DEGREE REDUCTION OF BÉZIER CURVES

DAVE MORGAN

Abstract. This paper opens with a description of Bézier curves. Then, techniques for the degree reduction of Bézier curves, along with a discussion of error analysis, are presented. This information is taken from a selection of previously published material on the subject, so this report should be seen as a compilation of related material rather than a presentation of original research.

1. The bézier curve

Bézier curves were developed independently by P. de Casteljau (whilst working for the Citroen car company in France) in 1959 and by Pierre E. Bézier (working for Rénault) in 1962 [1], although de Casteljau's work didn't come to light until the mid-1970s. Before that time, surfaces such as airplane wings or car body panels (which required a high degree of accuracy to be practically useful) were defined by a series of drawings and templates derived from cross-sectional curves. These were subject to deterioration, so it became preferable to store these curves as coordinates of the points located on the curves. "In any case, the description of space curves remained a tedious and time consuming task ..... it became evident that parametric definition, based on the theoretical works of Bernstein and Schoenberg would be more suitable. [2] This provided the motivation for the development of Bézier curves.

1.1. The analytical definition.

Definition. The **Bernstein polynomials** of degree n are defined analytically:

$$B_i^n(t) = \begin{cases} \binom{n}{i} t^i (1-t)^{n-i} & 0 \le i \le n,\ 0 \le t \le 1 \\ 0 & otherwise \end{cases}$$

Example 1. The degree 3 Bernstein polynomials are thus:

$$\begin{aligned}
B_0^3(t) &= \tbinom{3}{0} t^0 (1-t)^3 &&= \tfrac{3!}{0!3!}(1-t)^3 &&= (1-t)^3 \\
B_1^3(t) &= \tbinom{3}{1} t^1 (1-t)^2 &&= \tfrac{3!}{1!2!} t(1-t)^2 &&= 3t(1-t)^2 \\
B_2^3(t) &= \tbinom{3}{2} t^2 (1-t)^1 &&= \tfrac{3!}{2!1!} t^2(1-t) &&= 3t^2(1-t) \\
B_3^3(t) &= \tbinom{3}{3} t^3 (1-t)^0 &&= \tfrac{3!}{3!0!} t^3(1-t) &&= t^3
\end{aligned}$$

Note. These polynomials are the same ones that we get from the following binomial expansion:

$$[(1-t)+t]^3 = (1-t)^3 + 3t(1-t)^2 + 3t^2(1-t) + t^3$$

A consequence of this expansion is that the Bernstein polynomials sum to unity.

This provides a simpler way to calculate the Bernstein polynomials. For example, the Bernstein polynomials of degree 5 can (with a little knowledge of Pascal's triangle) be immediately written down :

$$(1-t)^5, 5t(1-t)^4, 10t^2(1-t)^3, 10t^3(1-t)^2, 5t^4(1-t), t^5$$

We are now in a position to define the Bézier curve using the Bernstein polynomials as a basis.

Definition. Given a set of $(n+1)$ control points, $\{P_0, P_1, ..., P_n\}$, the Bézier curve of degree n is given by:

$$P_n(t) = \sum_{i=0}^{n} P_i B_i^n(t), \qquad 0 \le t \le 1$$

For arithmetical convenience and graphical clarity, we now restrict our discussion to planar curves. The following example which is based in two dimensions can be readily extended to a 3-dimensional curve in space by using 3-tuples to represent points in space.

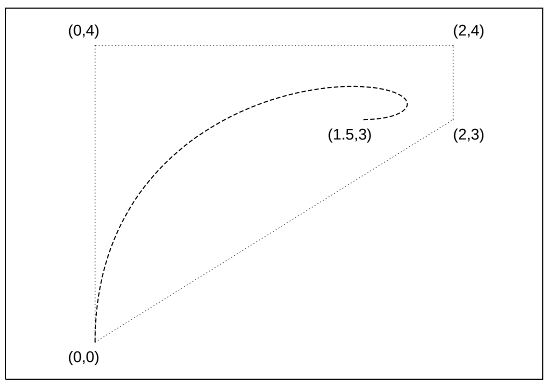

Figure 1. Bezier Curve with control points.

Example 2. Consider the five (arbitrary) control points in the plane,

$$\begin{aligned} P_0 &= (x_0, y_0) = (0,0) \\ P_1 &= (x_1, y_1) = (0,4) \\ P_2 &= (x_2, y_2) = (2,4) \\ P_3 &= (x_3, y_3) = (2,3) \\ P_4 &= (x_4, y_4) = (1.5,3) \end{aligned}$$

They define the quartic Bézier curve (see Figure 1) derived below:

$$\begin{aligned} P_n(t) &= \textstyle\sum_{i=0}^{4} P_i B_i^4(t) \\ &= (0,0)(1-t)^4 + 4(0,4)t(1-t)^3 + 6(2,4)t^2(1-t)^2 \\ &\quad +4(2,3)t^3(1-t) + (1.5,3)t^4 \\ &= (5.5t^4 - 16t^3 + 12t^2, -t^4 + 12t^3 - 24t^2 + 16t) \end{aligned}$$

1.2. The geometric definition.

Definition. (de Casteljau's Algorithm) Given a set of $(n+1)$ control points, $\{P_0, P_1, ..., P_n\}$, the Bézier curve of degree n is given by $P_0^n(t)$, $0 \leq t \leq 1$, where:

$$
\begin{aligned}
P_i^r(t) &= (1-t)P_i^{r-1}(t) + tP_{1+1}^{r-1}(t) && \textit{where } r = 1, 2...., n \\
& && \textit{and } i = 0, 1, ..., n-r \\
P_i^0(t) &= P_i && P_i \textit{ are the control points}
\end{aligned}
$$

This procedure uses repeated linear interpolation, starting with the curve's control points, to arrive at a point on the curve that correspond to a particular value of the parameter t. This, in turn, can be repeated with different t values as necessary to find other points on the curve. Values for the P_i^r are derived from the two entries to the left and above left in the table below. The table starts, in the first column, with the control points.

$$
\begin{array}{lllll}
P_0 \\
P_1 & P_0^1 \\
P_2 & P_1^1 & P_0^2 \\
P_3 & P_2^1 & P_1^2 & P_0^3 \\
P_4 & P_3^1 & P_2^2 & P_1^3 & P_0^4 = P_n(t)
\end{array}
$$

Example 3. We will use the de Casteljau algorithm to find the co-ordinates of a point on the Bézier curve with the same control points as the previous numerical example:

$$
\begin{array}{lllll}
\begin{bmatrix}0\\0\end{bmatrix} \\
\begin{bmatrix}0\\4\end{bmatrix} & \begin{bmatrix}0\\4t\end{bmatrix} \\
\begin{bmatrix}2\\4\end{bmatrix} & \begin{bmatrix}2t\\4\end{bmatrix} & \begin{bmatrix}2t^2\\8t-4t^2\end{bmatrix} \\
\begin{bmatrix}2\\3\end{bmatrix} & \begin{bmatrix}2\\4-t\end{bmatrix} & \begin{bmatrix}4t-2t^2\\4-t^2\end{bmatrix} & \begin{bmatrix}6t^2-4t^3\\12t-12t^2+3t^3\end{bmatrix} \\
\begin{bmatrix}1.5\\3\end{bmatrix} & \begin{bmatrix}2-0.5t\\3\end{bmatrix} & \begin{bmatrix}2-0.5t^2\\4-2t+t^2\end{bmatrix} & \begin{bmatrix}6t-6t^2+1.5t^3\\4-3t^2+2t^3\end{bmatrix} & \begin{bmatrix}12t^2-16t^3+5.5t^4\\16t-24t^2+12t^3-t^4\end{bmatrix} = P_4(t)
\end{array}
$$

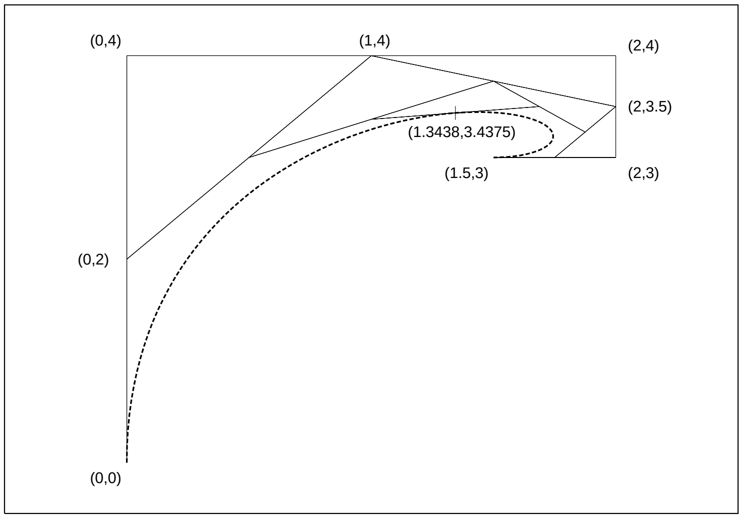

Figure 2. Repeated interpolation with the de Casteljau algorithm.

As we should expect, the values for $P_4(t)$ are the same as those derived from the analytical definition above. However, de Casteljau is not generally used in this way. Instead, co-ordinates on the curve are calculated for a given t values. The entries for $t = 0.5$ are shown below:

$$
\begin{array}{ccccc}
\begin{bmatrix}0\\0\end{bmatrix} & & & & \\
\begin{bmatrix}0\\4\end{bmatrix} & \begin{bmatrix}0\\2\end{bmatrix} & & & \\
\begin{bmatrix}2\\4\end{bmatrix} & \begin{bmatrix}1\\4\end{bmatrix} & \begin{bmatrix}0.5\\3\end{bmatrix} & & \\
\begin{bmatrix}2\\3\end{bmatrix} & \begin{bmatrix}2\\3.5\end{bmatrix} & \begin{bmatrix}1.5\\3.75\end{bmatrix} & \begin{bmatrix}1\\3.375\end{bmatrix} & \\
\begin{bmatrix}1.5\\3\end{bmatrix} & \begin{bmatrix}1.75\\3\end{bmatrix} & \begin{bmatrix}1.875\\3.25\end{bmatrix} & \begin{bmatrix}1.6875\\3.5\end{bmatrix} & \begin{bmatrix}1.3438\\3.4375\end{bmatrix}
\end{array}
$$

1.3. **Bézier curve properties.**

(1) A Bézier curve is a polynomial, the degree of which is one less than the number of control points. (See degree elevation below.) Degree 3, or cubic, Bézier curves are usually used in computer graphics. "Quadratic curves are not flexible enough and going above degree 3 gives rise to complications..." [3]
(2) The curve is contained within the convex hull of its control points. This has a practical application. If the convex hulls of two curves do not intersect, then neither do the curves; the converse is not necessarily true. It is easier to check for the intersection of two polygons than for the intersection of two curves.
(3) Moving any of the control points affects all of the curve, although the effect is most pronounced in the "region" of the moved control point.
(4) The Bézier curve interpolates its end control points.
(5) The tangent vectors to the curve at its end point are in the same direction with the polygon formed by the control points. This can be seen in Figure 2 above.
(6) (The variation diminishing property.) The curve does not cross any straight line more often than the control point polygon does.

(7) The curve is affinely invariant; it does not change shape under any number of linear transformations.

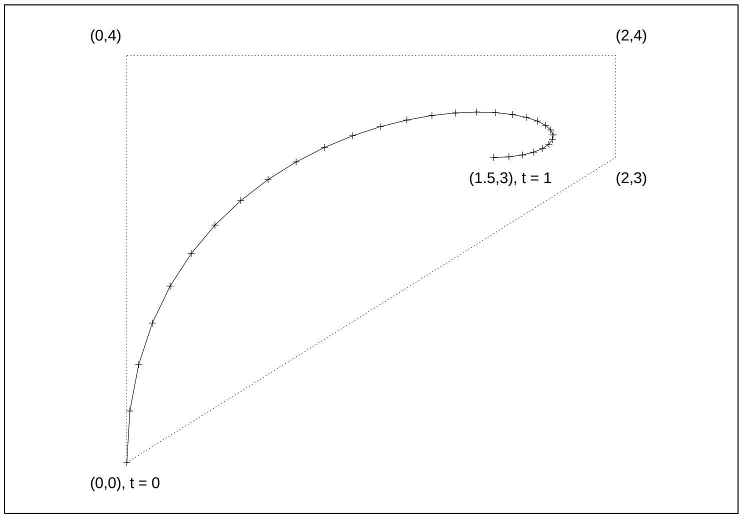

Figure 3. Points on Bezier curve with t = 0.04 step size.

1.4. **Degree elevation of Bézier curves.**

Since degree reduction has some relevance to our main topic of degree reduction, we will discuss it briefly here.

Why would we want to elevate the degree of a Bézier curve?

(1) In general, it is not possible to display the characteristics of a curve of degree $n+1$ with a curve of degree n. For example, we cannot describe a cubic curve with a quadratic function. "Suppose that we are unable to produce a curve of the desired shape with a degree n Bézier curve. One option is to use a Bézier curve of higher degree." [4]
(2) "Degree elevation has important applications in surface design: for several algorithms that produce surfaces from curve input, it is necessary that these curves be of the same degree. Using degree elevation, we may achieve this by raising the degree of all the input curves to the one of the highest degree." [5]
(3) "Another application lies in the area of *data transfer* between different CAD/CAM or graphics systems: Suppose you have generated a parabola (i.e. a degree two Bézier curve), and you want to feed it into a system that only knows about cubics. All you have to do is degree elevate your parabola." [5]

Bartels, Beatty and Barsky tell us [4] that a polynomial of degree n is also a polynomial of degree $n+1$. Therefore, there exists a set of $n+2$ control points Q_i that defines a degree n Bézier curve originally defined by the $n+1$ control points P_i.

For a degree n Bézier curve, the relationship between the original $n+1$ control points P_i and the new $n+2$ control points Q_i is given by the following formula [3]:

$$Q_i = \left(\frac{i}{n+1}\right) P_{i-1} + \left(1 - \frac{i}{n+1}\right) P_i \qquad for\ i = 0, 1, 2, ..., n, n+1$$

Notice that $Q_0 = P_0$ and $Q_{n+1} = P_0$ as we should expect; the end control-points are unchanged by degree elevation.

Example 4. Consider again the quartic Bézier curve defined by the control points:

$$P_0 = (0,0), P_1 = (0,4), P_2 = (2,4), P_3 = (2,3), P_4 = (1.5,3)$$

Degree elevating the curve gives rise to the new set of control points:

$$\begin{array}{lclcl}
Q_0 & = & P_0 & = & (0,0) \\
Q_1 & = & \left(\frac{1}{5}\right)(0,0) + \left(\frac{4}{5}\right)(0,4) & = & \left(0, \frac{16}{5}\right) \\
Q_2 & = & \left(\frac{2}{5}\right)(0,4) + \left(\frac{3}{5}\right)(2,4) & = & \left(\frac{6}{5}, 4\right) \\
Q_3 & = & \left(\frac{3}{5}\right)(2,4) + \left(\frac{2}{5}\right)(2,3) & = & \left(2, \frac{18}{5}\right) \\
Q_4 & = & \left(\frac{4}{5}\right)(2,3) + \left(\frac{1}{5}\right)(1.5,3) & = & \left(\frac{19}{10}, 3\right) \\
Q_5 & = & P_4 & = & (1.5,3)
\end{array}$$

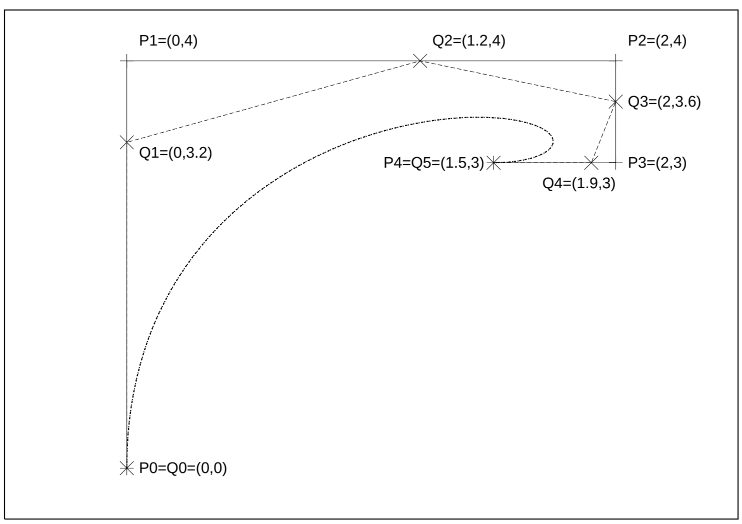

Figure 4. Single Bezier Curve with 5 and 6 Control Points.

Note that:

(1) As soon as one of the new control points Q_i is moved, the Bézier curve is no longer of degree n but immediately becomes degree $n + 1$.
(2) Repeated degree elevation causes the control points to “shrink” inwards towards the curve.

2. DEGREE REDUCTION OF THE BÉZIER CURVE

2.1. Reversing the Degree-Elevation Process.

If degree elevation is viewed as a process that introduces redundancy (using $n+1$ control points to describe a curve that can be accurately described with only n control points), is it not possible that we may be able to reduce possible redundancy by writing a curve of degree n as one of degree $n-1$?

Farin tells us [6] that if we are given an arbitrary Bézier curve of degree $n+1$, we cannot expect to find an identical curve of degree n (unless the $n+1$ degree curve is the result of degree elevation and contains the necessary redundancy). The best that we can hope for is an approximation to the original curve. Farin suggests [5] that we suppose that the curve we wish to degree reduce has been obtained by degree elevation (using the process described above). Then we reverse this imagined degree-elevation process.

We know (from Section 1.4 above) that

$$Q_i = \left(\frac{i}{n+1}\right) P_{i-1} + \left(1 - \frac{i}{n+1}\right) P_i \qquad for\ i = 0, 1, 2, ..., n, n+1$$

gives the $n+2$ control points, Q_i, of the curve, $Q_{n+1}(t)$, degree elevated from $P_n(t)$. Let us rewrite this algorithm for the case where we are degree elevating the curve $P_{n-1}(t)$ to $Q_n(t)$. All this requires is the replacement of n by $n-1$, but will be more consistent with our following discussion of reducing the degree of a Bézier curve from n to $n-1$. The new equation is given by:

$$Q_i = \left(\frac{i}{n}\right) P_{i-1} + \left(\frac{n-i}{n}\right) P_i \qquad for\ i = 0, 1, 2, ..., n$$

Our objective now is to generate the P_i from the Q_i, since the P_i are the n control points of the degree reduced curve in this elevation-reversal process. This gives rise to the following:

$$\begin{aligned} \overrightarrow{P}_i &= \left(\tfrac{n}{n-i}\right)\left(Q_i - \left(\tfrac{i}{n}\right)\overrightarrow{P}_{i-1}\right) \quad for\ i = 0, 1, ..., n-1 \\ &= \left(\tfrac{n}{n-i}\right) Q_i - \left(\tfrac{i}{n-i}\right)\overrightarrow{P}_{i-1} \quad for\ i = 0, 1, ..., n-1 \end{aligned}$$

The Q_i are known, and $P_0 = Q_0$, so the $\overrightarrow{P}_i$ can be derived recursively. (The $\overrightarrow{P}_i$ are the new control points obtained in a "left to right" order starting at $i = 0$ and continuing up to $i = n$.) According to Farin, this recurrence provides a "reasonable" [5][7][8] approximation near P_0.

Note. Although Farin fails to mention it, it is hardly surprising that the approximation does not behave well closer to point Q_n; $Q_n(t)$ interpolates Q_n but the approximation $\overrightarrow{P}_{n-1}(t)$ makes no use at all of the information provided by the end-point of $Q_n(t)$. We could change the end-point Q_n which would dramatically change the path of $Q_n(t)$ while having no effect at all on $\overrightarrow{P}_{n-1}(t)$!

Another formula can be similarly derived, this one counting down from $n-1$:

$$\begin{aligned} \overleftarrow{P}_{i-1} &= \left(\tfrac{n}{i}\right)\left(Q_i - \left(\tfrac{n-i}{n}\right)\overleftarrow{P}_i\right) \quad for\ i = n, n-1, ..., 1 \\ &= \left(\tfrac{n}{i}\right) Q_i - \left(\tfrac{n-i}{i}\right)\overleftarrow{P}_i \quad for\ i = n, n-1, ..., 1 \end{aligned}$$

This approximation performs "decently" [5][7][8] near P_0. If these recursive formulae are applied to a degree elevated curve, the original curve is returned. But, as mentioned above, it is not generally the case that we are dealing with an artificially degree-elevated curve and the best that we can hope for is an approximation.

Since $\overrightarrow{P}$ is reasonable near Q_0 and $\overleftarrow{P}$ the same near Q_n, Farin suggests combining both approximations,

$$P_i = (1-\lambda_i)\overleftarrow{P}_i + \lambda_i \overrightarrow{P}_i \quad for\, i = 0, 1, ..., n-1$$

Farin claims that choosing $\lambda_i = i/n$ does not provide good results, but choosing $\lambda_i = 0$ for $i < n/2$ and $\lambda_i = 1$ for $i > n/2$ provides a reasonable approximation[1]. We will use a numerical example to illustrate Farin's proposed choice for the λ_i.

Example 5. Consider the degree 6 Bézier curve with the control points:

$$Q_0 = (0,0), Q_1 = (2,6), Q_2 = (3,0), Q_3 = (5,4), Q_4 = (7,1), Q_5 = (5,5).Q_6 = (10,6)$$

Applying the first recursive formula,

$$\overrightarrow{P}_i = \left(\frac{6}{6-i}\right) Q_i - \left(\frac{i}{6-i}\right) \overrightarrow{P}_{i-1} \quad for\, i = 0.1, ..., 5$$

we derive the 6 new control points for the degree reduced curve:

$$\begin{array}{lllllll}
\overrightarrow{P}_0 & = & Q_0 & = & (0,0) & & \\
\overrightarrow{P}_1 & = & \left(\frac{6}{5}\right) Q_1 - \left(\frac{1}{5}\right) \overrightarrow{P}_0 & = & \left(\frac{6}{5}\right)(2,6) - \left(\frac{1}{5}\right)(0,0) & = & (2.4, 7.2) \\
\overrightarrow{P}_2 & = & \left(\frac{6}{4}\right) Q_2 - \left(\frac{2}{4}\right) \overrightarrow{P}_1 & = & \left(\frac{6}{4}\right)(3.0) - \left(\frac{2}{4}\right)\left(\frac{12}{5}, \frac{36}{5}\right) & = & (3.3, -3.6) \\
\overrightarrow{P}_3 & = & \left(\frac{6}{3}\right) Q_3 - \left(\frac{3}{3}\right) \overrightarrow{P}_2 & = & (2)(5,4) - (1)(3.3,-3.6) & = & (6.7, 11.6) \\
\overrightarrow{P}_4 & = & \left(\frac{6}{2}\right) Q_4 - \left(\frac{4}{2}\right) \overrightarrow{P}_3 & = & (3)(7,1) - (2)(6.7, 11.6) & = & (7.6, -20.2) \\
\overrightarrow{P}_5 & = & \left(\frac{6}{1}\right) Q_5 - \left(\frac{5}{1}\right) \overrightarrow{P}_4 & = & (6)(5,5) - (5)(7.6, -20.2) & = & (-8, 131)
\end{array}$$

How good is this approximation? $Q_5(1) = Q_6 = (10,6)$ but $\overrightarrow{P}_4(1) = \overrightarrow{P}_5 = (-8,131)$. $\overrightarrow{P}_5$ is supposed to be an approximation for Q_6, so the term approximation is somewhat exaggerated. Applying the second (counting down) recursion formula we obtain the following control points:

$$\overleftarrow{P}_0 = (-18,125), \overleftarrow{P}_1 = (6,-17.8), \overleftarrow{P}_2 = (1.5, 8.9), \overleftarrow{P}_3 = (8.5,-0.9), \overleftarrow{P}_4 = (4,4.8), \overleftarrow{P}_5 = (10,6)$$

Farin now suggests using $\overrightarrow{P}_i$ for $i = 0, 1, ..., \lfloor n/2 \rfloor$ and $\overleftarrow{P}_i$ for $i = \lceil n/2 \rceil, \lceil (n/2)+1 \rceil, ..., n$. (In the case where n is even and $\lfloor n/2 \rfloor = \lceil n/2 \rceil$, there is an odd number of control points, so we take the average value of $\overrightarrow{P}_{n/2}$ and $\overleftarrow{P}_{n/2}$ [7][2]) This gives us the following set of control points with which to construct our degree-reduced curve:

$$\overrightarrow{P}_0 = (0,0), \overrightarrow{P}_1 = (2.4, 7.2), \overrightarrow{P}_2 = (3.3,-3.6), \overleftarrow{P}_3 = (8.5,-0.9), \overleftarrow{P}_4 = (4,4.8), \overleftarrow{P}_5 = (10,6)$$

The original curve and its degree-reduced approximation are shown below:

[1]See Park and Chou [8]for an error analysis for these two choices. Since these choices are less accurate than that proposed by Eck below, there will be no more discussion of these choices in this report.

[2]Note that Eck considers the degree reduction of a curve defined by $n+1$ control points, so his argument applies when n is odd.

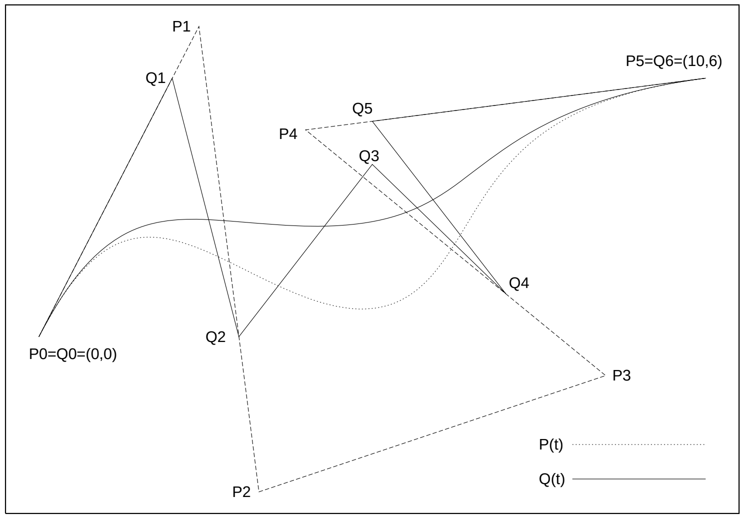

Figure 5. Bezier Curve Q(t) and Degree-Reduced Approximation P(t), Reversed-Elevation Method

This method obviously has serious inaccuracies, particularly in the middle range (as could be anticipated from the above discussion which indicated that both $\overrightarrow{P}(t)$ and $\overleftarrow{P}(t)$ were most satisfactory near their starting points). We look to Matthias Eck for a refinement of the reverse-elevation method.

In order to reduce error in the above discussion, we used the "left" half of $\overrightarrow{P}_i$ and the "right" half of $\overleftarrow{P}_i$. That is,

$$P_n = (1-\lambda_i)\overrightarrow{P}_i + \lambda_i \overleftarrow{P}_i \quad i = 0, 1, ..., n-1 \; and \; \lambda_i = \begin{cases} 0 & if\; i < n/2 \\ 1/2 & if\; i = n/2 \\ 1 & if\; i > n/2 \end{cases}$$

Eck [7] offers a different choice of λ for a "best" degree-reduction with the following theorem. Before that, however, we need to formally state the problem of "best" degree-reduction and to define the forward difference operator.

Problem. (Best degree reduction) *Let $Q_0, Q_1, ..., Q_n$ be a given set of control points defining a Bézier curve, $Q_n(t)$. Then we wish to find (real) control points $P_0, P_1, ..., P_{n-1}$ which define the best uniform approximative (to $Q_n(t)$) Bézier curve, $P_{n-1}(t)$, by minimising the uniform error function*

$$d(Q, P) = max\,\{|Q_n(t) - P_{n-1}(t)| : t \in [0, 1]\}$$

Definition. The nth forward difference of the control points, $\Delta^n Q_i$, is defined:

$$\Delta^n Q_i = \sum_{j=0}^{n} \binom{n}{j} (-1)^{n+j} Q_{i+j}$$

Example 6. $\Delta^4 Q_0 = Q_0 - 4Q_1 + 6Q_2 - 4Q_3 + Q_4$

Note. If $\Delta^n Q_0 = 0$, then the terms in t^n cancel out leaving a curve of degree $n-1$, so we know that the curve is "artificially elevated" and will be reproduced exactly by degree-reduction. So, testing whether $\Delta^n Q_i = 0$ is a simple check for the presence of redundancy.

Theorem. *For $\Delta^n Q_0 \neq 0$ (that is, $Q_n(t)$ is not an artificially elevated curve containing redundancy), the control points $P_0, P_1, ..., P_{n-1}$provide the best solution to the degree-reduction problem iff the factors λ_i are determined by*

$$\lambda_i = 2^{(1-2n)} \cdot \sum_{j=0}^{i} \binom{2n}{2j} \quad \textit{where } i = 0, 1, ..., n-1$$

Furthermore, the maximum approximation error is given by

$$d(Q, P) = 2^{(1-2n)} \cdot |\Delta^n Q_0|$$

For a proof of this theorem, see [7].

Example 7. Let us see how this refinement changes the accuracy of the approximation using the data from the previous example. We have the following:

$$Q_0 = (0,0), Q_1 = (2,6), Q_2 = (3,0), Q_3 = (5,4), Q_4 = (7,1), Q_5 = (5,5).Q_6 = (10,6)$$

$$\overrightarrow{P}_0 = (0,0), \overrightarrow{P}_1 = (2.4, 7.2), \overrightarrow{P}_2 = (3.3, -3.6), \overrightarrow{P}_3 = (6.7, 11.6), \overrightarrow{P}_4 = (7.6, -20.2), \overrightarrow{P}_5 = (-8, 131)$$

$$\overleftarrow{P}_0 = (-18, 125), \overleftarrow{P}_1 = (6, -17.8), \overleftarrow{P}_2 = (1.5, 8.9), \overleftarrow{P}_3 = (8.5, -0.9), \overleftarrow{P}_4 = (4, 4.8), \overleftarrow{P}_5 = (10, 6)$$

Then,

$$\begin{aligned} \Delta^6 Q_0 &= \textstyle\sum_{j=0}^{6} \binom{6}{j} (-1)^{6+j} \cdot Q_j \\ &= -(0,0) + 6(2,6) - 15(3,0) + 20(5,4) - 15(7,1) + 6(5,5) - (10,6) \\ &= (-18, 125) \neq (0,0) \end{aligned}$$

and,

$$\begin{aligned} \lambda_0 &= 2^{-11} \tbinom{12}{0} &&= 2^{-11} \\ \lambda_1 &= 2^{-11} \left\{1 + \tbinom{12}{2}\right\} &&= 2^{-11}(67) \\ \lambda_2 &= 2^{-11} \left\{67 + \tbinom{12}{4}\right\} &&= 2^{-11}(562) \\ \lambda_3 &= 2^{-11} \left\{562 + \tbinom{12}{6}\right\} &&= 2^{-11}(1486) \\ \lambda_4 &= 2^{-11} \left\{1486 + \tbinom{12}{8}\right\} &&= 2^{-11}(1981) \\ \lambda_5 &= 2^{-11} \left\{1981 + \tbinom{12}{10}\right\} &&= 2^{-11}(2047) \end{aligned}$$

Thus,

$$
\begin{aligned}
P_0 &= 2^{-11}(-18, 125) &&= (-0.01, 0.0)\\
P_1 &= \left(1 - 67 \cdot 2^{-11}\right)(2.4, 7.2) + 67 \cdot 2^{-11}(6, -17.8) &&= (2.52, 6.38)\\
P_2 &= \left(1 - 562 \cdot 2^{-11}\right)(3.3, -3.6) + 562 \cdot 2^{-11}(1.5, 8.9) &&= (2.81, -0.17)\\
P_3 &= \left(1 - 1486 \cdot 2^{-11}\right)(6.7, 11.6) + 1486 \cdot 2^{-11}(8.5, 0.9) &&= (8.01, 2.53)\\
P_4 &= \left(1 - 1981 \cdot 2^{-11}\right)(7.6, -20.2) + 1981 \cdot 2^{-11}(4, 4.8) &&= (4.51, 3.98)\\
P_5 &= \left(1 - 2047 \cdot 2^{-11}\right)(-8, 131) + 2047 \cdot 2^{-11}(10, 6) &&= (9.99, 6.06)
\end{aligned}
$$

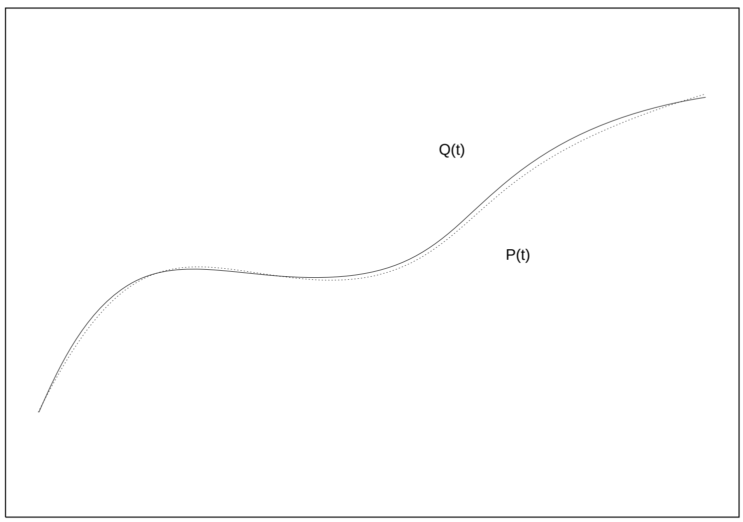

Figure 6. Bezier Curve Q(t) and Optimised Approximation P(t)

Notice that:

(1) Eck's refined method appears to provide a definite improvement over the one suggested by Farin.
(2) This method does not interpolate the end points of the original curve. The divergence between the end points of $Q_n(t)$ and $P_{n-1}(t)$ is more pronounced for small n. We can force endpoint interpolation, but then we are no longer assured a best degree reduction as defined in the prior problem statement. However, Watkins and Worsey [9] argue that if the error incurred in degree reduction is ϵ, then the error incurred in degree reduction with enforced endpoint interpolation is less than or equal to 2ϵ.
(3) For a discussion of higher than C^0 endpoint continuity between $Q_n(t)$ and $P_{n-1}(t)$, refer to Constrained Best Degree Reduction [7]

2.2. Further Reading.

There are more papers available on the subject of Bézier curve degree reduction. Bagacki, Weinstein and Xu discuss degree reduction with endpoint interpolation [10]. Eck improves upon his earlier work with a paper on least squares reduction of Bézier curves [11] although the mathematics is less accessible. Lutterkort, Peters and Reif [12] investigate polynomial degree reduction in the L_2-norm.

References

[1] Boehm, Farin and Hahmann. *A survey of curve and surgace methods in CAGD, Computer-Aided Geometric Design.* Volume 1. (1984), 6

[2] Pierre E. Bézier. *Fundamental Development of Computer-Aided Geometric Modelling*, Academic Press Limited (1993), 14

[3] Alan Watt. *3D Computer Graphics.* 3rd Edition. Addison Wesley, Harlow, Essex, UK. (2000), 77

[4] Bartels, Beatty and Barsky. *An introduction to Splines for use in Computer Graphics and Geometric Modeling,* Morgan Kaufmann Publishers, Los Altos, California 94022, USA. (1987) 213-214.

[5] Gerald Farin. *Curves and Surfaces for Computer-Aided Geometric Design.* Fourth Edition. Academic Press Limited, San Diego, California, USA. (1997), 65

[6] Gerald Farin. *Algorithms for rational Bézier curves.* Computer Aided Design. Volume 15, #2, (1983), 74

[7] Matthias Eck. *Degree reduction of Bézier curves.* Computer Aided Geometric Design. Volume 10. (1993), 240

[8] Y.Park and U.J.Chou. *The Error Analysis for Degree Reducition of Bézier Curves.* Computers Math. Applic. Vol 27 #12 (1994) 3

[9] M A Watkins and A J Worsey. *Degree Reduction of Bézier Curves.* Computer Aided Design. Volume 20, #7, (1988), 400

[10] Przemyslaw Bogacki, Stanley E. Weinstein and Yuesheng Xu. *Degree Reduction of Bézier Curves by Uniform Approximation with Endpoint Interpolation.* Computer Aided Design. Volume 27, #9, (1995) 651-661

[11] Matthias Eck. *Least Squares Degree Reduction of Bézier Curves.* Computer Aided Design, Volume 27, #11, (1995), 845-851

[12] D. Lutterkot, J.Peters,U.Reif. *Polynomial degree reduction in the L_2-norm equals best Euclidean approximation to Bézier coefficients.* Computer Aided Geometric Design, Volume 16, (1999), 607-612