Chapter 6

B-spline Curves

B-splines were investigated as early as the nineteenth century by N. Lobachevsky (see Renyi [?], p. 165); they were constructed as convolutions of certain probability distributions.[1] In 1946, I. J. Schoenberg [?] used B-splines for statistical data smoothing, and his paper started the modern theory of spline approximation. For the purposes of this book, the discovery of the recurrence relations for B-splines by C. de Boor [?], M. Cox [?], and L. Mansfield was one of the most important developments in this theory. The recurrence relations were first used by Gordon and Riesenfeld [?] in the context of parametric B-spline curves.

This chapter presents a theory for arbitrary degree B-spline curves. The original development of these curves makes use of divided differences and is mathematically involved and numerically unstable – see de Boor [?] or Schumaker [?]. A different approach to B-splines was taken by de Boor and Hollig [?]; they used the recurrence relations for B-splines as the starting point for the theory. In this chapter, the theory of B-splines is based on an even more fundamental concept: the *blossoming* method proposed by L. Ramshaw [?] and, in a different form, by P. de Casteljau [?].

6.1 Motivation

The Bézier points of a quadratic Bézier curve may be written as blossom values

$$\mathbf{b}[0,0], \mathbf{b}[0,1], \mathbf{b}[1,1].$$

Based on this, we could get the de Casteljau algorithm by repeated use of the identity $t = (1-t)\cdot 0 + t\cdot 1$. The pairs $[0,0],[0,1],[1,1]$ may be viewed as being obtained from the sequence $0,0,1,1$ by taking successive pairs.

Let us now generalize the sequence $0,0,1,1$ to a sequence u_0, u_1, u_2, u_3.

[1]However, those were only defined over a very special knot sequence.

The quadratic blossom $\mathbf{b}[u,u]$ may be written as

$$\begin{aligned}\mathbf{b}[u,u] &= \frac{u_2-u}{u_2-u_1}\mathbf{b}[u_1,u] + \frac{u-u_1}{u_2-u_1}\mathbf{b}[u,u_2] \\ &= \frac{u_2-u}{u_2-u_1}\big(\frac{u_2-u}{u_2-u_0}\mathbf{b}[u_0,u_1] + \frac{u-u_0}{u_2-u_0}\mathbf{b}[u_1,u_2]\big) \\ &+ \frac{u-u_1}{u_2-u_1}\big(\frac{u_3-u}{u_3-u_1}\mathbf{b}[u_1,u_2] + \frac{u-u_1}{u_3-u_1}\mathbf{b}[u_2,u_3]\big).\end{aligned}$$

This uses the identity

$$u = \frac{u_2-u}{u_2-u_1}u_1 + \frac{u-u_1}{u_2-u_1}u_2$$

for the first step and the two identities

$$u = \frac{u_2-u}{u_2-u_0}u_0 + \frac{u-u_0}{u_2-u_0}u_2$$

and

$$u = \frac{u_3-u}{u_3-u_1}u_1 + \frac{u-u_1}{u_3-u_1}u_3$$

for the second step. Note that we successively express u in terms of intervals of growing size.

Starting with the $\mathbf{b}[u_i,u_{i+1}]$ as input control points, we may rewrite this as:

$$\begin{array}{lll}\mathbf{b}[u_0,u_1] & & \\ \mathbf{b}[u_1,u_2] & \mathbf{b}[u1,u] & \\ \mathbf{b}[u_2,u_3] & \mathbf{b}[u,u_2] & \mathbf{b}[u,u].\end{array}$$

This is a first instance of the de Boor generalization of the de Casteljau algorithm. See Example 6.1 for a detailed computation.

Figure 6.1 illustrates, but using the knot sequence $u_0,u_1,u_2,u_3 = 0,1,3,4$ and $u = 1.5$.

6.2 B-spline Segments

B-spline curves consist of a sequence of polynomial curve segments. In this section, we focus on just one of them.

Let U be an interval in a sequence $\{u_i\}$ of knots. We define a family of intervals U_i^r, each containing U. The interval U_i^r is defined such that:

U_i^r consists of r intervals.

U is the $(r-i)$th interval of U_i^r, with $i = 0$ denoting the first of U_i^r's intervals.

Let $u_0, u_1, u_2, u_3 = 0, 2, 4, 6$. Let the control points be given by

$$\mathbf{b}[u_0, u_1] = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{b}[u_1, u_2] = \begin{bmatrix} 8 \\ 8 \end{bmatrix}, \quad \mathbf{b}[u_2, u_3] = \begin{bmatrix} 8 \\ 0 \end{bmatrix}.$$

Setting $u = 3$, we now compute the point $\mathbf{b}[3, 3]$. At the first level, we have

$$\mathbf{b}[2, 3] = \frac{4-3}{4-0} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \frac{3-0}{4-0} \begin{bmatrix} 8 \\ 8 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \end{bmatrix}$$

and

$$\mathbf{b}[3, 4] = \frac{6-3}{6-2} \begin{bmatrix} 8 \\ 8 \end{bmatrix} + \frac{3-2}{6-2} \begin{bmatrix} 8 \\ 0 \end{bmatrix} = \begin{bmatrix} 8 \\ 6 \end{bmatrix}.$$

Finally,

$$\mathbf{b}[3, 3] = \frac{4-3}{4-2} \begin{bmatrix} 6 \\ 6 \end{bmatrix} + \frac{3-2}{4-2} \begin{bmatrix} 8 \\ 6 \end{bmatrix} = \begin{bmatrix} 7 \\ 6 \end{bmatrix}.$$

Example 6.1: The de Boor algorithm for $n = 2$.

As a consequence, $U_1^1 = U$. We also define $U = [U_0^0, U_1^0]$ and use the term "interval" only if $U_0^0 \neq U_1^0$.

A degree n curve segment corresponding to the interval U is given by $n+1$ control points $\mathbf{d}_i$ which are defined by

$$\mathbf{d}_i = \mathbf{b}[U_i^{n-1}]; \; i = 0, \ldots, n. \tag{6.1}$$

The point $\mathbf{x}(u) = \mathbf{b}[u^{<n>}]$ on the curve is recursively computed as

$$\mathbf{d}_i^r(u) = \mathbf{b}[u^{<r>}, U_i^{n-1-r}]; \quad r = 1, \ldots, n; i = 0, \ldots, n-r \tag{6.2}$$

with $\mathbf{x}(u) = \mathbf{d}_0^n(u)$.[2] This is known as the *de Boor algorithm* after Carl de Boor see [**?**]. See Example 6.2 for the case $n = 3$.

Algorithm (6.2) may alternatively be written as

$$\mathbf{d}_i^r(u) = (1 - t_{i+1}^{n-r-1})\mathbf{d}_i^{r-1} + t_{i+1}^{n-r-1}\mathbf{d}_{i+1}^{r-1}; \quad r = 1, \ldots, n; i = 0, \ldots, n-r, \tag{6.3}$$

where t_{i+1}^{n-r-1} is the local parameter in the interval U_{i+1}^{n-r}.

A geometric interpretation is as follows. Each intermediate control polygon leg $\mathbf{d}_i^r, \mathbf{d}_{i+1}^r$ may be viewed as an affine image of U_{i+1}^{n-r-1}. The point $\mathbf{d}_i^{r+1}$ is then the image of u under that affine map.

[2]This notation is different from the one used in previous editions of this book.

Let part of a knot sequence be given by

$$\ldots\ u_3, u_4, u_5, u_6, u_7, u_8,\ \ldots$$

and let $U = [u_5, u_6]$. The standard blossom computation of $\mathbf{b}[u, u, u]$ proceeds as follows:

$$\begin{array}{llll}
\mathbf{b}[u_3, u_4, u_5] & & & \\
\mathbf{b}[u_4, u_5, u_6] & \mathbf{b}[u, u_4, u_5] & & \\
\mathbf{b}[u_5, u_6, u_7] & \mathbf{b}[u, u_5, u_6] & \mathbf{b}[u, u, u_5] & \\
\mathbf{b}[u_6, u_7, u_8] & \mathbf{b}[u, u_6, u_7] & \mathbf{b}[u, u, u_6] & \mathbf{b}[u, u, u].
\end{array}$$

We now write this as

$$\begin{array}{llll}
\mathbf{b}[U_0^2] & & & \\
\mathbf{b}[U_1^2] & \mathbf{b}[u, U_0^1] & & \\
\mathbf{b}[U_2^2] & \mathbf{b}[u, U_1^1] & \mathbf{b}[u, u, U_0^0] & \\
\mathbf{b}[U_3^2] & \mathbf{b}[u, U_2^1] & \mathbf{b}[u, u, U_1^0] & \mathbf{b}[u, u, u].
\end{array}$$

In terms of control points:

$$\begin{array}{llll}
\mathbf{d}_0 & & & \\
\mathbf{d}_1 & \mathbf{d}_0^1 & & \\
\mathbf{d}_2 & \mathbf{d}_1^1 & \mathbf{d}_0^2 & \\
\mathbf{d}_3 & \mathbf{d}_2^1 & \mathbf{d}_1^2 & \mathbf{d}_0^3.
\end{array}$$

The labeling in the first scheme depends on the subscripts of the knots, while the last two employ a relative numbering.

Example 6.2: The de Boor algorithm for $n = 3$.

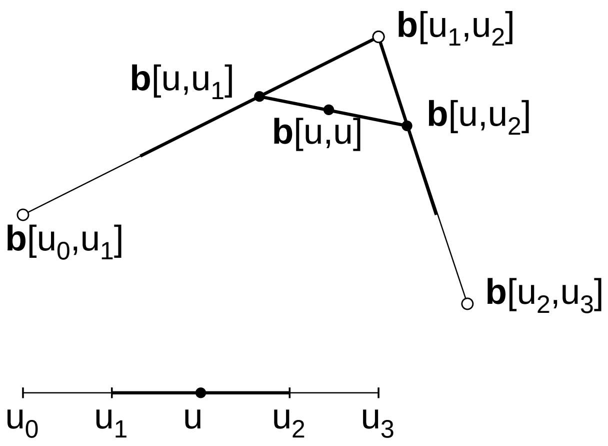

Figure 6.1: The de Boor algorithm: the quadratic case.

For the special knot sequence $0^{<n>}, 1^{<n>}$ and $U = [0, 1]$, the de Boor algorithm becomes

$$\mathbf{d}_i^r(u) = \mathbf{b}[u^{<r>}, 0^{<n-r-i>}, 1^{<i>}]; \quad i = 0, \dots, n-r, \tag{6.4}$$

which is simply the de Casteljau algorithm.

Derivatives of a B-spline curve segment are computed in analogy to the Bézier curve case (60)

$$\dot{\mathbf{x}}(u) = n\mathbf{b}[u^{<n-1>}, \vec{1}]. \tag{6.5}$$

Expanding this expression and using the control point notation, this becomes

$$\dot{\mathbf{x}}(u) = \frac{n}{|U|}(\mathbf{d}_1^{n-1} - \mathbf{d}_0^{n-1}), \tag{6.6}$$

where $|U| = U_1^0 - U_0^0$ denotes the length of the interval U. Thus the last two intermediate points $\mathbf{d}_0^{n-1}$ and $\mathbf{d}_1^{n-1}$ span the curve's tangent, in complete analogy to the de Casteljau algorithm.

Higher derivatives follow the same pattern:

$$\frac{\mathrm{d}^r}{\mathrm{d}u^r}\mathbf{x}(u) = \frac{n!}{(n-r)!}\mathbf{b}[u^{<n-r>}, \vec{1}^{<r>}]. \tag{6.7}$$

xxxxxxxxxxxx

How can we obtain the Bézier polygon of a curve segment from the B-spline representation? Given the blossom $\mathbf{b}$ of the segment, and assuming that xxxxxxx

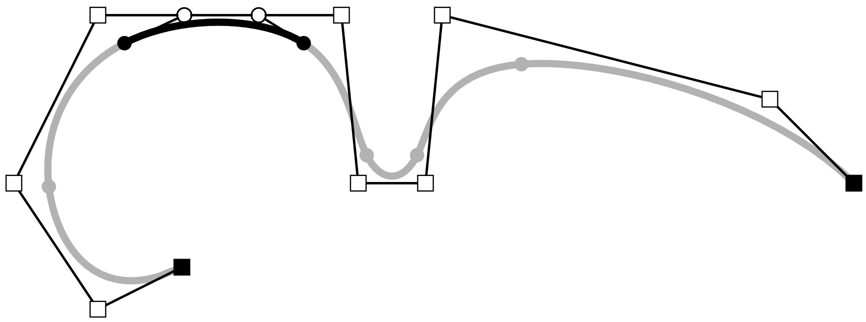

Figure 6.2: Conversion to Bézier form: the Bézierpoints of a cubic segment of a cubic B-spline curve.

6.3 B-spline Curves

A B-spline curve consists of several polynomial curve segments, each of which may be evaluated using the de Boor algorithm. A B-spline curve is defined by

the degree n of each curve segment,

the knot sequence $u_0, \dots, u_K$,

the control polygon $\mathbf{d}_0, \dots, \mathbf{d}_L$ with $L = K - n + 1$.

Some comments: the numbering of the control points $\mathbf{d}_i$ in this definition is *gobal*, whereas in Section 6.2 it was *local* relative to an interval U. Each $\mathbf{d}_i$ may be written as a blossom value with n subsequent knots as arguments. Hence the number of control points equals the number of $n-$tuples in the knot sequence.

There is a different de Boor algorithm for each curve segment. Each is "started" with a set of U_i^n, i.e., by sequences of $n+1$ knots. In order for local coordinates to be defined in (6.3), no subsequent $n+1$ knots may coincide.

In Section 6.2, we assumed that we could find the requisite U_i^n for each interval U. This is only possible if U is "in the middle" of the knot sequence; more precisely, the first possible de Boor algorithm is defined for $U = [u_{n-1}, u_n]$ and the last one is defined for $U = [u_{K-n}, u_{K-n+1}]$. We thus call $[u_n, u_{K-n+1}]$ the *domain* of the B-spline curve.

A B-spline curve has as many curve segments as there are nonzero intervals U in the domain. Example 6.3 illustrates these comments.

For more examples of B-spline curves, see Fgure **??**.

Let $n = 1$, and let the knot sequence be $0, 1, 2$, hence $K = 2$. There will be control points $\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_3$. The curve's domain is $[u_0, u_3]$, and there are two linear curve segments.
Let $n = 2$ with the knot sequence $0, 0.2, 0.4, 0.5, 0.7, 1$, hence $K = 5$. There will be control points $\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3, \mathbf{d}_4$ and three quadratic curve segments. If we now change the knot sequence to $0, 0.2, 0.45, 0.45, 0.7, 1$, then the number of curve segments will drop to two.

Example 6.3: Some examples of B-spline curve definitions.

In Figure **??**, just one de Boor step is carried for the parameter value $\hat{u}$. The two new resulting points, in blossom notation, are $\mathbf{b}[u_1, \hat{u}]$ and $\mathbf{b}\hat{u}, u_2]$. Let us now consider the points

$$\mathbf{b}[u_0, u_1], \mathbf{b}[u_1, \hat{u}], \mathbf{b}\hat{u}, u_2].$$

These are the B-spline control points of our curve $\mathbf{b}[u, u]$ for the interval $[u_1, \hat{u}]$! Similarly, the B-spline control points for the interval $[\hat{u}, u_2]$ are given by

$$\mathbf{b}[u_1, \hat{u}], \mathbf{b}\hat{u}, u_2], \mathbf{b}[u_2, u_3].$$

Example 6.4: Knot insertion.

6.4 Knot Insertion

Consider a B-spline curve segment defined over an interval U. It is defined by all blossom values $\mathbf{d}[U_i^n]$; $i = 0, \ldots, n$ where each n-tuple of successive knots U_i^n contains at least one of the endpoints of U. If we now split U into two segemtns by inserting a new knot $\hat{u}$, the curve will have two corresponding segments. What are the control points for these two egments? The answer is surprisingly simple: all blossom values $\mathbf{d}[\hat{U}_i^n]$; $i = 0, \ldots, n+1$ where each n-tuple of successive knots $\hat{U}_i^n$ contains at least one of the endpoints of U. This result is due to W. Boehm [**?**], although it was not originally derived using blossoms.

We begin with an example.

The knot insertion result works since B-spline control points are nothing but blossom values of successive knots – now they involve the new knot $\hat{u}$. We may also view the process of knot insertion as one level of the de Boor algorithm.

6.5 Degree Elevation

We may degree elevate in (almost) the same way we could degree elevate Bézier curvesusing (4.77). The difference: A given B-spline is a piecewise degree n curve over a given knot sequence. Its differentiablility is determined

Let a cubic B-spline curve be defined over $\{u_0 = u_1 = u_2, u_3, \ldots\}$. Then the interval $[u_4, u_5]$ corresponds to $[\hat{u}_7, \hat{u}_8]$. The new control point $\hat{\mathbf{d}}_4$ is computed as follows:

$$\begin{aligned}
\hat{\mathbf{d}}_4 &= \hat{\mathbf{d}}_7[\hat{u}_4, \hat{u}_5, \hat{u}_6, \hat{u}_7] \\
&= \frac{1}{4}\big(\mathbf{d}_4[\hat{u}_4, \hat{u}_5, \hat{u}_6] + \mathbf{d}_4[\hat{u}_4, \hat{u}_5, \hat{u}_7] + \mathbf{d}_4[\hat{u}_4, \hat{u}_6, \hat{u}_7] + \mathbf{d}_4[\hat{u}_5, \hat{u}_6, \hat{u}_7]\big) \\
&= \frac{1}{2}\big(\mathbf{d}_4[u_3, u_3, u_4] + \mathbf{d}_4[u_3, u_4, u_4]\big).
\end{aligned}$$

For the last step, we have utilized $\hat{u}_4 = \hat{u}_5 = u_3$ and $\hat{u}_6 = \hat{u}_7 = u_4$.

Example 6.5: B-spline degree elevation and blossoms.

by the knot multiplicites. If we write it as a piecewise degree $n+1$ curve, we need to increase the multiplictiy of every knot by one, thus maintaning the original differentiability properties. For example, if we degree elevate a C^0 piecewise linear curve to piecewise quadratic, it is still C^0. But for a piecewise quadratic to be C^0, it has to have double knots. Let us denote the knots in this augmented knot sequence by $\hat{u}_i$.

Let V^n be a sequence of n real numbers $v_1, \ldots, v_n$. Let $V^n|v_i$ denote the sequence V^n with the value v_i removed. Then the degree $n+1$ blossom $\hat{\mathbf{b}}$ may be expressed in terms of the degree n blossom $\mathbf{b}$ via

$$\hat{\mathbf{b}}[V^{(n+1)}] = \frac{1}{n}\big(\mathbf{b}[V^{(n+1)}|v_1] + \ldots + \mathbf{b}[V^{(n+1)}|v_n]\big). \tag{6.8}$$

The proof is identical to that for degree elevation of Bézier curves. The control points are then recovered from the blossom as before.

6.6 Greville Abscissae

Let $l[u] = u$ be the blossom of the linear function u (nothing parametric here). If we want to write this linear blossom as a quadratic one: $q[u, v] = l[u]$, we easily see that

$$l^2[u, v] = \frac{1}{2}l[u] + \frac{1}{2}l[v]$$

gives the desired quadratic form of our linear blossom. If we asked for a cubic form $l^{(2)}[u, v, w]$ of $l[u]$, we find that

$$l^3[u, v, w] = \frac{1}{3}q[v, w] + \frac{1}{3}q[u, w] + \frac{1}{3}q[u, v].$$

If we denote a degree $n+1$ version of the linear blossom by $l^n[V^n]$ with $V^n = v_1, \ldots, v_n$, it follows that

$$l^n[V^n] = \frac{1}{n}(v_1 + \ldots v_n)\big].$$

The proof is by induction and was anchored by the examples above. The inductive step starts with the degree elevation formula (**??**):

$$l^{n+1} = \frac{1}{n+1}\big[\frac{1}{n}(\star{+}v_2{+}\ldots{+}v_{n+1}){+}\frac{1}{n}(v_1{+}\star{+}\ldots{+}v_{n+1}){+}\frac{1}{n}(v_1{+}\ldots{+}\star{+}v_{n+1}){+}\frac{1}{n}(v_1{+}\ldots{+}v_n{+}\star)\big],$$

where $\star$ denotes an omitted term.[3] This is easily transformed to

$$l^{n+1} = \frac{1}{n+1}\big[v_1 + \ldots + v_{n+1})\big],$$

thus finishing the proof.

If we are given a knot sequence $u_0, \ldots, u_K$ and a degree n, then we know that any B-spline function $d(u)$ has control vertices $d[u_0, \ldots, u_{n-1}], \ldots, d[u_{K-n+1}, \ldots, u_K]$. In the case of a linear function l, we thus have control vertices

$$\frac{1}{n}(u_0 + \ldots + u_{n-1}), \ldots \frac{1}{n}(u_{K-n+1} + \ldots + u_n).$$

These terms are called *Greville abscissae* and are abbreviated as

$$\xi_i = \frac{1}{n}(u_i + \ldots + u_{i+n-1}).$$

A *nonparametric* B-spline function $d(u)$ may thus be written as a parametric curve with points

$$\mathbf{d}_i = \begin{bmatrix} \xi_i \\ d_i \end{bmatrix}; \quad i = 0, \ldots, L$$

with the usual $L = K - n + 1$. Figure **??** gives an example.

For the special case of the knot sequence $0^{<n>}, 1^{<n>}$, we obtain $\xi_i = \frac{i}{n}$, as already encountered in Section 4.31.

6.7 Smoothness

A B-spline curve consists of several polynomial segments $\mathbf{b}^U[u^{<n>}]$, one for each interval U. What is the smoothness of this piecewise curve?

Figures **??**, **??**, **??** show how knot multiplicities affect smoothness.

---

[3] We do not have to work with augmented knot sequences here since we always deal with *one* linear function.

In general, if a knot $\hat{u}$ is of multiplicity r, then a B-spline curve of degree n is has smoothness C^{n-r} at that knot. This follows since from considering the osculants at $\hat{u}$. The highest order osculant is given by

$$\mathbf{o}^{n-r}(u) = \mathbf{b}[\hat{u}^{<r>}, u^{<n-r>}],$$

assuring continuity of derivatives up to order $n-r$. Higher order continuity is possible, but cannot be guaranteed.

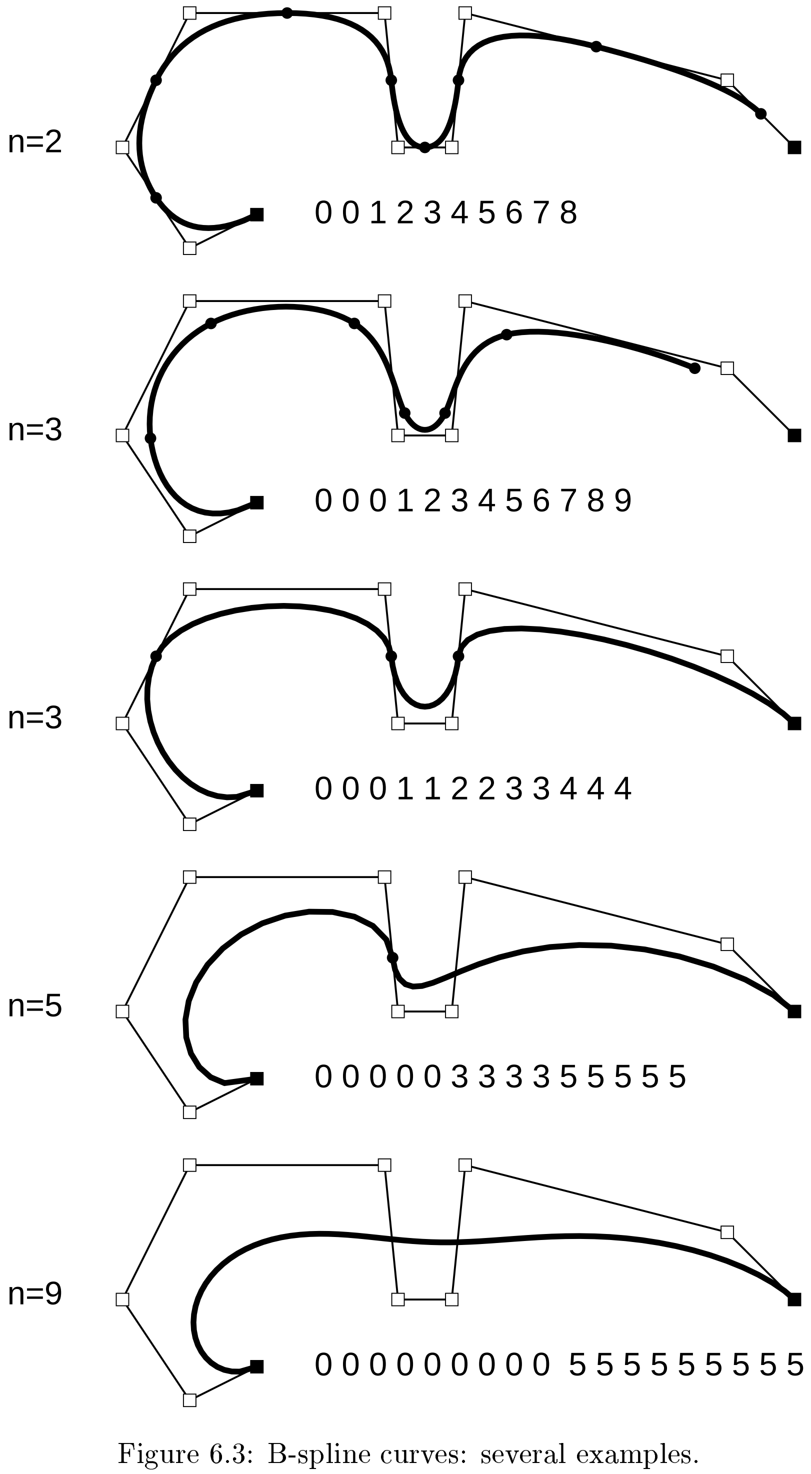

Figure 6.3: B-spline curves: several examples.

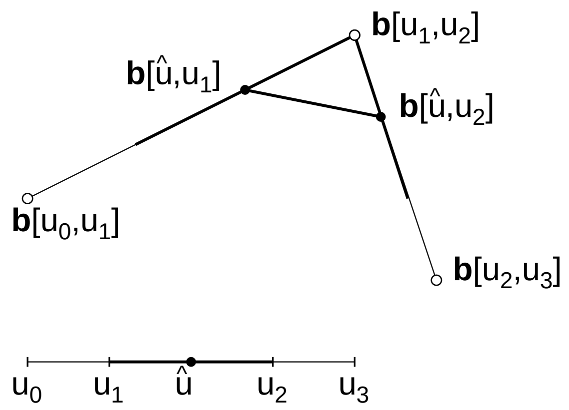

Figure 6.4: Knot insertion: a quadratic example.

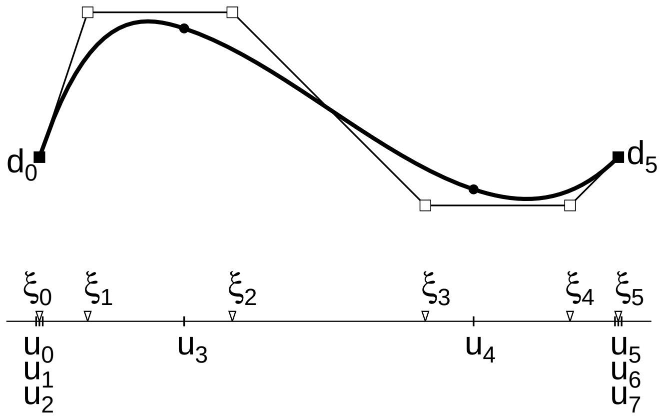

Figure 6.5: Nonparametric B-spline curves: a cubic example.

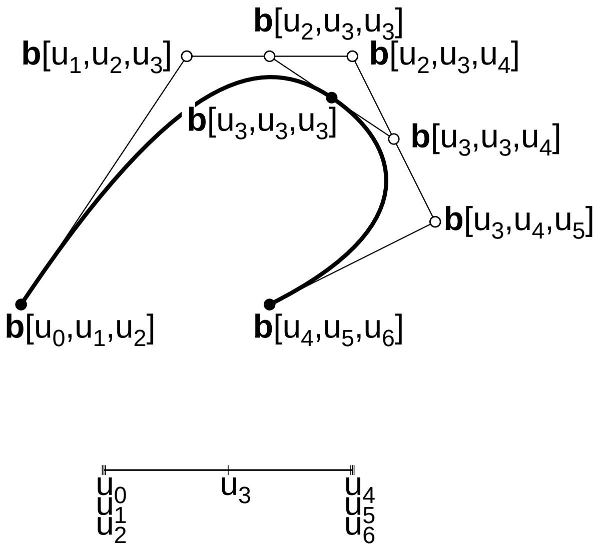

Figure 6.6: Smoothness: an interior knot of multiplicity one results in a C^2 piecewise cubic curve.

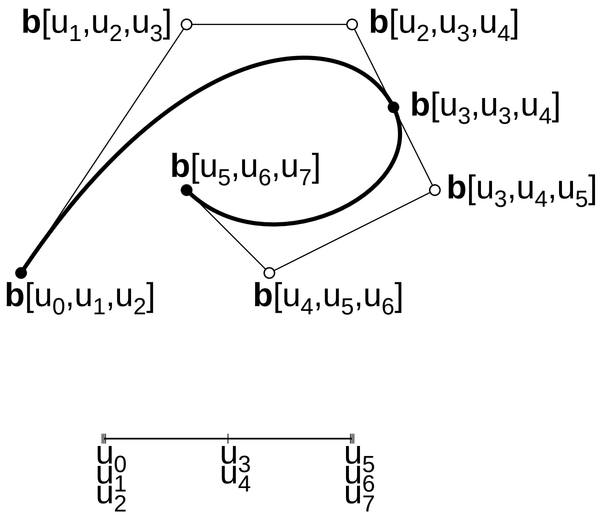

Figure 6.7: Smoothness: an interior knot of multiplicity two results in a C^1 piecewise cubic curve.

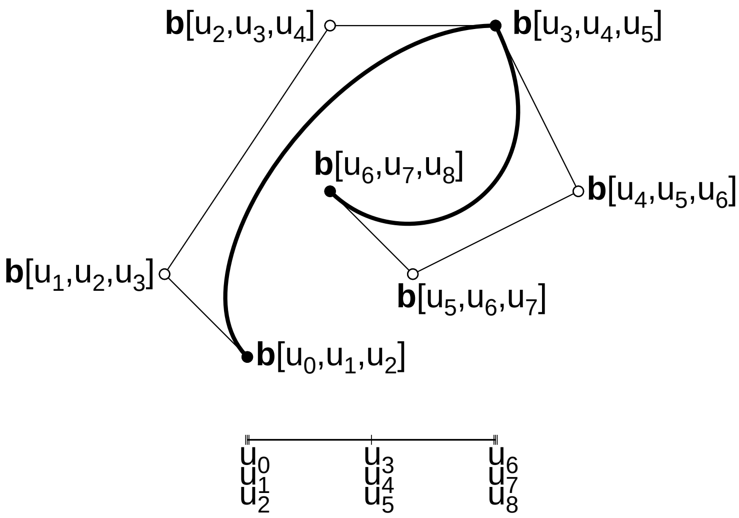

Figure 6.8: Smoothness: an interior knot of multiplicity three results in a C^0 piecewise cubic curve.