Knot placement for piecewise polynomial approximation of curves

Gerhard E Hölzle

For piecewise polynomial representation of curves, an algorithm to create knots is presented. The aim is to minimize the interpolation error for a given number of knots or, conversely, the number of knots needed to interpolate within a tolerance. The method used is a modification of de Boor's knot placement scheme. The algorithm described in this paper has been realized in the CADCAM system SYRKO, a Daimler-Benz development for car body design and manufacturing.

Free-form curves, polynomial approximation, knot placement, de Boor's method

THE PROBLEM

In CAD of free-form curves and surfaces, it is often necessary to change the internal representation of a given curve. Examples are:

- approximation of a given curve by a piecewise cubic curve with a fixed number of knots (eg as input for a surface generator)
- approximation of a given curve within a prescribed tolerance by a piecewise cubic curve with minimal number of knots (to save storage or to smooth the curve)
- approximation of a given curve within a prescribed tolerance by a polygon with minimal number of vertices (for output on vector displays, plotters or NC machines)

MATHEMATICAL MODEL

For a formal description of the problem, we suppose that the curve is given in parametric form, ie as a mapping $t \mapsto C(t)$ of an interval $[a, b]$ into Euclidean space. We further suppose that the curve is sufficiently smooth that the derivatives needed in the following analysis can be calculated.

After suitable determination of a set of knots on the curve corresponding to a sequence $t_0, \ldots t_m$ of parameter values with $a = t_0 < t_1 < \ldots < t_m = b$ the curve is to be represented piecewise by polynomials $P_1, \ldots, P_m$ of odd degree n using Hermite interpolation:

$$\left.\begin{aligned} P_i^{(r)}(t_{i-1}) &= C^{(r)}(t_{i-1}) \\ P_i^{(r)}(t_i) &= C^{(r)}(t_i) \end{aligned}\right\} \quad i = 1, \ldots, m\ ; r = 0, \ldots, (n-1)/2$$

Daimler-Benz AG D-7032 Sindelfingen, FRG

The distance between the original curve and its polynomial representation can be defined as the maximal Euclidean distance of two points on the respective curves with the same parameter value.

Setting:

$$\delta_i := \max\ \{|C(t) - P_i(t)|,\ t_{i-1} \leqslant t \leqslant t_i\},\ i = 1, \ldots, m$$

where the vertical bars denote Euclidean norm, the distance between the two curves becomes

$$\max\ \{\delta_i, i = 1, \ldots, m\}$$

DETERMINATION OF KNOTS

An estimate of the error of the interpolation used is given by[1,2]

$$|C(t) - P_i(t)| \leqslant \|C^{(n+1)}\|_{[t_{i-1},\, t_i]}$$
$$((t-t_{i-1})\,(t_i-t))^{(n+1)/2}\,/(n+1)!$$

with

$$\|C^{(n+1)}\|_{[t_{i-1},\, t_i]} := \max\ \{|C^{(n+1)}(s)|,\ t_{i-1} \leqslant s \leqslant t_i\}$$

As $(t-t_{i-1})\,(t_i-t) \leqslant ((t_i-t_{i-1})/2)^2$,

$$\delta_i \leqslant \alpha_i := \|C^{(n+1)}\|_{[t_{i-1},\, t_i]}\ (t_i-t_{i-1})^{n+1}\,/(2^{n+1}\,(n+1)!)$$

Since α_i increases with t_i and decreases with t_{i-1}, we can minimize the bound $\max\ \{\alpha_i, i = 1, \ldots, m\}$ for the curve distance by choosing the sequence t_i so that all α_i are equal. If the given curve is free from sharp bends and oscillations, we can suppose that $C^{(n+1)}$ does not vary too strongly within $[t_{i-1}, t_i]$. We can then use the approximation given by de Boor[1] for α_i:

$$\alpha_i = (\int_{t_{i-1}}^{t_i} \|C^{(n+1)}\|_{[t_{i-1},\, t_i]}^{1/(n+1)}\, ds)^{n+1}\,/(2^{n+1}\,(n+1)!)$$

$$\approx (\int_{t_{i-1}}^{t_i} |C^{(n+1)}(s)|^{1/(n+1)}\, ds)^{n+1}\,/(2^{n+1}\,(n+1)!)$$

Defining an increasing real-valued function g by

$$g(t) := \int_a^t |C^{(n+1)}(s)|^{1/(n+1)}\, ds$$

we get:

$$\alpha_i \approx (g(t_i) - g(t_{i-1}))^{n+1}\,/(2^{n+1}\,(n+1)!)$$

If the number $m + 1$ of knots is given, we find knots so that

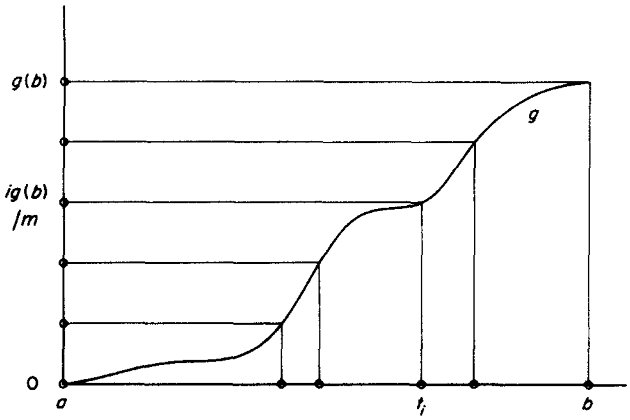

Figure 1. Determination of the knot sequence

all the α_i estimates are equal. Since $g(a) = 0$ conditions are (see Figure 1):

$$g(t_i) = i\,g(b)/m, i = 0, \ldots, m$$

If a tolerance δ is given, we calculate the number $m + 1$ of knots by

$$m := [g(b)/2((n+1)!\,\delta)^{1/(n+1)}] + 1$$

with $[x]$ the largest integer not exceeding x.

While de Boor used this approach to get knot distributions for general local approximation schemes with a fixed number of knots, we applied it here to a special interpolation method to estimate the minimum number of knots for a given error bound as well.

APPLICATION

Let us now consider those cases where the curve is given in piecewise polynomial form. Then we can explicitly calculate derivatives at the given knots up to a certain order and use a divided difference scheme for approximation of higher order derivatives. In this way we can obtain a piecewise constant or linear representation of $|C^{(n+1)}(t)|^{1/(n+1)}$ and by integration a piecewise linear or quadratic function $g(t)$ which is easily inverted to determine knot parameters $t_0, \ldots, t_m$.

We have found that for smooth curves the knots generated by this method with $n = 3$ are also well suited for interpolation by cubic splines instead of Hermite interpolation.

However, because of the approximations inherent in the different steps of the algorithm, one cannot be sure of reducing the number of knots by approximating a piecewise polynomial curve by a curve of the same degree.

EXAMPLES

The uppermost curve in Figure 2 is a centre line of a car bonnet represented as a cubic spline through 18 points. With a prescribed tolerance of .05 mm, the algorithm described above generates 8 knots for $n = 3$. Piecewise cubic Hermite interpolation leads to the second curve in Figure 2 which has an actual distance of .0473 mm from the input curve. For linear interpolation with the same tolerance, 24 points are generated (lowest curve in Figure 2). The resulting distance is here .0489 mm.

Figure 3 shows a part of a contour of a rear window that has been digitized on a 3D-measuring machine (upper curve) and subsequently approximated by a cubic spline with 15 knots (lower curve).

REFERENCES

1 **de Boor, C** *A practical guide to splines* Springer-Verlag, New York (1978)

2 **Conte, S D and de Boor, C** *Elementary numerical analysis* (2nd edition) McGraw-Hill, New York (1972)

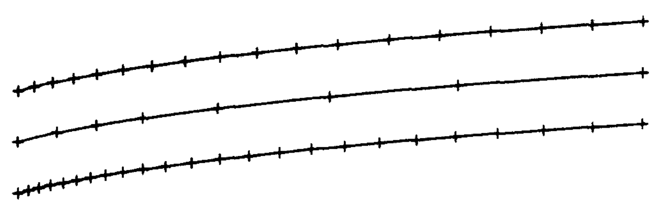

Figure 2. Knot distribution for input spline, piecewise cubic and piecewise linear approximation

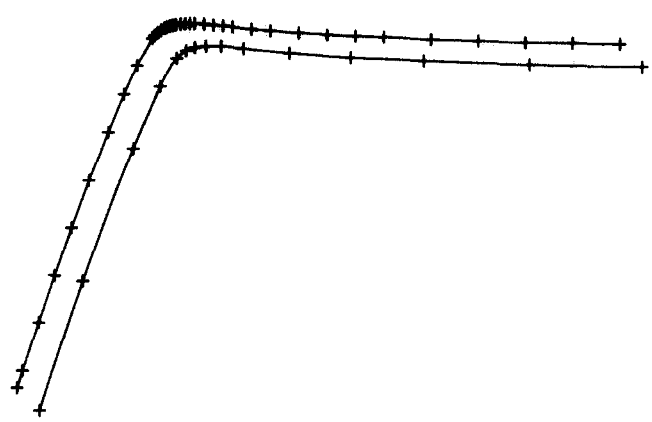

Figure 3. Knot distribution for input polygon and piecewise cubic approximation