Ten lectures on splines

Splines are important,
splines are useful,
splines are easy,
but most of it all, splines are beautiful.

Jesús M. Ruiz, 2006

Splines are piecewise polynomial functions. To represent them in the suitable form one must develop carefully the algorithmic aspects in their construction, but in the end, all reduces to understand deeply the basic notion on which affine geometry is supported: *linear interpolation*. These lectures try to justify this claim of ours in a convincing (and practical) manner. Also, we pretend to make clear that splines are easy and beautiful (that they are important and useful everybody agrees).

Lecture 1. Linear interpolation

We will be dealing with *real affine spaces* $\mathbb{R}^n$, mostly $n = 1, 2$ and 3. Recall that in $\mathbb{R}^n$ all operations are componentwise defined.

Points in $\mathbb{R}^n$ are joined by vectors. Namely, two points $a, b \in \mathbb{R}^n$ are joined by the vector $u = \overrightarrow{ab} = b - a$. We can also say that a is translated: $a + u = b$, and we see that any point b is the traslation of a fixed one a.

(1.1) Linear interpolation. Let us see what results from the most elementary operation: *scaling*. If we replace u by tu for a scalar $t \in \mathbb{R}$ we get:

$$c = a + tu = a + t(b - a) = (1 - t)a + tb = \alpha a + \beta b.$$

Thus, we indeed obtain a third point c from the two given a and b by a kind of linear combination. However the coefficients α and β are subject to the restriction

$$\alpha + \beta = (1 - t) + t = 1.$$

This is what we call *linear interpolation*, an operation that produces a third point c from two points a and b. The scalar t is called the *ratio*, and quite obviously it is denoted as

$$t = \frac{c - a}{b - a}.$$

(1.2) Lines. The natural fact that through any two points $a, b \in \mathbb{R}^n$ passes a line L is now clear: L consists of all points generated by linear interpolation from the two distinct points a and b. Namely,

$$L = \{c = (1 - t)a + tb \in \mathbb{A}^3 : t \in \mathbb{R}\} = \{c = a + tu \in \mathbb{A}^3 : t \in \mathbb{R}\} = a + [u],$$

where $[u] \subset \mathbb{R}^n$ is the linear span generated by the vector $u = b - a$. This linear span is the *direction* of L, denoted $\overrightarrow{L}$, and L can (and should) be thought as the traslation to a of that span: $L = a + \overrightarrow{L}$. Note that since the two points are distinct, $u \neq 0$, and $\overrightarrow{L}$ has dimension 1.

From this description, it is clear that $L = a' + [u']$ for any point $a' \in L$ and vector $u' \in [u]$, or in other words, that L is generated by any two distinct points $a', b' \in L$ (set $u' = b' - a'$). Or, we can also say that interpolations of points in L remain always in L.

Points in the same line are called *collinear*. Now suppose that $\alpha < \beta$ are real numbers. Three points $a, b, x \in \mathbb{R}^n$ are collinear if and only if there is a third real number s such that

$$x = \frac{\beta - s}{\beta - \alpha}a + \frac{s - \alpha}{\beta - \alpha}b.$$

This is nothing but linear interpolation with ratio $t = \frac{s-\alpha}{\beta-\alpha}$. However this formulation works for any interval $[\alpha, \beta]$ and will we used a lot. Also, we will say that *(the points)* a, b, x *are in the same ratio that (the scalars)* α, β, s.

Two lines L and L' are *parallel* when $\overrightarrow{L} = \overrightarrow{L}$.

(1.3) Planes. Suppose here $n \geq 2$. Then we can consider a line L and a point a outside, and think of the *plane* $\pi \subset \mathbb{R}^n$ generated by them. Quite naturally, we start with all points

$$p = (1-t)a + tq \quad \text{ where } q \in L.$$

Let us compute a little bit. Pick $b, c \in L$, so that

$$L = \{q = (1-s)b + sc : \ s \in \mathbb{R}\}.$$

Then for each p as above we have:

$$p = (1-t)a + tq = (1-t)a + t((1-s)b + sc) = (1-t)a + t(1-s)b + tsc,$$

or, more simply,

$$p = \alpha a + \beta b + \gamma c, \quad \alpha + \beta + \gamma = 1.$$

Thus, we find again the type of coefficients that add up to 1. But now we come to a difficulty to go backwards. Indeed, given α, β, γ, then q is determined by the ratio

$$s = \frac{\gamma}{1-\alpha}$$

whenever $\alpha \neq 1$. Let us analise this. For our starting p, that $\alpha = 0$ means $t = 1$, hence $p = a$. However, we cannot resource to the initial form of p, as we are trying to recover it. Thus, if $\alpha = 0$ we can only say that $\beta + \gamma = 0$ and

$$p = a + \gamma(c-b) \in a + \overrightarrow{L} = L'.$$

This L' is the line parallel to L through a, and obviously L' should be in π. But no point in L' comes from linear interpolation of a and a point of L, hence linear interpolation is not enough to describe π. Thus the plane π is in fact generated by a, b, c as follows:

$$\pi = \{p = \alpha a + \beta b + \gamma c : \ \alpha + \beta + \gamma = 1, \ \alpha, \beta, \gamma \in \mathbb{R}\}.$$

Next, setting $u = b - a, v = c - a$, we can write any point $p \in \pi$ in the form:

$$p = \alpha a + \beta b + \gamma c = a + \beta(b-a) + \gamma(c-a) = a + \beta u + \gamma v,$$

and so $\pi = a + [u, v]$, where $[u, v]$ stands for the linear span of the vectors u, v. Again, $[u, v]$ is called the *direction* of π and denoted by $\overrightarrow{\pi}$, so that $\pi = a + \overrightarrow{\pi}$. Note that $\overrightarrow{\pi}$ must have dimension 2 (if, say $v \in [u]$, then also $0 \neq w = c - b = v - u \in [u]$ and $a = b - u = b + \lambda w \in L$), which matches twe call π a plane.

It is not difficult to see that π is generated by any three independent points in it, or by any line and point outside lying in π. Thus, π is stable by linear interpolation.

Two planes π, π' are *parallel* if $\overrightarrow{\pi} = \overrightarrow{\pi'}$; a line L is parallel to a plane π if $\overrightarrow{L} \subset \overrightarrow{\pi}$.

(1.4) Barycentric combinations. In the preceding computations we were picking points quite carefully, but we need not. To see this, replace scalings as used to introduce interpolation by arbitrary linear combinations, say

$$p = a + (\alpha_1 u_1 + \cdots + \alpha_m u_m), \quad a \in \mathbb{A}^n,\, u_i \in \mathbb{R}^n.$$

We can rewrite this as follows.

$$p = \alpha_0 a_0 + \alpha_1 a_1 + \cdots + \alpha_m a_m,$$

where $a_0 = a, \alpha_0 = 1 - \sum_{i=1}^n \alpha_i$ and $a_i = a_0 + u_i$.

We claim that this is an *iterated linear interpolation,* but iteration means that whenever you obtain a new point you interpolate it with all you have previously obtained, not only the initial data. We do not need to discuss this further, but let us review in this light the obstruction described when introducting planes:

$$a \neq p = a + \beta b + \gamma c,\, \beta + \gamma = 0,$$

with all notations there. Clearly, β and γ cannot be both -1, hence suppose for instance $\gamma \neq -1$, and then:

$$p = (1+\gamma)a' - \gamma b, \qquad \text{where } a' = \tfrac{1}{1+\gamma}a' + \tfrac{\gamma}{1+\gamma}c,$$

that is, a' is an interpolation of a, b, c, and then p is another of a', b, c.

Henceforth, when we say that any given points a_i *generate* another p, we mean they generate it by *barycentric combination* (that is, iterated linear interpolation):

$$p = \alpha_0 a_0 + \cdots + \alpha_m a_m, \quad \sum_i \alpha_i = 1.$$

(1.5) Segments and convex hulls. Everybody knows what a segment is, but here there is the precise description: the *segment* that joins two points $a, b \in \mathbb{R}^n$ is

$$[a, b] = \{p = (1-t)a + tb :\, 0 \leq t \leq 1, t \in \mathbb{R}\} = a + \{t(b-a) :\, 0 \leq t \leq 1\},$$

that is, all linear interpolations of a and b with *non-negative* coefficients.

Now, a set is called *convex* when it contains all segments joining any two of its points. We see easily that *a set is convex if and only if it is stable under barycentric combinations with non-negative coefficients.*

Indeed, the definition applies to barycentric combinations of two points, and then one argues by induction writting:

$$\alpha_0 a_0 + \alpha_1 a_1 + \cdots + \alpha_n a_n = \alpha_0 a_0 + (1-\alpha_0)\Big(\frac{\alpha_1}{1-\alpha_0} a_1 + \cdots + \frac{\alpha_n}{1-\alpha_0} a_n\Big)$$

(at most one coefficient is 1, say α_0 is not).

The *convex hull* $\mathrm{ch}(S)$ of a set $S \subset \mathbb{R}^n$ is the smallest convex set containing it, and can be easily computed as follows:

$$\mathrm{ch}(S) = \{p = \alpha_1 a_1 + \cdots + \alpha_n a_n : \sum \alpha_i = 1,\ 0 \le \alpha_i \le 1,\ a_i \in S\}.$$

To prove this formula it suffices to see that the set in the right hand side is convex, but given $p = \alpha_1 a_1 + \cdots + \alpha_n a_n$ and $q = \beta_1 b_1 + \cdots + \beta_m b_m$, for $0 \le t \le 1$ we have

$$(1-t)p + tq = \sum_i (1-t)\alpha_i a_i + \sum_j t\beta_j b_j,$$

and this is a barycentric combination with non-negative coefficients, because

$$\sum_i (1-t)\alpha_i + \sum_j t\beta_j = (1-t)\cdot 1 + t \cdot 1 = 1.$$

In particular, for finite sets of points we have:

$$\mathrm{ch}(a_1, \ldots, a_n) = \{p = \alpha_1 a_1 + \cdots + \alpha_n a_n : \sum_i \alpha_i = 1,\ 0 \le \alpha_i \le 1\}$$

For instance, the convex hull of two distinct points is the segment they define, the convex hull of three non-collinear points is the triangle they form, the convex hull of four non-coplanar points is the tetrahedron they determine.

Lecture 2. Splines

Consider a compact interval $I = [a, b] \subset \mathbb{R}$, $a < b$. Continuous functions on I form a vector space $\mathcal{C}([a.b])$ by pointwise operations:

$$(\alpha f + \beta g)(t) = \alpha f(t) + \beta g(t) \quad \text{for } \alpha, \beta \in \mathbb{R}.$$

This space contains many distinguished subspaces: differentiable functions, smooth functions, analytic functions... , and in particular *polynomial functions* and *piecewise polynomial continuous functions.* The latter we call *splines.*

Thus, let us start be showing how polynomials come from linear interpolation.

(2.1) Blossoms.N A mapping

$$\theta : \mathbb{R}^n \to \mathbb{R} : (t_1, \ldots, t_n) \mapsto \theta(t_1, \ldots, t_n)$$

is called *symmetric n-ary affine form* when (i) it preserves linear interpolation in each variable separately, and (ii) it disregards the order of variables, that is

$$\theta(t_1, \ldots, t_n) = \theta(t_{\sigma_1}, \ldots, t_{\sigma_n})$$

for every permutation σ of $1, \ldots, n$.

There is an easy way to produce functions from forms. In fact, consider the linear mapping:

$$\mathsf{bl} : \theta(t_1, \ldots, t_n) \mapsto f(t) = \theta(t, \ldots, t),$$

and let us analise its image.

To start with, that image is a vector space E that contains each power t^m, $m = 0, \ldots, n$. Indeed, t^m is the image of the form

$$\theta(t_1, \ldots, t_n) = \tfrac{n-m!}{n!} \sum_{\varepsilon} t_{\varepsilon_1} \cdots t_{\varepsilon_m},$$

where the ε's range among the variations of order m of $1, \ldots, n$. But those powers are clearly linearly independent, hence the image has at least dimension $n + 1$.

Now, since a form θ preserves interpolation in each variable, it is fully determined by the values

$$\theta(\varepsilon_1, \ldots . \varepsilon_n), \quad \text{where } \varepsilon_k = 0 \text{ or } 1.$$

But θ is also symmetric, so we can reorder the ε_i's, and it is enough to know the values

$$\theta_k = \theta(0, \overset{n-k}{\ldots}, 0, 1, \overset{k}{\ldots}, 1), \quad \text{for } k = 0, \ldots, n.$$

This means that the linear mapping: $\theta \mapsto (\theta_0, \ldots, \theta_n) \in \mathbb{R}^{n+1}$ is injective, hence the dimension of the space of forms $\mathcal{F}$ is $\leq n + 1$. Thus

$$n + 1 \geq \dim(\mathcal{F}) \geq \dim(E) \geq n + 1,$$

and we conclude that bl is an isomorphism, and the image consists of all functions

$$f(t) = a_0 + a_1 t + \cdots + a_n t^n, \quad \text{where } a_i \in \mathbb{R}.$$

In other words, the image E is the vector space of all polynomials of degree $\leq n$, which we denote by $\mathcal{P}_n([a.b])$.

Using this isomorphism bl, we say that θ is the *blossom* of $f(t) = \theta(t, \ldots, t)$.

(2.2) Derivation. We can use differentiation to represent polynomial functions in different ways. As is well known, it holds

$$\frac{d}{dt} t^k = k t^{k-1} \quad \text{for } k \geq 1.$$

and by iteration one gets all derivatives of higher order, $\frac{d^r}{dt^r}$ (for $r = 0$ the derivative is the polynomial itself). Note that $\frac{d^n}{dt^n}$ is constant on polynomials of degree $\leq n$ is constant, and all subsequent derivatives are null. In fact, a small algebraic computation shows that we have a *Taylor formula* as follows

$$P(t) = P(c) + \tfrac{dP}{dt}(a)(t-a) + \tfrac{1}{2}\tfrac{d^2P}{dt^2}(a)(t-a)^2 + \cdots + \tfrac{1}{i!}\tfrac{d^iP}{dt^i}(a)(t-a)^i + \cdots .$$

which as remarqued before stops at the $(n+1)$-th term. We also remark here the immediate but important fact that the derivatives of the polynomial at any point are a linear function of its coefficients in the above Taylor espansions.

We also see that polynomials of degree $\leq n$ can also we generated by the basis

$$1, t-a, (t-a)^2, \ldots, (t-a)^n,$$

the coordinates being (up to some universal constants) the derivatives at a. We see also that a polynomial of degree $\leq n$ is completely determined by its first $n+1$ derivatives at any fixed point a (this is the so-called *identity principle*N). ∎

Finally, let us give the precise formalism for splines.

(2.3) The space of splines. A *spline* is a continuous function $f(t)$ equiped with a partition $a = t_0 < t_1 < \cdots < t_{m+1} = b$, such that each restriction to a subinterval $[t_i, t_{i+1}]$ is polynomial. Such a restriction is called *branch;* sometimes we say *right branch at t_i* and *left branch at t_{i+1}*. Each interior point in the partition, t_i, $i = 1, \ldots, m$, is called a *joint.*

The *degree* of a spline is the largest degree of its branches; degree $1, 2, 3$ splines are called respectively *linear, quadratic, cubic.*

For splines continuity simply means that the two consecutive branches map the common joint to the same value. Thus, differentiability at joints is the crucial matter. We will say that a spline $f(t)$ as above is of class $(n - r_1, \ldots, n - r_m)$ if it is a function of class $n - r_i$ at the joint t_i, for $i = 1, \ldots, m$. Then:

Proposition 2.4 *N The vector space* $\mathcal{SP}_n^{r_1\cdots r_m}([a.b])$ *of splines of degree* n *and class* $(n-r_1,\ldots,n-r_m)$ *has dimension* $n+1+r_1+\cdots+r_m$.

Proof. We must carefully exhibit all functions $f(t)$ that are polynomial of degree $\leq n$ on each interval $[t_i, t_{i+1}]$ for $i = 0,\ldots,m$. In other words, all sequences of degree $\leq n$ polynomials $P_0(t), P_1(t),\ldots,P_m(t)$, with the condition that $P_{i-1}(t)$ and $P_i(t)$ have the same $n-r_i$ derivatives at the joint t_i, for $i = 1,\ldots,m$. We will represent the polynomial $P_i(t)$ in the basis

$$1, t-t_i, (t-t_i)^2, \ldots, (t-t_i)^n,$$

which to start with gives $n+1$ coordinates. But it is not that easy. For $P_0(t)$ there is no problem, we choose it arbitrarily and have $n+1$ coordinates indeed. Next for $P_1(t)$, we know its derivatives at t_1 till order $n-r_1$: they are those of $P_0(t)$, which are in fact linear functions of the coordinates chosen for $P_0(t)$. Thus for $P_1(t)$ we have $n-r_1+1$ coordinates depending on those of $P_0(t)$, and the $n+1-(n-r_1+1) = r_1$ remaining ones are free. On the whole, for $P_0(t), P_1(t)$ we have $n+1+r_1$ free coordinates. It is clear that for $P_2(t)$ we will get $n+1-(n-r_2+1) = r_2$ free coordinates more, and so on so forth. In the end, we get

$$n+1+r_1+\cdots+r_m$$

free coordinates (and the others given by linear combinations of these). We conclude that the dimension is what the statement claimed. ∎

Note now that by the identity principle if a spline of degree n has class n at a joint, then the corresponding two consecutive branches must be actually defined by the same polynomial tuple, and the joint is actually superfluous. This gives the compromise between the freedom of arbitrary patching, and the smoothness of the differentiabilty class at the joints: $n-1$ is the maximal differentiability class of a (true) spline of degree n.

Finally, let us recall the standar costumary identification of a function $f(t) \in \mathcal{C}([a.b])$ with its graph, which is the *planar curve*

$$\Gamma_f = \{(t, f(t)) \in \mathbb{R}^2 : a \leq t \leq b\}.$$

This view makes geometric (hence much clearer) many discussions.

Lecture 3. The de Casteljeau interpolation algorithm

Here we concentrate on polynomial functions on a given interval $I = [a, b]$, and describe a basic algorithm to produce all of a given degree, jointly with its blossom. To that end, consider the following abscissas, uniformly distributed in our interval

$$\zeta_k = \tfrac{n-k}{n} a + \tfrac{k}{n} b, \quad k = 0, \dots, n+1.$$

Then choose any *control ordinates* $c_0, \dots, c_n$, which define in $\mathbb{R}^2$ a polygon with vertices

$$b_0 = (\zeta_0, c_0), \dots, b_{n+1} = (\zeta_{n+1}, c_{n+1}).$$

We will produce a function whose graph is *controlled* by this polygon,

This algorithm is represented by a triangular matrix.

(3.1) The de Casteljeau triangle. We consider the following diagram:

$$\begin{array}{cccccccccc}
0 & 1 & \dots & k-1 & k & \dots & \dots & n-1 & n \\
 & & & & & & & & \\
b_0^0 & & & & & & & & \\
b_1^0 & b_1^1 & & & & & & & \\
\vdots & \vdots & & \ddots & & & & & \\
\vdots & \vdots & & & \ddots & & & & \\
\vdots & \vdots & & b_{i-1}^{k-1} & & \ddots & & & \\
 & & & & \searrow & & & & \\
\vdots & \vdots & & b_i^{k-1} & \to b_i^k & & \ddots & & \\
b_{n-1}^0 & b_{n-1}^1 & & & & & & b_{n-1}^{n-1} & \\
b_n^0 & b_n^1 & & \cdots & \cdots & \cdots & \cdots & b_{n-1}^n & b_n^n
\end{array}$$

We call this trangle the *de Casteljeau triangle,* and its last entry b_n^n its *target.*

Notice also that if we pin up some entry, and think of the triangle with target at it, it is another de Casteljeau triangle. This simple remark makes many properties easy to check. Also, we recommend the reader to keep present a copy of this large triangle, and imagine parts of it to catch up with what is coming.

(3.2) The de Casteljau algorithm. What we depicted above is just a diagram, and we must define each entry properly. We do this by interpolation, of course:

$$b_i^k(t) = \frac{b-t}{b-a}\, b_{i-1}^{k-1}(t) + \frac{t-a}{b-a}\, b_i^{k-1}(t).$$

(inspired by the Pascal formula for combinatorial numbers). We will say that *the triangles are built on* $[a, b]$.

We will apply this interpolation scheme taking for entries b_k^0 the points b_k, or equivalently, their coordinates ζ_k and c_k.

(3.3) Linear precision. N For $b_k^0 = \zeta_k = a + \frac{k}{n}(b-a)$, we easily get by induction: (i) $b_i^k - b_{i-1}^k = \frac{1}{n}(b-a)$, and (ii) $b_i^k = b_{i-1}^{k-1} + \frac{1}{n}(t-a)$. Then repeating the last formula we conclude

$$b_n^n(t) = b_{n-1}^{n-1} + \frac{1}{n}(t-a) = b_{n-2}^{n-2} + \frac{2}{n}(t-a) = \cdots = b_0^0 + \frac{n}{n}(t-a) = \zeta_0 + (t-a) = t.$$

In other words, starting with our *uniformly spaced* abscissas, we obtain the polynomial t. This is the *linear precision* of the algorithm. Now, for the chosen control ordinates, from $b_k^0 = c_k$, we get a polynomial $f(t) = b_n^n(t)$ of degree $\leq n$, and summing up, we obtain a mapping $(t, f(t))$ that is in fact a graph. After this remark, what matters is the function $f(t) \in \mathcal{P}_n([a.b])$. Hence from now on, we can disregard abscissas in computations, and always apply the de Casteljeau triangle and algorithm to control *ordinates.*

(3.4) The multivariate de Casteljeau algorithm. It is easy to find out the blossom $\mathcal{B}(t_1, \ldots, t_n)$ of $f(t)$ (2.1, p.5). Indeed, just define a multivariate de Casteljeau algorithm by:

$$b_i^k(t_1, \ldots, t_k) = \frac{b - t_k}{b-a}\, b_{i-1}^{k-1}(t_1, \ldots, t_{k-1}) + \frac{t_k - a}{b-a}\, b_i^{k-1}(t_1, \ldots, t_{k-1}),$$

We claim that $\mathcal{B}(t_1, \ldots, t_n) = b_n^n(t_1, \ldots, t_n)$.

Proof. It is evident that $b_n^n(t, \ldots, t) = f(t)$, and what must be seen is that $b_n^n(t_1, \ldots, t_n)$ is a symmetric n-ary affine form. But it is clearly affine in each variable (being a linear interpolation), hence only symmetry requires some comment. For this, notice that since permutations are generated by transpositions, it is enough to check symmetry for the latter. This means we are dealing with the two triangles

$$\begin{array}{ccc} & t_k & t_{k+1} \\ b_{i-1}^{k-1} & & \\ b_i^{k-1} & b_i^k & \\ b_{i+1}^{k-1} & b_{i+1}^k & b_{i+1}^{k+1} \end{array} \qquad\qquad \begin{array}{ccc} & t_{k+1} & t_k \\ b_{i-1}^{k-1} & & \\ b_i^{k-1} & b_i^k & \\ b_{i+1}^{k-1} & b_{i+1}^k & b_{i+1}^{k+1} \end{array}$$

Indeed, we must see that the transposition of the two variables gives the same final vertices. To ease notations, write $u = t_k$ and $v = t_{k+1}$, and let us suppose I is the unit interval $[0,1]$. The final vertex of the first triangle is

$$\begin{aligned}(1-v)\big[(1-u)b_{i-1}^{k-1} + ub_i^{k-1}\big] + v\big[(1-u)b_i^{k-1} + ub_{i+1}^{k-1}\big] \\ = (1-v)(1-u)b_{i-1}^{k-1} + (u - 2uv + v)b_i^{k-1} + uvb_{i+1}^{k-1},\end{aligned}$$

and that of the second

$$\begin{aligned}(1-u)\big[(1-v)b_i^{k-1} + vb_i^{k-1}\big] + u\big[(1-v)b_i^{k-1} + vb_{i+1}^{k-1}\big] \\ = (1-u)(1-v)b_{i-1}^{k-1} + (v - 2vu + u)b_i^{k-1} + vub_{i+1}^{k-1},\end{aligned}$$

clearly the same thing! ∎

(3.5) Blossom and control ordinates.N The control ordinates c_k are codified by the blossom:

$$c_k = \mathcal{B}(a, \overset{n-k}{\ldots}, a, b, \overset{k}{\ldots}, b)$$

Indeed, interpolation at $t_{k+1} = a$ or b is very easy, as it gives respectively the initial or the final point:

$$\begin{array}{ll} & t_k = a \\ b_{i-1}^{k-1} & \\ b_i^{k-1} & b_i^k = b_i^{k-1} \end{array} \qquad\qquad \begin{array}{ll} & t_k = b \\ b_{i-1}^{k-1} & \\ b_i^{k-1} & b_i^k = b_i^{k-1} \end{array}$$

Thus, each interpolation at a simply drops the *last* entry of the column, hence after $n-k$ such interpolations, the column we obtain consists of $b_0^0, \ldots, b_k^0$. Then each interpolation at b simply drops the *first* entry of the column, hence after k of them we end up with $b_k^0 = c_k$. We are done. ∎

(3.6) Blossom and sides.N It is clear from the definition of the multivariate de Casteljeau algorithm and what was remarked in 3.5, on interpolation at a, that each entry in the hypothenuse is

$$b_k^k(t_1, \ldots, t_k) = b_k^n(t_1, \overset{k}{\ldots}, t_k, a, \overset{n-k}{\ldots}, a).$$

But this refers the computation to the blossom $\mathcal{B}$ of $\gamma(t)$, which is something intrinsic and does not depend on the interval. Hence we can write for the univariate de Casteljeau algorithm

$$b_k^k(t) = b_k^n(t, \overset{k}{\ldots}, t, a, \overset{n-k}{\ldots}, a) = \mathcal{B}(t, \overset{k}{\ldots}, t, a, \overset{n-k}{\ldots}, a) \quad \text{for } k = 0, \ldots n.$$

Similarly, for the basis of the triangle we have

$$b_n^k(t_1, \ldots, t_k) = b_n^n(t_1, \overset{k}{\ldots}, t_k, b, \overset{n-k}{\ldots}, b),$$

and

$$b_n^k(t) = b_n^n(t, \overset{k}{\ldots}, t, b, \overset{n-k}{\ldots}, b) = \mathcal{B}(t, \overset{k}{\ldots}, t, b, \overset{n-k}{\ldots}, b) \quad \text{for } k = 0, \ldots, n.$$

(3.7) Change of interval.N Given a polynomial function $f(t)$ we know well how to compute through its blossom $\mathcal{B}$ the control ordinates for any given interval $I = [a, b]$. These control ordinates are the initial data to built the de Casteljeau triangles on that interval. Thus, if we try to built up the triangles on a new interval, we need first to find the new control ordinates. However, in some important cases, de Casteljeau triangles built on different intervals relate each other well.

(1) *Same origin: the new interval is* $[a, s]$ *for some* $s > a$.

(i) The new control points are the hypothenuse $c'_k = b^k_k(s)$ of the triangle built on $[a, b]$ with target $f(s) = b^n_n(s)$.

By 3.6,

$$b^k_k(s) = b^n_n(s, \overset{k}{\ldots}, s, a, \overset{n-k}{\ldots}, a) = \mathcal{B}(a, \overset{n-k}{\ldots}, a, s, \overset{k}{\ldots}, s),$$

(here we have used the symmetry of blossoms) which are indeed the control points on $[a, s]$.

(ii) For any target $b^n_n(t)$, the hypothenuses of the triangles built on $[a, b]$ and $[a, s]$ coincide.

For, by 3.6 again, the entries on those hypothenuses depend solely on t and the common origin of the intervals.

(2) *Same end: the new interval is* $[s, b]$ *for some* $s < b$.

(i) The new control points are the base $c'_{n-k} = b^k_n(s)$ of the triangle built on $[a, b]$ with target $f(s) = b^n_n(s)$.

Again by 3.6,

$$b^k_n(s) = b^n_n(s, \overset{k}{\ldots}, s, b, \overset{n-k}{\ldots}, b) = \mathcal{B}(s, \overset{k}{\ldots}, s, b, \overset{k}{\ldots}, b),$$

which are the control points on $[s, b]$ (note: order reversed).

(ii) For any final vertex $b^n_n(t)$, the bases of the triangles built on $[a, b]$ and $[s, b]$ coincide.

For, by 3.6 again, those bases depend on t and the common end of the intervals.

Lecture 4. Bernstein polynomials

The de Casteljeau algorithms depend on the control ordenates tuple (c_k). To keep this in mind, we will sometimes write $b_n^n(t) = B^{(c_k)}(t)$. To tell the story another way, recall that the vector space $\mathcal{P}_n([a.b])$ has dimension $N = n+1$. What the de Casteljeau algorithm really gives is a mapping

$$\mathsf{deC} : \mathbb{R}^N \to \mathcal{P}_n([a.b]) : \, (c_k)_k \mapsto f(t) = B^{(c_k)}(t),$$

which is a *linear isomorphism.*

Proof. That the mapping is linear is evident from the definition of the algorithm. Then, since both spaces have the same dimension, it is enough to see that the mapping is surjective. To do that, we make a few computations to obtain an explicit description of any symmetric n-ary affine form θ (for instance the blossom of $f(t)$). First we write every variable as an interpolation

$$t_i = \alpha_i a + \beta_i b, \text{ where } \alpha_i = \frac{b - t_i}{b - a}, \quad \beta_i = \frac{t_i - a}{b - a},$$

and using that θ respects interpolation in each variable we get

$$\theta(\,\dots, t_i, \dots\,) = \theta(\,\dots, \alpha_i a + \beta_i b, \dots\,) = \sum_{(c_i,\rho_i)=(a,\alpha_i) \text{ or } (b,\beta_i)} \rho_1 \cdots \rho_n \, \theta(c_1, \dots, c_n).$$

But θ is symmetric, hence we can gather all a's together, and rewrite the sum as

$$= \sum_{k=0}^{n} \Big(\sum_{\nu} \alpha_{\nu_{k+1}} \cdots \alpha_{\nu_n} \beta_{\nu_1} \cdots \beta_{\nu_k} \Big) \, \theta(a, \overset{n-k}{\dots}, a, b, \overset{k}{\dots}, b),$$

where ν is simply a choice of k entries $1 \le \nu_1 < \cdots < \nu_k \le n$ (places for b if any), completed with the remaining entries $1 \le \nu_{k+1} < \cdots < \nu_n \le n$ (places for a). We get

$$= \sum_{k=0}^{n} \Big(\sum_{\nu} \frac{b - t_{\nu_{k+1}}}{b-a} \cdots \frac{b - t_{\nu_n}}{b-a} \cdot \frac{t_{\nu_1} - a}{b-a} \cdots \frac{t_{\nu_k} - a}{b-a} \Big) \, \theta(a, \overset{n-k}{\dots}, a, b, \overset{k}{\dots}, b).$$

Hence the $n+1$ functions $B_k^n(t_1, \dots, t_n)$ in the big parentheses, which are clearly symmetric n-ary affine forms, generate the vector space of all of them. Now substituting $\mathcal{B}$ for θ we get

$$\mathcal{B}(t_1, \dots, t_n) = \sum_{k=0}^{n} B_k^n(t_1, \dots, t_n) \mathcal{B}(a, \overset{n-k}{\dots}, a, b, \overset{k}{\dots}, b) = \sum_{k=0}^{n} B_k^n(t_1, \dots, t_n) c_k.$$

But the control ordinates c_k are arbitrary, hence the blossoms we get from the de Casteljeau multivariate algorithm are all there exist. Hence making $t_1 = \cdots = t_n = t$ we obtain all polynomials.

This means that the mapping deC is surjective as wanted. ∎

(4.1) Bernstein polynomials. Since deC is an isomorphism, it maps teh standard basis $e_k = (0, \dots, 1, \dots, 0)$ of $\mathbb{R}^N$ onto a basis of $\mathcal{P}_n([a.b])$. Looking at the last proof we find:

$$B_k^n = \mathsf{deC}(e_k) = D_k^n(t, \dots, t) = \binom{n}{k}\Big(\frac{t-a}{b-a}\Big)^k\Big(\frac{b-t}{b-a}\Big)^{n-k}, \quad k = 0, \dots, n,$$

which are the *Berstein polynomials of degree n for* $[a, b]$. Some properties of these polynomials are now immediate:

(1) The *reunion formula*

$$B_k^n(t) = \frac{b-t}{b-a}B_k^{n-1}(t) + \frac{t-a}{b-a}B_{k-1}^{n-1}(t).$$

This is just the last step in the de Casteljeau interpolation for control ordinates $(0, \dots, 1^{(k}, \dots, 0)$.

(2) The Bernstein polynomials are a *partition of unity:*

$$\sum_{k=0}^n B_k^n(t) = \sum_{k=0}^n \mathsf{deC}(e_k) = \mathsf{deC}(\sum_{k=0}^n e_k = \mathsf{deC}(1, \dots, c_1) \equiv 1.$$

And since clearly $B_k^n(t) \geq 0$ on $[a, b]$, we deduce $B_k^n(t) \leq 1$.

(3) And linear precision, as we already know what we get starting with equally spaced initial data:

$$\sum_{k=0}^n \Big(\frac{n-k}{n}a + \frac{k}{n}b\Big)B_k^n(t) = t.$$

(4.2) Bézier parametrization. Finally, the control ordinates we started with are the coordinates of $f(t)$ with respect to the basis of bernstein polynomials, and we obtain the so-called *Bézier representation*

$$f(t) = \sum_k B_k^n(t)c_k.$$

This can be seen as a direct formula for the de Casteljeau algorithm, and can be used for the whole triangle, *or for any subtriangle.* ∎

Next we turn to the derivatives. These are necessary to understand differentiability of splines at joints.

(4.3) Derivatives. We first compute the derivatives of the Bézier polynomials:

$$\begin{aligned}\frac{d}{dt}B_k^n(t) &= \frac{d}{dt}\left[\binom{n}{k}\left(\frac{t-a}{b-a}\right)^k\left(\frac{b-t}{b-a}\right)^{n-k}\right] \\ &= \frac{k}{b-a}\binom{n}{k}\left(\frac{t-a}{b-a}\right)^{k-1}\left(\frac{b-t}{b-a}\right)^{n-k} - \frac{n-k}{b-a}\binom{n}{k}\left(\frac{t-a}{b-a}\right)^{k-1}\left(\frac{b-t}{b-a}\right)^{n-k-1} \\ &= \frac{n}{b-a}\left[\binom{n-1}{k-1}\left(\frac{t-a}{b-a}\right)^{k-1}\left(\frac{b-t}{b-a}\right)^{n-k} - \binom{n-1}{k}\left(\frac{t-a}{b-a}\right)^{k}\left(\frac{b-t}{b-a}\right)^{n-k-1}\right] \\ &= \frac{n}{b-a}B_{k-1}^{n-1}(t) - \frac{n}{b-a}B_k^{n-1}(t).\end{aligned}$$

Then, for a Bézier representation $f(t) = \sum_k B_k^n(t)c_k$, we obtain

$$\begin{aligned}\frac{df}{dt}(t) &= \sum_k\left(\frac{d}{dt}B_k^n(t)\right)c_k = \frac{n}{b-a}\sum_k\left[B_{k-1}^{n-1}(t) - B_k^{n-1}(t)\right]c_k \\ &= \frac{n}{b-a}\left[\sum_{k=1}^{n}B_{k-1}^{n-1}(t)c_k - \sum_{k=0}^{n-1}B_k^{n-1}(t)c_k\right] \\ &= \frac{n}{b-a}\left[\sum_{k=0}^{n-1}B_k^{n-1}(t)c_{k+1} - \sum_{k=0}^{n-1}B_k^{n-1}(t)c_k\right] \\ &= \frac{n}{b-a}\sum_{k=0}^{n-1}B_k^{n-1}(t)(c_{k+1}-c_k) = \frac{n}{b-a}\sum_{k=0}^{n-1}B_k^{n-1}(t)\Delta c_k\,,\end{aligned}$$

where $\Delta c_k = c_{k+1} - c_k$. ∎

(4.4) Local control. It is clearly important to guess how perturbations in the control points affect to the resulting polynomial functions. It is clear from the definition that replacing c_k by c_k' will be of bigger consequence there where $B_k^n(t)$ is bigger. To understand better where, we look for the maxima of $B_k^n(t)$ in the open interval (a,b), that is, for zeros of $\frac{d}{dt}B_k^d(t)$. Pursuing the computations in the preceding paragraph one deduces

$$\frac{d}{dt}B_k^d(t) = \alpha(t)\left(\frac{n-k}{n}a + \frac{k}{n}b - t\right), \quad \text{with } \alpha(t)\neq 0 \text{ in the open interval.}$$

Hence the unique inner zero is $t = \frac{n-k}{n}a + \frac{k}{n}b$. This unique inner extreme is of course a maximum, which moves from a to b as k varies from 0 to d. ∎

Now, the computations for the first derivative can be repeated to come without surprise to similar formulas for higher derivatives.

(4.5) Higher order derivatives. Given control ordinates $c_0,\dots,c_n$ define:

$$\Delta^r c_k = \sum_{i=0}^{r}(-1)^i\binom{r}{i}c_{k+r-i} \qquad \text{for } r = 0,\dots n-k.$$

We see Δ^0 is the identity and Δ^1 is the operator Δ used above to express first derivatives. It is easy to check also that this is an inductive definition:

$$\Delta^r c_k = \Delta^{r-1} c_{k+1} - \Delta^{r-1} c_k.$$

With these notations, the following formula holds true for $r = 0, \dots, n$:

$$\frac{d^r f}{dt^r}(t) = \frac{n(n-1)\cdots(n-r+1)}{(b-a)^r} \sum_{k=0}^{n-r} B_k^{n-r}(t) \Delta^r c_k.$$

(4.6) Particular cases.N Let us compute the derivatives at the two extremes of our interval $[a, b]$. We find out:

At the origin $t = a$: The only Bernstein function that does not vanish at a is the first one, and in fact $B_0^{n-1}(a) = 1$, hence we get

$$\frac{d^r f}{dt^r}(a) = \frac{n(n-1)\cdots(n-r+1)}{(b-a)^r} \Delta^r c_0 = \frac{n!}{(n-r)!} \frac{1}{(b-a)^r} \sum_{i=0}^{r} (-1)^i \binom{r}{i} c_{r-i}$$

In particular, for $r = 1$, we find that the tangent to the graph of $f(t)$ at its origin (a, c_0) is (proportional to) the first edge of the control polygon:

$$\frac{d}{dt}(t, f(t))\Big|_{t=a} = (1, ; \frac{df}{dt}(a)) = (1, n(c_1 - c_0)).$$

At the end $t = b$: The only Bernstein function that does not vanish at b is the last one, and in fact $B_{n-1}^{n-1}(b) = 1$, hence we get

$$\frac{d^r f}{dt^r}(b) = \frac{n(n-1)\cdots(n-r+1)}{(b-a)^r} \Delta^r c_{n-r} = \frac{n!}{(n-r)!} \frac{1}{(b-a)^r} \sum_{i=0}^{r} (-1)^i \binom{r}{i} c_{n-i}$$

Again for $r = 1$, we find that the tangent to the graph of $f(t)$ at its end (b, c_n) is (proportional to) the last edge of the control polygon:

$$\frac{d}{dt}(t, f(t))\Big|_{t=b} = (1, \frac{df}{dt}(b)) = (1, n(c_n - c_{n-1})).$$

Back to arbitrary r, the formulas show that the first r derivatives at a (resp b) depend solely on the first (resp. last) $r + 1$ control ordinates. ∎

Lecture 5. Differentiability

Since a spline is built up by patching two polynomials at each joint, the class at such a joint can be determined by comparing left and right derivatives at them. But those left and right derivatives is what we have described in the preceding lecture.

(5.1) Checking differentiability at a joint. N Suppose we look at a joint s of a spline $f(t)$ of degree n, and the branches at c are: (i) $g(t)$, defined on $[a, s]$ by $n + 1$ control ordinates c_k, and (ii) $h(t)$, defined on $[s, b]$ by other $n + 1$ control ordinates $c_{n+\ell}$. Notice that by continuity the last control ordinate c_n of $g(t)$ must be the first one of $h(t)$, hence we enumerate consecutively all control ordinates.

Then the spline is of class r at s if and only if $\frac{d^i g}{dt^i}(s) = \frac{d^i h}{dt^i}(s)$, which by the formulas 4.6, p.15 is equivalent to

$$\frac{1}{(s-a)^i}\Delta^i c_{n-i} = N\frac{1}{(b-s)^i}\Delta^i c_n\,.$$

for $i = 0, \dots, r$. We see that the $2r+1$ control ordinates involved here are $c_{n-r}, \dots, c_{2n-r}$. Consider now the two polynomial functions of degree r: (i) $g^*(t)$, defined on $[a.s]$ by the control ordinates $c_{n-r}, \dots, c_n$, and (ii) $h^*(t)$, defined on $[s, b]$ by the control ordinates $c_n, \dots, c_{2n-r}$. The formulas above mean that $g^*(t)$ and $h^*(t)$ match of class r at c. Since class=degree, we conclude that $g^*(t) = h^*(t)$. In other words, $f(t)$ has class r at s if and only if the last $r+1$ control ordinates of $g(t)$ and the first $r+1$ control ordinates of $h(t)$ are control ordinates of one and the same polynomial function of degree r.

This can be checked as follows. Use the multivariate de Casteljeau algorithm on $[a, s]$ with control ordinates $c_{n-r}, \dots, c_n$ to compute the blossom $b_r^r(t_1, \dots, t_r)$ of such a polynomial function of degree r. Then use that blossom to obtain the control ordinates on $[s, b]$. The conclusion is that the spline $f(t)$ is of class r at s if and only if

$$c_{n+k} = b_r^r(\overset{r-k}{s, \dots, s}, \overset{k}{b, \dots, b}),$$

for $k = 0, \dots, r$. ∎

Next we take a closer look at the class 1 and 2 cases.

(5.2) Class 1 differentiabilty.N Let $g(t)$ be defined on $[a, s]$ by $n+1$ control ordinates $\dots, c_{n-1}, c_n$, and let $h(t)$ be defined on $[s, b]$ by $n+1$ control ordinates $c_n, c_{n+1}, \dots$. The formula in 5.1 for $r = 1$ gives

$$\frac{c_n - c_{n-1}}{s-a} = \frac{c_{n+1} - c_n}{b-s},$$

which after a small manipulation we write as

$$c_n = \frac{b-s}{b-a}c_{n-1} + \frac{a-s}{b-a}c_{n+1},$$

again linear interpolation. To tell it: *the three ordinates* c_{n-1}, c_n, c_{n+1} *are in the same ratio that* a, s, b. This is consequently the condition for class 1 differentiability.

(5.3) Class 2 differentiability.N Let $g(t)$ be defined on $[a, s]$ by $n+1$ control ordinates $\dots, c_{n-2}, c_{n-1}, c_n$, and let $h(t)$ be defined on $[s, b]$ by $n+1$ control ordinates $c_n, c_{n+1}, c_{n+2} \dots$. We will proceed as described at the end of 5.1. First we look at the blossom $\mathcal{B}$ obtained by the de Casteljeau algorithm

$$\begin{array}{lll} & t_{k+1} & t_{k+2} \\ c_{n-2} & & \\ c_{n-1} & b_0^1 & \\ c_n & b_1^1 & b_0^2 \end{array}$$

in the interval $[a, s]$, which means that we built the above triangle by interpolation using the coefficients $\frac{s-t}{s-a}, \frac{t-a}{s-a}$ $(t \in \mathbb{R})$. Class 2 differentiability is equivalent to the conditions:

$$c_n = \mathcal{B}(s, s), \quad c_{n+1} = \mathcal{B}(s, b), \quad c_{n+2} = \mathcal{B}(b, b).$$

Now, the first equality is trivial. The second reads:

$$c_{n+1} = \frac{s-b}{s-a} c_{n-1} + \frac{b-a}{s-a} c_n,$$

which is nothing but the class 1 differentiability condition in 5.2. The last condition is something new:

$$\begin{aligned} c_{n+2} &= \frac{s-b}{s-a} \left[\frac{s-b}{s-a} c_{n-2} + \frac{b-a}{s-a} c_{n-1} \right] + \frac{b-a}{s-a} \left[\frac{s-b}{s-a} c_{n-1} + \frac{b-a}{s-a} c_n \right] \\ &= \frac{s-b}{s-a} \left[\frac{s-b}{s-a} c_{n-2} + \frac{b-a}{s-a} c_{n-1} \right] + c_{n+1} \end{aligned}$$

(by the previous equality), and after a small manipulation we come to

$$\frac{s-b}{s-a} c_{n-2} + \frac{b-a}{s-a} c_{n-1} = \frac{b-a}{b-s} c_{n+1} + \frac{a-s}{b-s} c_{n+2}.$$

This is linear interpolation anew! To understand it, call q that common value. Then, left hand and right hand side tells us respectively that c_{n-2}, c_{n-1}, q *are in the same ratio that* a, s, b, *and* q, c_{n+1}, c_{n+2} *in the same that* a, s, b.

Lecture 6. Quadratic and cubic splines

Here we give direct descriptions of quadratic and cubic splines of different classes. From the discussion on differentiability and the de Casteljau algorithm all these can be described quite well.

For the balance of the section, we fix a compact interval $I = [a, b]$ and a partition $a = t_0 < t_1 < \cdots < t_{m+1} = b$. There we will have to define splines $f(t)$ which are polynomial on each subinterval $I_i = [t_i, t_{i+1}]$, $i = 0, \ldots, m$. We consider the graph $\gamma(t) = (t, f(t))$, so that

(6.1) Quadratic splines. N In this case, $\gamma(t)$ has degree 2 on each I_i, and consequently, three control points. We enumerate those control points consecutively, say

$$c_0, c_1, c_2, \ldots, c_{2i}, c_{2i+1}, c_{2i+2}, \ldots, c_{2m}, c_{2m+1}, c_{2m+2},$$

(we have $\gamma(t_i) = c_{2i}$). The fact that the third control point on I_i is the first on I_{i+1} says that the spline is continuous. Concerning differentiability, in this case we are interested in class 1. But this was completely settled in 5.2, p.16, and the condition to impose at each joint t_i $i = 1, \ldots, m$, is that $c_{2i-1}, c_{2i}, c_{2i+1}$ are collinear, in the same ratio that t_{i-1}, t_i, t_{i+1}. Now, since the partition is fixed, those ratios are data, and we can drop one of those three control points. We choose to drop c_{2i}, internal to the segment $[c_{2i-1}, c_{2i+1}]$, so that the control polygon is also described by the remaining points, and in case we need, we know that

$$c_{2i} = \frac{t_{i+1} - t_i}{t_{i+1} - t_{i-1}} c_{2i-1} + \frac{t_i - t_{i-1}}{t_{i+1} - t_{i-1}} c_{2i+1}.$$

Consequently, $\gamma(t)$ is completely determined by the control polygon, but now seen as generated by

$$c_0, c_1, c_3, \ldots, c_{2i+1}, \ldots, c_{2m+1}, c_{2m+2}.$$

These are renamed as

$$p_0, p_1, \ldots, p_M,$$

and called *de Boor control points.*

In addition we confirm that the vector space of all quadratic class 1 splines (for the given partition) has dimension

$$\#\{\text{de Boor control points}\} = \#\{\text{control points}\} - \#\{\text{joints}\} = m + 3,$$

which coincides with 2.4, p.7:

$$n + 1 + r_1 + \cdots + r_m = 2 + 1 + (1 + \overset{(m}{\cdots} + 1) = m + 3.$$

Let us explore a little further how the abscissas of the de Boor control points are determined. We write $p_i = (\xi_i, d_i)$, and call ξ_i the *Greville abscissa* and d_i the *de Boor* *ordinate* of the control point. Then, linear precision (3.3, p.9)says that:

$$\begin{cases} \xi_0 = t_0 = \frac{1}{2}(t_0 + t_0) \\ \xi_1 = \frac{1}{2}(t_0 + t_1) \\ \xi_2 = \frac{1}{2}(t_1 + t_2) \\ \vdots \\ \xi_i = \frac{1}{2}(t_{i-1} + t_i) \\ \vdots \\ \xi_{M-1} = \frac{1}{2}(t_m + t_{m+1}) \\ \xi_M = t_{m+1} = \frac{1}{2}(t_{m+1} + t_{m+1}) \end{cases}$$

To tiny this up, we insert some repetitions in the partition $a = t_0 < t_1 < \cdots < t_m < t_{m+1}$ and obtain the so-called *knot sequence* $u_0 \le u_1 \le \cdots \le u_{M+1}$, as follows:

$$\begin{cases} u_0 = u_1 = t_0 \\ u_{i+1} = t_i & \text{for } i = 1, \ldots, m \\ u_M = u_{M+1} = t_{m+1} \end{cases}$$

This has the advantage that now we have a uniform formula for all Greville abscissas, namely

$$\xi_j = \tfrac{1}{2}(u_j + u_{j+1}) \quad \text{for all } j = 0, \ldots, M.$$

∎

Next, we turn to cubic splines. In this case we can look at class 1 and class 2 splines.

(6.2) Cubic class 1 splines.N These can be described in the vein of quadratic class 1 splines. Briefly, we enumerate the control points as

$$c_0, c_1, c_2, c_3 \ldots, c_{3i}, c_{3i+1}, c_{3i+2}, c_{3i+3}, \ldots, c_{3m}, c_{3m+1}, c_{3m+2}, c_{3m+3}$$

and again class 1 says the control points $\gamma(t_i) = c_{2i}$ at joints can be recovered by linear interpolation, hence we drop them to get

$$c_0, c_1, c_2, c_4 \ldots, c_{2i-1}, c_{2i+1}, c_{2i+2}, c_{2i+4}, \ldots, c_{2m-1}, c_{2m+1}, c_{2m+2}, c_{2m+3},$$

which are again renamed $p_0, p_1, \ldots, p_M$ and *de Boor control points.* Note that again these fewer points still generate the same control polygon.

In this case the dimension is

$$\#\{\text{de Boor control points}\} = \#\{\text{control points}\} - \#\{\text{joints}\} = 2m + 4,$$

of course as predicted by 2.4, p.7:

$$n + 1 + r_1 + \cdots + r_m = 3 + 1 + (2 + \overset{(m}{\cdots} + 2) = 2m + 4.$$

Finally, let us look at the Greville abscissas. We already know that the abscissas of $c_{3i}, c_{3i+1}, c_{3i+2}, c_{3i+3}$ are distributed in thirds on the interval $I_i = [t_i.t_{i+1}]$ for each $i = 0, \dots, m$. Thus a small computation gives

$$\begin{cases} \xi_0 = t_0 = \frac{1}{3}(t_0 + t_0 + t_0) \\ \xi_1 = \frac{2}{3}t_0 + \frac{1}{3}t_1 = \frac{1}{3}(t_0 + t_0 + t_1) \\ \xi_2 = \frac{1}{3}t_0 + \frac{2}{3}t_1 = \frac{1}{3}(t_0 + t_1 + t_1) \\ \vdots \\ \xi_{2i+1} = \frac{2}{3}t_i + \frac{1}{3}t_{i+1} = \frac{1}{3}(t_i + t_i + t_{i+1}) \\ \xi_{2i+2} = \frac{1}{3}t_i + \frac{2}{3}t_{i+1} = \frac{1}{3}(t_i + t_{i+1} + t_{i+1}) \\ \vdots \\ \xi_{M-2} = \frac{2}{3}t_m + \frac{1}{3}t_{m+1} = \frac{1}{3}(t_m + t_m + t_{m+1}) \\ \xi_{M-1} = \frac{1}{3}t_m + \frac{2}{3}t_{m+1} = \frac{1}{3}(t_m + t_{m+1} + t_{m+1}) \\ \xi_M = t_{m+1} = \frac{1}{3}(t_{m+1} + t_{m+1} + t_{m+1}) \end{cases}$$

Again, we insert suitable repetitions into the initial partition to obtain the knot sequence $u_0 \leq u_1 \leq \cdots \leq u_{M+1}$, as follows:

$$\begin{cases} u_0 = u_1 = u_2 = t_0 \\ u_{2i+1} = t_{2i+2} = t_i & \text{for } i = 1, \dots, m \\ u_M = u_{M+1} = u_{M+2} = t_{m+1} \end{cases}$$

This gives

$$\xi_j = \tfrac{1}{3}(u_j + u_{j+1} + u_{j+2}) \quad \text{for all } j = 0, \dots, M.$$

(6.3) Cubic class 2 splines.N Here we have something else to do. We start the same with our control points

$$c_0, c_1, c_2, c_3 \dots, c_{3i}, c_{3i+1}, c_{3i+2}, c_{3i+3}, \dots, c_{3m}, c_{3m+1}, c_{3m+2}, c_{3m+3}$$

and the class 1 differentiability condition says we can drop c_{2i} for $i = 1, \dots, m$, as they can be defined through the two nearby control points by interpolation. But then the spline is class 2, and differentiability tells us more. By 5.3, p.17, for every joint t_i, the two lines through c_{3i-2}, c_{3i-1} and through c_{3i+1}, c_{3i+2} meet at a point q_i such that c_{3i-2}, c_{3i-1}, q_i *and* q_i, c_{3i+1}, c_{3i+2} *are in the same ratio that* t_{i-1}, t_i, t_{i+1}. Reordering these data a bit, we can interpolate q_i and q_{i+1} to get the two points c_{3i+1}, c_{3i+2}, for $i = 2, \dots, m-1$. On the other hand, at the first joint t_1 we can recover c_2 interpolating c_1 and d_1, and at t_m, we recover c_{3m+1} interpolating d_m and c_{3m+2}.

Summing up, we replace the control points c_{3i} correponding to joints by the q_i's, and drop all the other control points that can be recovered by interpolation: thus we still get the same control polygon. As before, the new control points are named $p_0, p_1, \dots, p_M$ and called *de Boor control points*.

In this case the number of de Boor control points is

$$\begin{aligned}\#\{\text{de Boor control points}\} &= \#\{\text{control points}\} - \#\{\text{joints}\}\\ &\quad + \#\{\text{meeting points } d_i\} - \#\{\text{points in pairs } c_{3i+1}, c_{3i+2}\}\\ &\quad - \#\{c_2, c_{3m+1}\}\\ &= \big(3(m+1)+1\big) - m + m - 2(m-1) - 2 = m+4,\end{aligned}$$

matching 2.4, p.7:

$$n + 1 + r_1 + \cdots + r_m = 3 + 1 + (1 + \overset{(m}{\cdots} + 1) = m + 4.$$

We next look at the Greville abscissas ξ_j. The two first and the two last are easy:

$$\begin{cases} \xi_0 = t_0 = \frac{1}{3}(t_0 + t_0 + t_0) \\ \xi_1 = \frac{2}{3}t_0 + \frac{1}{3}t_1 = \frac{1}{3}(t_0 + t_0 + t_1) \end{cases} \qquad \begin{cases} \xi_{M-1} = \frac{1}{3}t_m + \frac{2}{3}t_{m+1} = \frac{1}{3}(t_m + t_{m+1} + t_{m+1}) \\ \xi_M = t_M = \frac{1}{3}(t_M + t_M + t_M) \end{cases}$$

Now for $i = 1, \ldots, m$, we have

$$p_{i+1} = q_i = \frac{t_i - t_{i+1}}{t_i - t_{i-1}} c_{3i-2} + \frac{t_{i+1} - t_{i-1}}{t_i - t_{i-1}} c_{3i-1},$$

hence

$$\begin{aligned}\xi_{i+1} &= \frac{t_i - t_{i+1}}{t_i - t_{i-1}} \left[\tfrac{2}{3}t_{i-1} + \tfrac{1}{3}t_i\right] + \frac{t_{i+1} - t_{i-1}}{t_i - t_{i-1}} \left[\tfrac{1}{3}t_{i-1} + \tfrac{2}{3}t_i\right]\\ &= \frac{(t_i - t_{i+1})(2t_{i-1} + t_i) + (t_{i+1} - t_{i-1})(t_{i+1} + 2t_i)}{3(t_i - t_{i-1})}\\ &= \frac{(t_i - t_{i+1})t_{i-1} + (t_i - t_{i+1})(t_{i-1} + t_i) + (t_{i+1} - t_{i-1})(t_{i+1} + t_i) + (t_{i+1} - t_{i-1})t_i}{3(t_i - t_{i-1})}\\ &= \frac{(t_i - t_{i-1})(t_i + t_{i+1}) + t_{i+1}(t_i - t_{i-1})}{3(t_i - t_{i-1})}\\ &= \tfrac{1}{3}(t_{i-1} + t_i + t_{i+1}),\end{aligned}$$

and we come to the same magic relation! Thus we define the knot sequence $u_0 \le u_1 \le \cdots \le u_{M+2}$ by:

$$\begin{cases} u_0 = u_1 = u_2 = t_0 & \\ u_{i+2} = t_i & \text{for } i = 1, \ldots, m \\ u_M = u_{M+1} = u_{M+2} = t_{m+1} & \end{cases}$$

This gives

$$\xi_j = \tfrac{1}{3}(u_j + u_{j+1} + u_{j+2}) \quad \text{for all } j = 0, \ldots, M.$$

∎

The discussion so far shows how we must turn from a simple partition $t_0 < t_1 < \cdots$ of the interval to a modified sequence with repetitions $u_0 \le u_1 \le \cdots$. We also see how those repetitions depend on the degree and the class of the spline. Note also that interpolation produces all control points from the de Boor control points. Now we are ready to describe the algorithm we are after.

Lecture 7. The de Boor knot insertion algorithm

We start with the basic procedure on which the de Boor algorithm is based.

(7.1) Knot insertion procedure.N Let two positive integers n and N be fixed. We consider the following data:

1. a knots sequence $u_0 \leq u_1 \leq \cdots \leq u_{N+2n-2}$, and
2. some de Boor ordinates $d_0, d_1, \ldots, d_{N+n-1}$,

to which we associate:

3. Greville abscissas $\xi_j = \frac{1}{n}(u_j + \cdots + u_{j+n-1})$ for $j = 0, \ldots, N+n-1$, and
4. a de Boor polygon P with consecutive vertices

$$(\xi_0, d_0), (\xi_1, d_1), \ldots, (\xi_{N+n-1}, d_{N+n-1}).$$

Obviously, we call d_j the de Boor ordinate at ξ_j.

We will produce new data for any new knot v we choose in the domain $[u_0, u_{N+n-1}]$. First, to simplify formulas, we repeat on demand the last Greville abscissa and de Boor ordinate: $\xi_j = \xi_{N+n-1}$, $d_j = d_{N+n-1}$ for $j > N+n-1$. Then $u_J \leq v$ for a unique largest index J, and we define:

(i) a new knot sequence $u_0^v \leq \cdots \leq u_{N^v+2n-2}^v$, with $N^v = N+1$, by *insertion of* v*:*

$$u_j^v = \begin{cases} u_j & \text{for } j = 0, \ldots, J \\ v & \text{for } j = J+1 \\ u_{j-1} & \text{for } j > J+1 \end{cases}$$

(ii) new Greville abscissas for $j = 0, \ldots, N^v + n - 1$:

$$\xi_j^v = \frac{1}{n}(u_j^v + \cdots + u_{j+n-1}^v) = \begin{cases} \xi_j & \text{for } j \leq J-n+1 \\ \xi_j - \frac{1}{n}(u_{j+n-1} - v) & \text{for } J-n+1 < j \leq J+1 \\ \xi_{j-1} & \text{for } j > J+1 \end{cases}$$

(iii) new de Boor ordinates: since in any case $\xi_{j-1} \leq \xi_j^v \leq \xi_j$, we interpolate

$$d_j^v = \frac{u_{j+n-1} - v}{u_{j+n-1} - u_{j-1}} d_{j-1} + \frac{v - u_{j-1}}{u_{j+n-1} - u_{j-1}} d_j \,.$$

(iv) a new de Boor polygon, which we denote of course P^v, with vertices

$$(\xi_0^v, d_0^v), \ldots, (\xi_{N^v+n-1}^v, d_{N^v+n-1}^v).$$

This procedure can be iterated, and we will use for iterations the self-explanatory notations

$$u_j^{v_1 \ldots v_k}, \quad \xi_j^{v_1 \ldots v_k}, \quad d_j^{v_1 \ldots v_k}, \quad \text{and} \quad P^{v_1 \ldots v_k},$$

where $v_1, \ldots, v_k$ are the k knots consecutively inserted in that order. ∎

Remarks 7.2 The bare definitions show that:

(1) If the new Greville abscissa is one that already existed, the new de Boor ordinate is the same we already had. This applies in particular to $v = u_{N+n-1}$.

(2) ln case:

$$u_{J-n+1} = \cdots = u_J = v$$

(that is, when v occurs already n times in the knot sequence), nothing really happens, as the procedure only repeats already existing Greville abscissas with their de Boor ordinates.

(3) The procedure introduces exactly n new Greville abscissas, for $J-n+1 < j \leq J+1$, in which case $v < u_{J-n+1}$. Thus, we have n new de Boor ordinates and n new vertices in the de Boor polygon. In case v is not a knot already, these are all truly new data. Otherwise there are some repetitions of vertices in the new de Boor polygon.

(4) The domain to choose a second knot remains the same, because $u_0 == u_0^v$ (evident) and $u_{N^v+n-1}^v = u_{N+n-1}$ (check in the case $v = u_{N+n-1}$).

∎

The property we now prove is that our procedure is symmetric:

Proposition 7.3 *N Let* $v_1, \ldots, v_k \in [u_0, u_{N+n-1}]$. *Then, for any permutation* σ *of* $1, \ldots, k$, *it holds*

$$P^{v_1 \ldots v_k} = P^{v_{\sigma_1} \ldots v_{\sigma_k}}$$

Proof. It is enough to prove this for the insertion of two knots $v \leq w$. But it is clear that the two polygons P^{vw} and P^{wv} have the same knot sequence, hence the same Greville abscissas, so we only bother for the de Boor ordinates. Thus let us fix an index j. We must distinguish several cases, concerning the way v, w are placed among the knots $u_0 \leq \cdots$, and make the corresponding computations. Some are very simple, some are not, but mistery nowhere. We will include here the details in case the knots are placed as follows

$$u_{j-1}, u_j, \ldots, v, \ldots, w, \ldots, u_{j+n-2}.$$

In this situation

$$u^v_{j-1} = u_{j-1}, \quad u^v_{j+n-1} = u_{i+n-2},$$

hence:

$$\begin{aligned} d^{vw}_j &= \tfrac{u^v_{j+n-1}-w}{u^v_{j+n-1}-u^v_{j-1}} \left[\tfrac{u_{j+n-2}-v}{u_{j+n-2}-u_{j-2}} d_{j-2} + \tfrac{v-u_{j-2}}{u_{j+n-2}-u_{j-2}} d_{j-1}\right] \\ &\qquad + \tfrac{w-u^v_{j-1}}{u^v_{j+n-1}-u^v_{j-1}} \left[\tfrac{u_{j+n-1}-v}{u_{j+n-1}-u_{j-1}} d_{j-1} + \tfrac{v-u_{j-1}}{u_{j+n-1}-u_{j-1}} d_j\right] \\ &= \tfrac{u_{j+n-2}-w}{u_{j+n-2}-u_{j-1}} \left[\tfrac{u_{j+n-2}-v}{u_{j+n-2}-u_{j-2}} d_{j-2} + \tfrac{v-u_{j-2}}{u_{j+n-2}-u_{j-2}} d_{j-1}\right] \\ &\qquad + \tfrac{w-u_{j-1}}{u_{j+n-2}-u_{j-1}} \left[\tfrac{u_{j+n-1}-v}{u_{j+n-1}-u_{j-1}} d_{j-1} + \tfrac{v-u_{j-1}}{u_{j+n-1}-u_{j-1}} d_j\right] \\ &= A(v,w) d_{j-2} + B(v,w) d_{j-1} + C(v,w) d_j. \end{aligned}$$

Clearly, after similar computations we conclude

$$d^{wv}_j = A(w,v) d_{j-2} + B(w,v) d_{j-1} + C(w,v) d_j,$$

and so all reduces to check that A, B, C are symmetric, which is evident for A and C. For $B(v,w)$, we write it down:

$$\begin{aligned} &\frac{(u_{j+n-1}-u_{j-1})(u_{j+n-2}-w)(v-u_{j-2}) + (u_{j+n-2}-u_{j-2})(w-u_{j-1})(u_{j+n-1}-v)}{(u_{j+n-2}-u_{j-1})(u_{j+n-2}-u_{j-2})(u_{j+n-1}-u_{j-1})} \\ &\qquad = \frac{\alpha_0 + \alpha_1 v + \alpha_2 w + \alpha_3 vw}{(u_{j+n-2}-u_{j-1})(u_{j+n-2}-u_{j-2})(u_{j+n-1}-u_{j-1})} \end{aligned}$$

for some coefficients $\alpha_i \in \mathbb{R}$. Consequently, this is symmetric if $\alpha_1 = \alpha_2$, but:

$$\begin{cases} \alpha_1 = (u_{j+n-1}-u_{j-1})u_{j+n-2} + (u_{j+n-2}-u_{j-2})u_{j-1} \\ \alpha_2 = (u_{j+n-1}-u_{j-1})u_{j-2} + (u_{j+n-2}-u_{j-2})u_{j+n-1} \end{cases}$$

hence

$$\alpha_1 - \alpha_2 = (u_{j+n-1}-u_{j-1})(u_{j+n-2}-u_{j-2}) + (u_{j+n-2}-u_{j-2})(u_{j-1}-u_{j+n-1}) = 0.$$

This completes the argument of the case under consideration.

As said before, there are several other cases that we urge the reader to explore, to come to terms with symmetry. ▮

Remark 7.4 It is a common place in many texts on splines to state the above symmetry result as a consequence of classical Menelao's theorem. However, despite how suggestive pictures are, the conditions of that theorem are not exactly those in a double knot insertion, as ratios vary. Thus a direct application of that classical theorem does not seem likely.

Lecture 8. The de Boor splines

Here we come to the core of the de Boor algorithm: repeated insertion.

(8.1) Repeated insertion of the same knot. Consider, as in the preceding lecture (7.1, p.22), a knot sequence $u_0 \leq \cdots \leq u_{N+2n-2}$ with de Boor polygon $P = \{(\xi_j, d_j)\}$. We insert $v \in [u_0, u_{N+n-1}]$, say $u_J \leq v$ with J largest, to get

$$d_j^v = \frac{u_{j+n-1} - v}{u_{j+n-1} - u_{j-1}} d_{j-1} + \frac{v - u_{j-1}}{u_{j+n-1} - u_{j-1}} d_j\,, \qquad \text{for } J - n + 1 < j \leq J + 1.$$

Here we change a bit notations, and write

$$d_j^1(v) = \frac{u_{j+n-1} - v}{u_{j+n-1} - u_{j-1}} d_{j-1}^0(v) + \frac{v - u_{j-1}}{u_{j+n-1} - u_{j-1}} d_j^0(v)\,, \qquad \text{for } J - n + 1 < j \leq J + 1.$$

Next, we insert v again:

$$\begin{aligned} d_j^2(v) &= \frac{u_{j+n-1}^v - v}{u_{j+n-1}^v - u_{j-1}^v} d_{j-1}^1(v) + \frac{v - u_{j-1}^v}{u_{j+n-1}^v - u_{j-1}^v} d_j^1(v)\,, \\ &= \frac{u_{j+n-2} - v}{u_{j+n-2} - u_{j-1}} d_{j-1}^1(v) + \frac{v - u_{j-1}}{u_{j+n-2} - u_{j-1}} d_j^1(v)\,, \qquad \text{for } J - n + k < j \leq J + 2. \end{aligned}$$

Clearly, after k insertions we obtain a degree k polynomial:

$$d_j^k(v) = \frac{u_{j+n-k} - v}{u_{j+n-k} - u_{j-1}} d_{j-1}^{k-1}(v) + \frac{v - u_{j-1}}{u_{j+n-k} - u_{j-1}} d_j^{k-1}(v)\,, \qquad \text{for } J - n + k < j \leq J + k.$$

Suppose now we iterate these insertions till v occurs n times in the knot sequence. Here multiplicities appear naturally. We say that v has *multiplicity* r, when

$$u_{J-r} < u_{J-r+1} = \cdots = u_J = v.$$

Now, if that is the case, and we insert it $(n-r)$ times, v becomes the $(j-r+1)$-th Greville abscissa, and

$$d_{J-r+1}^{n-r}(v),$$

is the corresponding de Boor ordinate. Of course, the case v was not a knot, corresponds to $u_J < v$ and multiplicity $r = 0$. ∎

We thus see how polynomial functions are related to repeated insertion of knots. What we really have is a kind of *piecewise* de Casteljeau algorithm. Indeed, we can represent the above computations by an interpolation triangle of the type we are familiar to:

$$
\begin{array}{rcccccccc}
 & 0 & 1 & \dots & k-1 & k & \dots & \dots & \dots & n-r \\
d_{J-n+1} = & d^0_{J-n+1} & & & & & & & & \\
d_{J-n+2} = & d^0_{J-n+2} & d^1_{J-n+2} & & & & & & & \\
 & \vdots & \vdots & \ddots & & & & & & \\
 & \vdots & \vdots & & & \ddots & & & & \\
 & \vdots & \vdots & & d^{k-1}_{j-1} & & \ddots & & & \\
 & & & & & \searrow & & & & \\
 & \vdots & \vdots & & d^{k-1}_{j} & \to d^k_j & & \ddots & & \\
 & \vdots & \vdots & & & & & & \ddots & \\
d_{J-r+1} = & d^0_{J-r+1} & d^1_{J-r+1} & \cdots & & \cdots & \cdots & \cdots & \cdots & d^{n-r}_{J-r+1}
\end{array}
$$

We will call the elements in the first column the *insertion control points*. But this, although close, *is not* a de Casteljeau interpolation: *ratios depend on indices.* Namely, the ratio of the subtriangle $d^{k-1}_{j-1}, d^k_j, d^{k-1}_j$ spotted in the picture is that of u_{j-1}, v, u_{j+n-k}. Whatsoever, we will find out the way to bring in the de Casteljeau interpolation. ∎

We are ready for the basic result on de Boor splines:

Proposition and Definition 8.2 *Let as above $u_0 \leq \cdots \leq u_{N+2n-2}$ be a knot sequence with de Boor ordinates $(d_j)_j$. Then the function $B^{(d_j)} : [u_0, u_{N+n-1}] \to \mathbb{R}$ defined by*

$$B^{(d_j)}(v) = \textit{the de Boor ordinate at } v \textit{ after insertion to become a Greville abscissa}$$

is called a deBoor spline.

In fact, $B^{(d_j)}$ is a polynomial function of degree n between consecutive knots, and has class $n-r$ at each knot of multiplicity r.

Proof. We must understand the de Boor spline at a knot $u = u_J$ of multiplicity r, say

$$a = u_{J-r} < u_{J-r+1} = \cdots = u_J = u < u_{J+1} = b.$$

To that end, we first remark that the two de Boor polygons $P^{u \overset{(n}{\cdots} u}$ and $P^{u \overset{(n}{\cdots} uv}$ have the same de Boor ordinate at u. Indeed, $u = \xi_j = \xi^v_j$ for $v \geq u$, $u = \xi_j = \xi^v_{j+1}$ for $v > u$, keeping in any case the ordinate invariant. This remarked, by symmetry (7.3, p.23), $P^{u \overset{(n}{\cdots} uv} = P^{vu \overset{(n}{\cdots} u}$, which means that, before computing near u, we can insert what we want. And consequently we insert knots for a and b to have both multiplicity n:

$$a = u_{J-r-n+1} = \cdots = u_{J-r} < u_{J-r+1} = \cdots = u_J = u < u_{J+1} = \cdots = u_{J+n} = b.$$

Once this is the setting, we must look at three interpolation patches:

(1) For $a < v < u$ we perform n insertions of v to get

$$\alpha(v) = B^{(d_j)}(v) = d_J^n(v).$$

This corresponds to an interpolation triangle T_1 as described above with the following $n+1$ insertion control points

$$d_{J-r-n+1}, \ldots, d_{J-n+1}, \ldots, d_{J-r+1}.$$

(2) For $v = u$, after $n-r$ insertions, we obtain

$$d_{J-r+1}^{n-r}(v),$$

which corresponds to a triangle T_2 with the $n-r+1$ insertion control points

$$d_{J-n+1}, \ldots, d_{J-r+1}.$$

(3) For $u_J \leq v < u_{J+1}$, after n insertions we have

$$\beta(v) = B^{(d_j)}(v) = d_{J+1}^n(v).$$

The triangle T_3 to depict here has the $n+1$ insertion control points

$$d_{J-n+1}, \ldots, d_{J-r+1}, \ldots, d_{J+1}.$$

Now, an attentive look at those three triangles will complete the proof. We firstly see that, regarding indices, the triangles overlap: $T_2 = T_1 \cap T_2 = T_3 \cap T_2$. Next, concerning interpolation ratios, the reader will readily realize the following:

(i) In T_2 interpolation is constant over the interval $[a, b]$.

(ii) In T_1 interpolation is: over $[a, u]$ above T_2, and over $[a, b]$ in T_2.

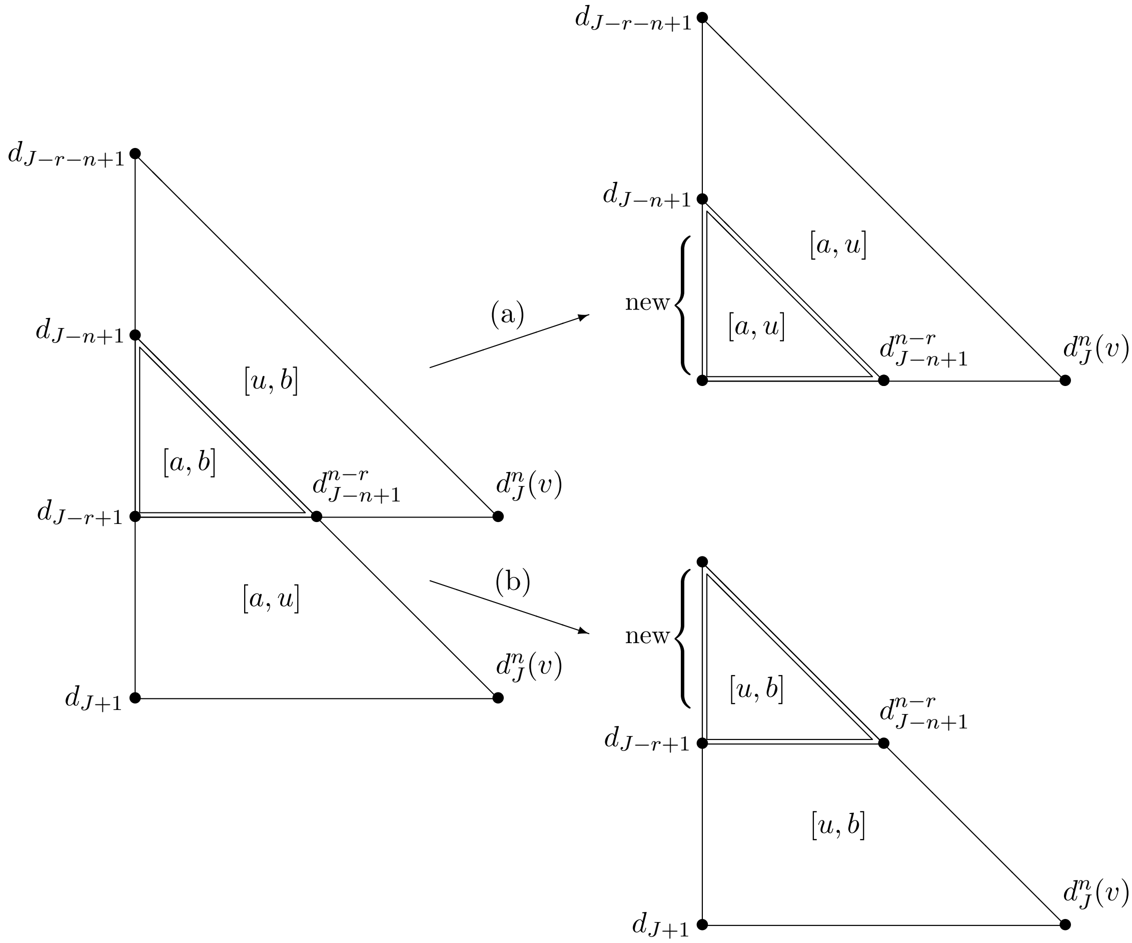
(iii) In T_3 interpolation is: over $[a, b]$ below T_2, and over $[a.b]$ in T_2.

Hence T_2 is in fact a de Casteljeau triangle, which defines on $[a, b]$ a polynomial curve $\gamma(t)$ of degree $n-r+1$. Then, we modify the others as follows:

(a) Replace T_2 in T_1 by the de Casteljeau triangle of the restriction of $\gamma(v)$ to $[a, u]$. This changes the control points, but not the hypothenuse of T_2 (3.7(1.ii), p.10). Consequently the change matches that diagonal of T_1, which becomes the de Casteljeau triangle over $[a, u]$ of the polynomial $\alpha(v)$ with final $n-r+1$ control points coming from the restriction of $\gamma(v)$ to $[a, u]$.

(b) Replace T_2 in T_3 by the de Casteljeau triangle of the restriction of $\gamma(v)$ to $[u, b]$. This changes the control points, but not the base of T_2 (3.7(2.ii), p.10). Consequently the change matches that row of T_3, which becomes the de Casteljeau triangle over $[a, u]$ of the polynomial $\beta(v)$ with first $n-r+1$ control points coming from the restriction of $\gamma(v)$ to $[u, b]$.

All of this we represent in the figure below.



Thus we see that the de Boor spline is a true spline of degree n around u, and has class $n-r$ at u. Indeed, this follows from 5.1, p.16, because as stated above: *the last* $n-r+1$ *control points of* $\alpha(v)$ *and the first* $n-r+1$ *control points of* $\beta(v)$ *are control points of* $\gamma(v)$ *in* $[a,u]$ *and* $[u,b]$*, respectively.*

We are done. ∎

Lecture 9. Nodal splines

In this final lecture we come back to the setting prior to the de Boor elaboration, and present in a more abstract form what that elaboration implies.

Fix a compact interval $[a, b]$ and a partition $a = t_0 < t_1 < \cdots < t_{m+1} = b$. Then consider the vector space $\mathcal{SP}$ of all planar topographical splines of degree n and class $(n - r_1, \ldots, n - r_m)$, which has dimension $N = n + 1 + r_1 + \cdots + r_m$ (2.4, p.7). Recall that a spline $\gamma(t) \in \mathcal{SP}$ has the form $\gamma(t) = (t, f(t))$ and is identified with $f(t)$.

Choose the following knot sequence

$$t_0 = \overset{n}{\cdots} = t_0 < \cdots < t_i = \overset{r_i}{\cdots} = t_i < \cdots < t_{m+1} = \overset{n}{\cdots} = t_{m+1},$$

of course labeled $u_0 < \cdots < u_{N+2n-2}$ (check it!). Then we define a linear mapping:

$$\mathsf{deB} : \mathbb{R}^N \to \mathcal{SP} : (d_j) \mapsto f(t) = B^{(d_j)}$$

(that such a thing is linear is immediate from the nature of the knot insertion procedure). What our main result 8.2, p.26, says is the essential part: that the mapping is well defined.

Once we have defined this mapping deB, it turns out it is *isomorphism*. For, both spaces have the same dimension, and

(∗) deB *is injective.*

Proof. Suppose $(d_j) \not\equiv 0$ and let d_J its first non-null entry. If $J = 0$, then $B^{(d_j)}(a) = B^{(d_j)}(u_0) = d_0 \neq 0$. If $J > 0$, we can pick $u_{J-1} < v < u_J$, and since $d_{J-1} = 0$, we have

$$\begin{cases} d^v_{J-1} = 0, \\ d^v_J = \frac{v - u_{J-1}}{u_{J+n-1} - u_{J-1}} d_J \neq 0. \end{cases}$$

Thus, for the next insertion the situation is analogous. Whence, after n insertions of v, we obtain $B^{(d_j)}(v) \neq 0$. ∎

Next we look at the most natural basis. This is the standard one in $\mathbb{R}^N$, namely

$$e_j = (0, \ldots, 1^{(i}, \ldots, 0), \quad j = 1, \ldots, N,$$

which maps into a basis of our splines vector space:

$$\mathsf{N}_j = B^{e_j}, \quad j = 1, \ldots, N.$$

These are the so-called *nodal* splines of degree n, which are

(1) bump functions:

$$\begin{cases} \mathsf{N}_j(v) = 0 & \text{for } v \leq u_{j-1}, \\ \mathsf{N}_j(v) > 0 & \text{for } u_{j-1} < v < u_{j+n-1}, \\ \mathsf{N}_j(v) = 0 & \text{for } v \geq u_{j+n-1}, \end{cases}$$

and

(2) a partition of unity: $\sum_j \mathsf{N}_j \equiv 1$.

Proof. We see (2) first. Since $\sum_j e_j = (1, \dots, 1)$ and $B^{(1,\dots,1)} \equiv 1$, we have

$$\sum_j \mathsf{N}_j = \sum_j \mathsf{deB}(e_j) = \mathsf{deB}\big(\sum_j e_j\big) = \mathsf{deB}(1, \dots, 1) = B^{(1,\dots,1)} \equiv 1.$$

For (1), note that if all de Boor ordinates are ≥ 0 so is the corresponding de Boor spline. With (2), this implies $0 \leq \mathsf{N}_j \leq 1$. Next notice that when inserting $v < u_{j-1}$ all Greville abscissas we must deal with are Greville abscissas preceding the j-th, hence all de Boor ordinates involved are null entries of e_j, and we deduce that $\mathsf{N}_j(v) = 0$. Similarly for $v \geq u_{j+n-1}$. The remaining fact that $\mathsf{N}_j(v) \neq 0$ for $u_{j-1} < v < u_{j+n-1}$, follows by an argument alike $(*)$ above. ∎

Finally, for any given spline $\gamma(t) \equiv f(t)$ in $\mathcal{SP}$, we have $f(t) = \mathsf{deB}(d_j)$ for uniquely determined de Boor ordinates d_j. Then:

$$f(t) = \mathsf{deB}(d_j) = \mathsf{deB}\big(\sum_j d_j e_j\big) = \sum_j d_j \mathsf{deB}(e_j) = \sum_j d_j \mathsf{N}_j,$$

hence *the de Boor ordinates are the coordinates with respect to the basis of nodal splines.*

Lecture 10. Curves