Computer Aided Geometric Design 18 (2001) 117–127
COMPUTER AIDED GEOMETRIC DESIGN
www.elsevier.com/locate/comaid

Degree reduction of B-spline curves

Jun-Hai Yong [a,*], Shi-Min Hu [a], Jia-Guang Sun [a], Xing-Yu Tan [b]

[a] *Department of Computer Science and Technology, Tsinghua University, Beijing, 100084 PR China*
[b] *Graduate Academy, Beijing Medical University, Beijing, 100083 PR China*

Received 2 February 2000; revised 7 July 2000

Abstract

In this paper, we propose the generalized B divided difference, with which the $(k-1)$th derivative of B-spline curves of order k can be obtained directly without the need to compute the first $(k-2)$ derivatives as before. Based on the generalized B divided difference, the necessary and sufficient condition for degree-reducible B-spline curves is presented. Algorithms for degree reduction of B-spline curves are proposed using the constrained optimization methods. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: B-spline curve; Degenerate condition; Degree reduction; Generalized B divided difference; Constrained optimization

1. Introduction

Degree reduction of parametric curves and surfaces was first proposed as the inverse problem of degree elevation (Forrest, 1972; Farin, 1983). At the beginning, Forrest presented an algorithm for degree reduction of Bézier curves (Forrest, 1972). Lachance applied Chebyshev polynomials and constrained Chebyshev polynomials for degree reduction of polynomial curves and piecewise polynomial curves (Lachance, 1988). Later, Watkins and Worsey proposed an optimal algorithm for degree reduction of Bézier curves via Chebyshev polynomials (Watkins and Worsey, 1988). Now there are many papers about degree reduction of Bézier curves (Brunnett et al., 1996; Brunnett and Schreiber, 1998; Eck, 1993, 1995; Hu et al., 1998; Watkins and Worsey, 1988), Bézier surfaces (Hu et al., 1997) and Ball curves (Hu et al., 1996).

* Corresponding author.
E-mail address: yjh@ncc.cs.tsinghua.edu.cn (J.-H. Yong).

PII: S0167-8396(01)00017-6

By considering each segment of B-spline curves, Piegl and Tiller (1995a, 1995b), Wolters et al. (1998) presented algorithms for degree reduction of B-spline curves. In the algorithm of Piegl and Tiller, each segment of B-spline curves is represented by a Bézier curve via knot insertion. Thus, algorithms for degree reduction of Bézier curves can be applied. After each Bézier curve segment is degree reduced, the B-spline curves with the desired knots are obtained by the knot removal (Tiller, 1992). In the algorithm of Wolters, Wu and Farin, the blossoming principle and the least squares method are used to reduce the degree of each polynomial segment of a B-spline curve directly. And then, the control points of the degree reduced B-spline curve are obtained by the weighting scheme.

This paper presents the necessary and sufficient condition for a degree-reducible B-spline curve, i.e., the curve that can be precisely represented by a B-spline curve of a lower degree. This may be very useful in judging whether the curve is represented in the most compact format. As shown in Section 3, an order k B-spline curve is degree reducible if and only if it is a degenerate curve, i.e., it has vanishing $(k-1)$th derivatives for each polynomial segment. The corresponding equations for the degenerate condition are presented in the form of generalized B divided difference. Via the generalized B divided difference, the mth derivative of B-spline curves can be obtained without calculating the first $(m-1)$ derivatives as usual. Particularly, the $(k-1)$th derivative of B-spline curves of order k can be directly obtained by the generalized B divided difference. Based on the degenerate condition, this paper presents the constraint optimal approaches to obtain degree reduced B-spline curves. Section 4 also gives a method of knot refinement such that the degree reduction is controlled under the given error tolerance.

2. Generalized B divided difference

We represent the equations, which are the necessary and sufficient condition for a degree-reducible B-spline curve, in the form of generalized B divided difference. Hence, we introduce the B divided difference first. Via the generalized B divided difference, the derivatives of B-spline curve can be obtained in a similar way of calculating derivatives of Bézier curves via the forward difference. The generalized B divided difference is a mutation of the generalized divided difference, which is introduced in (Mühlbach, 1973). Given a series of points $\{d_i\}_{i=0,1,\ldots,n}$ and a series of knots $\{u_i\}_{i=0,1,\ldots,n+k}$, the generalized B divided difference of order k can be recursively defined as

$$\begin{cases} \delta_k^0 d_i = d_i, \\ \delta_k^{m+1} d_{i-m-1} = \begin{cases} \dfrac{\delta_k^m d_{i-m} - \delta_k^m d_{i-m-1}}{u_{i+k-m-1} - u_i}, & u_{i+k-m-1} \neq u_i, \\ 0, & u_{i+k-m-1} = u_i, \end{cases} \end{cases} \tag{1}$$

for $m = 0, 1, \ldots .$

We notice that the prescription of $\frac{0}{0} = 0$ makes the de Boor–Cox recursive definition of B-spline basis functions concise and easily used. Similarly we prescribe the quotient $\frac{x}{0} (\forall x \in R)$ to be zero. Thus, formula (1) is collapsed as

$$\begin{cases} \delta_k^0 d_i = d_i, \\ \delta_k^{m+1} d_{i-m-1} = \dfrac{\delta_k^m d_{i-m} - \delta_k^m d_{i-m-1}}{u_{i+k-m-1} - u_i}, \quad \text{for } m = 0, 1, \ldots, \\ \quad \text{Prescribe:} \quad \dfrac{x}{0} = 0, \ \forall x \in R. \end{cases} \tag{2}$$

From formula (2), using mathematical induction, we can prove the following formula

$$\delta_k^m d_{i-m} = \sum_{j=0}^{m} (-1)^j d_{i-j} \sum_{l=0}^{C_m^j - 1} \prod_{h=0}^{m-1} \frac{1}{u_{i-k-m+h-\sigma(h,l,m,j)} - u_{i-\sigma(h,l,m,j)}}, \quad \text{for } m = 1, 2, \ldots, \ k-1, \tag{3}$$

where $\sigma(h,l,m,j) = \sum_{g=0}^{h} \tau(g,l,m,j)$, for $h = 0, 1, \ldots, \ m-1$

$$\tau(g,l,m,j) = \frac{[f(l,m,j) \bmod 2^{m-g}] - [f(l,m,j) \bmod 2^{m-g-1}]}{2^{m-g-1}}, \quad \text{for } g = 0, 1, \ldots, \ m-1,$$

$$\begin{cases} f(l,m,0) = 0, \quad \text{for } l = 0, \\ f(l,m,1) = \begin{cases} 0, & \text{for } l = 0, \\ 2^{l-1}, & \text{for } l = 1, 2, \ldots, m-1, \end{cases} \\ f(l,m,j) = 2^{j+r-2} + f(l - b(j,r), m, j-1), \quad \begin{cases} j = 2, 3, \ldots, m, \\ l = 0, 1, \ldots, C_m^j - 1, \\ b(j,r) \leqslant l < b(j,r+1), \end{cases} \end{cases}$$

$$b(j,t) = \begin{cases} 0, & \text{for } t = 0, \\ C_{j+t-1}^j, & \text{for } t = 1, 2, \ldots. \end{cases}$$

From formula (2), we can also get the following properties of the generalized B divided difference.

(1) Finite. When m is larger than $(k-1)$, $\delta_k^m d_i$ is equal to zero.

(2) Distributive. Suppose there is another series of points $\{\xi_i\}_{i=0,1,\ldots,n}$, we have

$$\delta_k^m (d_{i-m} + \xi_{i-m}) = \delta_k^m d_{i-m} + \delta_k^m \xi_{i-m}.$$

3. Necessary and sufficient condition for degree reducible B-spline curves

Given control points d_i $(i = 0, 1, \ldots, n)$ and a knot vector $U = \{u_0, u_1, \ldots, u_{n+k}\}$, where

$$u_0 = u_1 = \cdots = u_{k-1} < u_k \leqslant u_{k+1} \leqslant \cdots \leqslant u_n < u_{n+1} = u_{n+2} = \cdots = u_{n+k},$$

and the multiplicity of each knot is no more than k, the B-spline curve of order k is

$$p(u)=\sum_{i=0}^{n} d_i N_{i,k}(u). \tag{4}$$

Using the de Boor–Cox recursive definition of B-spline basis functions and the generalized B divided difference, we obtain the mth derivative of B-spline curves

$$p^{(m)}(u)=\frac{d^m p(u)}{du^m}=\frac{(k-1)!}{(k-m-1)!}\sum_{i=m}^{n}\delta_k^m d_{i-m}N_{i,k-m}(u),$$
$$\text{for } m=0,1,\ldots,k-1.$$

Especially, when m is equal to $(k-1)$, we have

$$p^{(k-1)}(u)=(k-1)!\sum_{i=k-1}^{n}\delta_k^{k-1}d_{i-k+1}N_{i,1}(u).$$

We define a degenerate B-spline curve of order k as a B-spline curve of order k which has vanishing $(k-1)$th derivatives for each polynomial segment. Suppose the knot vector is rewritten as

$$U=\Big\{\overbrace{t_0,\ldots,t_0}^{k};\ \overbrace{t_1,\ldots,t_1}^{z_1};\ \overbrace{t_2,\ldots,t_2}^{z_2};\ \ldots;\ \overbrace{t_{T-1},\ldots,t_{T-1}}^{z_{T-1}};\ \overbrace{t_T,\ldots,t_T}^{k}\Big\},$$

where $\{t_i\}_{i=0,1,\ldots,T}$ is a given strictly increasing sequence, $\{z_i\}_{i=1,2,\ldots,T-1}$ is a given positive integer sequence with

$$1\leqslant z_i\leqslant k;\quad i=1,2,\ldots,T-1;$$

the multiplicities of t_0 and t_T are k, and the multiplicities of t_i are z_i, for $i=1,2,\ldots,T-1$. Then, we have the following theorem.

Theorem 1. *The necessary and sufficient condition on the degeneracy of a B-spline curve is*

$$p^{(k-1)}(u)=0,$$

i.e.,

$$\delta_k^{k-1}d_{i-k+1}=0,\quad \textit{for } i=I_0,I_1,\ldots,I_{T-1}, \tag{5}$$

where $\{I_j\}_{j=0}^{T-1}$ *satisfies* $u_{I_j}=t_j$ *and* $u_{I_j}<u_{I_j+1}$.

According to the Curry–Schoenberg theorem (Curry and Schoenberg, 1966); see also (de Boor, 1978) and the basic properties of B-spline (Mühlbach, 1973), a degenerate

B-spline curve can be represented by a B-spline curve with a lower degree, i.e., the following theorem.

Theorem 2. *Suppose a degenerate B-spline curve is a B-spline curve defined by formula* (4)*, and satisfies Eq.* (5)*, then it can be represented as a B-spline curve with a lower degree as follows,*

$$p(u)=\sum_{i=0}^{\tilde{n}}\tilde{d}_i\widetilde{N}_{i,k-1}(u), \tag{6}$$

where

$$\tilde{n}=k-2+\sum_{i=1}^{T-1}y_i,$$

the new knot vector is

$$\widetilde{U}=\Big\{\overbrace{t_0,\ldots,t_0}^{k-1};\ \overbrace{t_1,\ldots,t_1}^{y_1};\ \overbrace{t_2,\ldots,t_2}^{y_2};\ \ldots;\ \overbrace{t_{T-1},\ldots,t_{T-1}}^{y_{T-1}};\ \overbrace{t_T,\ldots,t_T}^{k-1}\Big\},$$

and

$$y_i=\begin{cases}1; & z_i=1,\\ z_i-1; & z_i>1,\end{cases}\quad \textit{for } i=1,2,\ldots,T-1,$$

are the multiplicities of t_i in the degree reduced B-spline curve.

Therefore, the necessary and sufficient condition for a degenerate B-spline curve is also the necessary and sufficient condition for a degree reducible B-spline curve. According to Theorem 2, it is easy to get the following algorithm for degree reduction of a degenerate B-spline curve.

Algorithm 1. Degree reduction of a degenerate B-spline curve

1. Produce the new knot vector as given in Theorem 2.
2. Extend the piecewise polynomial curves defined by Eqs. (4) and (6) respectively according to de Boor–Cox recursive definition.
3. Because the right side of Eq. (4) is equal to the right side of Eq. (6), the corresponding coefficients of every monomial item of the same degree are equivalent for every polynomial segment in (t_i,t_{i+1}) $(i=0,1,\ldots,T-1)$. Therefore, we have a linear equation system, in which variables are the new control points.
4. According to Theorem 2, we know that the linear equation system has only one unique solution. Solve the linear equation system, and get the control points of the degree reduced B-spline curve.
5. End of Algorithm 1.

4. Degree reduction of B-spline curves

4.1. Constrained optimization methods

A degenerate B-spline curve can be degree reduced by Algorithm 1. Now the problem is how to construct a degenerate B-spline curve from the original B-spline curve. A simple way is to disturb control points using a constrained optimization method such that the new curve is a degenerate B-spline curve, and the perturbations between the new control points and the old ones are minimized. Suppose the perturbations of control points d_i are ξ_i, i.e., the new control points are

$$q_i = d_i + \xi_i, \quad i = 0, 1, \ldots, n.$$

Thus, the new B-spline curve is

$$q(u) = \sum_{i=0}^{n} q_i N_{i,k}(u),$$

and the objective is

$$\min\left(\sum_{i=0}^{n} \|\xi_i\|^2\right).$$

The constraint, i.e., Eqs. (5), can be easily written in the matrix form,

$$AQ = 0,$$

where Q is a vector made up of $(n+1)$ control points of the new B-spline curve. The solution to such a constrained optimization problem can be obtained by

$$Q = D + A^{\mathrm{T}}X,$$

where D is a vector made up of $(n+1)$ control points of the original B-spline curve, and X is a vector, which is obtained by

$$AA^{\mathrm{T}}X = -AD.$$

The following considers the case when it is mandatory that the two end points of the new curve should be the coincident with those of the original one. Thus, $q_0 = d_0$ and $q_n = d_n$ are known control points; and the other control points, $q_1, q_2, \ldots, q_{n-1}$, are unknown. Via the transposition of Eqs. (5), the items containing unknown control points are retained on the left side of the equations, and the other items are transposed to the right side. The result can be written in the following matrix form,

$$A_1 Q_1 = b,$$

where Q_1 is a vector made up of $(n-1)$ unknown control points, i.e., $q_1, q_2, \ldots, q_{n-1}$. The solution to the corresponding constrained optimization problem can be obtained by

$$Q_1 = D_1 + A_1^{\mathrm{T}} X_1,$$

where D_1 is a vector made up of $(n-1)$ original inner control points, i.e., $d_1, d_2, \ldots, d_{n-1}$, and X_1 is a vector, which is obtained by

$$A_1 A_1^{\mathrm{T}} X_1 = b - A_1 D_1.$$

4.2. Knot refinement

If a B-spline curve is required to be approximated by a B-spline curve with a lower degree under the given error tolerance, knot refinement can lead to a desired result. Knot refinement is to insert new real numbers as the knots of the original curve before the degree reduction. The corresponding algorithm is shown as follows.

Algorithm 2. Approximate a B-spline curve $p(u)$ with a degree reduced B-spline curve $q(u)$ under the given error tolerance

1. For every simple knot t_i (i.e., the knot which multiplicity is 1), insert t_i as a knot into $p(u)$.
2. Using the constrained optimization methods, obtain the degenerate B-spline curve $c(u)$ corresponding to $p(u)$ (note that, here, $p(u)$ is the B-spline curve after knot insertion).
3. If the error between $c(u)$ and $p(u)$ is larger than the given error tolerance, then
   3.1 Find t_j, at which the maximum error between $c(u)$ and $p(u)$ is obtained.
   3.2 Find the interval $[u_i, u_{i+1}]$ such that
      (1) u_i and u_{i+1} are knots of $p(u)$;
      (2) $[u_i, u_{i+1}]$ contains t_j;
      (3) $[u_i, u_{i+1}]$ has the largest size among all intervals which satisfy (1) and (2).
   3.3 Insert $\frac{u_i+u_{i+1}}{2}$ as a new knot of $p(u)$ twice.
   3.4 Go to Step 2.
4. If the error between $c(u)$ and $p(u)$ is not larger than the given error tolerance, then perform Algorithm 1 on the degenerate curve $c(u)$, and get the degree reduced B-spline curve $q(u)$.
5. End of Algorithm 2.

Suppose that there are two B-spline curves $p_1(u)$ and $p_2(u)$. $p_2(u)$ contains simple knots, while $p_1(u)$ is the result of inserting every simple knot of $p_2(u)$ once as a new knot into $p_2(u)$. Thus, $p_1(u)$ and $p_2(u)$ are parametrically equivalent. After the methods in Section 4.1 are applied, $q_1(u)$ and $q_2(u)$ become the degree reduced B-spline curves of $p_1(u)$ and $p_2(u)$ respectively with no more knots inserted. Then, $q_1(u)$ and $q_2(u)$ have the same knot vector and the same number of control points; however, the error tolerance between $q_1(u)$ and $p_1(u)$ is in general, smaller than that between $q_2(u)$ and $p_2(u)$. Therefore, although the constrained optimization methods can directly deal with B-spline curves which contain simple knots, Step 1 converts them into the B-spline curves which have no simple knots.

Step 3 seems a little complicated. However, if Step 3 simply inserts t_j as a new knot into the $p(u)$, it will have the risk that $p(u)$ may have the knot of multiplicity greater than the order of $p(u)$ after knot insertion. This may also lead to a dead loop, i.e., the same real number may be inserted again and again and the given error tolerance will never be satisfied. The method for new knots inserted in (Wolters et al., 1998) ignores this risk.

5. Examples

Fig. 1 explains how degree reduction algorithms in Section 4.1 works. In Fig. 1, the original curve is an order 6 B-spline curve over the knots {0, 0, 0, 0, 0, 0, 1.5, 1.5, 3, 4.5, 4.5, 6, 6, 6, 6, 6, 6}. At the first step, the original curve is approximated by a degenerate B-spline curve. At the second step, the degenerate B-spline curve is represented by the degree reduced B-spline curve.

Fig. 2 shows that the error between the degree reduced curve and the original curve becomes smaller after all the simple knots are inserted as knots once. The original curve in Fig. 2 is an order 5 B-spline curve over the knots {0, 0, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5, 5, 5}. In Fig. 2 and the following figure, the notation "M1" refers to our algorithm without the end points constraint, whereas "M2" refers to our algorithm with the two end points constraint.

The results of different algorithms are illustrated in Fig. 3, where Piegl and Tiller's algorithm is marked with "P1", and Wolters, Wu and Farin's algorithm is noted as "W1". In this figure, the original curve is an order 8 B-spline curve over the knots {0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2}. Using the same curve, Table 1 gives

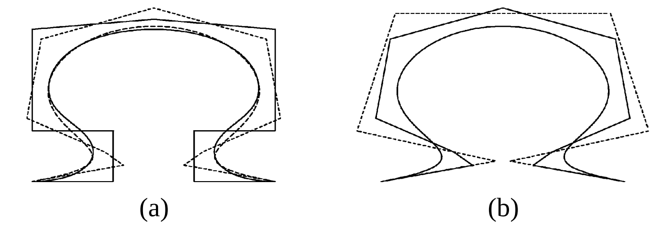

Fig. 1. Procedures of our algorithms for degree reduction. (a) Step 1: to the degenerate curve. (b) Step 2: to the degree reduced curve.

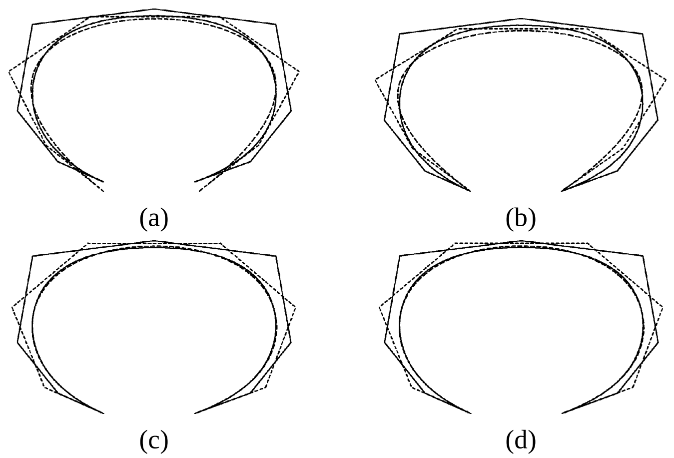

Fig. 2. Degree reduction can be improved by knot insertion. (a) M1 (error = 0.20). (b) M2 (error = 0.24). (c) M1 after all simple knots are inserted once (error = 0.03). (d) M2 after all simple knots are inserted once (error = 0.03).

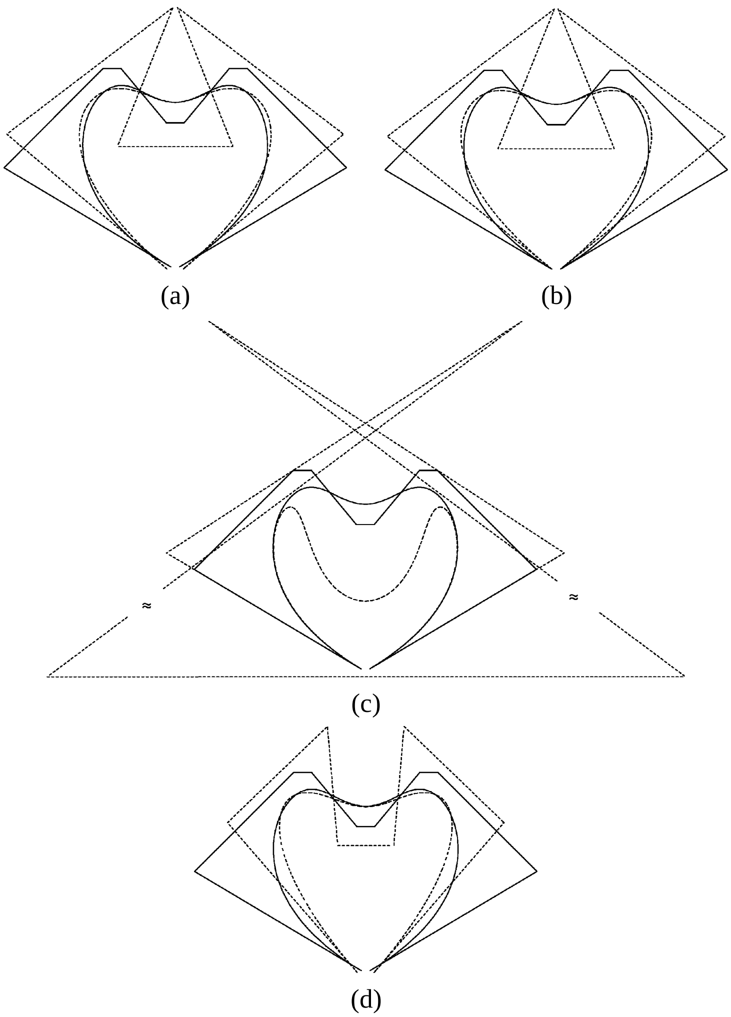

Fig. 3. Results of degree reduction by different algorithms. (a) M1 (error = 0.22). (b) M2 (error = 0.21). (c) P1 (error = 2.15). (d) W1 (error = 0.34).

Table 1
Numbers of control points used by different methods under given error tolerances

Given errors	P1	W1	M1	M2
1	10	8	8	8
10^{-1}	10	10	9	9
10^{-2}	13	16	11	11
10^{-3}	15	20	12	12
10^{-4}	20	22	15	15

the numbers of the control points of the resulting B-spline curves produced by the corresponding algorithms after knot refinement under the given error tolerances. In Table 1, the same notations are used for the corresponding algorithms for degree reduction after the knot refinement, which is coincident with that in Algorithm 2 in Section 4.2.

According to Fig. 3, our final curves are closer to the original curve than the results of the other two methods. Piegl and Tiller's algorithm consists of three steps (Piegl and Tiller, 1995a, 1995b):

(1) decompose the B-spline curve into Bézier pieces on the fly,
(2) degree reduce each Bézier piece, and
(3) remove unnecessary knots.

Both step (2) and step (3) will produce error that may increase the total error. Wolters, Wu and Farin's Algorithm has two steps. The first step uses the least squares method for degree reduction of each polynomial segment, and the second step applies the weighting scheme for merging multiple copies of the control points produced by the first step. In general, both these two steps may introduce error. Our algorithms also have two steps, but the error may be produced only at the first step using the constrained optimization methods to get the degenerate B-spline curves. At the second step, the degenerate B-spline curves can be represented by B-spline curves with a lower degree, i.e., the degenerate B-spline curves are degree reducible.

Acknowledgements

The authors acknowledge with thanks the comments and suggestions of the reviewers. The research was supported by the National Science Foundation of China (Project No 69902004).

References

Brunnett, G., Schreiber, T., Braun, J., 1996. The geometry of optimal degree reduction of Bézier curves. Comput. Aided Geom. Design 13, 773–788.

Brunnett, G., Schreiber, T., 1998. Optimal degree reduction of free form curves. Computing Supplement 13, 67–75.

Curry, H.B., Schoenberg, I.J., 1966. On PÓLYA frequency functions IV: The fundamental spline functions and their limits. J. Anal. Math. 17, 71–107.

de Boor, C., 1978. A Practical Guide to Splines. Springer-Verlag, New York.

Eck, M., 1993. Degree reduction of Bézier curves. Comput. Aided Geom. Design 10, 237–251.

Eck, M., 1995. Least squares degree reduction of Bézier curves. Computer-Aided Design 27, 845–851.

Farin, G., 1983. Algorithms for rational Bézier curves. Computer-Aided Design 15, 73–77.

Forrest, A.R., 1972. Interactive interpolation and approximation by Bézier polynomials. Comput. J. 15, 71–79.

Hu, S., Wang, G., Jin, T., 1996. Properties of two types of generalized Ball curves. Computer-Aided Design 28, 125–133.

Hu, S., Zheng, G., Sun, J., 1997. Approximate degree reduction of rectangular Bézier surfaces. Chinese J. Software Res. 4, 353–361.

Hu, S., Sun, J., Jin, T., Wang, G., 1998. Approximate degree reduction of Bézier curves. Tsinghua Sci. & Tec. 3, 977–1000.

Lachance, M.A., 1988. Chebyshev economization for parametric surfaces. Comput. Aided Geom. Design 5, 195–208.

Mühlbach, G., 1973. A recurrence formula for generalized divided differences and some applications. J. Approx. Theory 9, 165–172.

Piegl, L., Tiller, W., 1995a. Algorithm for degree reduction of B-spline curves. Computer-Aided Design 27, 101–110.

Piegl, L., Tiller, W., 1995b. The NURBS Book. Springer, Berlin.

Tiller, W., 1992. Knot-removal algorithms for NURBS curves and surfaces. Computer-Aided Design 24, 445–453.

Watkins, M.A., Worsey, A.J., 1988. Degree reduction of Bézier curves. Computer-Aided Design 20, 398–405.

Wolters, H.J., Wu, G., Farin, G., 1998. Degree reduction of B-spline curves. Computing Supplement 13, 235–241.