LS-INGRID

Examples Manual
for
Modeling and Mesh Generation

Version 1.0

August 1998

<u>Mailing Address:</u>
Livermore Software Technology Coporation
2876 Waverley Way
Livermore, California 94550-1740

<u>Support Address:</u>
Livermore Software Technology Corporation
97 Rickenbacker Circle
Livermore, California 94550-7612

FAX: 925-449-2507
TEL: 925-449-2500
EMAIL: sales@lstc.com
WEBSITE: www.lstc.com

TABLE OF CONTENTS

INTRODUCTION

This document is divided in two sections called MODELING AND MESH GENERATION and EXAMPLES FOR LS-DYNA.

MODELING AND MESH GENERATION explains the Index Progression and its usage combined with the Projection Method. It contains several Examples to show the capabilities of LS-INGRID.

EXAMPLES FOR LS-DYNA are included to demonstrate how to define Contact, Loads and Boundary Conditions, Outputs, etc.

LS-INGRID is started with the execution syntax

ingrid i={*filename*} o={*output filename*}{*graphics device*}

{*graphics device*} is, e.g. *gl (SGI GL graphics), sb (HP Starbase Graphics), x (X-Windows)*, etc. The default name for *{output filename}* is *ingrido* which is overwritten each time LS-INGRID is started. Therefore it is a good idea to define a unique filename.

The input file can be chosen interactively by starting LS-INGRID with the command

ingrid

and choosing proper the {*graphics device*}. At the beginning of the Data Input-Phase the input file is chosen with a click on the [*Input File*] button and the corresponding file. The Graphics- Phase of LS-INGRID is invoked either with the

grap(hics)

command or by selecting on the [*Graphics*] button from the menu. The menu selection or the keybord commands can be used to visualize the model and most of the definitions.

Some useful commands for visualizing graphics are

DRAW	draw your model with all lines
GOOD	hidden display
SHADE	shaded display
RESTORE	reset the position of your model
DI SD	display all surface definitions

The model is translated in the XY-plane by using the <shift> key and the middle mouse button, rotated by using <shift> key and the left mouse button and scaled by pressing <shift> and the right mouse button. For more details see the *LS-INGRID Graphics Interface User's Manual*.

BASICS IN MODELING AND MESH GENERATION

LS-INGRID uses Index Progression in Index Space to define a Block (Surface and Line) as a Region. This Region is modified by geometry manipulating commands such as Deletion, Projection, etc. to create a Part. If there are enough Parts then they can be connected to a Model together, i.e. Merge, define Contact, Loads and Boundary Conditions, etc.

Index Space and Index Progression

In general a Region in Index Space is defined with a set of Six Indices $(I_{MIN}\ J_{MIN}\ K_{MIN})$ and $(I_{MAX}\ J_{MAX}\ K_{MAX})$ called Index Progressions. The I, J and K-Axes in Index Space correspond to the X, Y and Z-Axes in the Cartesian Coordinate System.
To demonstrate the use of Index Progression consider the following example shown in Figure 1.

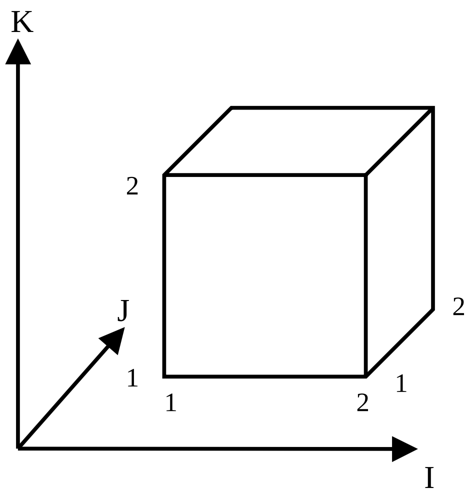


Figure 1. A single Solid defined with Index Progression

The Solid is defined by the following Index Progression: 1 2; 1 2; 1 2; (I_1 I_2; J_1 J_2; K_1 K_2;). This is due to the fact that there are two points in each direction and therefore 2-1=1 element in each direction. The main purpose of the Index Progression is to define the number of points for each Region of a Part. The definition always begins with a real number≠0 but does not necessarily have to be equal to 1. Figure 2 shows another simple example.

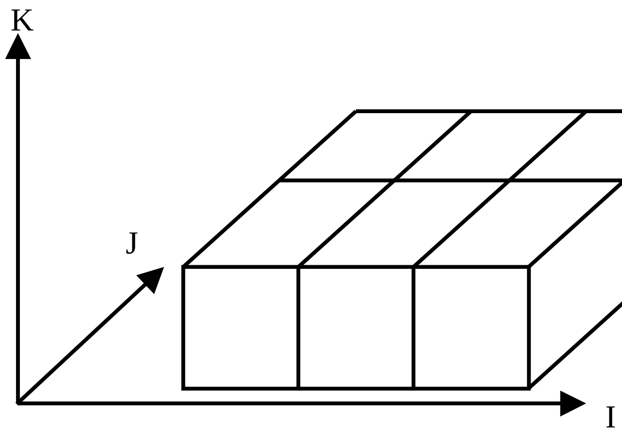


Figure 2. Solid with Index Progression 1 4; 1 3; 1 2;

In order to generate a finer mesh in a particular space of a Part it is necessary to define more Regions. Figure 3 shows such a Part.

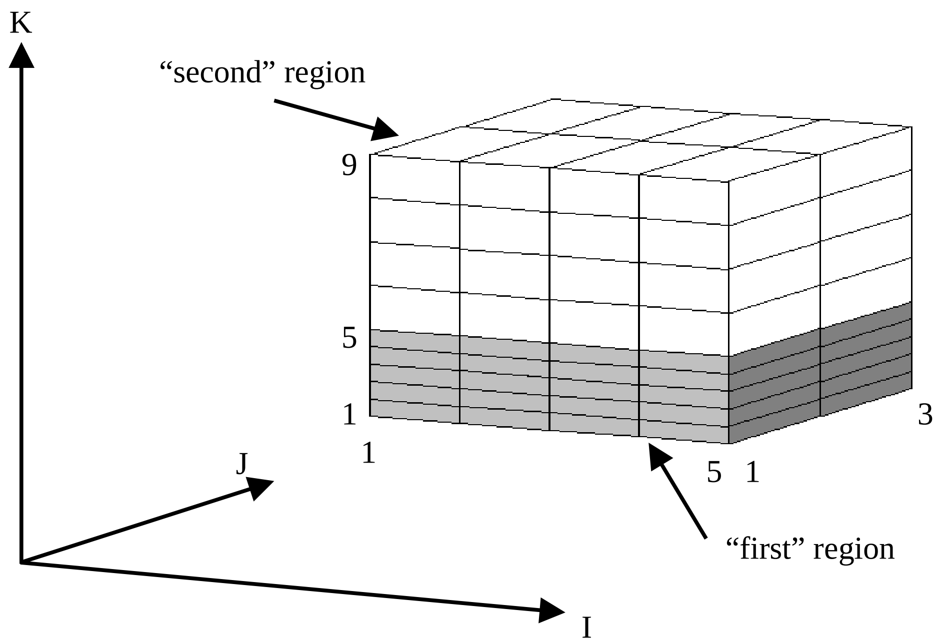
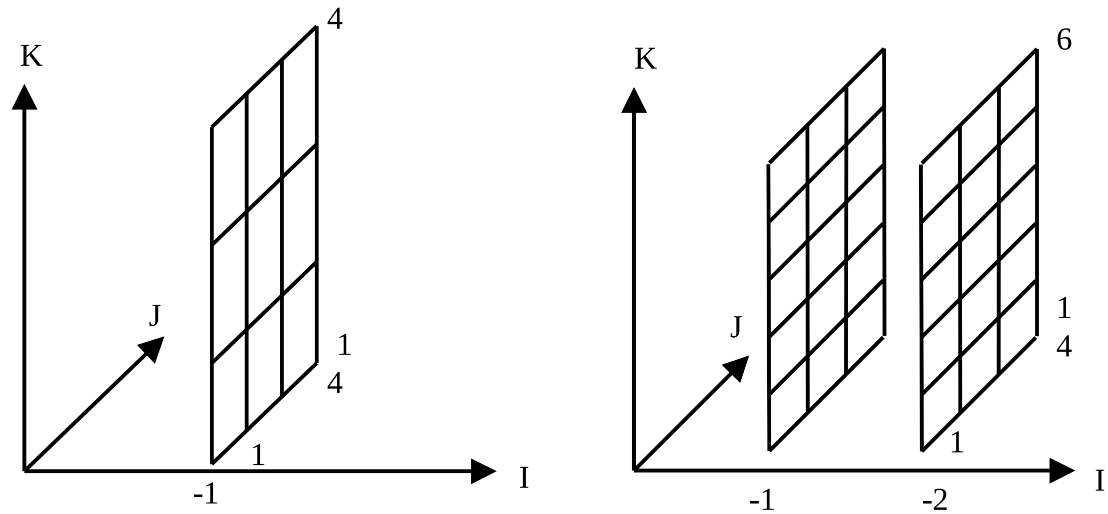


Figure 3. Part with 2 Regions

This Part has the Index Progression 1 5; 1 3; 1 5 9;. The “first” Region is 1 5; 1 3; 1 5;, the “second” is 1 5; 1 3; 5 9;.

To define a Plane using Index Progression assign a negative number in one direction.
-1; 1 4; 1 4; defines the Plane shown in Figure 4a and -1 –2; 1 4; 1 6; defines the Plane in Figure 4b.



(a) (b)

Figure 4. Planes in Index Progression

One can also define two separate Regions at one time. This is shown in Figure 5 where two Solids are generated as one Part.

Index Progression: 1 2 0 4 7; 1 4 ; 1 4;

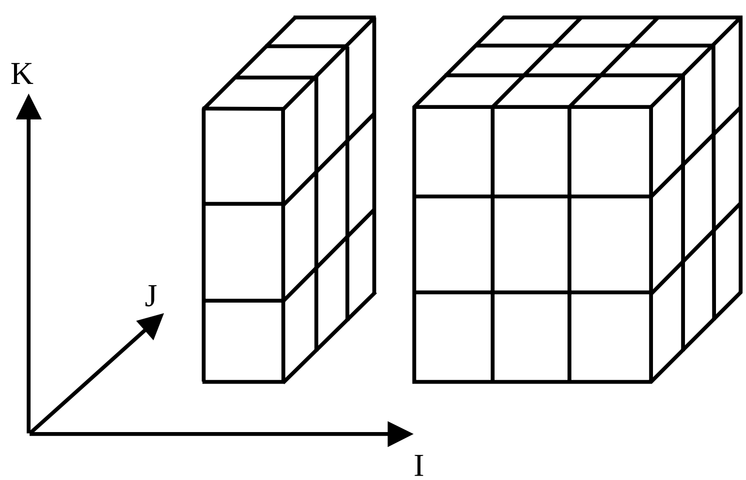


Figure 5. Two separated solids in one PART

The **ZERO** in the I-Direction indicates that the structure is **discontinuous**. This feature makes it possible to create complex structures.

It is possible to define a Line in Index Space by holding two directions constant and varying the third. An example of this option is shown in Figure 6a where I and K are constant and J is varied. This results in a Line parallel to the J-axis. Figure 6b shows another example of a Line definition.

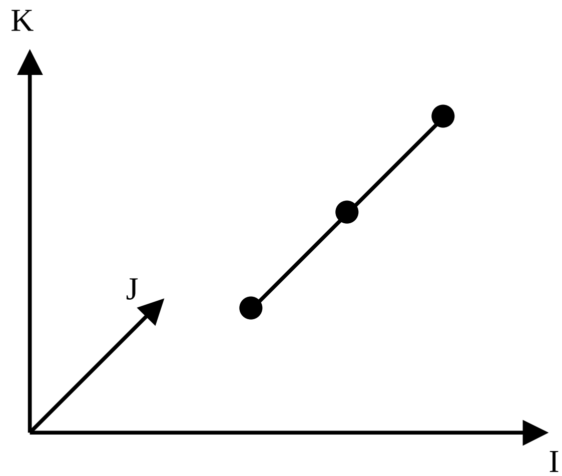


Figure 6a. Line in Index Space (1; 1 3; 1;)

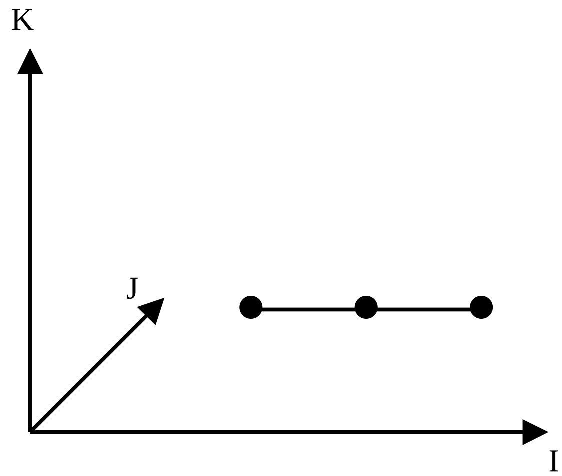


Figure 6b. Line parallel to the I-axis (1 2 3; 1; 1;)

Reduced Index Space

Once a Part is defined in Index Progression LS-INGRID creates a Reduced Index Space. For example, given the Index Progression 1 21 25 37; 15 22 27; 1 11 17; which generates the Solid in Figure 7a, LS-INGRID defines the Reduced Index Space as shown in Figure 7b. The Reduced Index is ***independent of the actual value*** of the Index Progression. By this it is meant that the following Index Progression 1 18 25 37; 15 22 27; 1 11 17; the same set of Reduced Indices is generated. The only difference would be the number of elements between successive Reduced Indices. It is used in all geometry manipulating and constraining commands such as Deletion, Surface Definition, Boundary Conditions, Loads, Contact, etc.

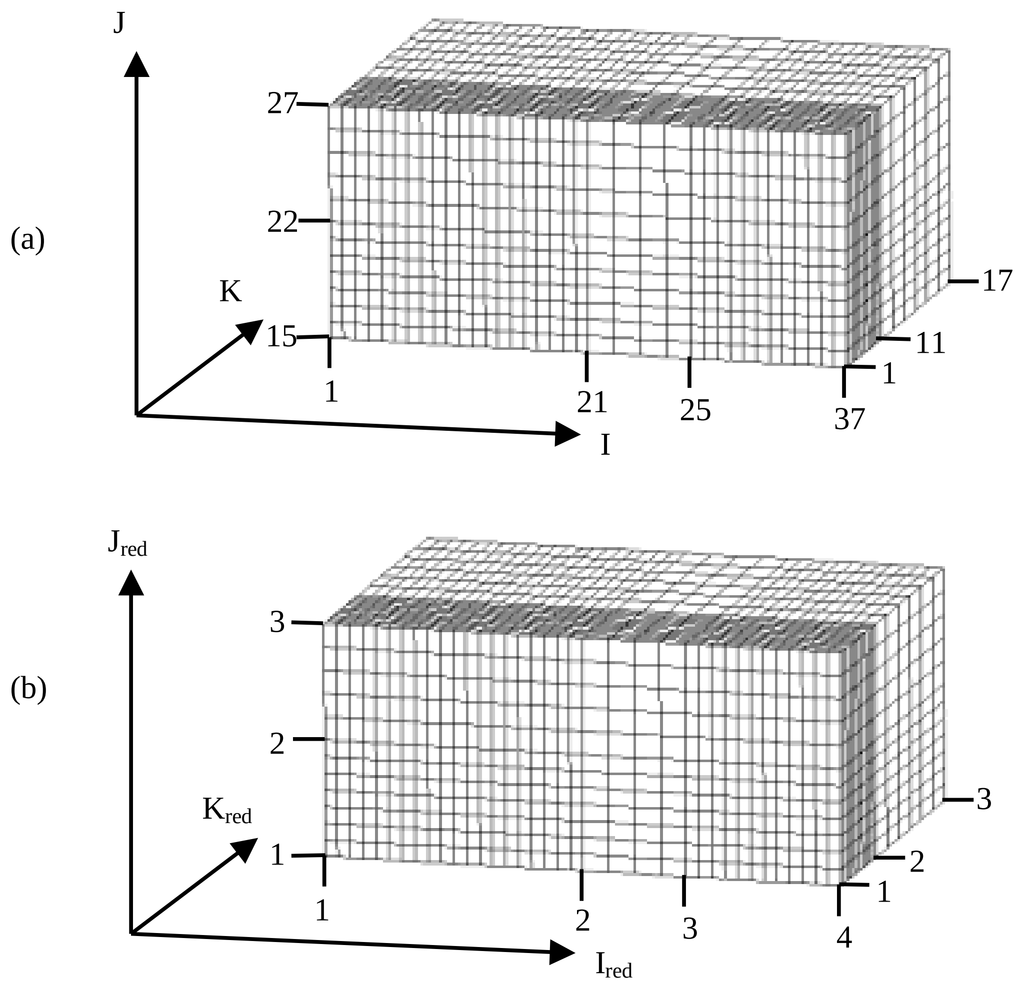


Figure 7. Index Space and Reduced Index Space

Deletion

The next example shows how to delete a Region generated by Index Progression. There are two commands available for Deletion.

D $I_1\ J_1\ K_1\ I_n\ J_n\ K_n$

DI $I_1\ I_2 \ldots I_n;\ J_1\ J_2 \ldots J_n;\ K_1\ K_2 \ldots K_n;$

where I_n, J_n *and* K_n *are in Reduced Index Space*

For example to create a Part in the shape of an uppercase "T", first build a block in Index Progression and the delete the two shaded regions shown in Figure 8.

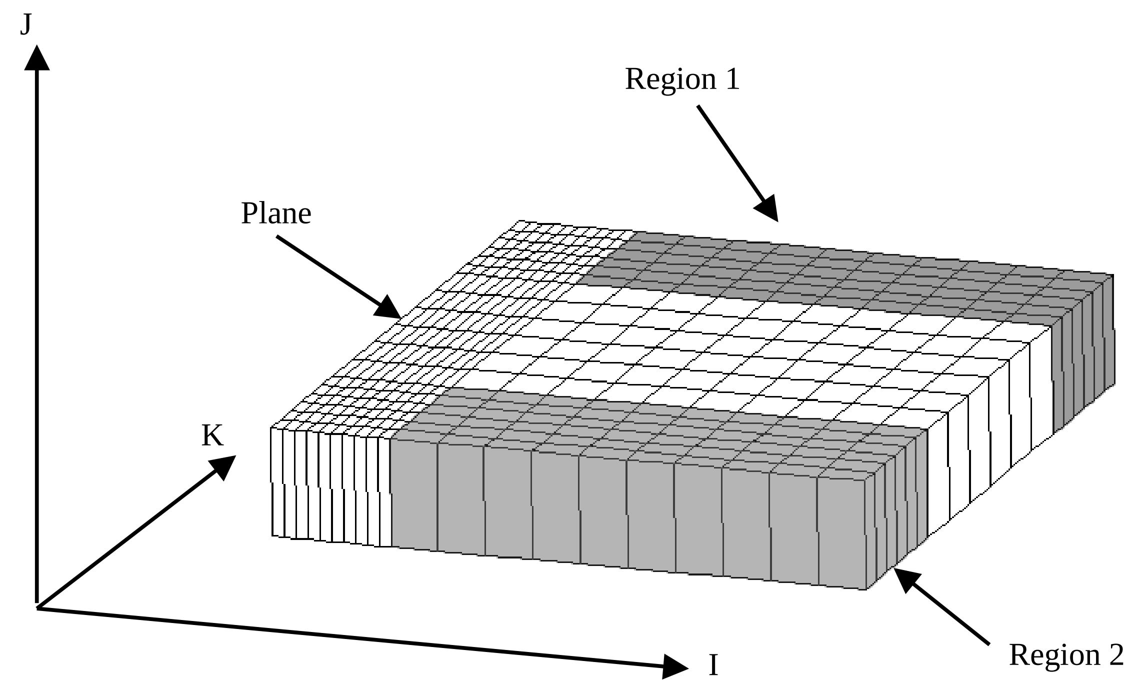


Figure 8.

The Index Progression for the block is

1 ***11 21***; **1 2**; 1 7 13 19;

To delete Region 1 and 2 use the following commands:

DI ***2 3***;**1 2**; 1 2; (Region 1)

DI ***2 3***; **1 2**; 3 4; (Region 2)

An alternative way to delete both Regions at once will be the following:

DI ***2 3***; **1 2**; 1 2 0 3 4;

ZERO indicates not to delete the Region between 2 and 3 in the K-Direction. The D-command can also be used to delete Region 1 and 2 as follows:

D ***2*** **1** 1 ***3*** **2** 2 (Region 1)

D ***2*** **1** 3 ***3*** **2** 4 (Region 2)

Note: One can not use a single D-command to delete both Regions.
For example one want to delete the Plane of the previous model shown in Figure 8.
The input is

DI **–1**; 1 2; 1 4;

or

D **1** 1 1 **1** 2 4

The DI-command requires minus sign in one direction whereas the D-command requires holding one direction constant.

Projection

Projection is the most important concept in modeling because it allows one to create very complex geometries.

The projection itself is divided into two steps:

- *The first step is to define the geometry to project onto*
- *The second step is the actual projection*

The experienced user can combine the options in one command, however one cannot visualize it with the DI SD-command (Display Surface Definition). In this example both steps will be used in order to demonstrate clearly the procedure.

For example consider creating a solid cylinder with a good hexahedral mesh as shown in Figure 9.

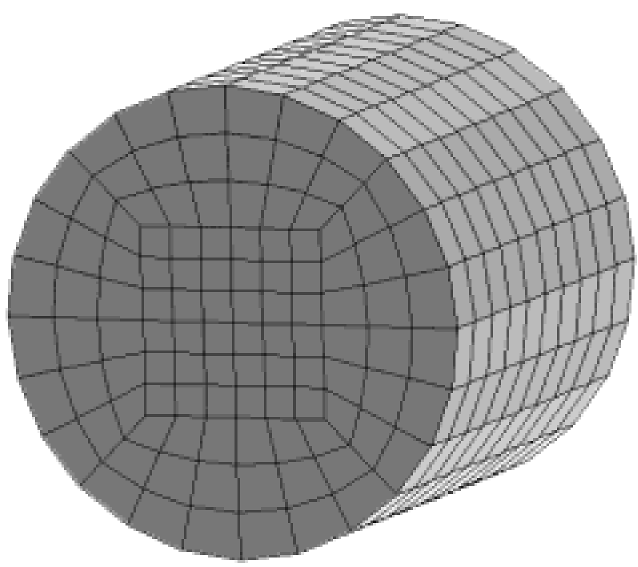


Figure 9. Solid cylinder with good hexahedral mesh

The complete Input Deck for LS-INGRID is:

```
1.  solid cylinder with good hexahedral mesh
2.  sd 1 cyli  0 0 0  0 0 1  5.0
3.  start
4.  1 4 7 10 13;
5.  1 4 7 10 13;
6.  1 11;                                        [1.0]
7.  -2 -2 0 2 2    c two identical coordinates
8.  -2 -2 0 2 2    c two identical coordinates
9.  0 10
10. di  1 2 0 4 5; 1 2 0 4 5; 1 2;
11. sfi  -1 -5; -1 -5; 1 2; sd 1
12. end
13. end
```

Note: LS-INGRID Input Decks require no line numbering.

Line 1:	Header
Line 2:	Cylindrical Geometry (visualize it with DI SD) SD 1: Surface Definition 1 (general SD n) CYLI 0 0 0 0 0 1 5.0: a cylinder at the location 0 0 0 (x y z), the orientation vector 0 0 1 (X Y Z) and the radius 5.0
Line 3:	Start of Index Progression (Part Definiton)
Line 4:	Index Progression in I-Direction
Line 5:	” in J-Direction
Line 6:	” in K-Direction
Line 7:	Coordinates in X-Direction
Line 8:	” in Y-Direction
Line 9:	” in Z-Direction
Line 10:	Deletion command
Line 11:	Projection command SFI <Region in Reduced Index Space> SD n This command allows the projection of Lines(s), Plane(s) or Block(s) onto an exactly specified Surface. This Surface is defined with a Surface Definition. SD n is used when the Surface has already been defined, as in this example, otherwise SD n has to be replaced by the whole Surface Definition expression (cyli 0 0 0 0 0 1 5.0).
Line no.12:	End of Part Definition
Line no.13:	End

This model shows a technique commonly used in LS-INGRID. Two Indices of the Index Progression have identical coordinates (Line no.7 and no. 8) and therefore one Region has zero thickness ("outer" Region, see also Figure 11). The surfaces of the "outer" Region are projected onto the cylinder defined in Surface Definition (Figure10).

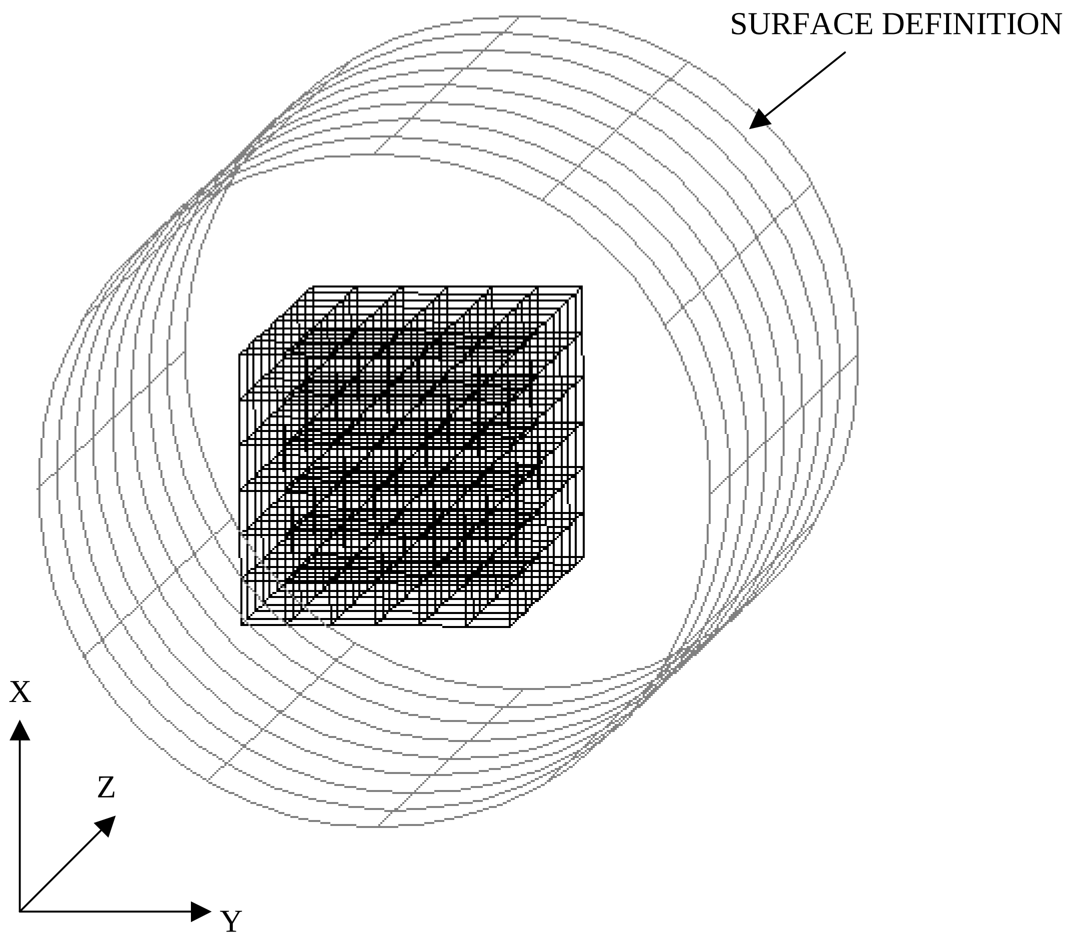


Figure 10. Box and Cylinder

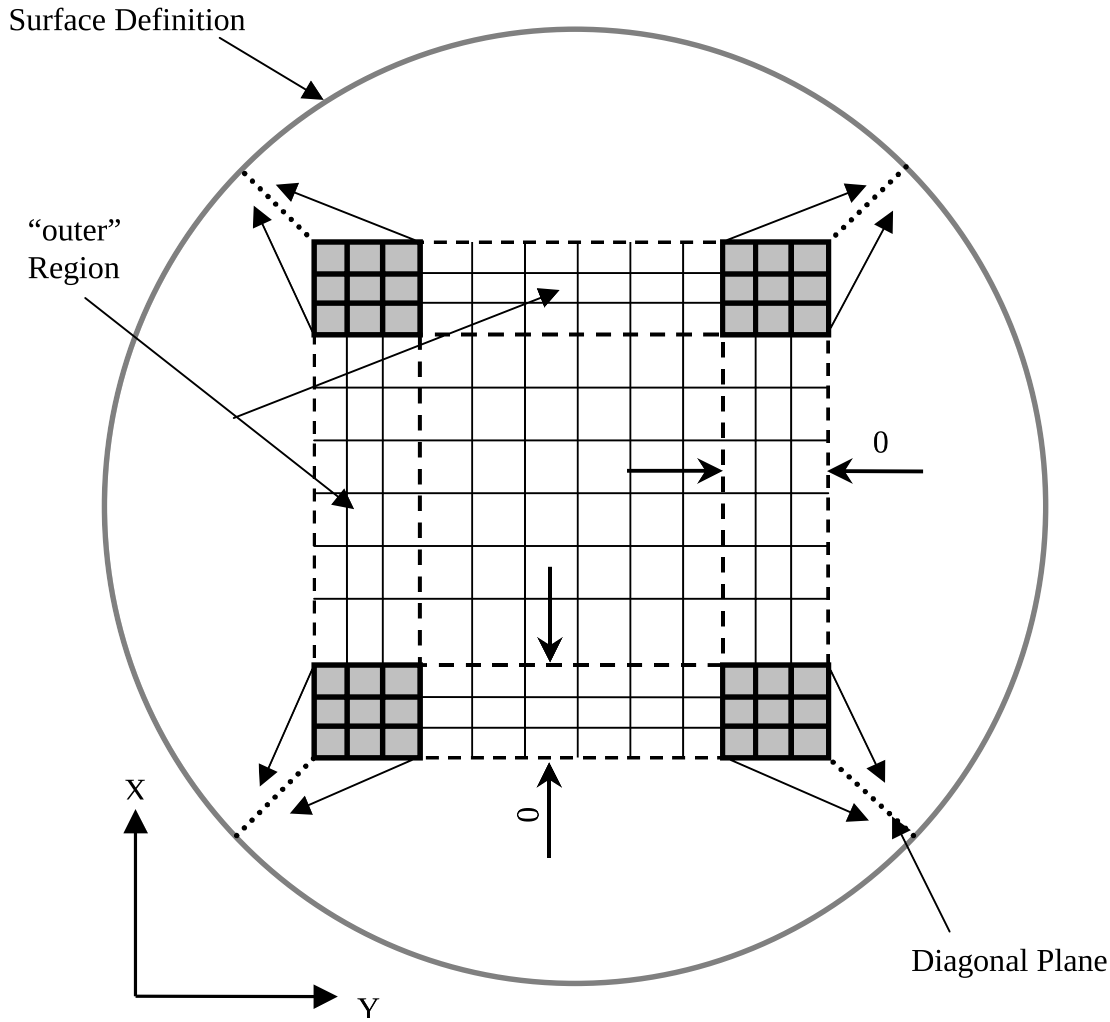


Figure 11. Explanation for the DI-command

Note the dashed lines in Figure 11 are actually at the same location, signified by the zero dimensions. However for the purpose of clarity they are displayed a finite distance apart. The DI-command deletes the shaded regions, otherwise the projection would fail because the Regions overlap. Four Diagonal Planes also result from the projection (dotted lines in Figure 11). At the end of a model definition these Planes have to be merged together with the T *tol* or TP *tol* -commands in the Graphics-Phase of LS-INGRID.

This Model can also be generated using Cylindrical Coordinates.
The Input Deck is:

```
1.    header
2.    start
3.    1 6;    c I-Direction
4.    1 25;   c J-Direction
5.    1 11;   c K-Direction                 [1.1]
6.    0 5     c R-Coordinates
7.    0 360   c Θ-Coordinates
8.    0 10    c Z-Coordinates
9.    cyli    c Cylindrical Coordinate System
10.   end
11.   end
```

The outer and inner radius (R-direction) are specified in Line 6, in Line 7 the degrees of rotation in Θ-Direction and in Line 8 the length of the cylinder (Z-direction). Figure 12 shows the result.

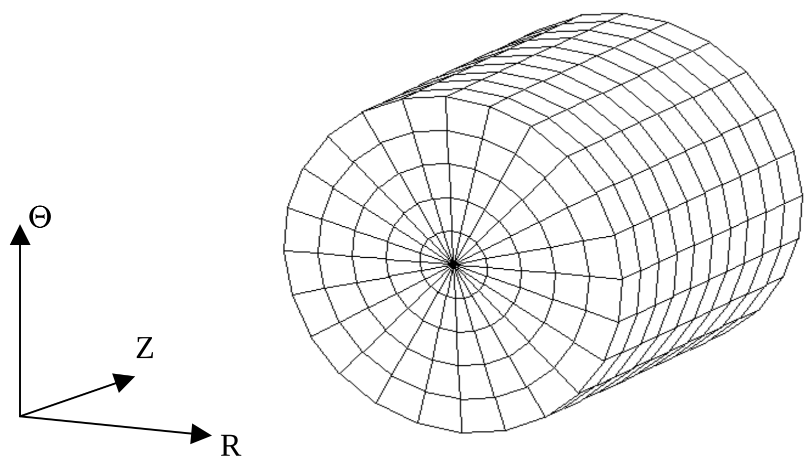


Figure 12. Example in Cylindrical Coordinates

As one can see the elements in the middle are very distorted making them unacceptable for use as a deformable body, but perfectly acceptable for use as a rigid body (material type 20).

EXAMPLES FOR MESH GENERATION

In order to understand the following examples it is necessary to be familiar with the structure of an input file.

```
MODEL NAME (TITLE)

CONTROL COMMANDS
PATR DEFINITION 1
START
     INDEX PROGRESSION
     INITIAL COORDINATES
     LOADS AND BOUNDARY CONDITIONS
     PART CONTROL COMMANDS AND FUNCTIONS
END
CONTROL COMMANDS
PART DEFINITION 2
START
     INDEX PROGRESSION
     INITIAL COORDINATES
     PART CONTROL COMMANDS AND FUNCTIONS
     LOADS AND BOUNDARY CONDITIONS
END

CONTROL COMMANDS
PART DEFINITION N
.
.
.
END
```

1. Circular Plate

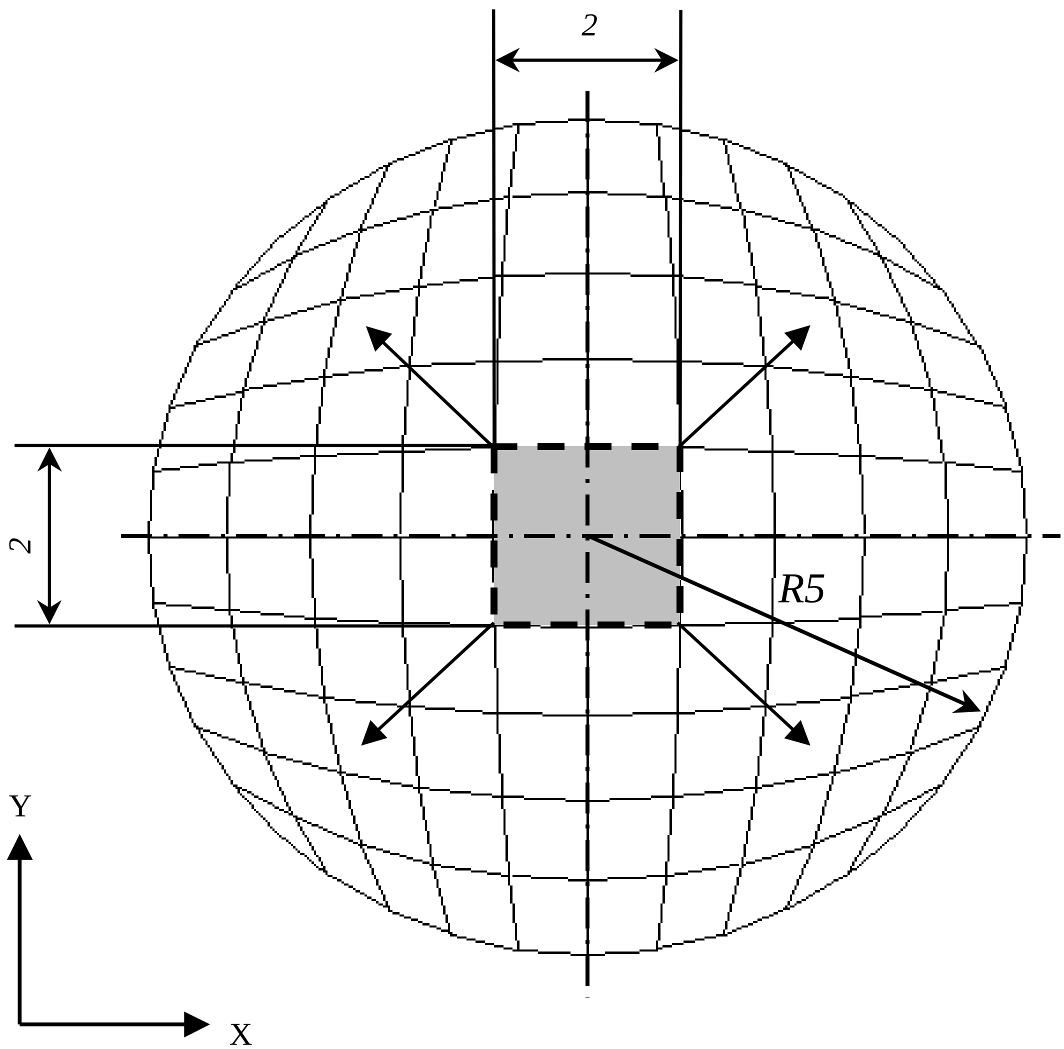


Figure 13. Circular Plate

Input Deck

```
*************************************************************************
simple circular plate

start                     c start defining a part
1 11;                     c 10 elements I-direction
1 11;                     c 10 elements J-direction
-1;                       c flat plane (minus in K-direction)
-1 1                      c coordinates X-direction
-1 1                      c coordinates Y-direction
0                         c location in Z-direction
a 1 1 1 2 2 1 3 5.0       c projection onto circle (radius 5.0)
end                       c end of part
end                       c end of input deck
*************************************************************************
```

Comment

The A-command is used to make the circular plate.

A <region> idir radius

A	arc keyword
<region>	can be a plane or a solid in reduced index space
idir	axes of rotation 1=I, 2=J and 3=K
r	radius

The shaded cube is the Index Progression with 11 points in the I-and J-direction. It is projected onto the circle indicated by the arrows.

2. Nicely Zoned Circular Plate

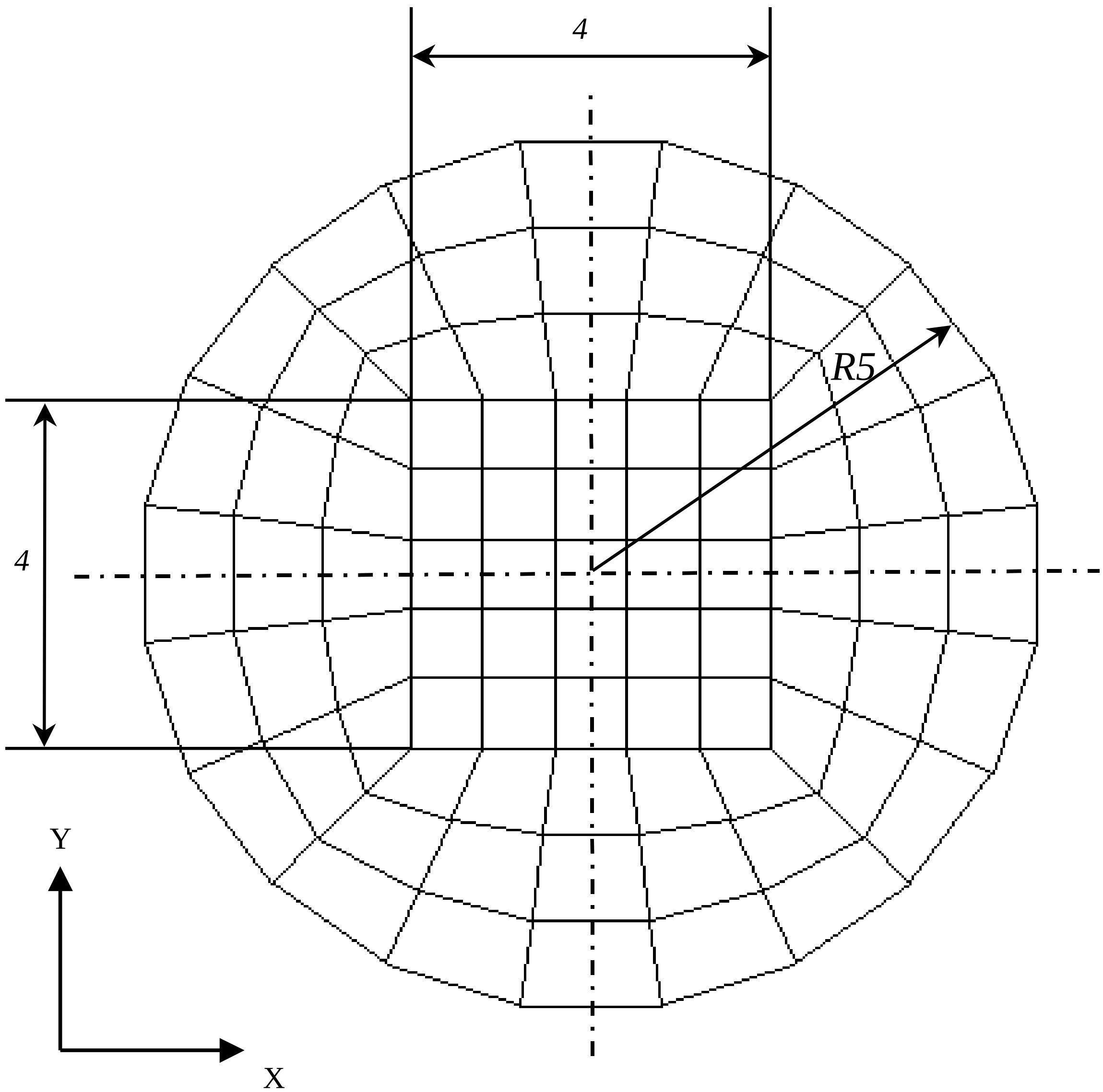


Figure 13. Nicely zoned circular plate

Input Deck

```
*************************************************************************
nicley zoned circular plate

start

1 4 9 12;
1 4 9 12;
-1;

-2 -2 2 2           c two equal coordinates in X-direction
-2 -2 2 2           c two equal coordinates in Y-direction
0                   c position in Z-direction

di 1 2 0 3 4; 1 2 0 3 4; -1;

a 1 1 0 4 4 0 3 5.0

end

end
*************************************************************************
```

Comment

This method is similar to the one used on page 10/11.

3. Half Nicely Zoned Circular Plane

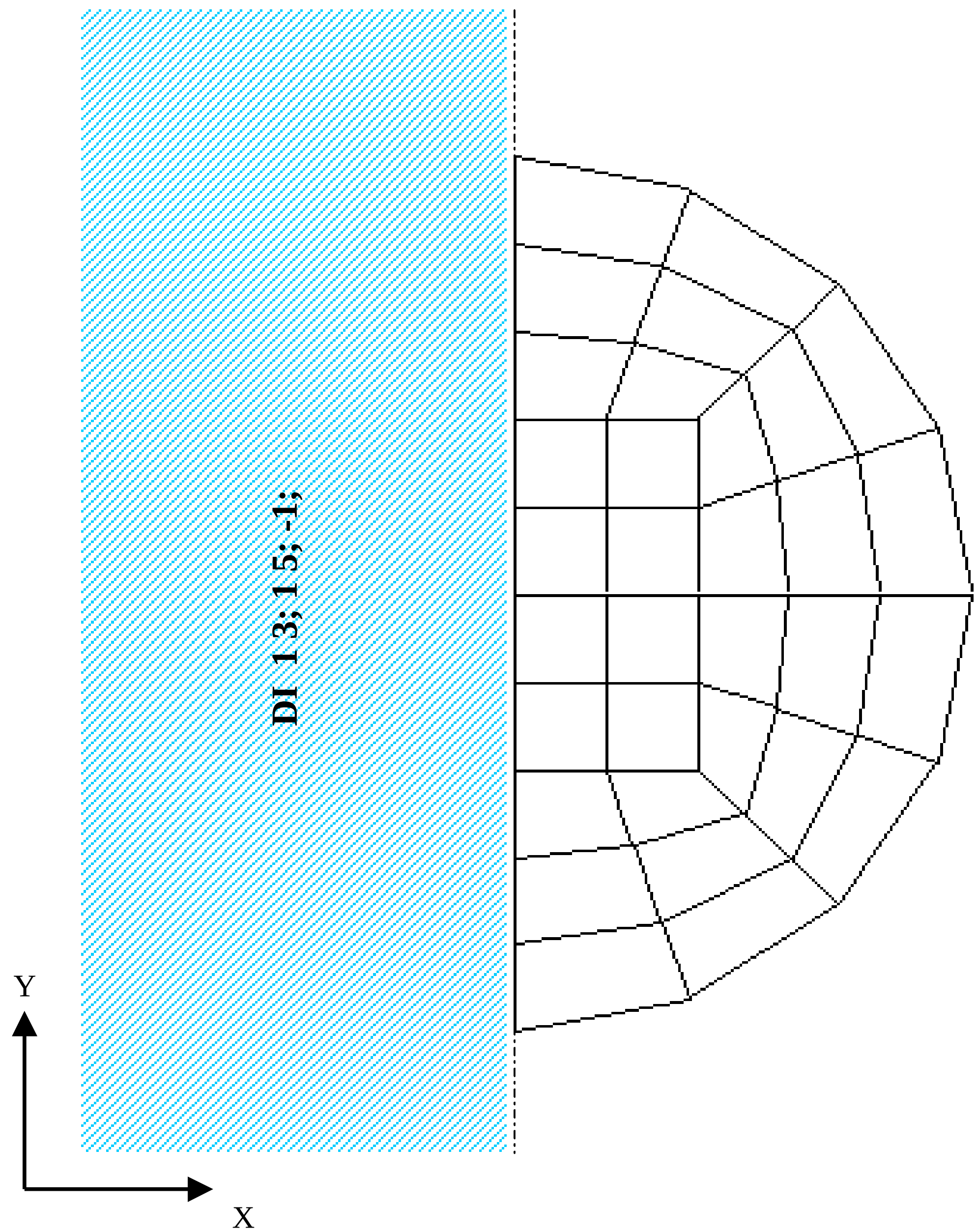


Figure 14. Half nicely zoned circular plate

Input Deck

```
*************************************************************************
half nicley zoned circular plate

start

1 4 6 8 11;                  c 5 indices
1 4 6 8 11;                  c 5 indices
-1;                          c minus sign for plane

-2 -2 0 2 2                  c 2 identical coordinates
-2 -2 0 2 2                  c 2 identical coordinates
0

di 1 2 0 4 5; 1 2 0 4 5; -1;
di 1 3;1 5 ;-1 ;             c deletes half of the model

a 1 1 0 5 5 0 3 5.0          c projection

end

end
*************************************************************************
```

Comment

This input file is similar to the previous example except that 5 Index Progressions are used instead of 4. This makes it possible to create only half of the circle.

4. Quarter of Circular Plane

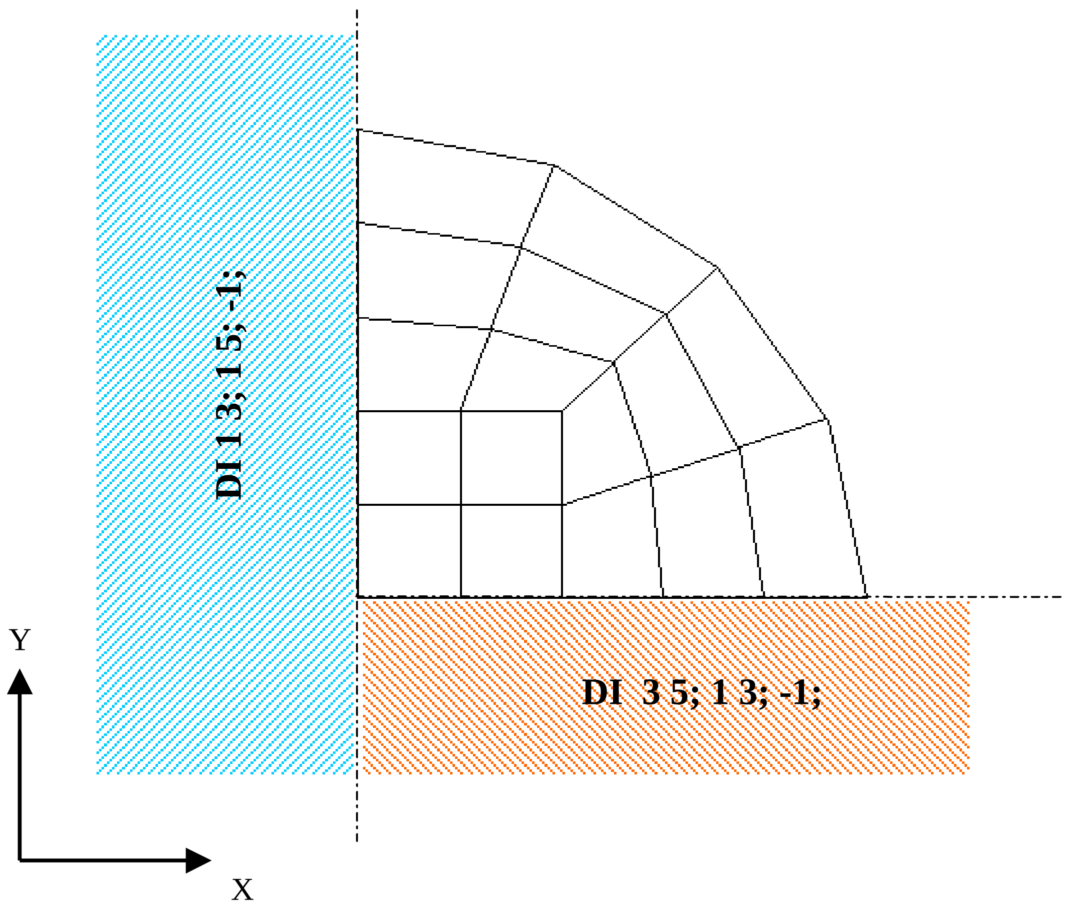


Figure 15. Quarter of nicely zoned circular plane

Input Deck

```
****************************************************************************
quarter nicley zoned circular plane

start

1 4 6 8 11;                    c 5 indices
1 4 6 8 11;                    c 5 indices
-1;

-2 -2 0 2 2                    c 2 identical coordinates
-2 -2 0 2 2                    c 2 identical coordinates
0

di 1 2 0 4 5; 1 2 0 4 5; -1;
di 1 3;1 5;-1 ;                c deletes half
di 3 5; 1 3; -1;               c deletes quarter

a 1 1 0 5 5 0 3 5.0            c projection on circle

end

end
****************************************************************************
```

Comment

Note the comment next to the DI-commands.

5. Circular Plane with outer Liner

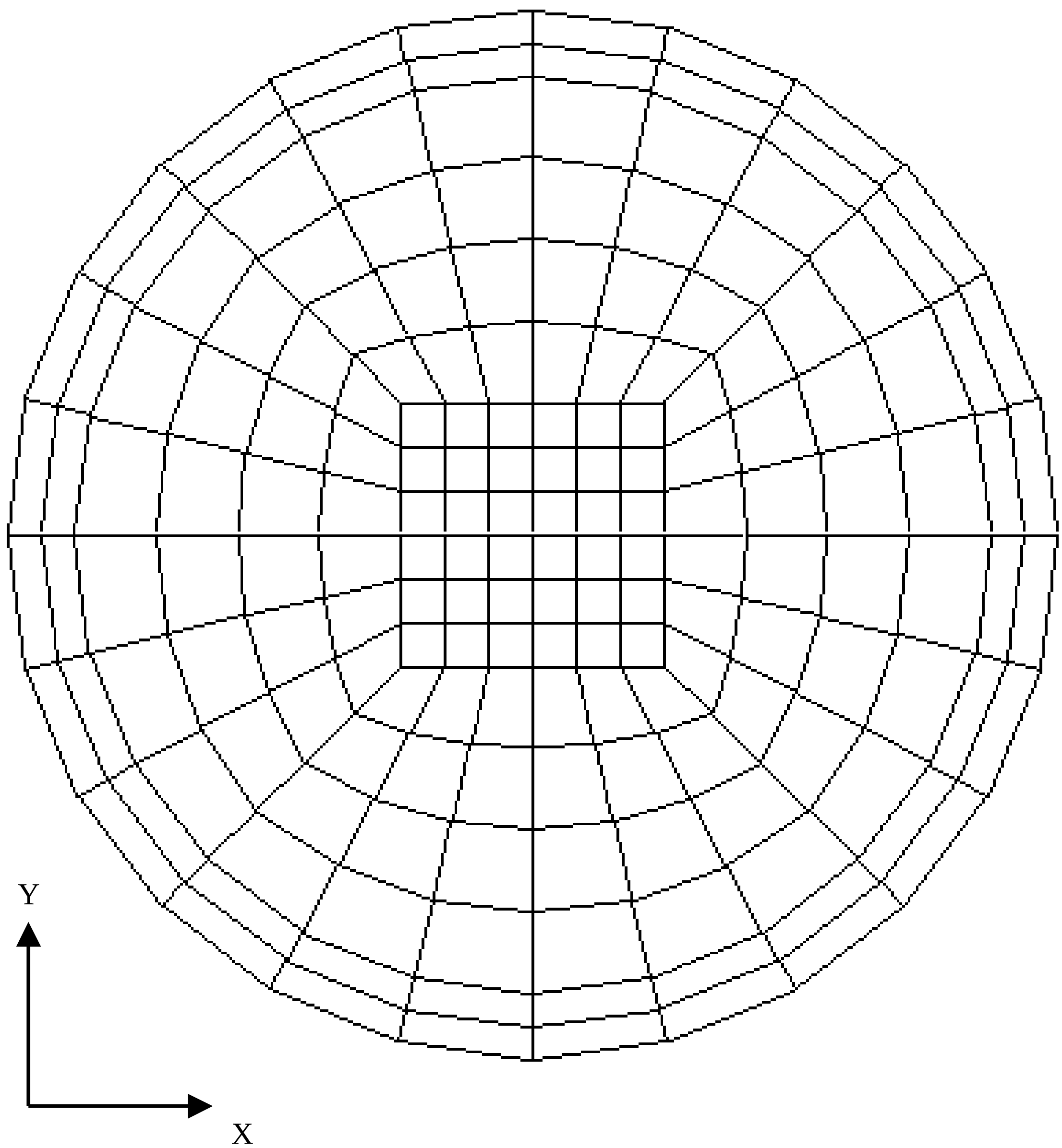


Figure 16. Circular plate with outer liner

Input Deck

```
*************************************************************************
circular plate with outer liner

start

1 3 7 13 17 19;
1 3 7 13 17 19;
-1;

-1 -1 -1 1 1 1                c 3 identical coordinates
-1 -1 –1 1 1 1                c 3 identical coordinates
0                             c location Z-direction

di 1 3 0 4 6; 1 3 0 4 6; -1;

a 1 1 0 6 6 0 3 4.0           c outer circle
a 2 2 0 5 5 0 3 3.5           c inner circle

end

end
*************************************************************************
```

Comment

In order to create this model the DI-command has to be adapted because there are 3 identical coordinates.

6. Circular Plane with Hole

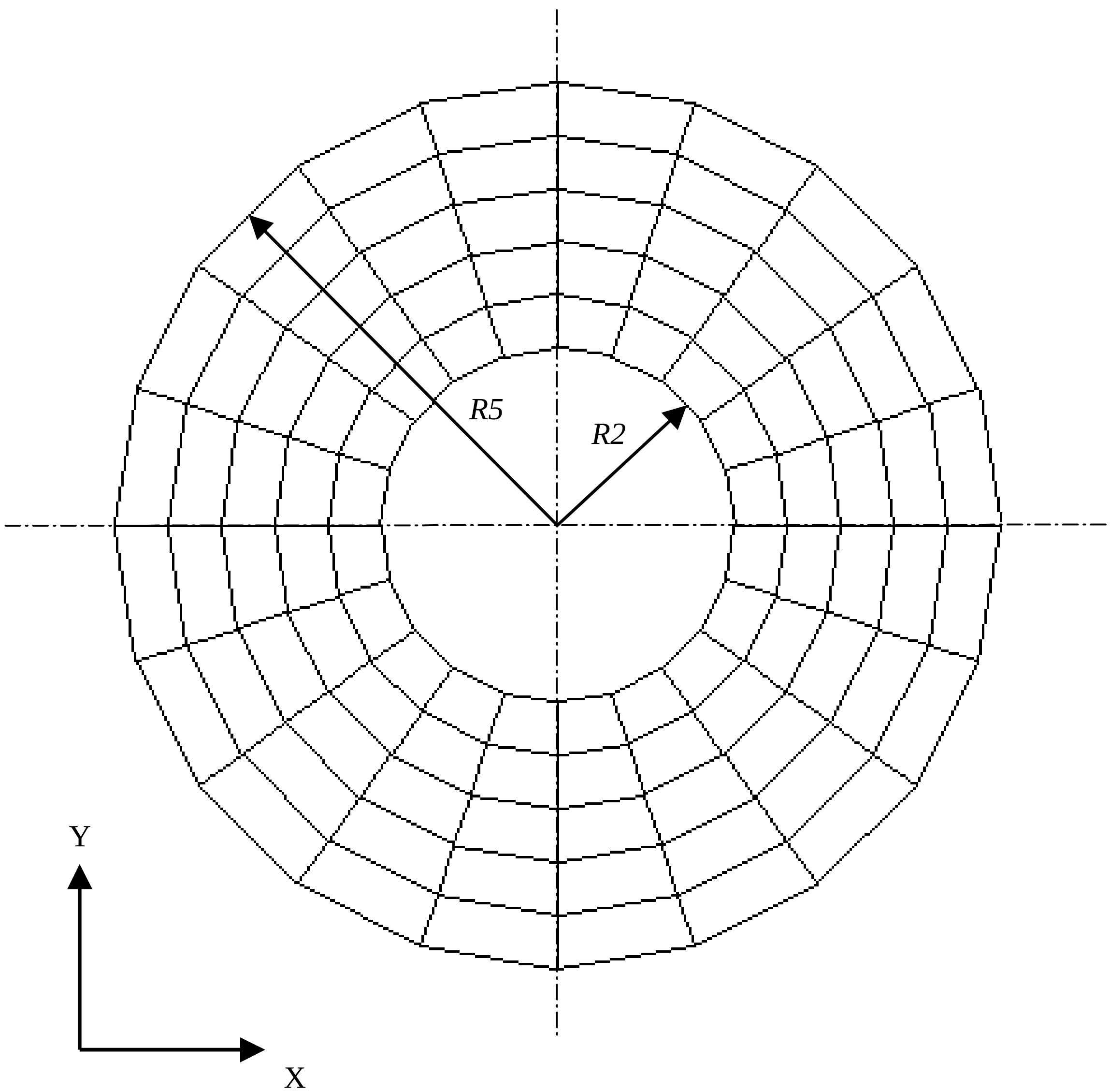


Figure 17. Circular plane with hole

Input Deck 1

```
************************************************************************
circular plate with hole

start
1 6;                  c 5 elements between inner and outer
                        radius
1 21;                 c 20 elements around cylinder
-1;                   c flat Plane
2.0 5.0               c inner and outer radius
0 360                 c rotation in Θ-direction in degrees
0                     c location in Z-direction
cyli                  c cylindrical coordinate system
end
end
************************************************************************
```

Input Deck 2

```
************************************************************************
circular plate with hole

start
1 4 9 12;
1 4 9 12;
-1;
-1 -1 1 1                       c two identical coordinates
-1 -1 1 1                       c two identical coordinates
0.0

di 1 2 0 3 4; 1 2 0 3 4; ;
a 1 1 0 4 4 0 3 3.0             c outer circle (r=3.0)

di 2 3; 2 3; -1;                c deletes inner region
a 2 2 0 3 3 0 3 2.0             c inner circle = hole(r=2.0)
end
end
************************************************************************
```

Comment

A Cylindrical Coordinate System is used in Input Deck 1. This is indicated by the cyli-command. Alternatively, a Cartesian Coordinate System in combination with the A-command can be used as demonstrated in Input Deck 2.

7. Hollow Cylinder

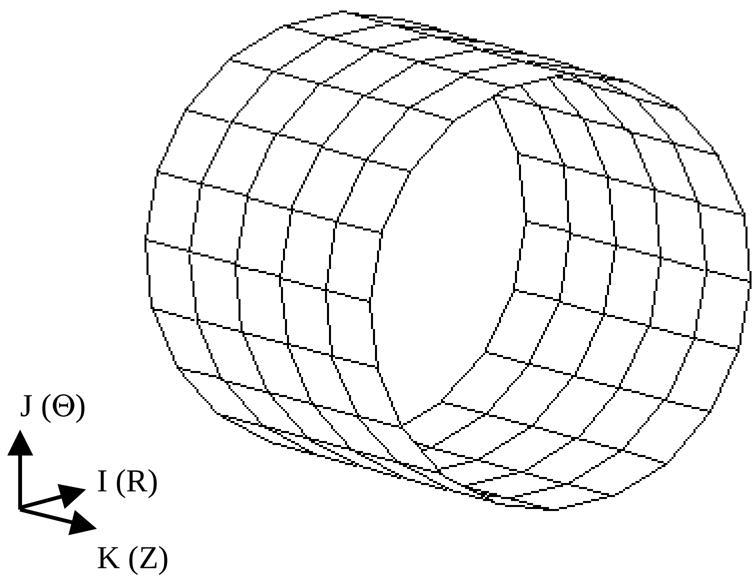


Figure 18. Hollow Cylinder

Input Deck

```
************************************************************************
hollow cylinder

start
-1 ;                    c plane in I-direction
1 21;                   c 20 elements around cylinder
1 6;                    c 5 elements along cylinder
5                       c radius of plane
0 360                   c rotation in degrees about z axes
0 10                    c length
cyli                    c cylindrical coordinate system
end
end
************************************************************************
```

Comment

This cylinder is defined as a plane in the I-direction with radius 5 (R-axis) and length 10 (Z-axis).

8. Hollow Cylinder with Stiffeners

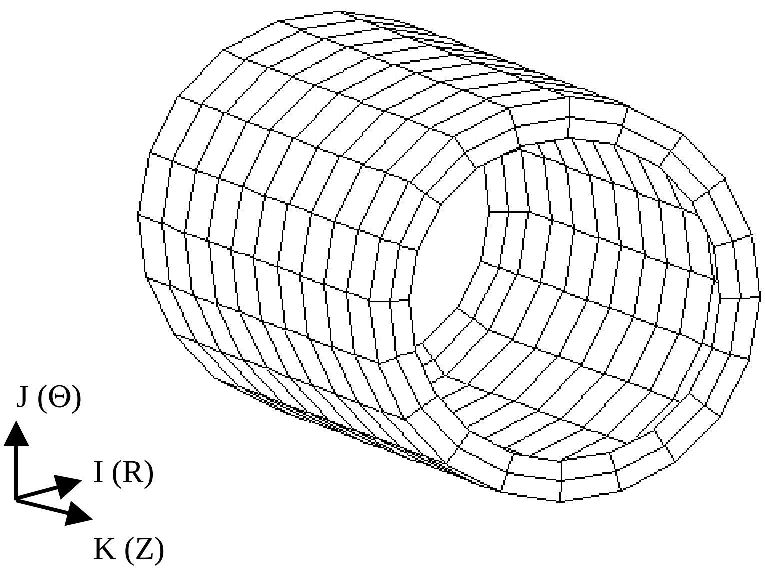


Figure 19. Hollow cylinder with stiffeners

Input Deck

```
*********************************************************************
hollow cylinders with stiffeners

start
1 -3;       c two regions in I-direction
1 21;       c 20 elements around cylinder
-1 -11;     c two planes in K-direction
2.0 2.5
0 360
0 10
cyli
end
end
*********************************************************************
```

Comment

Region 1 3; 1 21; -1 –11; creates the stiffeners. The additional minus sign in I-direction creates the cylinder.

9. Square Plane with Hole

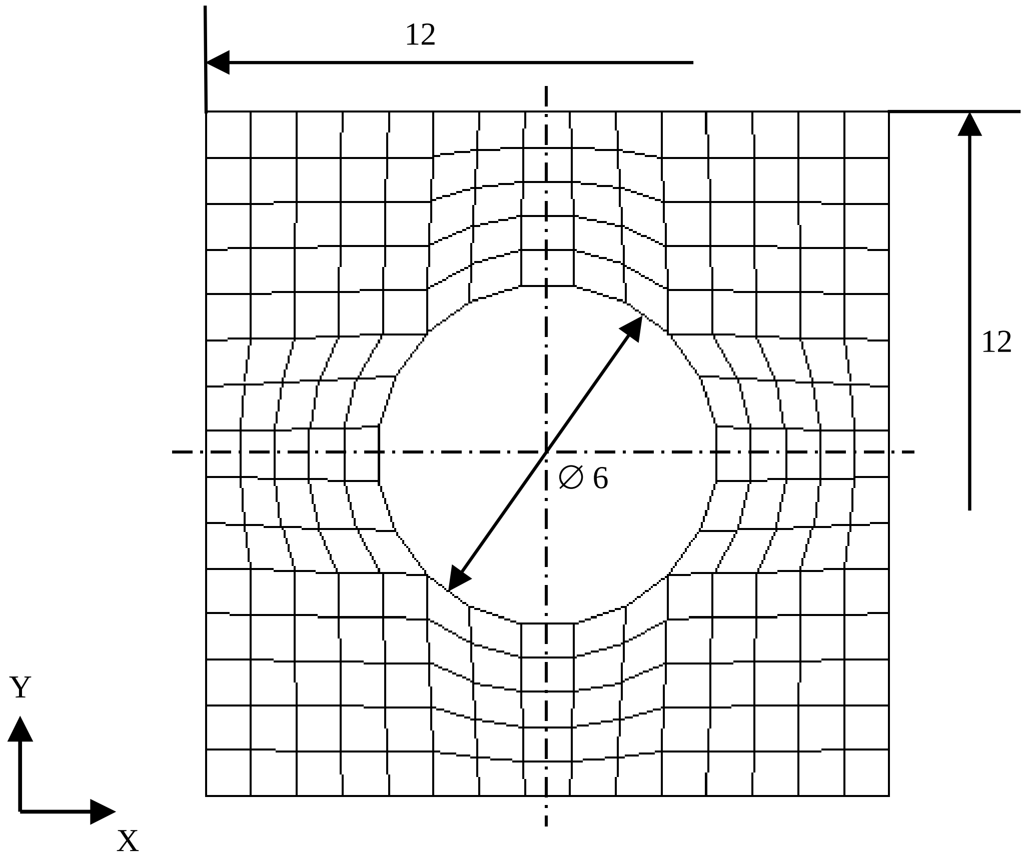


Figure 20. Square plane with hole

Input Deck

```
************************************************************************
plane with hole
start
1 6 11 16;
1 6 11 16;
-1;
-6 -2 2 6
-6 -2 2 6
0
d 2 2 0 3 3 0            c delete inner region
a 2 2 0 3 3 0 3 3.0      c deleted region projected onto
                           circle (r=3.0)
end
end
************************************************************************
```

10. Square Plane with Hole2

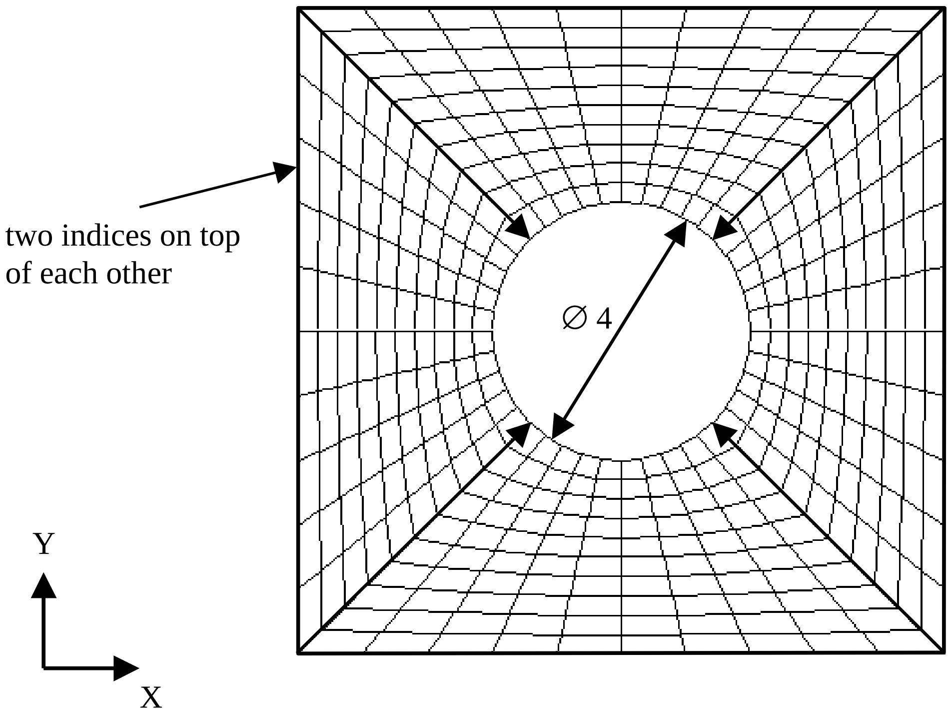


Figure 21. Square plane with hole (transition region)

Input Deck

```
************************************************************************
plate with hole2

start
1 11 21 31;
1 11 21 31;
-1;
-5 -5 5 5                     c two identical coordinates
-5 -5 5 5                     c  ″     ″             ″
0
di 1 2 0 3 4; 1 2 0 3 4; -1;
d 2 2 0 3 3 0                 c delete inner region
a 2 2 0 3 3 0 3 2.0           c projection of region
end
end
************************************************************************
```

11. Triangular Plane (bad mesh)

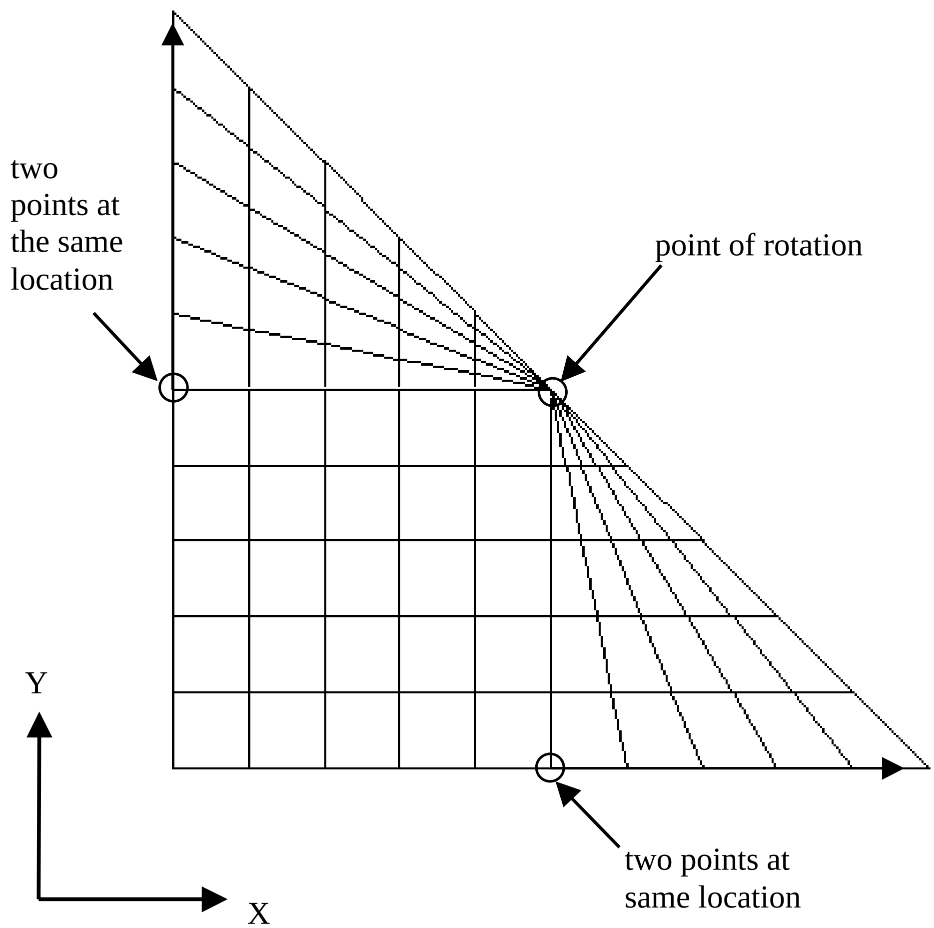


Figure 22. Triangular plane (bad mesh)

Input Deck

```
************************************************************************
triangular plane

start
1 6 11;
1 6 11;
-1;
0 5 5                c two identical coordinates
0 5 5                c two identical coordinates
0

di 2 3; 2 3; -1      c see page 11 if not clear

ma 3 1 0 x 5         c move point in x-direction 5 units
ma 1 3 0 y 5         c move point in y-direction 5 units

end
end
************************************************************************
```

Comment

The MA-command modifies the coordinates of a <point>.

MA <point> n d_x d_y d_z

ma	modify coordinate of a point
<point>	point in reduced index space
n	flag indicating which coordinates to change, e.g. x, y, z, xy, xz, xyz,…
d_x d_y d_z	value which *is* ***added*** to the old coordinate

12. Triangular Plane (good mesh)

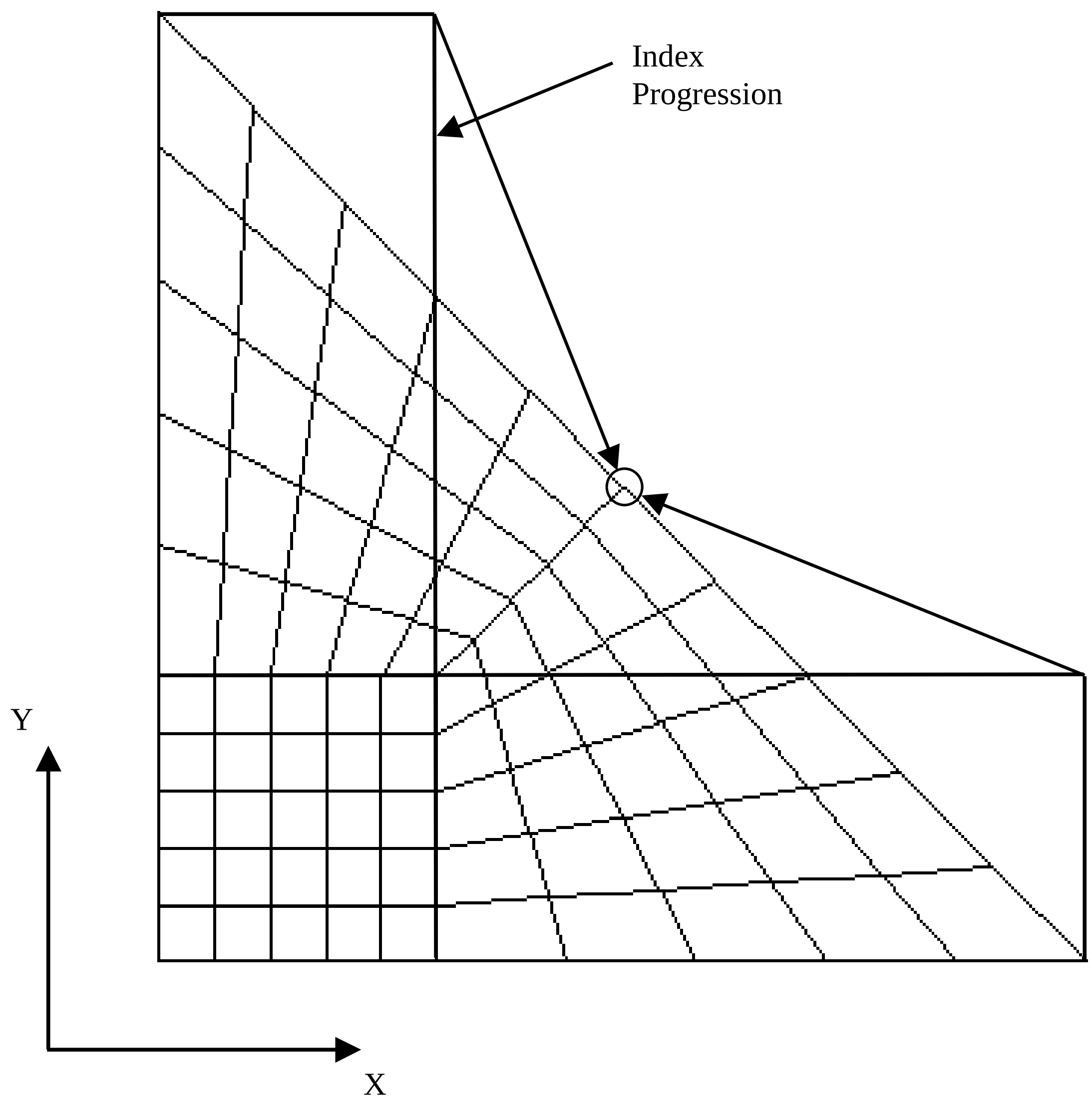


Figure 23. Triangular plane with good mesh

Input Deck

```
*************************************************************************
triangular plane

start
1 6 11;
1 6 11;
-1;
0 3 10
0 3 10
0
di 2 3; 2 3; -1;        c deletion of a quarter
ma 3 2 0 xy -5 2        c move these two points to
ma 2 3 0 xy 2 –5        c the same location
end
end
*************************************************************************
```

Comment

Figure 23 shows the initial design of this plane and the moving direction of the points (arrows) initiated by the MA-commands.

13. Circular Plane with offset Hole

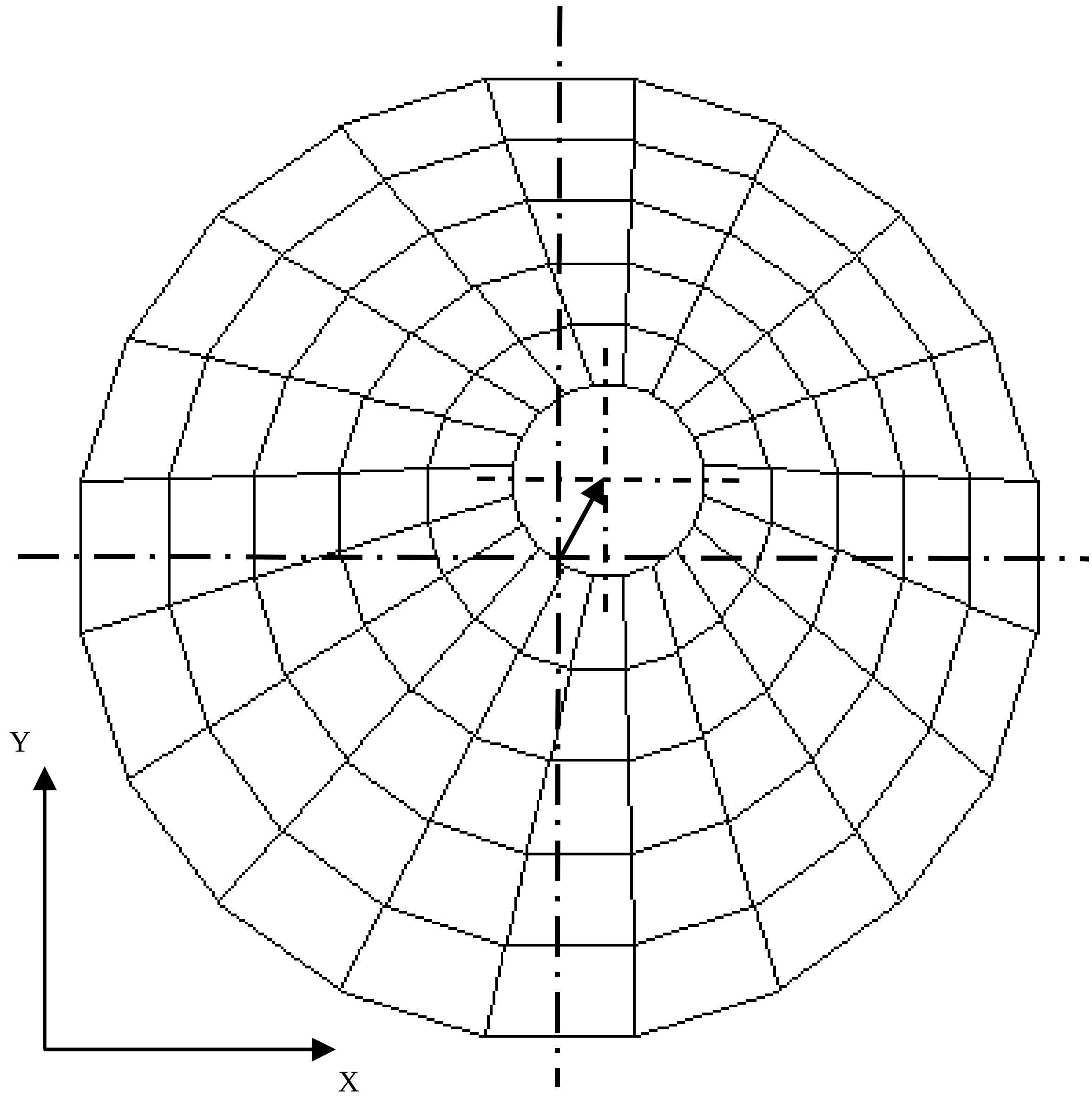


Figure 24. Circular plane with offset hole

Input Deck

```
************************************************************************
circular plate with offset hole

start
1 6 11 16;
1 6 11 16;
-1;
-1 -1 1 1                    c two indices on top of
-1 -1 1 1                    c each other
0

di 1 2 0 3 4; 1 2 0 3 4; -1;

a 1 1 0 4 4 0 3 5.0

di 2 3; 2 3; -1;             c delete inner region
mb 2 2 0 3 3 0 xy 0.5 0.8    c moving deleted region
                               outside the center

a 2 2 0 3 3 0 3 1.0          c projection onto moved
                               region
end
end
************************************************************************
```

Comment

The MB command is the same as the MA-command, but instead of moving a <point> it moves a <region>. Figure 24 shows the direction the region is moved and the resulting distortion of the mesh.

MB <region> n d_x d_y d_z

mb	modify coordinate of a region
<region>	point in reduced index progression
n	flag indicating which coordinates to change, e.g. x, y, z, xy, xz, xyz,…
d_x d_y d_z	value which is ***added*** to the old coordinate

14. Helix

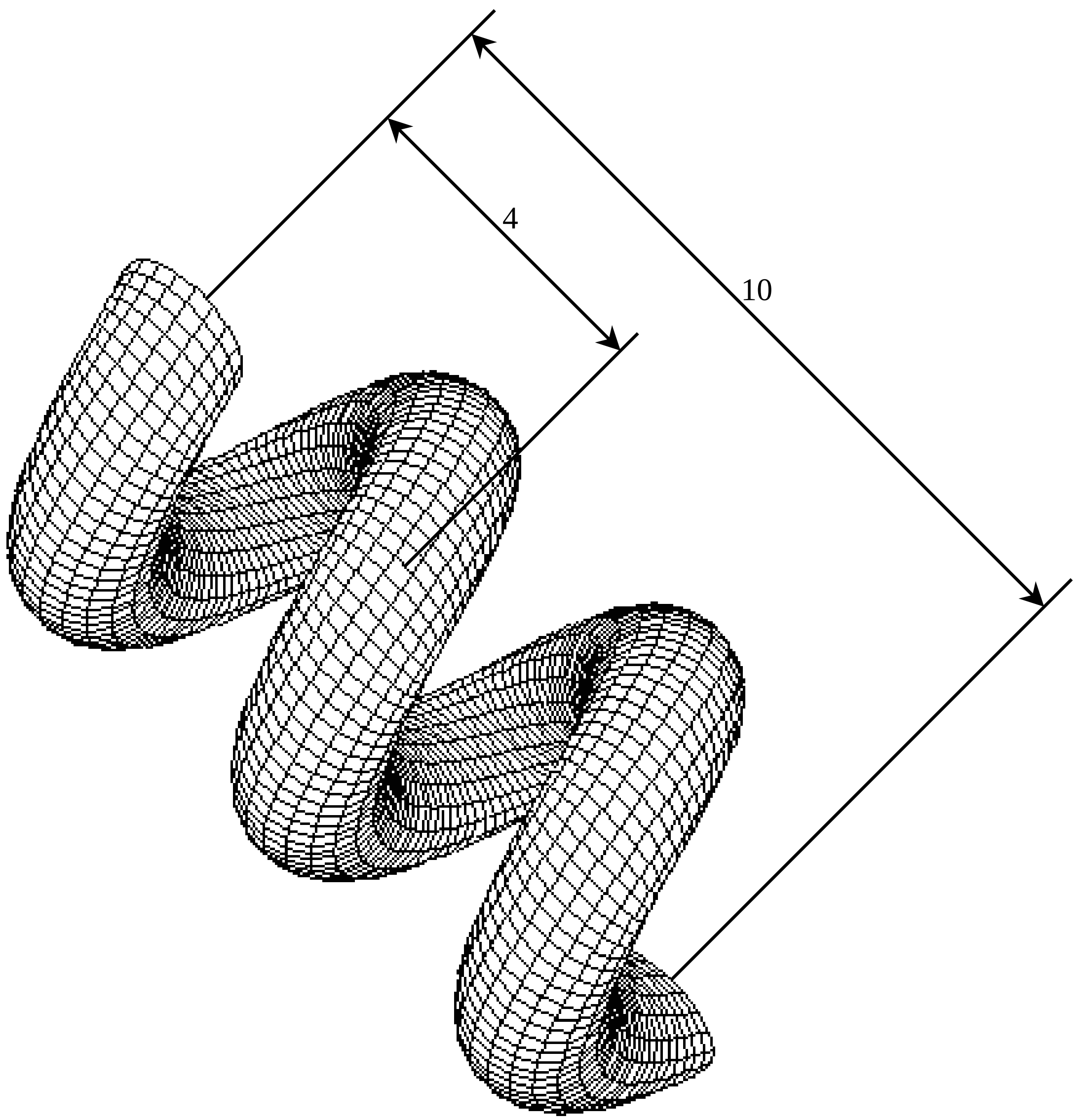


Figure 25. Helix

Input Deck

```
***********************************************************************
EX12: Helix

start

1 6;
1 200;
1 6;

1.5 2.5                c RI and Ra
0 900                  c 0°-900° rotation
-0.5 0.5               c length

ma 0 2 0  z  10.0      c point moving command

a 0 0 0 0 0 0  j  1.0  c projection circle r=1.0
cyli                   c cylindrical coordinates
end
end
***********************************************************************
```

Comment

This example is useful for creating threads. The logic behind this modeling technique is to first create a long beam and then to transform the beam into the cylindrical coordinate system. The first X-coordinate (R_i) is the inner radius of the helix while the other influences the mesh size (R_a). The value 10 in the Z-direction for the MA-command results from the assumption that one wants to have 2.5 rotations of the beam about the Z-axis (360°* 2.5= 900°) with the distance of 4 between the point at the beginning and the end of a complete rotation (Figure 25).

15. Spherical Shell made of Sheet

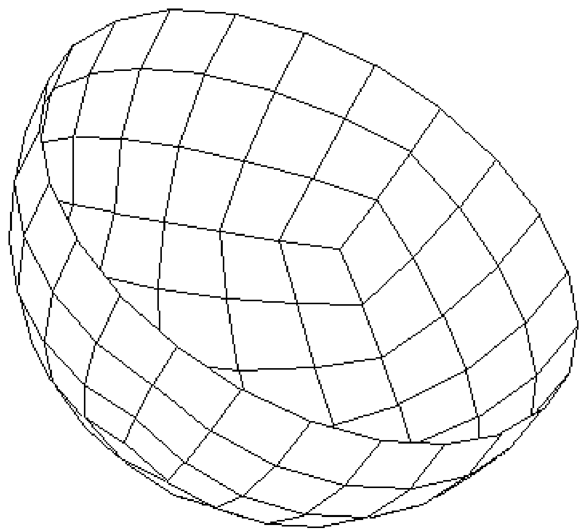


Figure 26. Spherical shell made of sheet

Input Deck

```
************************************************************************
spherical shell made of sheet
sd 1 sp 0 0 0 5.0              c surface definition 1
start
1 4 10 13;
1 4 10 13;
-1;
-5 -5 5 5                      c two indices on
-5 -5 5 5                      c top of each other
0
pb 2 2 0 3 3 0 z 5             c moving region Z-direction
di 1 2 0 3 4; 1 2 0 3 4; ;     c standard deletion when two
                                 indices are on top of each
                                 other
sf 0 0 1 0 0 1 sd 1            c projection of region in
                                 reduced index space onto
                                 surface definition 1
end end
************************************************************************
```

Comment

The Surface Definition is not in the Part definition because it can be visualized with the DI SD (DIsplay Surface Definition)-command. One can also use it more than once for the projection.

First a Plane is created in Index Progression with an additional frame at the same location. In order to create a one-sided opened box the Plane is moved with the PB-command. This is shown in Figure 26a. Then the SF-command is used to project the whole box onto the sphere.

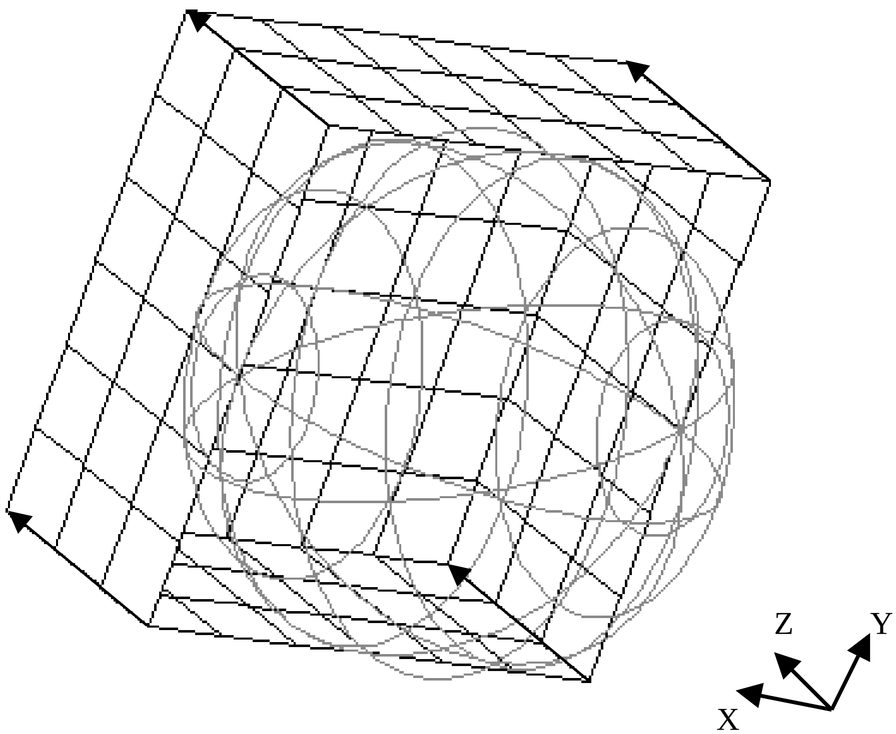


Figure 26a. Surface definition and non-projected box

SP $p_x p_y p_z$ r

SP	define a sphere
$p_x p_y p_z$	center of sphere
r	radius of sphere

PB <region> n $d_x d_y d_z$

PB	point function; change coordinates of <region>
<region>	region in reduced index space
n	flag indicating which coordinates to change, e.g. x, y, z, xy, xz, xyz,…
d_x d_y d_z	*new coordinates*

SF <region>itype
SFI <index progression> itype

SF, SFI	surface command
<index progression>	indices in reduced index space
<region>	region in reduced index space
itype	itype = sd n (surface definition n)
	itype ≠sd n then put in the complete Surface Definition

16. Spherical Shell with Hole

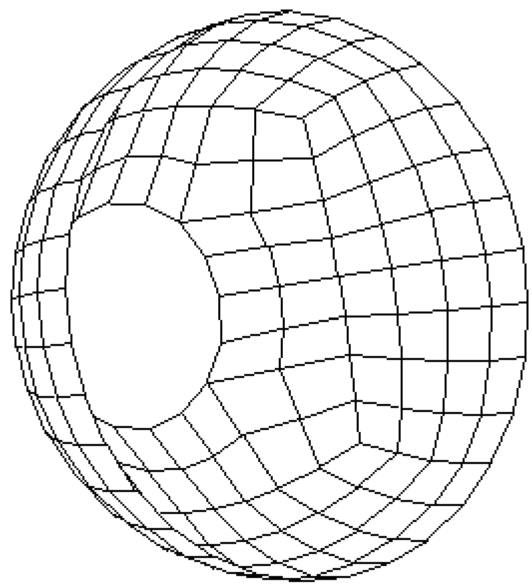


Figure 27. Spherical shell with hole

Input Deck

```
************************************************************************
spherical shell with hole

sd 1 sphe 0 0 0 5.0          c surface def. 1
sd 2 cyli 0 0 0 0 0 1 2.0    c surface def. 2
start
-1 3 7 -9;                   c define the box
-1 3 7 -9;                   c only with index progressions
1 -5;                        c (compare with box before!)
-5 -2 2 5
-5 -2 2 5
0 5
sfi -1 -4; -1 -4; ; sd 1     c projection onto sphere
d 2 2 0 3 3 0                c delete inner region for the
                               hole
sfi 2 3; 2 3; ; sd 2         c projection of the inner region
                               onto the cylinder of sd 2
end end
************************************************************************
```

Comment

This is another way to create the box. It is done without any coordinate manipulating commands, only with Index Progression. Figure 27a shows the box with all the surface definitions. First the whole box is projected onto the sphere then the hole onto the cylinder. There are other ways to make the hole, e.g. with the A-command.

CYLI $p_x p_y p_z$ $v_x v_y v_z$ r

CYLI	define cylindrical surface
$p_x p_y p_z$	point on the axis of rotation
$v_x v_y v_z$	orientation vector
r	radius r of cylinder

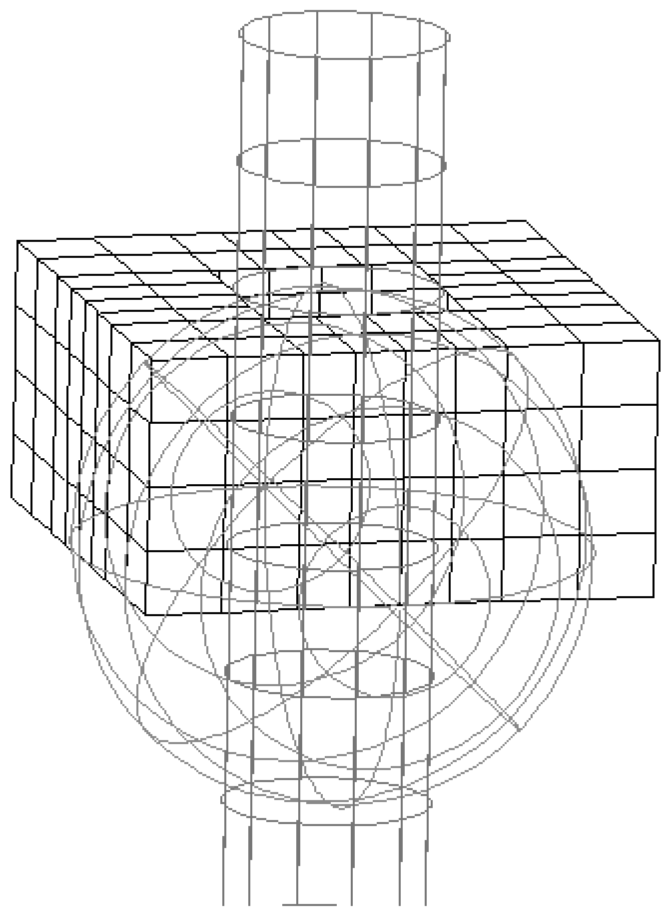


Figure 27a. Surface definition 1 and 2 with the box

17. Spherical Shell with transition Hole

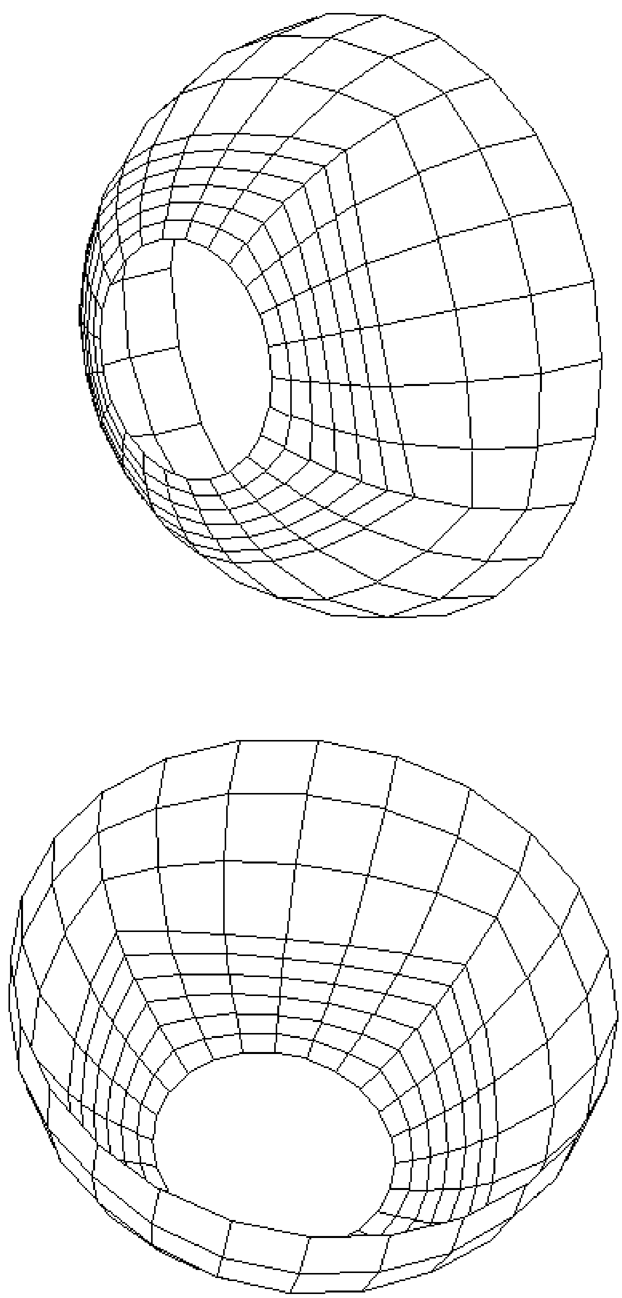


Figure 28. Spherical shell with hole (transition region)

Input Deck

```
**************************************************************************
spherical shell with transition hole (2 parts)

sd 1 sphe 0 0 0 5.0             c sphere r=5.0
sd 2 cyli 0 0 0 0 0 1 2.0       s cylinder r=2.0

start                           c start 1st part (sphere)
-1 -7;
-1 -7;
1 4;
-5 5
-5 5
0 5

sfi -1 -2; -1 -2; ; sd 1        c project whole box onto
                                  sphere
end                             c end 1st part

start                           c start 2nd part (flat plane)
1 7 13 19;
1 7 13 19;
-1;
-5 -5 5 5                       c two indices on
-5 -5 5 5                       c top of each other
5

di 1 2 0 3 4; 1 2 0 3 4; ;      c see page 11 if not clear

sf 0 0 1 0 0 1 sd 1             c projection onto sphere

d 2 2 0 3 3 0                   c delete hole

sf 2 2 0 3 3 0 sd 2             c project hole onto cylinder

end                             c end 2nd part
end
************************************************************
```

Comment

The outer lines of part 2 (flat plane) must have the same length as the sphere's diameter; otherwise it will not fit (Figure 28a). As an exercise try another size for the plane.

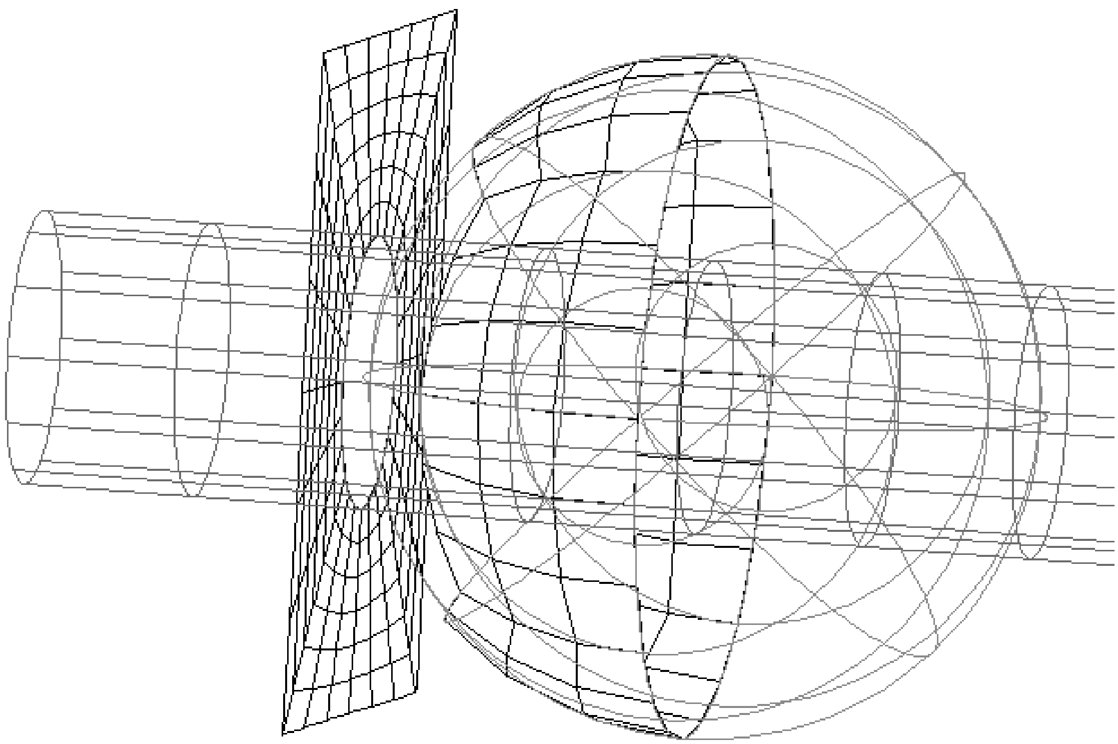


Figure 28a. Surface definitions 1 and 2 and the flat plane before projection

18. Solid Sphere with outer Liner

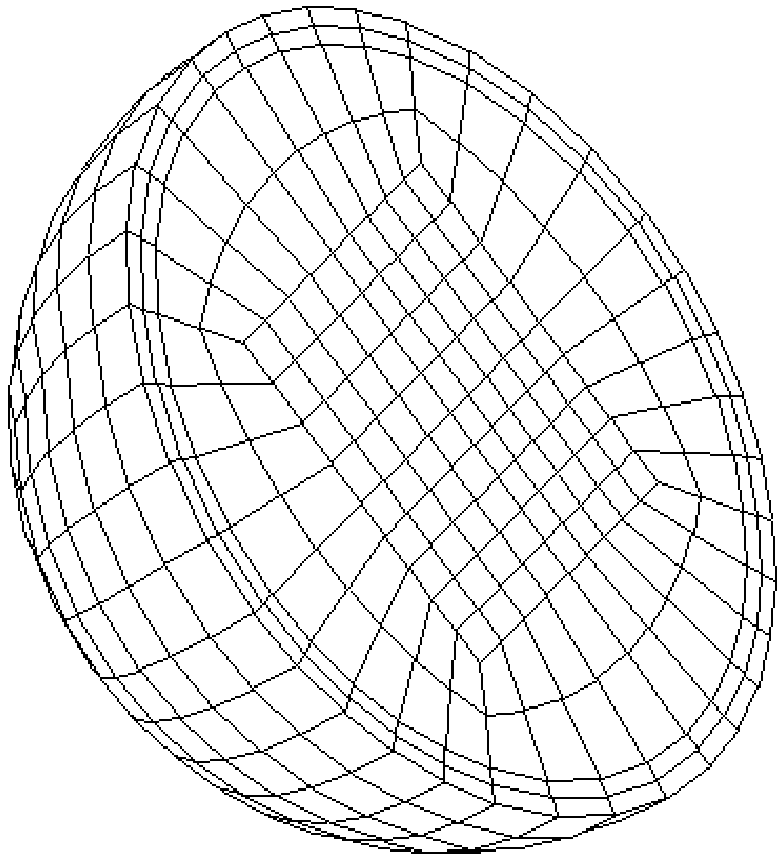


Figure 29. Solid sphere

Input Deck

```
************************************************************************
spherical solid with outer liner

sd 1 sphe 0 0 0 5                c two spheres as
sd 2 sphe 0 0 0 5.5              c surface definitions
start

1 3 5 9 14 16 18;                c index progressions
1 3 5 9 14 16 18;                c
1 3 5 9 14 16 18;                c

-2.5 -2.5 -2.5 0 2.5 2.5 2.5     c three indices on
-2.5 -2.5 -2.5 0 2.5 2.5 2.5     c top of
-2.5 -2.5 -2.5 0 2.5 2.5 2.5     c each other

di 1 3 0 5 7; 1 3 0 5 7; ;       c see page 11
di 1 3 0 5 7; ; 1 3 0 5 7;       c if it’s not
di ; 1 3 0 5 7; 1 3 0 5 7;       c clear

sfi -2 -6; -2 -6; -2 -6; sd 1    c projection onto sd 1
sfi -1 -7; -1 -7; -1 -7; sd 2    c projection onto sd 2

di 1 4;1 7; 1 7;                 c delete half

end
end
************************************************************************
```

19. Cylinder with Spherical Cap

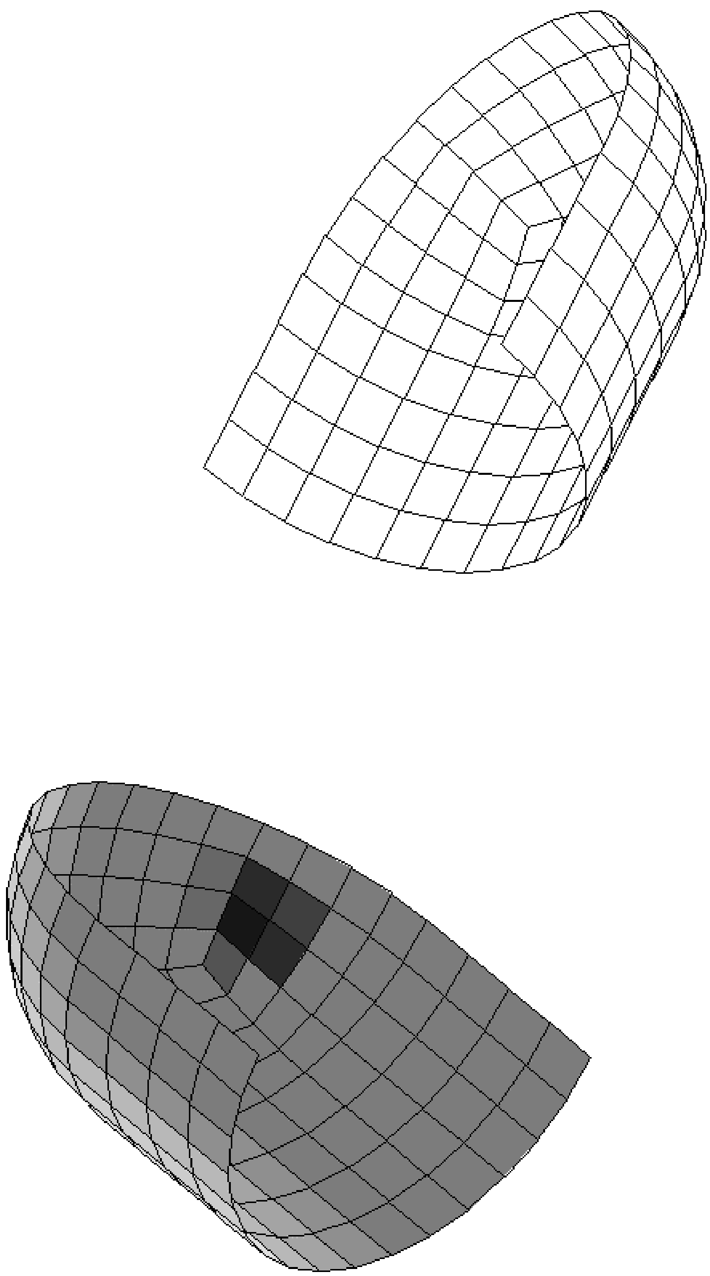
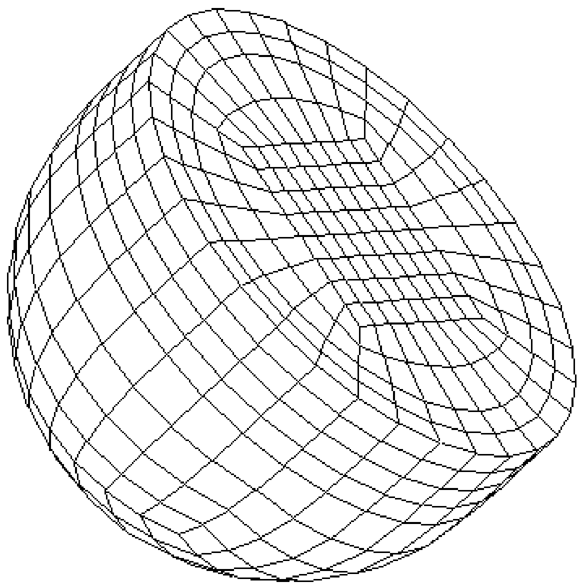


Figure 30. Cylinder with spherical cap

Input Deck

```
***********************************************************************
cylinder with spherical cap

sd 1 cyli 0 0 0 0 0 1 5.0     c sd 1 cylinder
sd 2 sphe 0 0 0 5.0           c sd 2 sphere

start

-1 5 -9;
-1 5 -9;
1 5 -9;

-1 0 1
-1 0 1
-5 0 1

sfi -1 -3; -1 -3; 1 2; sd 1   c projection onto cylinder
sfi -1 -3; -1 -3; 2 -3; sd 2  c projection onto sphere

di 1 2; 1 3; 1 3;             c delete half

end
end
***********************************************************************
```

20. Solid Cylinder with Sphere



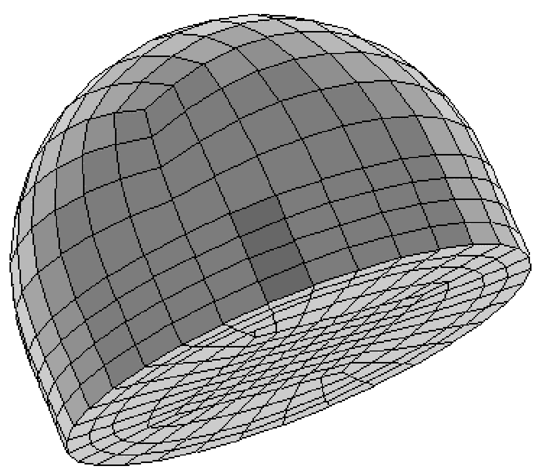


Figure 31. Solid cylinder with spherical cap

Input Deck

```
*************************************************************************
solid cylider with sphere

sd 1 cyli 0 0 0 0 0 1 5.0     c surface definitions
sd 2 cyli 0 0 0 0 0 1 4.0     c
sd 3 sphe 0 0 0 5.0           c
sd 4 sphe 0 0 0 4.0           c

start

1 3 5 9 13 15 17;             c indices
1 3 5 9 13 15 17;             c
1 5 9 13 17;                  c

-2 -2 -2 0 2 2 2              c three indices on
-2 -2 -2 0 2 2 2              c top of each
-2 0 2 2 2                    c other

di 1 3 0 5 7; 1 3 0 5 7; ;
di 1 3 0 5 7; ; 3 5;
di ; 1 3 0 5 7; 3 5;

sfi -1 -7; -1 -7; 1 2; sd 1   c projections
sfi -2 -6; -2 -6; 1 2; sd 2   c
sfi -1 -7; -1 -7; 2 -5; sd 3  c
sfi -2 -6; -2 -6; 2 -4; sd 4  c

end
end
*************************************************************************
```

21. Cylinder with Hole

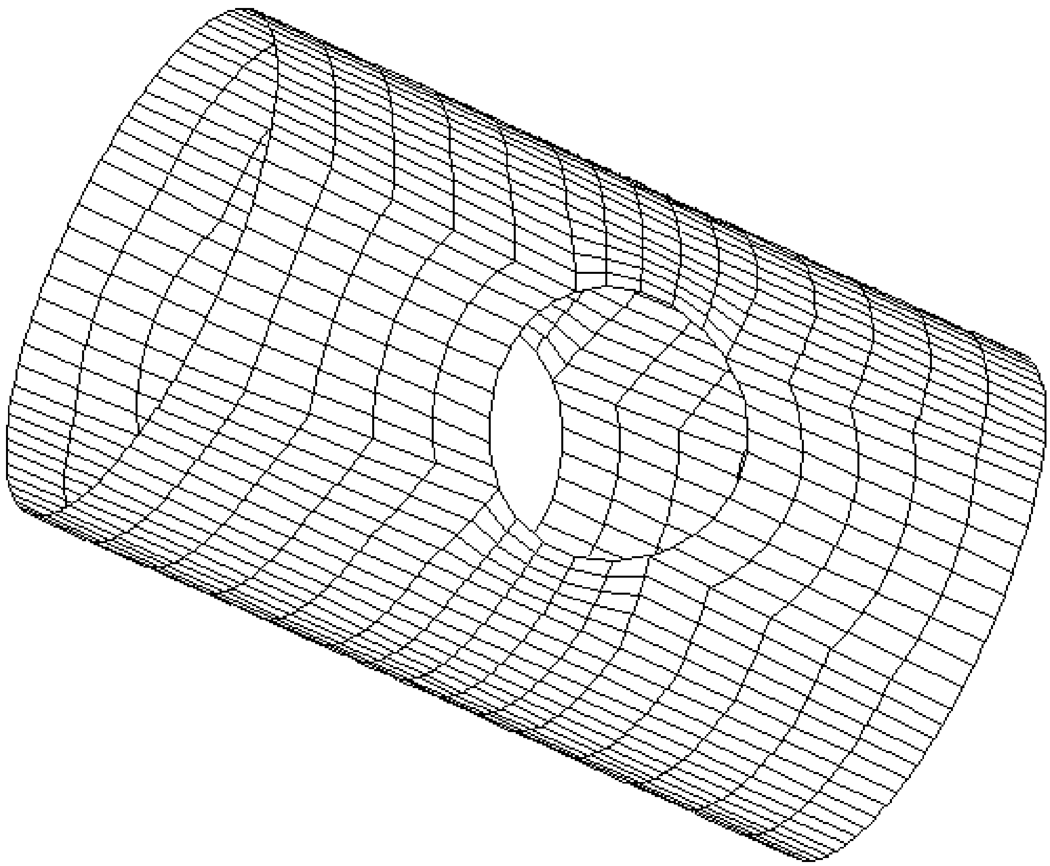


Figure 32. Cylinder with hole

Input Deck

```
*************************************************************************
cylinder with hole

sd 1 cyli 0 0 0 0 0 1 5.0
sd 2 cyli 0 0 0 0 1 0 2.5

start

-1 -21;
-1 6 16 -21;                  c indices 2 & 3 for the hole
1 6 11 16;                    c indices 2 & 3 for the hole
-2 2
-2 -1 1 2
-5 -2 2 5

di ; 2 3; 2 3;                c deletion of the hole

sfi -1 -2; -1 2 3 -4; ; sd 1
sfi -1 -2; 2 3; 2 3; sd 2

end
end
*************************************************************************
```

Comment

Four Regions are defined in the J-direction in order to generate a better mesh (Figure 32a).

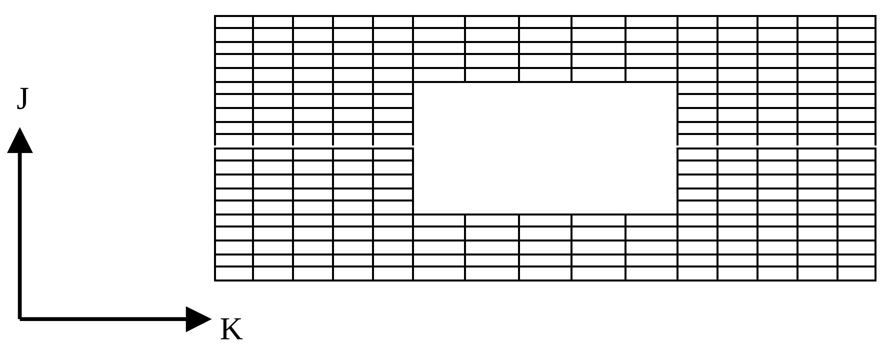


Figure 32a. Position of the deleted Region for the hole

22. Cylinder with Transition Hole

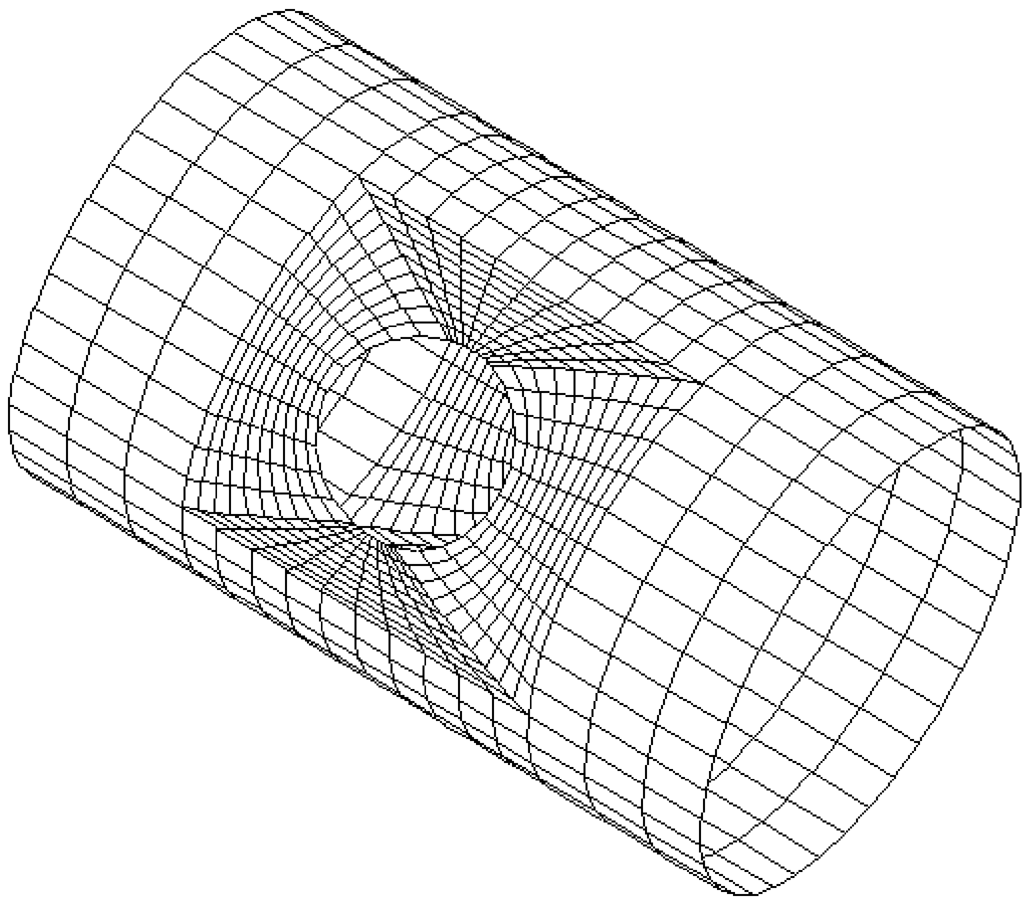


Figure 33. Cylinder with transition hole

Input Deck

```
**************************************************************************
cylinder with hole (transition region)

sd 1 cyli 0 0 0 0 0 1 5.0
sd 2 cyli 0 0 0 1 0 0 2.0
start                                c 1st part (cylinder)

-1 -11;
-1 -11;
8 11 21 24;
-1 1
-1 1
-5 -3 3 5                            c hole is –3 3

di -1 -2; ; 2 3;                     c deletion for the hole

sfi -1 -2; -1 -2; -1 -4; sd 1

end

start                                c 2nd part (transition
                                       region)
-1 -2;
1 11 21 31;
1 11 21 31;
-1 1
-1 -1 1 1                            c
-3 -3 3 3                            c exact same size as
                                       hole

di -1 -2; 1 2 0 3 4; 1 2 0 3 4;
d 0 2 2 0 3 3                        c deletion for the hole

sfi -1 -2; ; ; sd 1                  c projection onto sd 1
sfi ; 2 3; 2 3; sd 2                 c projection onto sd 2

end
end
***********************************************************************
```

Comment

Note the comments for the coordinates and see also example 17.

23. Cylinder with Cutout

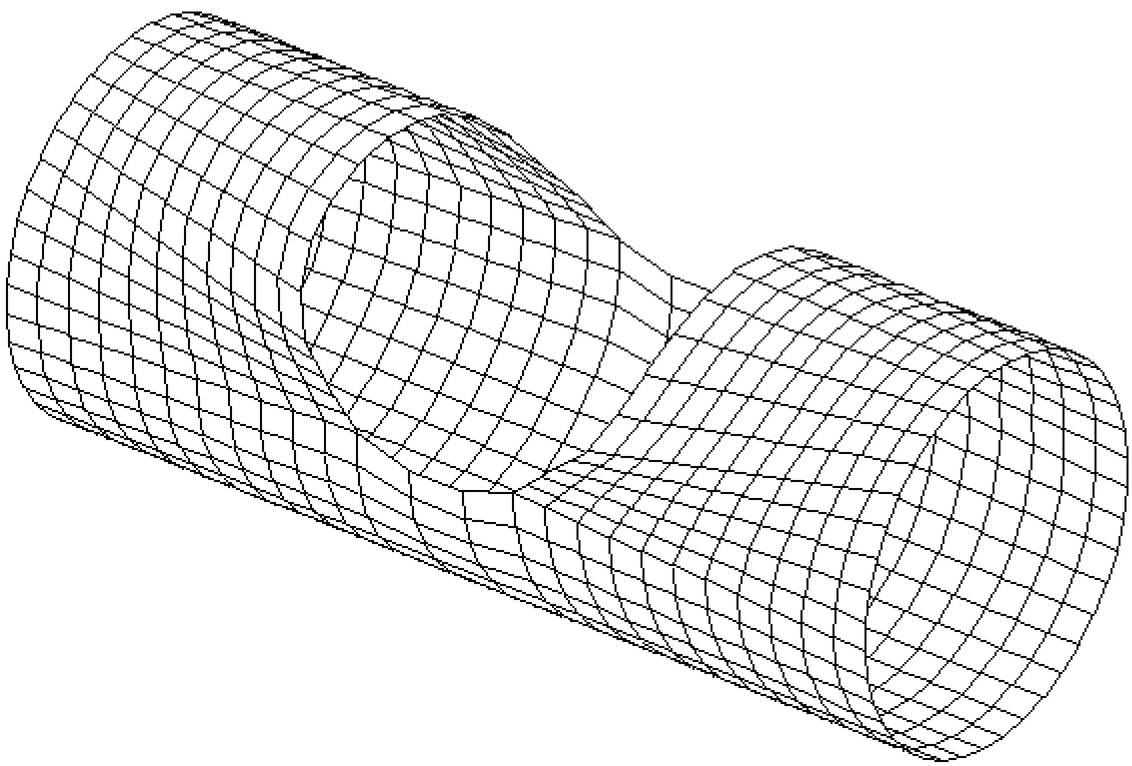


Figure 34. Cylinder with cutout

Input Deck

```
*************************************************************************
cylinder with cutout

sd 1 cyli 0 0 0 0 0 1 4.0
sd 2 cyli 0 4 0 1 0 0 4.0

start

-1 6 -11;
-1 6 -11;
1 12 15 26;

-1 0 1
-1 0 1
-10 -2 2 10

di 1 3; 2 3; 2 3;

sfi -1 -3; -1 -3; 1 4; sd 1
sfi –1 2 -3; -2 -3; 2 3; sd 2

end
end
*************************************************************************
```

Comment

Instead of writing `di 1 3; 2 3; 2 3;` the DI-command can also be specified as `di ; 2 3; 2 3;`. The blanket in the I-direction means that LS-INGRID is taking all Indices defined in that direction. This rule is valid for all commands requiring Reduced Index Space input, e.g. SFI, etc.

24. Pipe Bending with Hole

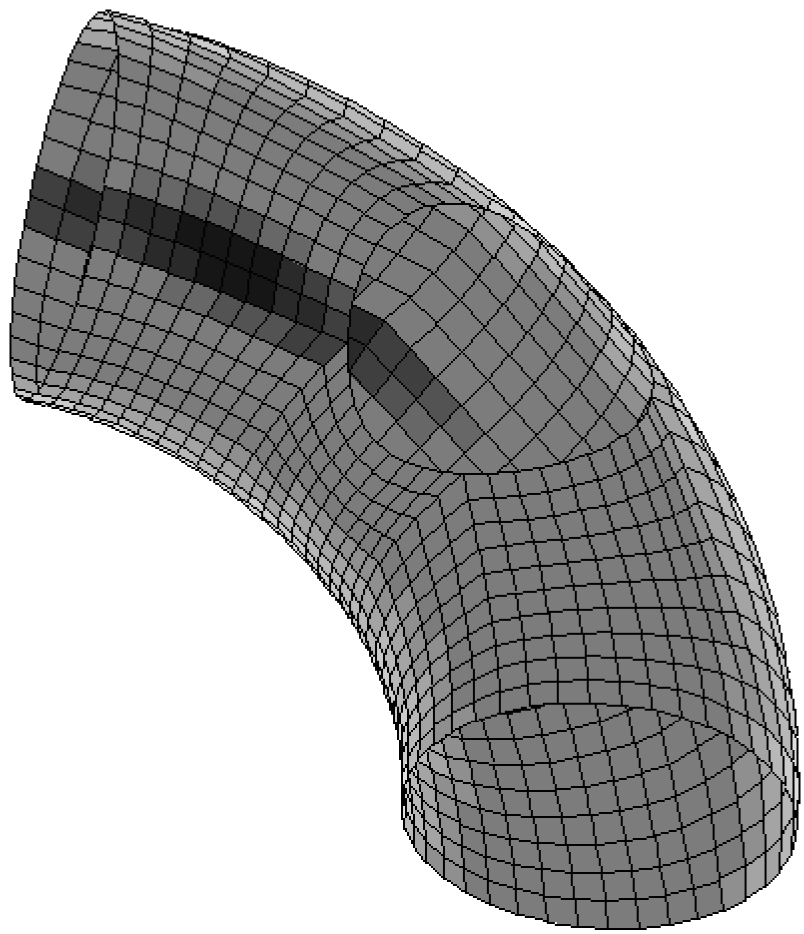


Figure 35. Pipe bending

Input Deck

```
***************************************************************************
pipe bending with hole

sd 1 ts 0 2.5 0 0 1 0 7.5 0 2.5    c sd 1 torus

sd 2 cyli 5.303 0 -5.303 0 1 0 2.0

start

-1 -11;
-1 -11;
1 11 21 31;

5 10
0 5
0 0 0 0                            c 4 regions on top of
                                     each other
rr 1 1 2 2 2 2 ry 30;              c rotate region 30°
rr 1 1 3 2 2 3 ry 60;              c      ″         60°
rr 1 1 4 2 2 4 ry 90;              c      ″         90°

sfi -1 -2; -1 -2; 1 4; sd 1        c projection onto torus

di -1 -2; -1; 2 3;                 c deletion for hole

sfi -1 -2; -1; -2 -3; sd 2         c projection onto
                                     cylinder

end
end
***************************************************************************
```

Comment

One might think that this model is simple to generate, however it requires a different way of creating the geometry. The problem is that the geometry of the Index Progression should come *as close as possible* to the Surface Definition. In this example the RR (rotate region)-command has to be used to get the required geometry as shown in Figure 35a.

RR <region> data

RR	rotate region
<region>	region in reduced index space
data	see section *Coordinate Transformation* *(LS-INGRID Commands and Reference Manual)*

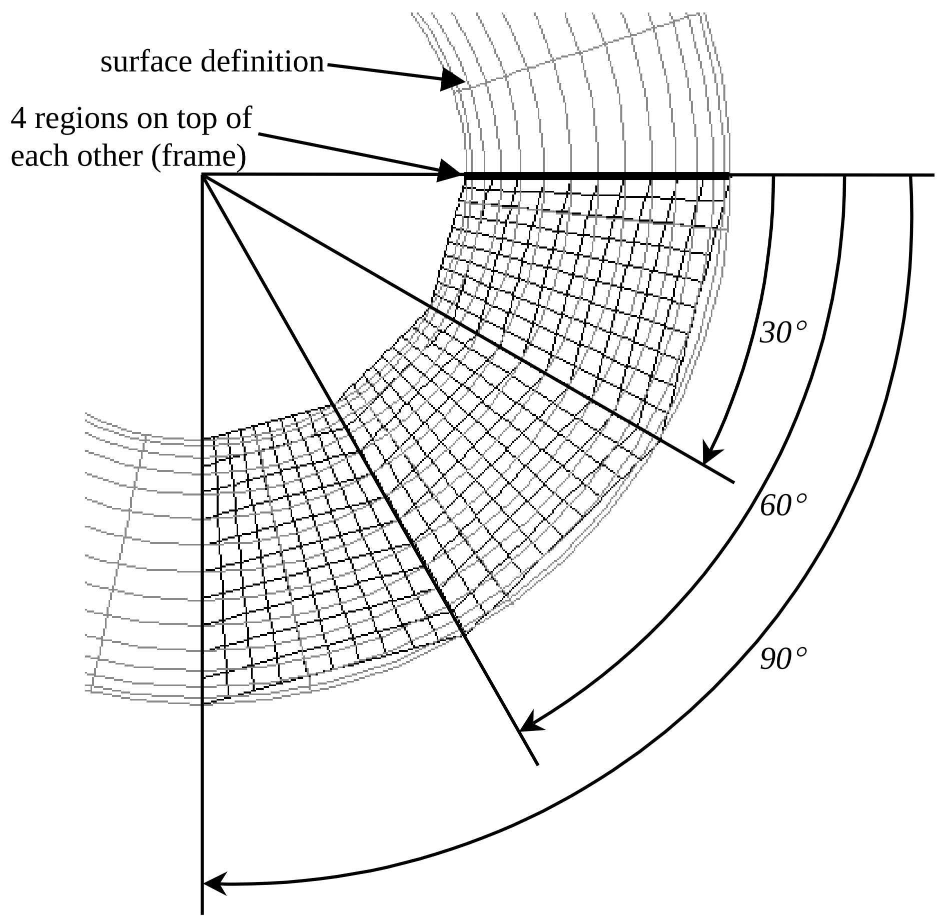
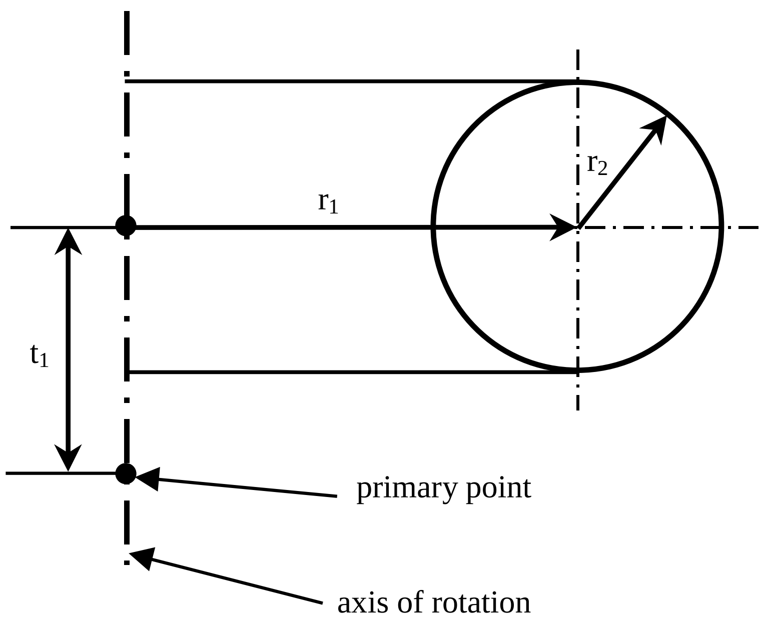


Figure 35a. Geometry created with RR-command and Surface Definition (torus)

TS $p_xp_yp_z$ $q_xq_yq_z$ r_1 t_1 r_2

TS	define a torus
$p_xp_yp_z$	primary point on the axis of rotation
$q_xq_yq_z$	orientation vector for axis of rotation
r_1	radius to the secondary axis
t_1	offset relative to the primary point on the axis of rotation
r_2	radius from the secondary axis to the torus surface



25. Circular Flange with Hole

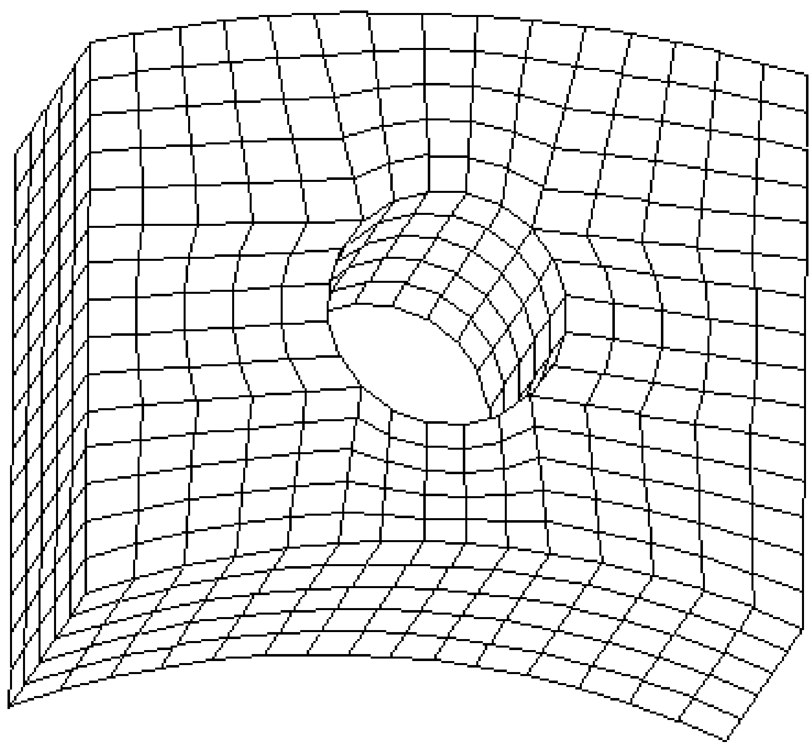


Figure 36. Circular flange with hole

Input Deck

```
*************************************************************************
circular flange with hole

sd 1 cyli 0 -9.5 0 0 0 1 12.0
sd 2 cyli 0 -10 0 0 0 1 8.0
sd 3 cyli 0 0 0 0 0 1 1.0
sd 5 plan -3 0 0 1 0 0          c surface definition
sd 6 plan 3 0 0 1 0 0           c of planes

start

1 6 11 16;
1 6 11 16;
1 6;

-1 -1 1 1                       c two indices on
-1 -1 1 1                       c top of each other
1 5

di 2 3; 2 3; 1 2;               c deletion for hole

sfi -1; 1 4; 1 2; sd 5          c projections
sfi -4; 1 4; 1 2; sd 6          c     ″
sfi -2 -3; -2 -3; 1 2; sd 3     c     ″
sfi 1 4; -1; 1 2; sd 2          c     ″
sfi 1 4; -4; 1 2; sd 1          c     ″

cpl 1 0 0 1 0 0 j +o i 3 j      c element spacing

end
end
*************************************************************************
```

Comment

Figure 36a shows the model without projections onto the planes (Surface Definitions 5 and 6). The Projection Method is searching for the closest point on the Surface Definitions (Cylinders with R12 and R8), therefore one have to force LS-INGRID to create a straight geometry on the side planes.

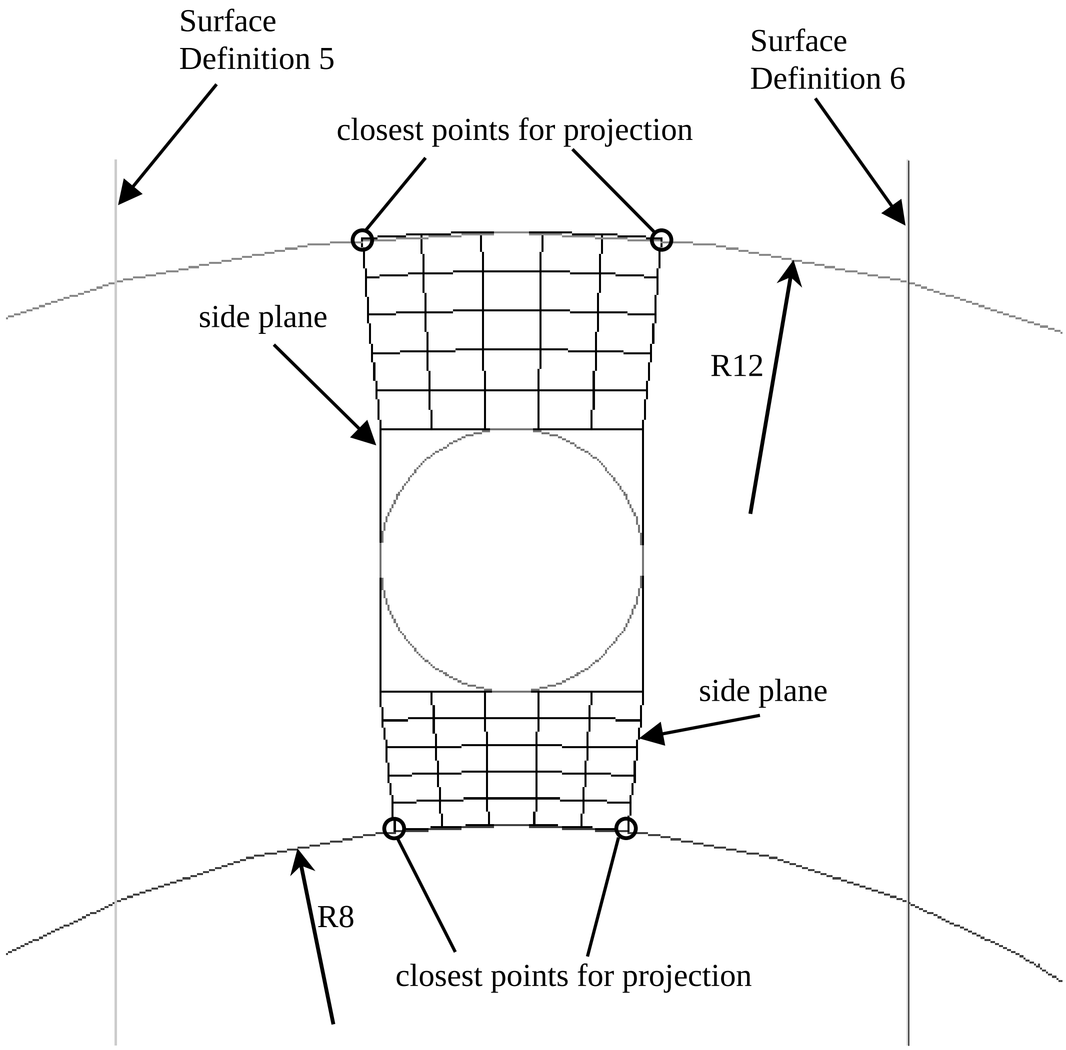


Figure 36a.

PLAN p_x p_y p_z v_x v_y v_z

Plan	define a plane
p_x p_y p_z	location of the plane
v_x v_y v_z	vector normal to the plane (no negative numbers)

The CPL-command defines equal spaced elements within lines, planes and volumes with the effect of an improvement of the mesh quality.

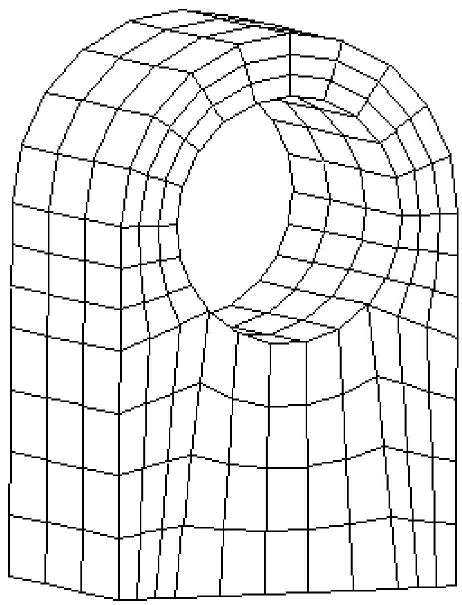
CPL <region> dir

CPL	center points along line
<region>	can be a line, plane or volume
dir	I, J or K-direction; do not input if <region> is a line

+ or +o dir inc

+ or +o	are offset commands; if + is used, as in this example, then the offset occurs from the last <region> defined. If +o is used then the offset is relative to the region defined by <region> or <reduced index space> (<index progression> in the LS-INGRID Manuals)
dir	dir is the increment direction, e.g. I, J, K, IJ, IK, IJK, SIJ, … SIJ means switch I with J indices
inc	inc is the increment for the indices (not for switches)

26. Bracket with Hole



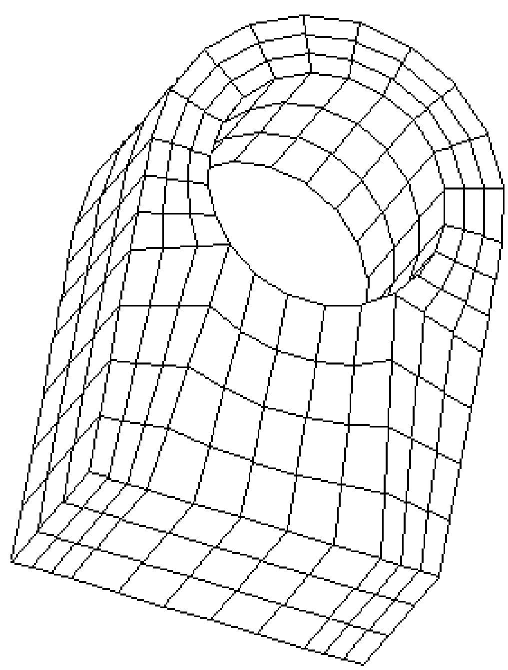


Figure 37. Bracket with hole

Input Deck

```
**************************************************************************
bracket with hole

sd 1 cyli 0 3 0 0 0 1 1.0
sd 2 cyli 0 3 0 0 0 1 1.5
sd 3 plan -1.5 0 0 1 0 0
sd 4 plan 1.5 0 0 1 0 0

start

1 4 9 12;
1 5 8 13;
1 4;

-1.5 -1.0 1.0 1.5
0 2 3 3
0 2

d 2 2 0 3 4 0

pb 1 4 1 4 4 2 xy  0 4      c very important movement!!

sfi -2 -3; -2 4; ; sd 1
sfi -1 -4; 3 4 ; ; sd 2
sf 1 1 0 1 3 0 sd 3
sf 4 1 0 4 3 0 sd 4

end
end
**************************************************************************
```

Comment

Figure 37a shows the definition of the model before the PB-command. Figure 37b is the result of the PB-command. Figure 37c is the result of the planes moving in the Y-direction. The gap shown in Figure 37d is produced because the shape of the Index Progression is not a closed structure.

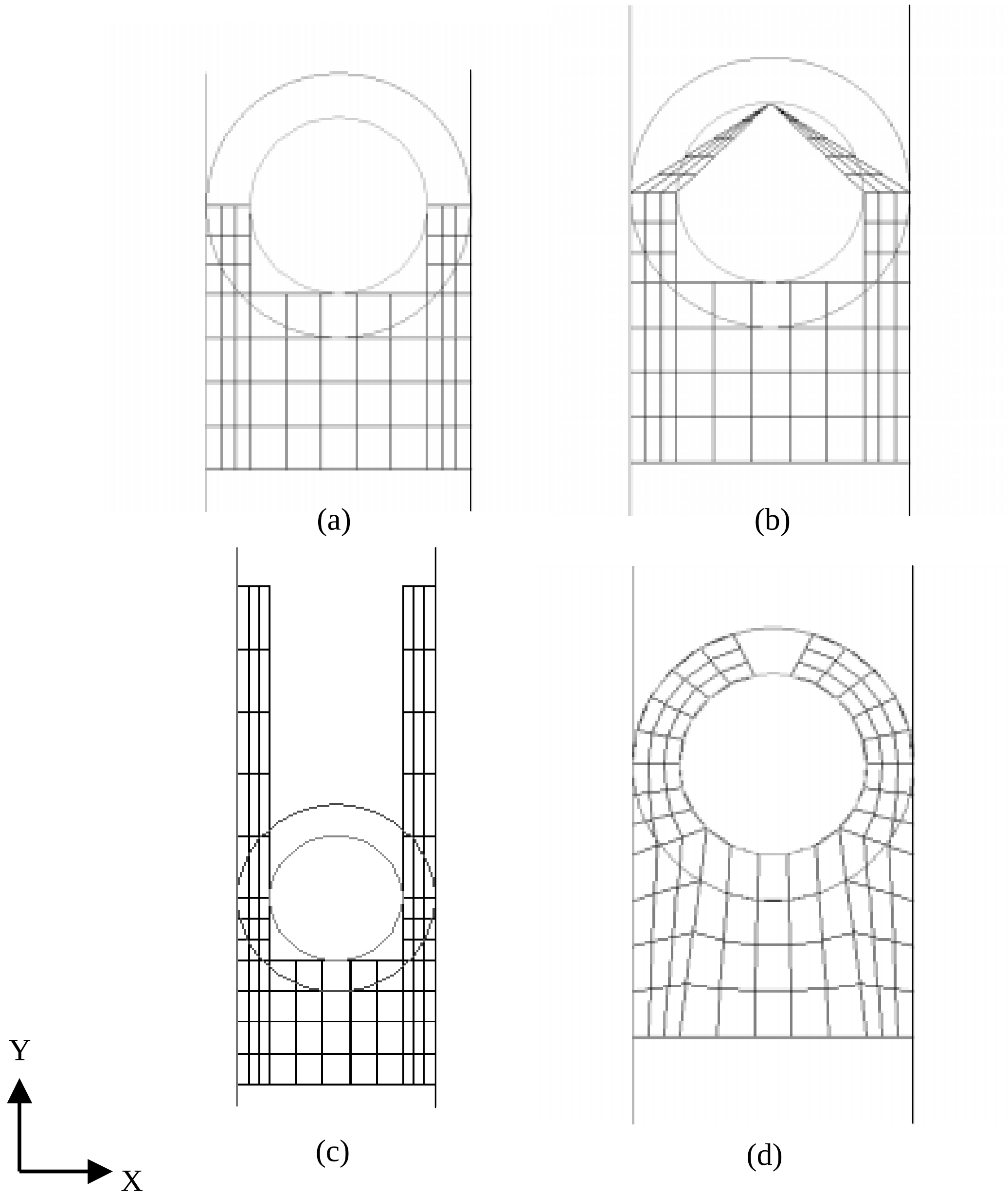


Figure 37.

27. Lower Die

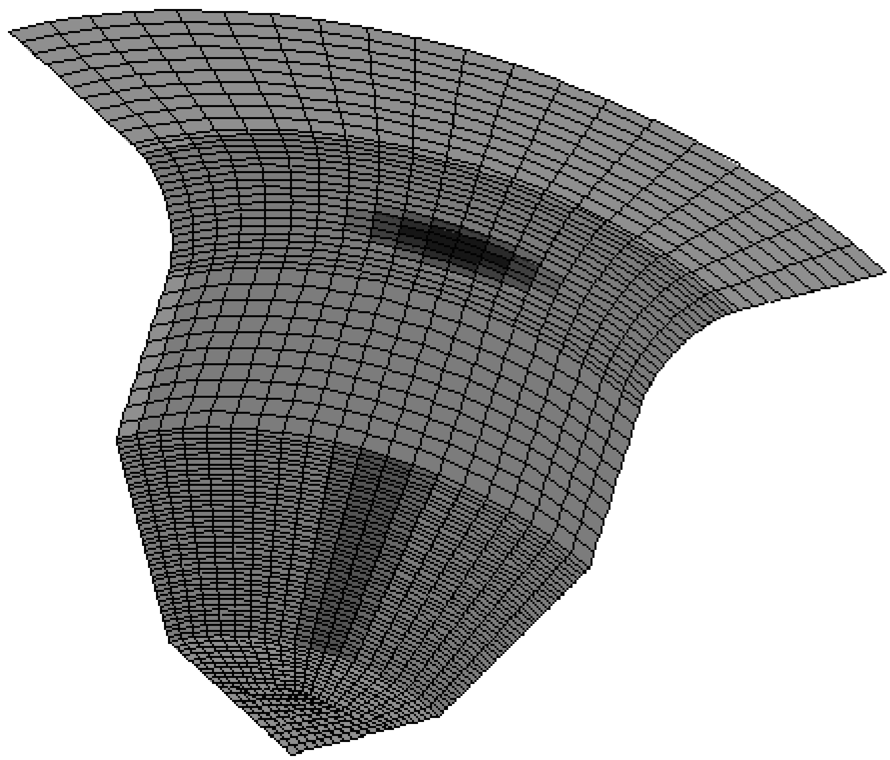


Figure 38. Lower die

Input Deck

```
****************************************************************************
lower die

c -------------------------
c --- LINE DEFINITION 1 ---
c -----------------------------------------------------------
-------
c --- INPUT OF THE REAL GEOMETRY IN THE  R-Z-COORDINATE
SYSTEM ---
c -----------------------------------------------------------
-------
ld 1 lp 3 10 10 17.5 17.5 17.5 27.5     c 3 points
     lar 22.5 32.5 –5                   c arc r=5
     lp 1 32.5 32.5                     c 1 point
c -----------------------------------------------------------
--------------
c --- SURFACE DEFINITION 1 ROTATE LINE DEFINITION 1 AROUND
THE Z-AXIS ---
c -----------------------------------------------------------
--------------
sd 1 crz 1                    c sd 1 rotate line definition
                                1 about Z-axis
start
-11 -31 41 51;
1 21;
1 -21 31 -41;
10 17.5 22.5 32.5
0 90
10 17.5 27.5 32.5

cyli                          c cylindrical coordinates !!!
di 2 4; 1 2; -1 -2;           c deleting unnecessary
                                regions
di 1 2; 1 2; -1 -4;
di -1; 1 2; 2 4;
di -2; 1 2; 1 2;

sfi -1 -2 -3 4  ; 1 2; 1 -2 3 -4; sd 1
end
```

```
c ------------------------------------------
c --- PART 2 (BOTTOM OF THE LOWER DIE) ---
c -----------------------------------------------------------
----------------
c --- SURFACE DEFINITION 2 AND 3 CREATE 2 CYLINDERS WITH
RADII 10 AND 6 ---
c -----------------------------------------------------------
----------------
c --- AND ROTATE THEM AROUND Z-AXIS
c -----------------------------------
sd 2 cyli 0 0 0 0 0 1 10.0
sd 3 cyli 0 0 0 0 0 1 [10.0*3/5.0]

start
1 11 21 31 41 ;
1 11 21 31 41 ;
-1;

-1 -1 0 1 1
-1 -1 0 1 1
10.0


sfi -1 -5 ; -1 -5 ; ; sd 2
sfvi -2 -4 ; -2 -4 ; ; sd 3   c vertices within <region> is
                                 projected

di 1 2 0 4 5 ; 1 2 0 4 5 ; ;
di 1 3 ; 1 5 ; ;
di 3 5 ; 1 3 ; ;
end
end
***********************************************************
```

Comment

The Surface Definition of Part 1 is created by a 2-dimensional Line Definition (LD) rotated about the Z-axis (SD 1 CRZ 1).

LD n options

LD	2-dimensional line definition
n	line definition number
options	several possibilities to create a line

For example:

```
ld 1 lp 3 10 10 17.5 17.5 17.5 27.5
     lar 22.5 32.5 -5
     lp 1 32.5 32.5
```

LP n $r_n z_n$

LP n	line consists of n points in the X-Y (R-Z)-Coordinate System
$r_n z_n$	coordinates of the points

LAR $r_1 z_1$ r

LAR	defines a circular arc by the radius.
$r_1 z_1$	The arc begins with the last defined point (here with LP n) and ends in $r_1 z_1$.
r	radius r. If $r>0$ then the center point of the arc lies on the right side, if $r<0$ the center point of the arc lies on the left side

The line definition can be visualized in LS-INGRID with the LV- or LVI-command as shown in Figure 38a. The dashed are included to show the Index Progression scheme.

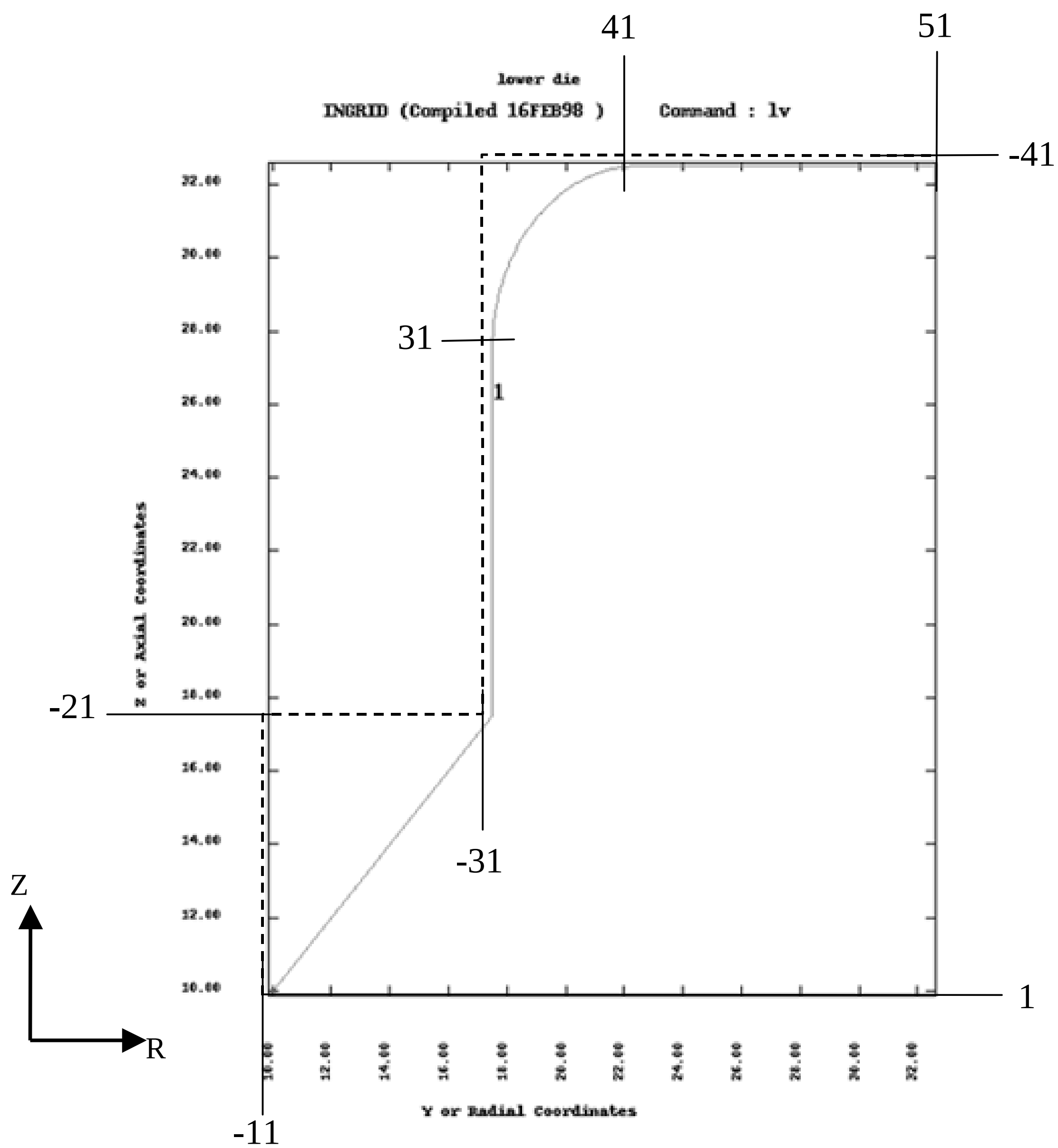
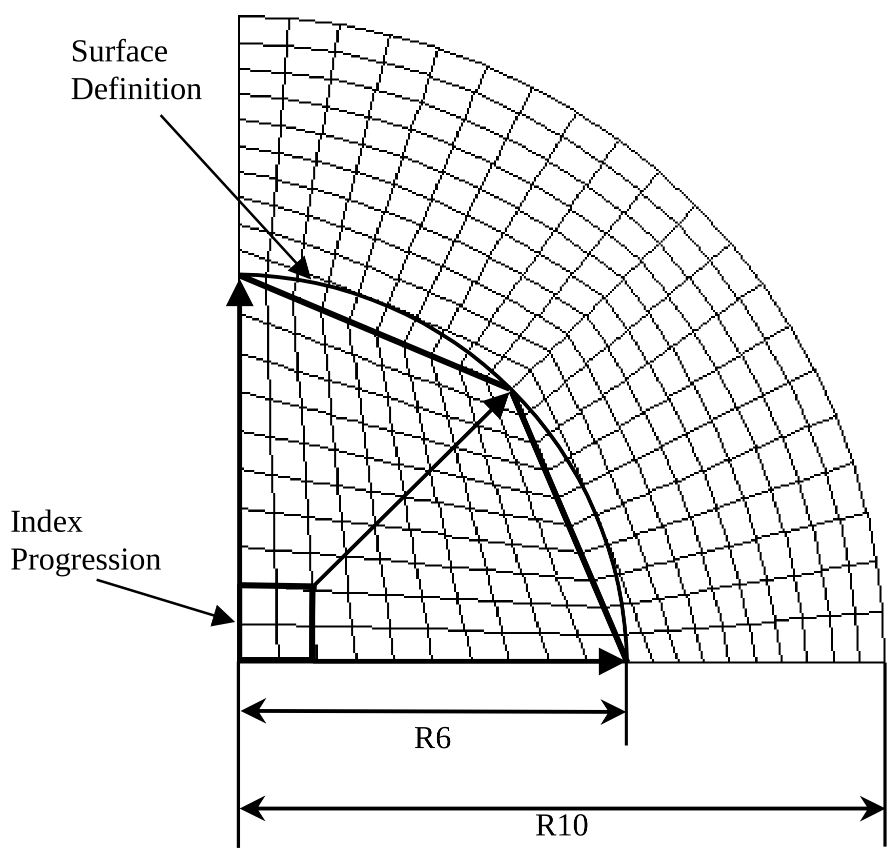


Figure 38a. Line definition and the Index Progression scheme (dashed lines)

Part 2 is shown in Figure 38b where the SFVI-command is used for the projection of the inner region. This produces a good mesh because only the vertices of the inner region were used for projection onto the closest point of the Surface Definition, as oppose using the nodes within the inner region as in the SF- and SFI-commands.

SFVI <index progression> itype
SFV <region> itype

SF	surface command, projects only vertices of blocks onto the closest point on a surface
<index progression>	index progression in reduced index space
<region>	region in reduced index space
itype	itype = sd n (surface definition n) itype ≠ sd n then put in an Surface Definition option



28. Screw with Nut

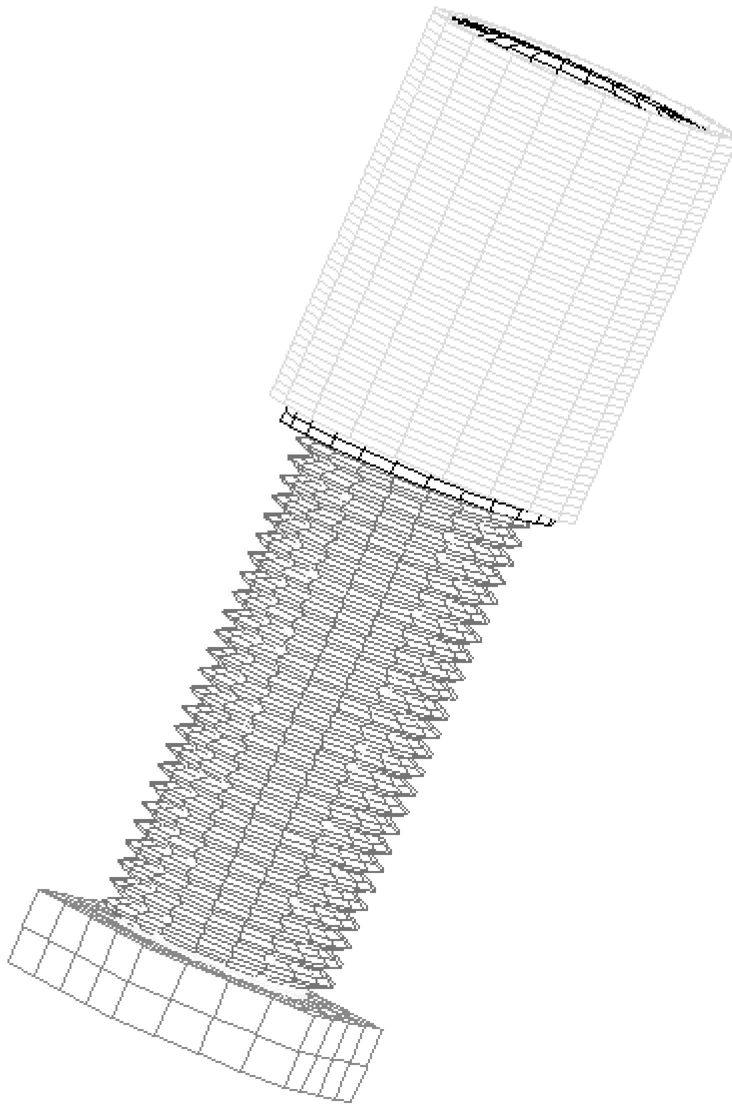


Figure 39. Screw with nut

Input Deck

```
*****************************************************************************
screw with nut

start                          c thread for nut
1 2 3;
1 501;
1 2 3;
0.255 0.33 0.35
0 7200
1.45 1.475 1.50

pa 1 0 2 x 0.305
ma 0 2 0 z 1.00
cyli
end

start                          c nut
1 3;
1 21;
1 51;
0.36 0.4
0 360
1.50 2.50
cyli
end

sd 1 cy 0 0 0 0 0 1 .245

start                          c head of screw
1 4;
1 5 9 13 17 21 25;
1 3;
25 0.45
0 0 0 0 0 0 0                  c 6 indices on top of each
                                 other !!
0 0.19
rr 0 2 0 0 2 0 rz 60;          c rotate <region>
rr 0 3 0 0 3 0 rz 120;         c       ″
rr 0 4 0 0 4 0 rz 180;         c       ″
rr 0 5 0 0 5 0 rz 240;         c       ″
rr 0 6 0 0 6 0 rz 300;         c       ″
rr 0 7 0 0 7 0 rz 360;         c       ″
```

```
sf 1 0 0 1 0 0 sd 1          c projection
end

start                        c bolt
1 11;
1 11;
1 3 10;
-1 1
-1 1
0 0.2 1.5
sfi -1 -2;-1 -2; ; sd 1      c projection

end

start                        c thread for bolt
1 2 3;
1 501;
1 2 3;
0.25 0.275 0.3
0 9000
0.20 0.225 0.25

d 2 0 2 3 0 3

pa 2 0 1 z 0.2125
pa 3 0 1 z 0.225
pa 2 0 2 x 0.27
pa 2 0 3 xz 0.275 0.2375
pa 3 0 2 xz 0.275 0.2375
ma 0 2 0 z 1.25

cyli
end
end
************************************************************
```

Comment

The threads are created with the same technique that was used to create the helix. The head of the screw is created with the RR-command as shown in Figure 39a.

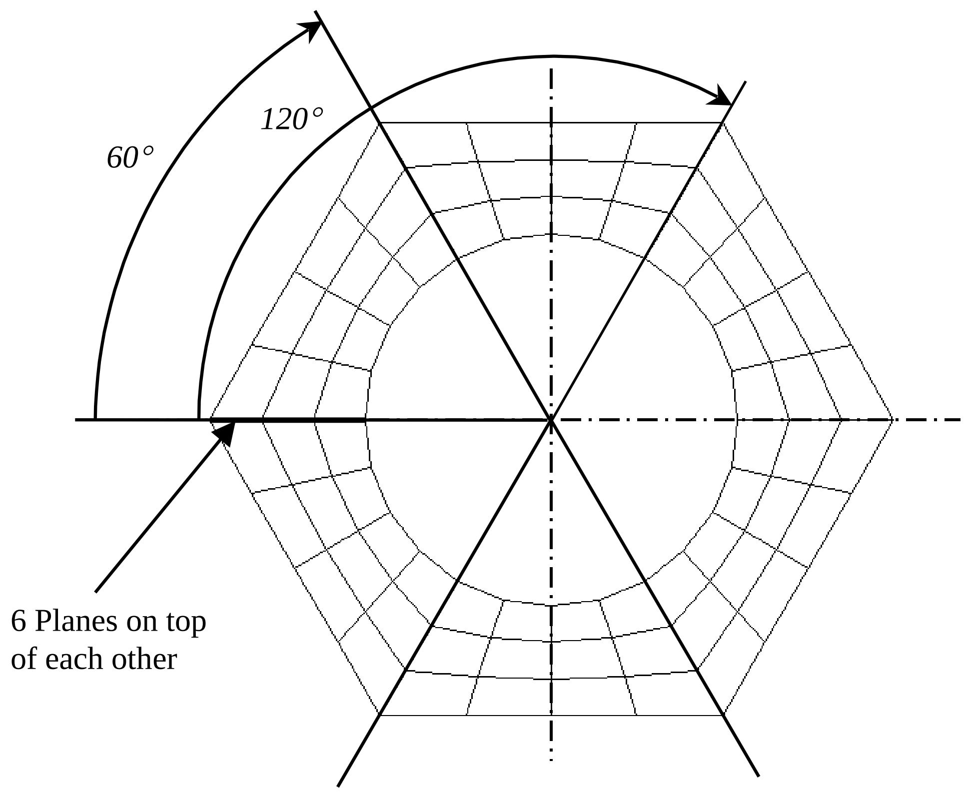


Figure 39a. Head of the screw

29. 2D->3D Mesh Generation[1]

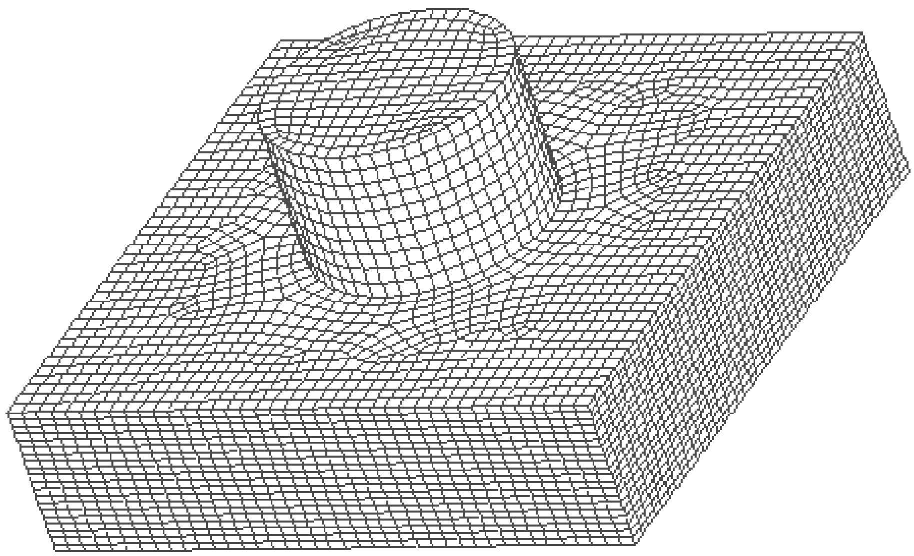


Figure 40. Simple model

[1] file is from Dr. Christine Espinosa @ DYNALIS, France

Input Deck

```
************************************************************************
DYNALIS: test utilisation des magic part

[ep=2]                           c parameters
[tm=0.1]
[diametre=1.6]
[distance=4]

ld 1 lp 2                        c line definition 1
[-distance/2] [-distance/2] [distance/2] [-distance/2]

ld 2 lp 2                        c line definition 2
[distance/2] [-distance/2] [distance/2] [distance/2]

ld 3 lp 2                        c line definition 3
[distance/2] [distance/2] [-distance/2] [distance/2]

ld 4 lp 2                        c line definition 4
[-distance/2] [distance/2] [-distance/2] [-distance/2]

lcc 1 0 0 -180 180 [diametre/2]  c define a arc

cont 1 -1 [tm] 5                 c automatic mesh command
cont 2 -4 [tm] 1 2 3 4           c     "       "      "

magic 1 1 2 y                    c start part 1
drag move [ep/tm] my [ep];;      c create solid elements
                                   with drag operation
end                              c end part 1

magic 2 2 1 2 y                  c start part 2
drag move [ep/tm] my [ep/2];;    c create solid elements
                                   with drag operation
end                              c end part 2

end
************************************************************************
```

Comment

4 Line Definitions (LD-command) and a Circle (LCC-command) are used in order to generate this example.

LCC n r_c z_c t_1 t_2 $r_1 \ldots r_n$

LCC	define n arcs
n	number of arcs
r_c z_c	center of arcs (R-Z-Coordinate System)
t_1 t_2	sweep angle from t_1 to t_2
$r_1 \ldots r_n$	radii of the next n arcs

The contours for the automatic mesh generation are specified with the CONT-command.

CONT c n s l_1 e_1 $r_1 \ldots$ l_n e_n r_n

CONT	define contour number c for automatic mesh generation
c	number of contour
n	number of lines ($l_1 \ldots l_n$) which represent the contour if n<0 then specify the global element size if n>0 then specify e_n
s	global element size (only if n<0)
$l_1 \ldots l_n$	number of the lines defined with LD, LCC, LEP, etc. LD is used for commands like LP, LAR, LAD, etc. which are dependent of the last point definition
$e_1 \ldots e_n$	if $e_n>0$ then e_n is the number of elements along the line l_n if $e_n<0$ then $\lvert e_n \rvert$ is the size of the elements along line l_n
$r_1 \ldots r_n$	element ratio, only if $l_n<0$ then r_n must be given

The MAGIC-command creates the Parts.

MAGIC n $c_1 \ldots c_n$ m k

MAGIC	create a part from n contours $c_1..c_n$
n	number of contours
$c_1..c_n$	contours $c_1 \ldots c_n$
m	material m
k	keep part y(es)/ n(o)

Note: "MAGIC"-Parts can not be discontinuous but can contain holes and inner unclosed contours to modify the mesh.

DRAG MOVE n data ;;

DRAG	perform a drag mesh operation to create solid elements from plane elements
MOVE	moving option
n	number of layers of solid elements to be formed by the moving option
data	new location, see section *Coordinate Transformation* in the *LS-INGRID Commands and Reference Manual*
;	end of data definition
;	end of drag definition

For more options for the DRAG-command see *LS-INGRID Commands and Reference Manual*.

30. Coordinate Transformation[1]

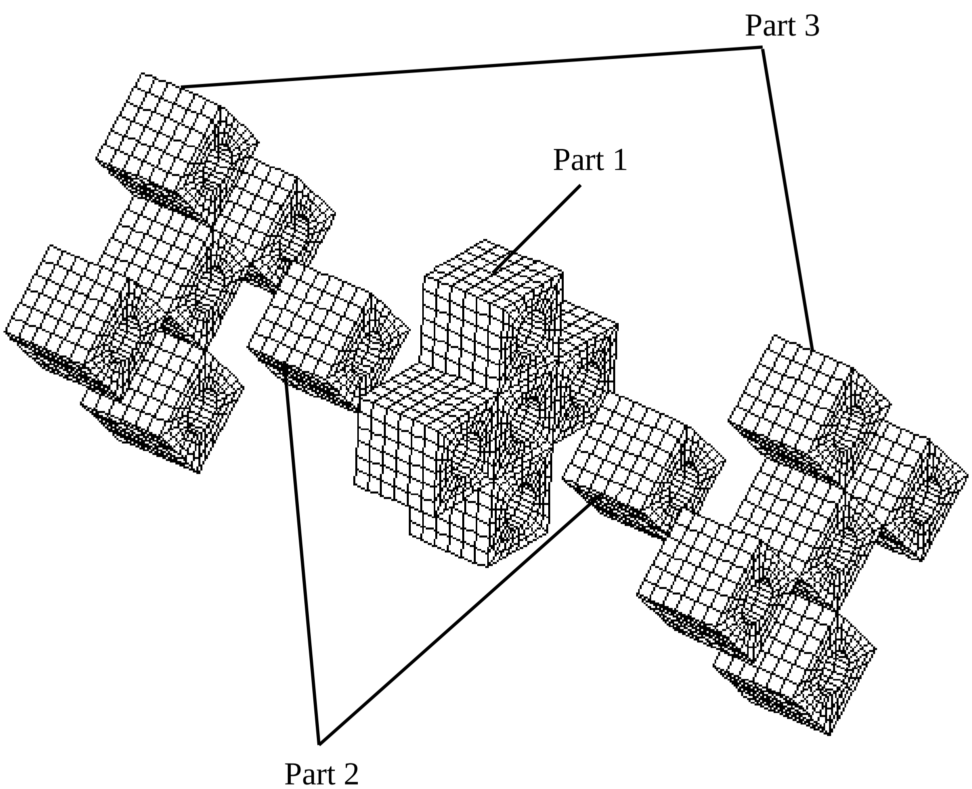


Figure 41. Coordinate Transformation example

[1] file is from Dr. Christine Espinosa @ DYNALIS, France

Input Deck

```
****************************************************************************
Dynalis: Exemple de transformation geometrique

[mx=6] [my=3] [mz=3]                    c parameters

coor 4 my 4; my -4; mz 4; mz -4;        c global coordinate
                                          transformations
start                                   c part 1
1 [1+mx];
1 [1+my] [1+2my] [1+3my] [1+4my];
1 [1+mz] [1+2mz] [1+3mz] [1+4mz];
-2 2
-2 -2 0 2 2
-2 -2 0 2 2

di ; 1 2 0 4 5; 1 2 0 4 5;
d 0 2 2 0 4 4                           c deletion for hole

a 0 2 2 0 4 4 i 1.0                     c projection for hole

repe 0 1 2 3 4;                         c copies into global
                                          coordinate system
end

start                                   c part 2
1 7;
1 4 7 10 13;
1 4 7 10 13;
-2 2
-2 -2 0 2 2
-2 -2 0 2 2
di ; 1 2 0 4 5; 1 2 0 4 5;
d 0 2 2 0 4 4                           c deletion for hole

a 0 2 2 0 4 4 i 1                       c projection for hole

coor 2 rx -45 mx -8; rx 45 mx 8;        c local coordinate
                                          transformations
lrep 1 2;                               c copies into local
                                          coordinate system
end
```

```
c part 3 is a combination of global and local coordinate
  transformation

coor 4 my [4*sqrt(2)];                 c global coordinate
 my [-4*sqrt(2)]; mz [4*sqrt(2)];        transformations
mz [-4*sqrt(2)];

start                                  c part 3
1 7;
1 4 7 10 13;
1 4 7 10 13;
-2 2
-2 -2 0 2 2
-2 -2 0 2 2

di ; 1 2 0 4 5; 1 2 0 4 5;
d 0 2 2 0 4 4

a 0 2 2 0 4 4 i 1

coor 2 rx -45 mx -16; rx 45 mx 16; c local coordinate
                                     transformations

lrep 1 2;                              c copies into local
                                         coordinate systems

repe 0 1 2 3 4;                        c copies into global
                                         coordinate systems

end

end
************************************************************
```

Comment

It is possible to perform Global and Local Coordinate Transformation as shown in this example.

COOR nc data

COOR	create global or local coordinate transformation
nc	number of coordinate transformations
data	data is described in section *Coordinate Transformation*

The Global Coordinate Transformation is always defined in the Control Commands section of an LS-INGRID input file whereas the Local Coordinate Transformation is defined within the Part Definition. Part 1 is created with a Global Coordinate Transformation.

For example:

```
coor 4 my 4; my -4; mz 4; mz -4;
```

There are 4 Global Coordinate Transformations defined in all. The first is an offset in the Y-direction about 4 units, the second in Y-direction about –4 units and the third and fourth in Z-direction about 4 and –4 units respectively. Every transformation is finished with a semicolon and represents a new Coordinate System. The REPE-command at the end of a Part definition (Part Control Commands) copies the Part into the new Global Coordinate Systems.

REPE l_1 … l_n;

REPE	repeat command, copies the Part into the Global Coordinate System l_n
l_1..l_n	number of Global Coordinate Systems defined in COOR
	if l_1=0 then the part is not transformed
;	end of repe definition

For example:

```
repe 0 1 2 3 4;
```

Part 1 is copied into the new Coordinate Systems l_1, l_2, l_3 and l_4, but it is also kept in its original Coordinate System.

For a Local Coordinate Transformation the REPE-command is replaced by the LREP-command.

LREP $l_1 \ldots l_n$;

LREP	copies the Part into the Local Coordinate System $l_1 \ldots l_n$
$l_1 \ldots l_n$	number of the local coordinate systems if l_1=0 no transformation (copy) of the part is performed
;	end of lrep definition

For example:

```
lrep 1 2;
```

Part 2 is transformed into the Local Coordinate Systems l_1 and l_2. Also possible is a combination of Local and Global Coordinate Transformation as shown in Part 3.
For a detailed description and further options see the *LS-INGRID Commands and Reference Manual.*

EXAMPLES FOR LS-DYNA

The following examples show the definition of Boundary Conditions, Contact, Load Curves, etc.

The default properties for a Part Definition are:

- 4-Node Shell Elements
- Shell Thickness = 0.0
- 8-Node Solid Elements
- Cartesian Coordinate System
- Material Properties = 0

It is possible to visualize several definitions in the Interactive Graphics-Phase. For details see the *LS-INGRID Graphical Interface User's Manual* and the *LS-INGRID Commands and Reference Manual,* Section *Interactive Commands*.

Some useful commands are:

DI CSYM *n*	display cyclic symmetry planes n
DI DX	display X-translational boundary conditions
DI DY	" Y-translational boundary conditions
DI DZ	" Z-translational boundary conditions
DI RX	" X-rotational boundary conditions
DI RY	" Y-rotational boundary conditions
DI RZ	" Z-rotational boundary conditions
DI SI *islid mslid*	display slide surface *islid* (contact definition) *mslid* = *m* master side *mslid* = *s* slave side *mslid* = *b* both sides
DI RBN	display nodal Rigid Bodies
M $n_1 \ldots n_n$;	display material $n_1 \ldots n_n$
P $n_1 \ldots n_n$;	display part $n_1 \ldots n_n$
LCV *n*	view loadcurve *n*
T *tol*	merge duplicate nodes within tolerance *tol*
TP *tol*	same as the T-command with printing the number of nodes merged together

1. Forging

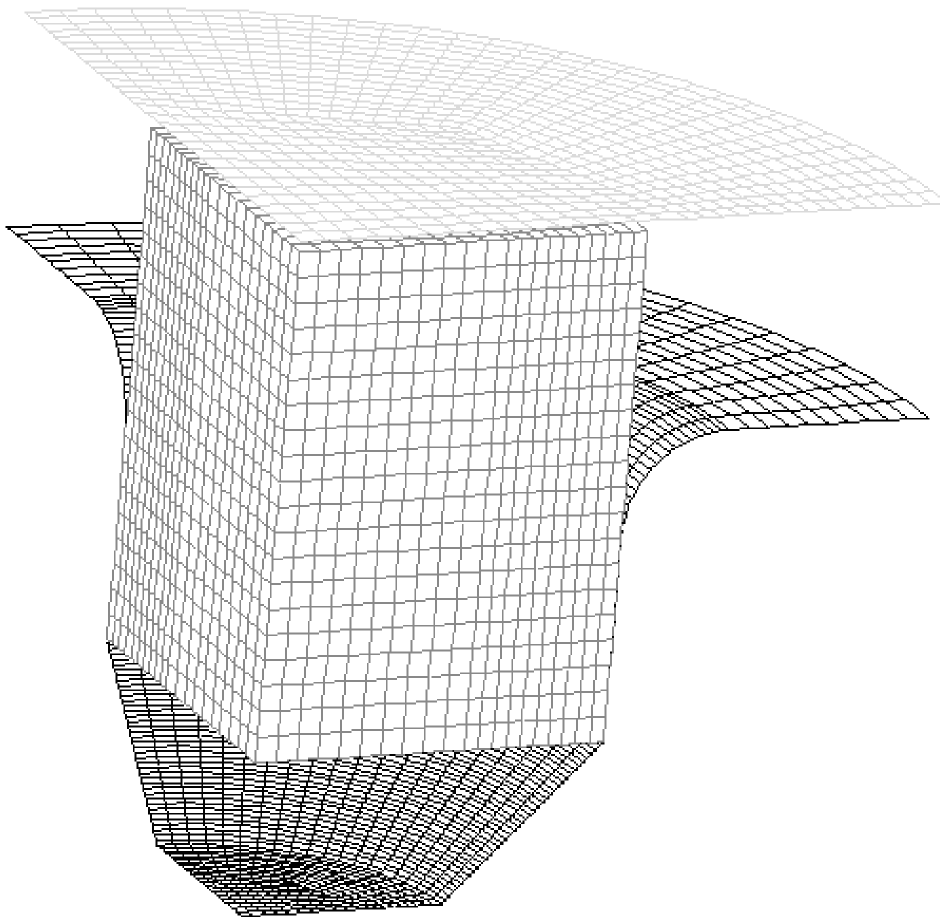


Figure 42. Simple forging model

Input Deck

```
************************************************************************
Header
c ----------------------------------------
c --- OUTPUT FORMAT LS-DYNA3D KEYWORD ---
c ----------------------------------------
dn3d kw93
c --------------------------------------------
c --- TERMINATION TIME OF THE CALCULATION ---
c --------------------------------------------
term 2.0
c --------------------------------------------------
c --- POST-PROCESSING (TAURUS) DATA OUTPUT TIME ---
c --------------------------------------------------
plti 0.01
c ----------------------------------------
c --- SPECIAL RESULTS IN ASCII-FORMAT ---
c ----------------------------------------
gmprt
matsum 0.01
rbout 0.01
secforc 0.01;
c ----------------------------
c --- MATERIAL DEFINITION ---
c ----------------------------
c ----------------------------------------------
c --- MATERIAL 1 AND 2 RIGID BODY (TYPE 20) ---
c ----------------------------------------------
mat 1 type 20 ro 7.85e-02 e 2.1e05 pr 0.3
              shell sform bely sthick 1.0
              endmat
mat 2 type 20 ro 7.85e-02 e 2.1e05 pr 0.3
              shell sform bely sthick 1.0
              endmat
c ----------------------------------------------------------
c --- MATERIAL 3 ELASTIC PLASTIC WITH FAILURE (TYPE 24) ---
c ----------------------------------------------------------
mat 3 type 24 ro 7.85e-02 e 0.7e05 pr 0.3
              sigy 80.0 etan 24.4 fail 4.0
              endmat
```

```
c ----------------------------------
c --- GLOBAL BOUNDARY CONDITIONS ---
c ----------------------------------
c --- SYMMETRY PLANES DEFINITION ---
c ----------------------------------
plane 2
0.0 0.0 0.0 0 -1 0 0.001 symm
0.0 0.0 0.0 -1 0 0 0.001 symm
c ------------------------------
c --- LOADCURVE 1 DEFINITION ---
c ------------------------------
lcd 1 2 0.0 0.0 2.0 7.0
c ---------------------------------------------------------
c --- RIGID BODY DISPLACEMENT BOUNDARY CONDITIONS MATERIAL
2 ---
c ---------------------------------------------------------
c --- WITH LOADCURVE 1 ---
c ------------------------
mdbc 2 1 1.0 0 0 -1 0.0 2.0
c ---------------------------------------------------------
c --- CONTACT DEFINITION CONTACT TYPE 14 SURFACE TO SURFACE
c ---------------------------------------------------------
si 1 t14 mate master 1,2 ; mate slave 3;
fric 0.1 pnlt 1.0;
c -----------------------
c --- BEGIN MODELLING ---
c -------------------------------------
c --- LOWER DIE CONSISTS OF 2 PARTS ---
c -------------------------------------
c --- PART 1 ---
c -------------------------
c --- LINE DEFINITION 1 ---
c ---------------------------------------------------------
-------
c --- INPUT OF THE REAL GEOMETRY IN THE R-Z-COORDINATE
SYSTEM ---
c ---------------------------------------------------------
-------
ld 1 lp 3 10 10 17.5 17.5 17.5 27.5
     lar 22.5 32.5 -5
     lp 1 32.5 32.5
```

```
c ------------------------------------------------------------
--------------
c --- SURFACE DEFINITION 1 ROTATE LINE DEFINITION 1 AROUND
THE Z-AXIS ---
c ------------------------------------------------------------
--------------
sd 1 crz 1
c ---------------------------
c --- INDEX PROGRESSION 1 ---
c ---------------------------
start
                         c -------------------
1 -11 -31 41 51;         c --- I-DIRECTION ---
1 21;                    c --- J-DIRECTION ---
-1 -21 31 -41;           c --- K-DIRECTION ---
                         c -------------------
c ---------------------------
c --- INITIAL COORDINATES ---
c ---------------------------
                         c -----------------------------
0 10 17.5 22.5 32.5      c --- (X-),     R-DIRECTION ---
0 90                     c --- (Y-), THETA-DIRECTION ---
10 17.5 27.5 32.5        c ---           Z-DIRECTION ---
                         c -----------------------------
c -----------------------------------------------
c --- CYLINDICAL COORDINATE SYSTEM FOR PART 1 ---
c -----------------------------------------------
cyli
c ----------------------------------------------
c --- DELETE UNNECESSARY BLOCKS AND SURFACES ---
c ----------------------------------------------
                         c --------------
di 3 5; 1 2; 1 2;        c --- BLOCKS ---
di 1 2; 1 2; -1 2 4;     c ----------------
di 2 3; 1 2; -4;         c --- SURFACES ---
di 2 3; 1 2; -1;         c ----------------
c ------------------------------------------------------------
---------
c --- PROJECTION OF THE INDEX PART ONTO THE SURFACE
DEFINITION 1 ---
c ------------------------------------------------------------
---------
sfi 1 -2 -3 4 5 ; 1 2;  -2 3 -4; sd 1
```

```
c -----------------------------------------------------------
----
c --- CONSTRAINS ON PART 1 FIXED IN ALL POSSIBLE DIRECTIONS
---
c -----------------------------------------------------------
----
b 1 1 1 5 2 4 111111
c -------------------------------------
c --- PART 1 CONSISTS OF MATERIAL 1 ---
c -------------------------------------
mate 1
c ---------------------
c --- END OF PART 1 ---
c ---------------------
end
c ----------------------------------------
c --- PART 2 (BOTTOM OF THE LOWER DIE) ---
c -----------------------------------------------------------
----------------
c --- SURFACE DEFINITION 2 AND 3 CREATE 2 CYLINDERS WITH
RADII 10 AND 6 ---
c -----------------------------------------------------------
----------------
c --- AND ROTATE THEM AROUND Z-AXIS
c ---------------------------------
sd 2 cyli 0 0 0 0 0 1 10.0
sd 3 cyli 0 0 0 0 0 1 [10.0*3/5.0]
c ---------------------------
c --- INDEX PROGRESSION 2 ---
c ---------------------------
start
                             c -------------------
1 11 21 31 41 ;              c --- I-DIRECTION ---
1 11 21 31 41 ;              c --- J-DIRECTION ---
-1;                          c --- K-DIRECTION ---
                             c -------------------
c ----------------------------
c --- INTITIAL COORDINATES ---
c ----------------------------
                             c -------------------
-1 -1 0 1 1                  c --- X-DIRECTION ---
-1 -1 0 1 1                  c --- Y-DIRECTION ---
10.0                         c --- Z-DIRECTION ---
                             c -------------------
```

```
c ---------------------------------
c --- DELETE UNNECESSARY BLOCK ---
c ---------------------------------
c   di 1 2 0 4 5 ; 1 2 0 4 5 ; ;
c ------------------------------------------------
c --- PROJECTIONS ONTO THE SURFACE DEFINITIONS ---
c ------------------------------------------------
sfi -1 -5 ; -1 -5 ; ; sd 2
sfvi -2 -4 ; -2 -4 ; ; sd 3
c ----------------------------------
c --- DELETE UNNECESSARY BLOCKS ---
c ----------------------------------
di 1 2 0 4 5 ; 1 2 0 4 5 ; ;
di 1 3 ; 1 5 ; ;
di 3 5 ; 1 3 ; ;
c ---------------------------------------------------------
-----
c --- CONSTRAINS ON PART 2: FIXED IN ALL POSSIBLE
DIRECTIONS ---
c ---------------------------------------------------------
-----
b 1 1 1 5 5 1 111111
c --------------------------------------
c --- PART 2 CONSISTS OF MATERIAL 1 ---
c --------------------------------------
mate 1
c ----------------------
c --- END OF PART 2 ---
c ----------------------
end
c ------------------------------------
c --- END OF LOWER DIE DEFINITION ---
c ------------------------------------
c -------------------------------------
c --- UPPER DIE CONSISTS OF 1 PART ---
c -------------------------------------
c --- PART 3 ---
c -----------------------------------
c --- SURFACE DEFINITION 4 AND 5 ---
c ------------------------------------------
c --- CYLINDER WITH RADII 32.5 AND 19.5 ---
c ------------------------------------------
sd 4 cyli 0 0 0 0 0 1 32.5
sd 5 cyli 0 0 0 0 0 1 [32.5*3/5.0]
```

```
c ------------------------
c --- INDEX PROGRESSION ---
c ------------------------
start
                  c ------------------
1 16 31 46 61 ;   c --- I-DIRECTION ---
1 16 31 46 61 ;   c --- J-DIRECTION ---
-1;               c --- K-DIRECTION ---
                  c ------------------
c --------------------------
c --- INITIAL COORDINATES ---
c --------------------------
                  c ------------------
-1 -1 0 1 1       c --- X-DIRECTION ---
-1 -1 0 1 1       c --- Y-DIRECTION ---
42.6              c --- Z-DIRECTION ---
                  c ------------------
c -------------------------------
c --- DELETE UNNECESSARY BLOCK ---
c -------------------------------
di 1 2 0 4 5 ; 1 2 0 4 5 ; ;
c -----------------------------------------------------------
--------------
c --- PROJECTION OF THE NECESSARY PLANES ONTO THE SURFACE
DEFINITIONS ---
c -----------------------------------------------------------
--------------
sfi -1 -5 ; -1 -5 ; ; sd 4
sfvi -2 -4 ; -2 -4 ; ; sd 5
c --------------------------------------------
c --- DELETE BLOCKS (WE NEED ONLY A QUARTER)---
c --------------------------------------------
di 1 3 ; 1 5 ; ;
di 3 5 ; 1 3 ; ;
c ------------------------------------
c --- PART 3 CONSISTS OF MATERIAL 2 ---
c ------------------------------------
mate 2
c -------------------------------
c --- END OF DEFINITION PART 3 ---
c ----------------------------------
c --- END OF UPPER DIE DEFINITION ---
c ----------------------------------
end
```

```
c --------------------------------------
c --- WORKPIECE CONSISTS OF 1 PART ---
c --------------------------------------
c --- PART 4 ---
c --------------------------------
c --- SURFACE DEFINITION 6,7 ---
c -------------------------------------------
c --- CYLINDER WITH RADII 17.49,10.494 ---
c -------------------------------------------
sd 6 cyli 0 0 0 0 0 1 17.49
sd 7 cyli 0 0 0 0 0 1 [17.49*3/5.0]
c -------------------------
c --- INDEX PROGRESSION ---
c -------------------------
start
                c -------------------
1 11 21 31 41;  c --- I-DIRECTION ---
1 11 21 31 41;  c --- J-DIRECTION ---
1 21;           c --- K-DIRECTION ---
                c -------------------
c ---------------------------
c --- INITIAL COORDINATES ---
c ---------------------------
                c -------------------
-1 -1 0 1 1     c --- X-DIRECTION ---
-1 -1 0 1 1     c --- Y-DIRECTION ---
17.500 42.500   c --- Z-DIRECTION ---
                c -------------------
c ---------------------------
c --- DELETE UNUSED BLOCK ---
c ---------------------------
di 1 2 0 4 5 ; 1 2 0 4 5 ; ;
c --------------------------------------------------------
c --- PROJECTION ONTO THE SURFACE DEFINITION 6 AND 7 ---
C --------------------------------------------------------
sfi -1 -5 ; -1 -5 ; ; sd 6
sfvi -2 -4 ; -2 -4 ; ;sd 7
c ---------------------------
c --- DELETE TO A QUARTER ---
c ---------------------------
di 1 3 ; 1 5 ; 1 2 ;
di 3 5 ; 1 3 ; 1 2 ;
```

```
c -------------------------------------
c --- PART 4 CONSISTS OF MATERIAL 3 ---
c -------------------------------------
mate 3
c ---------------------
c --- END OF PART 4 ---
c ---------------------
end
c ------------------------------
c --- END OF ALL DEFINITIONS ---
c ------------------------------
end
************************************************************************
```

Comment

This Model is created in the following order:

- *creating the lower die*
- *creating the upper die*
- *creating the workpiece*
- *material definitions*
- *loads and boundary conditions*
- *contact definition*
- *output format and general LS-DYNA output options*

Material Definition

Generally the code-dependent material is defined within the Control Commands section. Material type 20 (rigid body) is used for the dies and Material type 24 (elastic-plastic with failure) for the workpiece.

In general a material is defined with the following input data:

MAT n data ENDMAT

MAT	begin material definition
n	material number
data	code and material dependent input
ENDMAT	end of material definition n

For example:

```
mat 1 type 20 ro 7.85e-02 e 2.1e05 pr 0.3
              shell sform bely sthick 1.0
              endmat
```

Material 1 is defined as type 20 requiring the input of Density (ro), Young's Modulus (e), Poisson's Ratio (pr) and Bulk Modulus (k). Further it is defined as a shell (shell) with the shell formulation type Belytschko-Lin-Tsay (sform bely) and the thickness (sthick). The material definition is finished with the endmat-command.
For more LS-DYNA Materials see the *LS-INGRID Commands and Reference Manual*.

Loads and boundary conditions

Consider that the lower die is fixed. The movement of the upper die is defined with a displacement vs. time Load Curve. Symmetry-definitions are also necessary in order to calculate only a quarter of the model.

Symmetry is defined with the PLANE-command in the Control Commands section.
Note: The PLANE-command is used only once for the whole Model.

PLANE nplane data term

nplane	number of planes
data	p_x p_y p_z is any point on the plane
	q_x q_y q_z is the vector normal to the plane
	tolerance all nodes within a distance less than tolerance are included
term	termination input

For example:

```
plane 2
0.0 0.0 0.0 0 -1 0 0.001 symm
0.0 0.0 0.0 -1 0 0 0.001 symm
```

In this example there are 2 planes with the point (0,0,0) on each plane, the first normal vector is (0, -1, 0) and the second is (-1, 0, 0). Both have the tolerance 0.001 and apply symmertic boundary conditions to the nodes within the tolerance (symm-command, also the termination input for every plane definition n). Symmetry planes are displayed with the DI SY-command.

For applying *local boundary conditions* (degrees of freedom) the B-command is used within the part definition. This command is also valid for constraining rigid bodies.

B <region> code

<region>	all nodes within <region> are constrained with code
code	six digit binary number which affect from left to the right the following degrees of freedom: 0=free; 1=fixed 1st digit: x-displacement 2nd digit: y-displacement 3rd digit: z-displacement 4th digit: x-rotation 5th digit: y-rotation 6th digit: z-rotation

For example:

```
b 1 1 1 5 2 4 111111
```

The Region <1 1 1 5 2 4> is fixed in all possible directions. These definitions can be visualized with the DI DX, DI DY, etc.-commands.

All Load Curves are created with the LCD command.

LCD n m t_m f_m

lcd n	load curve definition n
m	number of m pairs of time function points
t_m	time of pair m
f_m	value of pair m

For example:

```
lcd 1 2 0.0 0.0 2.0 7.0
```

This is Load Curve definition 1 with 2 pairs of time function points. Visualizing is possible with the LCUR PLOT SELECT -commands input.

The commands to assign a Load Curve to a material boundary condition are material-dependent. All non-rigid bodies require the F**-commands set within the Part Definition (see section *Loads and Boundary Conditions* in the *LS-INGRID Commands and Reference Manual).*
Rigid bodies (material type20) require the MDBC (Material Displacement Boundary Condition)- and the MVBC (Material Velocity Boundary Condition)-commands set in the Control Commands section. MDBC is used in this example.

MDBC m lc amp f_x f_y f_z bt dt

mdbc	material displacement boundary condition
m	material number
lc	load curve number
amp	scale factor
$f_x f_y f_z$	load direction
bt dt	birth-and death-time

For example:

```
mdbc 2 1 1.0 0 0 -1 0.0 2.0
```

Load Curve 1 is assigned to material 2 with the scale factor 1.0 and its loaded is the negative Z-direction (0 0 –1) with birth-time 0.0 and death-time 2.0.

Contact definition

There are three ways of defining contact surfaces in LS-INGRID. The first is to define them by materials, as in this example, the second by specifying the master and the slave side by regions and the third to define contact surfaces within boxes.
They are all applied by the SI (Sliding Interface)-command.

SI islide options

SI	sliding interface definition
islide	number of SI definition
options	contact type, etc.

For example:

```
si 1 t14 mate master 1,2 ; mate slave 3;
fric 0.1 pnlt 1.0;
```

This is Sliding Interface definition 1 with Contact Type t14 (Automatic Surface to Surface). The Master side is defined by Material 1 and 2, the Slave side by Material 3. Furthermore Static and Dynamic Friction (fric) are 0.1 and the Penalty Factor is 1.

For further contact options and definitions see the *LS-INGRID Commands and Reference Manual* and the *LS-DYNA Manuals*. For manipulating and visualizing see LS-INGRID *Graphics Interface User's*-Manual.

Output format and general LS-DYNA options

The output formats in LS-INGRID include all LS-DYNA versions, the LLNL DYNA3D, Patran Neutral File, etc. (see *LS-INGRID Commands and Reference Manual*). LS-DYNA3D Keyword format is chosen with the following input

DN3D
KW93

and PLTI *t*ime for the d3plot-files writing interval for postprocessing

PLTI 0.01

Specific ASCII outputs like Material Energies, etc. are defined with the GMPRT-command.

GMPRT	c general printing options intervals
MATSUM 0.01	c material energies
RBOUT 0.01	c rigid body acceleration output
SECFORC 0.01**;**	c section forces

and the Termination Time for the calculation

TERM 2.0

Further LS-DYNA output options are available such as Hourglassing, etc. For details see the *LS-INGRID Commands and Reference Manual (section LS-DYNA Commands and Materials)* and the *LS-DYNA Manual*.

2. Cyclic Symmetry

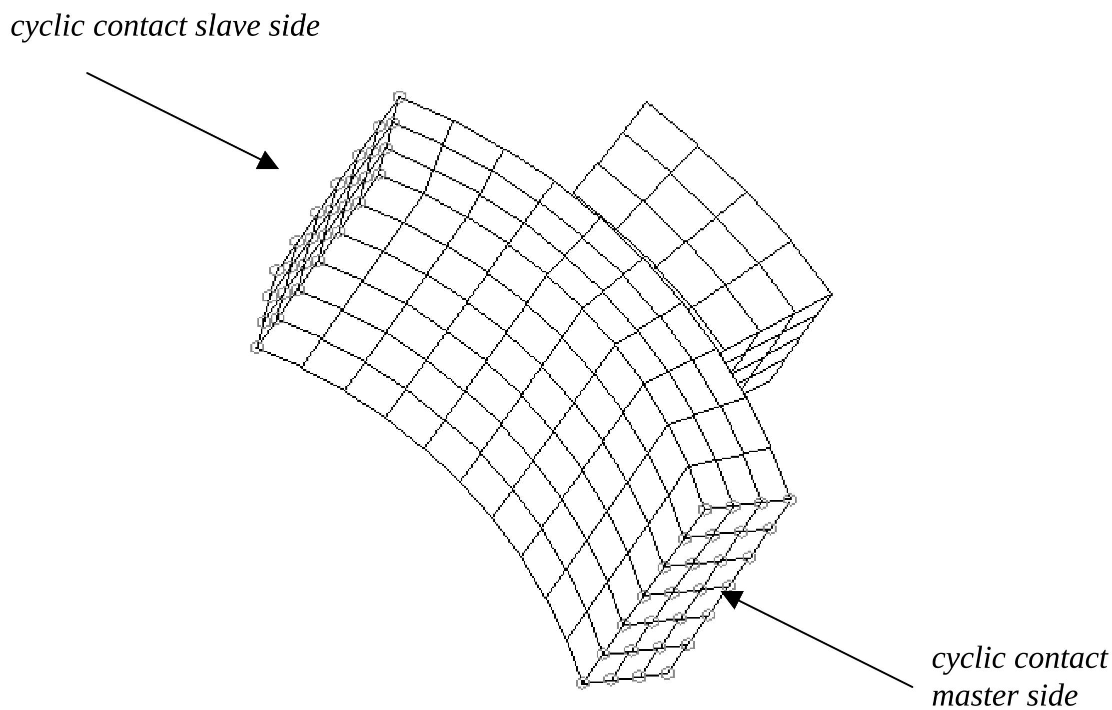


Figure 42. Model

Input Deck

```
************************************************************************
DYN27.ING: Cyclic Symmetry Test Case

dn3d v92 term 100.0 plti 1.0  c dyna3d options

mat 1 type 3 e 2 pr 0.3 ro 7.85 sigy 2e-3 etan 0.0 endmat
mat 2 type 3 e 2 pr 0.3 ro 7.85 sigy 2e-3 etan 0.0 endmat

csym axis 0 0 1 ;         c define cyclic symmetry axis
si 1 sv ;                 c contact definition 1 sliding
                            with voids (default)
start
 1 4 ;
 1 11 ;
 1 7 ;
 2.5 3.0
 0.0 60.0
 -1.0 1.0
csy 0 1 0 0 1 0  s             c cyclic symmetry contact
                                  side (slave side)
csy 0 2 0 0 2 0  m             c  ″ (master side)
si  2 0 0 2 0 0  1 m           c contact master side
cyli                           c cylindrical coordinates
mate 1                         c material 1
end
[vv=-150.0*100/1e6]            c calculation of variable vv
start
 1 5 ;
 1 5 ;
 1 5 ;
 3.1 4.1
 20.0 40.0
 -0.5 0.5
velo [vv*0.866] [vv*0.500] 0.0 c initial velocity vector
                                 (vxvyvz)
si  1 0 0 1 0 0  1 s           c contact slave side
cyli                           c cylindrical coordinates

mate2                          c material 2

end
end
************************************************************************
```

Comment

Loads and boundary conditions

The Cyclic Symmetry definition is similar to the Contact Definition. It requires the definition of Contact Nodes within the Part definition and also the definition of the Cyclic Symmetry Axis (Control Commands).

CSYM $p_x p_y p_z$

csym	cyclic symmetry interface
$p_x p_y p_z$	axis of rotation defined with the orientation vector $\vec{p}$ ($p_x p_y p_z$)

CSY <region> side
CSYI <index progression> side

csy	cyclic interface nodes
side	side is either Slave or Master

Contact definition

The contact between surfaces is defined with the SI-command explained in the previous example before with the difference that Slave and Master sides are defined within the Part Definition.

SI <region> islid mslid
SII <index progression> islid mslid

si	sliding interface definition
islid	sliding interface number
mslid	master/slave flag; M = Master, S = Slave

Output format and general LS-DYNA options

The output format in this example is `DN3D V92`, i.e. LS-DYNA3D Version 920 structured format. The Termination time is `term 100.0` and the d3plot-writing time interval for postprocessing is `plti 1.0`.

INDEX