Chapter - 3

Geometric Data Extraction from STEP Files (GDE)

CHAPTER - III

GEOMETRIC DATA EXTRACTION FROM STEP FILES (GDE)

There are different neutral files available in CAD software. Some of the neutral files are IGES, STEP, DXF, STL files etc. STEP and IGES are most popular. STEP is intended for product data exchange, whereas IGES is for geometry data exchange. STEP is worldwide recognized neutral file format of almost all commercial CAD software.

ISO 10303 is an ISO standard for the computer interpretable representation and exchange of product manufacturing information. Its official title is: Automation systems and integration - Product data representation and exchange. It is known informally as "STEP", which stands for **"STandard for the Exchange of Product model data"**. The description of product data for mechanical parts has been standardized by ISO10303 and different protocols are available in STEP. Some of the Application Protocols are AP203, AP214, AP224 etc used for different applications. AP203 is Configuration controlled 3D designs of mechanical parts and assemblies. ISO-10303-21 is the beginning keyword of STEP file and END-ISO-10303-21 is the ending keyword of the STEP file. It is based on the B-Rep.

3.1 HIERARCHY OF GEOMETRICAL STRINGS

Initially, design is drawn in CATIA and converted into a STEP file. The extension of the STEP file is .stp. This file is given as input to the developed program. The developed feature recognition program starts searching the STEP file with a string CLOSED_SHELL and it ends at a string CARTESIAN_POINT. In between various strings such as ADVANCED_FACE, FACE_OUTER_BOUND etc. are searched in a hierarchal manner. FACE_BOUND word will present if there is a presence of internal depression. The hierarchy of geometrical strings is shown in Figure 3.1. There is a duplication of data in STEP file. Entity is a number which

defines the geometry string in STEP file. In line "#35=CLOSED_SHELL('Closed Shell',(#75,#92,#123,#135,#166,#178,#218,#235,#249,#263));" the geometrical string is CLOSED_SHELL and #35 is entity number.

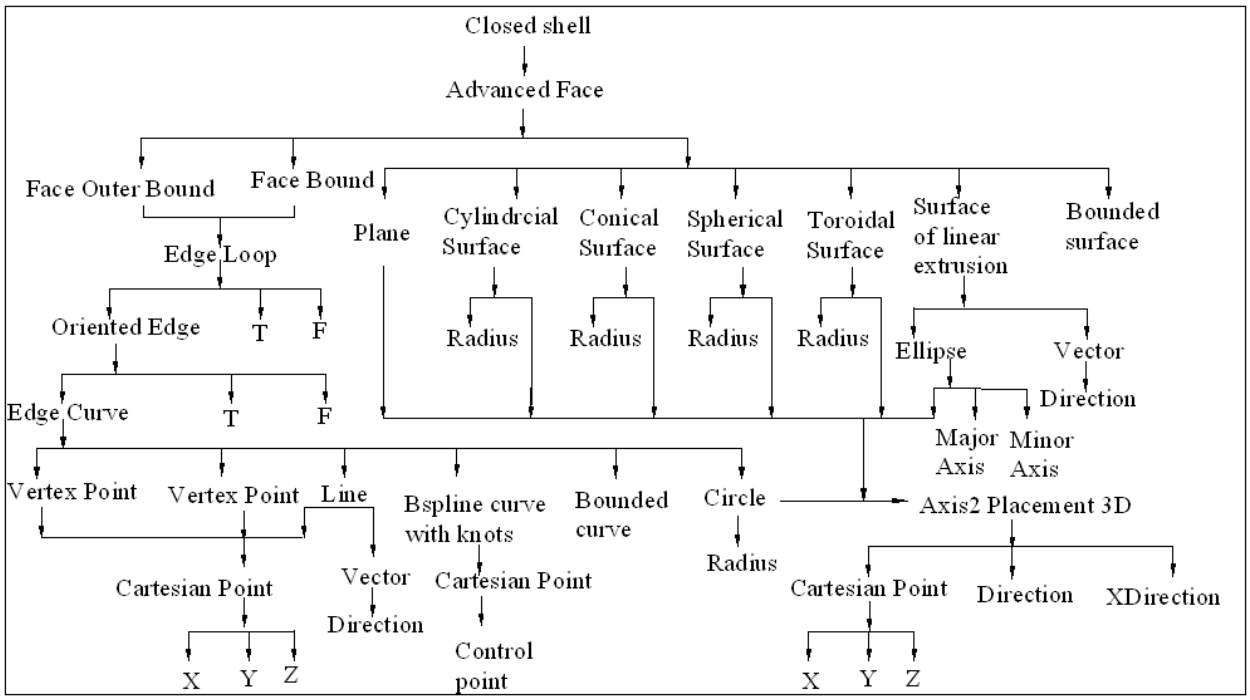

Fig. 3.1: Hierarchy of geometrical strings

3.2 GEOMETRIC DATA EXTRACTION FROM STEP FILES

The input for feature recognition is extracted geometrical data from STEP files. STEP files are saved in a text based ASCII format. Every line starts with # (hash) and followed by a positive number, this is called entity. Every line ends with a semicolon (;). Extraction of data starts searching various strings and placing corresponding entities in arrays as shown in Table 3.1. Place all these arrays in different tables as shown in Table 3.2.

Table 3.1: Searching strings in STEP file and placing in arrays

<table>
<tr><th>String name</th><th>Entity corresponding to string</th><th>Array name</th></tr>
<tr><td>CLOSED_SHELL</td><td colspan="2">ADVANCED_FACE</td><td>A_F</td></tr>
<tr><td>ADVANCED_FACE</td><td colspan="2">1. FACE_OUTER_BOUND
2. FACE_BOUND
3. Find the surface
Surface name | String in table
CYLINDRICAL_SURFACE | cylinder
CONICAL_SURFACE | conical
TOROIDAL_SURFACE | toroidal
SPHERICAL_SURFACE | spherical
PLANE | plane
BOUNDED_SURFACE | bsurface
SURFACE_OF_LINEAR_ EXTRUSION | ellipse</td><td>1. FOB
2. F_B
3. cy_str (string in table) and cysu</td></tr>
<tr><td>CYLINDRICAL_SURFACE</td><td colspan="2">1. AXIS2_PLACEMENT_3D ('Cylinder Axis2P3D'
2. last number indicates radius of cylinder</td><td>1. A2P3D
2. s_r</td></tr>
<tr><td>AXIS2_PLACEMENT_3D ('Cylinder Axis2P3D'</td><td colspan="2">1. CARTESIAN_POINT(' Axis2P3D Location'
2. Numbers without # indicates cylinder axis coordinates</td><td>1. cp
2. Ax,Ay,Az</td></tr>
<tr><td>TOROIDAL_SURFACE</td><td colspan="2">1. AXIS2_PLACEMENT_3D ('Torus Axis2P3D'
last numbers without # indicates radius of torus</td><td>1.A2P3D
2. s_r</td></tr>
<tr><td>FACE_OUTER_BOUND</td><td colspan="2">EDGE_LOOP</td><td>E_L</td></tr>
<tr><td>EDGE_LOOP</td><td colspan="2">ORIENTED_EDGE</td><td>O_E</td></tr>
<tr><td>ORIENTED_EDGE</td><td colspan="2">EDGE_CURVE</td><td>E_C</td></tr>
<tr><td>EDGE_CURVE</td><td colspan="2">1.VERTEX_POINT
2. Find the string name & string to be included in table
3. last letter(T/F)
String Name | String in table
CIRCLE | circle
LINE | line
BOUNDED_CURVE | bcurve
B_SPLINE_CURVE _WITH_KNOTS | bspline</td><td>1.V_P
2.circle, condition (EDGE_CURVE construction)
3. T/F</td></tr>
<tr><td>CIRCLE(‘generated circle’</td><td colspan="2">1. AXIS2_PLACEMENT_3D ('Circle Axis2P3D'
2. last number without # indicates radius of circle</td><td>1. A2P3D
2. c_r</td></tr>
<tr><td>AXIS2_PLACEMENT_3D ('Circle Axis2P3D'</td><td colspan="2">1. CARTESIAN_POINT('Axis2P3D Location'
2. last numbers without # indicates circle center co-ordinates</td><td>1.A2P3D
2. cx,cy,cz</td></tr>
</table>

Table 3.2: Extracted data in various tables of window

Table name	Array to be placed in table
FOB	A_F,FOB,E_L,O_E,E_C,V_P,C_P
FB	A_F,F_B,E_L,O_E,E_C,V_P,C_P,X,Y,Z
FOB vertex	Cpv1,x1,y1,z1,cpv2,x2,y2,z2
PLANE	A_F,cysu,A2P3D,cy_str,s_r,cp,Ax,Ay,Az
CIRCLE	A_F,E_C,circle,A2P3D,c_r,cx,cy,cz,condition
MAIN	A_F, E_C, cy_str, s_r, Ax, Ay, Az, condition, cx, cy, cz, T/F

For the purpose of recognition of feature, evaluation of edge curve construction, circle centers, radius of the circles, various surfaces and surface radius are required. For the explanation of above aspects, recognition of a cylindrical feature is described in detail in the following sections.

3.2.1. Edge Curve Construction

As shown in Figure 3.2 CLOSED_SHELL (#35) consists of various ADVANCED_FACEs (#75). Each ADVANCED_FACE is bound by number of edges which is termed as FACE_OUTER_BOUND (#74). Each FACE_OUTER_BOUND is formed with edges as a loop which is termed as EDGE_LOOP (#69). Further each EDGE_LOOP is formed by ORIENTED_EDGEs (#70, #71, #72, #73) termed as EDGE_CURVE (#49, #56, #63, #68). Each EDGE_CURVE is formed with different geometric shapes like LINE (#53), CIRCLE (#44) etc.

3.2.2. Finding the Centers of Circles of a Cylinder

As shown in Figure 3.2 and in continuation from EDGE_CURVE construction (eg. line, circle, line, circle) each circle (#44) consists of AXIS2_PLACEMENT_3D (#43). In turn,

AXIS2_PLACEMENT_3D consists of a circle center as CARTESIAN_POINT (#41). The Last three numbers without hashes of CARTESIAN_POINT indicate coordinates of the circle center (x=0, y=0, z=0).

3.2.3. Finding the Radius of Circle of a Cylinder

As shown in Figure 3.2 and in continuation from EDGE_CURVE construction (eg. line, circle, line, circle), the last number without hash (#) in the CIRCLE (#44) denotes the radius of the circle (50).

3.2.4. Finding the Surface and Radius

As shown in Figure 3.2 CLOSED_SHELL (#35) consists of various ADVANCED_FACEs (#75). Each ADVANCED_FACE consists of surfaces like CYLINDRICAL_SURFACE (#40), CONICAL_SURFACE,TOROIDAL_SURFACE,SPHERICAL_SURFACE etc. The last number without hash (#) in the CYLINDRICAL_SURFACE denotes the radius of the cylinder (50).

3.2.5. Recognition of Axis Coordinates of a Cylinder

As shown in Figure 3.2 CYLINDRICAL_SURFACE string line contains axis information (#39). This entity (#39) is in turn connected to AXIS2_PLACEMENT_3D. In this line entity #93 indicates CARTESIAN_POINT. In this line last three numbers (0, 30, 0) indicate axis coordinates of the cylinder. Above all explained extracted data from STEP file is shown Figure 3.3.

```
Line 571: #35=CLOSED_SHELL('Closed Shell',(#75,#92,……..#177,));
        Line 596: #75=ADVANCED_FACE('PartBody',(#74),#40,.T.);
Line 6: #40=CYLINDRICAL_SURFACE('generated cylinder',#39,50.);
        Line 775: #74=FACE_OUTER_BOUND(',#69,.T.) ;
Line 732: #69=EDGE_LOOP(',(#70,#71,#72,#73)) ;
        Line 70: #70=ORIENTED_EDGE(',*,*,#49,.F.) ;
        Line 71: #71=ORIENTED_EDGE(',*,*,#56,.T.) ;
        Line 72: #72=ORIENTED_EDGE(',*,*,#63,.T.) ;
        Line 73: #73=ORIENTED_EDGE(',*,*,#68,.F.) ;
Line 363: #49=EDGE_CURVE(',#46,#48,#44,.T.) ;
Line 364: #56=EDGE_CURVE(',#46,#55,#53,.F.) ;
Line 365: #63=EDGE_CURVE(',#55,#62,#60,.T.) ;
Line 366: #68=EDGE_CURVE(',#48,#62,#67,.F.) ;
        Line 630: #44=CIRCLE('generated circle',#43,50.) ;
        Line 53: #53=LINE('Line',#50,#52) ;
        Line 631: #60=CIRCLE('generated circle',#59,50.) ;
        Line 67: #67=LINE('Line',#64,#66) ;
Line 285: #43=AXIS2_PLACEMENT_3D('Circle Axis2P3D',#41,#42,$) ;
Line 286: #59=AXIS2_PLACEMENT_3D('Circle Axis2P3D',#57,#58,$) ;
        Line 153: #41=CARTESIAN_POINT('Axis2P3D Location',(0.,0.,0.)) ;
        Line 155: #57=CARTESIAN_POINT('Axis2P3D Location',(0.,20.,0.)) ;
Line 284: #39=AXIS2_PLACEMENT_3D('Cylinder Axis2P3D',#93,#94,#95) ;
        Line 156: #93=CARTESIAN_POINT('Axis2P3D Location',(0.,30.,0.)) ;
```

Fig. 3.2: Partial lines from STEP file for cylinder recognition

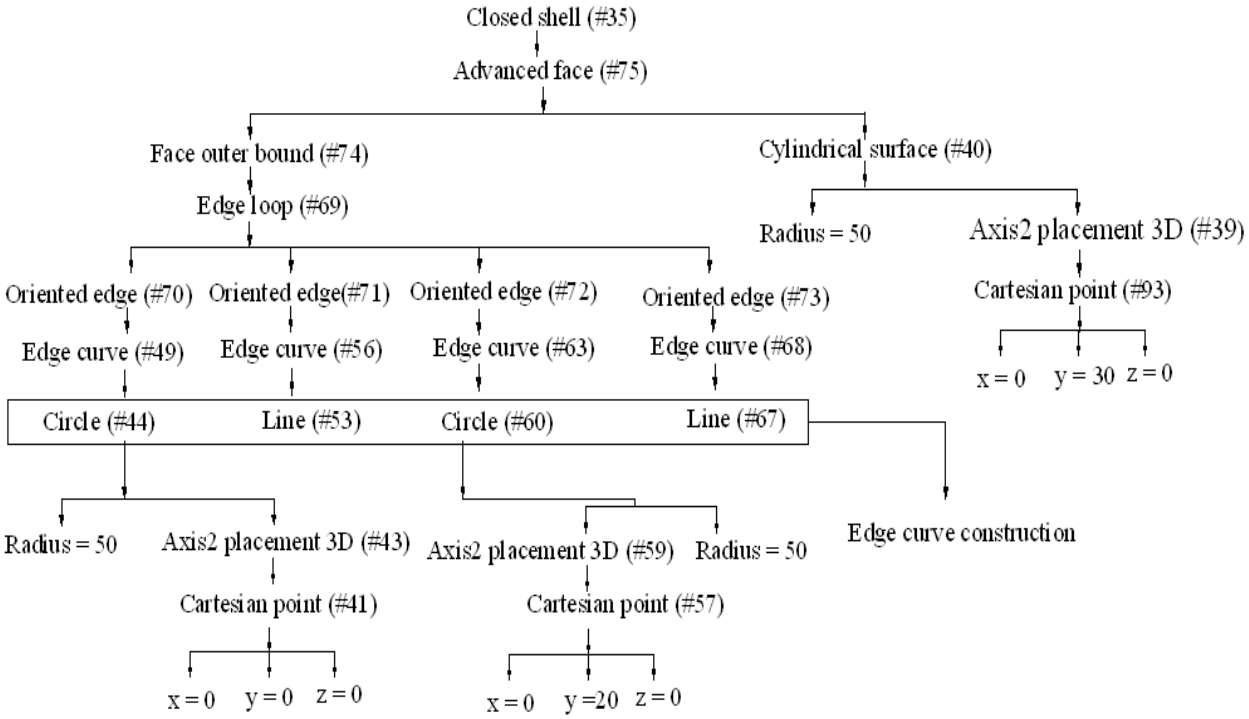

Fig. 3.3: Data extraction from STEP file for cylinder recognition

After geometrical data extraction, various rules are used for feature recognition process. Recognition of various features is explained in Chapter 4.