the curve but nevertheless provide remarkably effective control of the curve shape. A variant of B-splines uses control points that lie on the curve [200].

21-3 BEZIER METHODS

P. Bézier, of the French firm Régie Renault, pioneered the use of computer modeling of surfaces in automobile design. His UNISURF system, used by designers since 1972, has been applied to define the outer panels of several cars marketed by Renault [46].

Bézier Curves

Bézier defines the curve $P(u)$ in terms of the locations of $n + 1$ control points $\mathbf{p}_i$

$$P(u) = \sum_{i=0}^{n} \mathbf{p}_i B_{i,n}(u) \tag{21-1}$$

where $B_{i,n}(u)$ is a *blending function*

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

and $C(n, i)$ is the binomial coefficient, $C(n, i) = n!/(i!(n - i)!)$. Equation 21-1 is a vector equation: it could be expressed by writing equations for the x, y, and z parametric functions separately:

$$x(u) = \sum_{i=0}^{n} x_i B_{i,n}(u)$$

$$y(u) = \sum_{i=0}^{n} y_i B_{i,n}(u)$$

$$z(u) = \sum_{i=0}^{n} z_i B_{i,n}(u)$$

where the three-dimensional location of the control point $\mathbf{p}_i$ is $[x_i \; y_i \; z_i]$. We shall continue to use the vector notation as in Equation 21-1, with the understanding that the equations can always be expressed using separate scalar functions.

Figure 21-6 shows an example Bézier curve in the plane; the z coordinate of each control point is zero. The particular curve shown uses four control points, connected in the illustration to form an open "polygon."

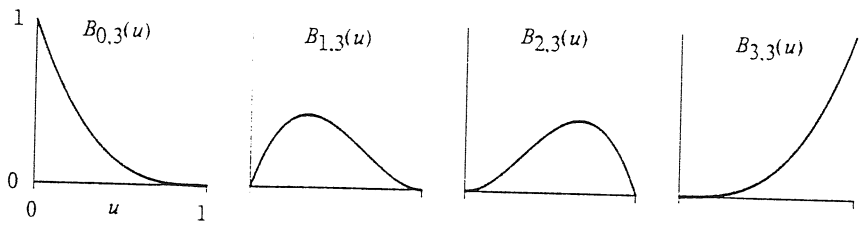

Figure 21-7 The four Bezier blending functions for $n = 3$.

The blending functions are the key to the behavior of Bézier curves. Figure 21-7 shows the four blending functions that correspond to a Bézier curve with four control points. These curves represent the "influence" that each control point exerts on the curve for various values of u. The first control point, $\mathbf{p}_0$, corresponding to $B_{0,3}(u)$, is most influential when $u = 0$; in fact, locations of all other control points are ignored when $u = 0$, because their blending functions are zero. The situation is symmetric for $\mathbf{p}_3$ and $u = 1$. The middle control points $\mathbf{p}_1$ and $\mathbf{p}_2$ are most influential when $u = 1/3$ and $2/3$, respectively.

The equations that define Bézier curves can be readily converted into a program for drawing the curves. A PASCAL program is given in Figure 21-8 that computes $P(u)$ from a vector of control points. The program is designed to correspond to our formulation above; it can easily be made more efficient.

We can evaluate Bézier curves in terms of our list of important properties:

1. *Control points.* At first, it might seem that Bézier curves are hard to use because not all control points lie on the curve. However, the curve is predictably related to the locations of control points—each seems to exert a "pull" on the portion of the curve near it. The control points also satisfy two important mathematical properties: the curve *does* pass through the two endpoints ($\mathbf{p}_0$ and $\mathbf{p}_n$), and the curve is tangent at the endpoints to the corresponding edge of the polygon of control points (e.g., the curve at $\mathbf{p}_0$ is tangent to the vector joining $\mathbf{p}_0$ and $\mathbf{p}_1$).
2. *Multiple values.* The parametric formulation of the Bézier curve allows it to represent multiple-valued shapes. In fact, if the first and last control points coincide, the curve is closed (Figure 21-9).
3. *Axis independence.* A Bézier curve is independent of the coordinate system used to measure the locations of control points.
4. *Global or local control.* These curves do not provide localized control: moving any control point will change the shape of every part of the curve. This can be seen from the blending functions illustrated in Figure 21-7: all functions are nonzero almost everywhere (the two values $u = 0$ and $u = 1$ are exceptions), and consequently the location of each control point will influence the curve location almost everywhere.
5. *Variation-diminishing property.* Bézier curves are variation-diminishing. In

```
function C(n, i: integer): integer;
    var j, a: integer;
begin
    a := 1;
    for j := i + 1 to n do a := a * j;
    for j := 1 to n - i do a := a div j;
    C := a
end;

function BBlend(i, n: integer; u: real): real;
    var j: integer; v: real;
begin
    v := C(n, i);
    for j := 1 to i do v := v * u;
    for j := 1 to n - i do v := v *(1 - u);
    BBlend := v
end;

procedure Bezier(var x, y, z: real; u: real; n: integer; var p: xyzArray);
    var i: integer; b: real;
begin
    x := 0; y := 0; z := 0;
    for i := 0 to n do begin
        b := BBlend(i, n, u);
        x := x + p[i, 1] * b; y := y + p[i, 2] * b; z := z + p[i, 3] * b
    end
end;

procedure DrawCurve;
    var ControlPoints: xyzArray; i: integer; x, y, z: real;
begin
    for i := 0 to 3 do ControlPoints[i, 3] := 0;
    ControlPoints[0, 1] := 0; ControlPoints[0, 2] := 0;
    ControlPoints[1, 1] := 1; ControlPoints[1, 2] := 2;
    ControlPoints[2, 1] := 3; ControlPoints[2, 2] := 2;
    ControlPoints[3, 1] := 4; ControlPoints[3, 2] := 0;

    for i := 0 to 40 do begin
        Bezier(x, y, z, i / 40, 3, ControlPoints);
        if i = 0 then MoveTo(x, y) else LineTo(x, y)
    end
end;
```

Figure 21-8 Procedures for drawing the Bézier curve in Figure 21-6.

addition a curve is guaranteed to lie within the convex hull of the control points that define it [200, 277]. Thus the Bézier curve never oscillates wildly away from its defining control points.

6. *Versatility.* The versatility of a Bézier curve is governed by the number of

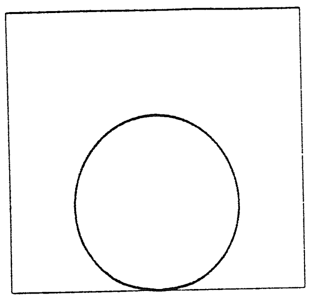
Figure 21-9 A closed Bezier curve specified with six control points. The first and last control points coincide.

control points used. In the example of Figure 21-6, four control points are used ($n = 3$) to determine two parametric cubic polynomial functions that specify x and y values. More control points can always be used to describe more complex shapes, but eventually the high-order polynomial equations become difficult to use because of the lack of localized control.

7. *Order of continuity.* Bézier curves of modest order can be pieced together to describe a more complex curve. In these cases, the joints between the curves must be smooth. To achieve zero-order continuity at a joint, it is necessary only to make the end control points of the two curves coincide (Figure 21-10a). To achieve first-order continuity, the edges of the two polygons adjacent to the common endpoints must lie in a line, as shown in Figure 21-10b (i.e., points $\mathbf{p}_{n-1}$ and $\mathbf{p}_n$ of one curve and $\mathbf{p}_0$ and $\mathbf{p}_1$ of the next curve must all be collinear). Thus it is rather easy for the curve designer to locate control points so as to achieve first-order continuity. Higher-order continuity can also be ensured by geometric constraints on control points, but beyond first-order continuity the constructions become complex.

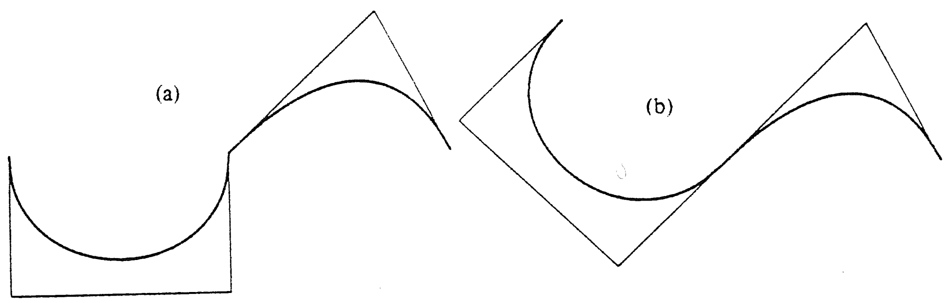

Figure 21-10 Continuity at joints between Bezier curves: (a) zero-order continuity; (b) first-order continuity.