Lecture 28

Main Points:

- Polar forms for splines

DeBoor points and polar forms for polynomial curves

We saw that any parametric polynomial curve $\gamma(t)$ of degree d (ie. with component functions polynomials in P_d) can be computed by Nested Linear Interpolation from its control points, which can be computed from the polar form F of the curve:

$$P_0 = F[0,0,\ldots,0],\ P_1 = F[0,0,\ldots,0,1],\ \ldots,\ P_d = F[1,1,\ldots,1].$$

We also saw that any reparametrization $\alpha(t) = \gamma((1-t)a+tb)$, also a parametric polynomial curve, can be computed from another set of control points, which come from the same polar form F:

$$Q_0 = F[a,a,\ldots,a],\ Q_1 = F[a,a,\ldots,a,b],\ \ldots,\ Q_d = F[b,b,\ldots,b].$$

Note: $\gamma(t)$ and $\alpha(t)$ are different functions of t, but their images (ranges) in $\mathbb{R}^n$ are the same set of points.

These special sets of inputs to the polar form, using 0 and 1 or a and b, with repeats, are not the only sets of inputs which produce control points. Next, we will show that $\gamma(t)$ can also be computed from other sets of special values computed from the polar form F. For example, suppose that we have a set of values:

$$a_d \le a_{d-1} \le \cdots \le a_1 < b_1 \le b_2 \le \cdots \le b_d,$$

and suppose that t is in the interval (a_1, b_1).

Then we define the DeBoor points P_i with respect to these values as follows. Note: There are i $b's$ in the definition of each P_i.

$$\begin{array}{ccccc}
P_0 & = & F[a_d,\ldots,a_1] & = & F[a_1,\ldots,a_d] \\
P_1 & = & F[a_{d-1},\ldots,b_1] & = & F[a_1,\ldots,a_{d-1},b_1] \\
P_2 & = & F[a_{d-2},\ldots,b_2] & = & F[a_1,\ldots,a_{d-2},b_1,b_2] \\
P_3 & = & F[a_{d-3},\ldots,b_3] & = & F[a_1,\ldots,a_{d-3},b_1,b_2,b_3] \\
 & \vdots & & \vdots & \\
P_{d-1} & = & F[a_1,\ldots,b_{d-1}] & = & F[a_1,b_1,\ldots,b_{d-1}] \\
P_d & = & F[b_1,\ldots,b_d] & &
\end{array}$$

We will show that the same curve $\gamma(t)$ can be computed by Nested Linear Interpolation from these DeBoor control points, using the DeBoor algorithm.

Note that the sequence of inputs in the first column consists simply of d consecutive values shifted from left to right. The lower index i is equal to the number of $b's$, as it was in the case of the reparametrization above using a and b. Since the polar form is symmetric, we can also write the DeBoor points, as in the second column above, as:

$$P_i = F[a_1,\ldots,a_{d-i},b_1,\ldots,b_i],\ i = 0,\ldots,d.$$

Next, since each $a_l < b_m$, we can express the parameter t as an affine sum:

$$t = \frac{b_m - t}{b_m - a_l} a_l + \frac{t - a_l}{b_m - a_l} b_m, \quad 1 \le l, m \le d.$$

Then we can define additional DeBoor points, in the same way that we defined the Bezier points. Note that the number of $b's$ is still equal to i, but the number of $a's$ is reduced by j, which equals the number of $t's$.

$$P_i^j = F[a_1, \ldots, a_{d-i-j}, t, \ldots, t, b_1, \ldots, b_i],\ i = 0, \ldots, d-j.$$

For $j = 0$ the DeBoor points are simply the control points:

$$P_i^0 = P_i.$$

Next, we apply the affine property of polar forms, to obtain recursion formulas. At the first stage, for $j = 1$, we have:

$$\begin{aligned}
P_i^1 &= F[a_1, \ldots, a_{d-i-1}, t, b_1, \ldots, b_i] \\
&= F[a_1, \ldots, a_{d-i-1}, \frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} a_{d-i} + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} b_{i+1}, b_1, \ldots, b_i] \\
&= \frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} F[a_1, \ldots, a_{d-i-1}, a_{d-i}, b_1, \ldots, b_i] + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} F[a_1, \ldots, a_{d-i-1}, b_{i+1}, b_1, \ldots, b_i] \\
&= \frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} P_i^0 + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} P_{i+1}^0, \quad i = 0, \ldots, d-1.
\end{aligned}$$

We can look at this in reverse, and focus on the affine sum:

$$\frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} P_i^0 + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} P_{i+1}^0.$$

The two control points in the sum are defined by their polar form values:

$$P_i^0 = F[a_1, \ldots, a_{d-i-1}, a_{d-i}, b_1, \ldots, b_i],$$

and

$$P_{i+1}^0 = F[a_1, \ldots, a_{d-i-1}, b_{i+1}, b_1, \ldots, b_i].$$

Clearly, these polar form values have identical inputs in all but one coordinate, where we have the values a_{d-i} and b_{i+1}. Moreover, the coefficients use precisely these two values in the affine sum:

$$\frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} = 1.$$

So, we can apply the affine property of F, in the coordinate involving a_{d-i} and b_{i+1}, using the affine sum:

$$\frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} a_{d-i} + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} b_{i+1} = t.$$

This allows us to transform

$$\frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} F[a_1, \ldots, a_{d-i-1}, a_{d-i}, b_1, \ldots, b_i] + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} F[a_1, \ldots, a_{d-i-1}, b_{i+1}, b_1, \ldots, b_i]$$

into

$$F[a_1, \ldots, a_{d-i-1}, \frac{b_{i+1}-t}{b_{i+1}-a_{d-i}} a_{d-i} + \frac{t-a_{d-i}}{b_{i+1}-a_{d-i}} b_{i+1}, b_1, \ldots, b_i]$$

which is by definition equal to:

$$F[a_1, \ldots, a_{d-i-1}, t, b_1, \ldots, b_i] = P_i^1.$$

In summary, since we are starting with control points which are polar form values coming from consecutive subsequences of a sequence of values, it is clear that consecutive control points, say P_i and P_{i+1}, will be defined with inputs that differ in only one coordinate, after rearranging if necessary using the symmetry property. We can then choose fractional coefficients using those two inputs, to form an affine sum of the control points P_i and P_{i+1}, to arrive at a new polar form value with t in that coordinate. This process continues at the next level, and by this iteration, finally gives the polar form value

$$F[t, t, \ldots, t] = \gamma(t).$$

At the second stage, for $j = 2$, we have:

$$\begin{aligned}
P_i^2 &= F[a_1, \ldots, a_{d-i-2}, t, t, b_1, \ldots, b_i] \\
&= F[a_1, \ldots, a_{d-i-2}, \frac{b_{i+1} - t}{b_{i+1} - a_{d-i-1}} a_{d-i-1} + \frac{t - a_{d-i-1}}{b_{i+1} - a_{d-i-1}} b_{i+1}, t, b_1, \ldots, b_i] \\
&= \frac{b_{i+1} - t}{b_{i+1} - a_{d-i-1}} F[a_1, \ldots, a_{d-i-1}, t, b_1, \ldots, b_i] + \frac{t - a_{d-i-1}}{b_{i+1} - a_{d-i-1}} F[a_1, \ldots, a_{d-i-2}, t, b_{i+1}, b_1, \ldots, b_i] \\
&= \frac{b_{i+1} - t}{b_{i+1} - a_{d-i-1}} P_i^1 + \frac{t - a_{d-i-1}}{b_{i+1} - a_{d-i-1}} P_{i+1}^1, \qquad i = 0, \ldots, d-2.
\end{aligned}$$

At the penultimate stage, for $j = d - 1$, we have:

$$\begin{aligned}
P_0^{d-1} &= F[a_1, t, t, \ldots, t] \\
&= F[a_1, \frac{b_1 - t}{b_1 - a_2} a_2 + \frac{t - a_2}{b_1 - a_2} b_1, t, \ldots, t] \\
&= \frac{b_1 - t}{b_1 - a_2} F[a_1, a_2, t, \ldots, t] + \frac{t - a_2}{b_1 - a_2} F[a_1, b_1, t, \ldots, t] \\
&= \frac{b_1 - t}{b_1 - a_2} P_0^{d-2} + \frac{t - a_2}{b_1 - a_2} P_1^{d-2}
\end{aligned}$$

and

$$\begin{aligned}
P_1^{d-1} &= F[b_1, t, t, \ldots, t] \\
&= F[b_1, \frac{b_2 - t}{b_2 - a_1} a_1 + \frac{t - a_1}{b_2 - a_1} b_2, t, \ldots, t] \\
&= \frac{b_2 - t}{b_2 - a_1} F[a_1, a_1, t, \ldots, t] + \frac{t - a_1}{b_2 - a_1} F[a_1, b_2, t, \ldots, t] \\
&= \frac{b_2 - t}{b_2 - a_1} P_1^{d-2} + \frac{t - a_1}{b_2 - a_1} P_2^{d-2}.
\end{aligned}$$

At the last stage, we have:

$$
\begin{aligned}
P_0^d &= F[t,t,t,\dots,t] \\
&= F[\frac{b_1-t}{b_1-a_1}a_1 + \frac{t-a_1}{b_1-a_1}b_1, t, \dots, t] \\
&= \frac{b_1-t}{b_1-a_1}F[a_1,t,\dots,t] + \frac{t-a_1}{b_1-a_1}F[b_1,t,\dots,t] \\
&= \frac{b_1-t}{b_1-a_1}P_0^{d-1} + \frac{t-a_1}{b_1-a_1}P_1^{d-1}
\end{aligned}
$$

We can also write:

$$\gamma(t) = F[t,t,t,\dots,t] = \frac{b_1-t}{b_1-a_1}P_0^{d-1} + \frac{t-a_1}{b_1-a_1}P_1^{d-1}.$$

We have shown previously that the DeBoor algorithm computes a B-spline sum, where the recursion in the DeBoor algorithm comes from the recursion property of the B-spline basis functions. Since a B-spline sum is made up of piecewise polynomials, we should be able to view each piece according to the above scheme. However, when we use a different polar form for each polynomial piece, it would appear that we will generate many sets of DeBoor points by the above process, which do not necessarily overlap consistently. In fact, as we show below, the DeBoor points do overlap and give a consistent evaluation of one spline curve or spline function.

Polar forms for splines

We can write a polar form for each polynomial piece of a spline function. Together, all of these polar forms are related to the nested linear interpolation properties, and hence to the Bernstein basis coefficients, or control points, of the polynomials. It turns out that we can also relate these polar forms to the B-spline coefficients, or DeBoor control points, of the spline function. In other words, the individual sets of control points for each polynomial piece merge together to form one consistent set of control points for a B-spline sum. This could be a B-spline function with control coefficients, or a B-spline parametric curve with control points. This observation can also be used to form a change of basis matrix from the B-spline basis to a truncated power basis.

In particular, suppose that a spline function

$$f(t) \ \in \ V = P_{d,\mathbf{r}}^k[u_0,\dots,u_k]$$

can be represented by polynomials $p_i(t)$ on each subinterval:

$$f(t) = p_i(t), \ t \in [u_{i-1}, u_i).$$

Then each $p_i(t)$ has a polar form

$$F_i[x_1,\dots,x_d].$$

Suppose also that

$$\mathbf{t} = \{t_0,\dots,t_N\}$$

is a knot sequence such that the B-splines of degree d associated to $\mathbf{t}$ are a basis of V.

We have seen that for any $t \in [t_d, t_{N-d})$, $f(t)$ can be written as a B-spline sum, say

$$f(t) = \sum_{j=0}^{N-d-1} c_j \mathcal{B}_j^d(t).$$

In particular, suppose that $t \in [t_J, t_{J+1})$. Then the knot t_J corresponds to some break point u_{i-1}, with

$$t_J = u_{i-1} < u_i = t_{J+1}.$$

We can then use the polar form $F_i[x_1, \ldots, x_d]$ on this interval to obtain the coefficients:

$$C_{i,q} = F_i[t_{q+1}, t_{q+2}, \ldots, t_{q+d}], \qquad q = J - d, \ldots, J.$$

Note: For this interval we can think of the a_k and b_k from the previous section as:

$$a_k = t_{J+1-k}, \;\; b_k = t_{J+k}, \; k = 1, \ldots, d.$$

Now, above we have shown that we can compute $f(t)$, for $t \in [t_J, t_{J+1})$, by the DeBoor algorithm, with control coefficients $C_{i,q}$ coming from the polar form F_i.

But this is exactly the same process used to compute a B-spline sum. Since f is in fact a B-spline sum, we must conclude that the coefficients $C_{i,q}$ are indeed equal to c_j the B-spline coefficients, otherwise we have the same function computed by two different expressions (or sets of coefficients) from the same basis, which is impossible.

Note: This is saying that many of the special values of a polar form for one interval are equal to special values of a polar form on different intervals. The entire set has overlaps which make a consistent set of control coefficients, which are the B-spline coefficients.

This means that if f is given in terms of a truncated power basis, or any other piecewise polynomial basis, then we can write f in terms of the B-spline basis without actually doing the change of basis. This is the same situation that we had in changing from any polynomial basis to the Bernstein basis using the polar form.

We can also extend this to spline curves $\gamma(t)$, where the coordinate functions, say $x(t)$ and $y(t)$, are now in V. As we saw before, we can write such a curve with point coefficients with respect to some basis of V. The DeBoor control points can then be extracted using the polar forms just as we did for the coefficients above, except that now the polar forms are with respect to the function $\gamma(t)$.

Again, suppose that $t \in [t_J, t_{J+1})$, and $t_J = u_{i-1} < u_i = t_{J+1}$. We can then use the polar form $F_i[x_1, \ldots, x_d]$ for $\gamma(t)$ on this interval to obtain the DeBoor control points:

$$P_q^{[0]} = F_i[t_{q+1}, \ldots, t_{q+d}], \qquad q = J - d, \ldots, J.$$

Example:

Let $f(t)$ be the spline function in the vector space $V = P^{10}_{2,\mathbf{r}}[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$, with continuity vector $\mathbf{r} = (1, -1, 1, 1, 1, 1, 0, 0, 1)$, defined in piecewise form as:

$$f(t) = \begin{cases} t, & 0 \le t < 1 \\ (t - \frac{1}{2})^2 + \frac{3}{4}, & 1 \le t < 2 \\ 1 - (t-2)^2, & 2 \le t < 3 \\ (t-4)^2 - 1, & 3 \le t < 4 \\ -1, & 4 \le t < 5 \\ (t-5)^2 - 1, & 5 \le t < 6 \\ 2(t-6), & 6 \le t < 7 \\ -2(t-8), & 7 \le t < 8 \\ (t-8), & 8 \le t < 9 \\ 1 + (t-9) - 2(t-9)^2, & 9 \le t \le 10 \end{cases}$$

The pieces were chosen in this example to give various orders of continuity at the break-points. In particular: order of continuity $r = 1$ at $t = 1, 3, 4, 5, 6, 9$, order of continuity $r = 0$ at $t = 7, 8$ and order of continuity $r = -1$ at $t = 2$.

A knot sequence for a B-spline basis of V can be written as:

$$\mathbf{t} = \{0, 0, 0, 1, 2, 2, 2, 3, 4, 5, 6, 7, 7, 8, 8, 9, 10, 10, 10\}$$

Now also let $f(t) = \sum_{i=0}^{15} c_i \mathcal{B}_i^2(t)$ be the quadratic B-spline form of the same function, with control coefficients $c_0, \ldots, c_{15}$.

We can solve for the coefficients using polar forms. First we need the polar forms of the polynomials on each piece. These can be given as:

$$\begin{array}{rcc}
F_0[u,v] & = & \frac{1}{2}(u+v), \\
 & = & \\
F_1[u,v] & = & (u-\frac{1}{2})(v-\frac{1}{2})+\frac{3}{4}, \\
 & = & \\
F_2[u,v] & = & 1-(u-2)(v-2), \\
 & = & \\
F_3[u,v] & = & (u-4)(v-4)-1, \\
 & = & \\
F_4[u,v] & = & -1, \\
 & = & \\
F_5[u,v] & = & (u-5)(v-5)-1, \\
 & = & \\
F_6[u,v] & = & u+v-12, \\
 & = & \\
F_7[u,v] & = & -(u+v)+16, \\
 & = & \\
F_8[u,v] & = & \frac{1}{2}(u+v)-8, \\
 & = & \\
F_9[u,v] & = & 1+\frac{1}{2}(u+v-18)-2(u-9)(v-9),
\end{array}$$

Next, we find the DeBoor coefficients on each subinterval. First we write them in the form

$$C_{i,j} = F_i[t_{j+1}, t_{j+2}, \ldots, t_{j+d}], \qquad j = J-d, \ldots, J.$$

Then we note that from the DeBoor algorithm for the B-spline sum, these must line up with the coefficients

$$c_j, \quad j = J-d, \ldots, J,$$

and find that indeed the coefficients are equal even though they are being computed with different polar forms. More specifically, for each $i = 1, \ldots, 10$ we find J so that $i-1 = t_J$, and we have:

$$C_{i,j} = c_j, \quad j = J-d, \ldots, J.$$

- On $[0,1)$ we have $J = 2$ and a_2, a_1, b_1, b_2 is given by $0, 0, 1, 2$. So we compute:

$$\begin{array}{ccccccc}
c_0 & = & C_{0,0} & = & F_0[0,0] & = & 0 \\
c_1 & = & C_{0,1} & = & F_0[0,1] & = & \frac{1}{2} \\
c_2 & = & C_{0,2} & = & F_0[1,2] & = & \frac{3}{2}
\end{array}$$

- On $[1,2)$ we have $J = 3$ and a_2, a_1, b_1, b_2 is given by $0, 1, 2, 2$. So we compute:

$$
\begin{aligned}
c_1 &= C_{1,1} = F_1[0,1] = \tfrac{1}{2} \\
c_2 &= C_{1,2} = F_1[1,2] = \tfrac{3}{2} \\
c_3 &= C_{1,3} = F_1[2,2] = 3
\end{aligned}
$$

- On $[2,3)$ we have $J=6$ and a_2, a_1, b_1, b_2 is given by $2,2,3,4$. So we compute:

$$
\begin{aligned}
c_4 &= C_{2,4} = F_2[2,2] = 1 \\
c_5 &= C_{2,5} = F_2[2,3] = 1 \\
c_6 &= C_{2,6} = F_2[3,4] = -1
\end{aligned}
$$

- On $[3,4)$ we have $J=7$ and a_2, a_1, b_1, b_2 is given by $2,3,4,5$. So we compute:

$$
\begin{aligned}
c_5 &= C_{3,5} = F_3[2,3] = 1 \\
c_6 &= C_{3,6} = F_3[3,4] = -1 \\
c_7 &= C_{3,7} = F_3[4,5] = -1
\end{aligned}
$$

- On $[4,5)$ we have $J=8$ and a_2, a_1, b_1, b_2 is given by $3,4,5,6$. So we compute:

$$
\begin{aligned}
c_6 &= C_{4,6} = F_4[3,4] = -1 \\
c_7 &= C_{4,7} = F_4[4,5] = -1 \\
c_8 &= C_{4,8} = F_4[5,6] = -1
\end{aligned}
$$

- On $[5,6)$ we have $J=9$ and a_2, a_1, b_1, b_2 is given by $4,5,6,7$. So we compute:

$$
\begin{aligned}
c_7 &= C_{5,7} = F_5[4,5] = -1 \\
c_8 &= C_{5,8} = F_5[5,6] = -1 \\
c_9 &= C_{5,9} = F_5[6,7] = 1
\end{aligned}
$$

- On $[6,7)$ we have $J=10$ and a_2, a_1, b_1, b_2 is given by $5,6,7,7$. So we compute:

$$
\begin{aligned}
c_8 &= C_{6,8} = F_6[5,6] = -1 \\
c_9 &= C_{6,9} = F_6[6,7] = 1 \\
c_{10} &= C_{6,10} = F_6[7,7] = 2
\end{aligned}
$$

- On $[7,8)$ we have $J=12$ and a_2, a_1, b_1, b_2 is given by $7,7,8,8$. So we compute:

$$
\begin{aligned}
c_{10} &= C_{7,10} = F_7[7,7] = 2 \\
c_{11} &= C_{7,11} = F_7[7,8] = 1 \\
c_{12} &= C_{7,12} = F_7[8,8] = 0
\end{aligned}
$$

- On $[8, 9)$ we have $J = 14$ and a_2, a_1, b_1, b_2 is given by $8, 8, 9, 10$. So we compute:

$$\begin{aligned} c_{12} &= C_{8,12} = F_8[8,8] = 0 \\ c_{13} &= C_{8,13} = F_8[8,9] = \tfrac{1}{2} \\ c_{14} &= C_{8,14} = F_8[9,10] = \tfrac{3}{2} \end{aligned}$$

- On $[9, 10)$ we have $J = 15$ and a_2, a_1, b_1, b_2 is given by $8, 9, 10, 10$. So we compute:

$$\begin{aligned} c_{13} &= C_{9,13} = F_9[8,8] = \tfrac{1}{2} \\ c_{14} &= C_{9,14} = F_9[8,9] = \tfrac{3}{2} \\ c_{15} &= C_{9,15} = F_9[9,10] = 0 \end{aligned}$$

We conclude that the piecewise function $f(t)$ defined above can be written in the B-spline basis for V, with knot sequence $\mathbf{t}$ above, as: (leaving out the first and last B-spline with coefficient zero)

$$f(t) = \frac{1}{2}\cdot\mathcal{B}_1^2(t) + \frac{3}{2}\cdot\mathcal{B}_2^2(t) + 3\cdot\mathcal{B}_3^2(t) + 1\cdot\mathcal{B}_4^2(t) + 1\cdot\mathcal{B}_5^2(t) - 1\cdot\mathcal{B}_6^2(t) - 1\cdot\mathcal{B}_7^2(t) - 1\cdot\mathcal{B}_8^2(t) + 1\cdot\mathcal{B}_9^2(t) + 2\cdot\mathcal{B}_{10}^2(t) + 1\cdot\mathcal{B}_{11}^2(t)$$