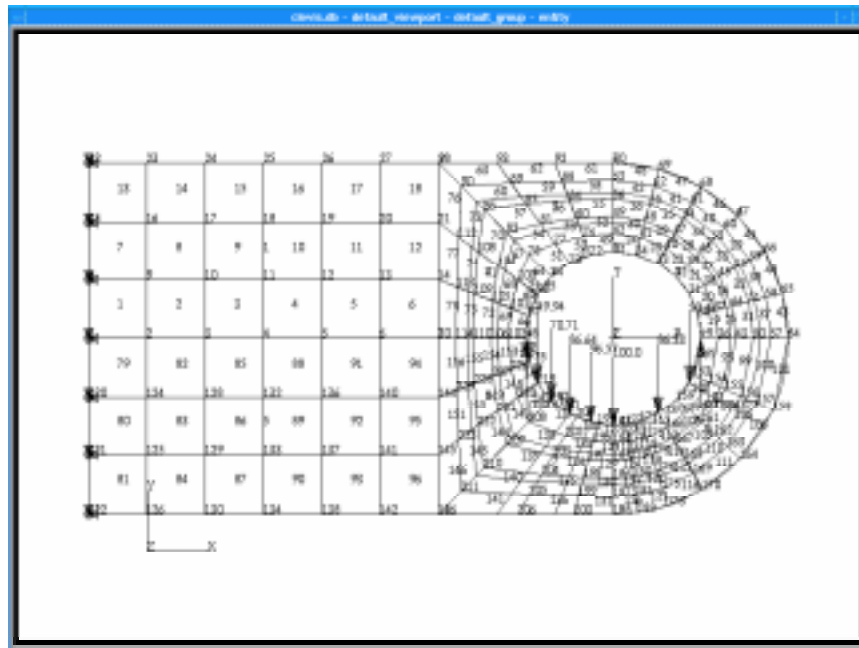


## Exercise 6

# *Load Lug Model*



### Objective:

- Write a function to apply the loads and element properties to the finite element mesh of the lug.



## Exercise Description:

This exercise, `lug_load ( )`, applies the loads and boundary conditions, material and element properties to the lug model. When complete, the model will have material properties, element properties, a fixed end, and a spatially varying distributed load in the lower half of the pin hole.

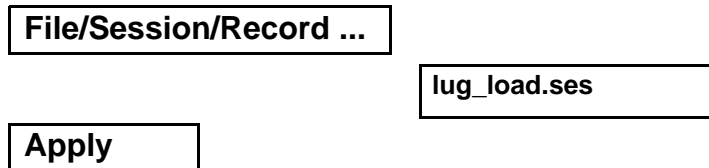
## Files:

All the files that are used in this exercise are listed below. Each list includes the file, where it originated, and a summary of information of how it relates to the exercise.

File	Supplied/Created	Description
lug_load.ses	Created	You will create this file while you are applying loads and boundary conditions to the model.
lug_load.pcl	Created	This will be created from the session file after it is made into a PCL function.

## Exercise Procedure:

1. Open the database that you have been using for the previous two exercises if it is not already open.
2. Start recording a new session file called **lug\_load.ses**.



3. Create the materials that you are going to put on the model.

### ◆ Materials

Action:



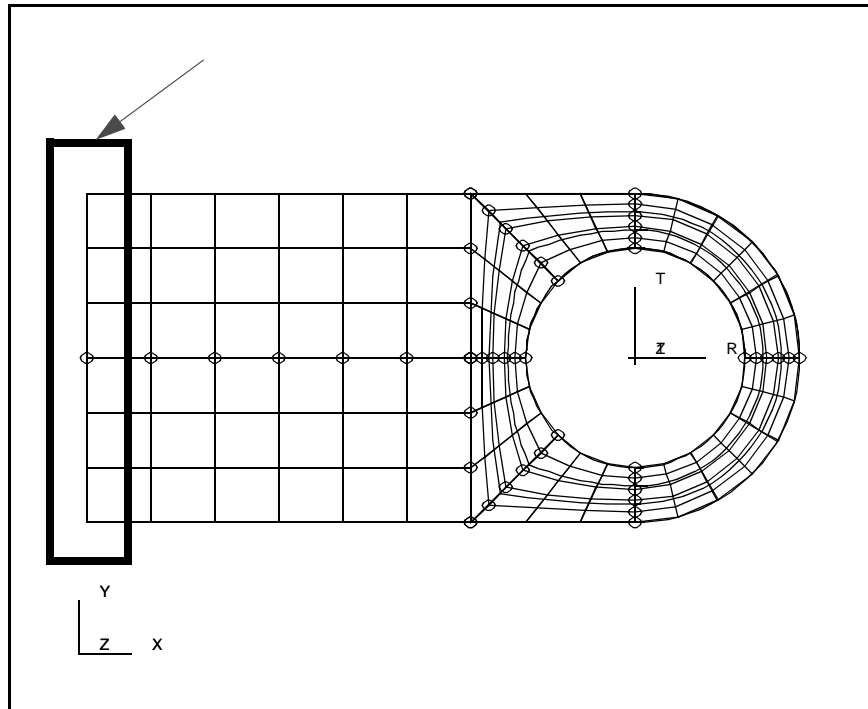
<i>Object:</i>	<b>Isotropic</b>
<i>Method:</i>	<b>Manual Input</b>
<i>Material Name:</i>	<b>steel</b>
<b>Input Properties...</b>	
<i>Elastic Modulus</i>	<b>30E6</b>
<i>Poisson Ratio</i>	<b>0.3</b>
<b>Apply</b>	
<b>Cancel</b>	

4. Create the properties for the steel material.

**◆ Properties**

<i>Action:</i>	<b>Create</b>
<i>Dimension:</i>	<b>2D</b>
<i>Type:</i>	<b>Shell</b>
<i>Property Set Name</i>	<b>eighth_in_steel_plate</b>
<b>Input Properties...</b>	
<i>Material Name</i>	<b>m:steel</b>
<i>Thickness</i>	<b>'thickness'</b>
<b>OK</b>	
<i>Select Members</i>	<b>Select all on screen</b>
<b>Add</b>	
<b>Apply</b>	

- Secure the left side of the lug. Make sure that entity selection is set to enclose entire entity.



◆ **Loads/BCs**

Action:

Create

Object:

Displacement

Type:

Nodal

New Set Name

solid\_wall

Input Data...

Translations

<0 0 0>

Analysis Coordinate Frame

Coord 0

OK

Select Application Region...

◆ **Geometry**



Curve or Edge Icon

---

*Select Geometry Entities*

See the graphic shown above

**Add**

**OK**

**Apply**

6. Create the fields on the model

◆ **Fields**

*Action:*

**Create**

*Object:*

**Spatial**

*Method:*

**PCL Function**

*Field Name*

**lug\_force**

*Field Type*

◆ **Vector**

*Coordinate System Type*

◆ **Real**

*Coordinate System*

**Coord 1**

*Vector Function ('R, 'T, 'Z)*

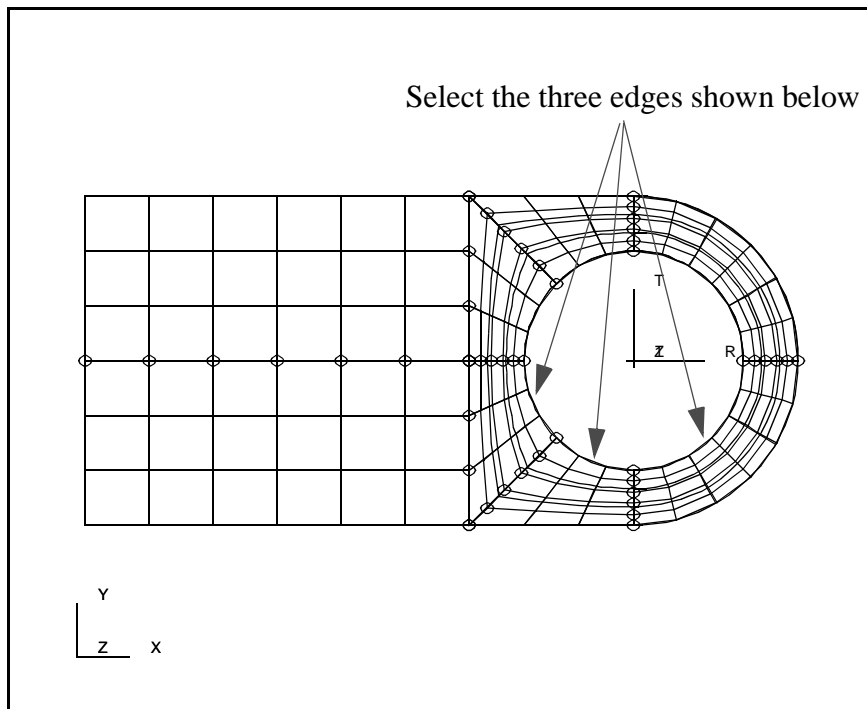
*Second Component*

**amplitude\*sinr( 'T)**

**Apply**

# Load Lug Model

7. Apply a force to the inner radius of the lug.



◆ **Loads/BCs**

Action:

Create

Object:

Force

Type:

Nodal

New Set Name

lug\_force

Input Data...

Spatial Fields

lug\_force

Force

f:lug\_force

Analysis Coordinate Frame

Coord 0

OK

Select Application Region...

◆ **Geometry**

Select Geometry Entities

See the graphic shown above

Add

---

**OK**

**Apply**

8. Stop recording the session file.

**File/Session/Record ...**

**lug\_load.ses**

**Stop**

**Cancel**

9. Use a vi editor or jot to create a PCL function from the session file. Name the function **lug\_load()**. Make sure to end the function.

10. Rename the session file **lug\_load.pcl**.

11. Instead of deleting all the Loads/BCs, Materials, Fields, and Properties close the database and quit MSC.Patran.

**File/Quit**

12. Create a new file call **p3epilog.pcl**. Enter the following into the file:

```
!!input lug_create.pcl
!!input lug_mesh.pcl
!!input lug_load.pcl
```

Make sure that the p3epilog.pcl file is in your current working directory.

13. Start MSC.Patran again by typing p3 in your xterm window, or patran in the dos command prompt.

14. Open a new database.

**File/New...**

**completed\_lug.db**

**OK**

Click **OK** when the *New Model Preferences Form* appears.

15. Enter the following command:

```
lug_create()
```

After committing this command the geometry for your lug should be created.

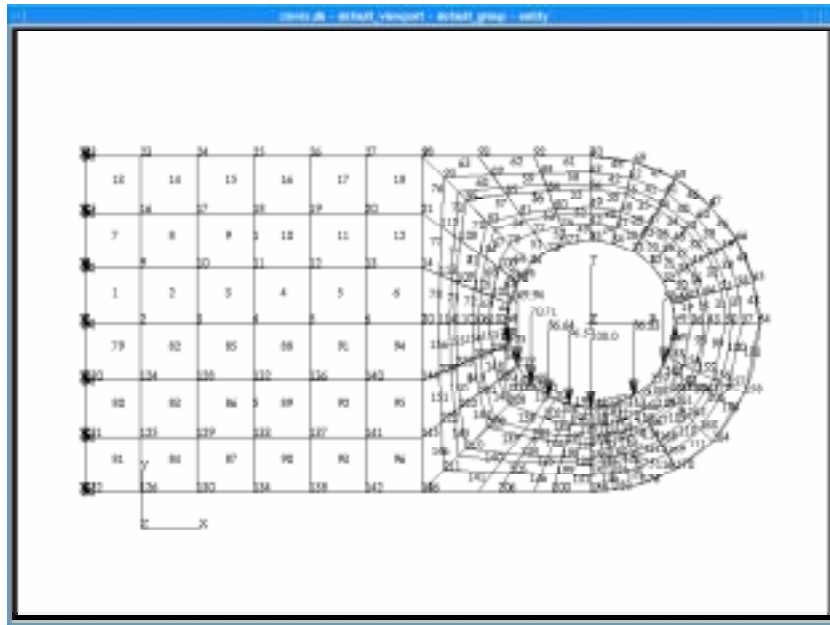
16. Enter the name of the function that should create the mesh seeds and mesh for the lug.

**lug\_mesh()**

17. After executing this last command the lug model should appear the same as when you created it by hand

**lug\_load()**

Your model should appear as shown.



18. Quit MSC.Patran.

**File/Quit**

---

## Sample Solution:

```
FUNCTION lug_load( )

/*
* Purpose:
*         Load a previously created 2-dimensional model of a lug with
*         parametric cubic patches.
*
* INPUT:
*         lug_geometry STRING[VIRTUAL] Picklist of the geometry contained
*                                     in the lug.
* OUTPUT:
*         none
*
* Side effects:
*         A 2D lug is loaded from the specified dimensions
*/

        STRING      lug_geometry[], surface_1[], surface_5[]
        STRING      surface_6[], surface_7[], surface_8[]
        STRING      temp[VIRTUAL](VIRTUAL)

material.create( "Analysis code ID", 1, "Analysis type ID", @
                1, "steel", 0, @
                "Date: 16-Mar-93 Time: 11:32:24", @
                "Isotropic", 1, @
                "Directionality", 1, "Linearity", 1, @
                "Homogeneous", 0, "Linear Elastic", 1,@
                "Model Options & IDs", ["","","",""], @
                [0,0,0,0,0], "Active Flag", 1, @
                "Create", 10, "External Flag", FALSE, @
                "Property IDs", ["Elastic Modulus",@
                "Poisson Ratio"], [2,5,0], "Property Values", @
                ["30e+6",".3",""] )

elementprops_create( "eighth_in_steel_plate", 57, 25, 19, 1, 1, @
                    20, [13,20,@
                    36,4037], [5,9,1,1], @
                    ["m:steel","","thickness"], @
                    lug_geometry )

loadsbc_create( "solid_wall", "Displacement", "Nodal", "", @
                "Static", temp, @
                "Geometry", "Coord 0", 1., @
                ["< 0 0 0 >","< >"], ["",""] )

fields_create( "lug_force", "Spatial", 1, "Vector", "Real", @
                "Coord 1", "", "Function", 1, "", "T", "", "", @
                "amplitude * sinr( 'T )", @
                "", FALSE, [0.], @
                [0.], [0.], [[0.]] )

sys_reallocate_string( temp, str_length( surface_6//".1 "//surface_7//@
                ".1 "//surface_8//".1 ")
sys_reallocate_array( temp, 1, 1 )
temp(1) = surface_6//".1 "//surface_7//".1 "//surface_8//".1 "
loadsbc_create( "lug_force", "Force", "Nodal", "", "Static", @
                temp, "Geometry", "Coord 0", 1., @
                ["f:lug_force","< >"], ["",""] )

END FUNCTION
```



