



Last updated: Yeh-Liang Hsu (2010-10-12).

Note: This is the course material for “ME550 Geometric modeling and computer graphics,” Yuan Ze University. Part of this material is adapted from *CAD/CAM Theory and Practice*, by Ibrahim Zeid, McGraw-Hill, 1991. This material is be used strictly for teaching and learning of this course.

Bezier curves

1. Defining a Bezier curve

Bezier curve was developed by the French engineer Pierre Bezier (1910-1999) in 1962 for use in the design of Renault automobile bodies. Pierre Bezier then went on to develop the UNISURF CAD/CAM system. As shown in Figure 1, **the Bezier curve is defined in terms of the locations of $n+1$ points.** These points are called data or control points. They form the vertices of what is called the **control or Bezier characteristic polygon.**

Figure 2 shows cubic Bezier curves for various control points. In a Bezier curve, **only the first and the last control points or vertices of the polygon actually lie on the curve.** The curve is also always tangent to the first and last polygon segments. In addition, **the curve shape tends to follow the polygon shape.** These three observations should enable the user to sketch or predict the curve shape once its control points are given.

A closed Bezier curve can simply be generated by closing its characteristic polygon or choosing \mathbf{P}_0 and \mathbf{P}_n to be coincident. Figure 2 also shows an example of closed curve.

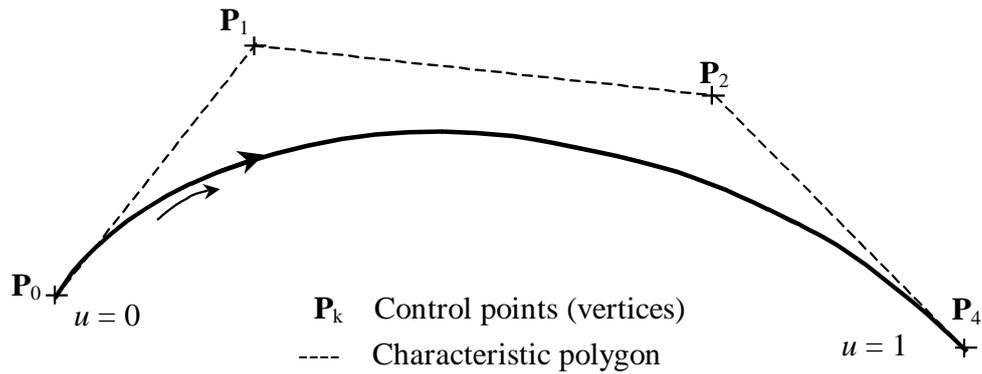


Figure 1. Cubic Bezier curve

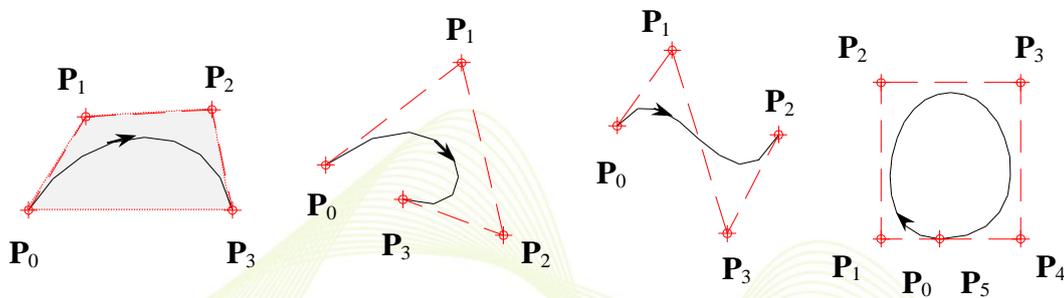


Figure 2. Cubic Bezier curves for various control points.

◇ Assignment 1

Construct Bezier curves similar to those in Figure 2 using your CAD software. Describe the procedure for constructing these curves. ◇

2. Parametric equations of the Bezier curve

In general, a Bezier curve section can be fitted to any number of control points. **The number of control points to be approximated determines the degree of the Bezier curve.** For $n+1$ control points, the Bezier curve is defined by the following polynomial of degree n :

$$\mathbf{P}(u) = \sum_{i=0}^n \mathbf{P}_i B_{i,n}(u), \quad 0 \leq u \leq 1 \quad (1)$$

where $\mathbf{P}(u)$ is any point on the curve and \mathbf{P}_i is a control point, $B_{i,n}$ are the **Bernstein polynomials**. The **Bernstein polynomial serves as the blending or basis function for the Bezier curve** and is given by

$$B_{i,n}(u) = C(n,i)u^i(1-u)^{n-i} \quad (2)$$

where $C(n,i)$ is the binomial coefficient

$$C(n,i) = \frac{n!}{i!(n-i)!} \quad (3)$$

Equation (1) can be expanded to give

$$\begin{aligned} \mathbf{P}(u) = & \mathbf{P}_0(1-u)^n + \mathbf{P}_1C(n,1)u(1-u)^{n-1} + \mathbf{P}_2C(n,2)u^2(1-u)^{n-2} \\ & + \cdots + \mathbf{P}_{n-1}C(n,n-1)u^{n-1}(1-u) + \mathbf{P}_n u^n, \quad 0 \leq u \leq 1 \end{aligned} \quad (4)$$

From Equation (2) and (3), we can get the followings:

$$n = 0, \quad B_{0,0}(u) = 1$$

$$n = 1, \quad B_{0,1}(u) = (1-u)$$

$$B_{1,1}(u) = u$$

$$n = 2, \quad B_{0,2}(u) = (1-u)^2$$

$$B_{1,2}(u) = 2u(1-u)$$

$$B_{2,2}(u) = u^2$$

$$n = 3, \quad B_{0,3}(u) = (1-u)^3$$

$$B_{1,3}(u) = 3u(1-u)^2$$

$$B_{2,3}(u) = 3u^2(1-u)$$

$$B_{3,3}(u) = u^3$$

Therefore for $n = 3$,

$$\mathbf{P}(u) = (1-u)^3 \mathbf{P}_0 + 3u(1-u)^2 \mathbf{P}_1 + 3u^2(1-u) \mathbf{P}_2 + u^3 \mathbf{P}_3. \quad (5)$$

◇ Assignment 2

Derive Bezier curve equations similar to Equation (5) for $n = 0, 1, 2$. Explain what happens when $n = 0, 1$. Derive Bernstein polynomials for $n = 4$, and construct the corresponding Bezier curve equation. ◇

◇ Assignment 3

Assume the coordinates of 4 control points, write a Matlab program to draw the corresponding control polygon. Use $\Delta u = 0.1$ to generate the intermediate points of the Bezier curve using Equation (5). Use straight lines to connect these points to generate the curve. Using proper transformation, create the front, top, side, and isometric views of the 3-dimensional curve and its control polygon. Show your Matlab program too. ◇

From the discussion above, we can see that the major differences between the Bezier curve and the cubic spline curve are:

- (1) The shape of Bezier curve is controlled by its defining points only. First derivatives are not used in the curve development as in the case of the cubic spline. This allows the designer a much better feel for the relationship between input (points) and output (curve).
- (2) The order or the degree of Bezier curve is variable and is related to the number of points defining it; $n+1$ points defined and n th degree curve which permits higher-order continuity. This is not the case for cubic splines where the degree is always cubic for a spline segment.
- (3) The Bezier curve is smoother than the cubic spline because it has higher-order derivatives.

3. Properties of the Bezier curves

A very useful property of a Bezier is that **it always passes through the first and last control points**. If we substitute $u = 0$ and 1 in Equation (4), the boundary conditions at the two end points are

$$\mathbf{P}(0) = \mathbf{P}_0, \mathbf{P}(1) = \mathbf{P}_3 \quad (6)$$

The curve is tangent to the first and last segments of the characteristic polygon.

From Equation (5), the first derivatives when there are 4 control points ($n = 3$) is given by

$$\mathbf{P}'(u) = -3(1-u)^2 \mathbf{P}_0 + (3(1-u)^2 - 6u(1-u)) \mathbf{P}_1 + (6u(1-u) - 3u^2) \mathbf{P}_2 + 3u^2 \mathbf{P}_3 \quad (7)$$

Therefore the tangent vectors at the starting and ending points are

$$\mathbf{P}'(0) = 3(\mathbf{P}_1 - \mathbf{P}_0) \quad (8)$$

$$\mathbf{P}'(1) = 3(\mathbf{P}_3 - \mathbf{P}_2) \quad (9)$$

Similarly, it can be shown that the second derivative at \mathbf{P}_0 is determined by \mathbf{P}_0 , \mathbf{P}_1 , and \mathbf{P}_2 ; or, in general, the r -th derivative at an endpoint is determined by its r neighboring vertices.

◇ Assignment 4

Prove from Equation (4) that in general, the first derivatives at the starting and ending points are given by Equation (10) and (11), respectively:

$$\mathbf{P}'(0) = n(\mathbf{P}_1 - \mathbf{P}_0) \quad (10)$$

$$\mathbf{P}'(1) = n(\mathbf{P}_n - \mathbf{P}_{n-1}) \quad (11)$$

where $(\mathbf{P}_1 - \mathbf{P}_0)$ and $(\mathbf{P}_n - \mathbf{P}_{n-1})$ define the first and last segments of the curve polygon.

◇

Another useful property of the Bezier curve is that **the curve is symmetric with respect to u and $(1-u)$** . This means that the sequence of control points defining the curve can be reversed without change of the curve shape; that is, **reversing the direction of parametrization does not change the curve shape**. This can be achieved by substituting $1-u = v$ in Equation (4) and noticing that $C(n, i) = C(n, n-i)$. This is a result of the fact that $B_{i,n}(u)$ and $B_{n-i,n}(u)$ are symmetric if they are plotted as functions of u .

The interpolation polynomial $B_{i,n}(u)$ has a maximum value of $C(n, i)(i/n)^i (1-i/n)^{n-i}$ occurring at $u = i/n$ which can be obtained from the equation $d(B_{i,n})/du = 0$. **This implies that each control point is most influential on the curve**

shape at $u = i / n$. For example, for a cubic Bezier curve, \mathbf{P}_0 , \mathbf{P}_1 , \mathbf{P}_2 , and \mathbf{P}_3 are most influential when $u = 0, \frac{1}{3}, \frac{2}{3}$, and 1 respectively. Therefore, each control point is weighed by its blending function for each u value.

◇Assignment 5

Derive the maximum values of these 4 functions $B_{0,3}(u)$, $B_{1,3}(u)$, $B_{2,3}(u)$, and $B_{3,3}(u)$, and the values of u at the maximum. Plot these 4 functions and check whether your derivation is correct. ◇

The curve shape can be modified by either changing one or more vertices of its polygon or by keeping the polygon fixed and specifying multiple coincident points at a vertex, as shown in Figure 3. In Figure 3(a), the vertex \mathbf{P}_2 is pulled to the new position \mathbf{P}_2^* . In Figure 3(b), \mathbf{P}_2 is assigned a multiplicity k , that is, 2 or 3 control points are placed on the same position. The higher the multiplicity, the more the curve is pulled toward \mathbf{P}_2 .

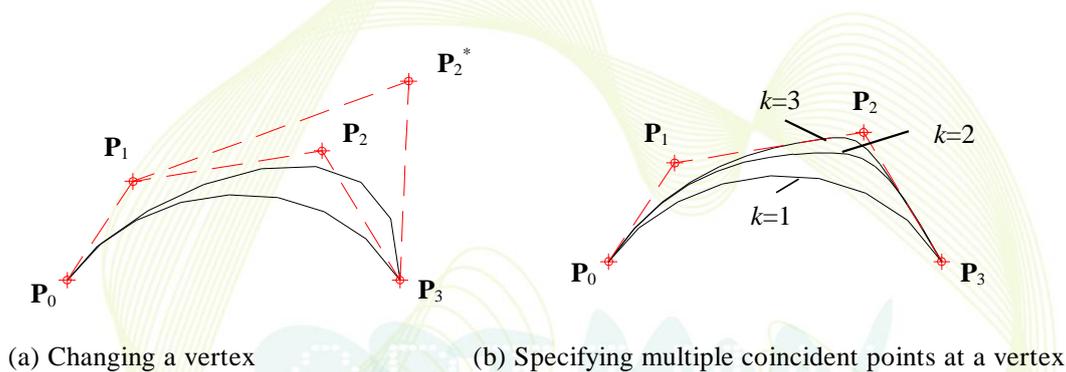


Figure 3. Modifications of cubic Bezier curve.

◇Assignment 6

Use your CAD software to generate figures similar to Figure 3. ◇

Another important property of any Bezier curve is that **it lies within the convex polygon boundary of the control points. This is called the “convex hull property.”** This follows from the properties of Bezier blending function: they are all positive and for any valid value of u the sum of the $B_{i,n}$ functions is always equal to 1 for any degree of Bezier curve. Any curve position is simply a weighted sum of the control point positions.

If the polygon defining a curve segment degenerates to a straight line, the resulting segment must therefore be linear. Also, the size of the convex hull is an upper bound on the size of the curve itself; that is, the curve always lies inside its convex hull. This is a useful property for graphics functions such as displaying or clipping the curve.

A Bezier curve still has some disadvantages. First, the curve does not pass through the control points, which may be inconvenient to some designers. Second, **the curve lacks local control**. It only has the global control nature. If one control point is changed, the whole curve changes. **The degree of the Bezier curve depends on the number of control points**. Only 4 control points are needed for a cubic Bezier Curve. **High order curves may result if there are many control points**.

When curves of many control points are to be generated, **they can be formed by piecing several Bezier sections of lower degree together**. Piecing together smaller sections also gives us better control over the shape of the curve in small regions. Since Bezier curves pass through endpoints, it is easy to match curve sections with C_0 continuity. Also, Bezier curves have the important property that they are tangent to the control polygon at the end points. Therefore we can obtain C_1 continuity by making the two lines joining the endpoint to the adjacent control points collinear.

