Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Optimal multi-degree reduction of C-Bézier surfaces with constraints*

Lian ZHOU[†‡1], Xin-hui LIN[2], Hong-yan ZHAO[3], Jun CHEN[4]

([1]*Department of Mathematics, Shanghai Maritime University, Shanghai 201306, China*)

([2]*Zhejiang Institute of Economics and Trade, Hangzhou 310018, China*)

([3]*College of Fundamental Studies, Shanghai University of Engineering Science, Shanghai 201620, China*)

([4]*Faculty of Science, Ningbo University of Technology, Ningbo 315211, China*)

[†]E-mail: lianzhou@shmtu.edu.cn

Received July 9, 2017; Revision accepted Sept. 13, 2017; Crosschecked Dec. 20, 2017

Abstract: We propose an optimal approach to solve the problem of multi-degree reduction of C-Bézier surfaces in the norm L_2 with prescribed constraints. The control points of the degree-reduced C-Bézier surfaces can be explicitly obtained by using a matrix operation that is based on the transfer matrix of the C-Bézier basis. With prescribed boundary constraints, this method can be applied to piecewise continuous patches or to a single patch with the combination of surface subdivision. The resulting piecewise approximating patches are globally G^1 continuous. Finally, numerical examples are presented to show the effectiveness of the method.

Key words: C-Bézier surfaces; Degree reduction; Boundary constraints
https://doi.org/10.1631/FITEE.1700458 **CLC number:** TP391.72

1 Introduction

Splines are important in engineering applications (Huang *et al.*, 2017; Nguyen-Thanh and Zhou, 2017; Nguyen-Thanh *et al.*, 2017; Tan *et al.*, 2017). As a new spline modeling tool, C-Bézier curves and surfaces have received much attention from geometric design researchers (Zhang, 1996; Fan *et al.*, 2002; Mainar and Peña, 2002). Compared with rational Bézier curves or rational B-splines, C-Bézier curves can exactly represent not only circular arc and polynomial curves of high order but also transcendental curves such as the helix and the cycloid (Chen and Wang, 2003; Yang and Wang, 2004). In particular, by using the C-Bézier model, repeated differentiation and integration can get rid of high-degree curves. Therefore, the geometric property and computation of C-Bézier curves have become a research focus in computer aided geometry design. Currently, most research focuses on the geometric properties and representation of C-Bézier curves (Shen and Wang, 2015; 2016).

Degree reduction is important and classical in computer aided design (CAD). For example, compressing data, finding intersection between two curves, and computing the roots of polynomials are all based on degree reduction (Liu *et al.*, 2009). There are fruitful studies on the degree reduction of Bézier curves and surfaces (Zheng and Wang, 2003; Zhou and Wang, 2009a; 2009b; Rababah and Mann, 2013; Zhou *et al.*, 2014; Gospodarczyk, 2015; Ait-Haddou and Bartoň, 2016). These methods have been extended to C-Bézier curves (Zhou, 2012; Qin *et al.*, 2013).

However, the degree reduction of C-Bézier surfaces has not been presented. It is necessary to

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 11401373, 61402281, and 11601322) and the Zhejiang Provincial Natural Science Foundation, China (No. LY16F020020)

ORCID: Lian ZHOU, http://orcid.org/0000-0001-7137-0162

develop an ideal algorithm for the degree reduction of C-Bézier surfaces. The ideal algorithm should have the following characteristics: (1) multi-degree reduction at one time; (2) certain constraints; (3) best precision. This paper is a summary of the results of research that aims to address these issues.

To achieve the above objective, we first give the explicit expression of the C-Bézier basis in space Γ_n= span$\{1, t, \ldots, t^{n-2}, \sin t, \cos t\}$. Then we present an algorithm for the multi-degree reduction of C-Bézier surfaces in the norm L_2 with prescribed constraints, based on the geometric joining condition. This algorithm aims to fulfill the need for multi-degree reduction of each patch of such a surface, which is piecewise continuous or is formed by combining some sub-surfaces, so that the resulting piecewise approximating surfaces are globally smooth. In this study, we assume that the continuous order is G^1, which usually appears in engineering applications. The control points of degree-reduced curves or surfaces are determined by some simple matrices. Since these matrices depend only on the degrees before and after degree reduction, we can calculate this series of matrices one time and then store them in a database before degree reduction. By doing this, our method is effective.

2 Preliminaries

2.1 Definition and properties of C-Bézier basis

Given an angle α, we define two initial functions

$$\begin{cases} u_{0,1}(t)=\sin(\alpha-t)/\sin\alpha, \\ u_{1,1}(t)=\sin t/\sin\alpha, \end{cases} \tag{1}$$

where $t\in[0, \alpha]$. The C-Bézier basis can be recursively defined by

$$\begin{cases} u_{0,n}(t)=1-\int_0^t \delta_{0,n-1}u_{0,n-1}(s)\mathrm{d}s, \\ u_{i,n}(t)=\int_0^t\left(\delta_{i-1,n-1}u_{i-1,n-1}(s)-\delta_{i,n-1}u_{i,n-1}(s)\right)\mathrm{d}s, \\ u_{n,n}(t)=\int_0^t \delta_{n-1,n-1}u_{n-1,n-1}(s)\mathrm{d}s, \end{cases} \tag{2}$$

where $\delta_{i,n-1}=\left(\int_0^\alpha u_{i,n-1}(s)\mathrm{d}s\right)^{-1},\ i=0,1,...,n-1.$

The C-Bézier basis has the following properties (Chen and Wang, 2003):

Property 1 (Properties at endpoints) $u_{0,n}(0)=1$, $u_{n,n}(\alpha)=1$, and $u_{i,n}^{(j)}(0)=u_{i,n}^{(l)}(\alpha)=0$, where j=0, 1, …, $i-1$, and l=0, 1, …, $n-i-1$.

Property 2 (Linear independence) $\{u_{i,n}(t)\}$ is a basis.

Property 3 (Normalization) $\sum_{i=0}^{n}u_{i,n}(t)=1.$

2.2 Transfer matrix of C-Bézier basis

We consider the approximation error in the norm L_2. So, it is necessary to compute the integration

$$g_{i,j}=\int_0^\alpha u_{i,n}(t)u_{j,m}(t)\mathrm{d}t. \tag{3}$$

However, it is difficult to obtain the above integration directly, according to the definition of the C-Bézier basis. We introduce a transfer matrix between the Γ_n basis and the C-Bézier basis to derive the integration $g_{i,j}$ explicitly. The transfer matrix is derived based on Properties 1 and 3.

Based on the linear independence of two bases, the relationship between them can be described as

$$\begin{cases} u_{0,n}(t)=a_{0,0}+\cdots a_{0,n-2}t^{n-2}+a_{0,n-1}\sin t+a_{0,n}\cos t, \\ u_{1,n}(t)=a_{1,0}+\cdots a_{1,n-2}t^{n-2}+a_{1,n-1}\sin t+a_{1,n}\cos t, \\ \quad\vdots \\ u_{n,n}(t)=a_{n,0}+\cdots a_{n,n-2}t^{n-2}+a_{n,n-1}\sin t+a_{n,n}\cos t. \end{cases} \tag{4}$$

Let

$$\boldsymbol{T}_n(t)=(1,t,\cdots,t^{n-2},\sin t,\cos t)^{\mathrm{T}}, \tag{5}$$

$$\boldsymbol{b}_{i,n}=(a_{i,0},a_{i,1},\cdots,a_{i,n})^{\mathrm{T}},\ \ i=0,1,\cdots,n. \tag{6}$$

Then Eq. (4) can be simplified as

$$u_{i,n}(t)=(\boldsymbol{b}_{i,n})^{\mathrm{T}}\cdot\boldsymbol{T}_n(t),\quad i=0,1,\cdots,n. \tag{7}$$

According to Property 1, we have

$$u_{i,n}^{(j)}(0)=(\boldsymbol{b}_{i,n})^{\mathrm{T}}\cdot\boldsymbol{T}_n^{(j)}(0)=0, \tag{8}$$

where $j=0,1,\cdots,i-1,\ 1\le i\le n.$

$$u_{i,n}^{(l)}(\alpha)=(\boldsymbol{b}_{i,n})^{\mathrm{T}}\cdot\boldsymbol{T}_n^{(l)}(\alpha)=0, \tag{9}$$

where

$$l=0,1,\cdots,n-i-1,\quad 0\le i\le n-1.$$

Let

$$\boldsymbol{A}(i)=\begin{cases}\left(\boldsymbol{T}_n^{(0)}(\alpha),\cdots,\boldsymbol{T}_n^{(n-1)}(\alpha)\right), & i=0,\\ \left(\boldsymbol{T}_n^{(0)}(0),\cdots,\boldsymbol{T}_n^{(i-1)}(0),\boldsymbol{T}_n^{(0)}(\alpha),\cdots\boldsymbol{T}_n^{(n-i-1)}(\alpha)\right), & 1\le i\le n-1,\\ \left(\boldsymbol{T}_n^{(0)}(0),\ \boldsymbol{T}_n^{(1)}(0),\cdots,\boldsymbol{T}_n^{(n-1)}(0)\right), & i=n.\end{cases} \tag{10}$$

Then Eqs. (8) and (9) can be rewritten as

$$(\boldsymbol{b}_{i,n})^{\mathrm{T}}\cdot\boldsymbol{A}(i)=\boldsymbol{0}. \tag{11}$$

The vectors $\boldsymbol{T}_n^{(k)}(0)$ and $\boldsymbol{T}_n^{(k)}(\alpha)$ can be calculated as follows:

When $0\le k\le n-2$, we have

$$\boldsymbol{T}_n^{(k)}(0)=\Big(\underbrace{0,0,\cdots,0}_{k},k!,\underbrace{0,0,\cdots,0}_{n-2-k},\ \sin\frac{k\pi}{2},\cos\frac{k\pi}{2}\Big), \tag{12}$$

$$\boldsymbol{T}_n^{(k)}(\alpha)=\Big(\underbrace{0,0,\cdots,0}_{k},\ \underbrace{P_k^k,P_{k+1}^k\alpha,P_{k+2}^k\alpha^2,\cdots,P_{n-2}^k\alpha^{n-2-k}}_{n-1-k},\ \sin\Big(\alpha+\frac{k\pi}{2}\Big),\cos\Big(\alpha+\frac{k\pi}{2}\Big)\Big), \tag{13}$$

where the arrangement number is

$$P_j^i=\frac{j!}{(j-i)!}. \tag{14}$$

When $k=n-1$ or n, we have

$$\boldsymbol{T}_n^{(k)}(0)=\Big(\underbrace{0,\cdots,0}_{n-1},\sin\frac{k\pi}{2},\cos\frac{k\pi}{2}\Big), \tag{15}$$

$$\boldsymbol{T}_n^{(k)}(\alpha)=\Big(\underbrace{0,\cdots,0}_{n-1},\ \sin\Big(\alpha+\frac{k\pi}{2}\Big),\ \cos\Big(\alpha+\frac{k\pi}{2}\Big)\Big). \tag{16}$$

Based on Properties 2 and 3, we obtain

$$\begin{cases}\sum_{i=0}^{n}a_{i,0}=1,\\ \sum_{i=0}^{n}a_{i,j}=0,\quad j=1,2,\cdots,n.\end{cases} \tag{17}$$

According to Eqs. (11) and (17), we derive

$$\left((\boldsymbol{b}_{0,n})^{\mathrm{T}},(\boldsymbol{b}_{1,n})^{\mathrm{T}},\cdots,(\boldsymbol{b}_{n,n})^{\mathrm{T}}\right)\cdot\begin{pmatrix}\boldsymbol{A}(0) & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{I}_{n+1}\\ \boldsymbol{0} & \boldsymbol{A}(1) & \cdots & \boldsymbol{0} & \boldsymbol{I}_{n+1}\\ \vdots & \vdots & & \vdots & \vdots\\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{A}(n) & \boldsymbol{I}_{n+1}\end{pmatrix}=\boldsymbol{L}, \tag{18}$$

where $\boldsymbol{L}=\Big(\underbrace{0,\cdots,0}_{n^2+n},1,\underbrace{0,\cdots,0}_{n}\Big)$. $\boldsymbol{I}_{n+1}$ denotes the identity matrix of degree $n+1$.

Now the transfer matrix of the C-Bézier basis can be described by the following lemma:

Lemma 1 The C-Bézier basis of degree n can be expressed in basis $\{1, t, \ldots, t^{n-2}, \sin t, \cos t\}$ as follows:

$$\boldsymbol{U}_n(t)=\left(\boldsymbol{T}_n(t)\right)^{\mathrm{T}}\cdot\boldsymbol{B}_n, \tag{19}$$

where

$$\boldsymbol{U}_n(t)=\left(u_{0,n}(t),u_{1,n}(t),\cdots,u_{n,n}(t)\right), \tag{20}$$

$$\boldsymbol{B}_n=(\boldsymbol{b}_{0,n},\boldsymbol{b}_{1,n},\cdots,\boldsymbol{b}_{n,n})_{(n+1)\times(n+1)}, \tag{21}$$

$$\left((\boldsymbol{b}_{0,n})^{\mathrm{T}},(\boldsymbol{b}_{1,n})^{\mathrm{T}},\cdots,(\boldsymbol{b}_{n,n})^{\mathrm{T}}\right)=\boldsymbol{L}\begin{pmatrix}\boldsymbol{A}(0) & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{I}_{n+1}\\ \boldsymbol{0} & \boldsymbol{A}(1) & \cdots & \boldsymbol{0} & \boldsymbol{I}_{n+1}\\ \vdots & \vdots & & \vdots & \vdots\\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{A}(n) & \boldsymbol{I}_{n+1}\end{pmatrix}^{-1}. \tag{22}$$

Remark 1 Although we need to calculate the inverse of a matrix of degree $(n+1)^2\times(n+1)^2$ in Eq. (22), it is a sparse matrix. So, we can obtain the transfer matrix easily and store it in a database before degree reduction.

According to Lemma 1, we compute the integration $g_{i,j}=\int_0^{\alpha}u_{i,n}(t)u_{j,m}(t)\mathrm{d}t$. Let

$$\boldsymbol{A}_{n,m} = \int_0^\alpha \left(\boldsymbol{T}_n(t)\right)^{\mathrm{T}} \boldsymbol{T}_m(t)\mathrm{d}t. \tag{23}$$

Then we have

$$g_{i,j} = \boldsymbol{b}_{i,n}^{\mathrm{T}} \boldsymbol{A}_{n,m} \boldsymbol{b}_{j,m}. \tag{24}$$

Now we denote

$$\boldsymbol{G}_{n,m} = (g_{i,j})_{(n+1)\times(m+1)}, \tag{25}$$

which has some good algebra properties.

Remark 2 $\boldsymbol{G}_{n,n}$ is invertible because it is a real symmetric positive definite matrix.

Remark 3 $\boldsymbol{G}_{n,m}$ depends only on the degrees of the input and output curves.

Remark 4 We can obtain the coefficients of matrix $\boldsymbol{G}_{n,m}$ iteratively by the following six integral formulas:

$$\int_0^\alpha t^k \mathrm{d}t = \frac{1}{k+1}\alpha^{k+1}, \tag{26}$$

$$\int_0^\alpha \sin t \cos t \mathrm{d}t = \frac{\sin^2\alpha}{2}, \tag{27}$$

$$\int_0^\alpha \sin^2 t \mathrm{d}t = \frac{\alpha - \sin\alpha\cos\alpha}{2}, \tag{28}$$

$$\int_0^\alpha \cos^2 t \mathrm{d}t = \frac{\alpha + \sin\alpha\cos\alpha}{2}, \tag{29}$$

$$\int_0^\alpha t^k \sin t \mathrm{d}t = -\alpha^k \cos\alpha + k\int_0^\alpha t^{k-1}\cos t \mathrm{d}t, \tag{30}$$

$$\int_0^\alpha t^k \cos t \mathrm{d}t = \alpha^k \sin\alpha - k\int_0^\alpha t^{k-1}\sin t \mathrm{d}t. \tag{31}$$

2.3 G¹-continuity splicing of C-Bézier surfaces

It is difficult to describe a complex surface using a single C-Bézier surface. A better way is to describe it using two or more patches. In engineering, G^1-continuity is a basic requirement between the patches with the same tangent plane along their common boundary.

Denote two C-Bézier surfaces with degrees $n\times m$ and $n\times l$, respectively, as

$$\boldsymbol{P}_1(s,t) = \sum_{i=0}^{n}\sum_{j=0}^{m} \boldsymbol{p}_{i,j}^1 u_{i,n}(s) u_{j,m}(t), \tag{32}$$

$$\boldsymbol{P}_2(s,t) = \sum_{i=0}^{n}\sum_{j=0}^{l} \boldsymbol{p}_{i,j}^2 u_{i,n}(s) u_{j,l}(t), \tag{33}$$

where $0\le\alpha\le\pi$, $0\le s$, and $t\le\alpha$.

Lemma 2 A necessary condition for the G^1-continuity condition of two C-Bézier surfaces in the t direction is

$$\boldsymbol{p}_{0,j}^2 = \boldsymbol{p}_{n,j}^1, \tag{34}$$

$$(\boldsymbol{p}_{1,j}^2 - \boldsymbol{p}_{0,j}^2) = \chi(\boldsymbol{p}_{n,j}^1 - \boldsymbol{p}_{n-1,j}^1), \tag{35}$$

where $\chi>0$ and $j=0, 1, \ldots, n$.

Remark 5 Lemma 2 indicates that C-Bézier surfaces have the same G^1 joining condition as Bézier surfaces.

3 Degree reduction of C-Bézier surfaces

Given control points $\{\boldsymbol{p}_{i,j}\}$ and shape parameter α, a degree $n\times m$ tensor product C-Bézier surface can be expressed as

$$\boldsymbol{P}(s,t) = \sum_{i=0}^{n}\sum_{j=0}^{m} \boldsymbol{p}_{i,j} u_{i,n}(s) u_{j,m}(t), \tag{36}$$

where

$$0\le\alpha\le\pi,\ 0\le s,t\le\alpha.$$

The unconstrained multi-degree reduction of surface $\boldsymbol{P}(s, t)$ is to find a degree $n_1\times m_1$ ($n_1<n$, $m_1<m$) C-Bézier surface

$$\boldsymbol{Q}(s,t) = \sum_{i=0}^{n_1}\sum_{j=0}^{m_1} \boldsymbol{q}_{i,j} u_{i,n_1}(s) u_{j,m_1}(t), \tag{37}$$

with the same shape parameter α, such that the distance ε between these two surfaces in the norm L_2 reaches a minimum, where

$$\varepsilon = \int_0^\alpha\int_0^\alpha \left\|\boldsymbol{P}(s,t) - \boldsymbol{Q}(s,t)\right\|^2 \mathrm{d}s\mathrm{d}t. \tag{38}$$

3.1 Constrained control points determination

The unconstrained degree reduction of C-Bézier surfaces has one limitation. If we start with several smoothly joined C-Bezier patches and apply degree reduction to each patch, gaps might arise along the boundary. Furthermore, we cannot derive continuous approximating patches from the original surface within a user-prescribed tolerance since this method cannot be combined with surface subdivision. So, we need some constraints to smooth the degree-reduced surfaces. In Fig. 1, the control points of the four boundaries of degree-reduced surfaces can be determined by the user's design.

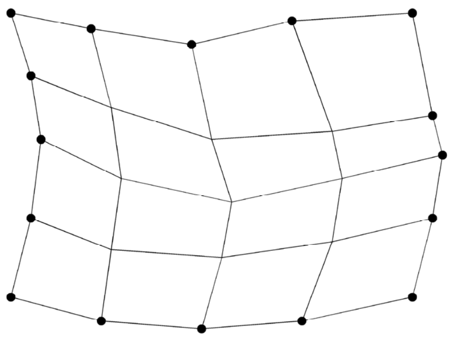

Fig. 1 Control grid of one C-Bézier surface with boundary constraints

In this study, we assume the G^1-continuity joining condition of two surfaces. Based on Lemma 2, we can determine the common boundary of the degree-reduced surfaces and its derivation of the boundary. Fig. 2 is a control net of a degree-reduced surface, which is combined with two C-Bézier surfaces that have boundary G^1-continuity. The thick line denotes the control net of the boundary. The thick dots denote the constrained control points, which are determined by the G^1-continuity condition and the assigned boundaries. Here, we denote these two surfaces as left degree $n\times m$ surface and right degree $n\times l$ surface, respectively.

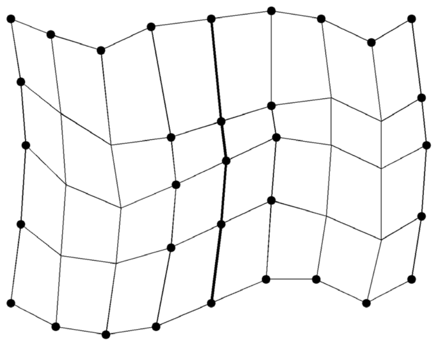

Fig. 2 Control grid of two G^1 joined C-Bézier surfaces with boundary constraints

3.2 Unconstrained control points determination

In Fig. 2, the thick dots along the boundaries have been determined based upon the prescribed constraints. First, we determine the unconstrained control points of the degree-reduced left surface. Next, we deal with the right surface in a similar way. Then we need to derive the internal control points $\{\boldsymbol{q}_{i,j}\}$ (i=1, 2, …, n_1-1; j=1, 2, …, m_1-2) such that the approximation error is minimized. After subtracting the determined term from the original surface $\boldsymbol{P}(s, t)$, we obtain a new surface

$$\begin{aligned}\boldsymbol{R}(s,t) &= \boldsymbol{U}_n(s)\boldsymbol{P}\left(\boldsymbol{U}_m(t)\right)^{\mathrm{T}} - \boldsymbol{U}_{n_1}(s)\bar{\boldsymbol{Q}}\left(\boldsymbol{U}_{m_1}(t)\right)^{\mathrm{T}} \\ &= \boldsymbol{U}_n(s)\boldsymbol{P}\left(\boldsymbol{U}_m(t)\right)^{\mathrm{T}} - \left(\boldsymbol{T}_{n_1}(s)\right)^{\mathrm{T}}\boldsymbol{B}_{n_1}\bar{\boldsymbol{Q}}(\boldsymbol{B}_{m_1})^{\mathrm{T}}\boldsymbol{T}_{m_1}(t) \\ &= \boldsymbol{U}_n(s)\boldsymbol{R}\left(\boldsymbol{U}_m(t)\right)^{\mathrm{T}},\end{aligned} \tag{39}$$

where

$$\boldsymbol{P} = (\boldsymbol{p}_{i,j})_{(n+1)\times(m+1)},$$

$$\bar{\boldsymbol{Q}} = \begin{pmatrix} \boldsymbol{q}_{0,0} & \boldsymbol{q}_{0,1} & \cdots & \boldsymbol{q}_{0,m_1-1} & \boldsymbol{q}_{0,m_1} \\ \boldsymbol{q}_{1,0} & \boldsymbol{0} & \cdots & \boldsymbol{q}_{1,m_1-1} & \boldsymbol{q}_{1,m_1} \\ \vdots & \vdots & & \vdots & \vdots \\ \boldsymbol{q}_{n_1-1,0} & \boldsymbol{0} & \cdots & \boldsymbol{q}_{n_1-1,m_1-1} & \boldsymbol{q}_{n_1-1,m_1} \\ \boldsymbol{q}_{n_1,0} & \boldsymbol{q}_{n_1,1} & \cdots & \boldsymbol{q}_{n_1,m_1-1} & \boldsymbol{q}_{n_1,m_1} \end{pmatrix}, \tag{40}$$

$$\begin{aligned}\boldsymbol{R} &= (\boldsymbol{r}_{i,j})_{(n+1)\times(m+1)} \\ &= \boldsymbol{P} - (\boldsymbol{B}_n)^{-1}\boldsymbol{E}_1\boldsymbol{B}_{n_1}\bar{\boldsymbol{Q}}(\boldsymbol{B}_{m_1})^{\mathrm{T}}(\boldsymbol{E}_2)^{\mathrm{T}}(\boldsymbol{B}_{m_1}^{\mathrm{T}})^{-1},\end{aligned} \tag{41}$$

where the degree-raised matrices

$$\boldsymbol{E}_1 = \begin{pmatrix} \boldsymbol{I}_{n_1-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_2 \end{pmatrix}_{(n+1)\times(n_1+1)} \tag{42}$$

and

$$\boldsymbol{E}_2 = \begin{pmatrix} \boldsymbol{I}_{m_1-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_2 \end{pmatrix}_{(m+1)\times(m_1+1)} \tag{43}$$

satisfy

$$\left(\boldsymbol{T}_{n_1}(s)\right)^{\mathrm{T}} = \left(\boldsymbol{T}_n(s)\right)^{\mathrm{T}}\boldsymbol{E}_1 \tag{44}$$

and

$$\left(\boldsymbol{T}_{m_1}(t)\right)^{\mathrm{T}}=\left(\boldsymbol{T}_m(t)\right)^{\mathrm{T}}\boldsymbol{E}_2, \tag{45}$$

respectively. According to Eq. (41), the approximation error, expressed as Eq. (38), can be rewritten as

$$\begin{aligned}\varepsilon=\int_0^\alpha\int_0^\alpha\Bigg\|&\sum_{i=0}^{n}\sum_{j=0}^{m}\boldsymbol{r}_{i,j}u_{i,n}(s)u_{j,m}(t)\\&-\sum_{i=1}^{n_1-1}\sum_{j=1}^{m_1-2}\boldsymbol{q}_{i,j}u_{i,n_1}(s)u_{j,m_1}(t)\Bigg\|^2\mathrm{d}s\mathrm{d}t.\end{aligned} \tag{46}$$

Taking the partial derivative of Eq. (46) with respect to $\boldsymbol{q}_{i,j}$ (i=1, 2, …, n_1−1; j=1, 2, …, m_1−2) and setting the derivative equal to zero, we obtain

$$\begin{aligned}\int_0^\alpha\int_0^\alpha u_{i,n_1}(s)u_{j,m_1}(t)\Bigg(&\sum_{i=0}^{n}\sum_{j=0}^{m}\boldsymbol{r}_{i,j}u_{i,n}(s)u_{j,m}(t)\\&-\sum_{i=1}^{n_1-1}\sum_{j=1}^{m_1-2}\boldsymbol{q}_{i,j}u_{i,n_1}(s)u_{j,m_1}(t)\Bigg)\mathrm{d}s\mathrm{d}t=0.\end{aligned} \tag{47}$$

Denote index sets

$$\Lambda^n=\{0,1,\cdots,n\},\ \Lambda^m=\{0,1,\cdots,m\}, \tag{48}$$

$$\Lambda^{n_1}=\{1,2,\cdots,n_1-1\},\ \Lambda^{m_1}=\{1,2,\cdots,m_1-2\}, \tag{49}$$

and let

$$\boldsymbol{A}=\boldsymbol{G}(\Lambda^{n_1};\Lambda^n),\ \boldsymbol{B}=\boldsymbol{G}(\Lambda^m;\Lambda^{m_1}), \tag{50}$$

$$\boldsymbol{C}=\boldsymbol{G}(\Lambda^{n_1};\Lambda^{n_1}),\ \boldsymbol{D}=\boldsymbol{G}(\Lambda^{m_1};\Lambda^{m_1}). \tag{51}$$

Then Eq. (47) can be rewritten as follows:

$$\boldsymbol{ARB}=\boldsymbol{C}\tilde{\boldsymbol{Q}}\boldsymbol{D}, \tag{52}$$

where

$$\begin{aligned}&\tilde{\boldsymbol{Q}}=(\boldsymbol{q}_{i,j})_{(n_1-1)\times(m_1-2)},\\&i=1,2,\cdots,n_1-1,\ \ j=1,2,\cdots,m_1-2.\end{aligned} \tag{53}$$

Obviously, $\boldsymbol{C}$ and $\boldsymbol{D}$ are both positive definite matrices sand thus are invertible. Hence, the distance ε is minimized by choosing

$$\tilde{\boldsymbol{Q}}=\boldsymbol{C}^{-1}\boldsymbol{ARBD}^{-1}. \tag{54}$$

Substituting Eq. (41) into Eq. (54), we obtain the control points of the degree-reduced left surface:

$$\begin{aligned}\tilde{\boldsymbol{Q}}=\boldsymbol{C}^{-1}\boldsymbol{A}\Big(&\boldsymbol{P}-(\boldsymbol{B}_n)^{-1}\boldsymbol{E}_1\boldsymbol{B}_{n_1}\bar{\boldsymbol{Q}}(\boldsymbol{B}_{m_1})^{\mathrm{T}}\\&\cdot(\boldsymbol{E}_2)^{\mathrm{T}}(\boldsymbol{B}_m^{\mathrm{T}})^{-1}\Big)\boldsymbol{BD}^{-1}.\end{aligned} \tag{55}$$

The right surface shown in Fig. 2 can be handled in the same way. Then, we derive the two degree-reduced surfaces with G^1-continuity along the common boundary.

3.3 Examples

For one single surface, we can first determine the control points of four boundaries as shown in Fig. 1, and then derive the internal control points. In Example 1, we perform the degree reduction of four boundaries using the degree reduction of curves method described by Zhou (2012). Then let the control points $\boldsymbol{q}_{i,m_1-1}$ (i=1, 2, …, n_1−1) in Eq. (47) be $\mathbf{0}$. Thus, according to Eq. (55), we can obtain the degree-reduced surfaces. To show the effectiveness of the approximation, we introduce error surface $\|\boldsymbol{P}(s,t)-\boldsymbol{Q}(s,t)\|$, which is defined by the l_2-norm.

Example 1 The degree of a C-Bézier surface is deduced from 5×5 to 3×3 with assigned boundaries. The approximation error is 0.1315 in Fig. 3.

Example 2 The degree of two C-Bézier surfaces is deduced from 6×6 to 3×3 with boundary G^1-continuity. The boundaries of the two degree-reduced surfaces are obtained using the method described by Zhou (2012) with G^1-continuity in Fig. 4.

4 Conclusions

In this study, we present an optimal method for the degree reduction of C-Bézier surfaces. We calculate the transfer matrix of the C-Bézier basis in space $\{1,\ t,\ \dots,\ t^{n-2},\ \sin t,\ \cos t\}$. The constrained multi-degree reduction of C-Bézier surfaces is explicitly derived in matrix form (Eq. (55)). The transfer matrix $\boldsymbol{B}_n$ in Eq. (22) is large but sparse. In particular, it depends only on the degrees of both the original and the degree-reduced surface. That is, we can calculate this series of matrices beforehand and store them in a database for use at any time. So, the calculation of degree reduction is simple and quick. Numerical examples show that our method is effective.

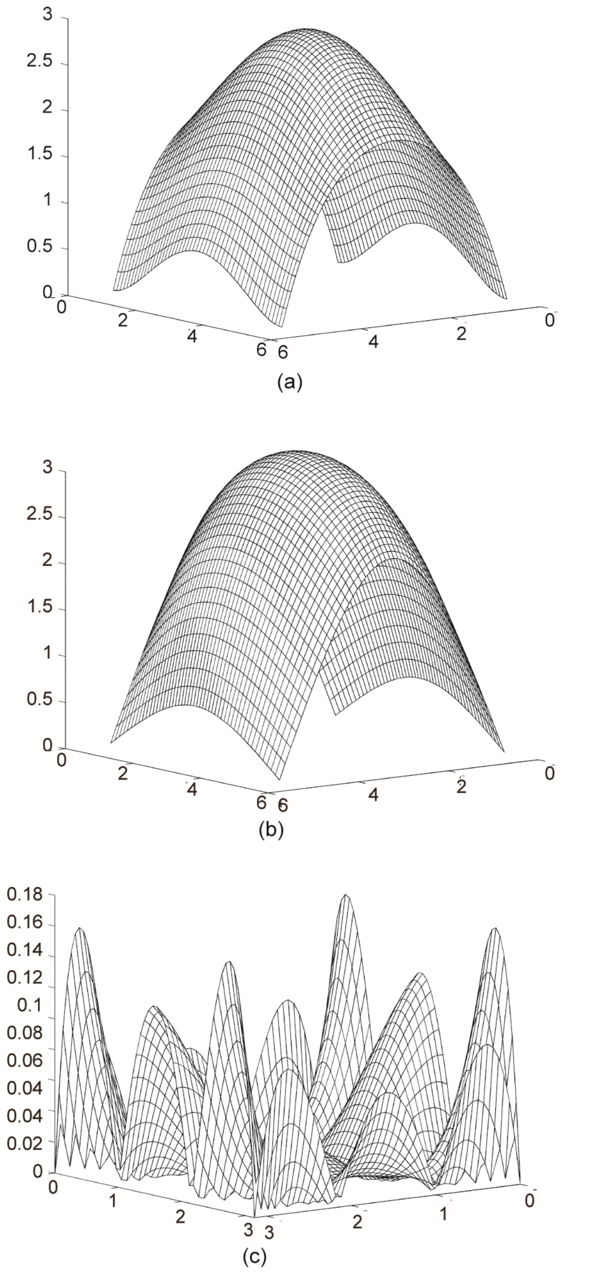

Fig. 3 The original surface of degree 5×5 (a), 3×3 degree-reduced surface (b), and error surface (c)

We consider the G^1-continuity along the common boundary of two patches. The resulting degree-reduced surfaces can keep G^1 continuous globally. In fact, our method can be generalized to cases with higher continuity such as C^1, G^1, C^2, G^2—we just need to modify the boundary control points in Eq. (40) to deal with higher continuity.

Degree reduction is often used in CAD systems

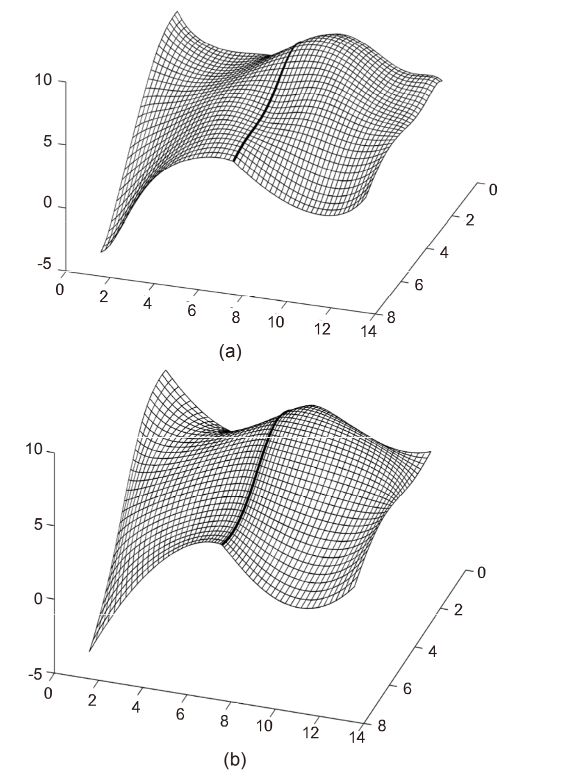

Fig. 4 Two G^1 joined degree 6×6 C-Bézier surfaces (a) and two G^1 joined 3×3 degree-reduced surfaces with boundary constraints (b)

with degree limitation. Our method is useful for transferring and compressing product model data.

We deal with tensor product C-Bézier surfaces in this study. We will focus on the degree reduction of triangular surfaces in future work.

References

Ait-Haddou, R., Bartoň, M., 2016. Constrained multi-degree reduction with respect to Jacobi norms. *Comput. Aided Geom. Des.*, **42**:23-30. https://doi.org/10.1016/j.cagd.2015.12.003

Chen, Q.Y., Wang, G.Z., 2003. A class of Bézier-like curves. *Comput. Aided Geom. Des.*, **20**(1):29-39. https://doi.org/10.1016/S0167-8396(03)00003-7

Fan, J.H., Wu, Y.J., Lin, X., 2002. Subdivision algorithm and G^1 condition for C-Bézier curves. *J. Comput. Aided Des. Comput. Graph.*, **14**(5):421-424 https://doi.org/10.3321/j.issn:1003-9775.2002.05.009

Gospodarczyk, P., 2015. Degree reduction of Bézier curves with restricted control points area. *Comput. Aided Des.*, **62**:143-151. https://doi.org/10.1016/j.cad.2014.11.009

Huang, J.Z., Nguyen-Thanh, N., Zhou, K., 2017. Extended isogeometric analysis based on Bézier extraction for the

buckling analysis of Mindlin-Reissner plates. *Acta Mech*., **228**(9):3077-3093.
https://doi.org/10.1007/s00707-017-1861-0

Liu, L.G., Zhang, L., Lin, B.B., *et al*., 2009. Fast approach for computing roots of polynomials using cubic clipping. *Comput*. *Aided Geom*. *Des*., **26**(5):547-559.
https://doi.org/10.1016/j.cagd.2009.02.003

Mainar, E., Peña, J.M., 2002. A basis of C-Bézier splines with optimal properties. *Comput*. *Aided Geom*. *Des*., **19**(4): 291-295.
https://doi.org/10.1016/S0167-8396(02)00089-4

Nguyen-Thanh, N., Zhou, K., 2017. Extended isogeometric analysis based on PHT-splines for crack propagation near inclusions. *Int*. *J*. *Numer*. *Methods Eng*., **112**(12):1777-1800. https://doi.org/10.1002/nme.5581

Nguyen-Thanh, N., Zhou, K., Zhuang, X., *et al*., 2017. Isogeometric analysis of large-deformation thin shells using RHT-splines for multiple-patch coupling. *Comput*. *Methods Appl*. *Mech*. *Eng*., **316**:1157-1178.
https://doi.org/10.1016/j.cma.2016.12.002

Qin, X.Q., Wang, W.W., Hu, G., 2013. Degree reduction of C-Bézier curve based on genetic algorithm. *Comput*. *Eng*. *Appl*., **49**(5):174-178.
https://doi.org/10.3778/j.issn.1002-8331.1107-0346

Rababah, A., Mann, S., 2013. Linear methods for G^1, G^2, and G^3-multi-degree reduction of Bézier curves. *Comput*. *Aided Des*., **45**(2):405-414.
https://doi.org/10.1016/j.cad.2012.10.023

Shen, W.Q., Wang, G.Z., 2015. Geometric shapes of C-Bézier curves. *Comput*. *Aided Des*., **58**:242-247.
https://doi.org/10.1016/j.cad.2014.08.007

Shen, W.Q., Wang, G.Z., 2016. Degree elevation from Bézier curve to C-Bézier curve with corner cutting form. *Appl*. *Math*. *A J*. *Chin*. *Univ*., **31**(2):165-176.
https://doi.org/10.1007/s11766-016-3369-0

Tan, P.F., Nguyen-Thanh, N., Zhou, K., 2017. Extended isogeometric analysis based on Bézier extraction for an FGM plate by using the two-variable refined plate theory. *Theor*. *Appl*. *Fract*. *Mech*., **89**:127-138.
https://doi.org/10.1016/j.tafmec.2017.02.002

Yang, Q.M., Wang, G.Z., 2004. Inflection points and singularities on C-curves. *Comput*. *Aided Geom*. *Des*., **21**(2):207-213. https://doi.org/10.1016/j.cagd.2003.11.002

Zhang, J.W., 1996. C-Curves: an extension of cubic curves. *Comput*. *Aided Geom*. *Des*., **13**(3):199-217.
https://doi.org/10.1016/0167-8396(95)00022-4

Zheng, J.M., Wang, G.Z., 2003. Perturbing Bézier coefficients for best constrained degree reduction in the L_2-norm. *Graph Models*, **65**(6):351-368.
https://doi.org/10.1016/j.gmod.2003.07.001

Zhou, L., 2012. Algorithm for explicit multi-degree reduction of C-Bézier curves. *J*. *Shanghai Marit*. *Univ*., **33**(4):86-90.
https://doi.org/10.3969/j.issn.1672-9498.2012.04.017

Zhou, L., Wang, G.J., 2009a. Optimal constrained multi-degree reduction of Bézier curves with explicit expressions based on divide and conquer. *J*. *Zhejiang Univ*.-*Sci*. *A*, **10**(4):577-582.
https://doi.org/10.1631/jzus.A0820290

Zhou, L., Wang, G.J., 2009b. Constrained multi-degree reduction of Bézier surfaces using Jacobi polynomials. *Comput*. *Aided Geom*. *Des*., **26**(3):259-270.
https://doi.org/10.1016/j.cagd.2008.10.003

Zhou, L., Wei, Y.W., Yao, Y.F., 2014. Optimal multi-degree reduction of Bézier curves with geometric constraints. *Comput*. *Aided Des*., **49**:18-27.
https://doi.org/10.1016/j.cad.2013.12.004