Journal of Zhejiang University SCIENCE A
ISSN 1009-3095 (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Optimal multi-degree reduction of Bézier curves with G^1-continuity*

LU Li-zheng†, WANG Guo-zhao

(*Institute of Computer Graphics and Image Processing, Department of Mathematics, Zhejiang University, Hangzhou 310027, China*)

†E-mail: lulz99@yahoo.com.cn

Received Dec. 29, 2005; revision accepted Feb. 20, 2006

Abstract: This paper presents a novel approach to consider optimal multi-degree reduction of Bézier curve with G^1-continuity. By minimizing the distances between corresponding control points of the two curves through degree raising, optimal approximation is achieved. In contrast to traditional methods, which typically consider the components of the curve separately, we use geometric information on the curve to generate the degree reduction. So positions and tangents are preserved at the two endpoints. For satisfying the solvability condition, we propose another improved algorithm based on regularization terms. Finally, numerical examples demonstrate the effectiveness of our algorithms.

Key words: Bézier curve, Optimal approximation, Degree reduction, Degree raising, G^1-continuity
doi:10.1631/jzus.2006.AS0174 **Document code:** A **CLC number:** TP391.72

INTRODUCTION

Degree reduction of polynomial curves and surfaces is a common process in Computer Aided Geometric Design (CAGD) and consists of approximating a polynomial by another one of a lower degree. This process is of great importance in geometric modelling, such as data exchange, data compression and data comparison. For example, degree reduction is needed when data are transferred from one modelling system to another, with these systems having different limitations on the maximum degree of polynomials. Furthermore, it can also be used to generate a piecewise continuous lower degree approximation to a given curve or surface so as to simplify some geometric or graphical algorithms like intersection calculation or rendering.

Previous work

There have been many methods developed for degree reduction. Forrest (1972) and Farin (1983) considered it as the inverse of degree elevation. Since degree reduction is an approximation problem in nature, methods in the classical approximation theory (Szegö, 1975) can be employed. In particular, the optimal approximations with respect to the L_∞ or L_2 metric are of interest. Watkins and Worsey (1988) used Chebyshev economization to produce the best L_∞-approximation of degree $n-1$ to a given degree n polynomial. This best approximation, however, does not interpolate the given curve at the endpoints. The endpoint constraints are frequently required in many applications and especially when degree reduction is combined with subdivision to generate piecewise continuous approximations.

Recently, C^k-constrained best degree reduction problem has been presented. Eck (1995) used constrained Legendre polynomials to do this by minimizing the L_2-norm between the two curves. Ahn (2003) also considered it in L_∞-norm. Ahn *et al.*(2004) proved that the best constrained degree reduction of a polynomial in L_2-norm equals the best weighted Euclidean approximation of Bézier coefficients. It is an open question whether an optimal approximation exists for norms other than L_p. And multi-degree reduction at one time avoiding stepwise computing

* Project supported by the National Natural Science Foundation of China (No. 60473130) and the National Basic Research Program (973) of China (No. G2004CB318000)

was investigated in (Ahn *et al.*, 2004; Chen and Wang, 2002).

Motivation and outline of the present paper

Traditional methods for degree reduction are applied to parameter curves simply by matching position and derivatives at the same parameter values, in general

$$f^{(i)}(t)=g^{(i)}(t),\ i=0,1,\dots,k.$$

Since parametric representations of curves are not unique, it will produce different forms for a given curve. For example, Fig.1 shows a quartic curve (solid):

$$f(x)=-19x^4/2+18x^3-15x^2+6x+1/2,$$

and its best cubic approximation with C^1-continuity (dotted) by the method in (Ahn *et al.*, 2004). However, if we reparameterize $f(x)$ by taking

$$x=\varphi(t)=3t/4+t^2/4,$$

the curve $f[\varphi(t)]$ remains unchanged but with a different approximation (dashed).

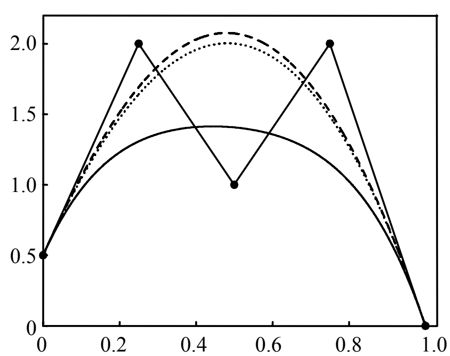


Fig.1 An example of C^1-constrained degree reduction

Our main motivation is to consider degree reduction with geometric continuity, which is independent on parameterizations. And such method often gives better approximation for degree reduction. Geometric continuity was introduced by de Boor *et al.*(1987) to curve interpolation, which is called Geometric Hermite Interpolation (GHI). They showed that if the curvature at one endpoint is not vanished, a planar curve can be interpolated by cubic spline with G^2-continuity and that the approximation order is 6. For more details about GHI, see the recent survey by Degen (2005).

In this paper, we consider multi-degree reduction with G^1-continuity by minimizing the sum of the Euclidean distances between corresponding control points of the two curves. Hu *et al.*(1998; 2001) used this method for curve approximation. According to G^1-continuity, it provides two more additional parameters. And we optimize these two parameters to obtain the optimal approximation. Endpoints information, such as positions and tangent directions, is also preserved.

PRELIMINARIES

Definitions and notations

In this paper, $\prod_n$ denotes the space of all real curves of degree n and $\|\cdot\|$ denotes the Euclidean vector norm $\|\boldsymbol{v}\|=\sqrt{<\boldsymbol{v},\boldsymbol{v}>}$.

A planar Bézier curve of degree n is given by the control points $\boldsymbol{p}_i\in\mathbb{R}^2$ in the form

$$\boldsymbol{P}(t)=\sum_{i=0}^{n} B_i^n(t)\boldsymbol{p}_i,\quad 0\le t\le 1, \tag{1}$$

where $B_i^n(t)$ are the Bernstein polynomials given by $B_i^n(t)=\binom{n}{i}(1-t)^{n-i}t^i$. Denoting $\boldsymbol{B}_n=(B_0^n(t),\dots,B_n^n(t))$ and $\boldsymbol{P}_n=(\boldsymbol{p}_0,\dots,\boldsymbol{p}_n)^{\mathrm{T}}$, we may express Eq.(1) in vector-vector form as

$$\boldsymbol{P}(t)=\boldsymbol{B}_n\boldsymbol{P}_n. \tag{2}$$

For raising the degree of Bézier curve by one without changing the shape of the curve, we can show that new points $\hat{\boldsymbol{p}}_i$ are obtained from the old ones by piecewise linear interpolation at the parameter values $i/(n+1)$ (Farin, 2001),

$$\hat{\boldsymbol{p}}_i=\frac{i}{n+1}\boldsymbol{p}_{i-1}+\left(1-\frac{i}{n+1}\right)\boldsymbol{p}_i,\quad i=0,1,\dots,n+1. \tag{3}$$

We can rewrite Eq.(3) as a linear system $\hat{\boldsymbol{P}}_n=\boldsymbol{T}_{n+1,n}\boldsymbol{P}_n$, where

$$\boldsymbol{T}_{n+1,n}=\frac{1}{n+1}\begin{pmatrix} n+1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & n & 0 & \cdots & 0 & 0 & 0 \\ 0 & 2 & n-1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & n-1 & 2 & 0 \\ 0 & 0 & 0 & \cdots & 0 & n & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & n+1 \end{pmatrix}$$

is the degree raising operator. Obviously, it has column full rank and can be viewed as an (n+2)×(n+1) matrix.

Problem formulation

Given a degree n Bézier curve $\boldsymbol{P}(t)=\boldsymbol{B}_n\boldsymbol{P}_n$, the problem of muti-degree reduction is to find control points $\boldsymbol{Q}_m$, which define the approximating curve $\boldsymbol{Q}(t)=\boldsymbol{B}_m\boldsymbol{Q}_m$ of lower degree m ($3\le m<n$), such that

(1) $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$ are G^1-continuous at t=0,1, i.e.,

$$\boldsymbol{P}(t)=\boldsymbol{Q}(t),\ \boldsymbol{P}'(t)/\|\boldsymbol{P}'(t)\|=\boldsymbol{Q}'(t)/\|\boldsymbol{Q}'(t)\|,\ t=0,1; \quad (4)$$

(2) $\boldsymbol{Q}(t)$ minimizes a suitable distance function $d(\boldsymbol{P}(t),\boldsymbol{Q}(t))$ for all possible curves in $\prod_m$ that satisfy the endpoint constraints Eq.(4).

In order to compare the control points of the two curves, we first raise $\boldsymbol{Q}(t)$ to degree n, i.e.,

$$\boldsymbol{Q}(t)=\boldsymbol{B}_n\hat{\boldsymbol{Q}}_m=\boldsymbol{B}_n\boldsymbol{T}_{n,m}\boldsymbol{Q}_m.$$

The degree raising operator $\boldsymbol{T}_{n,m}$ can be decomposed into a sequence of elementary degree-raising steps,

$$\boldsymbol{T}_{n,m}=\boldsymbol{T}_{n,n-1}\boldsymbol{T}_{n-1,n-2}\dots\boldsymbol{T}_{m+1,m}.$$

Then, we use the "discrete" coefficient norm, that is,

$$d(\boldsymbol{P}(t),\boldsymbol{Q}(t))=\|\boldsymbol{P}_n-\boldsymbol{T}_{n,m}\boldsymbol{Q}_m\|=\sqrt{\sum_{i=0}^{n}\|\boldsymbol{p}_i-\hat{\boldsymbol{q}}_i\|^2}. \quad (5)$$

The minimum of Eq.(5) will result in the least difference between the corresponding control points of the two curves.

Note that previous methods consider degree reduction of Bézier curves with C^k-continuity, which fixes the first and last (k+1) control points of the approximating curve. In contrast, G^1-constrained degree reduction is much looser and provides two more additional parameters. By using these parameters, we can optimize the approximation.

G^1-CONSTRAINED DEGREE REDUCTION

G^1 condition

Clearly, for G^0-cotinuity, the endpoints of $\boldsymbol{Q}(t)$ should coincide with the endpoints of $\boldsymbol{P}(t)$. And for G^1-cotinuity, the coincidence of the oriented tangents is additionally needed. Therefore, we can easily relate G^1 condition with the control points, more precisely,

$$\begin{aligned} \boldsymbol{q}_0&=\boldsymbol{p}_0, & \boldsymbol{q}_1&=\boldsymbol{p}_0+\frac{n}{m}\delta_0(\boldsymbol{p}_1-\boldsymbol{p}_0), \\ \boldsymbol{q}_m&=\boldsymbol{p}_n, & \boldsymbol{q}_{m-1}&=\boldsymbol{p}_n-\frac{n}{m}\delta_1(\boldsymbol{p}_n-\boldsymbol{p}_{n-1}). \end{aligned} \quad (6)$$

Note that geometric boundary conditions do not depend on the chosen parameterization, the control point $\boldsymbol{q}_1$ can thus move along the direction $\boldsymbol{p}_0\boldsymbol{p}_1$ without violating the G^1 condition (see Fig.2), and so is $\boldsymbol{q}_{m-1}$. Thus it provides two additional parameters to optimize the shape of the degree reduced curve. However, δ_0 and δ_1 should obey the following rule,

$$\delta_v>0,\ v=0,1. \quad (7)$$

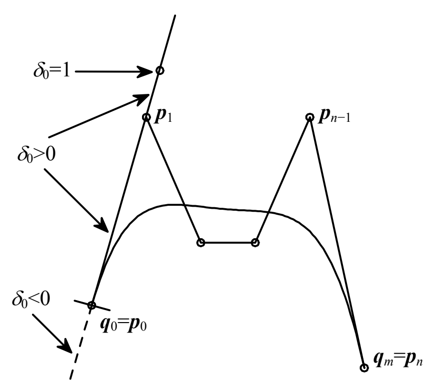


Fig.2 G^1 condition for degree reduction

When replacing G^1-cotinuity with C^1, $\boldsymbol{q}_1$ and $\boldsymbol{q}_{m-1}$ are uniquely determined by δ_v=1. Therefore, C^1 approximation is a special case of G^1 approximation. We can also imagine that G^1 approximation will lead to a smaller error between the original curve and the approximating one. We will propose the algorithms for G^1-constrained degree reduction.

Regular case

For a given degree n Bézier curve $\boldsymbol{P}(t)$, the degree reduction problem can be solved through two stages.

In the first stage, we construct a degree m Bézier curve $\boldsymbol{Q}(t)$ interpolating $\boldsymbol{P}(t)$ according to Eq.(6). More precisely, it can be written as

$$\begin{aligned}\boldsymbol{Q}(t) &= B_0^m(t)\boldsymbol{q}_0 + B_1^m(t)\boldsymbol{q}_1 + \sum_{i=2}^{m-2} B_i^m(t)\boldsymbol{q}_i \\ &\quad + B_{m-1}^m(t)\boldsymbol{q}_{m-1} + B_m^m(t)\boldsymbol{q}_m,\end{aligned} \tag{8}$$

where $\boldsymbol{q}_1$ and $\boldsymbol{q}_{m-1}$ contain the unknown variables δ_0 and δ_1, respectively. We then solve the interior control points $\boldsymbol{q}_i$ (i=2,…,m−2) by minimizing

$$E=d^2(\boldsymbol{P}(t),\boldsymbol{Q}(t))=\left\|\boldsymbol{P}_n - \boldsymbol{T}_{n,m}\boldsymbol{Q}_m\right\|^2 \tag{9}$$

Let $\boldsymbol{T}_{n,m}^c$ be the $(n+1)\times 4$ submatrix of the $(n+1)\times(m+1)$ matrix $\boldsymbol{T}_{n,m}$ obtained by extracting the first and last two columns and letting $\boldsymbol{T}_{n,m}^f$ be the $(n+1)\times(m-3)$ submatrix of $\boldsymbol{T}_{n,m}$ obtained by extracting columns from 3 to $(m-1)$. We then rewrite Eq.(9) as

$$E=\left\|\boldsymbol{P}_n - \boldsymbol{T}_{n,m}^c\boldsymbol{Q}_m^c - \boldsymbol{T}_{n,m}^f\boldsymbol{Q}_m^f\right\|^2, \tag{10}$$

where $\boldsymbol{Q}_m^c = (\boldsymbol{q}_0, \boldsymbol{q}_1, \boldsymbol{q}_{m-1}, \boldsymbol{q}_m)^{\mathrm{T}}$ and $\boldsymbol{Q}_m^f$ denotes the other control points of $\boldsymbol{Q}_m$.

Denoting $\boldsymbol{q}_i=(q_i^x,q_i^y)$, then for a minimum of Eq.(10) it is necessary that the derivatives of E with respect to q_i^x and q_i^y (i=2,…,m−2) are zero. And we write them in vector-matrix form as

$$\boldsymbol{0}= -\left(\boldsymbol{T}_{n,m}^f\right)^{\mathrm{T}}\left(\boldsymbol{P}_n - \boldsymbol{T}_{n,m}^c\boldsymbol{Q}_m^c\right) + \left(\boldsymbol{T}_{n,m}^f\right)^{\mathrm{T}}\boldsymbol{T}_{n,m}^f\boldsymbol{Q}_m^f. \tag{11}$$

Since $(\boldsymbol{T}_{n,m})^{\mathrm{T}}\boldsymbol{T}_{n,m}$ is a real symmetric positive definite matrix, so $\left(\boldsymbol{T}_{n,m}^f\right)^{\mathrm{T}}\boldsymbol{T}_{n,m}^f$ is invertible. Therefore, the least-squares error Eq.(10) is minimized by choosing

$$\begin{aligned}\boldsymbol{Q}_m^f &= \boldsymbol{Q}_m^f(\delta_0,\delta_1) \\ &= [(\boldsymbol{T}_{n,m}^f)^{\mathrm{T}}\boldsymbol{T}_{n,m}^f]^{-1}(\boldsymbol{T}_{n,m}^f)^{\mathrm{T}}(\boldsymbol{P}_n - \boldsymbol{T}_{n,m}^c\boldsymbol{Q}_m^c).\end{aligned} \tag{12}$$

Note that $\boldsymbol{q}_1$ and $\boldsymbol{q}_{m-1}$ contain the variables δ_0 and δ_1 respectively. Therefore, the interior control points $\boldsymbol{q}_i$ (i=2, …, m−2) are linear function of δ_v.

The second stage is to determine the two variables δ_v. Recall that our goal is to minimize the least-squares Eq.(10) which contains $\boldsymbol{Q}_m^c$ and $\boldsymbol{Q}_m^f$. By Eqs.(6) and (12), we can express the control points $\boldsymbol{q}_i$ (i=1, …, m−1) in linear functions of δ_v, rather than constants. After replacing them into Eq.(10), E forms a quadratic function with respect to δ_v, that is, $E=E(\delta_0, \delta_1)$. Since E is always nonnegative, it will come to a minimum at some values. Thus the degree reduction problem can be solved by two linear equations, i.e.,

$$\begin{cases}\dfrac{\partial E(\delta_0,\delta_1)}{\partial \delta_0} = 0, \\ \dfrac{\partial E(\delta_0,\delta_1)}{\partial \delta_1} = 0.\end{cases} \tag{13}$$

Unfortunately, because of the complexity of the expressions at the right side of Eq.(12), it is cumbersome to compute explicit formulas for δ_v. To solve the linear system Eq.(13), we refer to the solve procedure of MATLAB, an efficient and stable numerical procedure. After replacing all the variables in $\boldsymbol{Q}_m$ with the values solved above, we obtain the multiple degree reduced approximating curve $\boldsymbol{Q}(t)$ which preserves G^1-continuity at the endpoints. We summarize the degree reduction algorithm as follows.

Algorithm 1
Input: $\boldsymbol{p}_i$
Output: $\boldsymbol{q}_i$, ε
Step 1: Compute $\boldsymbol{q}_i$ (i=2, …, m−2) by Eq.(12);
Step 2: Obtain δ_v by solving the linear system Eq.(13);
Step 3: Compute $\boldsymbol{q}_i$ by Eqs.(6) and (12) and the approximating error ε by Eq.(9).

Remark 1 Despite the high effectiveness of Algorithm 1 (see Figs.3a, 5 and 6), it is impossible to guarantee the positivity of δ_v. In some singular cases (e.g. Fig.3b), G^1 condition is thus violated.

When solving Eq.(13) we assume that $\delta_v\in\mathbb{R}$. So

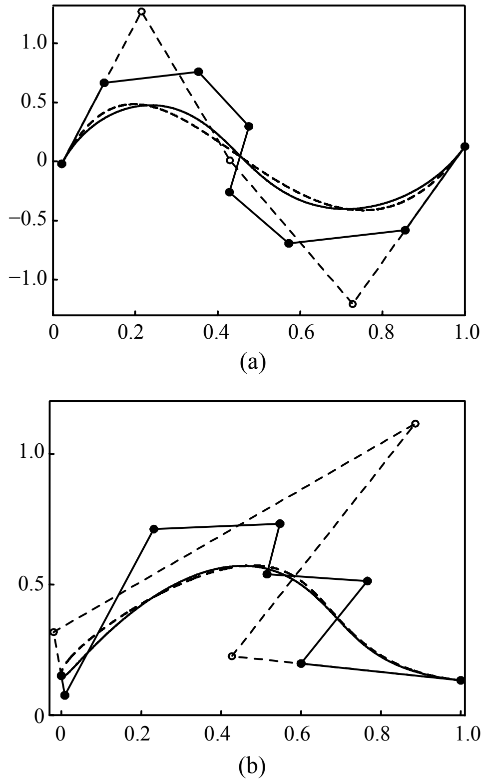


Fig.3 Degree reduction by Algorithm 1 (from degree 7 to degree 4): the original curves and polygons (solid) and the approximation curves and associated polygons (dashed). (a) G^1-continuous approximation; (b) Singular case: δ_0<0

the minimum may be reached on the negative semi-axis. For avoiding this, some other conditions are necessary to be imposed. We will provide an improved heuristic algorithm shown as follows.

Improvement

We investigate the positive solution existence of system Eq.(13) so as to meet the G^1 condition requirement for degree reduction. And if Algorithm 1 is valid, i.e., the condition Eq.(7) is satisfied, the following algorithm is not necessarily needed.

We recall that the free parameters are determined by minimizing the approximating error E, a metric which measures the difference between corresponding control points of the two curves. As is illustrated in Fig.3b, the points $\boldsymbol{p}_0$ and $\boldsymbol{p}_1$ are so close that $\|\boldsymbol{p}_1-\hat{\boldsymbol{q}}_1\|$ is negligible when compared with the others. Therefore, singular case arises potentially during the minimization of E.

Our intention is to penalize this phenomenon by introducing some regularization terms that help guide the approximation to a positive solution. But then the error may not be the global minimum.

To obtain a positive solution, we add regularization terms to E and modify it as follows,

$$E=\left\|\boldsymbol{P}_n-\boldsymbol{T}^c_{n,m}\boldsymbol{Q}^c_m-\boldsymbol{T}^f_{n,m}\boldsymbol{Q}^f_m\right\|^2+G(\|\Delta\boldsymbol{p}_0\|)(1-\delta_0)^2 \\ +G(\|\Delta\boldsymbol{p}_{n-1}\|)(1-\delta_1)^2, \tag{14}$$

where $\Delta\boldsymbol{p}_i=\boldsymbol{p}_{i+1}-\boldsymbol{p}_i$ and $G(x)=\exp(-x^2/2\sigma^2)$ is the standard Gaussian function with the parameter σ [also used in (Fleishman *et al.*, 2003)]. The first term represents the difference of the control points and the last two regulate δ_0 and δ_1, respectively.

Note that once δ_v=1 G^1-constrained degree reduction degenerates to the C^1-constrained case. As was described in (Ahn *et al.*, 2004; Hu *et al.*, 1998), C^1 approximation always exists. Since Gaussian function $G(x)$ converges to 1 when $x\to0$ and 0 when $x\to\infty$, it penalizes the small edge length and forces δ_v to converge to 1 while minimizing Eq.(14). Therefore, the condition Eq.(7) will be satisfied. Obviously, the approximating error becomes bigger than before.

The parameter σ in Gaussian function is adjustable and different values lead to different minima. Since we have no precise explicit representation of Eq.(14), we estimate this value by experimenting with various singular cases. And we find that it works well in practice and in all of the examples provided in this paper by setting $\sigma=e/4$ with e denoting the average edge length of the control polygon $\boldsymbol{P}_n$. In cases where it fails it can be solved by increasing or decreasing this value interactively.

We now modify Algorithm 1 and provide a heuristic algorithm as follows.

Algorithm 2
Input: $\boldsymbol{p}_i$
Output: $\boldsymbol{q}_i$, ε
Step 1: Set $\sigma=\tau e$ with initial values $\tau=\Delta\tau=1/20$;
Step 2: Use Algorithm 1 [just replace Eq.(10) with Eq.(14)] to obtain $\boldsymbol{q}_i$ and ε;
Step 3: If δ_v is invalid or ε decreases, set $\tau=\tau+\Delta\tau$ and go to Step 1. Otherwise, stop;
Step 4: Output $\boldsymbol{q}_i$ according to the minimal ε.

Fig.4 shows the improvement by Algorithm 2. The approximating curve obtained by Algorithm 2 is displayed in dots together with that one obtained by

Algorithm 1 displayed in dashes. It is clear that the approximation effect is reduced and that ε becomes bigger.

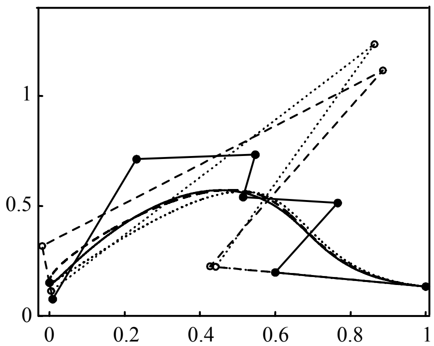


Fig.4 Improvement by Algorithm 2

Remark 2 Using regularization terms, a valid G^1 approximation can always be obtained and we increase σ stepwise so as to get the minimal error. In the extreme case, it will just lead to C^1 approximation. And for achieving better result, human interaction is recommended.

Remark 3 Combining Algorithm 1 with curves subdivision, we can also obtain a valid G^1-constrained degree reduction. Usually only one or two subdivisions are needed in practice. However, it will lead to piecewise G^1 curves (cf. Fig.6b), compared with Algorithm 2.

EXAMPLES

This section shows some numerical examples for our algorithms discussed in Section 3.

Example 1 [Example 2 in (Ahn *et al.*, 2004)] We consider a planar quintic Bézier curve with control points given by (0, 0), (0.2, 1), (0.4, 4), (0.6, 2), (0.8, 5), (1, 0). We want to find quartic and cubic Bézier curves to approximate it with G^1-continuity.

Fig.5 compares G^1-continuity degree reduction with C^1-continuity (Ahn *et al.*, 2004), with the quartic and cubic approximating curves shown in Figs.5a and 5b respectively. It is clearly seen that our method approximates the whole curve better, due to the G^1-continuity nature. Especially, cubic approximation by our method is nearly the same as quartic approximation.

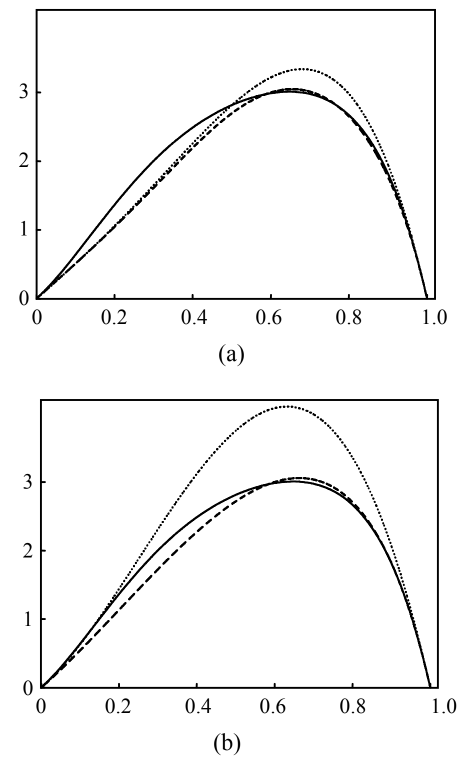


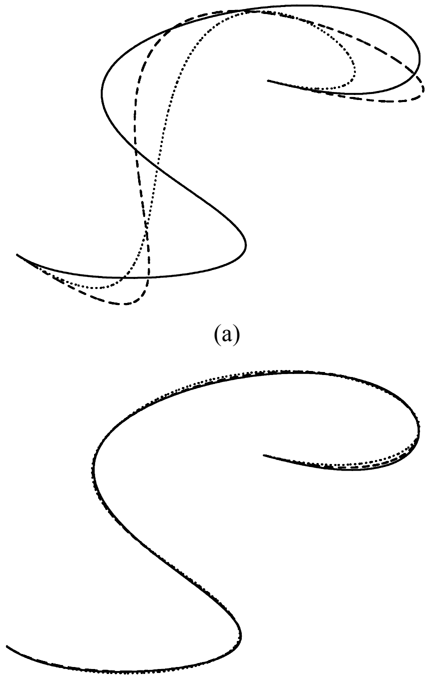
Fig.5 Degree reduction of a quintic Bézier curve (solid) by our method (Algorithm 1, dashed) and that of Ahn *et al.*(2004) (dotted). (a) Degree 5 to degree 4; (b) Degree 5 to degree 3

Example 2 We give a planar Bézier curve of degree fifteen which is a part of the out-lines of font "S". The control points are (0, 0), (1.5, −2.0), (4.5, −1.0), (9.0, 0.0), (4.5, 1.5), (2.5, 3.0), (0.0, 5.0), (−4.0, 8.5), (3.0, 9.5), (4.4, 10.5), (6.0, 12.0), (8.0, 11.0), (9.0, 10.0), (9.5, 5.0), (7.0, 6.0), (5.0, 7.0).

Fig.6 illustrates multi-degree reduction of a complex curve. In Fig.6a, our method represents more features of the original curve. To improve the effect, we subdivide the curve at t=1/2 and show the approximation result in Fig.6b. Better approximation can be obtained through curves subdivision. And we find that after one or two subdivisions, a given curve can be well approximated by piecewise curves.

CONCLUSION AND FUTURE WORK

In this paper we have introduced a new framework for multi-degree reduction of Bézier curve with G^1-continuity and obtain the optimal approximation by using the two additional parameters. So the posi-



(a)

(b)

Fig.6 Degree reduction of Bézier curve (solid) from degree 15 to degree 5: our method (Algorithm 1, dashed) and that of Ahn *et al.*(2004) (dotted). (a) Without subdivision; (b) With one subdivision

tion and tangent direction are preserved at each endpoint. Due to the geometric continuity, our methods approximate curves with a smaller error in contrast to traditional methods. For convenience, we present it in planar curves, but it can be applied directly to spatial curves.

From our results here it is easy to make a general conjecture. If we consider the problem of degree reduction with G^k-continuity (k>1), the approximating curve will be much closer to the original curve than that obtained by considering it with C^k-continuity.

However, to prove such a conjecture seems to be a very difficult task. See G^2-constrained degree reduction for example, we have to meet the curvature requirements at the endpoint, that is,

$$\kappa_0 = \frac{n-1}{n}\frac{\|\Delta \boldsymbol{p}_0 \times \Delta \boldsymbol{p}_1\|}{\|\Delta \boldsymbol{p}_0\|^3}.$$

When it comes to degree reduction, it turns out to be a nonlinear problem. And experiments show that not only the existence is hard to prove, but also the solvability and uniqueness. The only hope of success seems to tackle the problem with other methods. Since G^2-continuity is the most useful and important property for shape designing in CAGD, we hope to solve this problem in future work.

References

Ahn, Y.J., 2003. Degree reduction of Bézier curves with C^k-continuity using Jacobi polynomials. *Computer Aided Geometric Design*, **20**(7):423-434. [doi:10.1016/S0167-8396(03)00082-7]

Ahn, Y.J., Lee, B.G., Park, Y., Yoo, J., 2004. Constrained polynomial degree reduction in the L_2-norm equals best weighted Euclidean approximation of Bézier coefficients. *Computer Aided Geometric Design*, **21**(2):181-191. [doi:10.1016/j.cagd.2003.10.001]

Chen, G.D., Wang, G.J., 2002. Optimal multi-degree reduction of Bézier curves with constraints of endpoints continuity. *Computer Aided Geometric Design*, **19**(6):365-377. [doi:10.1016/S0167-8396(02)00093-6]

de Boor, C., Höllig, K., Sabin, M., 1987. High accuracy geometric Hermite interpolation. *Computer Aided Geometric Design*, **4**(4):269-278. [doi:10.1016/0167-8396(87)90002-1]

Degen, W.L.F., 2005. Geometric Hermite Interpolation—In memoriam Josef Hoschek. *Computer Aided Geometric Design*, **22**(7):573-592. [doi:10.1016/j.cagd.2005.06.008]

Eck, M., 1995. Least squares degree reduction of Bézier curves. *Computer-Aided Design*, **27**(11):845-851. [doi:10.1016/0010-4485(95)00008-9]

Farin, G., 1983. Algorithms for rational Bézier curves. *Computer-Aided Design*, **15**(2):73-79. [doi:10.1016/0010-4485(83)90171-9]

Farin, G., 2001. Curves and Surfaces for CAGD (5th Ed.). Morgan Kaufmann, San Francisco, p.81-93.

Fleishman, S., Drori, I., Cohen-Or, D., 2003. Bilateral mesh denoising. *ACM Trans. on Graphics*, **22**(3):950-953. [doi:10.1145/882262.882368]

Forrest, A.R., 1972. Interactive interpolation and approximation by Bézier curve. *The Computer Journal*, **15**(1):71-79.

Hu, S.M., Sun, J.G., Jin, T.G., Wang, G.Z., 1998. Approximate degree reduction of Bézier curves. *Tsinghua Science and Technology*, **3**(2):997-1000.

Hu, S.M., Tong, R.F., Ju, T., Sun, J.G., 2001. Approximate merging of a pair of Bézier curves. *Computer-Aided Design*, **33**(2):125-136. [doi:10.1016/S0010-4485(00)00115-9]

Szegö, G., 1975. Orthogonal Polynomials (4th Ed.). American Mathematical Society, Providence, RI, p.22-106.

Watkins, M.A., Worsey, A.J., 1988. Degree reduction of Bézier curves. *Computer-Aided Design*, **20**(7):398-405. [doi:10.1016/0010-4485(88)90216-3]