10. Feature Construction

Although the use of solid modeling grew steadily during the 1980s, companies were not realizing many of the productivity gains promised by CAD vendors. One of the reasons was that the process of creating a solid model was much more abstract than the process of designing real-world products.

10.1. CONSTRAINT-BASED MODELING

Constraint-based modeling is a technique that can help the CAD operator manage the model modification process. In a constraint-based modeler, describing the relationship of geometric elements with equations and logical relationships creates a part. For example, in a traditional CAD modeler, if a plate is to have a length equal to 32 mm and a width of half the length, a rectangle 32 mm long and 16 mm wide would be created. Traditional CAD modeler creates geometry a specified size (Figure 10.1A). In a constraint-based modeler, the geometric relationships are coded directly into the model with ***parameters:*** the length would be defined as $P_1 = 32$ mm and the width defined as $P_2 = P_1/2$. With a constraint-based modeler, geometry is defined through parameters represented as an equation (Figure 10.1B). With the geometry defined with parameters, the equations are solved to derive dimensional values for each of the features. The power of this approach is seen when the model is modified. Instead of having to individually update all related dimensions, one dimension can be altered, and all dimensions linked through parameters automatically reflect the change. For example, if the length of the plate is reduced ($P_1 = 20$), the width automatically reduces by the appropriate amount to reflect the change in length. Parameters can easily be altered to represent new values (Figure 10.1C).

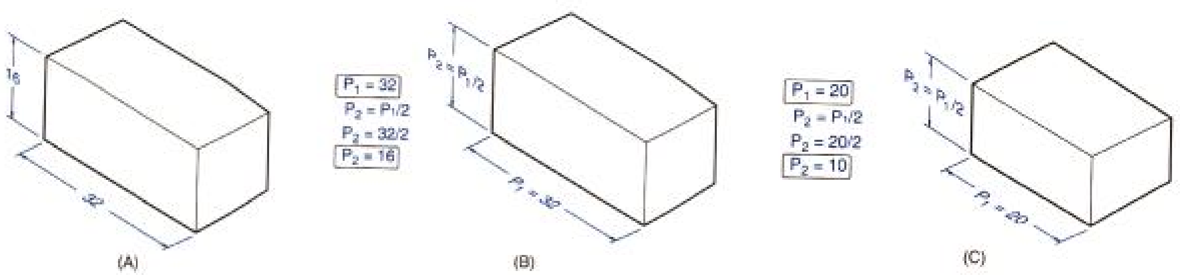

Figure 10.1. Traditional and Parametric Part Definition

Like dimensions, parameters can be associated with geometric features such as lines, curves, and planes. Unlike dimensions, parameters do not have to represent a single static value. Parameters can be

- Assigned numeric values (e.g., $P_1 = 32$ mm).
- Related to other parameters through equations (e.g., $P_2=P_1+20$).
- Related to other parameters through geometric relationships (e.g., P_3 is parallel to P4.
- Varied based on logical relationships (e.g., IF $P_5 > 24$ mm, THEN $P_6 =6$ mm, ELSE $P_6 =4$ mm).

It is important to remember that parameters are always related back to geometric features. A parameter that represents a numeric value measures a distance or angle between two features (i.e., points or lines). Likewise, geometric relationships such as parallelism are made between features such as lines or planes.

In many constraint-based modelers, the modeling process begins by creating a 2-D *sketch profile.* (Figure 10.2A) Unlike in a traditional 2-D or 3-D CAD package, the geometry does not have to be created with a high degree of accuracy; it only needs to represent the basic geometric features. The next step is to *constrain* the model by assigning enough parameters to fully define the size and shape of the 2-D profile.

(Figure 10.2B) Depending on the modeler, some of these parameters may be automatically assigned. For example, if the two horizontal elements are within a preset tolerance range (say, 5 degrees) of parallel, the two edges are constrained to be parallel. You might think of this as being like a snapping or grid system, but, in fact, it is much more powerful. As the part is resized, the sides will stay parallel no matter how far apart they are. Because all of the features are parametrically defined, one or more of the parametric values can be altered, and all related parameters will automatically update. (Figure 10.2C). Once the sketch profile has been constrained and the parameters assigned appropriate values, a 3-D part can be created by an extrusion or sweep operation (Figure 10.2D).

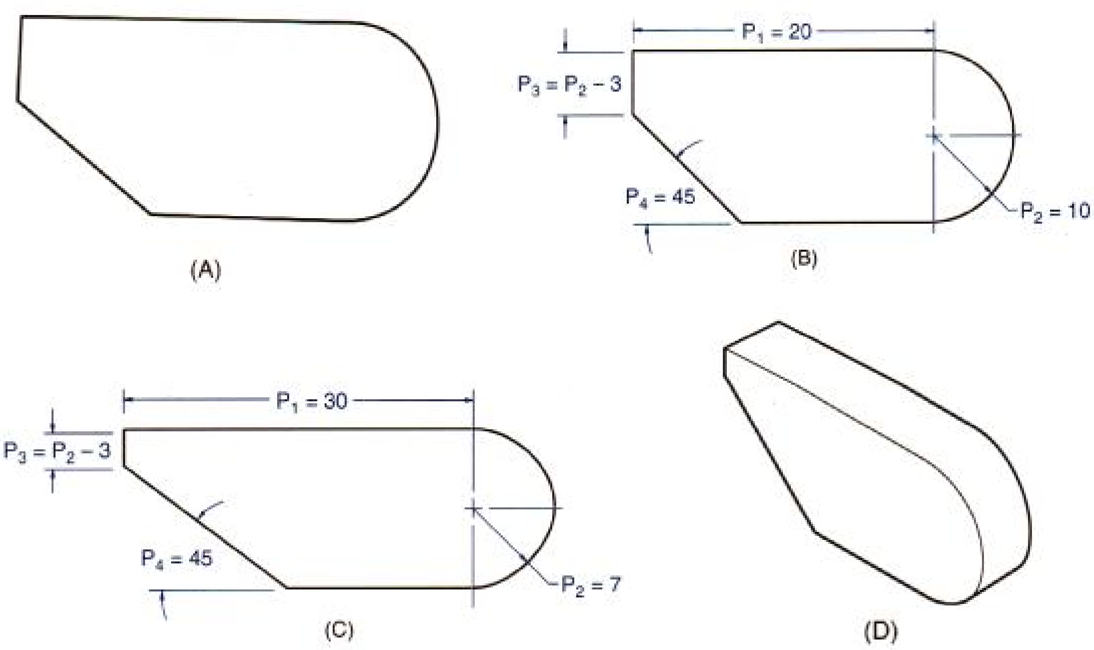

Figure 10.2. Creating a Constrained Model from a Sketch Profile

Two plates are created with holes and two different design intents in Figure 10.3A. In the plate on the left, the hole was intended to be placed 10 mm from the left edge, while the plate on the fight had the hole placed in the center. When the overall width is set to 20 mm, no difference is seen in the two models, but when the overall width is set to a different value, the difference in design intent is immediately seen (Figure 10.3B). With a traditional modeler, an engineer viewing the model in Figure 10.3A would not be able to tell what the design intent was and, therefore, how the hole should shift if the model was altered.

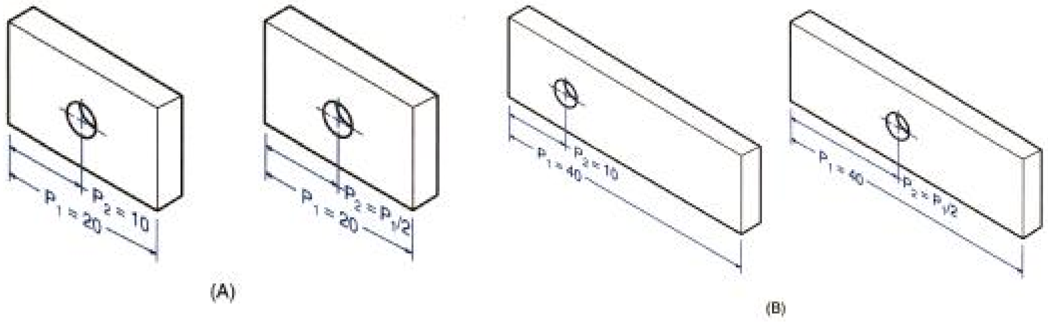

Figure 10.3. The Effect of Design Intent on Model Changes

Though these two parts are initially the same, the parametric constraints represent two different design intents. The part on the left intends to have the bole fixed at an offset of 10 from the left edge whereas the part on the right intends to have the hole centered. The differences in the design intent can be seen when the overall width of the part is altered. Only the hole on the part on the right shifted to the center.

10.2. FEATURE CONSTRUCTION TECHNIQUES

Computer-generated objects are typically made up of features. Most modern CAD software employs a methodology know as *feature-based modeling,* in which features are combined to create the desired final object. The features available in the software are usually designed to relate to how engineers think in their design and manufacturing work. Most software packages have predefined features that you use to build your objects, one feature at a time. As you create the specific features you need to define your required object, you supply dimensions and the location of the features relative to edges, points, and surfaces on the object

10.2.1. Combining Objects

To learn how features are used to create objects, it is important for you to first understand how two objects combine to form a third object. Basically, a cutting, a joining, or an intersecting operation combines objects. Two objects—a simple block and another object that was formed by revolving a section about an axis are shown in Figure 10.4. Notice that these two objects overlap each other. In other words, they share a common volume.

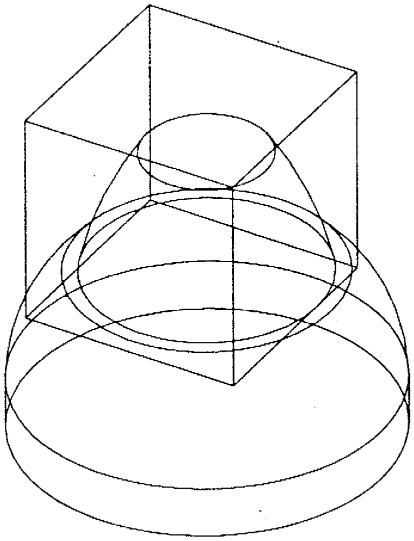

Figure 10.4 Initial Overlapping Objects

In general, cutting, joining, or intersecting them can combine two objects. Some software packages call these Boolean operations—terminology borrowed from mathematical theory and set theory. To review these operations, consider two sets: *A* consists of all the odd numbers (1, 3, 5, 7, 9, ...), and *B* consists of all multiples of three (3, 6, 9, 12, 15, ...).

When two *objects* are combined in a cutting operation by means of CAD software, the volume that the two objects share is removed from the object that is to be cut. In other words, one object acts as a cutting tool on the other object. When two objects are joined, they become a single object, and the shared volume is "absorbed" into the total object. When an intersect operation is performed, the new object consists of the volume that is common to the two original objects. Figure 10.5 shows the results of combining the two objects shown in Figure 10.4 by each of these Boolean operations. Note that, as with the mathematical Boolean operations, for the cutting operation, the result that you obtain depends on which object you specified as the cutting part and which one you specified as the part to be cut.

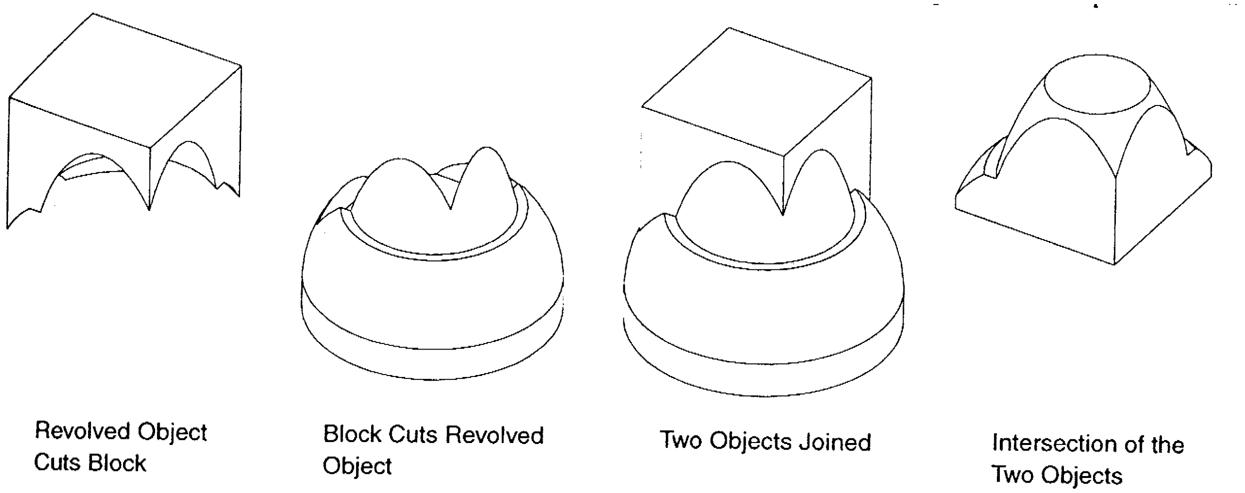
Revolved Object Cuts Block
Block Cuts Revolved Object
Two Objects Joined
Intersection of the Two Objects

Figure 10.5. Combining Objects by Different Methods

10.2.2. Combining Objects with Relations

When you are combining objects by cutting, joining, or intersecting them, you will oftentimes need to establish relationships between the two parts being combined in order to simplify the modification of the part at a later date or to create parts to exact size or shape specifications. These relationships are similar to constraints, in that you are able to define where the features of the object are located with respect to one another before combining them. In this way, when you modify the object at a later point in time, the relationships constrain the relative locations of the parts that make up the final object. When you are creating the geometry of your part, it is important that you understand that *how* you create the part will affect how it can be modified at later times.

10.2.3. Modifying the Features of an Object

One of the major advantages of feature-based modeling is your ability to change the size of individual features on an object at any time. Typically, this involves selecting the dimension that you wish to change and inputting a new value for the dimension. Dimensional values can be changed either on the 2-D figures that are used to create the objects or on the 3-D objects themselves. In this way, you can define the basic shape that your final object will have and then modify the individual sizes of the features that make up the object. However, there are other ways in which you may wish to modify an object's features as you are creating or changing the basic geometry of the object. These modifications usually consist of modifying the order of construction for the features included in your object or deleting the features themselves from the object.

Sometimes when you are creating an object, you may perform the construction operations in the "wrong" order, with the result that the object differs significantly from your design intent.

Another way that you may need to modify an object is by removing certain features from it. You may choose to do this, if your design intent has changed and certain features are no longer required.

Questions

1. Describe the design process.
2. Describe feature-based modeling.
3. Describe constraint-based modeling.

References

1. D.S. Kelley, **Pro/Engineer Instructor**, McGraw-Hill, 2001
2. R.W.Lueptow, M.T.Snyder, J.Steger, **Graphics Concepts with Pro/Engineer**, Prentice Hall, 2001
3. S.A. Sorby, **Solid Modeling with I-DEAS**, Prentice Hall, 2000.
4. J.Rooney, P.Steadman, **Principles of Computer-aided Design**, UCL Press, 1997
5. C.McMahon, J.Browne, **CAD/CAM From Principle to Practice**, Addison-Wesley,1995.
6. A.Tizzard, **An Introduction to Computer-aided Engineering**, McGraw-Hill, 1944.
7. S.J. Ethier, C.A.Ethier, **Instant AutoCAD Mechanical Desktop 5.0**, Prentice Hall, 2002.
8. R. Pizza, **Getting Started with Pro/Engineer**, Prentice Hall, 2002.
9. D.E. LaCourse, **Handbook of Solid Modeling**, McGraw-Hill, Inc. 1995.