Chapter 3 Interpolation and Polynomial Approximation

Parametric Interpolation/Approximation for Discrete Nodes

In this software, a polynomial or piecewise polynomial parametric interpolation for $(x_1, y_1), (x_2, y_2), \ldots, (x_{n+1}, y_{n+1})$, is done by first introducing its associated parametric sequence $0 = t_1 < t_2 < \cdots < t_{n+1} = 1$, with $\Delta t_i = t_{i+1} - t_i = 1/n$, and then constructing the interpolation polynomial or piecewise polynomial individually for (t_i, x_i), and (t_i, y_i), $i = 1,2,\ldots,n+1$, as in the previous section **Interpolation for Monotonically Increasing Discrete Nodes**. Besides Lagrange, Hermite, free and clamped cubic splines interpolation as in the previous section, Bezier and B-spline interpolation/approximation are new here and are described as below.

Given $n+1$ control points $(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \ldots, (\tilde{x}_{n+1}, \tilde{y}_{n+1})$, Bezier curve is defined as $\big(x(t), y(t)\big)$, where

$$x(t) = \sum_{i=1}^{n+1} \tilde{x}_i B_i^n(t), \quad \mathrm{y}(t) = \sum_{i=1}^{n+1} \tilde{y}_i B_i^n(t), \quad 0 \le t \le 1,$$

and $B_i^n(t)$ is Bernstein polynomial,

$$B_i^n(t) = \binom{n}{i-1} t^{i-1}(1-t)^{n+1-i}$$, where $\binom{n}{i-1}$ is the coefficient of binomial polynomial.

For interpolation in terms of Bezier curve, given $n+1$ interpolation points $(x_1, y_1),.(x_2, y_2),\ldots,(x_{n+1}, y_{n+1})$, $n+1$ control points $(\tilde{x}_1, \tilde{y}_1),(\tilde{x}_2, \tilde{y}_2),\ldots,(\tilde{x}_{n+1}, \tilde{y}_{n+1})$, are reversely determined and then the Bezier curve is computed based on those control points, by de Casteljau algorithm (illustrated in the **tracing point** menu option), and displayed. For approximation in terms of Bezier curve, the input $n+1$ points are seen as the control points and the Bezier curve based on that is computed and displayed.

In the same way, given $n+1$ control points $(\tilde{x}_1, \tilde{y}_1),(\tilde{x}_2, \tilde{y}_2),\ldots,(\tilde{x}_{n+1}, \tilde{y}_{n+1})$, a B-spline of degree $k, k \le n$, is defined as $(x(t), y(t))$, where

$$x(t) = \sum_{i=1}^{n+1} \tilde{x}_i N_i^k(t), \quad y(t) = \sum_{i=1}^{n+1} \tilde{y}_i N_i^k(t),$$

where

$$N_i^k(t) = \left(\frac{t - t_i}{t_{i+k} - t_i} \right) N_i^{k-1}(t) + \left(\frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} \right) N_{i+1}^{k-1}(t),$$

and $N_i^0(t) = \begin{cases} 1, & \text{if } x_i \le t < x_{i+1}, \\ 0, & \text{Otherwise.} \end{cases}$

$N_i^k(t)$ is the basis polynomial of degree k of B-spline which is nonzero only in $[t_i, t_{i+k+1}]$. $[t_1, t_2, \ldots, t_{k+n+2}]$ is called the knot vector, or knot sequence. In this software, the knot vector is determined in the way for uniform non-periodic B-spline:

when $n > k$,

$$t_i = \begin{cases} 0 & i = 1, \cdots, k+1, \\ (i-k-1)/(n-k+1) & i = k+2, \cdots, n+1, \\ 1 & i = n+2, \cdots, k+n+2, \end{cases}$$

with $n-k+2$ distinct values, equally spaced in $[0,1]$, for t_i.

When $n = k$, the upper case is reduced to

$$t_i = \begin{cases} 0 & i = 1, \cdots, k+1, \\ 1 & i = n+2, \cdots, k+n+2. \end{cases}$$

Also when $n = k$, $N_i^k(t) = N_i^n(t) = B_i^n(t)$, and the B-spline will be equivalent to the Bezier curve.

To illustrate knot vector, for example, given $P_1, P_2, \cdots, P_6$ six control points $(n = 5)$, choosing B-spline of degree $k = 3$, the knot vector will be

$$t_i = \begin{cases} 0 & i = 1, \cdots, 4, \\ 1/3 & i = 5, \\ 1/3 & i = 6, \\ 1 & i = 7, \cdots, 10, \end{cases} \quad \text{and}$$

t_i involved with $N_1^3(t)$, associated with P_1, are $[0, 0, 0, 0, 1/3]$;

t_i involved with $N_2^3(t)$, associated with P_2, are $[0, 0, 0, 1/3, 2/3]$;

t_i involved with $N_3^3(t)$, associated with P_3, are $[0, 0, 1/3, 2/3, 1]$;

t_i involved with $N_4^3(t)$, associated with P_4, are $[0, 1/3, 2/3, 1, 1]$;

t_i involved with $N_5^3(t)$, associated with P_5, are $[1/3, 2/3, 1, 1, 1]$;

t_i involved with $N_6^3(t)$, associated with P_6, are $[2/3, 1, 1, 1, 1]$.

For interpolation in terms of a B-spline of degree k, given $n+1$ interpolation points $(x_1, y_1), (x_2, y_2), \ldots, (x_{n+1}, y_{n+1})$, $n+1$ control points $(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \ldots, (\tilde{x}_{n+1}, \tilde{y}_{n+1})$, are reversely determined and then the B-spline of degree k is computed based on those control points, by Cox de Boor algorithm (illustrated in the **tracing point** menu option), and displayed. For approximation in terms of B-spline of degree k, the input $n+1$ points are seen as the control points and the B-spline of degree k based on that is computed and displayed.

References:

【1】 R. L. Burden and J. D. Faires, *Numerical Analysis*, PWS, Boston, 1993.

【2】 D. Kincaid and W. Cheney, *Numerical Analysis*, Brooks/Cole, Pacific Grove, 1996.

【3】 G. Farin, *Curves and Surfaces for Computed Aided Geometric Design*, Academic Press, Boston, 1993.