2

RAPID PROTOTYPING PROCESS

The objective of rapid prototyping is to quickly fabricate any complex-shaped, three-dimensional part from CAD data. Rapid prototyping is an example of an additive fabrication process. In this method, a solid CAD model is electronically sectioned into layers of predetermined thickness. These sections define the shape of the part collectively.

The focus of this chapter is on those system elements that affect the shape of the part: the CAD file, the STL (stereolithography) file, problems and repairs of STL files, and other file formats. In short, the modeling principles of rapid prototyping will be discussed in this chapter.

2.1 PRINCIPLES OF AUTOMATED PROCESSES

Rapid prototyping is essentially a part of *automated fabrication,* a technology that lets us make three-dimensional parts from digital designs [1]. There are several advantages of *automated fabrication* over manual fabrication and molding processes. Some of these advantages are computer-aided design, quick design changes, and precise dimensioning. Fabrication processes, manual or automated, can be classified as subtractive, additive, or formative. These processes are shown in Figure 2.1.

Subtractive Process In this process, one starts with a solid block of material larger than the final size of the finished object, and then material is removed slowly until the desired shape is reached.

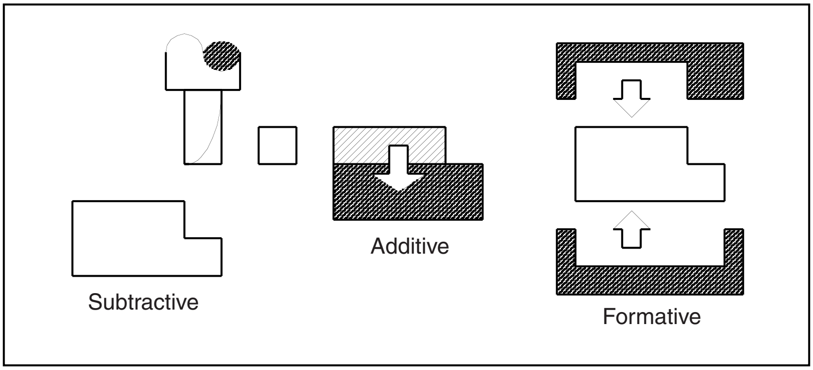


Figure 2.1 Three types of fundamental fabrication processes.

Subtractive processes include most forms of machining processes—computer numerical control (CNC) or otherwise. Most widely used examples include milling, turning, drilling, planning, sawing, grinding, electrical discharge machining (EDM), laser cutting, water-jet cutting, and many other methods.

Additive Process Unlike the subtractive process, this process involves manipulation of material so that successive pieces of it combine in the right form to produce the desired object.

The rapid prototyping process (layered manufacturing) falls into the additive fabrication category. Figure 2.2 shows the additive layer-by-layer process of rapid prototyping. Examples of RP processes include SLA, FDM, LOM, SLS, solid ground curing (SGC), direct shell production casting (DSPC), and 3D printing (3DP).

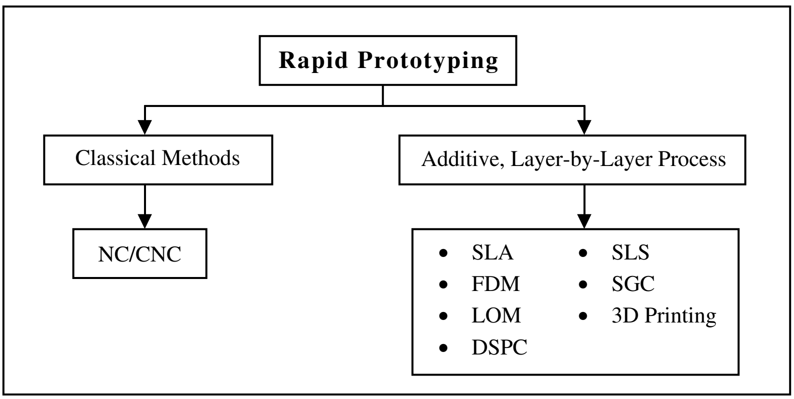


Figure 2.2 Additive layer-by-layer process.

Formative Process In this process, mechanical forces are applied to material so as to form the desired shape.

Examples of the formative fabrication process include bending, forging, electromagnetic forming, and plastic injection molding.

Two or three of these processes can be combined to form a *hybrid* process. Hybrid processes are expected to contribute significantly to the production of goods in the future. Progressive press working is an example of hybrid machines that combine two or more fabrication processes. In progressive press working a hybrid of subtractive (as in blanking or punching) and formative (as in bending and forming) is used.

2.2 RP FUNDAMENTALS

All prototypes made with both the current and evolving RP processes have several features in common [2, 3]. A solid or surface CAD model is electronically sectioned into layers of predetermined thickness. These sections define the shape of the part collectively. Information about each section is then electronically transmitted to the RP machine layer by layer. The RP machine processes materials only at "solid" areas of the section. Subsequent layers are sequentially processed until the part is complete. It is this sequential, layered, or lithographic approach to parts manufacturing that defines RP. The RP process basically uses the following steps to make prototypes:

1. Create a CAD model of the design.
2. Convert the CAD model to STL file format.
3. Slice the STL file into 2D cross-sectional layers.
4. Grow the prototype.
5. Postprocessing.

The five-step process is shown in Figure 2.3.

2.2.1 Creation of Solid Models

The first step in creating a prototype is the creation of a CAD solid model. RP requires that we make a fully closed, water-tight model such that even if we were to pour water into the volume of the model, it would not leak.

A solid is a volume completely bounded by surfaces, which means the edges of all surfaces must be coincident with one, and only one, other surface edge. Unlike wire-frame and surface modeling, solid modeling stores volume information. A CAD solid model not only captures the complete geometry of an object, it can also differentiate the inside and the outside of the space of that object. Many other volume-related data can be obtained from the model.

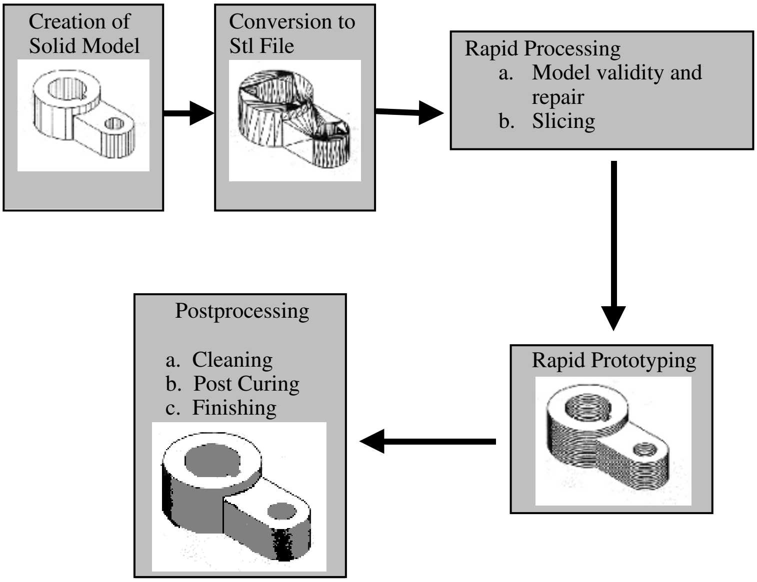


Figure 2.3 Five-step process of rapid prototyping.

Solid models can be created using a CAD software package such as AutoCAD, Pro/Engineer, CATIA, Solid Works, or many other commercially available solid modeling programs.

2.2.2 Conversion to STL File

Once a solid model is created and saved, it is then converted to a special file format known as STL (stereolithography) [4]. This file format originated from 3D Systems, which pioneered the stereolithography process. Actually, the Albert Consulting Group under contract to 3D Systems developed the STL file format to support the new revolutionary manufacturing technology, called stereolithography. Though not ideal, it is sufficient to meet the needs of today's rapid prototyping technology, which generally build monomaterial parts. The success of this file format has been impressive. Today, a decade later, the STL file format remains the de facto standard for the rapid prototyping industry.

The success of the STL file format is due to its sufficiency, its simplicity, and its monopoly [5]. Its mathematical sufficiency stems from the fact that it describes a solid object using a boundary representation (B-rep) technique. An STL file format represents the virtual CAD model of the object to be prototyped as a collection of triangular facets. These triangular facets, when taken together, describe a polyhedral approximation of the objects' surface, that is, a polyhedral approximation of the boundary between material and nonmaterial. In short, an STL file is nothing more than a list of x, y, and z coordinate triplets that describe a connected set of triangular facets. It also includes the direction of the normal vector for each triangle, which points to the outer surface of the model.

Most CAD/CAM software vendors supply the STL file interface. Since 1990, many CAD/CAM vendors have developed and integrated this interface into their systems. Tessellation is the process of approximating a surface by triangular facets. The CAD STL file interface performs surface tessellation and then outputs the facet information to either a binary or ASCII STL file format.

The output of STL file formats can be expressed in binary or ASCII format. The characteristics of binary and ASCII STL outputs are shown in Table 2.1.

Figures 2.4 and 2.5 show a binary STL file format and an ASCII STL file format, respectively.

There are three steps to STL file creation [6]:

1. Select the part(s) to be converted to STL representation.
2. Set the various tolerance parameters for the process.
3. Create a triangular representation of the geometry into an output file.

As mentioned, both surface and solid models can be converted to STL file formats. However, it is very difficult to create STL files from surface models. When processing surface models, the following steps should be followed:

1. Determine all surface adjacencies.
2. Triangulate each surface.
3. Make sure all edge vertices match.
4. Determine a normal that points to the outside of the model for each surface.
5. Output triangles and *normals* to an output file.

During creation of the STL file, the following interface options must be addressed:

TABLE 2.1 Output Types: Binary Versus ASCII STL

Binary	ASCII
Default output type	Referred to as human-readable format
Referred to as machine-readable format	Easily read and understood by humans
More compact and efficient, easier to move through network/transmit	Not very efficient, slower to process, larger file sizes
Not easily read or understood by humans without some translation	Not recommended if moving files through a network

Address	Length	Type	Description
0	80	char	Header information
80	4	long	Number of facets in solid
First facet (50 bytes):			
84	4	float	Normal (x component)
88	4	float	Normal (y component)
92	4	float	Normal (z component)
96	4	float	Vertex 1 (x component)
100	4	float	Vertex 1 (y component)
104	4	float	Vertex 1 (z component)
108	4	float	Vertex 2 (x component)
112	4	float	Vertex 2 (y component)
116	4	float	Vertex 2 (z component)
120	4	float	Vertex 3 (x component)
124	4	float	Vertex 3 (y component)
128	4	float	Vertex 3 (z component)
132	2	short	Attribute info. (not used)
Second facet (50 bytes):			
134...			

Figure 2.4 Binary STL format.

1. Triangulation tolerance
2. Adjacency tolerance
3. Auto-normal generation (on/off)
4. Normal display (on/off)
5. Triangle display (on/off)
6. Header information (text)

```
    SOLID Test_Part_567.Mainfold_z
    FACET NORMAL   0.000000e+00 1.000000e+00 2.634149e-09
OUTER LOOP
    VERTEX         3.000000e+00 1.400000e+00 4.000000e+00
    VERTEX         4.000000e+00 1.400000e+00 4.000000e+00
    VERTEX         4.000000e+00 1.400000e+00 3.000000e+00
    ENDLOOP
    ENDFACET
    FACET NORMAL...
    :
    <remaining facets>
    :
    ENDSOLID
```

Figure 2.5 ASCII STL format.

Some of the features of triangulation tolerance are as follows:

1. Determines how smooth the approximation of the surface or solid will be. In other words, how close the triangles approximate the surface.
   - How close the sides of the triangles that lie along the edges are to the actual edges of the surface.
   - Usually set to one half the desired accuracy of the RP process being utilized.
   - Default is set at 0.0025 in., or 0.05 mm.

The dramatic effects of decreasing the values of triangle tolerances are shown in Figure 2.6.

Some of the features of adjacency tolerance are as follows:

1. It does not affect processing of solids.
2. The default value is 0.005 in., or 0.12 mm.

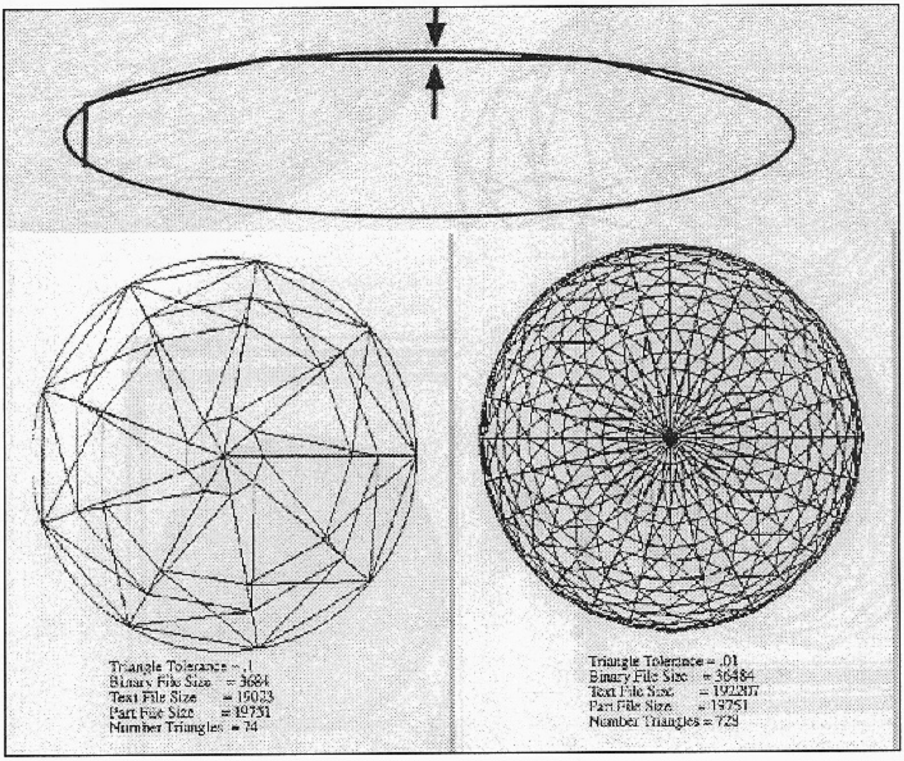


Figure 2.6 Effects of decreasing values of triangle tolerances.

3. The system uses this value to determine if two surfaces will be attached to one another.
4. Edges whose length is smaller than the adjacency tolerance can cause adjacency problems.

The effects of adjacency tolerance are shown in Figure 2.7.

Some of the features of auto-normal generation are as follows:

1. It does not affect processing of solids.
2. Choose a base surface, check the normal, and calculate all others from this surface.
3. Default should be on.

Example 2.1

For the object shown in Figure 2.8:

(a) Draw the part using Pro/Engineer or a similar CAD/CAM system.

(b) Create STL ASCII file with chord height of 1.0 and 0.1.

(c) Discuss the changes that take place when chord height is varied (file size, number of triangles, etc.).

Solution

(a) To generate the solid object, a simple rectangular *protrusion* was first created. Using the *cut* command in the file menu, the rectangular sections were *blind* cut to the specified depths. The circles were then cut completely through the part. The solid object is shown in Figure 2.9.

(b) The part was saved in an ASCII STL file format with the specified chord heights and is shown in Figure 2.10.

(c) Varying the chord height changes the number of triangles created. When the chord height is decreased, there are more triangles, thus a larger file size. The smaller the chord height, the more accurate the

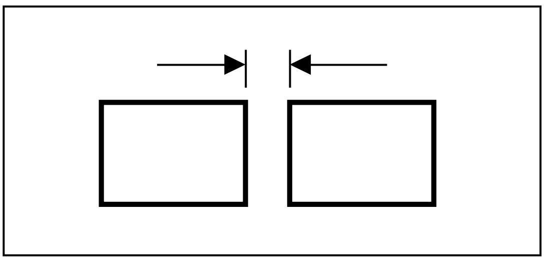


Figure 2.7 Adjacency tolerance.

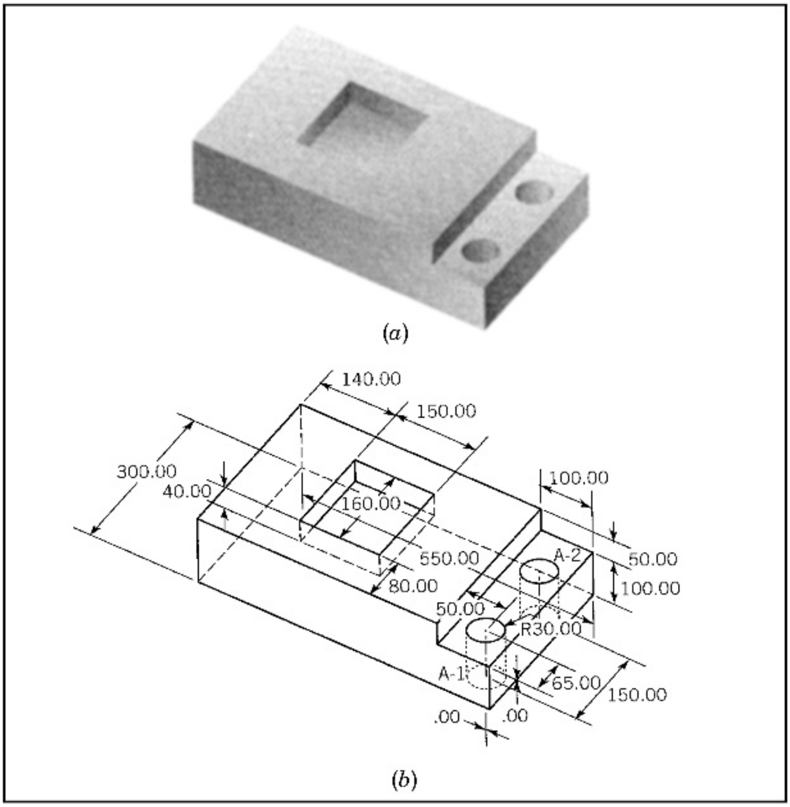


Figure 2.8 Part geometry.

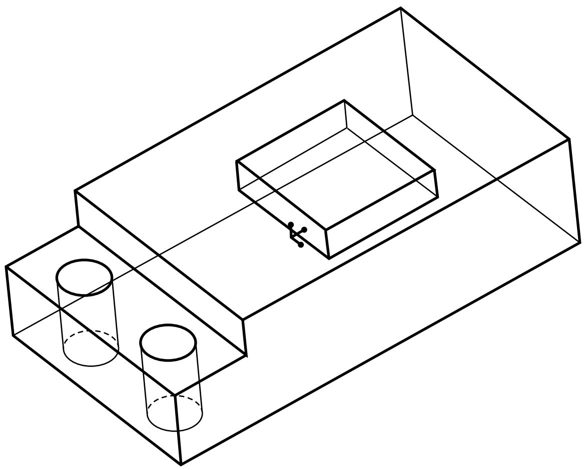


Figure 2.9 Solid object.

```
                    STL ASCII File Format
solid PROBLEM1
 facet normal 0.000000e+00 −0.000000e+00 −1.000000e+00
  outer loop
    vertex 5.500000e+02 1.00000e+02 −1.500000e+02
    vertex 5.500000e+02 1.00000e+00 −1.500000e+02
    vertex 0.000000e+00 0.00000e+00 −1.500000e+02
  endloop
 endfacet
 facet normal 0.000000e+00 −1.000000e+00 0.000000e+00
  outer loop
    vertex 0.000000e+00 0.00000e+00 1.500000e+02
    vertex 0.000000e+00 0.00000e+00 −1.500000e+02
    vertex 5.500000e+02 0.00000e+00 −1.500000e+02
  endloop
 endfacet
 facet normal 0.000000e+00 −0.000000e+00 −1.000000e+00
  outer loop
    vertex 4.500000e+02 1.00000e+02 −1.500000e+02
    vertex 5.500000e+02 1.00000e+02 −1.500000e+02
    vertex 0.000000e+00 0.00000e+00 −1.500000e+02
  endloop
 endfacet
 facet normal 0.000000e+00 −0.000000e+00 −1.000000e+00
  outer loop
    vertex 4.500000e+02 1.50000e+02 −1.500000e+02
    vertex 4.500000e+02 1.00000e+02 −1.500000e+02
    vertex 0.000000e+00 0.00000e+00 −1.500000e+02
  endloop
 endfacet
```

Figure 2.10 STL ASCII.

object. However, the more accurate model will take longer to produce. Figure 2.11 shows the object sliced at different chord heights.

2.2.3 Slicing the File

The file is taken from its 3D model surfaces and converted to many triangles, a step referred to as *slicing*. The more complex the object, the more triangles are required, and thus the bigger the file that makes up the CAD model as well as a support structure for the part to be grown on. The sliced object is saved as an STL file and is now in a format the RP computer recognizes.

2.2.4 Making or "Growing" the Prototype

The part is submitted to the RP computer and the machine runs until the part is complete. RP machines build one layer at a time from polymers, paper, or

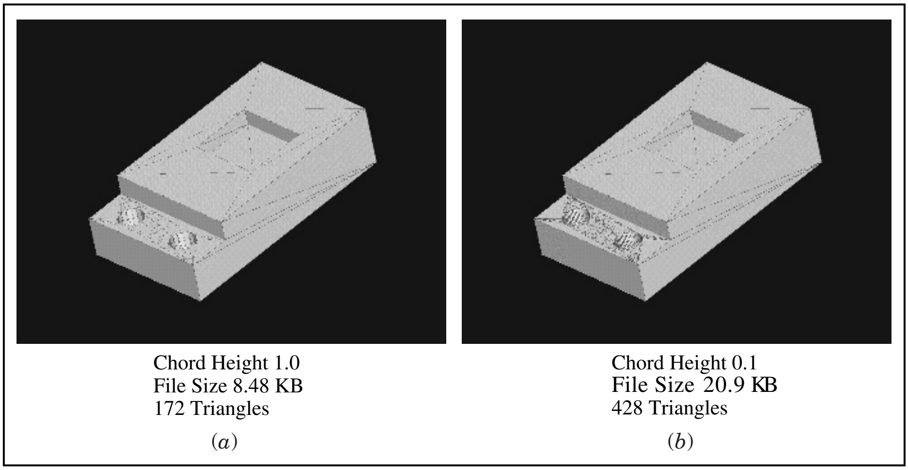


Figure 2.11 (*a*) Chord Height 1.0 and (*b*) chord height 0.1.

powdered metals. Most machines are fairly autonomous needing little human intervention. Build times vary depending on size and number of parts required.

2.2.5 Postprocessing

The final step in rapid prototyping is postprocessing. It essentially consists of part removal and cleaning and of postcuring and finishing. This step generally involves manual operations where an operator does the postprocessing with extreme care. Otherwise, the part may be damaged and may need to be prototyped again [7].

The tasks for postprocessing are different for different prototyping systems. Table 2.2 shows the necessary postprocessing tasks for some selected prototyping systems.

Part removal and cleaning refers to taking the prototype out of the prototyping machine and removing excess materials, including support materials, which may have remained on the part. As shown in Table 2.2, there is a need for cleaning with both the FDM machine and SLA machine. With SLA parts, excess resins residing in entrapped areas need to be removed. Likewise, excess powder for SLS parts and excess woodlike blocks of paper for LOM parts need to be removed.

Postcuring is a task that is usually only needed for SLA and SLS parts. In the SLA process, the laser scans each layer along the boundary and hatching lines only, resulting in side portions of the layers not being completely solidified. Postcuring is needed to complete the solidification process and to improve the mechanical properties of the prototype.

Postcuring is carried out in a specially designed apparatus using ultraviolet (UV) radiation. Optimizing the output wavelength of the postcuring apparatus

TABLE 2.2 Postprocessing Tasks for Various RP Systems

Postprocessing Tasks	Fused Deposition Modeling (FDM)	Solidscape	Stereolithography Apparatus (SLA)	Laminated Object Manufacturing (LOM)	Selective Laser Sintering (SLS)
1. Cleaning	√	√	√	√	√
2. Postcuring		√	√		√
3. Finishing	√	√	√	√	√

determines the uniformity of postcuring with minimal temperature rise and maximum part accuracy. How long it takes to postcure a part depends on the part size. In general, time for postcure is less than that of part building. In any case, both postcuring and part building are unattended processes.

Part finishing is done once the part is postcured. For simple part application, part finishing involves basic cleaning such as sanding or machining to remove additional materials. Care should be taken when handling parts made of wax materials because the materials are very brittle.

2.3 PROBLEMS WITH STL FILE FORMAT

While the STL file format is meeting the needs of the industries that are using RP and while it is the de facto standard in RP industry, it has some inadequacies [2, 7]. Some of these inadequacies are due to the very nature of the STL file format as it does not contain topological data. Also, many CAD vendors use tessellation algorithms that are not robust. Consequently, they tend to create polygonal approximation models, which exhibit the following types of problems:

1. *Gaps* (*Cracks, Holes, Punctures*) *Indicating Missing Faces* When a solid model is converted into an STL file format, the solid model forms are replaced with a simplified mathematical form (triangles). However, if the simplified operation is not done properly, it introduces undesirable geometric anomalies, such as holes or gaps in the boundary surface. This problem is more prone to surfaces with large curvature. Such gaps are shown in Figure 2.12.
2. *Inconsistent Normals* In general, surface normals should be pointed outward. However, the normals of some surfaces could be flipped over, as shown in Figure 2.13, thus, becoming inconsistent with the outward orientation of the original surface.

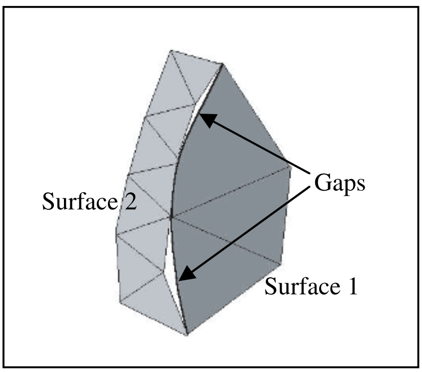


Figure 2.12 Polygonal approximations resulting in gaps.

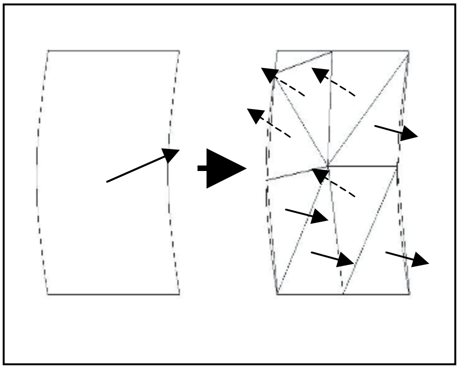


Figure 2.13 Polygonal approximations resulting in inconsistent and incorrect normals.

3. *Incorrect Normals* Sometimes, surface normals stored in the STL file are not the same as those computed from the vertices of the corresponding surfaces.
4. *Incorrect Intersections* Facets may sometimes intersect at locations other than their edges resulting in overlapping facets. (Fig. 2.14)
5. *Internal Wall Structure* Geometric algorithms are used for closing gaps in STL files. However, faulty geometric algorithms could generate internal walls and structures that can cause discontinuities in the solidification of the material (Fig. 2.15)
6. *Facet Degeneracy* Degeneration of facets occurs when they may not represent a finite area and consequently have no normals. Generally, there are two kinds of facet degeneracies: geometric degeneracy and topological degeneracy. A geometric degeneracy takes place when all the vertices of the facet are distinct and all the edges of the facet are collinear. A topological degeneracy takes place when two or more vertices of a facet coincide. Since it does not affect the geometry or the

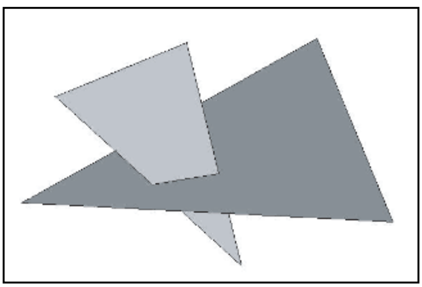


Figure 2.14 Polygonal approximations resulting in incorrect intersections.

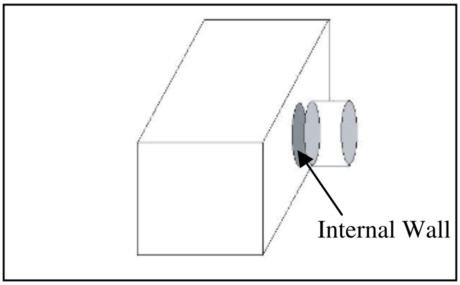


Figure 2.15 Polygonal approximations resulting in internal wall structures.

connectivity of the remaining facets, the faults can be discarded (Fig. 2.16)

7. *Inconsistencies* Sometimes two STL files are combined to create a prototype. If these STL files were created using different tolerance values, it will lead to inconsistencies such as gaps.

2.4 OTHER TRANSLATORS

The problems and inefficiencies of the STL file format have prompted the search for alternate translators. Examples of some of these translators are IGES, HPGL, and computed tomography (CT) data.

1. *IGES File* Initial graphics exchange specification is a common format to exchange graphics information between various CAD systems. It was initially developed and promoted by the then American National Standards In-

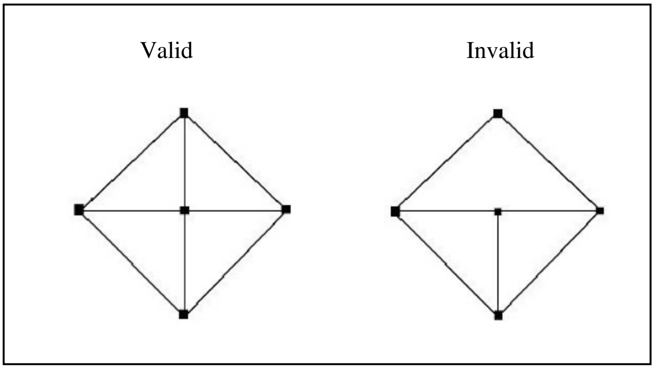


Figure 2.16 Polygonal approximations resulting in geometric degeneracy.

stitute (ANSI) in 1981. The IGES file can precisely represent both geometry and topological information for a CAD model. An IGES file contains information about surface modeling, constructive solid geometry (CSG) and boundary representation (B-rep). The Boolean operations for solid modeling such as *union, intersection,* and *difference* are also defined in the IGES file. It can precisely represent a CAD model by providing entities of points, lines, arcs, splines, NURBS surface, and solid elements. The primary advantage of IGES format is its widespread adoption and comprehensive coverage.

However, there are some disadvantages of the IGES format as it relates to its use as an RP format. These are:

1. The inclusion of redundant information for RP systems.
2. The algorithms for slicing IGES file are more complex than those for slicing STL file.
3. The support structures that are needed for some RP systems cannot be created using IGES format.

IGES is a very good interface standard for exchanging information between various CAD systems. It does, however, fall short of meeting the standards for RP system.

2. *Hewlett–Packard Graphics Language* (*HPGL*) *File* HPGL has been the standard data format for graphic plotters for many years [1, 2]. Generally, data types are two-dimensional representing lines, circles, splines, text, and so forth. Many commercial CAD systems have the interface to output HPGL format, which is a two-dimensional geometry data format and does not require slicing. However, there are two major problems with the HPGL format. First, since HPGL is a 2D data format, the files are not appended, leaving hundreds of small files needing logical names and transformation. Second, all the required support structures must be generated in the CAD system and sliced in the same way.

3. *CT Data* CT scan data is a new format used for medical imaging. The format has not been standardized yet. Formats are proprietary and vary from machine to machine. The CT scan generates data as a grid of three-dimensional points, where each point has a varying shade of gray indicating the density of body tissue present at that point. Data from CT scan are being regularly used to prototype skull, femur, knee, and other biomedical components on FDM, SLA, and other RP systems. The CT data essentially consist of raster images of the physical objects being imaged. It is used to produce models of human temporal bones.

Models using CT scan images can be made using CAD systems, STL interfacing, and direct interfacing. CT data is used to make human parts such as leg prostheses, which are used by doctors for implants. However, the main problem with CT image data is the complexity of the data and the need for a special interpreter to process this data.

2.5 CASE STUDY: DESIGNING AND PROTOTYPING A SPUR GEAR

This section describes a complete prototyping process using the FDM system. It also describes step-by-step procedures for making the prototype, as well as the hardware and software used for making the prototype.

2.5.1 Introduction

The spur gear has teeth parallel to the axis of rotation and is used to translate motion from one shaft to another. This type of gear-and-chain assembly is the same as the one used on motorcycles. A three-dimensional CAD file of a spur gear was created using AutoCAD and then rapid prototyped using QuickSlice software and an FDM 1650 rapid prototyping machine. The machine uses FDM technology to create parts in a relatively short period of time. The material used is ABS plastic filament, which is heated above its melting point and then extruded through a nozzle onto a build platform. When one layer is finished, the platform is lowered and a new layer is formed on top of it. The fact that FDM is an additive process allows it to make more complex parts than conventional subtractive CNC methods.

The AutoCAD software allowed for the part to be dimensioned and created accurately. In Figure 2.17 a two-dimensional drawing of the gear is shown to indicate the size of the part. The advantage of creating this part in AutoCAD is that the dimensions on the gear can be changed to allow for different gear ratios easily and quickly. This software allows almost any geometry to be created and then saved as an STL file.

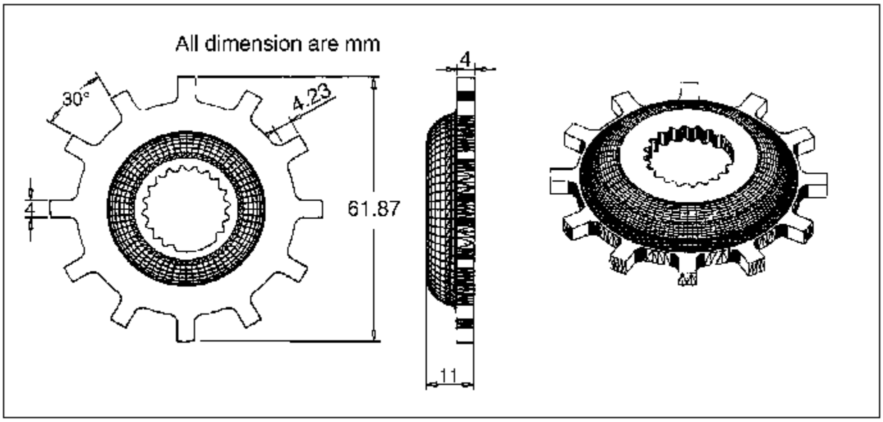


Figure 2.17 Spur gear CAD drawing.

2.5.2 Project Procedure

The fabrication process for this engineering prototype consisted of five steps: These include the CAD process, conversion to STL file, slicing the file, setup and build, and postprocessing. An image of the prototyping equipment used can be seen in Figure 2.18.

1. *CAD Process*
   a. A detailed CAD drawing of the part was created on AutoCAD (Version 14).
   b. The drawing created was a three-dimensional drawing and was made as a solid object.
   c. If there is more than one component to the model that is to be created, the "unity" function must be applied on AutoCAD. This is because the model should be one solid object in order to convert later to a stereolithography (STL) file.
   d. The CAD drawing was then converted into an STL file by choosing File and then Export commands.
   e. The file was now ready to be opened on the QuickSlice software.
2. *Part Preparation*
   a. The first step after opening the file was to properly orient the part to be created. Proper orientation can reduce support needed for the

Figure 2.18 FDM 1650 and attendant workstation.

model and lessen valuable prototyping time. The orientation was set by choosing the tab at the left of the screen, which says Orient stl, and choosing the Rotate option. In this case, the gear was laid flat, which was the optimal position for this particular part.

b. Next, the part was sized in order to fill the build envelope. This can be done by selecting Orient .stl and choosing the Scale option.

c. Now that the part had been oriented and sized, it was ready to be sliced. The Slice tab was chosen and the Slice icon was selected. The software then did all the calculations necessary to slice the part. Once the part was sliced, only the bottom layer could be viewed, so the All icon was chosen. This allows for all the layers to be viewed at the same time, the model should be all red, if not there is a discontinuity in the structure of the model. Therefore, another .stl file must be created to fix the problem.

d. If the model is completely red, then support can now be added to the model. The Create Support icon was chosen and the software analyzed the part and added support where necessary. After the necessary support was added, the Create Base icon was selected and a base was added to the part. This gives the machine a flat surface to lay down the ABS plastic.

e. The next, and one of the most crucial steps, was creating the roads for the part. This is the path the extrusion head will follow to create the part. The Create Roads icon was selected. The software then created the appropriate paths.

f. The final step was to save the created file to .sml format. This file format allows the machines to read the information contained in the file.

3. *Set Up and Build*

   a. The FDM machine was operating at an overall temperature of 68°C (envelope temperature). The model extrusion head was operated at 270°C and the support material head was operating at 265°C.

   b. The FDM Send icon was selected and a file window popped up. The appropriate file was chosen from this window. The machine was now receiving the .sml file information. The Pause button on the machine was blinking at this time.

   c. A 10 × 10-in. substrate was placed into the building chamber. The Pause button was pushed. The machine now reset itself to the home position. Now the part origin was set and the nozzle tip was placed slightly into the substrate. The Pause button was pressed again and the machine began making the part.

4. *Postprocessing*

   a. Support removal was accomplished with the use of simple hand tools (e.g., spatula, knife, etc.).

b. Sand paper was also used to completely remove the base material from the base.

5. *Conclusions* This project allowed a part to be designed and then created using RP technology. An image of the final part produced can be seen in Figure 2.19. A project such as this can provide an invaluable experience to a group of young engineers in order to understand the process of making a design, and having a prototype to test before full production of the part begins. This is important when entering industry, where knowledge of cutting edge technology such as RP can give one an advantage over others.

   This experience can also help prove the efficiency and flexibility of a CAM process, which involves rapid prototyping. The machine can produce over eight different parts with completely different geometry in less than 3 weeks. This is actually remarkable for a machine that takes up no more room than a closet. There are size and material limitations, but the advantages definitely outweigh the disadvantages, which are limited part size and less than optimum surface finish.

2.6 SUMMARY

In this chapter the basic concepts of the rapid prototyping process were introduced and discussed. Rapid prototyping is essentially a part of *automated fabrication technology* that allows one to make three-dimensional parts from a digital design. Fabrication process is generally classified as subtractive, additive, and formative or even hybrid. Rapid prototyping is an additive process. All rapid prototyping processes basically consist of five steps: creation of solid model, conversion of solid model to stereolithography file format, slicing the file using a software package, actually making the prototype, and post-

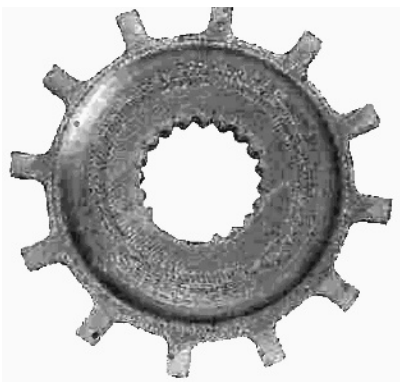


Figure 2.19 Prototype of the spur gear.

processing of the prototype. The most common file format commonly used in industries and academia is the STL format, developed by the Albert Consulting Group for 3D Systems. Although the file format is not perfect, it is still being used as de facto standard for rapid prototyping industry.

PROBLEMS

2.1. What is a rapid prototyping system? Describe briefly a system showing all its components.

2.2. What are the purposes of data creation?

2.3. What is the most common file format used for RP?

2.4. What are the advantages of the STL file format?

2.5. What are the shortcomings (problems) of the STL file format?

2.6. Mention several translators that can be used in place of STL format. What are some drawbacks or limitations of these?

2.7. What is postprocessing? Do you need postprocessing for all RP systems?

2.8. Laboratory project: Design and prototype a product.

a. Use a standard CAD to design the part shown in Figure P2.1.

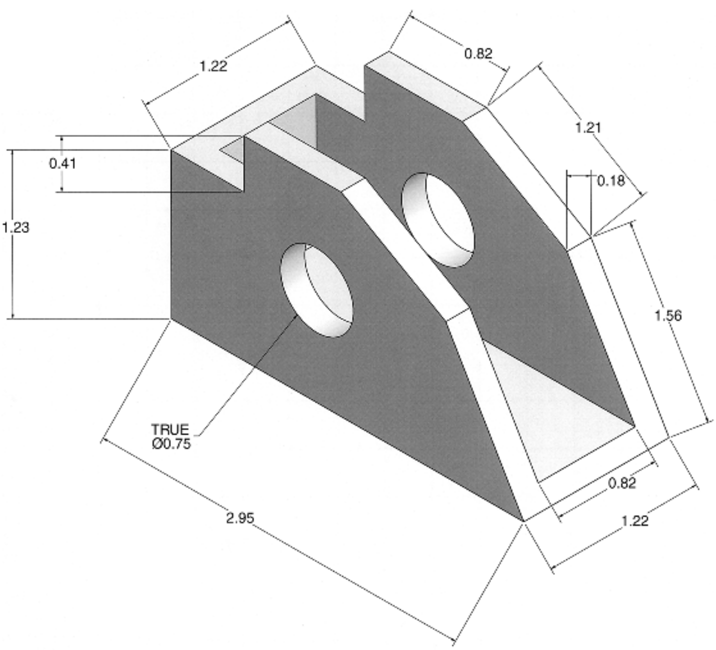
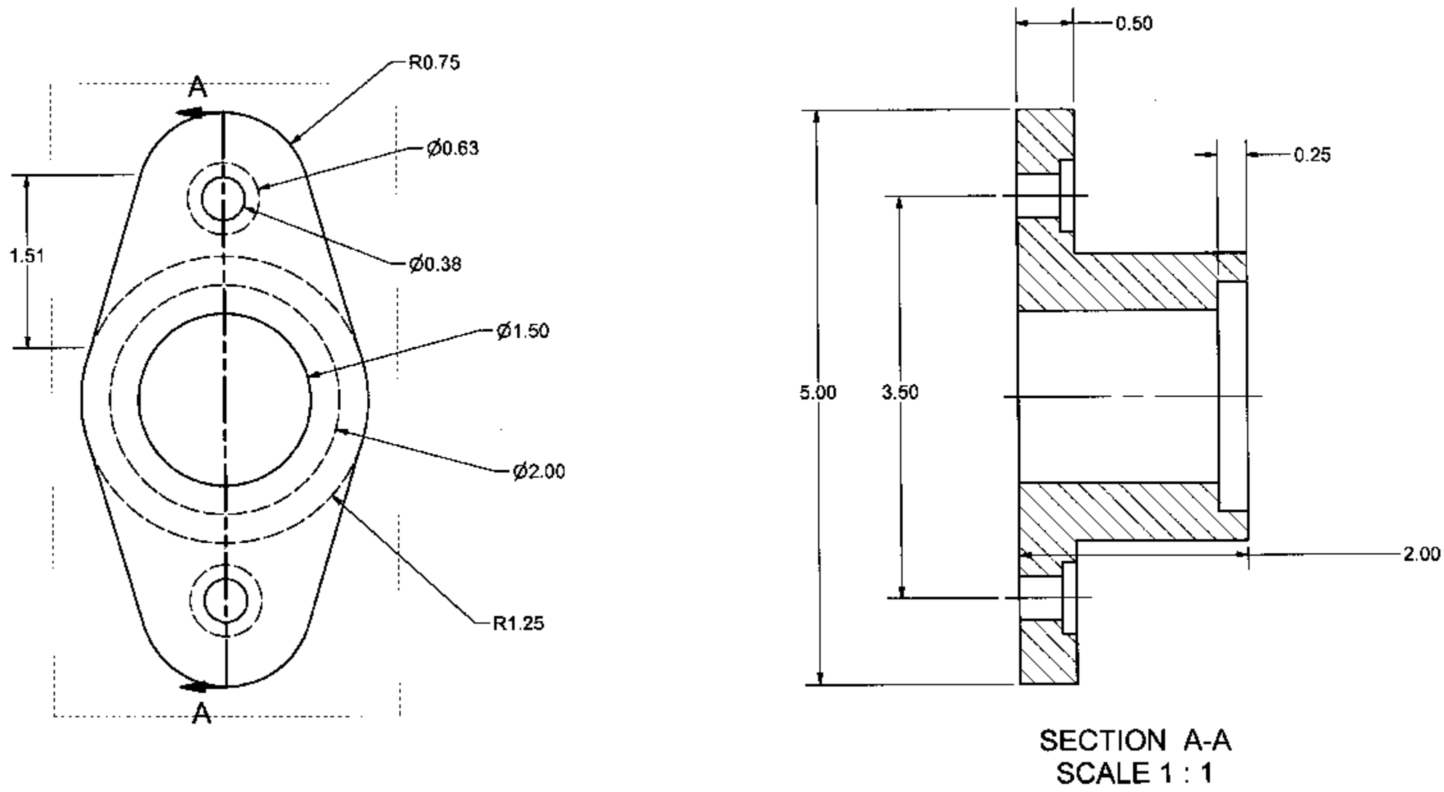


Figure P2.1



A
R0.75
Ø0.63
Ø0.38
1.51
Ø1.50
Ø2.00
R1.25
A
0.50
0.25
5.00
3.50
2.00
SECTION A-A
SCALE 1 : 1

Figure P2.2

b. Convert the drawing file into an STL file using conversion rules described in the text.

c. Use an available RP system (FDM, SLA, LOM, SLC, or 3DP) to prototype the part.

d. Use postprocessing techniques to complete the prototyping cycle.

2.9. For the object as shown in Figure P2.2:

a. Draw the part using Pro/Engineer or a similar CAD/CAM system.

b. Create STL binary file with chord height of 1.0 and 0.1.

c. Create STL ASCII file with chord height of 1.0 and 0.1.

d. Discuss the changes that take place when chord height is varied (file size, number of triangles, etc.).

REFERENCES

1. M. Burns, *Automated Fabrication: Improving Productivity in Manufacturing,* Prentice-Hall, Englewood Cliffs, NJ, 1993.
2. P. Jacobs, *Rapid Prototyping and Manufacturing & Fundamentals of Stereolithography,* SME Publications, Hong Kong, 1992.
3. S. Ashley, Rapid Prototyping Systems—Special Report, *Mechanical Engineering,* 34–43, 1991.
4. L. Wood, *Rapid Automated Prototyping: An Introduction,* Industrial Press, New York, 1993.
5. H. Bohn, File Format Requirements for the Rapid Prototyping Technologies of Tomorrow, International Conference on Manufacturing Automation Proceedings, Hong Kong, 1997.
6. J. Miller, Fundamentals of Rapid Prototyping and Applications in Manufacturing, Westec Conference, Los Angeles, CA, 1996.
7. C. Kai and K. Fai, *Rapid Prototyping: Principles and Applications in Manufacturing,* Wiley, New York, 1997.