only

Figure 13.5 Meshing parameters dialog with <u>Do coarsen crack</u>… turned off.

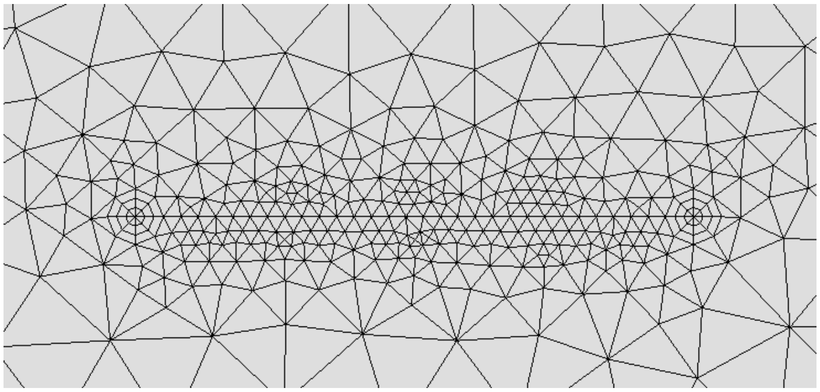
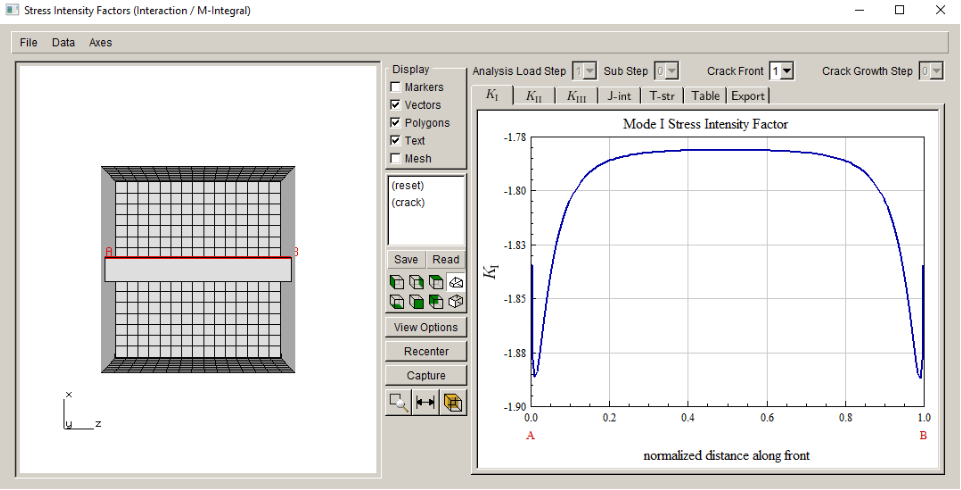


Figure 13.6 Surface mesh for default template parameters.

13.1.1 No Crack Face Contact (CFC)

We first run an analysis without turning on CFC. Negative Mode I SIFs are computed as shown in Fig 13.7a. The two crack fronts produce the same SIFs, so we only show the first front. The SIFs are equal, but with opposite sign, to those for uniform tension (see Section 6.1 of the Benchmark document). The VCCT SIFs differ by less than 1% and the DC SIFs differ by about 1.1%, Fig 13.7b.

The negative sign indicates that the crack surfaces are overlapping; as this is not physically realistic, CFC can be turned on to prevent it.



Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.7a Mode I SIFs for uniform compression without CFC using M-integral.

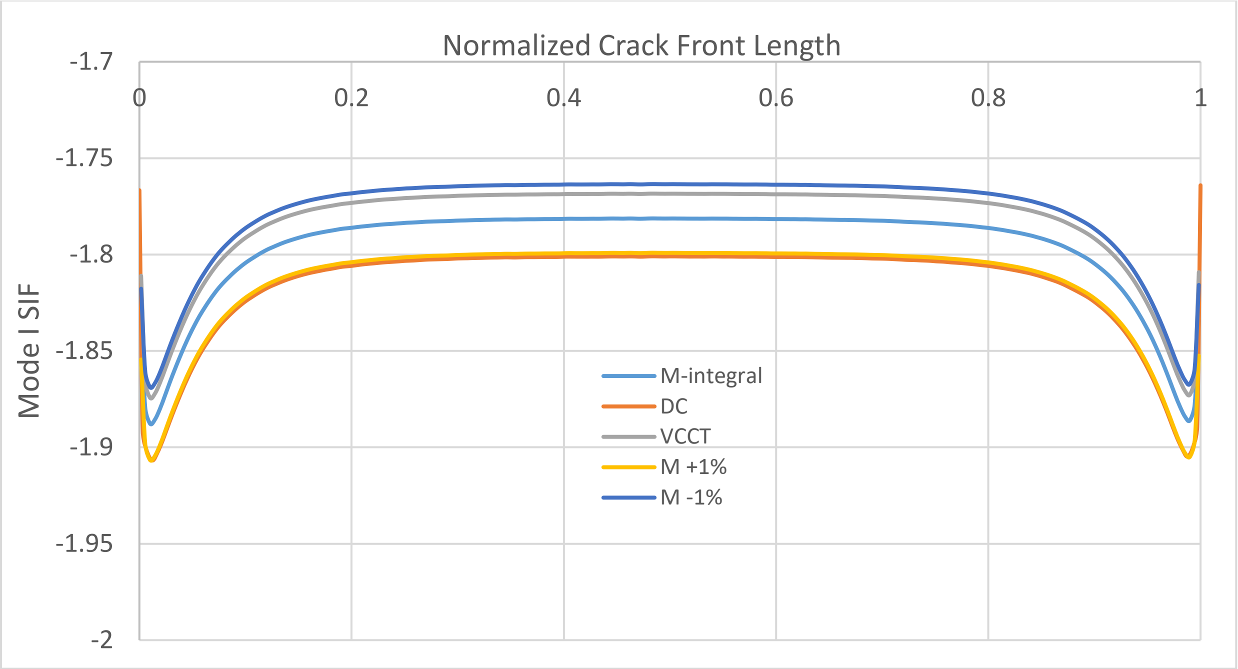
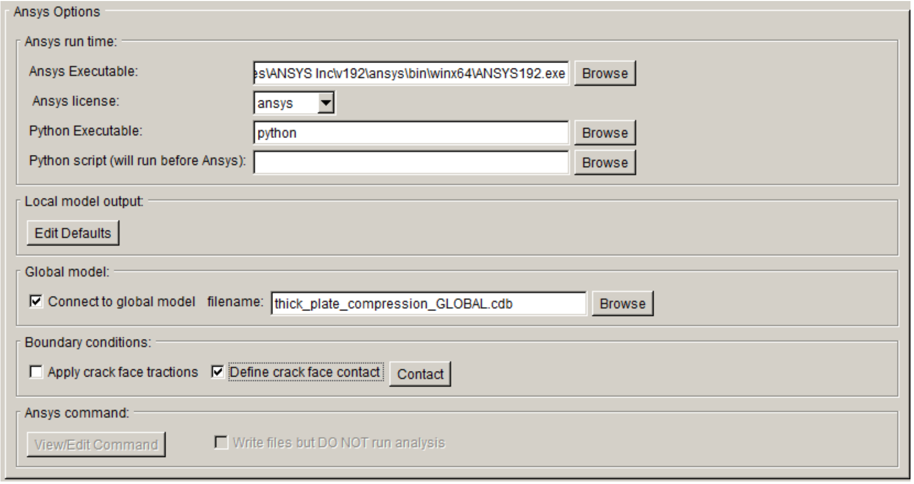


Figure 13.7b Mode I SIFs for uniform compression without CFC.

13.1.2 Default Crack Face Contact (CFC)

FRANC3D has a checkbox in the analysis dialog that allows one to define CFC, Fig 13.8. The **Contact** button invokes the dialog shown in Fig 13.9. The dialog provides many of the ANSYS contact settings; the default values should correspond to ANSYS defaults. Note that the material ID (811) should be changed if it is already used in the uncracked model.



Ansys Executable: s\ANSYS Inc\v192\ansys\bin\winx64\ANSYS192.exe

Ansys license: ansys

Python Executable: python

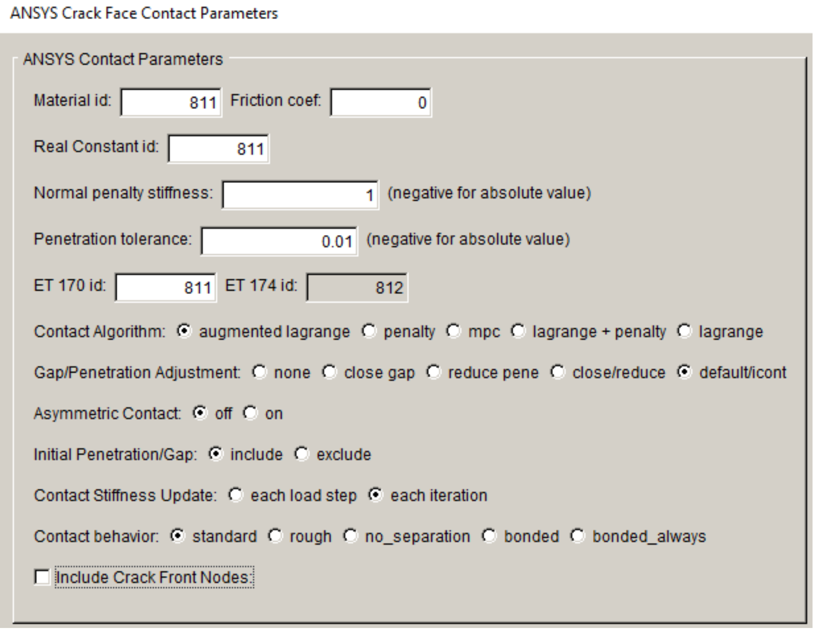
Python script (will run before Ansys):

☑ Connect to global model filename: thick_plate_compression_GLOBAL.cdb

□ Apply crack face tractions ☑ Define crack face contact

□ Write files but DO NOT run analysis

Figure 13.8. CFC check box and button.



ANSYS Crack Face Contact Parameters

ANSYS Contact Parameters

Material id: 811 Friction coef: 0

Real Constant id: 811

Normal penalty stiffness: 1 (negative for absolute value)

Penetration tolerance: 0.01 (negative for absolute value)

ET 170 id: 811 ET 174 id: 812

Contact Algorithm: ● augmented lagrange ○ penalty ○ mpc ○ lagrange + penalty ○ lagrange

Gap/Penetration Adjustment: ○ none ○ close gap ○ reduce pene ○ close/reduce ● default/icont

Asymmetric Contact: ● off ○ on

Initial Penetration/Gap: ● include ○ exclude

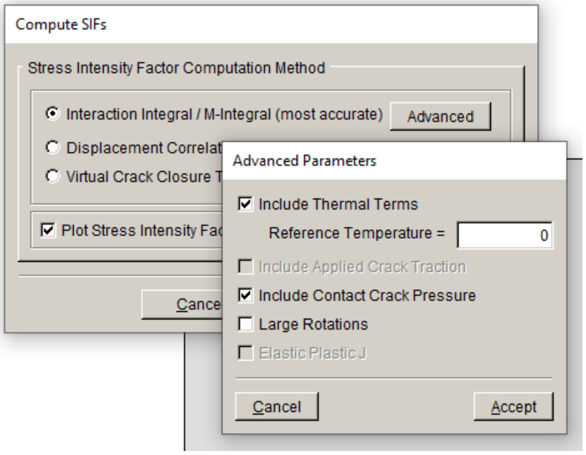
Contact Stiffness Update: ○ each load step ● each iteration

Contact behavior: ● standard ○ rough ○ no_separation ○ bonded ○ bonded_always

□ Include Crack Front Nodes

Figure 13.9 ANSYS CFC parameters dialog.

Once the ANSYS analysis is finished, SIFs can be computed; note that the analysis time increases significantly compared to the non-CFC analysis. The M-integral includes a term for crack face traction (contact pressure in this case). ANSYS exports the contact pressure, along with displacements and temperatures, to the *.dtp* file (which FRANC3D reads). Fig 13.10 shows the FRANC3D Compute SIFs dialog with the Include Contact Crack Pressure option checked – this should be checked automatically if there are contact pressures in the *.dtp* file. Fig 13.11a shows the resulting Mode I SIFs. Note that the values are not exactly 0.0.



Compute SIFs

Stress Intensity Factor Computation Method

● Interaction Integral / M-Integral (most accurate)

○ Displacement Correlat

○ Virtual Crack Closure T

☑ Plot Stress Intensity Fac

Advanced Parameters

☑ Include Thermal Terms

Reference Temperature = 0

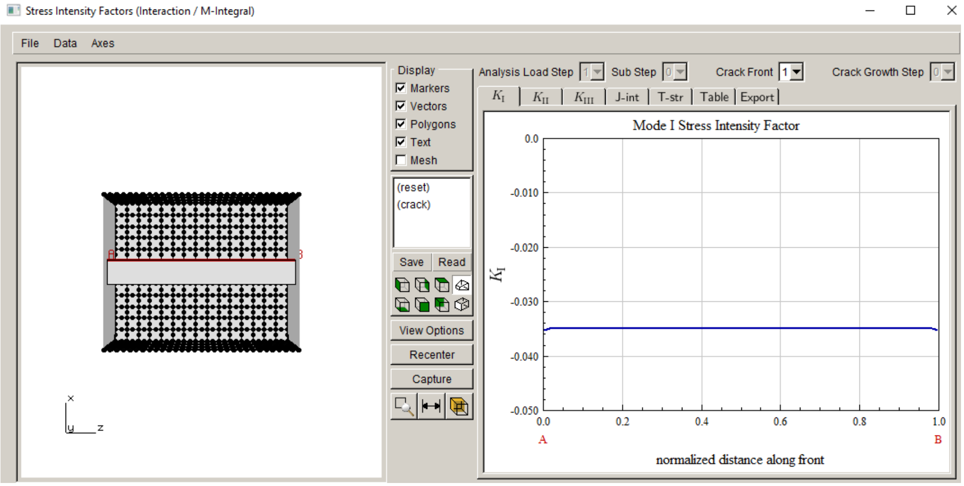
□ Include Applied Crack Traction

☑ Include Contact Crack Pressure

□ Large Rotations

□ Elastic Plastic J

Figure 13.10. Compute SIFs dialog using M-Integral with contact pressure included.



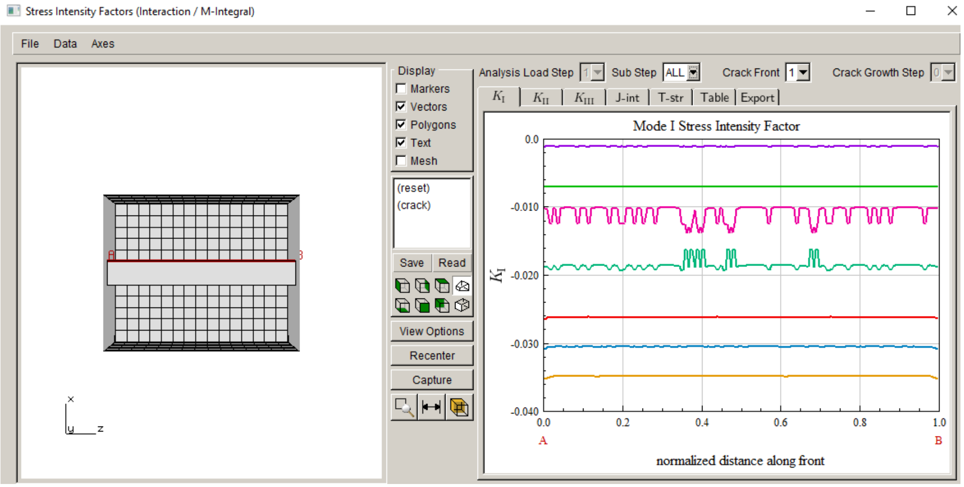
Stress Intensity Factors (Interaction / M-Integral)

Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 13.11a Mode I SIFs for uniform compression with default CFC settings using M-integral.

The ANSYS solution is completed using seven substeps (the number of substeps can be modified by the user). The SIFs for all substeps are shown in Fig 13.11b (there is a FRANC3D setting that requests output for all substeps). It is possible that forcing additional ANSYS substeps could improve the SIFs – at the cost of additional computation time.



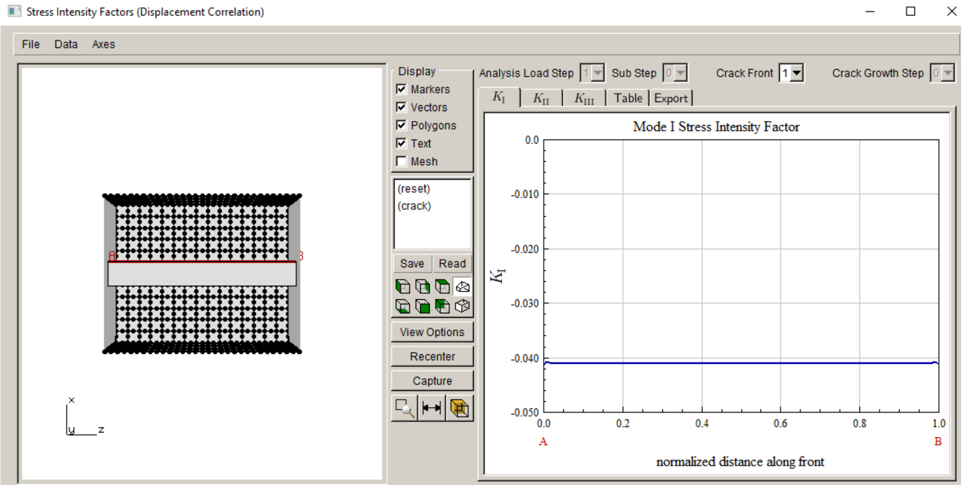
Stress Intensity Factors (Interaction / M-Integral)

Display: □ Markers ☑ Vectors ☑ Polygons ☑ Text □ Mesh

Analysis Load Step 1 | Sub Step ALL | Crack Front 1 | Crack Growth Step 0

Figure 13.11b Mode I SIFs with default CFC settings using M-integral for all solution substeps.

FRANC3D has three options for computing SIFs, and in general the M-integral is the most accurate, while displacement correlation (DC) is the least accurate. Fig 13.12 shows the SIFs computed using the DC method; these SIFs are close to the M-integral SIFs (for the last substep).



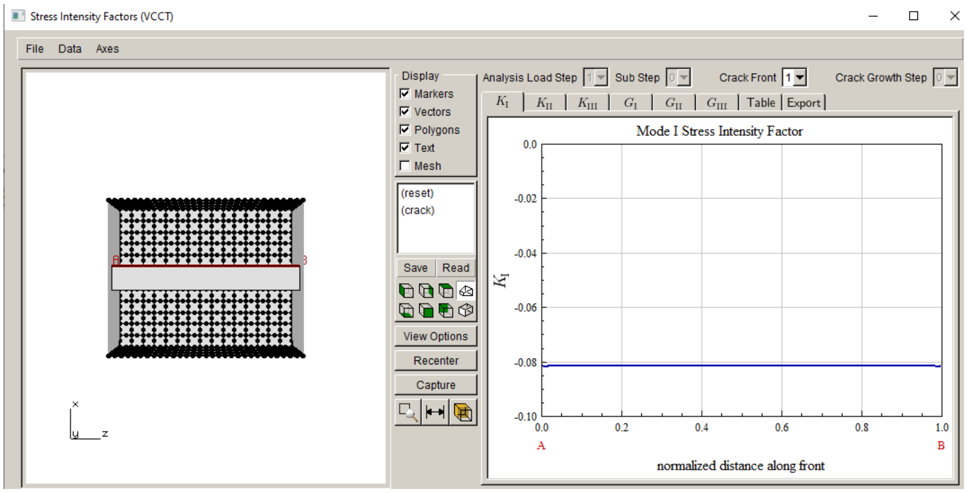
Stress Intensity Factors (Displacement Correlation)

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.12 Mode I SIFs for uniform compression with default CFC settings using DC.

The virtual crack closure (VCCT) Mode I SIFs are shown in Fig 13.13. As with the M-integral, the VCCT method includes a term for crack face traction. The VCCT SIFs are close to M-integral and DC SIFs, although they are a little further from zero.



Stress Intensity Factors (VCCT)

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.13. Mode I SIFs for uniform compression with default CFC settings using VCCT.

Fig 13.14 shows the ANSYS contact pressure contours on the crack surface. Most nodes on the crack surface have a contact pressure near 1.0, which is equal to the applied compressive load. Along the contact element boundary, adjacent to the crack front, the pressure reaches a peak near 2.0. The listing shown in Fig 13.15 gives typical values for the whole crack surface, and the peak values are highlighted.

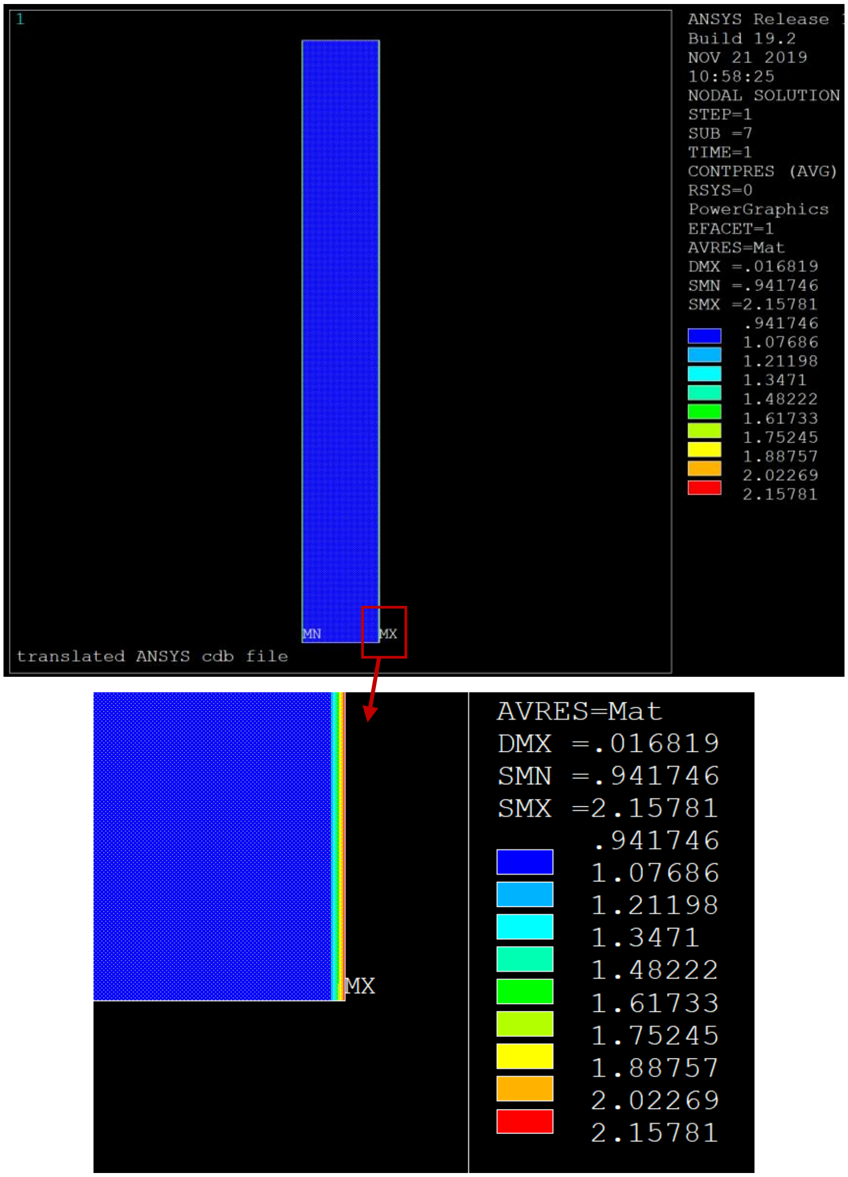
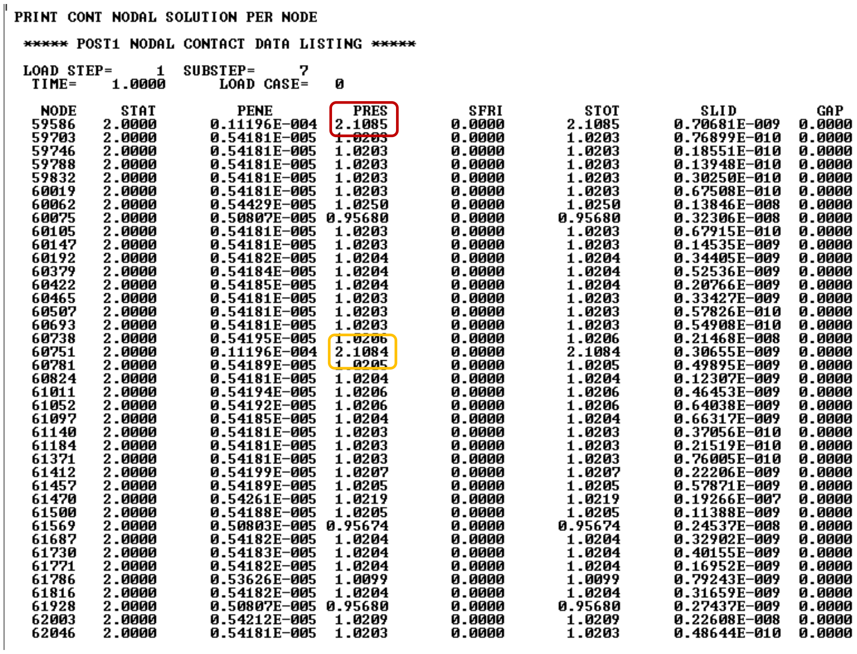


Figure 13.14. ANSYS contact pressure contours.

The contact surface does not include the crack front nodes; therefore, the contact elements do not extend to the front. Defining mating contact elements with shared nodes (at crack front) and with quarter-point rather than mid-side nodes can cause errors when solving so we do not include them by default.



```
PRINT CONT NODAL SOLUTION PER NODE

 ***** POST1 NODAL CONTACT DATA LISTING *****

 LOAD STEP=     1  SUBSTEP=     7
  TIME=    1.0000      LOAD CASE=   0

   NODE     STAT          PENE         PRES         SFRI          STOT          SLID        GAP
  59586   2.0000      0.11196E-004  2.1085       0.0000        2.1085       0.70681E-009  0.0000
  59703   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.76899E-010  0.0000
  59746   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.18551E-010  0.0000
  59788   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.13948E-010  0.0000
  59832   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.30250E-010  0.0000
  60019   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.67508E-010  0.0000
  60062   2.0000      0.54429E-005  1.0250       0.0000        1.0250       0.13846E-008  0.0000
  60075   2.0000      0.50807E-005 0.95680       0.0000       0.95680       0.32306E-008  0.0000
  60105   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.67915E-010  0.0000
  60147   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.14535E-009  0.0000
  60192   2.0000      0.54182E-005  1.0204       0.0000        1.0204       0.34405E-009  0.0000
  60379   2.0000      0.54184E-005  1.0204       0.0000        1.0204       0.52536E-009  0.0000
  60422   2.0000      0.54185E-005  1.0204       0.0000        1.0204       0.20766E-009  0.0000
  60465   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.33427E-009  0.0000
  60507   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.57826E-010  0.0000
  60693   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.54908E-010  0.0000
  60738   2.0000      0.54195E-005  1.0206       0.0000        1.0206       0.21468E-008  0.0000
  60751   2.0000      0.11196E-004  2.1084       0.0000        2.1084       0.30655E-009  0.0000
  60781   2.0000      0.54189E-005  1.0205       0.0000        1.0205       0.49895E-009  0.0000
  60824   2.0000      0.54181E-005  1.0204       0.0000        1.0204       0.12307E-009  0.0000
  61011   2.0000      0.54194E-005  1.0206       0.0000        1.0206       0.46453E-009  0.0000
  61052   2.0000      0.54192E-005  1.0206       0.0000        1.0206       0.64038E-009  0.0000
  61097   2.0000      0.54185E-005  1.0204       0.0000        1.0204       0.66317E-009  0.0000
  61140   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.37056E-010  0.0000
  61184   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.21519E-010  0.0000
  61371   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.76005E-010  0.0000
  61412   2.0000      0.54199E-005  1.0207       0.0000        1.0207       0.22206E-009  0.0000
  61457   2.0000      0.54189E-005  1.0205       0.0000        1.0205       0.57871E-009  0.0000
  61470   2.0000      0.54261E-005  1.0219       0.0000        1.0219       0.19266E-007  0.0000
  61500   2.0000      0.54188E-005  1.0205       0.0000        1.0205       0.11388E-009  0.0000
  61569   2.0000      0.50803E-005 0.95674       0.0000       0.95674       0.24537E-008  0.0000
  61687   2.0000      0.54182E-005  1.0204       0.0000        1.0204       0.32902E-009  0.0000
  61730   2.0000      0.54183E-005  1.0204       0.0000        1.0204       0.40155E-009  0.0000
  61771   2.0000      0.54182E-005  1.0204       0.0000        1.0204       0.16952E-009  0.0000
  61786   2.0000      0.53626E-005  1.0099       0.0000        1.0099       0.79243E-009  0.0000
  61816   2.0000      0.54182E-005  1.0204       0.0000        1.0204       0.31659E-009  0.0000
  61928   2.0000      0.50807E-005 0.95680       0.0000       0.95680       0.27437E-009  0.0000
  62003   2.0000      0.54212E-005  1.0209       0.0000        1.0209       0.22608E-008  0.0000
  62046   2.0000      0.54181E-005  1.0203       0.0000        1.0203       0.48644E-010  0.0000
```

Figure 13.15 ANSYS contact data listing.

Fig 13.16 shows a small set of crack front and near-front nodes and a couple of the contact elements to highlight the location of the peak pressures. Nodes 75140 and 437860 both have pressure values of 2.2084. Midside nodes, such as 362709 and 282555 do not have results as ANSYS only provides contact results for corner nodes. Nodes such as 470545, 107280 and 439441 have pressures of about 0.96. The crack front nodes (such as 287836) are not included in the contact and thus they do not have contact data listed.

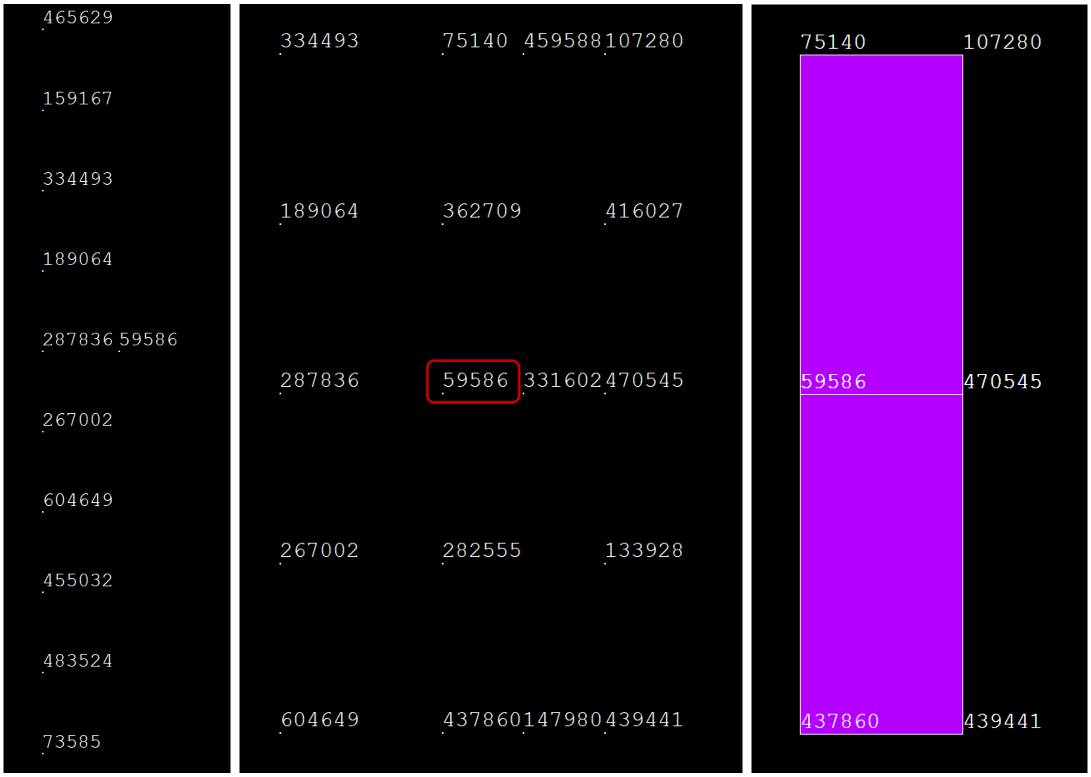


Figure 13.16 ANSYS crack front and adjacent nodes and contact elements.

13.1.3 Modified Crack Face Contact (CFC) Settings

FRANC3D provides some options/parameters for the CFC. In this section, we change these settings to see how the SIFs are affected. First, we change the settings to include the crack front nodes in the contact elements (last option in the dialog in Fig 13.9). The resulting SIFs are shown in Figs 13.17-19 and are comparable to SIFs in Figs 13.11-13.

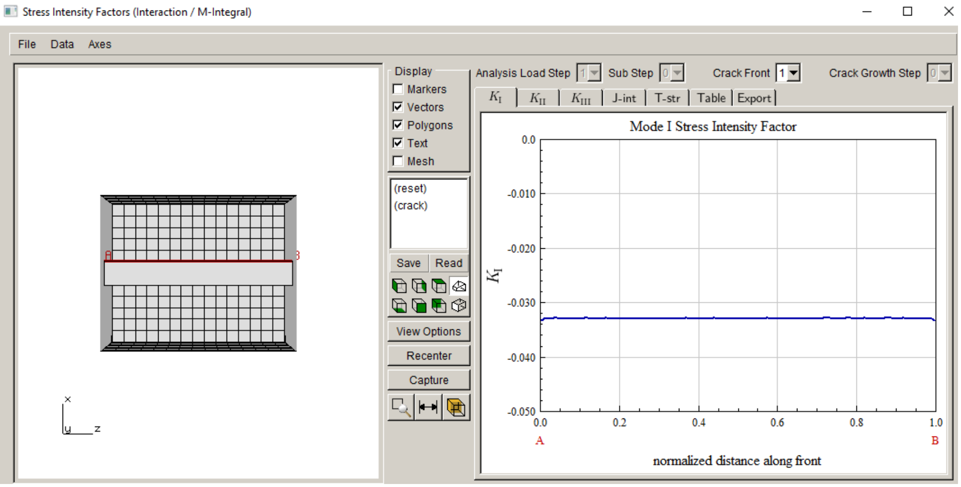
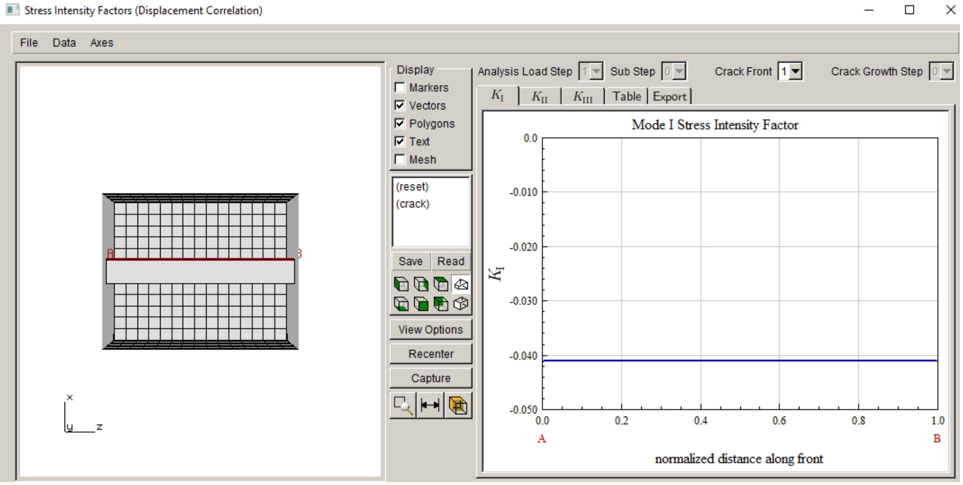


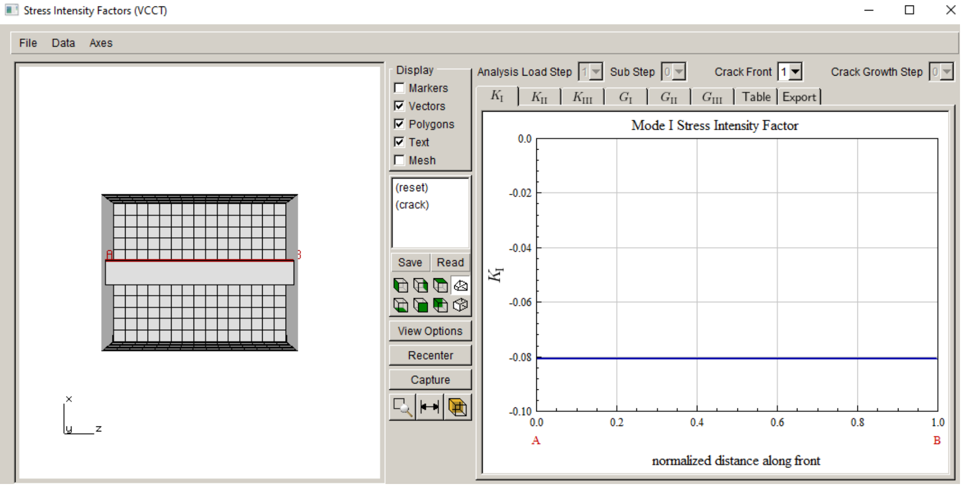
Figure 13.17 Mode I SIFs with crack front nodes included in CFC using M-integral.



Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 13.18 Mode I SIFs with crack front nodes included in CFC using DC.

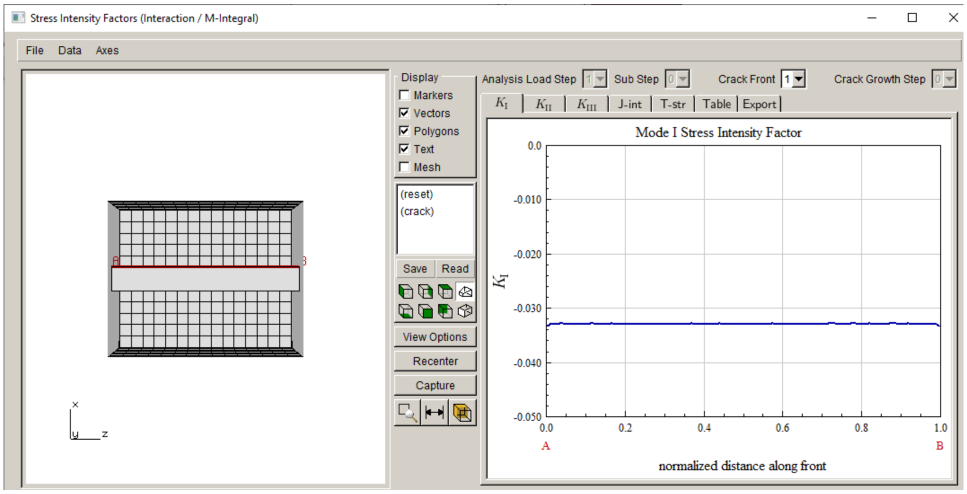


Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

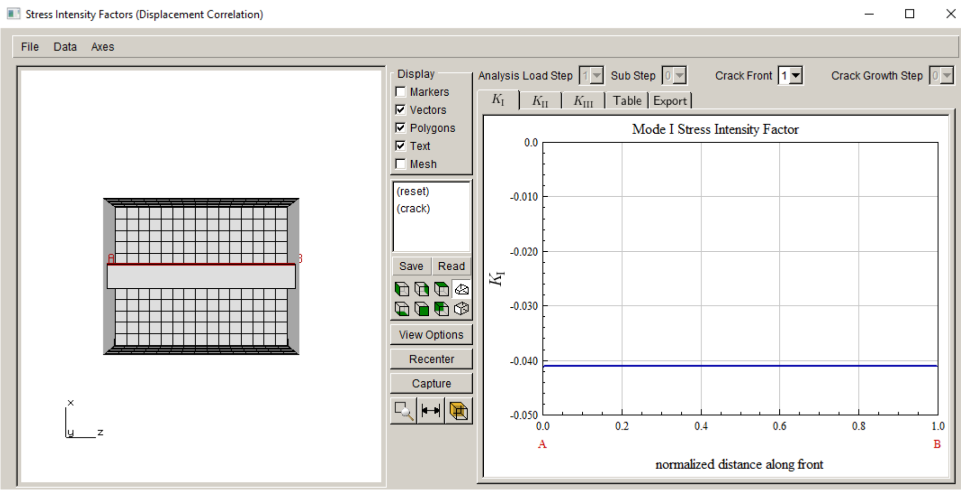
Figure 13.19 Mode I SIFs with crack front nodes included in CFC using VCCT.

The next change is to switch the contact behavior to “bonded” and the Mode I SIFs are shown in Figs 13.20-22. ANSYS provides options that we could try, but for this model it is unlikely that we will improve the results.



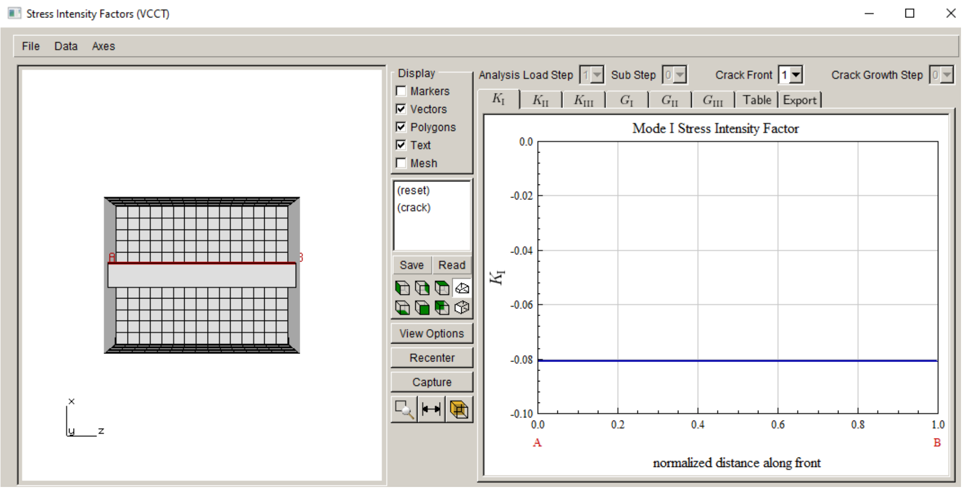
Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.20 Mode I SIFs with crack front nodes included in bonded-CFC using M-integral.



Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.21 Mode I SIFs with crack front nodes included in bonded-CFC using DC.

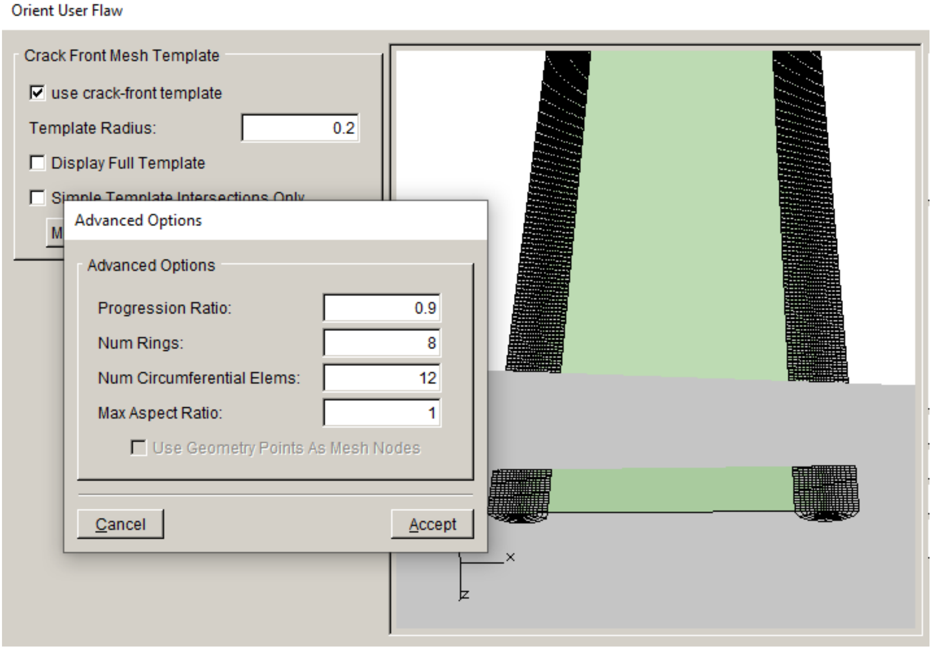


Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.22 Mode I SIFs with crack front nodes included in bonded-CFC using VCCT.

13.1.4 Modified Crack Front Template

In this section, we show how one can modify the default crack front template mesh to achieve slightly better SIF results at the expense of significantly more computation time. The revised template parameters are shown in Fig 13.23, while the meshing parameters are the same as above (see Fig 13.5). The default ANSYS CFC settings are used. The resulting SIFs are shown in Figs 13.24-26. These values can be compared to those in Figs 13.11-13. The difference in SIFs is relatively small, while the analysis time increases significantly.



Orient User Flaw

Crack Front Mesh Template

☑ use crack-front template

Template Radius: 0.2

☐ Display Full Template

☐ Simple Template Intersections Only

Advanced Options

Advanced Options

Progression Ratio: 0.9

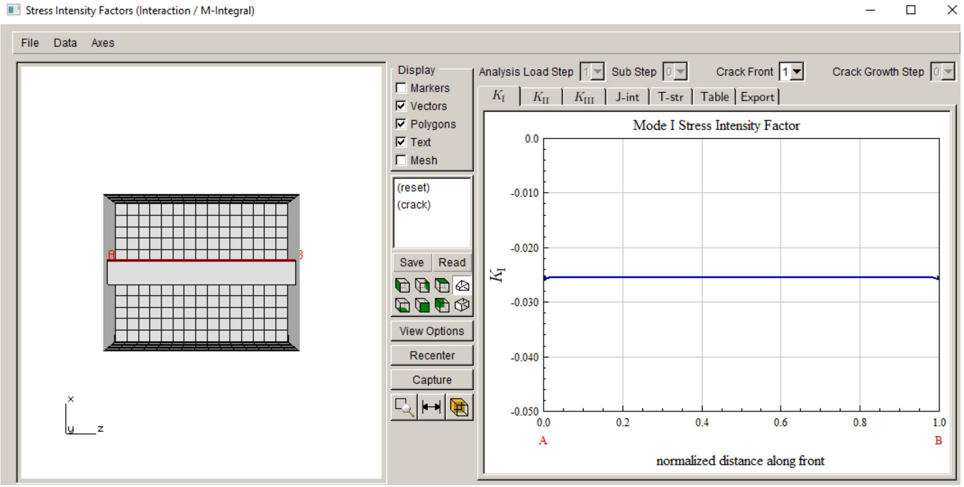
Num Rings: 8

Num Circumferential Elems: 12

Max Aspect Ratio: 1

☐ Use Geometry Points As Mesh Nodes

Figure 13.23 Center-through crack with edited template mesh parameters.



Stress Intensity Factors (Interaction / M-Integral)

Display

☐ Markers

☑ Vectors

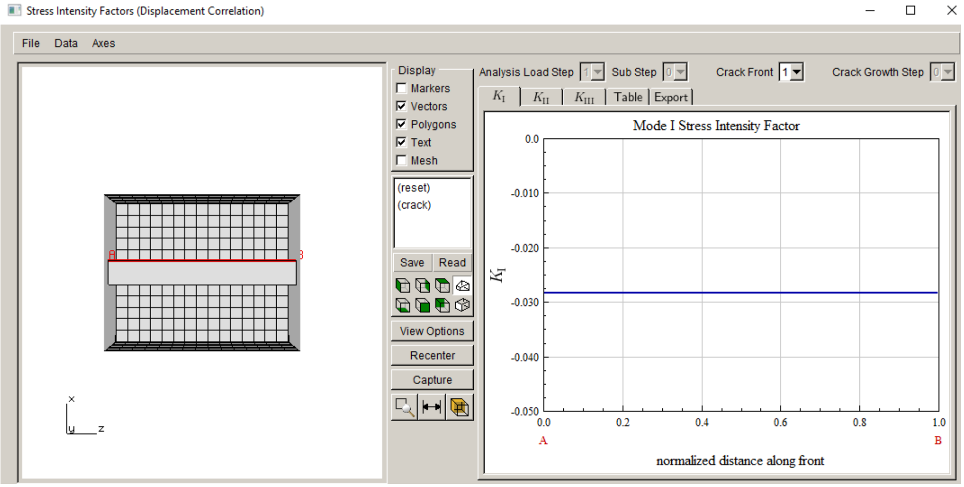
☑ Polygons

☑ Text

☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

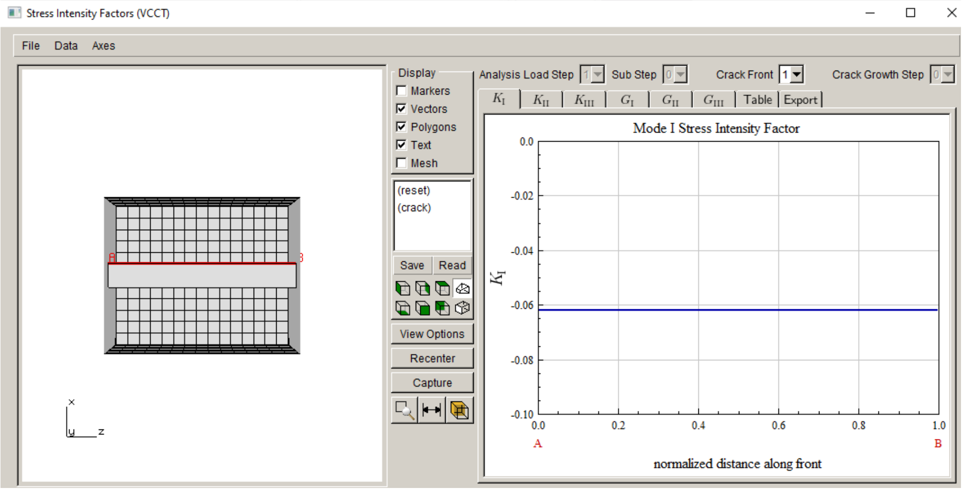
Figure 13.24 Mode I SIFs with modified template and default CFC using M-integral.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.25 Mode I SIFs with modified template and default CFC using DC.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

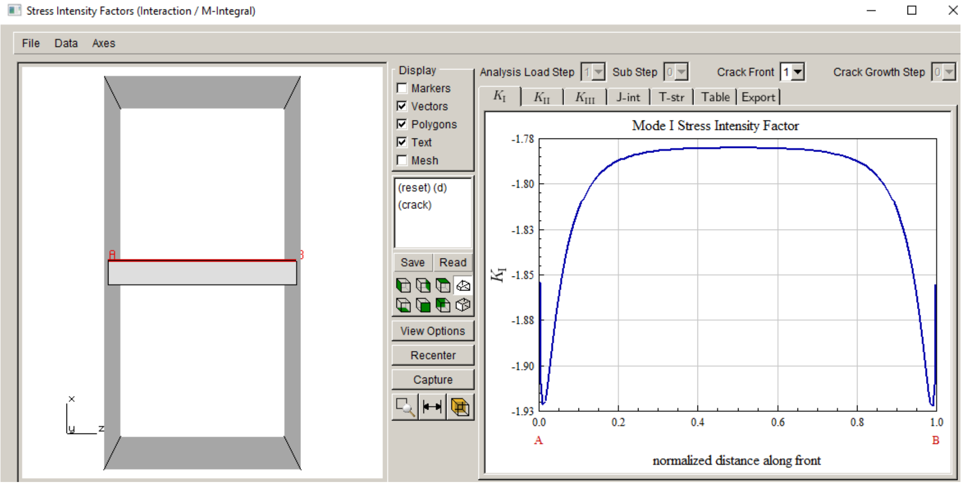
Figure 13.26 Mode I SIFs with modified template and default CFC using VCCT.

13.2 ABAQUS Thick Plate

The ABAQUS thick plate model is basically the same as the ANSYS model, the uncracked mesh is slightly different, but that has negligible effect on the SIFs.

13.2.1 No Crack Face Contact (CFC)

The first analysis is completed without turning on CFC, and the Mode I SIFs are computed as shown in Figs 13.27-29. The M-integral SIFs can be compared with the ANSYS SIFs in Fig 13.7; the difference is mostly less than 1%, Fig 13.30.



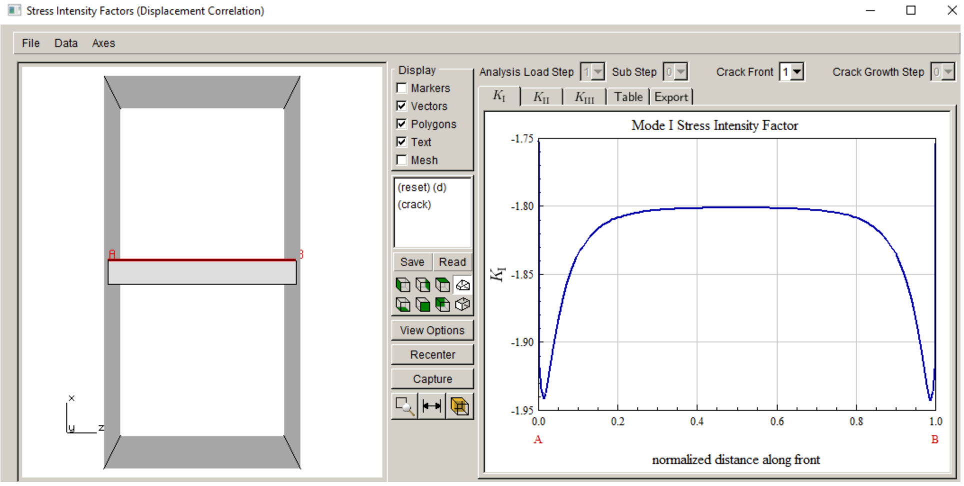
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

K_I | K_{II} | K_{III} | J-int | T-str | Table | Export

Figure 13.27 ABAQUS Mode I SIFs for uniform compression without CFC using M-integral.


Stress Intensity Factors (Displacement Correlation)
File
Data
Axes
Display
View Options
Recenter
Capture
Save
Read
Mode I Stress Intensity Factor
-1.75
-1.80
-1.85
-1.90
-1.95
K_I
0.0
0.2
0.4
0.6
0.8
1.0
A
B
normalized distance along front
x
y
z

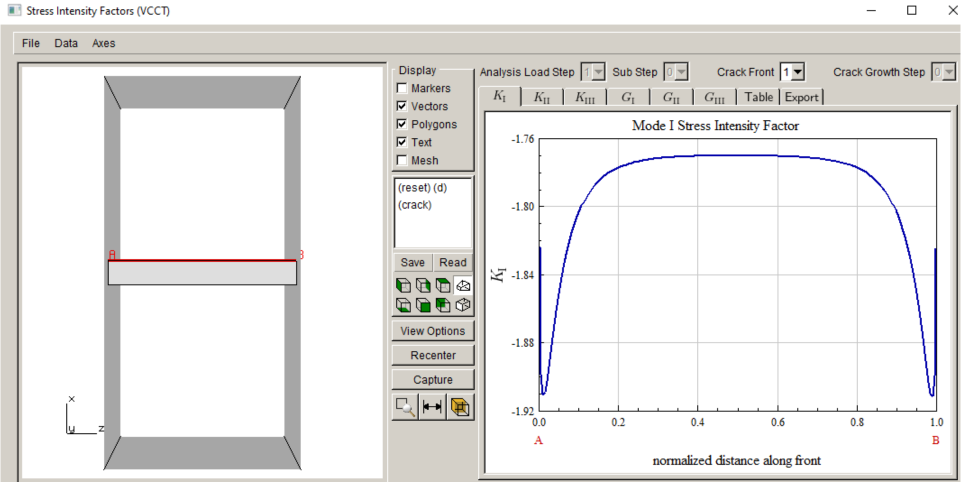
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

K_I | K_{II} | K_{III} | Table | Export

Figure 13.28 ABAQUS Mode I SIFs for uniform compression without CFC using DC.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

K_I | K_{II} | K_{III} | G_I | G_{II} | G_{III} | Table | Export

Figure 13.29 ABAQUS Mode I SIFs for uniform compression without CFC using VCCT.

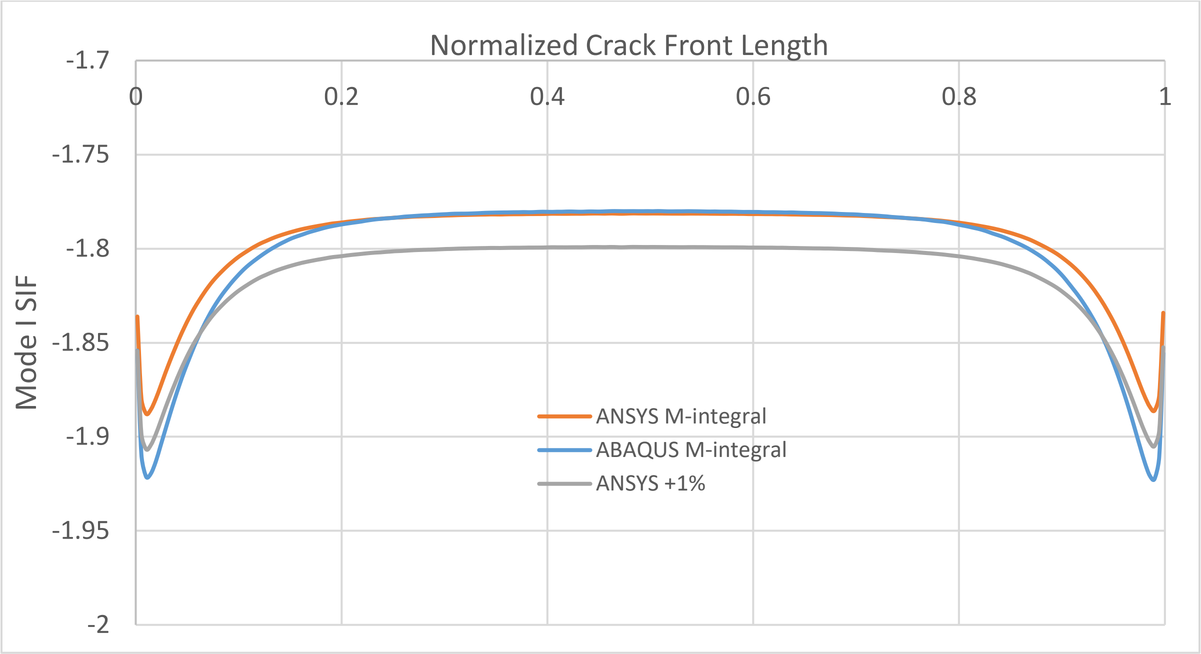
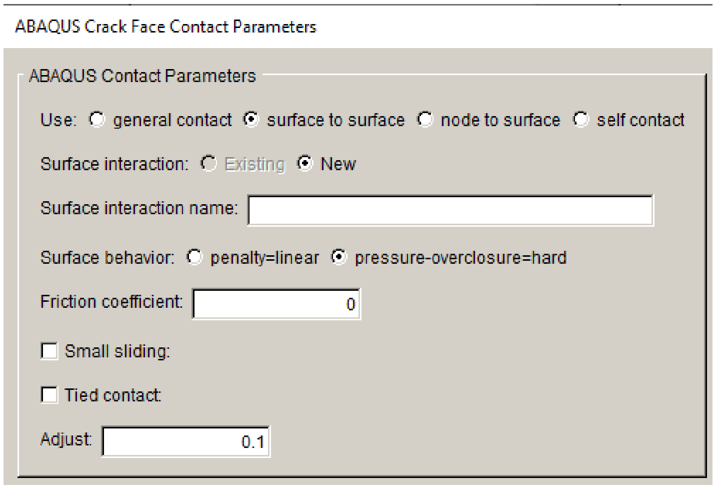


Figure 13.30 ABAQUS and ANSYS Mode I SIFs for uniform compression without CFC.

13.2.2 Default Crack Face Contact (CFC)

The ABAQUS contact dialog is shown in Fig 13.31. The default settings are used here – the user must supply a (unique) Surface interaction name. The resulting Mode I SIFs are shown in Figs 13.32-34; these results can be compared with the ANSYS results in Figs 13.11-13. The DC SIFs are similar to those for ANSYS. The M-integral SIFs are quite different – positive rather than negative; the contact pressure adjacent to the crack front is about 2.9 compared to 2.1 for ANSYS.



ABAQUS Crack Face Contact Parameters

ABAQUS Contact Parameters

Use: ○ general contact ● surface to surface ○ node to surface ○ self contact

Surface interaction: ○ Existing ● New

Surface interaction name:

Surface behavior: ○ penalty=linear ● pressure-overclosure=hard

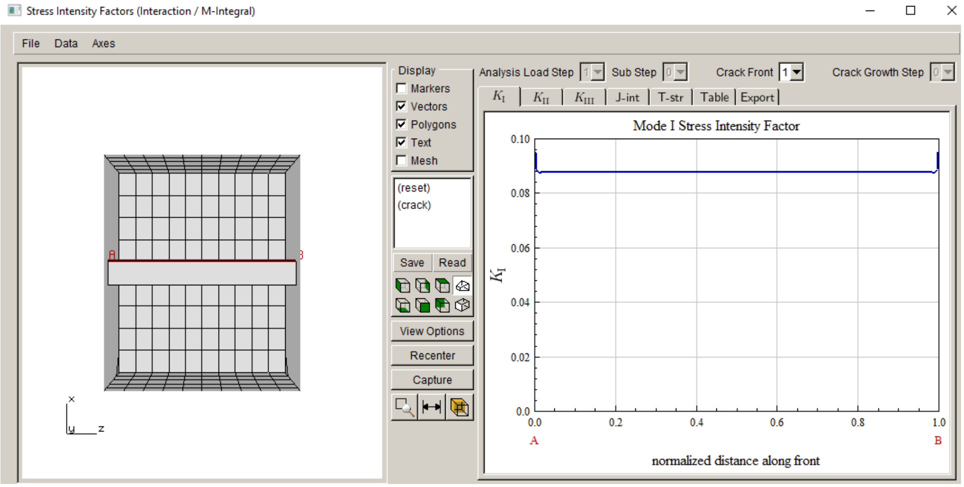
Friction coefficient: 0

□ Small sliding:

□ Tied contact:

Adjust: 0.1

Figure 13.31 ABAQUS CFC parameters dialog.

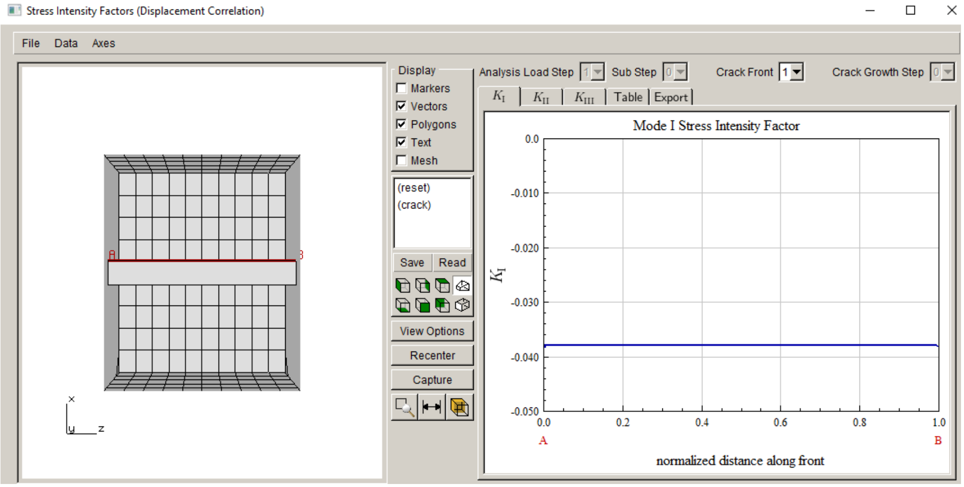


Stress Intensity Factors (Interaction / M-Integral)

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.32 ABAQUS Mode I SIFs for uniform compression with default CFC settings using M-integral.

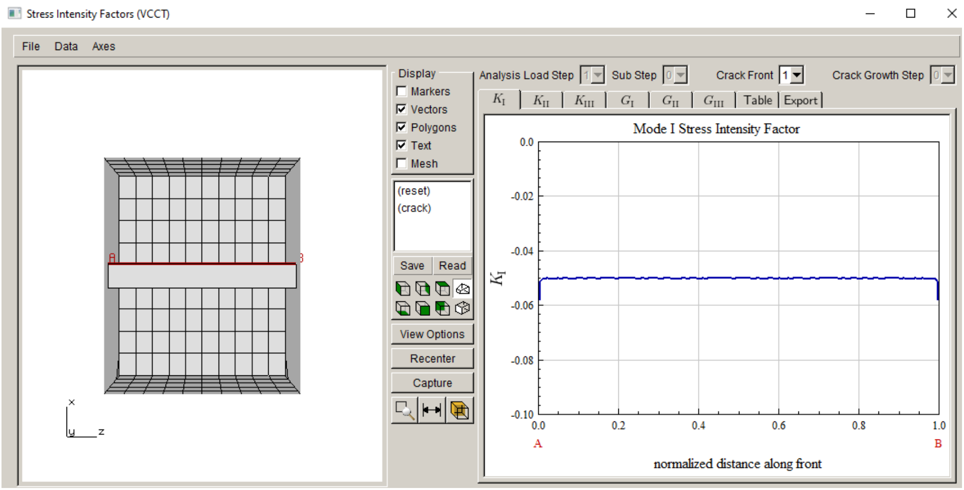


Stress Intensity Factors (Displacement Correlation)

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

Figure 13.33 ABAQUS Mode I SIFs for uniform compression with default CFC settings using DC.



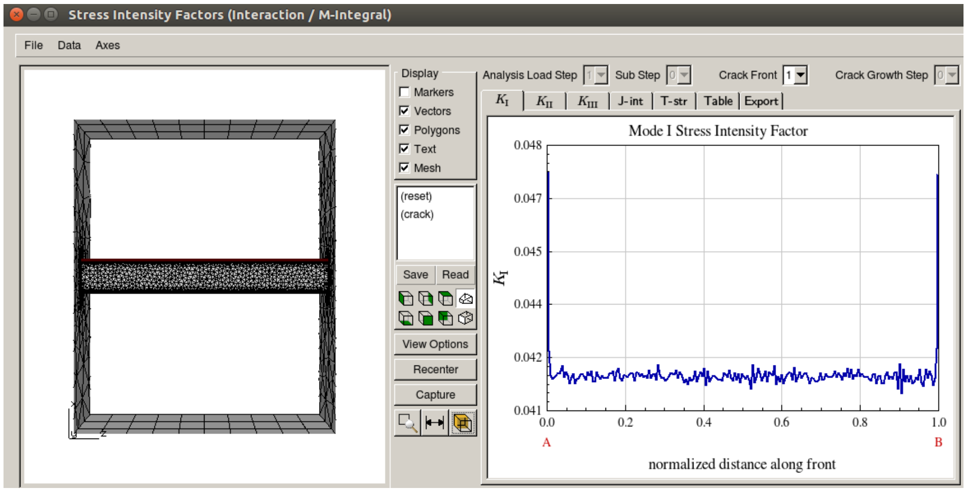
Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

□ Markers
☑ Vectors
☑ Polygons
☑ Text
□ Mesh

Figure 13.34 ABAQUS Mode I SIFs for uniform compression with default CFC settings using VCCT.

13.2.2.1 Coarse Mesh with Default Crack Face Contact (CFC)

Interestingly, a coarser mesh on the crack surface produces M-integral SIFs that are closer to zero. If we do not turn off the Crack Mouth Coarsening (see Fig 13.5), the SIFs are computed as shown in Fig 13.35.



Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

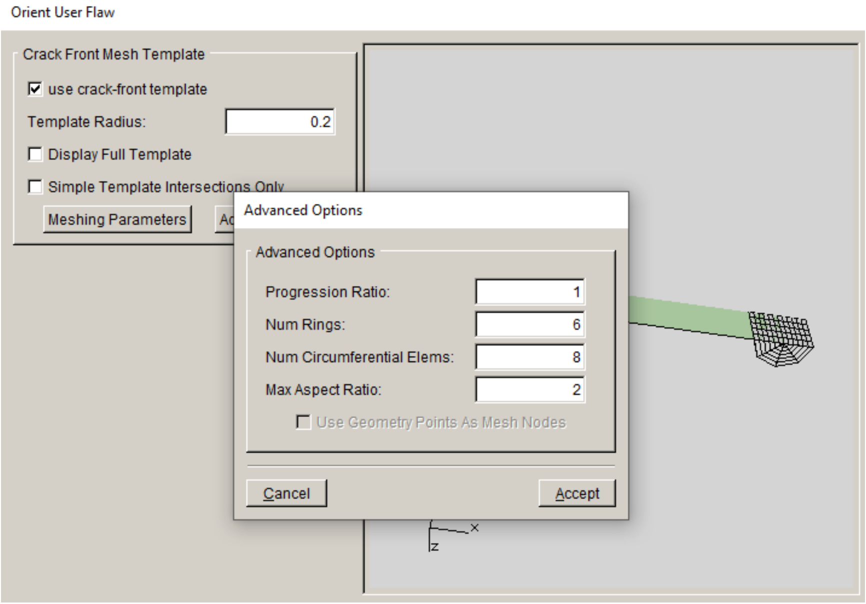
□ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

Figure 13.35 ABAQUS Mode I SIFs for uniform compression with a coarser mesh on the crack surface and default CFC settings using M-integral.

13.2.3 Modified Crack Front Template

Using the same meshing parameters as was shown in Fig 13.5, but with template parameters shown in Fig 13.36, the next analysis uses default CFC settings. The resulting SIFs are shown if

Figs 13.37-39; these values can be compared to the ANSYS results in Figs 13.24-26, and to the prior ABAQUS results for the default template.



Orient User Flaw

Crack Front Mesh Template

☑ use crack-front template

Template Radius: 0.2

☐ Display Full Template

☐ Simple Template Intersections Only

Advanced Options

Progression Ratio: 1

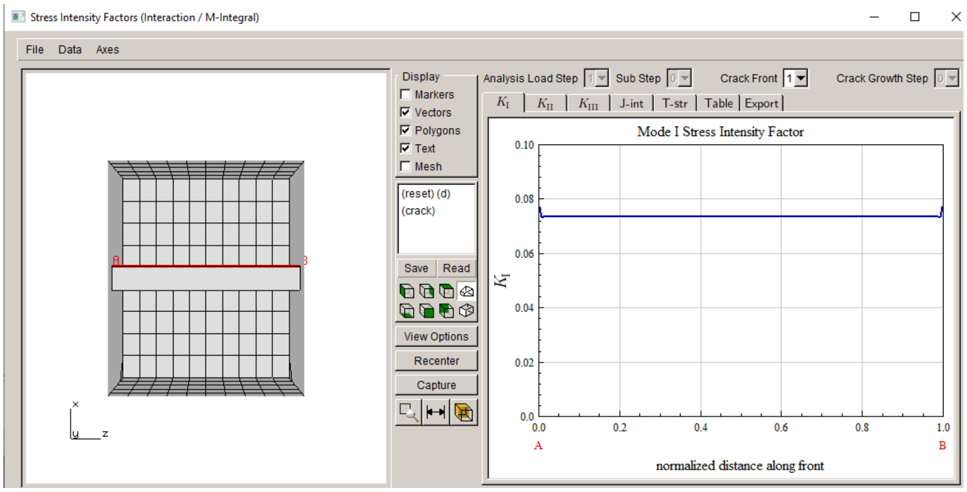
Num Rings: 6

Num Circumferential Elems: 8

Max Aspect Ratio: 2

☐ Use Geometry Points As Mesh Nodes

Figure 13.36 Center-through crack in ABAQUS plate with edited template mesh parameters.



Stress Intensity Factors (Interaction / M-Integral)

Display

☐ Markers

☑ Vectors

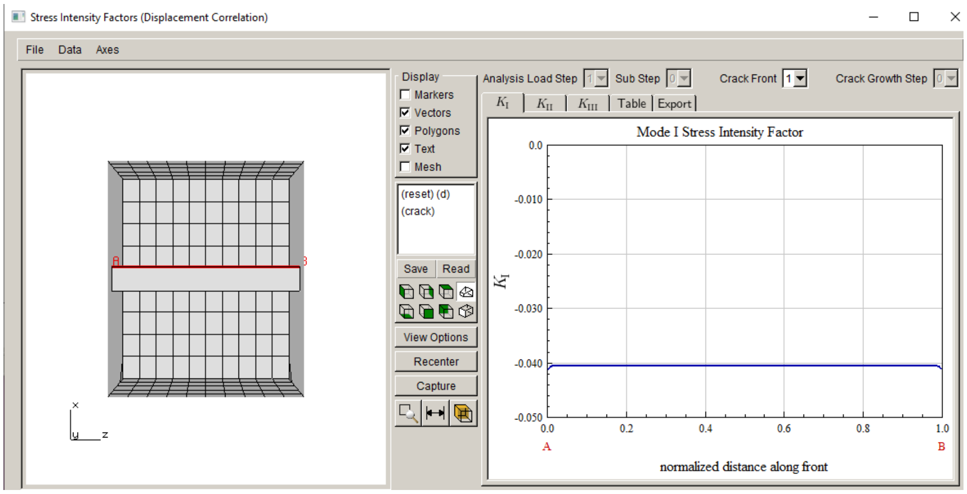
☑ Polygons

☑ Text

☐ Mesh

Analysis Load Step 1 Sub Step 0 Crack Front 1 Crack Growth Step 0

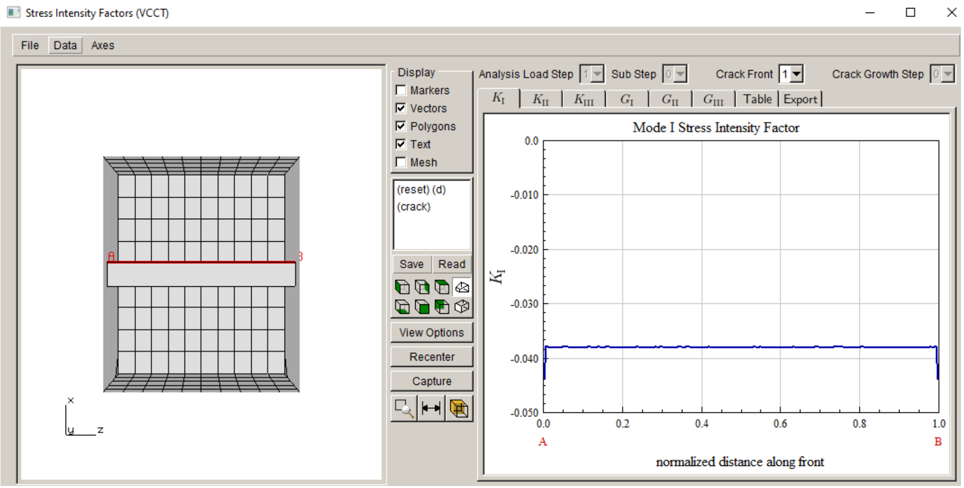
Figure 13.37 Mode I SIFs with modified template and default CFC using M-integral.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1
Sub Step 0
Crack Front 1
Crack Growth Step 0

Figure 13.38 Mode I SIFs with modified template and default CFC using DC.



☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

Analysis Load Step 1
Sub Step 0
Crack Front 1
Crack Growth Step 0

Figure 13.39 Mode I SIFs with modified template and default CFC using VCCT.

13.2.4 Tied Contact

For ANSYS, we tried “bonded” contact, which produced SIFs that are similar to standard contact. ABAQUS allows one to use ‘tied’ contact (see Fig 13.31). For this condition, ABAQUS ties the crack surfaces together such that the relative crack surface displacements are zero, Fig 13.40. However, there are positive pressures on the crack surface, with peak values of about 4.9. The combination of these crack face pressures and the stress (and strain) in the elements around the crack front leads to non-zero SIFs when using the M-integral, Fig 13.41. DC and VCCT SIFs are zero, Figs 13.42-43, as these are based on the zero-displacements.

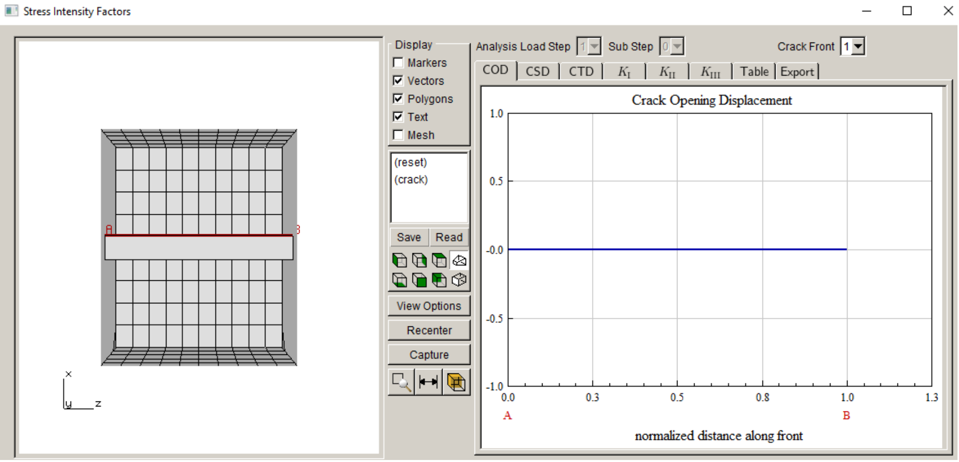


Figure 13.40 Crack opening displacement with tied CFC.

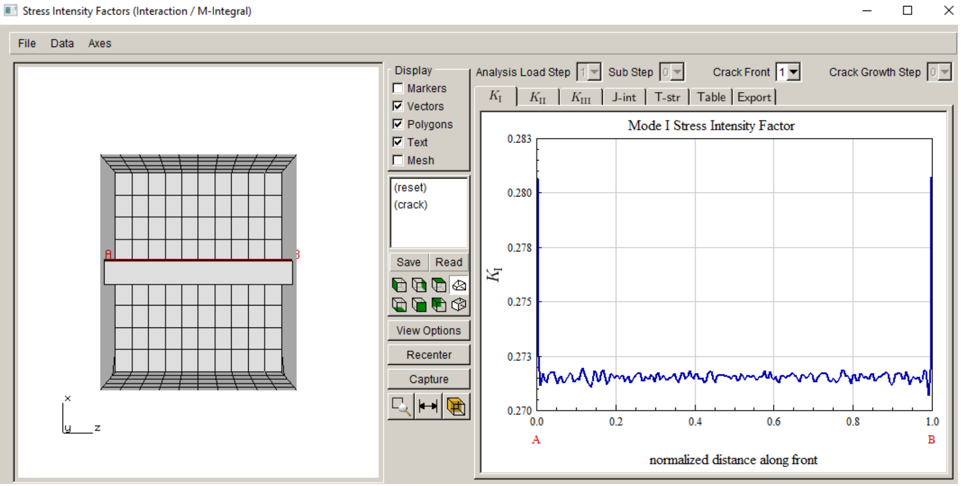


Figure 13.41 Mode I SIFs with default template and tied CFC using M-integral.

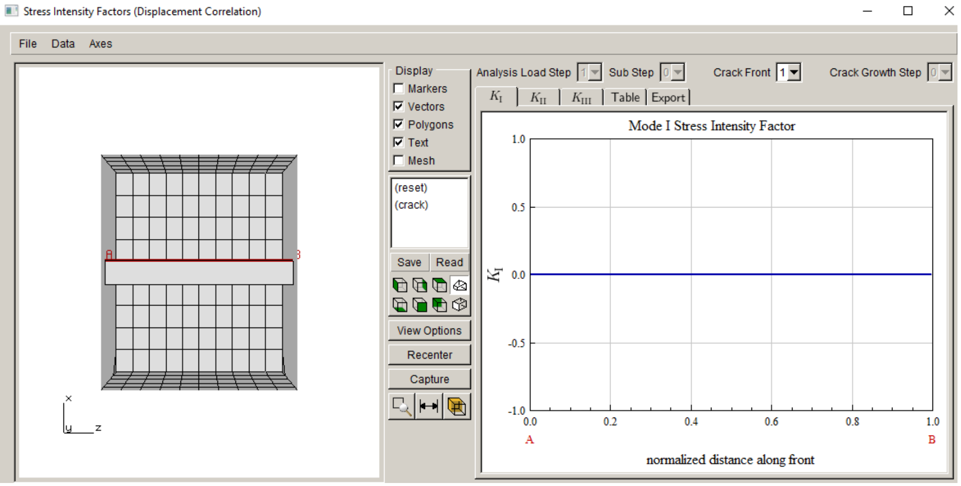


Figure 13.42 Mode I SIFs with default template and tied CFC using DC.

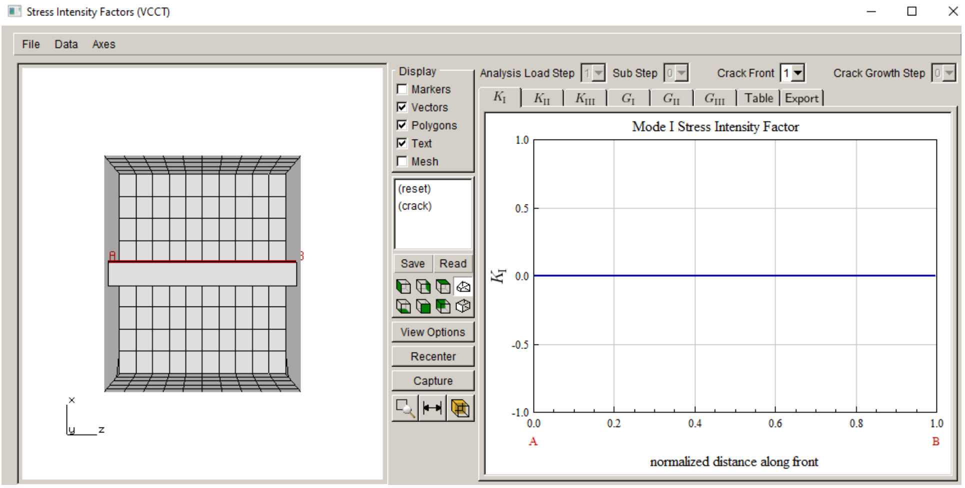


Figure 13.43 Mode I SIFs with default template and tied CFC using VCCT.

14. ABAQUS Initial Stress

The purpose of this chapter is to illustrate the different methods for incorporating "residual" stress in a FRANC3D crack growth simulation. The first section describes the base ABAQUS model that is used to generate the residual stress. The second section describes how to apply this stress as either crack face traction (CFT) or as initial stress. This chapter is mainly for ABAQUS users.

14.1 ABAQUS Residual Stress

We start with a simple plate model with sufficient constraint to prevent rigid body motion. The material properties include a yield stress and perfect plasticity. The plate is kept at a uniform constant temperature. The first load step applies a uniform displacement (in the –y direction) to the upper surface, as shown in Fig 14.1. In the second load step, the y-displacement is reset back to 0.0. Some "residual" stress (and strain) remains at the end of the second load step.

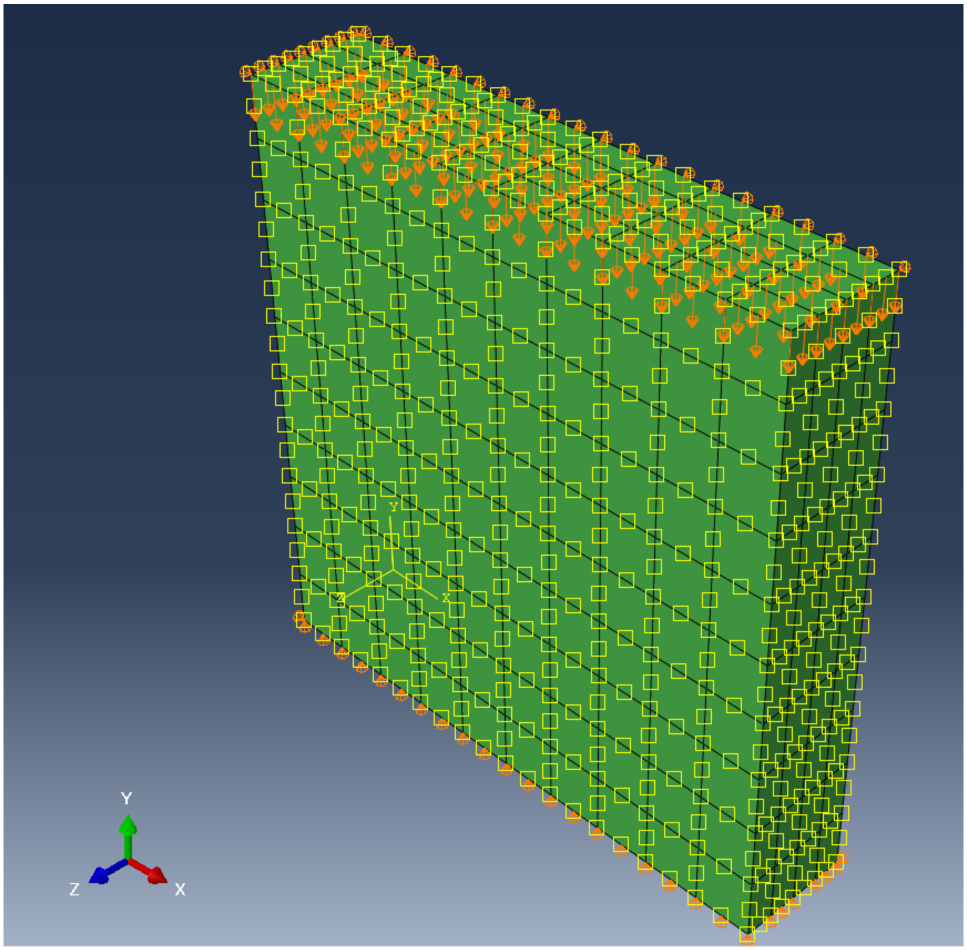


Figure 14.1 Plate with applied displacement from load step 1.

The von Mises stress at the end of the unloading (load step 2) is shown in Fig 14.2a; the maximum principal strain is shown in Fig 14.2b. The residual stress is uniform throughout the plate; this makes applying initial stress conditions for the cracked (remeshed) model quite simple (described later).

The stress components for this model are exported to a .dtp file, which FRANC3D can import when applying CFTs. We export the stress for all load steps and substeps, and then select the appropriate data from this file in FRANC3D for the CFTs (described later).

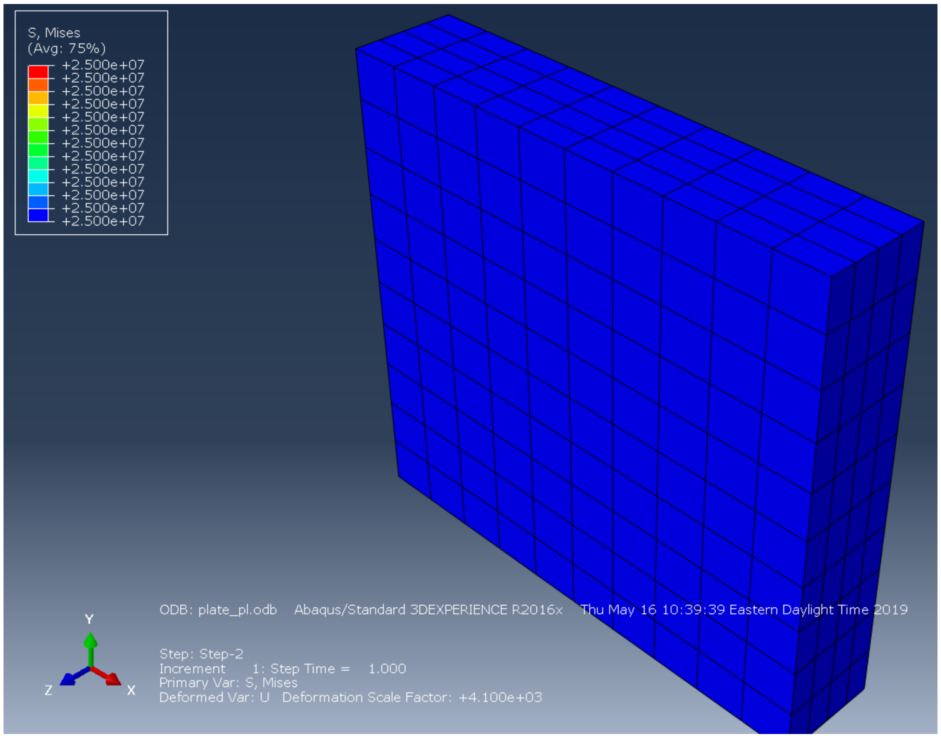
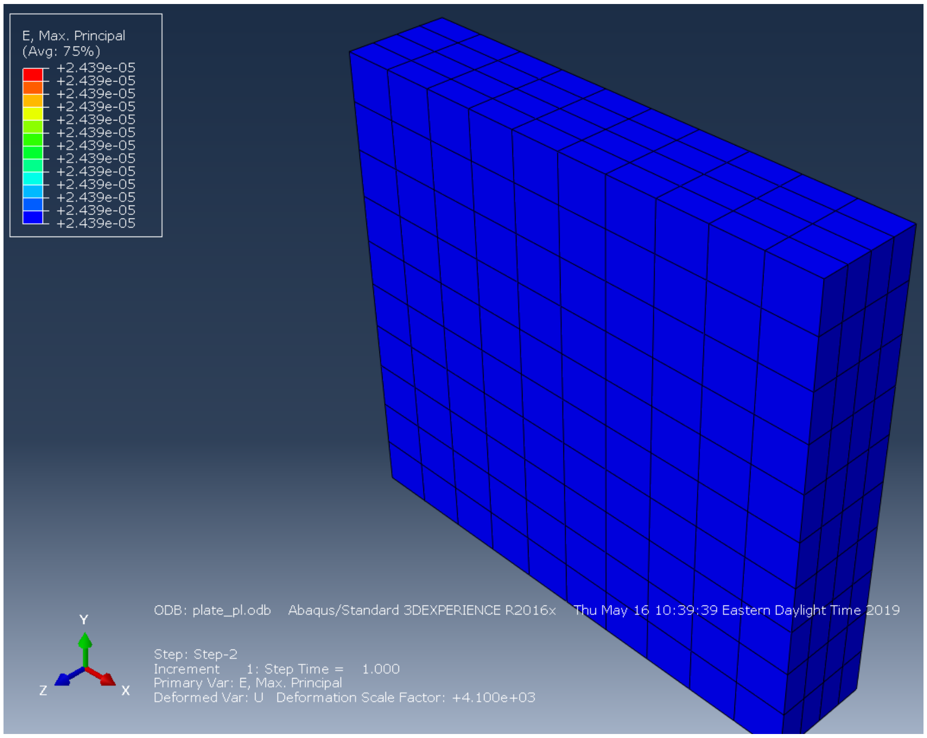

S, Mises
(Avg: 75%)
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
+2.500e+07
ODB: plate_pl.odb   Abaqus/Standard 3DEXPERIENCE R2016x   Thu May 16 10:39:39 Eastern Daylight Time 2019
Step: Step-2
Increment   1: Step Time =   1.000
Primary Var: S, Mises
Deformed Var: U   Deformation Scale Factor: +4.100e+03
Y
Z
X

Figure 14.2a von Mises stress after unloading (load step 2: frame 1).

14.2 Residual Stress as Initial Stress

This same plate is used to illustrate how to apply the initial stress conditions in the uncracked case. This section shows how ABAQUS can apply residual stress as initial conditions using ABAQUS files – if the mesh is the same.

First, the plastic material data is removed from the above model; we simply edit the .inp file and comment out the *plastic data. The new elastic model file is named: plate.inp. In addition, we remove the y-displacement boundary condition on the upper surface; all other constraints remain. For the first elastic model, the initial stress conditions are not applied. Therefore, the first load step does not include any loads or non-zero displacements. The second load step applies a fixed uniform displacement to the top surface, Fig 14.3. The von Mises stress at the end of load step 2, Fig 14.4; load step 1 produces no stress, so we do not plot it.

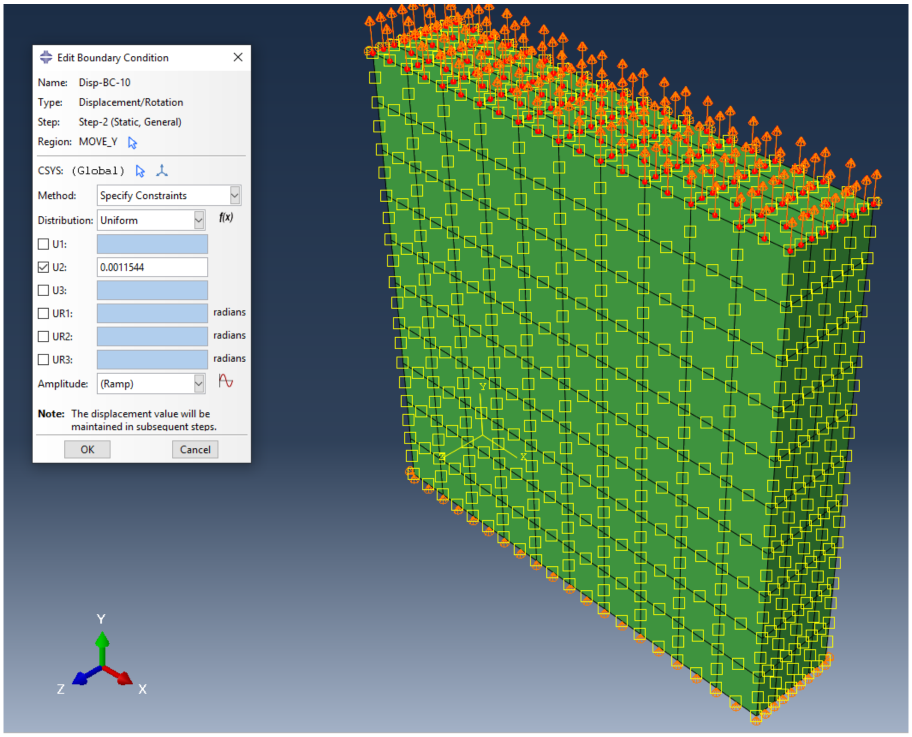


E, Max. Principal
(Avg: 75%)

🟥 +2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
+2.439e-05
🟦 +2.439e-05

```
ODB: plate_pl.odb    Abaqus/Standard 3DEXPERIENCE R2016x    Thu May 16 10:39:39 Eastern Daylight Time 2019

Step: Step-2
Increment      1: Step Time =    1.000
Primary Var: E, Max. Principal
Deformed Var: U   Deformation Scale Factor: +4.100e+03
```

Figure 14.2b Max principal strain after unloading (load step 2: frame 1).



Edit Boundary Condition

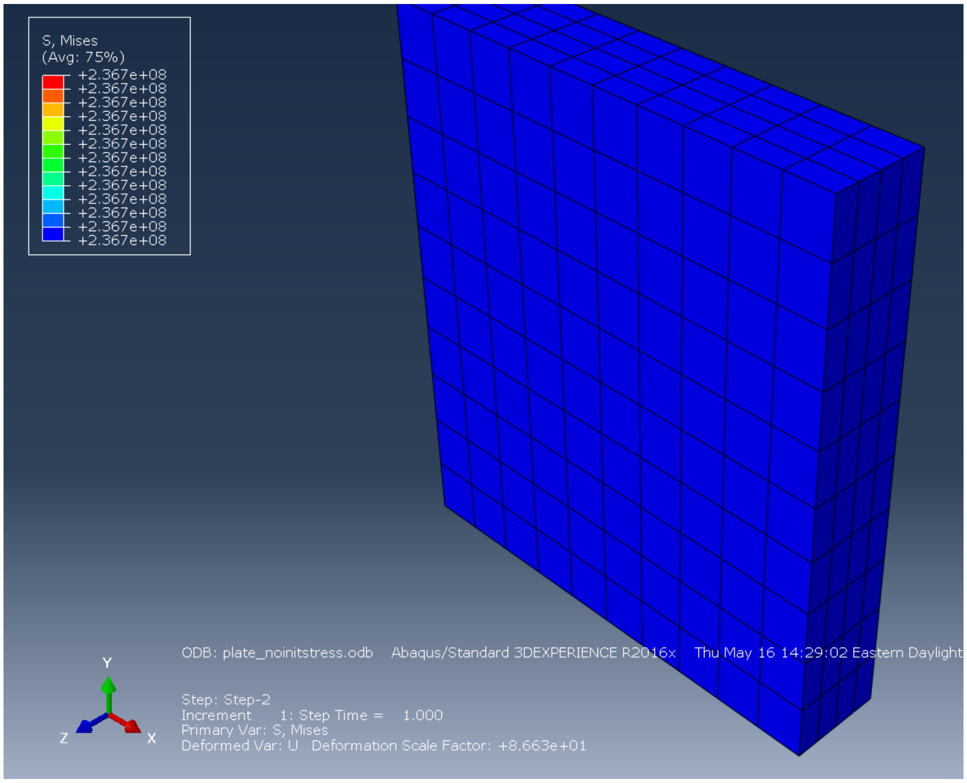
Name: Disp-BC-10
Type: Displacement/Rotation
Step: Step-2 (Static, General)
Region: MOVE_Y

CSYS: (Global)

Method: Specify Constraints
Distribution: Uniform
□ U1:
☑ U2: 0.0011544
□ U3:
□ UR1: radians
□ UR2: radians
□ UR3: radians
Amplitude: (Ramp)

Note: The displacement value will be maintained in subsequent steps.

Figure 14.3 Plate model without plastic properties, showing applied displacement for load step 2.



ODB: plate_noinitstress.odb   Abaqus/Standard 3DEXPERIENCE R2016x   Thu May 16 14:29:02 Eastern Daylight

Step: Step-2
Increment   1: Step Time =   1.000
Primary Var: S, Mises
Deformed Var: U   Deformation Scale Factor: +8.663e+01

Figure 14.4 von Mises stress at the end of load step 2 for elastic plate without initial stress.

The second elastic model (modified from the first elastic model) includes the following initial stress condition data:

*Initial Conditions, type=STRESS, file=plate_pl.odb, step=2, inc=1

where plate_pl.odb corresponds to the results for the model with plastic properties. The initial stress is the only "loading" condition in load step 1. The von Mises stress for this model for load step 1 (frame 0 and frame1) is shown in Figs 14.5a-b; the initial stress (in frame 0) matches the residual stress shown in Fig 14.2.

The same uniform applied displacement in load step 2 of the first elastic model (see Fig 14.3), is applied in this second model. The von Mises stress at the end of load step 2 is shown in Fig 14.6. The stress is slightly different from that shown in Fig 14.4 because it includes the effects of the initial stress.

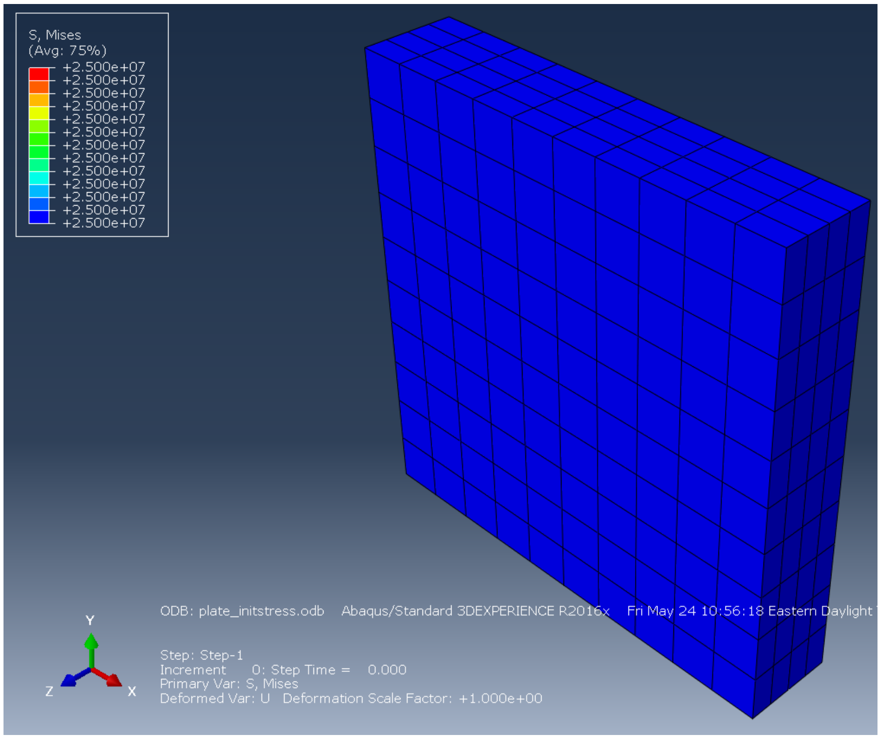


Figure 14.5a von Mises stress for load step 1 – frame 0 in the elastic plate with initial stress.

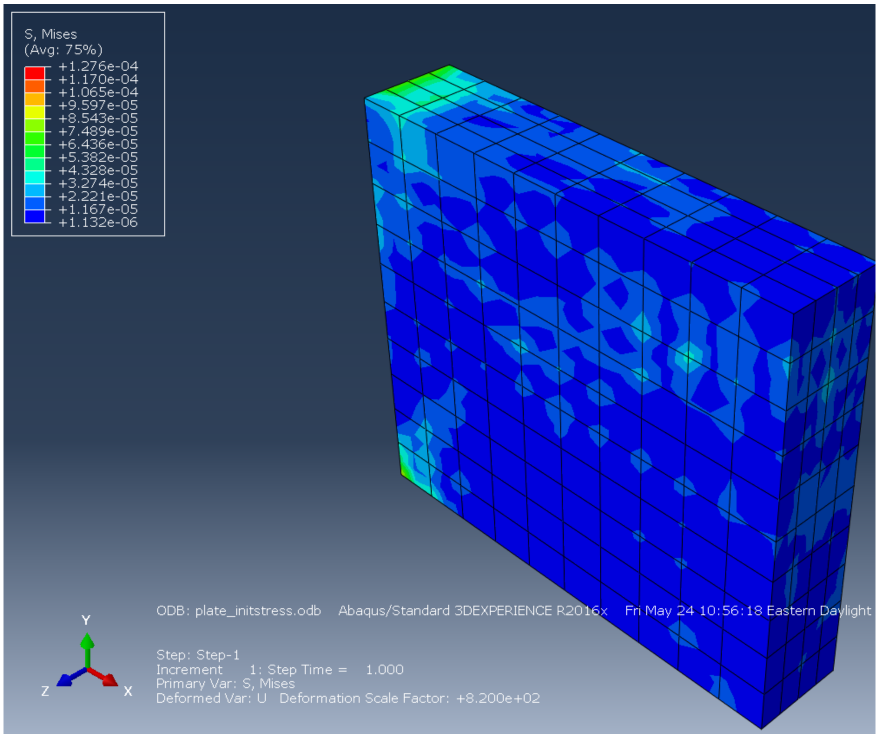


Figure 14.5b von Mises stress for load step 1 – frame 1 in the elastic plate with initial stress.

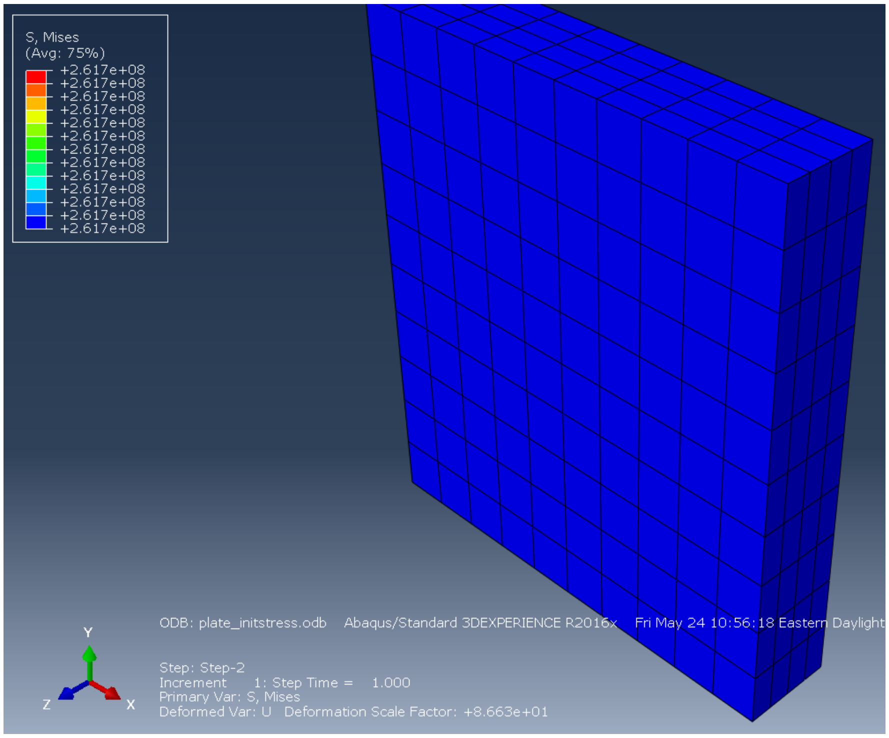


Figure 14.6 von Mises stress for load step 2 in the elastic plate with initial stress.

14.3 Residual Stress Included in FRANC3D

This section describes how one can apply a “residual” stress in a FRANC3D simulation. Using the first elastic model (no initial stress), an edge crack is inserted, Fig 14.7. The original boundary conditions and load steps are retained. An extra crack face traction (CFT) is added, Fig 14.8, using the plastic model mesh file and the “residual” stress from load step 2 (frame 1) of the same plastic model.

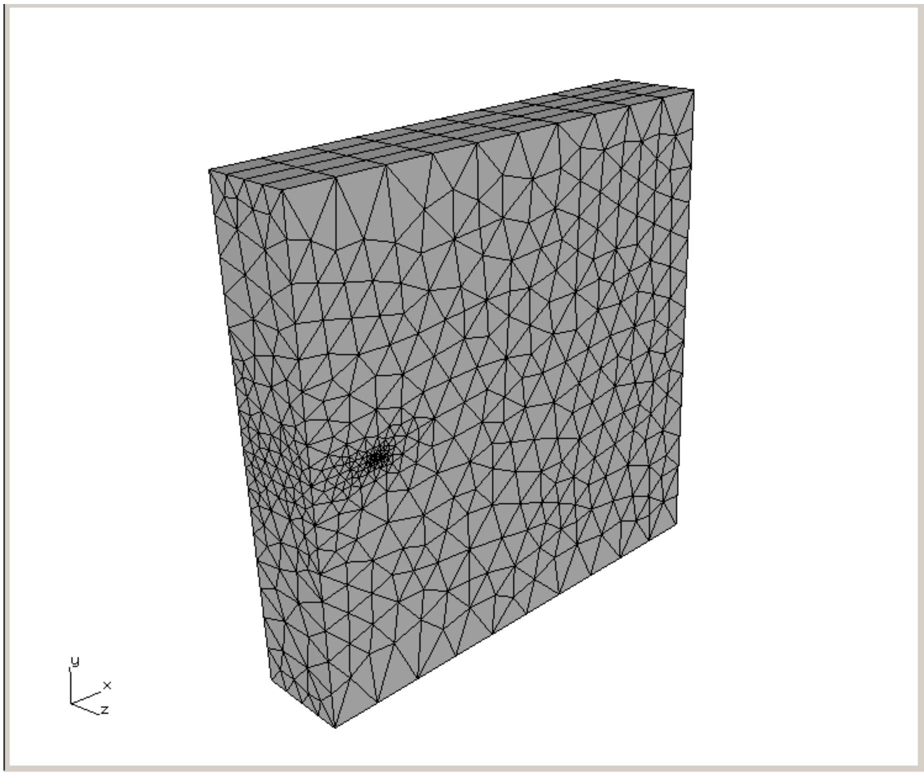
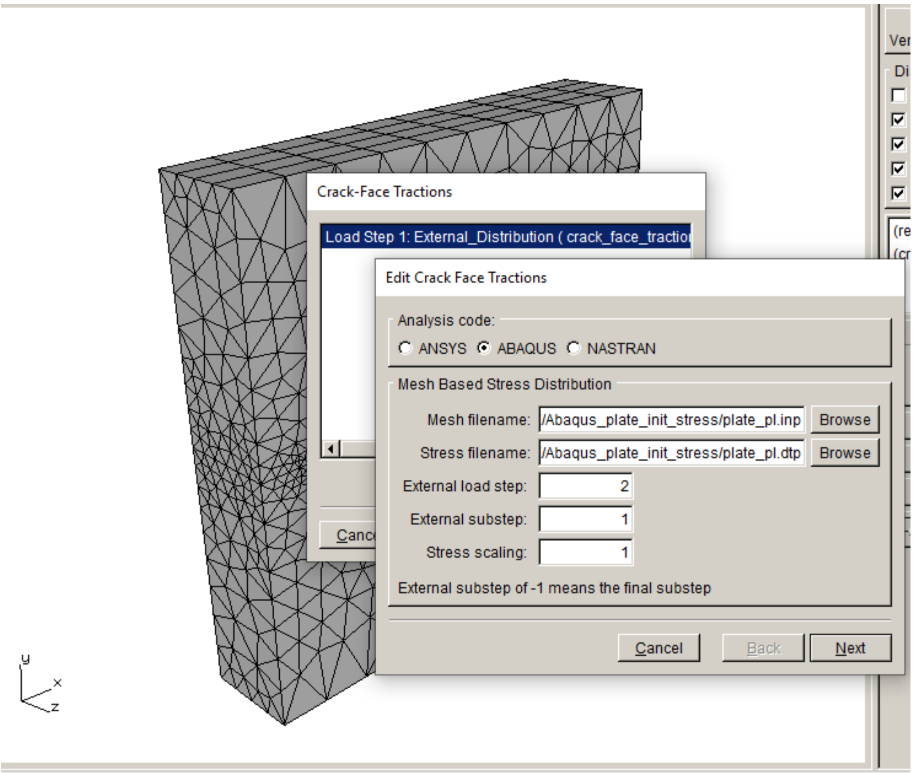


Figure 14.7 Edge crack inserted into the plate model in FRANC3D.



Crack-Face Tractions

Load Step 1: External_Distribution (crack_face_tractio

Edit Crack Face Tractions

Analysis code:

○ ANSYS ● ABAQUS ○ NASTRAN

Mesh Based Stress Distribution

Mesh filename: /Abaqus_plate_init_stress/plate_pl.inp Browse

Stress filename: /Abaqus_plate_init_stress/plate_pl.dtp Browse

External load step: 2

External substep: 1

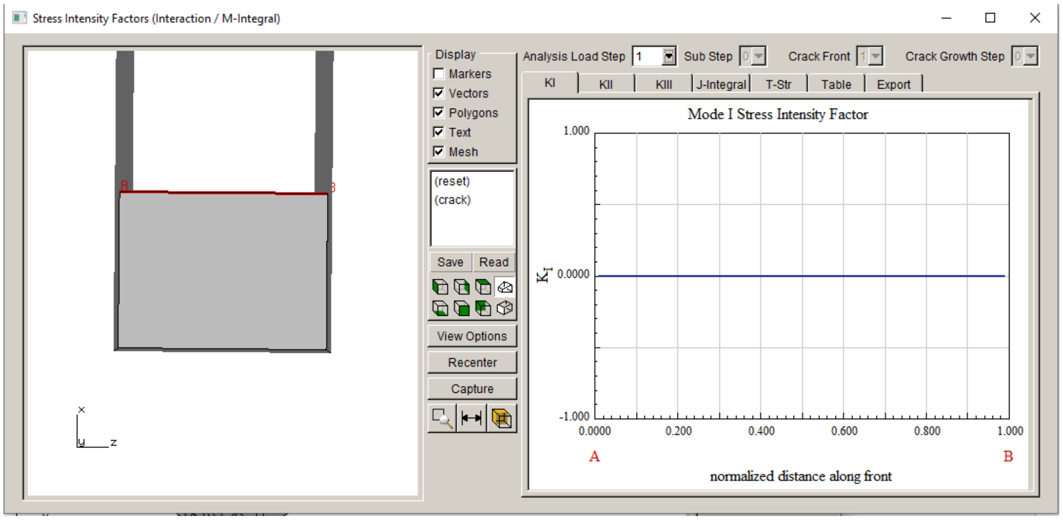
Stress scaling: 1

External substep of -1 means the final substep

Cancel Back Next

Figure 14.8 Crack face traction load step added using the plastic model and stress data.

The resulting stress intensity factors (SIFs) are shown in Figs 14.9a-d. The first load step does not include any loading condition nor initial stress, so the SIFs are all zero. The second load step produces positive mode I SIFs, Fig 14.9b, and the third load step, which is the CFT load, (based on the residual stress) produces the SIFs shown in Fig 14.9c. Fig 14.9d shows the sum of the Mode I SIFs.

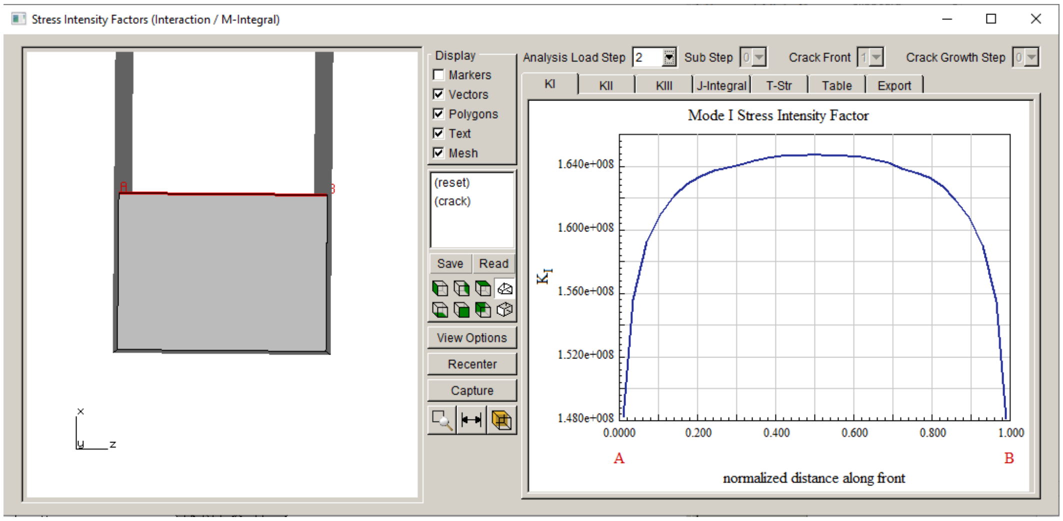


Stress Intensity Factors (Interaction / M-Integral)

Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 14.9a Mode I SIFs for load step 1.

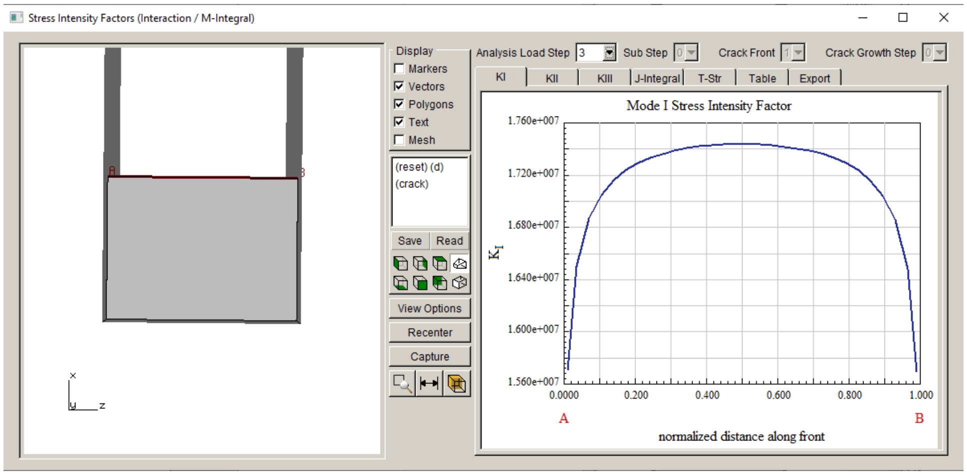


Stress Intensity Factors (Interaction / M-Integral)

Display
□ Markers
☑ Vectors
☑ Polygons
☑ Text
☑ Mesh

Analysis Load Step 2 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 14.9b Mode I SIFs for load step 2.



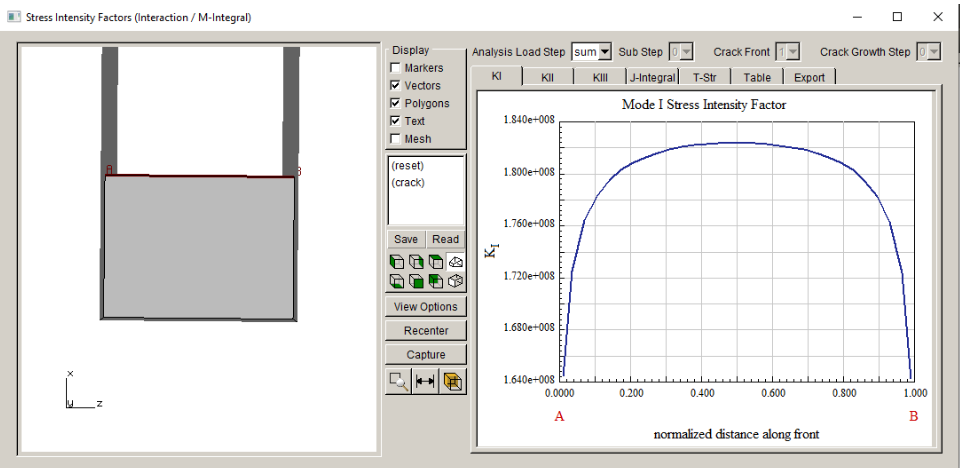
Stress Intensity Factors (Interaction / M-Integral)

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

(reset) (d)
(crack)

Analysis Load Step: 3 · Sub Step: 0 · Crack Front: 1 · Crack Growth Step: 0

Figure 14.9c Mode I SIFs for load step 3 – the CFT loading.



Stress Intensity Factors (Interaction / M-Integral)

Display: ☐ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

(reset)
(crack)

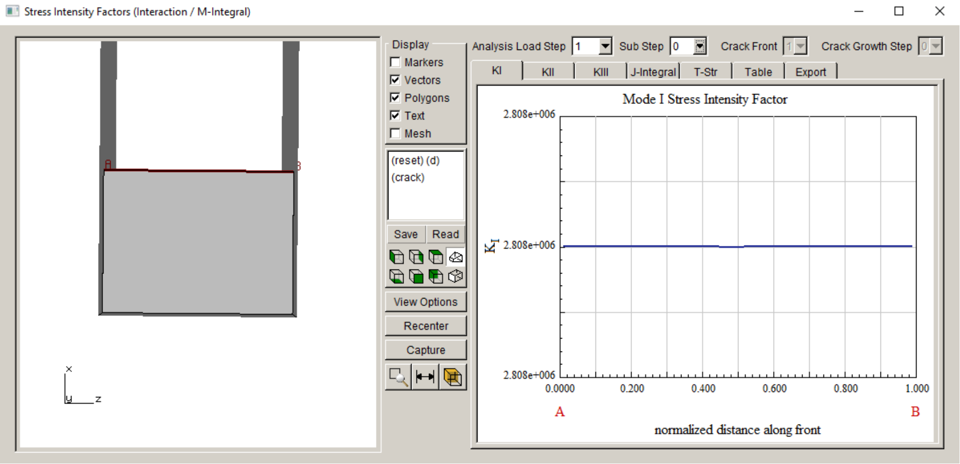
Analysis Load Step: sum · Sub Step: 0 · Crack Front: 1 · Crack Growth Step: 0

Figure 14.9d Sum of Mode I SIFs.

The same edge crack is inserted into the second elastic model, which includes the initial stress conditions. The initial stress condition is manually edited in the cracked *.inp* file to apply a uniform stress state to all elements:

```
 *Initial Conditions, type=STRESS
   all_elements, 0.0, 25000002.0, 0.0, 0.0, 0.0, 0.0
```

where the stress values are taken from the plastic model "residual" stress state. The resulting SIFs are shown in Figs 14.10a-c. Fig 14.10a shows the SIFs for load step 1 and frame 0, which are due to the residual stress. Fig 14.10b shows the SIFs for load step 1 and frame 1, and Fig 14.10c shows the SIFs for the final frame of load step 2; these SIFs match those shown in Fig 14.9d.



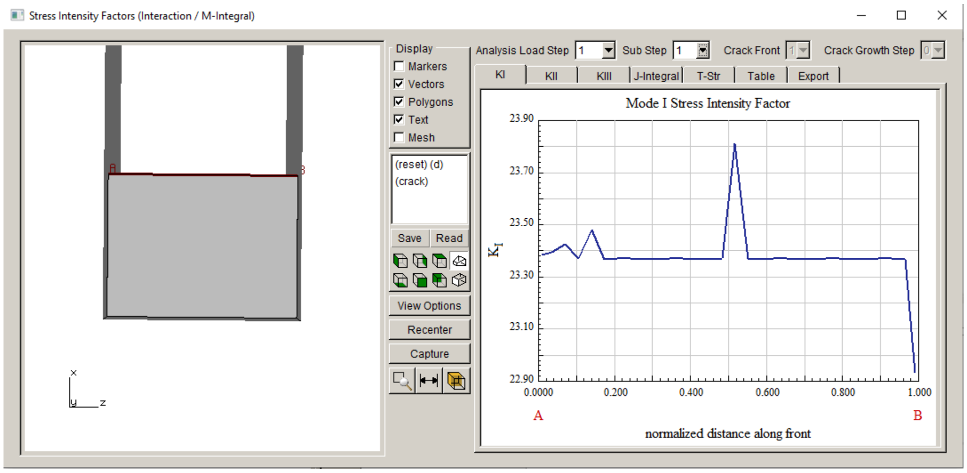
Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Figure 14.10a Mode I SIFs for load step 1 and frame 0 – initial stress condition.



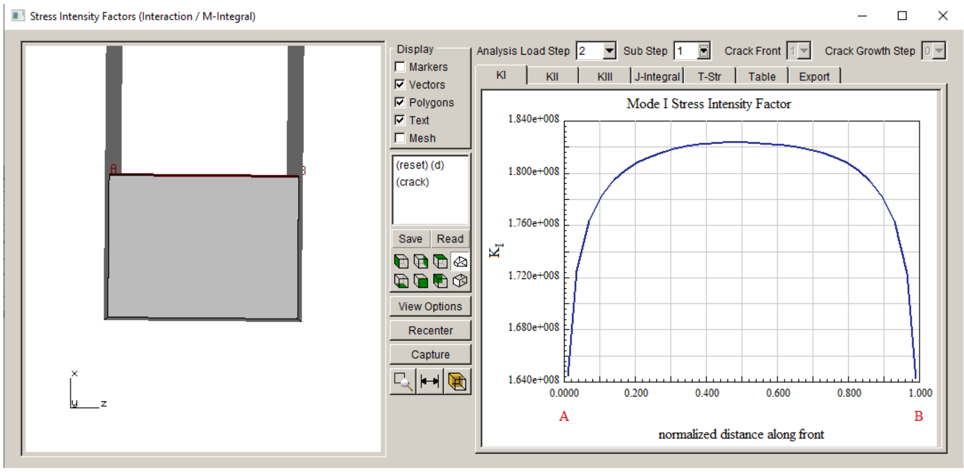
Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 1 | Crack Front 1 | Crack Growth Step 0

Figure 14.10b Mode I SIFs for load step 1 and frame 1 – initial stress balanced.



Stress Intensity Factors (Interaction / M-Integral)

Display
☐ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 2 | Sub Step 1 | Crack Front 1 | Crack Growth Step 0

Figure 14.10c Mode I SIFs for load step 2.

The ‘*initial conditions, type=stress’ option in ABAQUS can be based on a file as was done in the second section, but this only works if the mesh is identical. The option used in this section assigns a uniform stress state to all elements in the cracked-remeshed model. This works in this case because the residual stress is uniform. For more general cases, the ‘*initial conditions, type=stress, USER’ option, or possibly the *Map Solution command, could be used.

Note that CFT loads added as extra load steps cannot permit thermal expansion (thermal stress or strain). However, new CFT capabilities have been added to allow CFTs to be added to existing load steps, and this will allow for thermal expansion. This new CFT capability should be able to reproduce the same SIFs as an initial stress condition subjected to temperature changes.

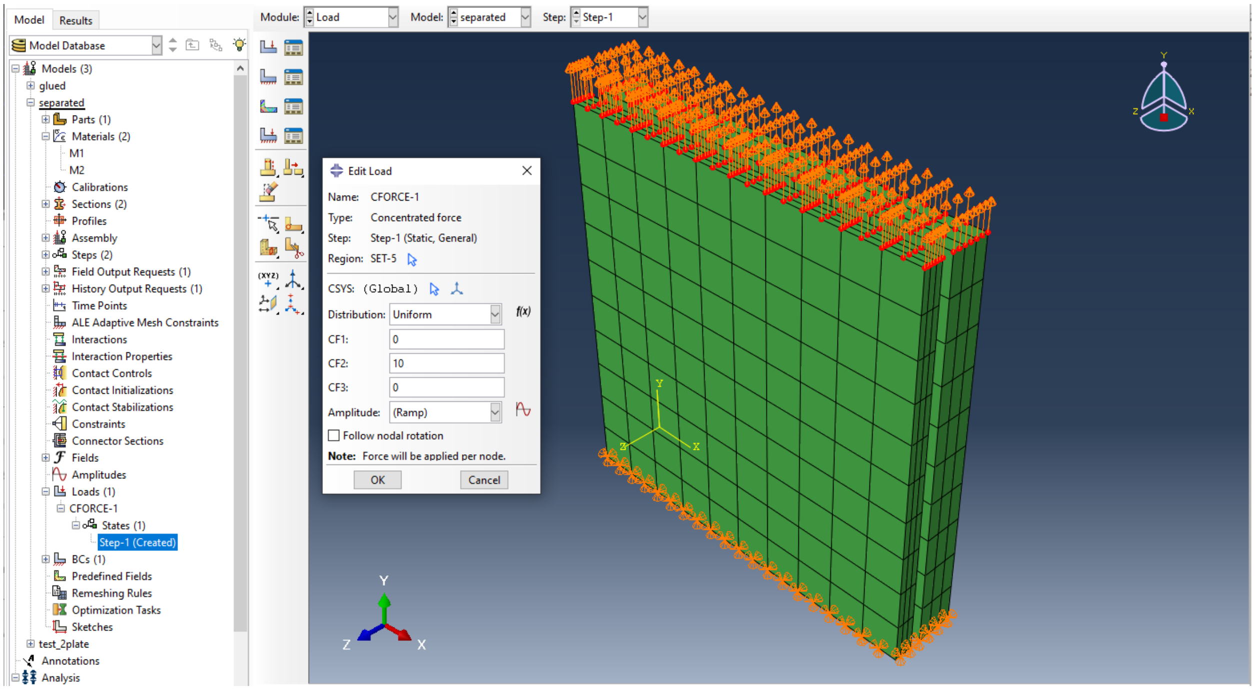
15. Using Python to Extend FRANC3D Capabilities:

Separate Constrained Plates Compared with Glued Plates

This chapter describes how one can simulate crack growth in two plates and compares the SIFs for three cases: separate plates with no interaction, separate plates that are constrained after crack insertion, and glued plates with a crack inserted using FRANC3D Version 8. This chapter uses ABAQUS, but ANSYS or NASTRAN models could be modified in an analogous manner.

15.1 Uncracked Models

The first model consists of two separate plates with a space between the plates, Fig 15.1. The bottoms of the two plates are fully constrained. Nodal forces are applied to the top surface. The plates are 100x100 units in the x and y directions. One plate is 10 units deep and the other plate is 5 units deep. The space between the plates is 3 units. Two isotropic elastic materials are defined; the first has E=16000 and Nu=0.3 and is attached to the thicker plate. The second material has E=15999 and Nu=0.299 and is attached to the thinner plate. The mesh is 10x10 in the x and y directions with 3 elements through the thickness for both plates.



Module: Load   Model: separated   Step: Step-1

Edit Load

Name: CFORCE-1

Type: Concentrated force

Step: Step-1 (Static, General)

Region: SET-5

CSYS: (Global)

Distribution: Uniform

CF1: 0

CF2: 10

CF3: 0

Amplitude: (Ramp)

□ Follow nodal rotation

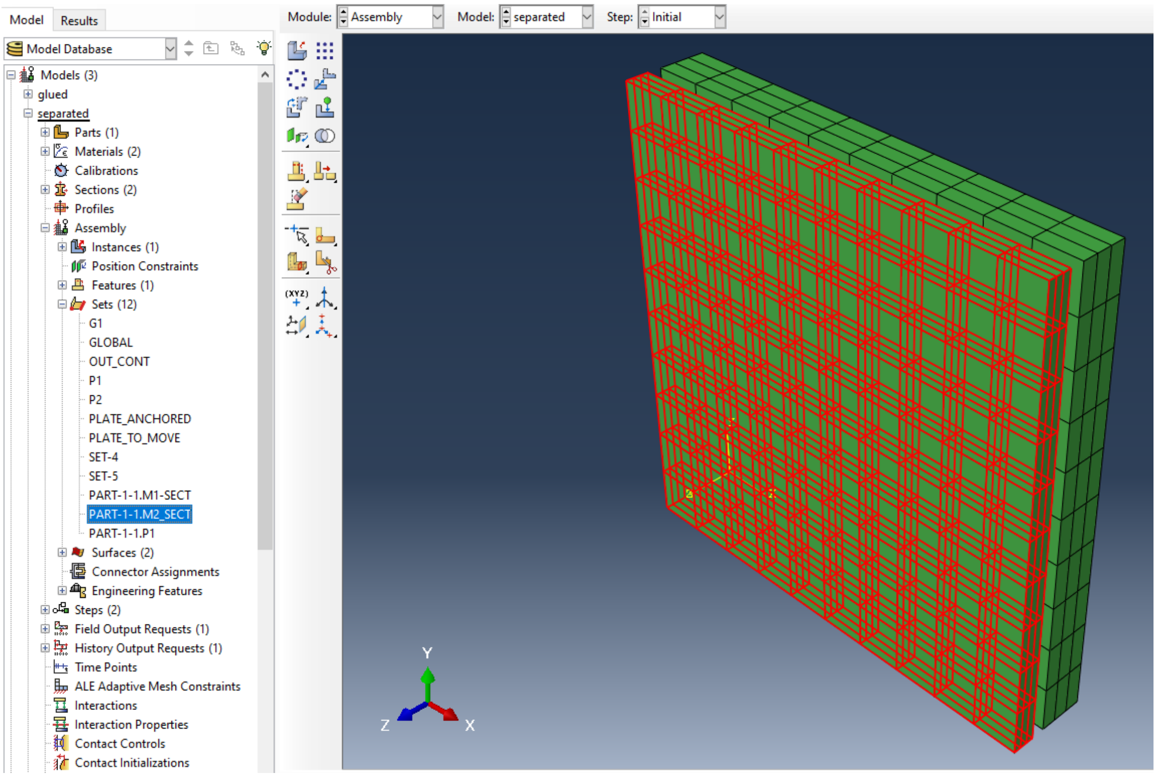
Note: Force will be applied per node.

OK   Cancel

Figure 15.1 Separate plate model.

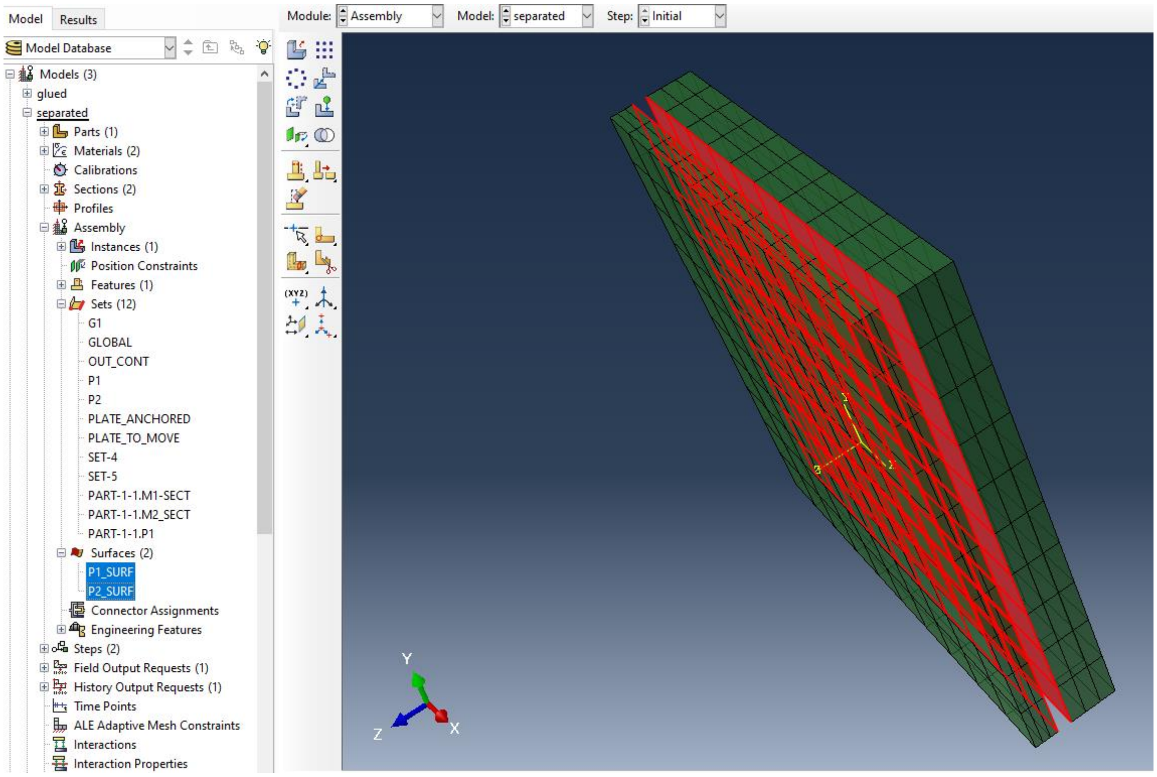
15.1.1 ABAQUS Sets and Surfaces

Sets and surfaces are defined for the uncracked model and will be used in a Python script later. Each plate has an element set associated with it; Fig 15.1b shows the M2_SECT set. The ‘mate’ surfaces for both plates are created, Fig 15.1c.



Module: Assembly Model: separated Step: Initial

Figure 15.1b Separate plate model – element set for the thinner plate.

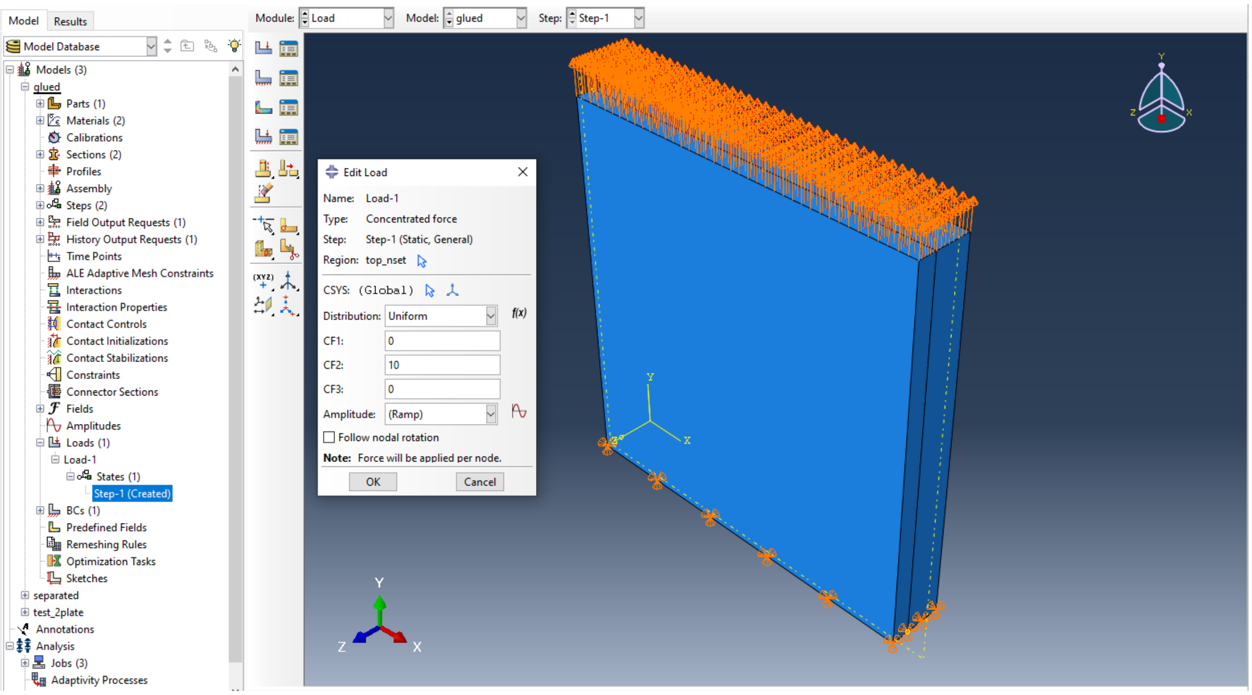


Module: Assembly Model: separated Step: Initial

Figure 15.1c Separate plate model – ‘mate’ surfaces for both plates.

15.1.2 Glued Plates

The second model, Fig 15.2, does not have a space between the plates. The interface is shared by the two plates; the element faces on the interface are identical and shared by elements on either side of the interface. The dimensions, constraints, loads, material properties and mesh density are all consistent with the first model.



Module: Load Model: glued Step: Step-1

Edit Load

Name: Load-1

Type: Concentrated force

Step: Step-1 (Static, General)

Region: top_nset

CSYS: (Global)

Distribution: Uniform

CF1: 0

CF2: 10

CF3: 0

Amplitude: (Ramp)

□ Follow nodal rotation

Note: Force will be applied per node.

Figure 15.2. Glued plate model.

15.2 Uncracked Plate Analysis

The initial analysis of both models is performed to create the *.inp* files. The deformed shapes for both models are shown in Figs 15.3 and 15.4.

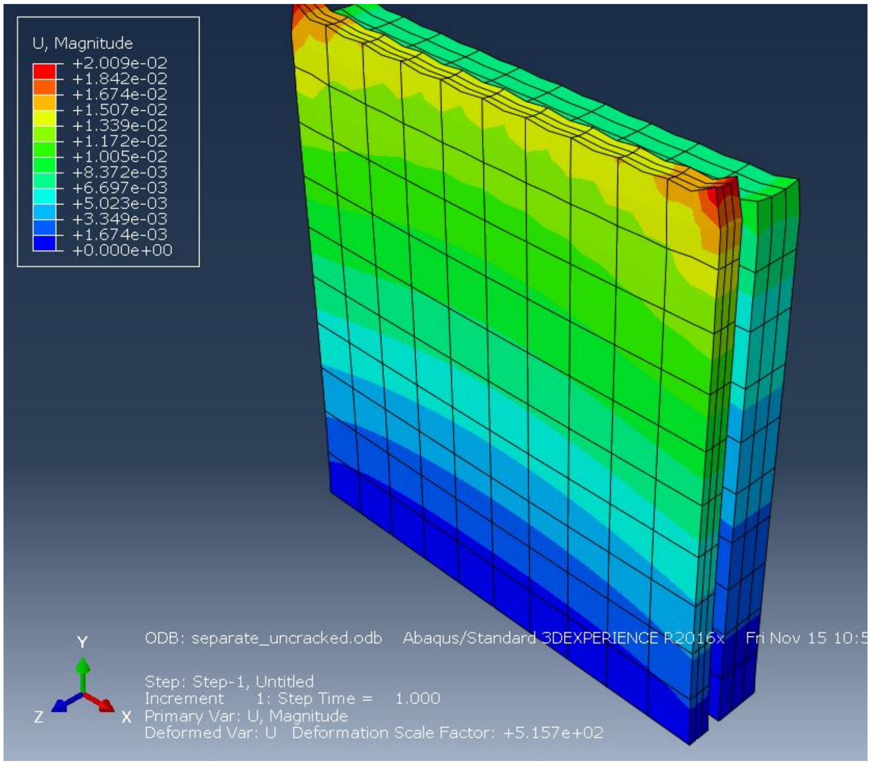

U, Magnitude
+2.009e-02
+1.842e-02
+1.674e-02
+1.507e-02
+1.339e-02
+1.172e-02
+1.005e-02
+8.372e-03
+6.697e-03
+5.023e-03
+3.349e-03
+1.674e-03
+0.000e+00
Y
Z
X
ODB: separate_uncracked.odb Abaqus/Standard 3DEXPERIENCE R2016x Fri Nov 15 10:5
Step: Step-1, Untitled
Increment 1: Step Time = 1.000
Primary Var: U, Magnitude
Deformed Var: U Deformation Scale Factor: +5.157e+02

Figure 15.3 Separate plate model deformed shape.

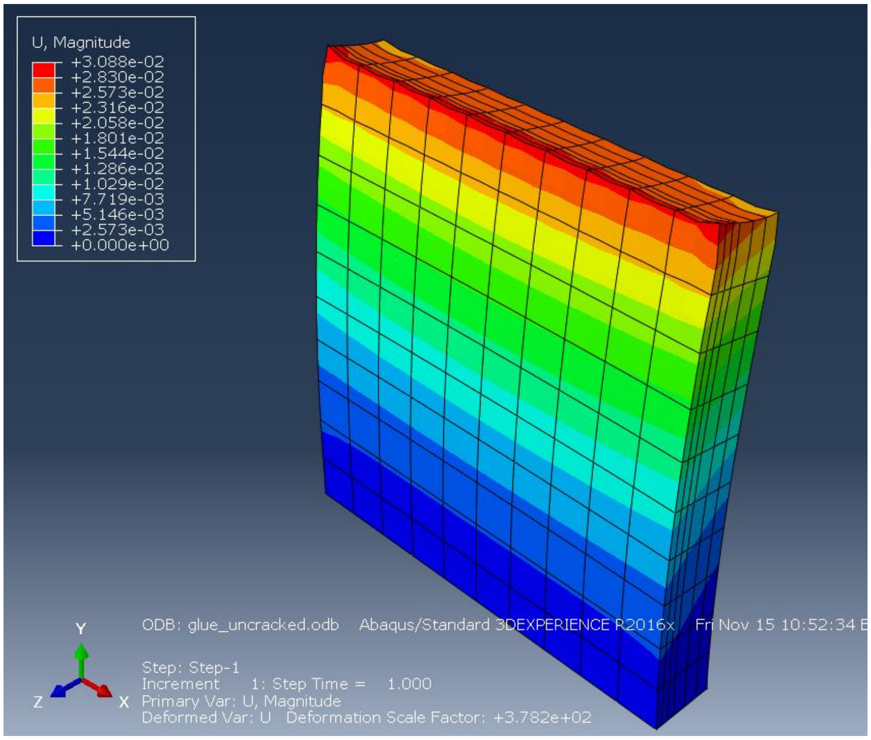

U, Magnitude
+3.088e-02
+2.830e-02
+2.573e-02
+2.316e-02
+2.058e-02
+1.801e-02
+1.544e-02
+1.286e-02
+1.029e-02
+7.719e-03
+5.146e-03
+2.573e-03
+0.000e+00
Y
Z
X
ODB: glue_uncracked.odb Abaqus/Standard 3DEXPERIENCE R2016x Fri Nov 15 10:52:34 E
Step: Step-1
Increment 1: Step Time = 1.000
Primary Var: U, Magnitude
Deformed Var: U Deformation Scale Factor: +3.782e+02

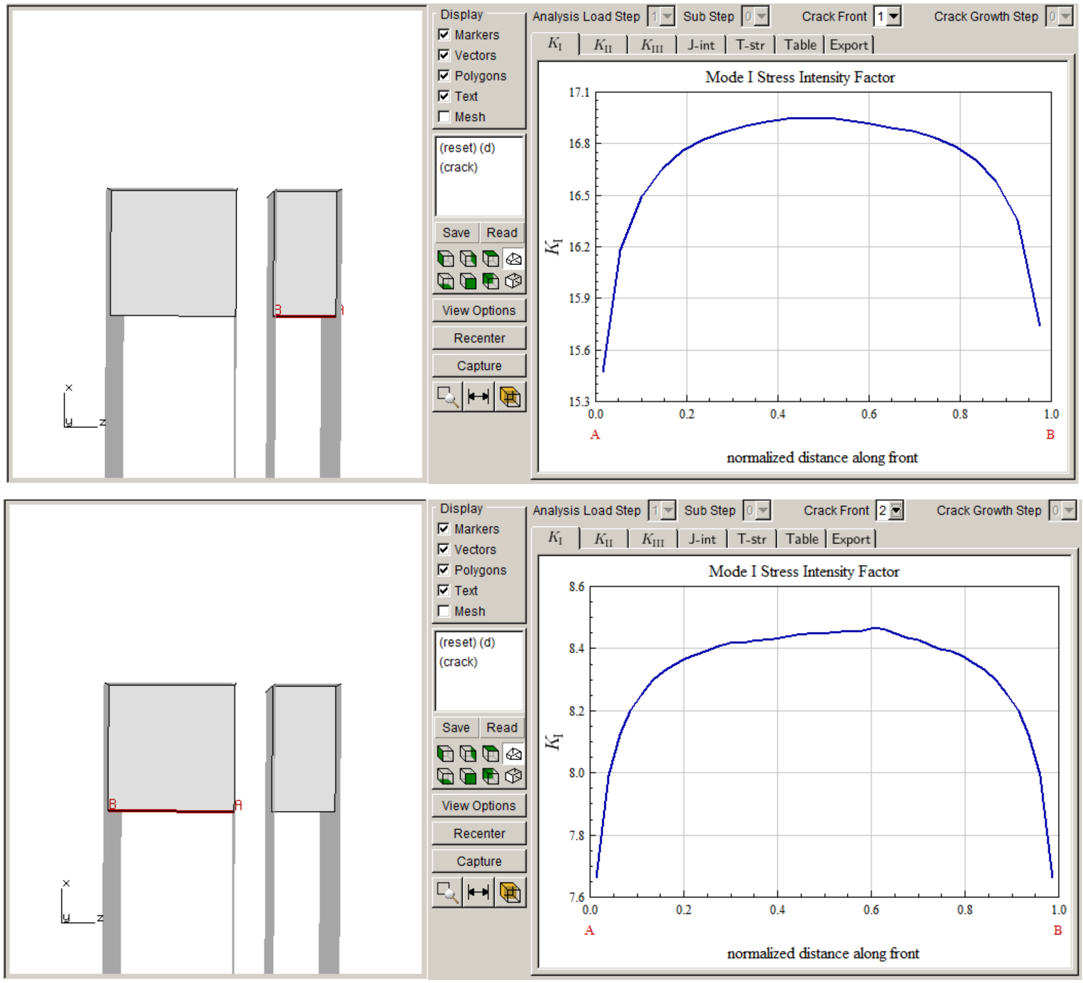
Figure 15.4 Glued plate model deformed shape.

15.3 Initial Crack Analysis

Through-edge cracks are inserted into the two models. The cracks are 10 units long once inserted (the FRANC3D defined dimensions are slightly bigger). The cracks are inserted at the mid-y location and the +100-x location.

15.3.1 Separate Plates

For the separate plate model, two separate edge cracks are defined and inserted.
The first analysis consists of the two separate plates with no interaction between the plates. The Mode I SIFs are shown in Fig 15.5; the crack fronts are identified in the image on the left. The SIFs for the thinner plate are double those of the thicker plate (as expected).



Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Analysis Load Step 1 · Sub Step 0 · Crack Front 1 · Crack Growth Step 0

(reset) (d)
(crack)

Display: ☑ Markers ☑ Vectors ☑ Polygons ☑ Text ☐ Mesh

Analysis Load Step 1 · Sub Step 0 · Crack Front 2 · Crack Growth Step 0

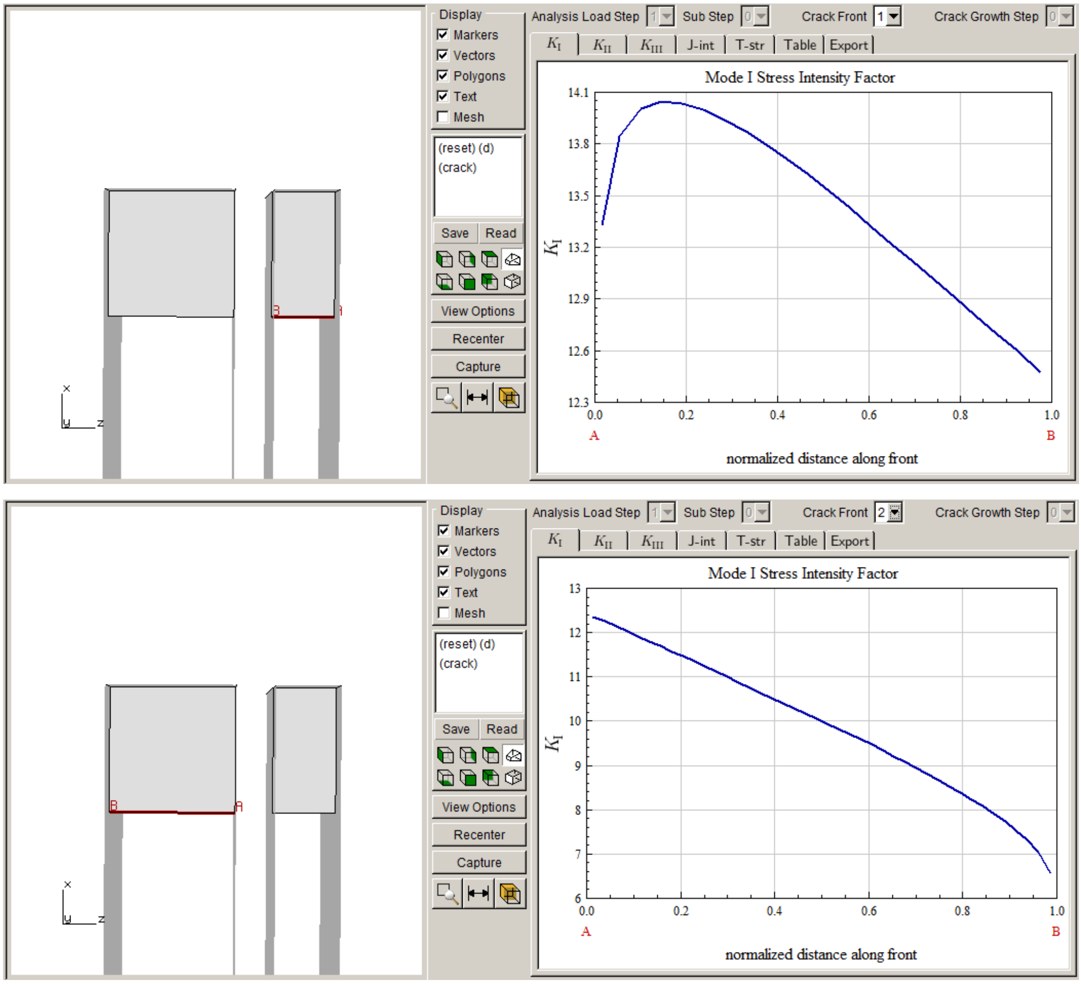
(reset) (d)
(crack)

Figure 15.5 Separate plate model SIFs – no interaction between plates.

The separate plate model is re-analyzed with *tie constraint added between the plates. The constraint is added using a Python script that moves the thinner plate over 3 units and adds the *tie using surfaces that were pre-defined in the original uncracked model. The Python script finds all elements with the second material and then finds all nodes belonging to those elements, and then modifies the node coordinates.

The Python script runs before the ABAQUS analysis. The Static Analysis dialog (see Section 8 of the Reference document) allows one to set the Python executable and the user-script. FRANC3D will run the Python script first using the cracked (*_full*) *.inp* file as input and output. Once the Python script is finished, ABAQUS will be started as usual.

The resulting Mode I SIFs are shown in Fig 15.6. FRANC3D still shows the plates in their original configuration. The SIFs are quite different from those in Fig 15.5 (as expected).



Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 0 | Crack Front 1 | Crack Growth Step 0

Display
☑ Markers
☑ Vectors
☑ Polygons
☑ Text
☐ Mesh

(reset) (d)
(crack)

Analysis Load Step 1 | Sub Step 0 | Crack Front 2 | Crack Growth Step 0

Figure 15.6 Separate plate model SIFs – tie constraint between plates.

15.3.2 Glued Plates

The glued plate model has a single crack inserted. Due to the material interface, FRANC3D Version 8 is used to insert the crack. The crack has the same length as the separate cracks above. The Mode I SIFs are shown in Fig 15.7. FRANC3D shows the plates in the actual glued configuration. Each material has a separate crack front. The SIFs are similar to those in Fig 15.6; the numbers differ slightly, which could be due to differences in the mesh surrounding the crack front template or due to the constraint versus shared interface or both.

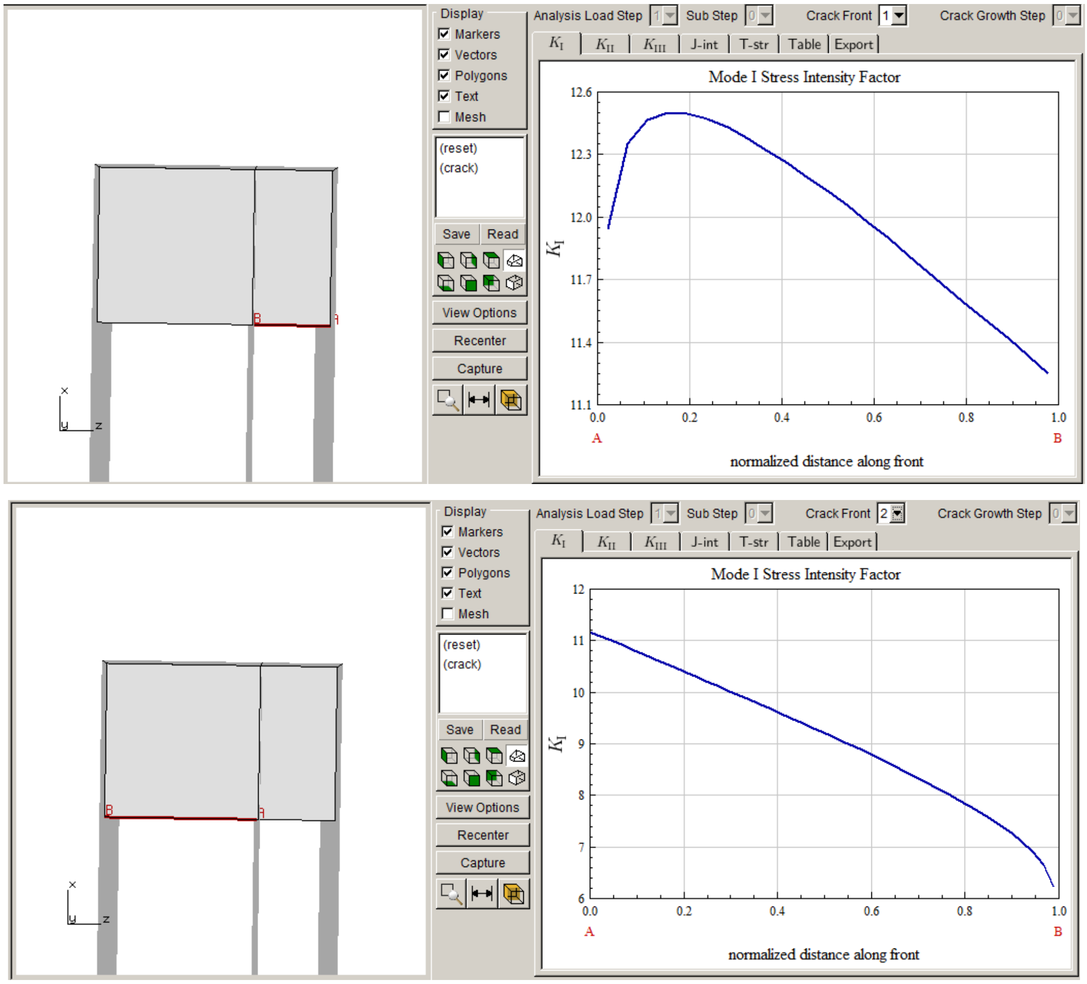


Figure 15.7 Glued plate model SIFs – tie constraint between plates.

15.3.3 Deformed Configurations

The deformed shapes for all three models are shown in Fig 15.8-10. The constrained-separate plates deform like the glued-plates; the maximum displacement (U, Magnitude) of the constrained-separate plates is slightly higher, which corresponds with the difference in SIFs.

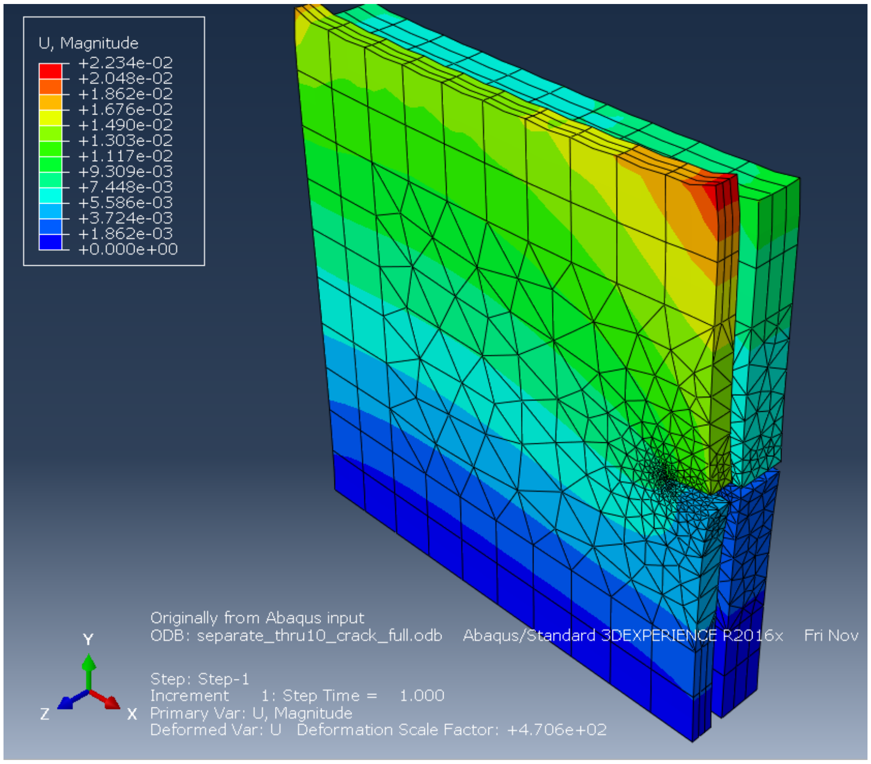

U, Magnitude
+2.234e-02
+2.048e-02
+1.862e-02
+1.676e-02
+1.490e-02
+1.303e-02
+1.117e-02
+9.309e-03
+7.448e-03
+5.586e-03
+3.724e-03
+1.862e-03
+0.000e+00
Originally from Abaqus input
ODB: separate_thru10_crack_full.odb   Abaqus/Standard 3DEXPERIENCE R2016x   Fri Nov
Step: Step-1
Increment   1: Step Time =   1.000
Primary Var: U, Magnitude
Deformed Var: U   Deformation Scale Factor: +4.706e+02
Y
Z
X

Figure 15.8 Separate plates with through-crack - deformed shape.

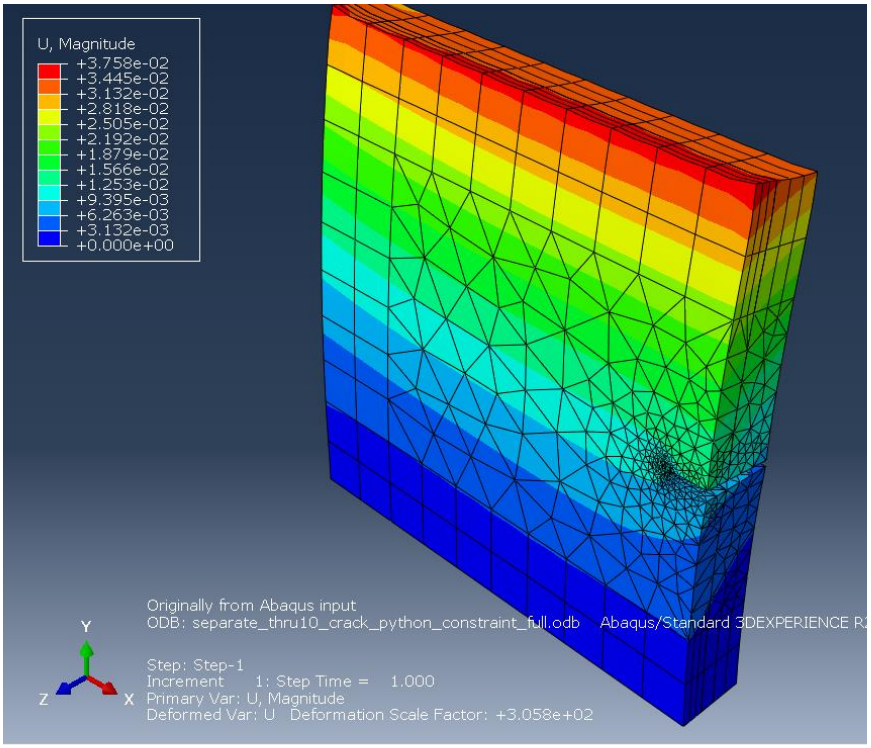

U, Magnitude
+3.758e-02
+3.445e-02
+3.132e-02
+2.818e-02
+2.505e-02
+2.192e-02
+1.879e-02
+1.566e-02
+1.253e-02
+9.395e-03
+6.263e-03
+3.132e-03
+0.000e+00
Originally from Abaqus input
ODB: separate_thru10_crack_python_constraint_full.odb   Abaqus/Standard 3DEXPERIENCE R
Step: Step-1
Increment   1: Step Time =   1.000
Primary Var: U, Magnitude
Deformed Var: U   Deformation Scale Factor: +3.058e+02
Y
Z
X

Figure 15.9 Constrained separate plates with through-crack - deformed shape.

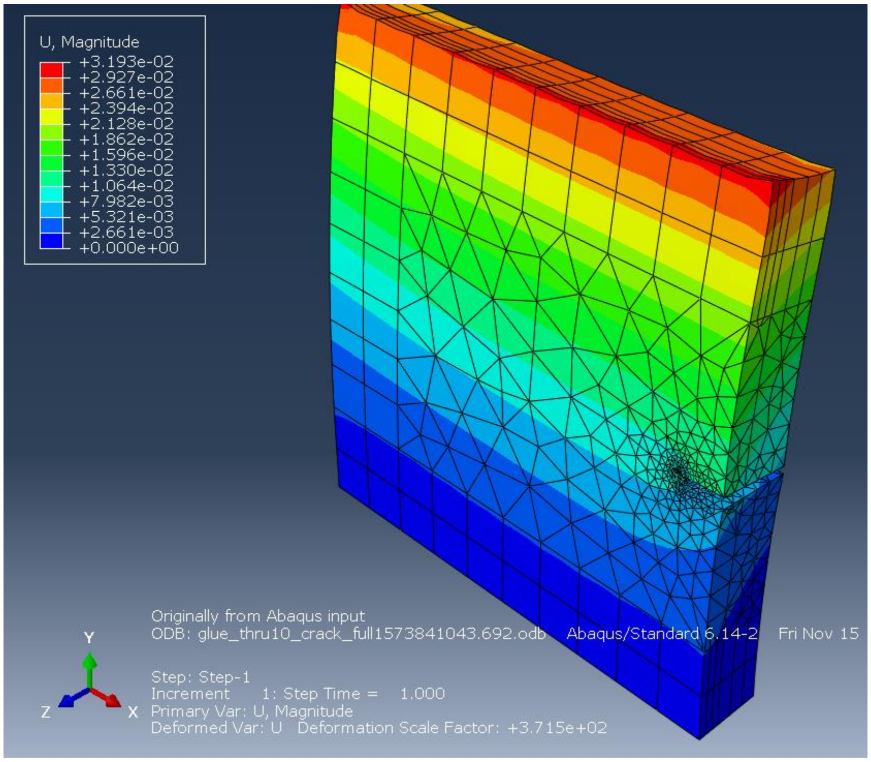

U, Magnitude
+3.193e-02
+2.927e-02
+2.661e-02
+2.394e-02
+2.128e-02
+1.862e-02
+1.596e-02
+1.330e-02
+1.064e-02
+7.982e-03
+5.321e-03
+2.661e-03
+0.000e+00
Originally from Abaqus input
ODB: glue_thru10_crack_full1573841043.692.odb Abaqus/Standard 6.14-2 Fri Nov 15
Step: Step-1
Increment 1: Step Time = 1.000
Primary Var: U, Magnitude
Deformed Var: U Deformation Scale Factor: +3.715e+02
Y
Z
X

Figure 15.10 Glued plates with through-crack - deformed shape.

16. Reducing Analysis Time

This chapter describes ways to reduce the crack growth simulation time, without reducing the number of crack growth steps (or increasing the crack growth increment).

16.1 Use Smaller Local Submodel

One of the simplest ways to reduce the simulation time is to use a smaller local submodel. During a crack growth simulation, new surface geometry is added to the crack geometry, and the crack geometry is inserted into the uncracked original FE model for each step. Crack insertion involves geometric intersection computations followed by surface and volume meshing. Having a small local submodel reduces the time for all operations.

If the crack outgrows the local submodel, a larger local model can be extracted, and the crack growth can continue – see Section 2.10 in the first base tutorial.

16.1.1 Multiple Portions

If you have multiple cracks, you can create multiple pieces for the local model – see Section 5.9 in the tutorial document. This describes how to extract two pieces for two cracks, but you can extract more than two pieces and insert more than one crack per piece if needed.

16.1.2 Reasonable Initial Crack Shape

When the crack geometry is inserted into the FE model, geometric computations are required to determine the intersection of crack surface and model surface so that only the crack surface inside the model is retained. This requires that the user create a crack geometry that exceeds the model geometry – see Section 5.9.2 of the tutorial for example. While the crack should be larger than the model, it will save time if the geometry is not excessive. For example, Fig 16.1 shows a reasonable crack along with an excessively large crack; all the crack geometry that falls outside of the model is simply discarded after determining that it is outside.

Fig 16.2 shows another case where an elliptical crack shape is chosen instead of the edge-through crack shape. To get a similar crack front inside the model – compared to that in Fig 16.1 – a relatively large ellipse might be required. In addition to the excess geometry that is discarded, the crack front template is formed along the entire crack front (right side panel in Fig 16.2), and all the excess template mesh must be discarded as well.

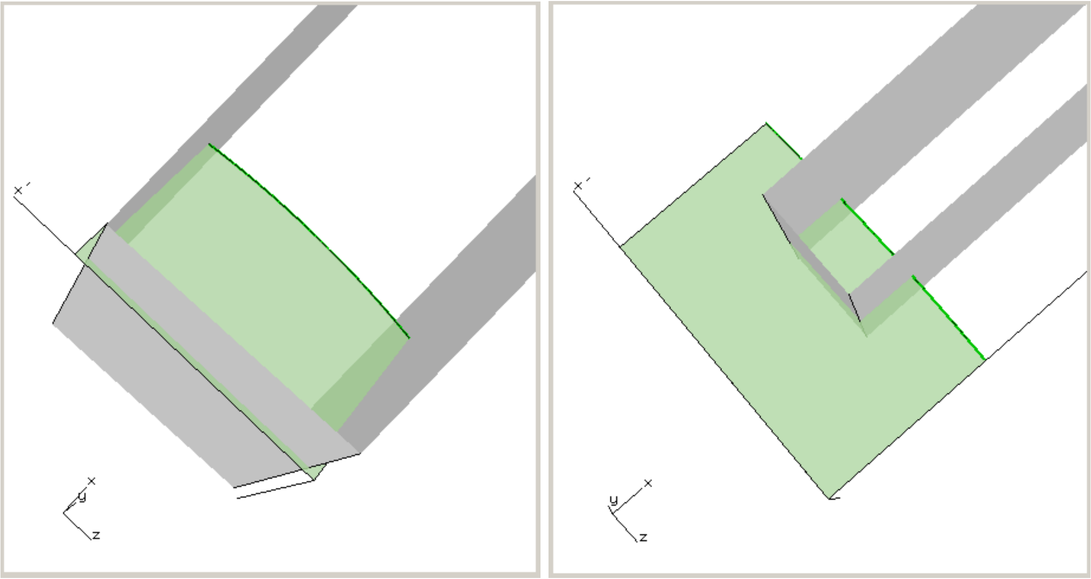


Figure 16.1 Use a reasonable amount (left panel) of crack geometry.

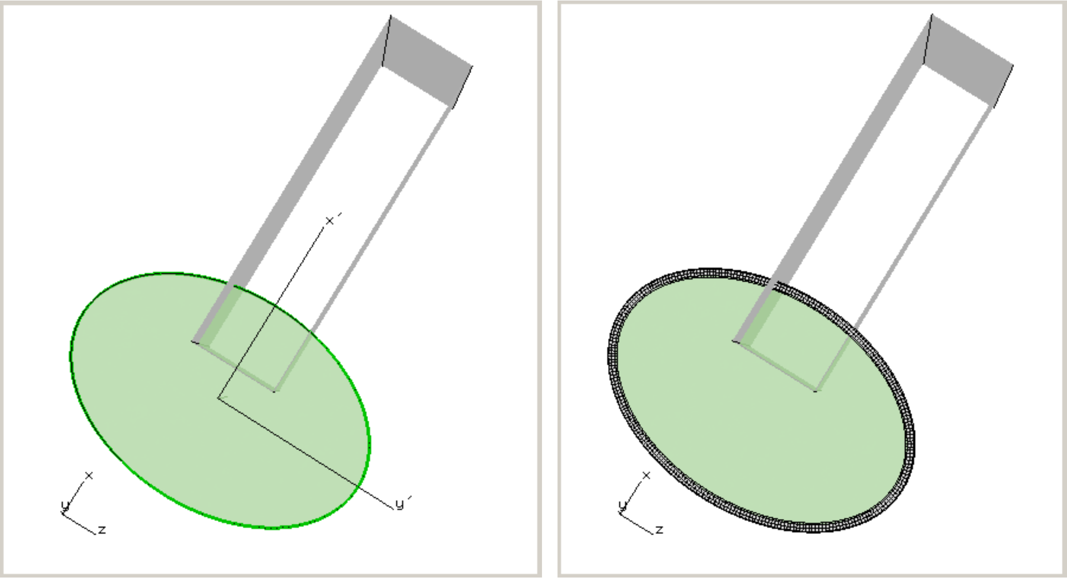


Figure 16.2 Poor choice of crack geometry for an edge crack.

16.2 Output Results for Template Nodes/Elements Only

The analysis results can be requested for the full model, the cracked local portion, or just the crack front template portion. The default is to extract results for the cracked local portion – see Section 8.1.3.1 of the Reference document for the ANSYS default settings (the same settings exist for ABAQUS and NASTRAN). If the local portion is large, with many nodes and elements, the output file can become quite large. To compute SIFs, FRANC3D only needs the results for the crack front template, so if the results files are large, one can reduce this by selecting results for the template only.

Note that if you select results for the template only, you will not be able to view the deformed shape correctly using the View Response dialog – see Section 11.1 of the Reference document. You will have to view the deformed shape of the model using the analysis code post-processing tools instead.

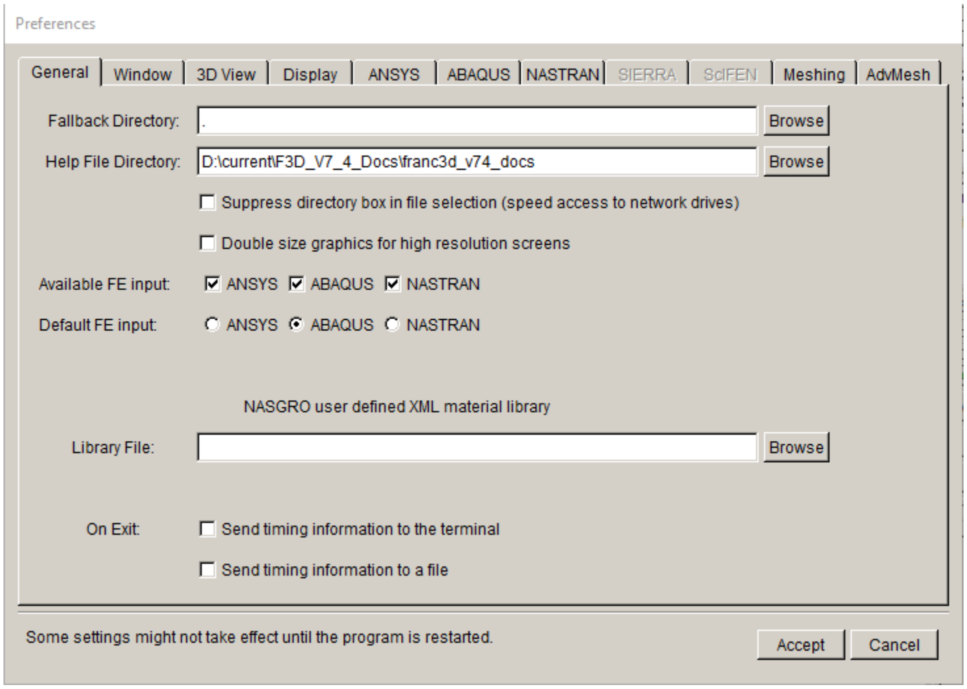
16.3 Reduce Time Points/Load Steps

Users might have dozens or hundreds of time points or load steps in their analysis. While FRANC3D can process this data, it can require significant amounts of time and storage. The results for each load step are used to compute SIFs, where the results often include displacement and temperature (and possibly crack surface pressure or traction) for an LEFM analysis. And for an elasto-plastic J computation from ABAQUS results, FRANC3D also requires stress and strain-energy, which is obtained at element integration points.

In addition to requesting output for only the template elements as described in the previous section, one might consider carefully selecting the most important load steps for the crack growth simulation.

16.4 Timing Output

The user can output timing for some FRANC3D operations. The Preferences dialog under the **Edit** menu contains an option to write timing results to a file and/or to the terminal window. Note that if you run FRANC3D by ‘double clicking’ the icon, the CMD (terminal) window will disappear when you exit FRANC3D, so you should send the information to a file instead. A “timing.txt” file will be saved in the working folder. It lists the operations and the corresponding wall clock time.



Fallback Directory: .

Help File Directory: D:\current\F3D_V7_4_Docs\franc3d_v74_docs

☐ Suppress directory box in file selection (speed access to network drives)

☐ Double size graphics for high resolution screens

Available FE input: ☑ ANSYS ☑ ABAQUS ☑ NASTRAN

Default FE input: ○ ANSYS ● ABAQUS ○ NASTRAN

NASGRO user defined XML material library

Library File:

On Exit: ☐ Send timing information to the terminal

☐ Send timing information to a file

Some settings might not take effect until the program is restarted.

Figure 16.3 Preferences dialog with timing output option.

17. FRANC3D Error Messages and Possible Solutions

This chapter describes potential errors that a user might encounter along with potential solutions. Error messages might be displayed as pop-up messages in the FRANC3D GUI; there might be additional information displayed in the FRANC3D CMD window so a user should check that window as well.

17.1 Generic File and Folder Access Errors

BAD_FILENAME

The program cannot open a given file. Either the file does not exist in the specified location, or the user does not have permission to access the file.

BAD_DIRECTORY

The program cannot open a given folder (directory). Either the folder does not exist, or the user does not have permission to access the folder.

CANNOT_EXECUTE

The program cannot execute an external program. Either the external program does not exist in the specified location, or the user does not have permission to execute the program.

FILE_READ_ERR

The file exists but there is an error during reading. This is a generic error message for unspecified errors that are not explicitly captured.

F3D_RLM_LIC_ERROR

This error indicates that a FRANC3D license is not available. The user should determine whether they have a node-lock or floating license and ensure that the license is current and valid.

17.2 Input Model and FRANC3D Restart File Errors

FRANC3D works with ANSYS, ABAQUS and NASTRAN, and can import and export the ASCII input files for these programs. The file extensions are:

.fdb – FRANC3D restart file that stores crack geometry, growth model, SIF history, and references to other files described below.

.cdb – ANSYS FE model file. The *.dat* file from ANSYS WorkBench is supported; see Appendix A of the FRANC3D/ANSYS Tutorial.

.inp – ABAQUS FE model file.

.bdf (*.dat*, *.nas*) – NASTRAN FE model file.

.dtp – results file with displacements, temperatures and contact pressures. This file is created for ANSYS results using ANSYS macro commands and for ABAQUS using Python commands. For NASTRAN, the .pch results file is imported into FRANC3D.

.crk – crack geometry. The geometry is a collection of Bezier patches, with the crack front identified by the corner vertex IDs of the patches.

.log – GUI session log file that contains FRANC3D commands.

.fcg – fatigue crack growth data (SIF history and crack growth model).

STEP###.* – automatically named and numbered crack growth step files.

<u>BAD_MODEL_TYPE</u>

The user is attempting to import an unsupported type of input FE model file.

<u>BAD_RESULTS_TYPE</u>

The user is attempting to import an unsupported type of FE results file.

<u>NO_MESH_DATA</u>

This error indicates that there are no elements or nodes defined in the imported model. Check the input FE file to ensure it is not corrupted or missing data. FRANC3D imports 3D volume elements, and these can be bricks, wedges, pyramids or tetrahedral or a combination.

<u>NO_DISP_DATA</u>

This error indicates that there are no displacements results. FRANC3D requires nodal displacements at all crack front template nodes to compute SIFs. Results must exist for all load steps. Check the FE results file to ensure that displacement data exists; the *.fdb* file contains a list of all template node IDs.

<u>NO_TRACT_DATA</u>

This error indicates that there is no crack face traction (CFT). This error might occur for mesh-based CFTs if the input mesh or stress files are invalid.

<u>FDB_FORMAT_ERR</u>

The *.fdb* file has a specific format, as described in Section 4.1.1 FRANC3D Restart (*.fdb*) Files. This error message is displayed when reading the *.fdb* file if a data block is formatted incorrectly. This could indicate a bug in the write/read routine or a corrupted file.

FDB_VERSION_ERR

The *.fdb* file contains version numbers for each block of data. If the version number does not match existing versions, when reading the *.fdb* file, this error will be displayed. This could indicate a bug in the write/read routine or a corrupted file.

FDB_EMPTY_ERR

This error message indicates that the *.fdb* file is empty or invalid. Check the file to ensure that it has appropriate data.

ANSYS_OUT_ERROR

This error message indicates that there is an error attempting to run ANSYS. This can be due to an incorrect ANSYS license string, a lack of ANSYS licenses, or other ANSYS related errors. The user should look for ANSYS related .err and .log files as well as the .out file, when the command (in generic form): ansys.exe –b –p lic_str –i in.cdb –o out.out
is executed.

ABAQUS_DAT_ERROR

This error message indicates that there is an error attempting to run ABAQUS. This can be due to a lack of license tokens or other ABAQUS related errors. The user should examine the ABAQUS related *.dat* and *.msg* files for errors.

NASTRAN_LOG_ERROR

This error message indicates that there is an error attempting to run NASTRAN. This can be due to a lack of NASTRAN licenses or other NASTRAN related errors. The user should examine the NASTRAN related *.log* file for errors.

BAD_”…”_STRING

A collection of error messages exists for the command line and Python script interface. If the data is incorrect, a BAD_”type”_STRING message is printed. The user should examine session logs and consult the Commands & Python reference to determine the correct data and format.

17.3 Flaw/Crack Insertion Errors

During crack insertion, a user might encounter errors. Typically, these will be related to geometric intersection issues as the crack geometry is inserted and connected to the model geometry or meshing issues after the geometric intersection. Surface and volume meshing errors might be encountered. In some cases, a user can by-pass the error with minor changes to settings. However, if needed a user can send us the *debug.tst* file that is written at the beginning of the crack insertion process; from this file, we can determine the problem and find a solution.

<u>FLAW_INSERT_ERR</u>

This is a generic error message for any errors not explicitly captured with error messages described below. Check the FRANC3D CMD window for any additional messages. Send us the *debug.tst* file if possible.

<u>FLAW_IN_RETAINED_ERR</u>

This error message indicates that the user is attempting to insert (or grow) a crack into a surface where mesh facets have been retained. The auto-cut-surface mesh facets are retained by default when using a local+global model. Surfaces with boundary conditions and named node-sets or surfaces can be retained also. Retaining surface mesh facets aids in merging the local-cracked model with the global and aids in transferring boundary conditions.

From Section 4.5.1: Note that cracks cannot be inserted into or propagated into a surface mesh that has been retained. If a surface has boundary conditions and a crack must be inserted into this surface, do not retain the mesh facets on this surface.

<u>INVALID_FLAW_ERR</u>

This error is a generic error for cases of invalid flaw/crack data not handled by more specific error messages below.

<u>PARAM_MISSMATCH</u>

This error message indicates a mismatch between provided and expected flaw/crack parameters. For example, an elliptical flaw requires the dimension for major and minor axes; if only one dimension is provided that would cause an error. This type of error is most likely to occur if a user has created their own command or Python script.

<u>BAD_FLAW_TYPE</u>

This error message indicates that the specified flaw/crack type is not supported. For example, Section 2.1.21 of the Commands reference lists the type of cracks. This type of error is most likely to occur if a user has created their own command or Python script.

<u>BAD_ROTATION_DATA</u>

This error message indicates that the specified flaw/crack rotation data is invalid. The rotation is defined using three axes numbered 1, 2 and 3 with rotation magnitudes about each axis. This type of error is most likely to occur if a user has created their own command or Python script.

SURF_MESH_ERR

Surface meshing errors can happen. One potential error occurs due to a “dangling” geometry edge on a surface. FRANC3D rebuilds geometry from the volume mesh. Depending on the surface, bounding edges might not be complete, which might lead to surface mesh errors. A user can try changing the kink angle setting, using the dialog described in Section 12.1 Edges Wizard, to eliminate dangling edges.

Another source of error is the crack and model surface intersection. Changing the template radius or turning on Simple Intersections, as described in Section 6.1.14 Crack Front Mesh Template Panel, might solve the problem.

Send us the *debug.tst* file if needed and we can provide help with this error.

VOL_MESH_ERR

This error message is a generic volume meshing error for cases not specifically handled below. Send us the *debug.tst* file and we can provide help with this error.

MAX_ELEMS_ERR

There is a setting for the maximum number of volume elements (see Section 6.1.14.1 Meshing Parameters Dialog). When using the FRANC3D volume meshing option, if this number is exceeded, the program stops and displays this message. A user can increase the maximum value or try switching to ANSYS or ABAQUS volume meshing. Another option is to increase the template radius, but this is not always possible.

MAX_RESTART_ERR

There is a setting for the maximum number of volume meshing restarts (see Section 6.1.15.1 of the Reference document). When using the FRANC3D volume meshing option, if the meshing gets stuck, it restarts with a different starting point. In most cases, if the program exceeds the default 4 restarts, then FRANC3D likely will not be able to mesh the volume.
In this case, ANSYS or ABAQUS volume meshing can be attempted. Another option is to change the template radius or turn on simple intersections. In some cases, it is regions away from the crack that cause issues, Fig 17.1.

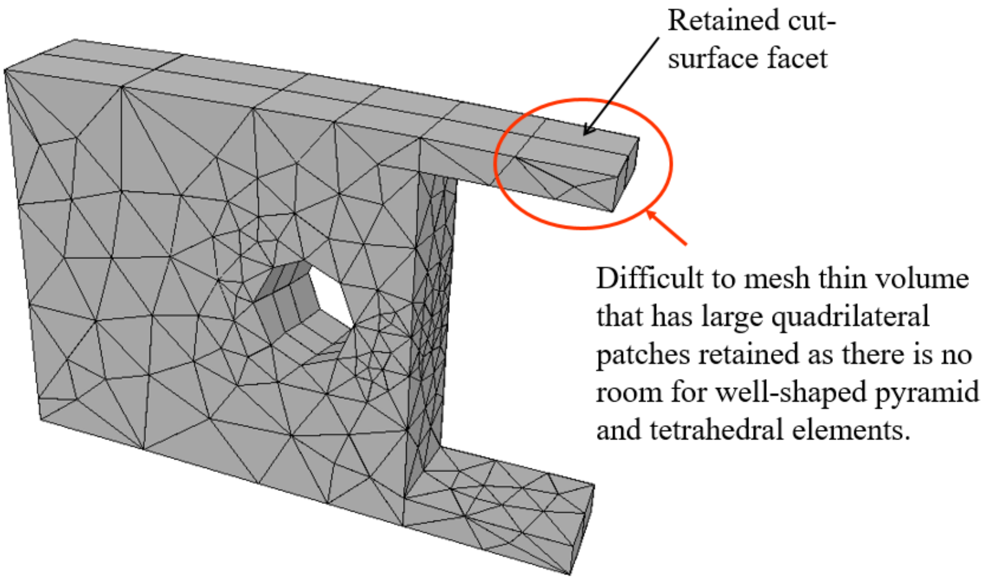

Retained cut-surface facet
Difficult to mesh thin volume that has large quadrilateral patches retained as there is no room for well-shaped pyramid and tetrahedral elements.

Figure 17.1 Volume meshing errors.

PYRAMID_ERR

This error message occurs when pyramid elements cannot be added to the volume. Pyramid elements are created in FRANC3D where quadrilateral mesh facets exist; this includes the crack front template as well as cut-surfaces as shown in Fig 17.1. If there is not sufficient room to add a pyramid, the program will stop meshing. Solutions will depend on where the problem mesh is located.

For cases like that shown in Fig 17.1, the user can change the local volume or not retain the mesh facet.

For cases where the problem is at the crack front template, reducing the template radius and/or turning on Simple Intersections might help.

FRANC3D currently does not indicate where the problem is located, so if you need help with this error, send us the *debug.tst* file.

VOID_MESH_ERR

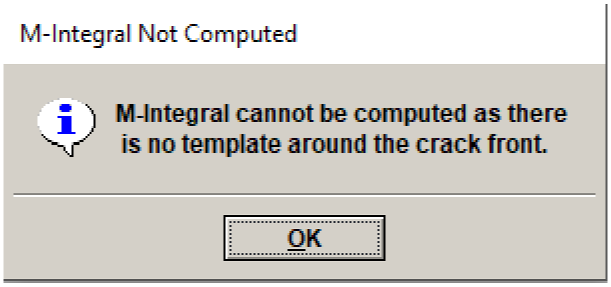
This error message is a generic volume meshing error for finite volume voids (not cracks).

VOL_INSERT_ERR

This error message is a generic volume meshing error for situations where the user has chosen ANSYS or ABAQUS volume meshing and the resulting volume mesh is empty or invalid. The user can examine the ANSYS or ABAQUS specific files to look for error messages. FRANC3D writes a surface mesh in *.cdb* or *.inp* format and then executes ANSYS or ABAQUS to read this input and create a volume mesh. These programs sometimes fail. In these cases, changing the crack front template radius or turning on Simple Intersections might help. If you need help with this error, send us the *debug.tst* file; we cannot make ANSYS or ABAQUS volume meshing work, but we can usually find the problem and provide solutions.

17.4 SIF Computation and Crack Growth Errors

There are several reasons why SIFs or M-integral SIFs cannot be computed. The errors are listed below. In FRANC3D, the user will see a message displayed, such as this, Fig 17.2:


M-Integral Not Computed

M-Integral cannot be computed as there is no template around the crack front.

Figure 17.2 M-integral not computed warning.

SIF_COMP_ERR

This is a generic error message when computing SIFs for cases not explicitly handled by errors described below.

SIF_NO_MESH

This error message indicates that SIFs cannot be computed as there is no mesh data. It is not likely one will encounter this error.

SIF_NO_DISP

This error message indicates that SIFs cannot be computed as there is no displacement data. FRANC3D reads results from *.dtp* or *.pch* files and uses this data to compute SIFs. Displacements are required for all crack front template nodes for all load steps. See Section B.2 for additional information. A debug_sif.txt might be created that will list the node that is missing data.

SIF_NO_TEMPLATE

This error message indicates that SIFs cannot be computed as there is no crack front template mesh. The M-integral requires a template mesh. Displacement correlation can be used to compute SIFs if a template mesh is not present.

SIF_WRONG_CIRC

This error message indicates that SIFs cannot be computed as there is an incorrect number of elements around the crack front in the template mesh. The M-integral requires a template mesh with an even number of elements around the front. Displacement correlation can be used to compute SIFs in this case.

SIF_NO_MAT

This error message indicates that SIFs cannot be computed as there is no material data. Material data comes from the original input FE model.

SIF_NO_TEMP

This error message indicates that SIFs cannot be computed as there is no temperature data. If thermal terms are turned on for the M-integral computation, nodal temperatures must exist for all template nodes for all load steps.

PROP_COMP_ERR

This is a generic error message when growing the crack for cases not explicitly handled by errors described below.

PROP_READ_ERR

This is a generic error message when reading the crack growth from a file.

NO_CRACK_GROWTH

This is a generic error message when growing the crack if there is no crack growth.

UNSTABLE_GROWTH

This error message indicates that the compute SIFs exceed the user-specified critical toughness. Crack growth is not performed in this case. The user can modify the critical value if growth is desired.

USER_MODEL_ERR

This error message is displayed if one is using Python extensions and an error occurs. Currently, it is up to the user to determine the cause of this error.

EXT_NEED_DATA

This is a generic message indicating that data required for crack extension has not been provided.

EXT_NEED_RES_TEMPS

This message indicates that temperatures are needed for crack extension.

EXT_NEED_SCH_TEMPS

This message indicates that the load schedule requires a temperature setting.

EXT_NEED_YIELD_STRESS

This message indicates that the crack growth model requires a value for yield stress.

USER_EXT_ERR

This error message is displayed if one is using Python extensions and an error occurs. Currently, it is up to the user to determine the cause of this error.

NO_SIF_DATA

This error message is displayed if the user is attempting to export SIF data to a file when there is no SIF data. This error is most likely to occur from a command or Python file.

NO_COD_DATA

This error message is displayed if the user is attempting to export COD (crack opening displacement) data to a file when there is no COD data. This error is most likely to occur from a command or Python file.

NO_SIFPATH_DATA

This error message is displayed if the user is attempting to export SIF path data to a file when there is no SIF path data. This error is most likely to occur from a command or Python file.

NO_LIFE_DATA

This error message is displayed if the user is attempting to export fatigue life data to a file when there is no fatigue life data. This error is most likely to occur from a command or Python file.

17.5 Analysis Code Errors

When executing the FE analysis codes (ANSYS, ABAQUS, NASTRAN) from FRANC3D, a user might encounter errors associated with: 1) the input files created by FRANC3D, 2) the command line created by FRANC3D, or 3) accessing the analysis code executable (or its license).

17.5.1 Input FE File

Errors in the input file might occur if data is not yet supported by FRANC3D. The FE analysis code will create error and/or log files that can be opened by the user to help find the error. ANSYS creates *.err* and *.log* files. ABAQUS creates *.dat* and *.msg* files. NASTRAN creates *.log* and *.out* files. FRANC3D looks for these files and searches for error messages inside the files. If errors are listed, FRANC3D should display a warning message. The user should examine the files to determine the cause of the error.

17.5.2 Analysis Command Line

FRANC3D generates a command to execute the FE analysis code in the background (and saves the command to a *.txt* file for cases where the user might need to run the command manually). The command line is created from user-specified entries – most of which are in the Preferences tabs.

17.5.3 Analysis Executable or License Access

FRANC3D allows the user to specify the executable file for the FE analysis code. Typically, this is done once in the Preferences. For ANSYS, the license string should be specified also as this is required for the Command Line. If FRANC3D cannot execute the analysis code, you should check the location (folder/path) and verify that you have a valid license available.