ELSEVIER

Computer Aided Geometric Design 15 (1998) 399–416

COMPUTER AIDED GEOMETRIC DESIGN

A method for determining knots in parametric curve interpolation

Caiming Zhang [a], Fuhua (Frank) Cheng [b,*,1], Kenjiro T. Miura [c]

[a] *Department of Computer Science, Shandong University, Jinan, 250100, China*

[b] *Department of Computer Science, University of Kentucky, Lexington, Kentucky 40506, USA*

[c] *Department of Mechanical Engineering, University of Shizuoka, Shizuoka 432, Japan*

Received January 1996; revised July 1997

Abstract

The process of constructing a parametric quadratic polynomial with four data points is discussed and a new method for determining knots in parametric curve interpolation is presented. The method has a parametric polynomial reproduction degree of two, i.e., an interpolation scheme which reproduces quadratic polynomials would reproduce parametric quadratic polynomials if the new method is used to construct knots in the interpolation process. Testing results on the efficiency of the new method are also included. © 1998 Elsevier Science B.V.

Keywords: Parametric curves; Knots; Quadratic polynomials; Interpolation

1. Introduction

In the construction of a parametric curve $P(t)$ to interpolate a set of 2D or 3D data points P_i, $1 \leqslant i \leqslant n$, with $P_i \neq P_{i+1}$, determining the parametric knots t_i, $1 \leqslant i \leqslant n$, where the interpolation takes place, i.e., $P(t_i) = P_i$, is very important as the shape of the constructed curve has much to do with the knots. Using uniform parametrization to determine knots generally leads to unsatisfactory result if the physical spacing of the data points are very uneven. In parametric curve construction, the *accumulated chord length parametrization* (or, simply *chord length method*) is a widely accepted and used method to determine knots (Ahlberg et al., 1967; de Boor, 1978; Brodlie, 1980; Späth, 1974; Faux and Pratt, 1979). The accumulated chord length can be considered as an

* Corresponding author.

[1] Work by this author is supported by the National Science Foundation (NSF-9400823).

PII S0167-8396(97)00041-1

approximation of the accumulated arc length. It has been suggested that by iteration, the chord length parametrization may essentially become the *arc length parametrization* (see (Ahlberg et al. 1967, p. 51; de Boor, 1978, p. 318; Brodlie, 1980, p. 19; Späth, 1974, p. 65)). However, Su and Liu (1982) have proved that if a parametric cubic curve takes its arc length as its parameter, then the curve is a straight line.

Several attempts have been made over the years to improve the knot parametrization process. One direction was to determine the knots using optimization based techniques (Topfer, 1981; Marin, 1984; Hartley and Judd, 1980). The reported results seem to be fair. The optimization process involved in these methods, however, is expensive. Lee (1989) proposed a centripetal model method to assign knots. The knots are taken as the accumulated square root of chord length. This method has been used in several applications (Cheng and Barsky, 1991, 1993; Wang et al., 1997) and the results seem to be satisfactory. Another method to determine knots was proposed by Foley (referred as Foley's method in (Farin, 1988)). In this method the interval of each pair of successive knots is determined not only by the chord length of the corresponding data points, but also by the two adjacent chord lengths, and the two angles between the chord and its two adjacent chords. These methods, according to our experiment results, however, do not seem to generate better approximation than the chord length method (see Section 7).

In this paper we present a new method to determine knots in the construction of planar parametric interpolating curves such as spline curves and parametric polynomial curves. While we are not sure if we have the characteristics of an optimal parametrization of a parametric interpolating curve, it seems to us that the new parametrization technique we are going to present has some interesting features and advantages over the previous approaches. The basic idea of the new method is described in Section 2. The construction of a parametric quadratic polynomial interpolating four data points which is the building block of the new method is discussed in Section 3. Based on the results in Sections 2 and 3, a method for determining the knots is derived in Sections 4, 5 and 6. The comparison of the new method with the chord length, centripetal model and Foley's methods is performed in Section 7.

2. Basic idea

Let $P_i = (x_i, y_i)$, $1 \leqslant i \leqslant n$, be a given set of data points with $P_i \neq P_{i+1}$. The goal is to construct a knot t_i for each P_i, $1 \leqslant i \leqslant n$, so that if the set of data points are taken from a parametric quadratic polynomial, i.e.,

$$P_i = A\xi_i^2 + B\xi_i + C, \quad 1 \leqslant i \leqslant n, \tag{1}$$

where $A = (a_1, a_2)$, $B = (b_1, b_2)$ and $C = (c_1, c_2)$ are 2D points then

$$t_i - t_{i-1} = \alpha(\xi_i - \xi_{i-1}), \quad 1 \leqslant i \leqslant n, \tag{2}$$

for some constant α.

Such a set of knots t_i, $1 \leqslant i \leqslant n$, is known to have a parametric polynomial reproduction degree of two, i.e., an interpolation scheme which reproduces quadratic polynomials will reproduce parametric quadratic polynomials if the knots satisfying Eq. (2) are used in the parametrization process. On the other hand, if the knots determined by the chord length, centripetal model or Foley's methods are used to construct polynomial interpolants, the constructed interpolant can only reproduce straight lines. Hence the knots determined by these three methods have a parametric polynomial reproduction degree of one only.

Let $Q(\xi)$ be a parametric Lagrange polynomial (of degree two) which interpolates P_{i-1}, P_i and P_{i+1} at ξ_{i-1}, ξ_i, and ξ_{i+1}, respectively. $Q(\xi)$ can be converted to a parametric Lagrange polynomial (of degree two) defined on the unit interval $[0, 1]$ as follows:

$$Q(s) = \varphi_1(s)(P_{i-1} - P_i) + \varphi_2(s)(P_{i+1} - P_i) + P_i, \tag{3}$$

where

$$\varphi_1(s) = \frac{(s - s_i)(s - 1)}{s_i},$$

$$\varphi_2(s) = \frac{s(s - s_i)}{1 - s_i}, \tag{4}$$

$$s_i = \frac{\xi_i - \xi_{i-1}}{\xi_{i+1} - \xi_{i-1}}, \tag{5}$$

Obviously, if P_{i-1}, P_i and P_{i+1} are taken from a parametric quadratic polynomial $Q(s)$, and the parameter interval corresponding to P_{i-1} and P_{i+1} is taken to be $[0, 1]$, then there is a unique s_i to make the $Q(s)$ be represented by (3) and (4).

The basic idea in determining the knots t_i, $1 \leqslant i \leqslant n$, may be described as follows: the knots t_i, $1 \leqslant i \leqslant n$, will be constructed in a way so that for each data point P_i, $2 \leqslant i \leqslant n-1$, if a parametric quadratic polynomial $Q_i(s)$ which interpolates the data points P_{i-1}, P_i and P_{i+1} at 0, s_i and 1, respectively, is constructed, then we have

$$\frac{1 - s_i}{s_i} = \frac{t_{i+1} - t_i}{t_i - t_{i-1}},$$

where $0 < s_i < 1$. This equation is equivalent to (5), namely,

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

The motivation is clear—to guarantee a second degree parametric polynomial reproduction rate on each triplet of data points. Let $\Delta_i = t_i - t_{i-1}$, then

$$(1 - s_i)\Delta_i - s_i\Delta_{i+1} = 0, \quad 2 \leqslant i \leqslant n-1. \tag{6}$$

There are $(n-2)$ equations with $(n-1)$ unknowns $\Delta_2, \Delta_3, \ldots, \Delta_n$. If the values of Δ_i, $2 \leqslant i \leqslant n$, are known, the value of the knot t_i can be defined by

$$t_1 = 0,$$

$$t_i = t_{i-1} + \Delta_i, \quad i = 2, 3, \ldots, n. \tag{7}$$

Hence, the key in determining the knots t_i, $1 \leqslant i \leqslant n$, is to determine s_i for each $Q_i(s)$, $i = 2, 3, \ldots, n-1$.

3. Parametric quadratic polynomial

In parametric interpolation, three data points usually can not determine a parametric quadratic polynomial uniquely. For each i between 2 and $n-1$, the parametric quadratic polynomial $Q_i(s)$ defined by (3) will be constructed using 5 data points P_j, $i-2 \leqslant j \leqslant i+2$; P_{i-2} and P_{i+2} are used to determine the parameter s_i.

3.1. Constructing a quadratic polynomial with four points

Let $P_j = (x_j, y_j)$, $i-2 \leqslant j \leqslant i+1$, be four points which are not on the same straight line. The following discussion shows that four points may uniquely determine a quadratic polynomial if they satisfy certain conditions.

For simplicity, a transformation defined by

$$\begin{aligned} v &= \frac{y_{i-1}-y_i}{d}(x-x_i) + \frac{x_i - x_{i-1}}{d}(y-y_i), \\ w &= \frac{y_i - y_{i+1}}{d}(x-x_i) + \frac{x_{i+1}-x_i}{d}(y-y_i) \end{aligned} \tag{8}$$

with

$$d = (x_{i+1}-x_i)(y_{i-1}-y_i) - (x_{i-1}-x_i)(y_{i+1}-y_i)$$

will be applied to P_{i-2}, P_{i-1}, P_i and P_{i+1} first. This transformation changes the coordinates of these points to (v_{i-2}, w_{i-2}), $(0,1)$, $(0,0)$ and $(1,0)$, respectively. In the vw coordinate system, $Q_i(s)$ defined by (3) becomes

$$\begin{aligned} v &= \frac{s(s-s_i)}{1-s_i}, \\ w &= \frac{(s-s_i)(s-1)}{s_i}. \end{aligned} \tag{9}$$

By simple algebra, one gets the following equation:

$$s_i^2 + A(v,w)s_i + B(v,w) = 0, \tag{10}$$

where

$$\begin{aligned} A(v,w) &= -\frac{2v}{v+w}, \\ B(v,w) &= \frac{(1-v)v}{(1-v-w)(v+w)}. \end{aligned}$$

The roots of (10) are

$$s_i = \bar{s}_i = \frac{1}{v+w}\left(v \pm \sqrt{\frac{vw}{v+w-1}}\right). \tag{11}$$

Hence, for any (v,w), there are two possible values for s_i. We now discuss when $(v,w) = (v_{i-2}, w_{i-2})$, which one of the two roots is s_i.

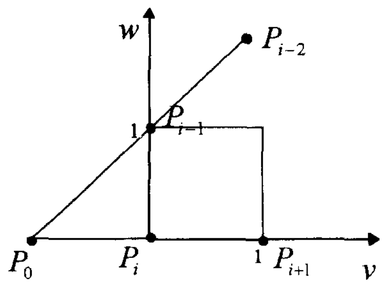

Fig. 1.

Let τ_{i-2} denote the knot corresponding to (v_{i-2}, w_{i-2}). Since the knots corresponding to the data points $(0,1)$, $(0,0)$ and $(1,0)$ are 0, s_i and 1 respectively, $\tau_{i-2} < 0$. Following (9) and (11) we have

$$\tau_{i-2} = v_{i-2} - (v_{i-2} + w_{i-2})s_i + s_i$$
$$= \mp\sqrt{\frac{v_{i-2}w_{i-2}}{v_{i-2} + w_{i-2} - 1}} + s_i < 0. \tag{12}$$

This shows that if $(v, w) = (v_{i-2}, w_{i-2})$, then

$$s_i = \bar{s}_i = \frac{1}{v_{i-2} + w_{i-2}}\left(v_{i-2} + \sqrt{\frac{v_{i-2}w_{i-2}}{v_{i-2} + w_{i-2} - 1}}\right) \tag{13}$$

From (3)–(5) we know that if the given data points are taken from a parametric quadratic polynomial $Q(t)$, then there is a unique s_i satisfying $0 < s_i < 1$ to make the curve $Q_i(s)$ (3) pass through the given data points. Since s_i is determined uniquely by (13), $Q_i(s)$ is equivalent to $Q(t)$. Therefore Theorem 1 follows.

Theorem 1. *If P_j, $i-2 \leqslant j \leqslant i+1$, are taken from a parametric quadratic polynomial $Q(t)$ that is not a straight line, and the knots corresponding to P_{i-1}, P_i and P_{i+1} are ξ_{i-1}, ξ_i and ξ_{i+1} respectively, then the $Q_i(s)$ defined by* (3), (4) *and* (13) *reproduces $Q(t)$ exactly, and s_i* (13) *satisfies*

$$s_i = \frac{\xi_i - \xi_{i-1}}{\xi_{i+1} - \xi_{i-1}}. \tag{14}$$

Substituting $\tau_{i-2} < 0$ into (9) *one obtains*

$$v_{i-2} > 0 \quad \text{and} \quad w_{i-2} > 1 \tag{15}$$

It is easy to prove that if conditions (15) hold, then s_i defined by (13) satisfies $0 < s_i < 1$. Therefore Theorem 2 follows (see Fig. 1).

Theorem 2. *Let $P_j = (x_j, y_j)$, $i-2 \leqslant j \leqslant i+1$, be four given data points, and P_0 the intersection of the line passing through P_{i-2} and P_{i-1} and the line passing through P_i and P_{i+1}. If P_0 and P_{i-2} are not on the same side but P_{i+1} and P_{i-2} are on the same side of the line passing through P_{i-1} and P_i, then P_j, $i-2 \leqslant j \leqslant i+1$, determine a parametric quadratic polynomial $Q_i(s)$ defined by* (3) *uniquely.*

3.2. Determining s_i

In this subsection and the following sections we shall use D_i to denote the distance from point P_i to point P_{i+1}, and Θ_i to denote the angle between the line segment from P_i to P_{i-1} and the line segment from P_i to P_{i+1}.

Let $P_j = (x_j, y_j)$, $i-2 \leqslant j \leqslant i+1$, be four points satisfying the conditions described in Theorem 2. The above discussion shows that they may be used to construct two parametric quadratic polynomials $Q_{i-1}(s)$ and $Q_i(s)$ both defined by (3). The s_{i-1} corresponding to $Q_{i-1}(s)$ is

$$s_{i-1} = \tilde{s}_{i-1} = -\frac{\tau_{i-2}}{\bar{s}_i - \tau_{i-2}}, \tag{16}$$

where τ_{i-2} and $\bar{s}_i$ are defined by (12) and (13), respectively. Therefore the s_i corresponding to $Q_i(s)$ may be taken as $\bar{s}_i$ determined by P_{i-2}, P_{i-1}, P_i and P_{i+1} and (13), or as $\tilde{s}_i$ determined by P_{i-1}, P_i, P_{i+1} and P_{i+2} and (16). The arithmetic mean of $\bar{s}_i$ and $\tilde{s}_i$ is taken as s_i,

$$s_i = \frac{\bar{s}_i + \tilde{s}_i}{2}. \tag{17}$$

If the data points are not on a straight line and their convexity does not change sign, then s_i, $i = 3, 4, \ldots, n-2$, can be determined by (17) uniquely. If their convexity changes sign, s_i in $Q_i(s)$ is determined as follows. The parametric quadratic $Q_i(s)$ is viewed as the trajectory of a moving particle which is required to pass through P_{i-1}, P_i and P_{i+1} with speed $\mathrm{d}Q_i(s)/\mathrm{d}s$. Let $G(s_i)$ denote the speed of the particle at the point P_i or the time s_i. We have

$$G(s_i) = \sqrt{\frac{(1-s_i)^2}{s_i^2} D_{i-1}^2 + \frac{s_i^2}{(1-s_i)^2} D_i^2 + 2D_{i-1}D_i \cos\Theta_i}.$$

To make the particle turn at P_i easily, one needs to make $G(s_i)$ as small as possible. This is achieved by setting

$$\frac{\mathrm{d}G(s_i)}{\mathrm{d}s_i} = 0.$$

The solution is

$$s_i = \frac{\sqrt{D_{i-1}}}{\sqrt{D_{i-1}} + \sqrt{D_i}}.$$

Note that this is the so-called centripetal model (Lee, 1989). However, our derivation seems to be simpler and more reasonable.

The algorithm for determining s_i, $i = 3, 4, \ldots, n-2$, is described as follows:

if P_{k-2}, P_{k-1}, P_k and P_{k+1}, $k = i, i+1$, satisfy the conditions of Theorem 2,
then $s_i = (\bar{s}_i + \tilde{s}_i)/2$; where $\bar{s}_i$ and $\tilde{s}_i$ are defined by (13) and (16), respectively.
else begin
    if P_{i-2}, P_{i-1}, P_i and P_{i+1} satisfy the conditions of Theorem 2,
    then $s_i = \bar{s}_i$;

else begin
    if P_{i-1}, P_i, P_{i+1} and P_{i+2} satisfy the conditions of Theorem 2,
    then $s_i = \tilde{s}_i$;
    else $s_i = \sqrt{D_{i-1}}/(\sqrt{D_{i-1}} + \sqrt{D_i})$;
  end
end

For the end data points, s_2 corresponding to $Q_2(s)$ is determined by (16) using four points P_j, $j = 1, 2, 3, 4$, and s_{n-1} corresponding to $Q_{n-1}(s)$ is determined by (13) using points P_j, $j = n-3, n-2, n-1, n$.

4. Determining knots

Assume a parametric quadratic polynomial $q_i(s)$ has been constructed for each $P_i = (x_i, y_i)$, $2 \leqslant i \leqslant n-1$. From (6), we have

$$(1 - s_i)\Delta_i - s_i\Delta_{i+1} = 0$$

which is linear in two unknowns Δ_i and Δ_{i+1}, $i = 2, 3, \ldots, n-1$. There are $(n-2)$ equations in $(n-1)$ unknowns $\Delta_2, \Delta_3, \ldots, \Delta_n$. We need one additional condition to solve the system of equations for the unknowns. Once one of $\Delta_2, \Delta_3, \ldots, \Delta_n$ is given, the rests are determined uniquely. But this sometimes could results in a bad set of knots. To overcome this shortage, we assign two additional conditions Δ_2 and Δ_n to solve the system. Then we have $(n-2)$ equations in $(n-3)$ unknowns $\Delta_3, \Delta_4, \ldots, \Delta_{n-1}$. These $(n-3)$ unknowns are determined by the least square method. Let $G(\Delta_3, \Delta_4, \ldots, \Delta_{n-1})$ (18) denotes the sum of the squared deviations between $(1 - s_i)\Delta_i$ and $s_i\Delta_{i+1}$,

$$G(\Delta_3, \Delta_4, \ldots, \Delta_{n-1}) = \sum_{i=2}^{n-1} \left[(1 - s_i)\Delta_i - s_i\Delta_{i+1}\right]^2. \tag{18}$$

The $(n-3)$ unknowns $\Delta_3, \Delta_4, \ldots, \Delta_{n-1}$ are determined by minimizing the function $G(\Delta_3, \Delta_4, \ldots, \Delta_{n-1})$. By setting the first partial derivative of $G(\Delta_3, \Delta_4, \ldots, \Delta_{n-1})$ with respect to Δ_i to zero:

$$\frac{\partial G(\Delta_3, \Delta_4, \ldots, \Delta_{n-1})}{\partial \Delta_i} = 0, \quad i = 3, 4, \ldots, n-1,$$

we obtain

$$-h_{i-1}\Delta_{i-1} + f_i\Delta_i - h_i\Delta_{i+1} = 0, \quad i = 3, 4, \ldots, n-1, \tag{19}$$

where

$$h_i = s_i(1 - s_i),$$

$$f_i = s_{i-1}s_{i-1} + (1 - s_i)(1 - s_i).$$

Equations (19) may be written in the following matrix form:

$$\begin{bmatrix} f_3 & -h_3 & 0 & \cdots & 0 \\ -h_3 & f_4 & -h_4 & \cdots & 0 \\ & & \vdots & & \\ 0 & \cdots & -h_{n-3} & f_{n-2} & -h_{n-2} \\ 0 & \cdots & 0 & -h_{n-2} & f_{n-1} \end{bmatrix} \begin{bmatrix} \Delta_3 \\ \Delta_4 \\ \vdots \\ \Delta_{n-2} \\ \Delta_{n-1} \end{bmatrix} = \begin{bmatrix} h_2\Delta_2 \\ 0 \\ \vdots \\ 0 \\ h_{n-1}\Delta_n \end{bmatrix}. \tag{20}$$

When Δ_2 and Δ_n are given, the solutions of (20) are uniquely determined. If the set of data points $P_i = (x_i, y_i)$ with the knots ξ_i, $1 \leqslant i \leqslant n$, are taken from a parametric quadratic polynomial, as shown in (1), and Δ_2, Δ_n are taken as

$$\begin{aligned} \Delta_2 &= \alpha(\xi_2 - \xi_1), \\ \Delta_n &= \alpha(\xi_n - \xi_{n-1}), \end{aligned} \tag{21}$$

where α is a constant, then

$$\Delta_i = \alpha(\xi_i - \xi_{i-1}), \quad i = 3, 4, \ldots, n-1, \tag{22}$$

are the solutions of (20).

The Δ_2 and Δ_n which satisfy (21) are called *compatible end conditions*. Since the solutions of (20) are unique, we have the following theorem.

Theorem 3. *If the system of* (20) *with two* compatible end conditions Δ_2 *and* Δ_n *is used to determine* Δ_i, $3 \leqslant i \leqslant n-1$, *then, the knots defined by*

$$\begin{aligned} t_1 &= 0, \\ t_i &= t_{i-1} + \Delta_i, \quad i = 2, 3, \ldots, n, \end{aligned} \tag{23}$$

have a parametric polynomial reproduction degree of two.

Proof. Follow from (22) and (23), we have

$$t_i - t_{i-1} = \Delta_i = \alpha(\xi_i - \xi_{i-1}). \quad \square$$

5. Determining end conditions

When the given data points are closed, i.e., $(x_1, y_1) = (x_n, y_n)$, we take $\Delta_2 = \Delta_{n+1} = 1$. These are *compatible end conditions*. In the following we consider the end conditions Δ_2 and Δ_n when the data set is not closed.

In parametric case, two seemingly different functions $P(t)$ and $Q(t)$ may represent the same curve. When the given data points $P_i = (x_i, y_i)$, $1 \leqslant i \leqslant n$, are taken from a parametric quadratic polynomial, the $Q_2(s)$ and $Q_{n-1}(s)$ may look different although they represent the same curve. This is because their parameters being different. If there are parameter transformations to transform the parameters of $Q_2(s)$ and $Q_{n-1}(s)$ to the same one, determining the values of Δ_2 and Δ_n becomes easy. When $Q_2(s)$ and $Q_n(s)$

represent the same curve, we may transform them by rotating them to the same parabola of the following form:

$$\bar{y} = a_1\bar{x}^2 + b_1\bar{x} + c_1. \tag{24}$$

Then Δ_2 and Δ_n may be taken as $\Delta_2 = \bar{x}_2 - \bar{x}_1$ and $\Delta_n = \bar{x}_n - \bar{x}_{n-1}$ respectively, where $(\bar{x}_i, \bar{y}_i)$ are coordinates of data point (x_i, y_i) in the $\bar{x}\bar{y}$ coordinate system.

Assume that the transformation

$$\bar{x} = x\cos\beta_i + y\sin\beta_i,$$
$$\bar{y} = -x\sin\beta_i + y\cos\beta_i$$

transforms $Q_i(s)$ to a parabola. In the $\bar{x}\bar{y}$ coordinate system, we have, from (3),

$$\begin{aligned}\bar{x} &= \left[\psi_1(s)(x_{i-1} - x_i) + \psi_2(s)(x_{i+1} - x_i) + x_i\right]\cos\beta_i \\ &\quad + \left[\psi_1(s)(y_{i-1} - y_i) + \psi_2(s)(y_{i+1} - y_i) + y_i\right]\sin\beta_i, \\ \bar{y} &= -\left[\psi_1(s)(x_{i-1} - x_i) + \psi_2(s)(x_{i+1} - x_i) + x_i\right]\sin\beta_i \\ &\quad + \left[\psi_1(s)(y_{i-1} - y_i) + \psi_2(s)(y_{i+1} - y_i) + y_i\right]\cos\beta_i.\end{aligned} \tag{25}$$

When $\sin\beta_i$ and $\cos\beta_i$ are taken as

$$\begin{aligned}\sin\beta_i &= -X_i/\sqrt{X_i^2 + Y_i^2}, \\ \cos\beta_i &= Y_i/\sqrt{X_i^2 + Y_i^2},\end{aligned} \tag{26}$$

where

$$X_i = x_{i-1} - x_i + (x_{i+1} - x_{i-1})s_i,$$
$$Y_i = y_{i-1} - y_i + (y_{i+1} - y_{i-1})s_i$$

the first expression of (25) becomes

$$\bar{x} = L_i s + x_{i-1}\cos\beta_i + y_{i-1}\sin\beta_i, \tag{27}$$

where

$$L_i = (x_{i+1} - x_{i-1})\cos\beta_i + (y_{i+1} - y_{i-1})\sin\beta_i.$$

Substituting (27) into the second expression of (25) to eliminate s and rearranging, a parabola as shown in (24) is obtained. Δ_2 and Δ_n are taken as

$$\begin{aligned}\Delta_2 &= |\bar{x}_2 - \bar{x}_1| = |L_2 s_2|, \\ \Delta_n &= |\bar{x}_n - \bar{x}_{n-1}| = |L_{n-1}(1 - s_{n-1})|.\end{aligned} \tag{28}$$

Theorem 4. *Δ_2 and Δ_n defined by* (28) *are* compatible end conditions.

Proof. Suppose that the set of data points (x_i, y_i) with knots ξ_i, $1 \leqslant i \leqslant n$, is taken from a parametric quadratic polynomial, as shown in (1). From (1) and Theorem 1, we have

$$x_i - x_{i-1} = \big(a_1(\xi_i + \xi_{i-1}) + b_1\big)(\xi_i - \xi_{i-1}),$$
$$y_i - y_{i-1} = \big(a_2(\xi_i + \xi_{i-1}) + b_2\big)(\xi_i - \xi_{i-1})$$

and

$$s_i = (\xi_i - \xi_{i-1})/(\xi_{i+1} - \xi_{i-1})$$

thus it follows from (26) that

$$X_i = a_1(\xi_i - \xi_{i-1})(\xi_{i+1} - \xi_i),$$
$$Y_i = a_2(\xi_i - \xi_{i-1})(\xi_{i+1} - \xi_i).$$

Hence

$$L_2 s_2 = \big[(x_3 - x_1)\cos\beta_2 + (y_3 - y_1)\sin\beta_2\big](\xi_2 - \xi_1)/(\xi_3 - \xi_1)$$
$$= (\xi_2 - \xi_1)(b_1 a_2 - b_2 a_1)/\sqrt{a_1^2 + a_2^2},$$
$$L_{n-1}(1 - s_{n-1}) = (\xi_n - \xi_{n-1})(b_1 a_2 - b_2 a_1)/\sqrt{a_1^2 + a_2^2}.$$

This completes the proof of the theorem. □

6. Algorithm summary

The discussion in Section 3.1 shows that if P_{i-2}, P_{i-1}, P_i and P_{i+1} satisfy the conditions described in Theorem 2, then s_i is defined by (13) uniquely. However, this approach produces bad results sometimes, for example, when P_{i-2}, P_{i-1} and P_i are obviously not on a straight line but P_{i-1}, P_i and P_{i+1} are nearly on a straight line, then s_i defined by (13) is not good. To overcome this shortage, when $\cos\Theta_{i-1} \leqslant -0.99$ or $\cos\Theta_i \leqslant -0.99$, we consider P_{i-2}, P_{i-1}, P_i and P_{i+1} not satisfying the conditions described in Theorem 2.

In Section 5, the formulas of computing Δ_2 and Δ_n are given. However, our experiment (see Section 7) shows that by taking $\Delta_2 = \Delta_n = 1$, one also gets good results when constructing an interpolant to approximate a curve and gets better results when constructing shape preserving interpolants.

The algorithm for assigning knots to data points P_i, $i = 1, 2, \ldots, n$, is summarized as follows:

Step 1: computing $\tilde{s}_i$ and $\bar{s}_i$.

for $i = 3$ to $n - 1$ step 1 do

if P_{i-2}, P_{i-1}, P_i and P_{i+1} satisfy the conditions described in Theorem 2, $\cos\Theta_{i-1} > -0.99$ and $\cos\Theta_i > -0.99$,

then begin

$d = (x_{i+1} - x_i)(y_{i-1} - y_i) - (x_{i-1} - x_i)(y_{i+1} - y_i);$

$v = [(y_{i-1} - y_i)(x_{i-2} - x_i) + (x_i - x_{i-1})(y_{i-2} - y_i)]/d;$

$w = [(y_i - y_{i+1})(x_{i-2} - x_i) + (x_{i+1} - x_i)(y_{i-2} - y_i)]/d;$

$\bar{s}_i = [v + \sqrt{vw/(v + w - 1)}]/(v + w);$

$\tau = v - (v + w)\bar{s}_i + \bar{s}_i;$

$\tilde{s}_{i-1} = -\tau/(\bar{s}_i - \tau)$;
$\bar{w}_i = \tilde{w}_{i-1} = 1$;
end
else begin
$\bar{w}_i = \tilde{w}_{i-1} = 0$;
$\tilde{s}_{i-1} = \sqrt{D_{i-2}}/(\sqrt{D_{i-2}} + \sqrt{D_{i-1}})$;
$\bar{s}_i = \sqrt{D_{i-1}}/(\sqrt{D_{i-1}} + \sqrt{D_i})$;
end
Step 2: computing s_i;
$s_2 = \tilde{s}_2$;
$s_{n-1} = \bar{s}_{n-1}$;
for $i = 3$ to $n-2$ step 1 do
begin
if $\bar{w}_i + \tilde{w}_i > 0$ then $s_i = (\bar{w}_i\bar{s}_i + \tilde{w}_i\tilde{s}_i)/(\bar{w}_i + \tilde{w}_i)$;
else $s_i = \bar{s}_i$;
end
Step 3: computing end conditions Δ_2 and Δ_n;
$X = x_1 - x_2 + (x_3 - x_1)s_2$;
$Y = y_1 - y_2 + (y_3 - y_1)s_2$;
$L = ((x_3 - x_1)Y - (y_3 - y_1)X)/(X^2 + Y^2)$;
$\Delta_2 = |s_2 L|$;
$X = x_{n-2} - x_{n-1} + (x_n - x_{n-2})s_{n-1}$;
$Y = y_{n-2} - y_{n-1} + (y_n - y_{n-2})s_{n-1}$;
$L = ((x_n - x_{n-2})Y - (y_n - y_{n-2})X)/(X^2 + Y^2)$;
$\Delta_n = |(1 - s_{n-1})L|$;
or simply taking $\Delta_2 = \Delta_n = 1$
Step 4: computing Δ_i, $i = 3, 4, \ldots, n-1$, using Eqs. (20);
Step 5: computing t_i, $i = 1, 2, \ldots, n$;
$t_1 = 0$;
for $i = 2$ to n step 1 do $t_i = t_{i-1} + \Delta_i$

7. Experiments

The new method has been compared with the chord length, centripetal and Foley's methods. The comparison is performed by using the knots computed using these methods in the construction of a parametric cubic spline interpolant. If the reproduction degree of the constructed spline is better, then the corresponding knot computation method is considered to be better. For brevity, the cubic splines produced using these methods are called chord spline, centripetal spline, Foley's spline and new spline, respectively. In the new method, two different end conditions are used, 1) Δ_2 and Δ_n is defined by (28), the corresponding spline is called new-1 spline; 2) taking $\Delta_2 = \Delta_n = 1$, the corresponding spline is called new-2 spline; The data points used in the comparison are taken from a primitive cubic curve $F(\tau) = (x(\tau), y(\tau))$,

$$x = x(\tau) = K\psi_0(\tau) + K\psi_1(\tau) + 3\varphi_1(\tau),$$
$$y = y(\tau) = K\psi_0(\tau) - K\psi_1(\tau), \quad K = 1, 2, \ldots, 12,$$

where

$$\varphi_0(\tau) = (\tau - 1)^2(2\tau + 1), \quad \psi_0(\tau) = (\tau - 1)^2\tau,$$
$$\varphi_1(\tau) = \tau^2(-2\tau + 3), \quad \psi_1(\tau) = \tau^2(\tau - 1)$$

are cubic Hermite basis functions on $0 \leqslant \tau \leqslant 1$.

The cubic curve $F(\tau)$ has the following properties: it is convex for $K = 1, 2, 3, 4$, it has two inflection points for $K = 5, 6, 7, 8$, it has one cusp for $K = 9$, and it has one loop for $K = 10, 11, 12$. For $K = 3, 6, 9, 12$, the figures of $F(\tau)$ on the interval $[0, 1]$ are shown in Fig. 2. For $K = 2, 4, 6, 8, 10, 12$, the curvature curves of $F(\tau)$ are given in Fig. 3. In Fig. 3, the curvature curves of $F(\tau)$ is multiplied by 40 for $K = 2$, by 20 for $K = 4$, by 10 for $K = 6$, by 5 for $K = 12$.

First, the comparison is performed on uniform data points in the interval $[0, 1]$ defined as follows:

$$\tau_i = \frac{i}{20}, \quad i = 0, 1, 2, \ldots, 20.$$

To avoid the maximum error occurred near the end points (x_0, y_0) and (x_{20}, y_{20}), the tangent vectors of $F(\tau)$ at $\tau = 0$ and $\tau = 1$ are used as the end conditions to construct the cubic splines. These four methods are compared in terms of absolute error curve $E(t)$ defined by

$$E(t) = \min\{|P(t) - F(\tau)|\}$$
$$= \min\left\{\left|P_i(t) - F(\tau)\right|, \ \tau_i \leqslant \tau \leqslant \tau_{i+1}\right\}, \quad i = 0, 1, 2, \ldots, 19,$$

where $P(t)$ denotes one of the chord spline, centripetal spline, Foley's spline or new spline, $P_i(t)$ is the corresponding part of $P(t)$ on the subinterval $[t_i, t_{i+1}]$, and $F(\tau)$ is the primitive cubic curve constructed above.

The maximum values of the error curve $E(t)$ produced by these four methods are given in Table 1. For $K = 3$, $F(\tau)$ is a quadratic polynomial, the new spline reproduces it exactly. The error 4.194e–15 in Table 1 is due to rounding in the finite precision computation process.

Second, these four methods are compared on non-uniform data points in the interval $[0, 1]$ defined as follows:

$$\tau_i = \left(i + \frac{\sin(i * (20 - i))}{4}\right)\frac{1}{20}, \quad i = 0, 1, 2, \ldots, 20.$$

The maximum values of the error curve $E(t)$ generated by these four methods are shown in Table 2. The four methods have also been compared on data points which divide $[0, 1]$ into 10, 40, ... etc. subintervals. The results are basically the same as those shown in Tables 1–2.

The theoretical derivation in Sections 3–5 shows that when used to assign knots in the construction of polynomial interpolants to data points whose sign of convexity does not change, our method will reproduce parametric quadratic polynomials, while chord length,

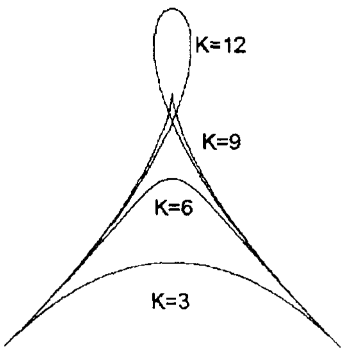

Fig. 2.

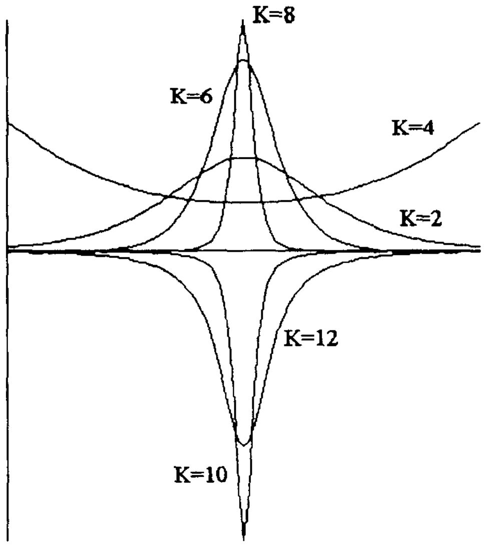

Fig. 3.

centripetal and Foley's methods can only reproduce straight lines. This shows that the reproduction degree of our method is better than those of the chord length, centripetal and Foley's methods. Therefore, we have also used a convex curve ellipse to define data points to compare the reproduction degree of these methods. The ellipse is

$$x = x(\tau) = 3\cos(2\pi\tau),$$
$$y = y(\tau) = 2\sin(2\pi\tau).$$

The interval $[0, 1]$ is divided into 36 sub-intervals to define data points, τ_i as follows:

$$\tau_i = \big(i + \sigma \sin\big((36 - i) * i\big)\big)\frac{1}{36}, \quad i = 0, 1, 2, \ldots, 36.$$

Table 1
Maximum absolute errors

Error	chord	centripetal	Foley	new-1	new-2
$K = 1$	2.348e–4	9.983e–5	3.960e–4	1.770e–5	1.770e–5
$K = 2$	1.982e–5	9.915e–6	1.253e–4	9.616e–6	9.616e–6
$K = 3$	2.223e–5	1.110e–5	1.967e–5	4.194e–15	4.284e–15
$K = 4$	1.594e–5	8.008e–6	1.111e–5	6.817e–6	6.817e–6
$K = 5$	3.064e–5	1.434e–5	1.125e–5	1.338e–4	1.884e–4
$K = 6$	9.499e–5	4.440e–5	3.890e–5	1.046e–4	1.911e–4
$K = 7$	4.216e–4	1.368e–4	6.304e–5	5.296e–4	6.394e–4
$K = 8$	9.107e–4	2.882e–4	8.481e–5	1.612e–4	1.612e–4
$K = 9$	9.097e–4	2.461e–4	1.475e–4	6.412e–4	6.412e–4
$K = 10$	2.048e–3	5.720e–4	1.736e–4	9.399e–5	9.399e–5
$K = 11$	1.190e–3	3.564e–4	1.063e–4	1.427e–4	1.063e–4
$K = 12$	5.744e–4	2.231e–4	1.603e–4	1.187e–4	1.187e–4

Table 2
Maximum absolute errors

Error	chord	centripetal	Foley	new-1	new-2
$K = 1$	3.913e–4	8.065e–4	4.968e–4	1.889e–5	1.680e–5
$K = 2$	2.234e–5	1.290e–3	2.770e–4	1.031e–5	9.236e–6
$K = 3$	2.421e–5	1.870e–3	5.502e–4	2.979e–15	4.885e–6
$K = 4$	4.501e–5	2.365e–3	9.404e–4	1.423e–5	1.466e–5
$K = 5$	5.430e–5	2.701e–3	1.361e–3	2.380e–4	3.882e–4
$K = 6$	2.597e–4	2.728e–3	1.681e–3	4.015e–4	6.209e–4
$K = 7$	1.444e–3	2.056e–3	1.609e–3	1.128e–3	1.645e–3
$K = 8$	4.254e–3	1.372e–3	1.074e–3	3.126e–4	3.078e–4
$K = 9$	1.112e–3	1.838e–3	4.574e–4	7.548e–4	9.343e–4
$K = 10$	6.315e–3	2.310e–3	1.927e–3	4.243e–4	4.255e–4
$K = 11$	3.145e–3	3.703e–3	3.818e–3	3.847e–4	3.834e–4
$K = 12$	1.233e–3	5.550e–3	5.173e–3	3.812e–4	3.786e–4

Table 3
Maximum absolute errors

Error	chord	centripetal	Foley	new-1
$\sigma = .0$	5.104e–5	2.849e–5	2.545e–5	7.271e–6
$\sigma = .05$	8.398e–5	1.384e–3	7.576e–4	1.842e–5
$\sigma = .10$	1.205e–4	2.837e–3	1.558e–3	3.086e–5
$\sigma = .15$	1.604e–4	4.390e–3	2.424e–3	4.455e–5
$\sigma = .20$	2.036e–4	6.040e–3	3.351e–3	5.943e–5
$\sigma = .25$	2.500e–4	7.788e–3	4.337e–3	7.542e–5

where $0 \leqslant \sigma \leqslant 0.25$. Taking $0 \leqslant \sigma \leqslant 0.25$ makes the data points have the property that the adjacent chord lengths D_{i-1} and D_i of the data points satisfy

$$\frac{1}{3} \leqslant \frac{D_{i-1}}{D_i}, \frac{D_i}{D_{i-1}} \leqslant 3.$$

The maximum values of the error curve $E(t)$ generated by these methods are shown in Table 3. In this example, new-1 and new-2 splines are the same.

Additional comparison has also been performed and it shows that if the data points are taken from convex functions, the new method in general gives better approximation than the chord length, the centripetal model or Foley's method.

Four sets of data points have been used to compare the shape of the curves produced by the four methods, with Δ_2 and Δ_n both being set to 1. The four sets of data points are taken from (Akima, 1970; Fritsch and Carlson, 1980; Brodlie, 1980; Lee, 1989), respectively. The curves produced by these methods are shown in Figs. 4–7. Although the new method gives at least as visually pleasing curves as the ones produced by the other methods, it is not suitable for constructing shape preserving interpolants. Just as general spline and spline with tension are used for different purposes, we think that determining knots for constructing interpolants with high precision and for constructing interpolants with visually pleasing shapes are different problems. Our method is suitable for constructing interpolants with high precision. A method for constructing interpolants with visually pleasing shapes suggested by the data points will be studied in a different paper.

8. Conclusions and future works

A new method for determining knots in parametric curve interpolation is presented. The new method can be used in polynomial curve interpolation as well as in spline curve interpolation.

The knots are determined by assuming that the given data points are taken from a parametric quadratic polynomial. The parametric polynomial reproduction degree of the new method is two, i.e., if the data points are taken from a parametric polynomial $Q(t)$ of degree two, then the computed knots can be used to construct interpolants which

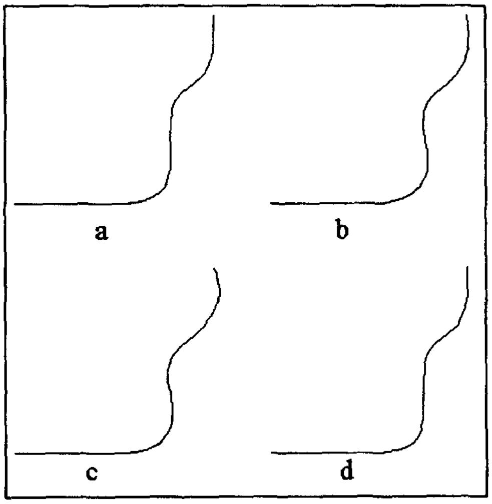

Fig. 4. Data points in (Akima, 1970), $\{x, y\} = \{(0,\ 10), (2,\ 10), (3,\ 10), (5,\ 10), (6,\ 10), (8,\ 10), (9,\ 10.5), (11,\ 15), (12,\ 50), (14,\ 60), (15,\ 85)\}$. (a) centripetal spline, (b) Foley's spline, (c) chord spline, (d) new-2 spline.

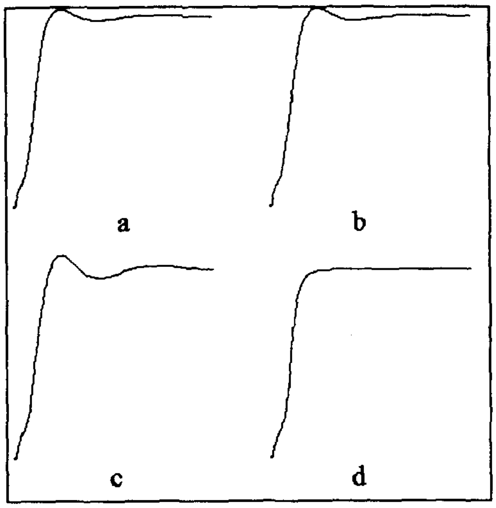

Fig. 5. Data points in (Fritsch and Carlson, 1980), $\{x, y\} = \{(7.99,\ 0), (8.09,\ 2.76429\text{e}{-5}), (8.19,\ 4.37498\text{e}{-2}), (8.7,\ 0.169183), (9.2,\ 0.469428), (10,\ 0.94374), (12,\ 0.998636), (15,\ 0.999919), (20,\ 0.999994)\}$. (a) centripetal spline, (b) Foley's spline, (c) chord spline, (d) new-2 spline.

reproduce $Q(t)$ exactly if the interpolation scheme reproduces quadratic polynomials. On the other hand, if the knots computed using the chord length, centripetal or Foley's method are used to construct the interpolant, $Q(t)$ will not be reproduced even though the interpolation scheme reproduces quadratic polynomials. This means that from the

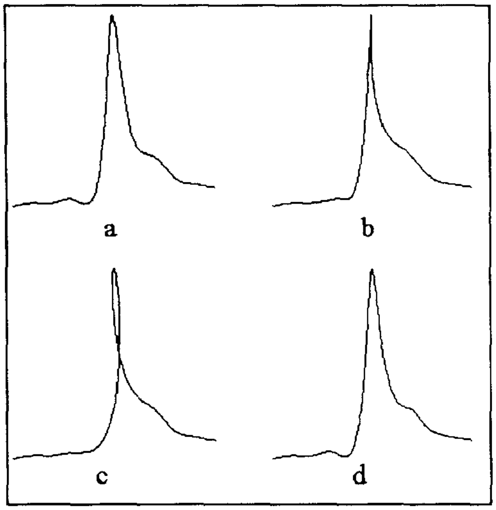

Fig. 6. Data points in (Brodlie, 1980), $\{x, y\} = \{(0,\ 1), (1,\ 1.1), (2,\ 1.1), (3,\ 1.2), (4,\ 1.3), (5,\ 7.2), (6,\ 3.1), (7,\ 2.6), (8,\ 1.9), (9,\ 1.7), (10,\ 1.6)\}$. (a) centripetal spline, (b) Foley's spline, (c) chord spline, (d) new-2 spline.

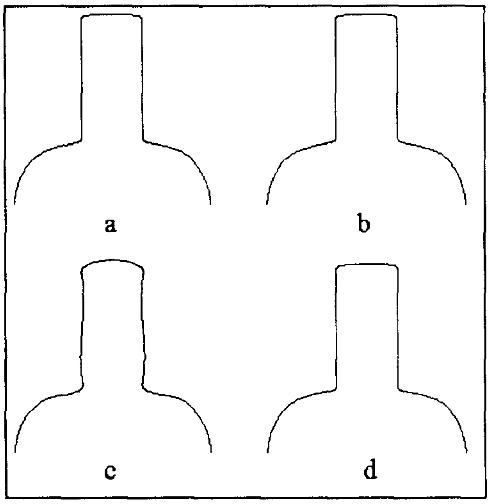

Fig. 7. Data points in (Lie, 1989), $\{x, y\} = \{(0,\ 0), (1.34,\ 5), (5,\ 8.66), (10,\ 10), (10.6,\ 10.4), (10.6,\ 10.4), (10.7,\ 12), (10.7,\ 28.6), (10.8,\ 30.2), (11.4,\ 30.6), (19.6,\ 30.6), (20.2,\ 30.2)\}$. (a) centripetal spline, (b) Foley's spline, (c) chord spline, (d) new-2 spline.

approximation point of view, the new method is better than the chord length, centripetal and Foley's methods in terms of error evaluation in the associated Taylor series.

Our experiment results also indicate that (1) if the convexity of the data points does not change sign, then the new method in general gives better approximation than the other three methods; 2) for uniform data points whose convexity changes sign, the new method has no advantage in approximation over the other three methods; 3) for nonuniform data points whose convexity changes sign, the new method in general gives better approximation results than the other three methods.

It is known that in constructing a cubic spline interpolant, if the two end conditions are taken as the tangent vectors and the knots are compatible, then the constructed parametric cubic spline reproduces parametric cubic polynomials. Our next work is to investigate if there is a method of determining knots whose parametric polynomial reproduction degree is three. Another work is to extend the idea in this paper to determine knots for 3D data points.

References

Ahlberg, J.H., Nilson, E.N. and Walsh, J.L. (1967), *The Theory of Splines and Their Applications*, Academic Press, New York, NY.

de Boor, C. (1978), *A Practical Guide to Splines*, Springer, New York.

Brodlie, K.W. (1980), A review of methods for curve and function drawing, in: Brodile, K.W, ed., *Mathematical Methods in Computer Graphics and Design*, Academic Press, London, 1–37.

Späth, H. (1974), Spline algorithms for curves and surfaces, Utilitas Mathematica, Winnipeg, Canada.

Faux, I.D. and Pratt, M.J. (1979), *Computational Geometry for Design and Manufacture*, Ellis Horwood.

Buqing Su and Dingyuan Liu (1982), *Computational Geometry*, Shanghai Science and Technology Press, Shanghai, 47–48.

Topfer, H.-J. (1981), Models for smooth curve fitting, in: Conf. Proc.: *Numerical Methods of Approximation Theory*, Oberwolfach, FRG, 209–224.

Marin, S.P. (1984), An approach to data parametrization in parametric cubic spline interpolation problems, J. Approx. Theory 41, 64–86.

Hartley, P.J. and Judd, C.J. (1980), Parametrization and shape of B-spline curves for CAD, Computer-Aided Design 12, 235–238.

Lee, E.T.Y. (1989), Choosing nodes in parametric curve interpolation, Computer-Aided Design 21, 363–370.

Cheng, F. and Barsky, B.A. (1991), Interproximation: interpolation and approximation using cubic spline curves, Computer-Aided Design 23, 700–706.

Cheng, F. and Barsky, B.A. (1993), Interproximation using cubic B-spline curves, in: Falcidieno, B. and Kunii, T.L., eds., *Modeling in Computer Graphics: Methods and Applications*, Springer, Berlin, 359–374.

Wang, X., Cheng, F. and Barsky, B.A. (1997), Energy and B-spline interproximation, Computer-Aided Design 27, to appear.

Farin G. (1988), *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, Academic Press.

Fritsch F.N. and Carlson R.E. (1980), Monotone piecewise cubic interpolation, SIAM J. Numer. Anal. 17, 238–246.

Akima H. (1970), A new method of interpolation and smooth curve fitting based on local procedures, J. ACM 17, 589–602.