ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.17, No.10, October 2006, pp.2211−2220
DOI: 10.1360/jos172211

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

任意三角形网格的基于二元四次箱样条分片 C^1 曲面*

张永春，达飞鹏+，宋文忠

(东南大学 自动化所,江苏 南京 210096)

Piecewise C^1 Surfaces Based on Bivariate Quartic Box-Splines for Arbitrary Triangular Meshes

ZHANG Yong-Chun, DA Fei-Peng+, SONG Wen-Zhong

(Institute of Automation, Southeast University, Nanjing 210096, China)

+ Corresponding author: Phn: +86-25-83795931, E-mail: dafp@seu.edu.cn

Zhang YC, Da FP, Song WZ. Piecewise C^1 surfaces based on bivariate quartic box-splines for arbitrary triangular meshes. *Journal of Software*, 2006,17(10):2211−2220. http://www.jos.org.cn/1000-9825/17/2211.htm

Abstract: For arbitrary triangular control meshes, a surface algorithm based-on bivariate box-spline is developed. Bivariate 3-direction is a triangulation with the least directions. Box spline built on it is widely applied in CAGD. Its standard surface algorithm is only for normal control mesh in which every point has valence 6. Starting with bivariate 3-directional quartic box-splines, the paper proposes an algorithm for arbitrary triangular control meshes. The analysis of its properties especially continuity are presented in detail. The constructed surfaces by the algorithm are convex preserving, and they are piecewise C^1. The algorithm can be easily applied for global or local interpolation, which is indispensable in 3D surface reconstruction from scattered points.
Key words: bivariate 3-directional quartic box-splines; box-splines surface; piecewise C^1; arbitrary triangular meshes

摘 要: 提出一种以任意三角剖分为控制网格的二元箱样条曲面算法.二元三方向剖分是方向最少的三角剖分,建立在其上的二元三向四次箱样条在 CAGD 等领域有着广泛的应用.其规范的箱样条曲面计算仅适用于控制点的价数均为 6 的网格.从规范的算法出发,提出了一种任意价数控制网格的曲面计算算法,并对算法的连续性等进行了详细的分析.生成的曲面具有保凸性,且是分片 C^1 连续的.该算法可进行 3D 离散点全局或局部插值,并可应用于 3D 曲面重构等领域.
关键词: 二元三向四次箱样条;箱样条曲面;分片 C^1;任意三角形网格
中图法分类号: TP391 文献标识码: A

1 Introduction

Triangular splines were first considered by Sabin in 1977, and later they were found to be 'box-splines'[1]. With

* Supported by the Natural Science Foundation of Jiangsu Province of China under Grant No.BK2003405 (江苏省自然科学基金); the R&D Foundation for Excellent Young Teachers of Southeast University (东南大学优秀青年教师科研基金)

Received 2004-08-18; Accepted 2005-07-08

the requirements of practical problems, researches and applications of multi-variate box-splines were paid intensive attention to, and developed rapidly. 1993 saw the monograph Box Splines by de Boor, *et al*[2].

Bivariate 3-directional subdivision is a triangular subdivision with least directions. It is widely applied in surface modeling, approximation etc. for its beautiful properties. Some scholars have done abundant researches[3−7] on it.

Bivariate 3-directional surfaces take triangular meshes as control meshes. The standard surface algorithm is confined to the restriction that every de Boor point in meshes must have valence (all edges with the point as one of the end points) of 6. When interpolating 3D scattered point-set, its triangulation is built firstly. Then the control meshes are computed. Because the topological structure of the mesh is the same as the triangulation that is arbitrary, the standard algorithm cannot be used directly. It is necessary to transform the standard algorithm for arbitrary triangular meshes.

Loop subdivision[8] can be applied over triangular meshes, but it appears somewhat difficult for interpolation. Smooth subdivision interpolations to scattered data[9−11] are excellent for triangular meshes. Subdivision algorithms lead to limit surfaces. However, in engineering practice precise coordinates of points are necessary.

Kolb and Seidel[12], Li[13] discussed functional surfaces interpolations. The former is based on Nielson's minimum norm network without convex hull property, and the later is a convex preserving interpolation including a kind of complicated nonlinear optimization process.

Ueshiba and Roth[14], Loop[15] presented respectively creations of G^1 surfaces with Bézier patches over closed meshes of triangular meshes. The convex hull property, very important one, is ignored there.

Starting from the standard bivariate quartic box-splines surface algorithm, this paper proposes an algorithm for arbitrary triangular control meshes. The created surfaces are piecewise C^1, geometric invariant, convex preserving and local for computation.

2 Surfaces of Bivariate 3-Directional Quartic Box-Splines

The analytical definition of box-spline is given as:

For a given $B_{s\times n}$, each row of which $\in \boldsymbol{R}^s\backslash 0$. Its box-spline M_A can be defined by the following distribution

$$M_B : C(R^s) \mapsto R : \varphi \mapsto \langle M_B, \varphi \rangle = \int_b \varphi(Bt)\mathrm{d}t$$

where $b=[0,\ldots,1)^n$ is a semi-closed regular polyhedron with n edges. For $B\cup\zeta$ with $\zeta\in\boldsymbol{R}^s\backslash 0$, $M_{B\cup\zeta}$ can be computed by the following convolution

$$M_{B\cup\zeta} = \int_0^1 M_B(\cdot - t\zeta)\mathrm{d}t$$

More basic properties of box splines can be referred to Ref.[3] and will not be stated here. Bivariate 3-directional quartic box-splines (hereafter noted as M) are defined on Bivariate 3-directional subdivision. In the light of the inductive definition of box-splines, it is the convolution of two Courant hat functions, i.e. $M = M_{\begin{bmatrix}1&0&1\\0&1&1\end{bmatrix}} * M_{\begin{bmatrix}1&0&1\\0&1&1\end{bmatrix}}$. It is a bivariate function of piecewise quartic polynomials as shown in Fig.1 (left, right is its support).

Taking M as basic functions, a bivariate 3-directional surface can be defined as: $S(\boldsymbol{u}) = \sum d_j M(\boldsymbol{u}-\boldsymbol{j})$, $\boldsymbol{u}\in\boldsymbol{R}^2$, $\boldsymbol{j}\in\boldsymbol{Z}^2$, d_j are de Boor points. It is C^2 and is enough for engineering applications of CAGD.

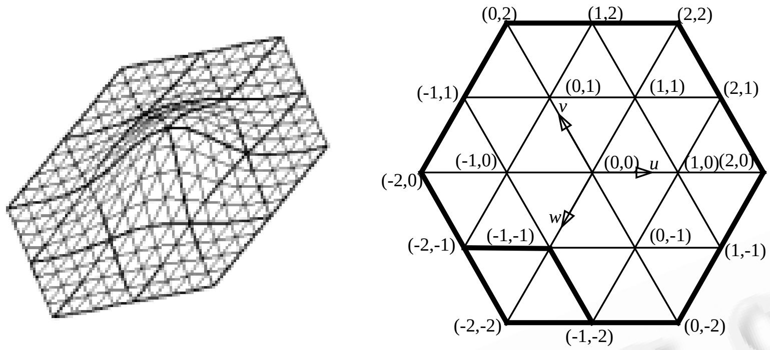


Fig.1 Bivariate 3-directional quartic box splines

3 Surfaces for Arbitrary Triangulations

For a de Boor point d_j, its 1-neighborhood is defined as the set of all points indented to it. For a triangle, its 1-neighborhood is defined as the union of 1-neighborhoods of its 3 vertices. The valence V_j of d_j is defined as the number of all points in its 1-neighborhood. In the following, a surface algorithm will be clarified for meshes with de Boor points of arbitrary valences.

3.1 Algorithm

As a triangle $\Delta d_{0,0}d_{1,0}d_{1,1}$ of control mesh is concerned (see Fig.2), the patch in the surface corresponds to the parametric domain S: $0\le v\le u\le 1$. After a simple analysis, M_j ($M_j=M(\boldsymbol{u}-\boldsymbol{j})$) restricted to the triangle is zero for $\boldsymbol{j}$s, which are not in the set of the listed 12 index pairs as shown in Figure 2. In other words, a triangular patch in the surface is entirely determined by 3 vertices of its control triangle and 9 de Boor points around the triangle, and its 1-neighborhood. So, for a control mesh, the whole surface can be generated patch by patch.

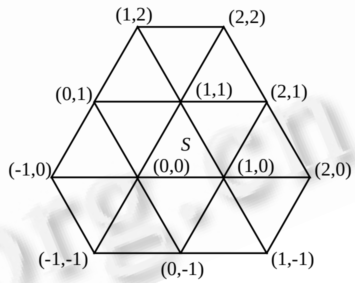


Fig.2 12 de Boor points that determine a triangular patch

For the de Boor points with arbitrary valences, an algorithm is constructed here. A triangular patch is determined by the 1-neighborhood of the corresponding control triangle.

In control mesh, 3 vertices of a triangle $\Delta d_i d_j d_k$ (listed anticlockwise in triangulation) are noted as $d_{0,0}$, $d_{1,0}$ and $d_{1,1}$. Corresponding to the edge $d_{0,0}d_{1,0}$, in the quadrilateral with $d_{0,0}d_{1,0}$ as the diagonal, the point across $d_{0,0}d_{1,0}$ from $d_{1,1}$ is noted as $d_{0,-1}$. Similarly $d_{2,1}$ and $d_{0,1}$ are got corresponding to $d_{1,0}d_{1,1}$ and $d_{1,1}d_{0,0}$ respectively. Now 6 points are obtained and put in a point matrix

$$\boldsymbol{d}_b=\begin{bmatrix} d_{0,1} & d_{1,1} & d_{2,1} \\ d_{0,0} & d_{1,0} & 0 \\ d_{0,-1} & 0 & 0 \end{bmatrix}.$$

Each triangle has such a relevant point matrix.

Relative to d_b, $\boldsymbol{N}_b$ is defined

$$\boldsymbol{N}_b=\begin{bmatrix} M_{0,1}-\beta_{0,0}-\beta_{1,1} & M_{1,1}-\beta_{0,0}-\beta_{1,0} & M_{2,1}-\beta_{1,0}-\beta_{1,1} \\ M_{0,0}-\beta_{1,0}-\beta_{1,1} & M_{1,0}-\beta_{0,0}-\beta_{1,1} & 0 \\ M_{0,-1}-\beta_{0,0}-\beta_{1,0} & 0 & 0 \end{bmatrix}$$

Where β_j are the functions relevant to vertices d_j, $\boldsymbol{j}\in\{(0,0),(1,0),(1,1)\}$.

The patch corresponding to $\Delta d_i d_j d_k$ can be computed as

$$S(\boldsymbol{u})=\sum_{(i,j)=(1,1)}^{(3,3)}\boldsymbol{d}_b(i,j)\boldsymbol{N}_b(i,j)+\sum_{\boldsymbol{j}}A_{\boldsymbol{j}} \tag{1}$$

β_j in N_b and A_j in (1) are calculated according to the valence of d_j. For an open triangulation, there are no patches for boundary triangles, so only inner points are considered. V_j of an inner point is at least 3. A_j are calculated as follows:

(1) V_j=3

$$A_{0,0}=d_{-1,-1}M_{-1,-1}+d_{-1,0}M_{-1,0};\ A_{1,0}=d_{2,0}M_{2,0}+d_{1,-1}M_{1,-1};\ A_{1,1}=d_{1,2}M_{1,2}+d_{2,2}M_{2,2};$$

(2) V_j=4

$$A_{0,0}=d_{0,1}(M_{-1,-1}+2\beta_{0,0})+d_{0,-1}(M_{-1,0}+2\beta_{0,0});\ A_{1,0}=d_{0,-1}(M_{2,0}+2\beta_{1,0})+d_{2,1}(M_{1,-1}+2\beta_{1,0});$$
$$A_{1,1}=d_{2,1}(M_{1,2}+2\beta_{1,1})+d_{0,1}(M_{2,2}+2\beta_{1,1});$$

(3) V_j=5

$$A_{0,0}=d_{-1,0}(\alpha_{0,0}-\beta_{0,0});\ A_{1,0}=d_{2,0}(\alpha_{1,0}-\beta_{1,0});\ A_{1,1}=d_{1,2}(\alpha_{1,1}-\beta_{1,1})$$

(4) V_j>6

$$A_{0,0}=\frac{6\beta_{0,0}}{V_{0,0}-6}\sum_{i=1}^{V_{0,0}-6}d_{0,0}^i+d_{-1,0}(M_{-1,0}-\beta_{0,0})+d_{-1,-1}(M_{-1,-1}-\beta_{0,0});$$
$$A_{1,0}=\frac{6\beta_{1,0}}{V_{1,0}-6}\sum_{i=1}^{V_{1,0}-6}d_{1,0}^i+d_{1,-1}(M_{-1,-1}-\beta_{1,0})+d_{2,0}(M_{2,0}-\beta_{1,0});$$
$$A_{1,1}=\frac{6\beta_{1,1}}{V_{1,1}-6}\sum_{i=1}^{V_{1,1}-6}d_{1,1}^i+d_{1,2}(M_{1,2}-\beta_{1,1})+d_{2,2}(M_{2,2}-\beta_{1,1}).$$

Where $d_{0,0}^i$, $d_{1,0}^i$ and $d_{1,1}^i$ are $V_{0,0}$–6 points between $d_{-1,0}$ and $d_{-1,-1}$, $V_{1,0}$–6 points between $d_{1,-1}$ and $d_{2,0}$, $V_{1,1}$–6 points between $d_{1,2}$ and $d_{2,2}$ respectively. When V_j=3, β_j=0; when V_j=4, V_j=5 and V_j>6, $\beta_j=\frac{V_j-6}{V_j}\theta_j$. θ_j are

$$\begin{cases}\theta_{0,0}=(1-\phi_{0,0})M_{-1,0}+\phi_{0,0}M_{-1,-1}\\ \theta_{1,0}=(1-\phi_{1,0})M_{1,-1}+\phi_{1,0}M_{2,0}\\ \theta_{1,1}=(1-\phi_{1,1})M_{2,2}+\phi_{1,1}M_{1,2}\end{cases}$$ and α_j are $$\begin{cases}\alpha_{0,0}=(1-\phi_{0,0})M_{-1,-1}+\phi_{0,0}M_{-1,0}\\ \alpha_{1,0}=(1-\phi_{1,0})M_{2,0}+\phi_{1,0}M_{1,-1}\\ \alpha_{1,1}=(1-\phi_{1,1})M_{1,2}+\phi_{1,1}M_{2,2}\end{cases}$$, where ϕ_j are defined as

$$\begin{cases}\phi_{0,0}(\boldsymbol{u})=(-2u_2^3+3u_1u_2^2)/u_1^3\\ \phi_{1,0}(\boldsymbol{u})=(-2(1-u_1)^3+3(1-u_1+u_2)(1-u_1)^2)/(1-u_1+u_2)^3\\ \phi_{1,1}(\boldsymbol{u})=(-2(u_1-u_2)^3+3(1-u_2)(u_1-u_2)^2)/(1-u_2)^2\end{cases}\quad(\boldsymbol{u}=(u_1,u_2)\in\boldsymbol{R}^2).$$

3.2 Analysis of the algorithm

As an approximation of the de Boor control mesh, the surface $S(\boldsymbol{u})$ is a piecewise polynomial surface. It is clear that $S(\boldsymbol{u})$ holds geometric invariant, local properties, etc.

$S(\boldsymbol{u})$ can also be expressed as $S(\boldsymbol{u})=\sum\lambda_i d_i$. According to the algorithm, $\lambda_i\geq 0$ and $\sum\lambda_i\equiv 1$. So the surfaces by the algorithm hold the property of convex-preserving.

The following will concentrate on the analysis of continuity. Let two patches S and $\tilde{S}$ have the common boundary $d_{0,0}d_{1,0}$, i.e. $\tilde{d}_{1,0}\tilde{d}_{0,0}$ (can be referred to Fig.4). At the boundary, the parametric points (u,v) and $(\tilde{u},\tilde{v})$ meet $\tilde{u}=1-u$ and $v=\tilde{v}=0$. Under such common boundary, there exists the following lemma.

Lemma. For S and $\tilde{S}$, at the boundary ($\tilde{u}=1-u$, $v=\tilde{v}=0$), $\tilde{M}_{2,0}=M_{-1,0}$, $\frac{\partial\tilde{M}_{2,0}}{\partial u}=\frac{\partial M_{-1,0}}{\partial u}$,

$\frac{\partial \tilde{M}_{2,0}}{\partial v}=\frac{\partial M_{-1,0}}{\partial v}$, $\frac{\partial^2 \tilde{M}_{2,0}}{\partial u^2}=\frac{\partial^2 M_{-1,0}}{\partial u^2}$, $\frac{\partial^2 \tilde{M}_{2,0}}{\partial v^2}=\frac{\partial^2 M_{-1,0}}{\partial v^2}$, $\frac{\partial^2 \tilde{M}_{2,0}}{\partial u\partial v}=\frac{\partial^2 M_{-1,0}}{\partial u\partial v}$ and $\frac{\partial^2 \tilde{M}_{2,0}}{\partial v\partial u}=\frac{\partial^2 M_{-1,0}}{\partial v\partial u}$. Similarly, $\tilde{M}_{-1,0}$ and $M_{2,0}$, $\tilde{M}_{0,1}$ and $M_{1,-1}$, $\tilde{M}_{1,-1}$ and $M_{0,1}$, $\tilde{M}_{0,-1}$ and $M_{1,1}$, $\tilde{M}_{1,1}$ and $M_{0,-1}$, $\tilde{M}_{-1,-1}$ and $M_{2,1}$, $\tilde{M}_{2,1}$ and $M_{-1,-1}$, $\tilde{M}_{0,0}$ and $M_{1,0}$, $\tilde{M}_{1,0}$ and $M_{0,0}$ have the same relationship as $\tilde{M}_{2,0}$ and $M_{-1,0}$.

Proof: From the symmetry of M and boundary condition, it is clear that: $\tilde{M}_{2,0}=M(\tilde{u}-2,\tilde{v})=M(-u-1,-v)=M(u+1,v)=M_{-1,0}$. Because of $v=\tilde{v}=0$, then

$$\frac{\partial \tilde{M}_{2,0}}{\partial u}=\frac{\partial M(\tilde{u}-2,\tilde{v})}{\partial \tilde{u}}\frac{\partial \tilde{u}}{\partial u}=-\left.\frac{\partial M(\tilde{u}-2,0)}{\partial \tilde{u}}\right|_{\tilde{u}=1-u}.$$

After a brief computation, $-\left.\frac{\partial M(\tilde{u}-2,0)}{\partial \tilde{u}}\right|_{\tilde{u}=1-u}=\frac{\partial M(u+1,0)}{\partial u}=\left.\frac{\partial M_{-1,0}}{\partial u}\right|_{v=0}$, i.e. $\frac{\partial \tilde{M}_{2,0}}{\partial u}=\frac{\partial M_{-1,0}}{\partial u}$. It is also easy to get other relations stated in the lemma between $\tilde{M}_{2,0}$ and $M_{-1,0}$.

Similarly, the rest between M and $\tilde{M}$ can be proved without any difficulty.

The relation between M and $\tilde{M}$ in lemma is under the boundary condition $\tilde{u}=1-u$, $v=\tilde{v}=0$. There exists a similar relation if under another kind of boundary.

Theorem 1. Each patch of the surface is C^2.

According to Eq.(1) and the calculation of A_j, it is known that in the inner of the patch, every second derivative of $S(\boldsymbol{u})$ exists and is continuous. So each patch of the surface is C^2.

Theorem 2. If each de Boor point has valence 3, every two adjacent patches have C^2 continuity at the common boundary.

Proof: A close mesh of 4 de Boor points is shown in Fig.3. Each point is 4 duplicated. The patches S and $\tilde{S}$ have the common boundary of $\tilde{u}=1-u$ and $v=\tilde{v}=0$. For M, when v=0, $M_{1,2}$, $M_{2,2}$ and their first and second derivatives with respect to u and v are 0. Then at the boundary

$$S=d_{0,0}(M_{0,0}+M_{2,0})+d_{1,0}(M_{1,0}+M_{-1,0})+d_{1,1}(M_{1,1}+M_{-1,-1}+M_{1,-1})+d_{0,-1}(M_{0,-1}+M_{0,1}+M_{2,1})$$

and

$$\begin{aligned}\tilde{S}&=\tilde{d}_{0,0}(\tilde{M}_{0,0}+\tilde{M}_{2,0})+\tilde{d}_{1,0}(\tilde{M}_{1,0}+\tilde{M}_{-1,0})+\tilde{d}_{1,1}(\tilde{M}_{1,1}+\tilde{M}_{-1,-1}+\tilde{M}_{1,-1})+\tilde{d}_{0,-1}(\tilde{M}_{0,-1}+\tilde{M}_{0,1}+\tilde{M}_{2,1})\\&=d_{1,0}(\tilde{M}_{0,0}+\tilde{M}_{2,0})+d_{0,0}(\tilde{M}_{1,0}+\tilde{M}_{-1,0})+d_{0,-1}(\tilde{M}_{1,1}+\tilde{M}_{-1,-1}+\tilde{M}_{1,-1})+d_{1,1}(\tilde{M}_{0,-1}+\tilde{M}_{0,1}+\tilde{M}_{2,1})\end{aligned}$$

Through comparing the coefficients of the same points in the two formulae, S and $\tilde{S}$ is C^2 at the boundary according to the lemma.

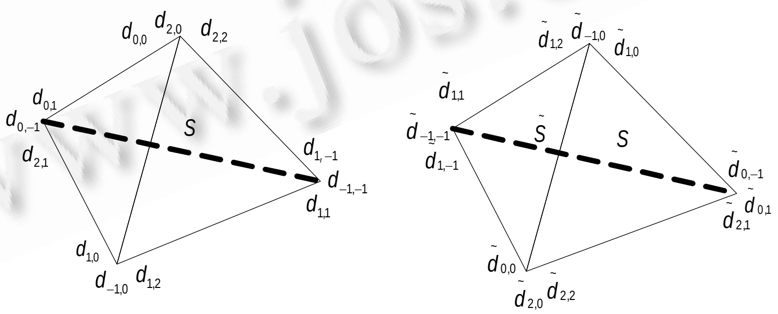


Fig.3 When every de Boor point has valence 3

Theorem 3. For an arbitrary triangular mesh, surface by the algorithm is piecewise C^1.

Proof: It is still supposed that the boundary of S and $\tilde{S}$ is $d_{0,0}d_{1,0}$ ($\tilde{d}_{1,0}\tilde{d}_{0,0}$) and meets $\tilde{u}=1-u$ and $v=\tilde{v}=0$.

Because when v=0, $M_{1,2}$, $M_{2,2}$ and their first and second derivatives with respect to u and v are all 0, $\theta_{1,1}$ is so as well. Then $\alpha_{1,1}$ and $\beta_{1,1}$ are 0. Then $A_{1,1}$=0, $\frac{\partial A_{1,1}}{\partial u}=0$, $\frac{\partial A_{1,1}}{\partial v}=0$. Given ϕ_j and α_j, when, $\phi_{0,0}$=0, $\frac{\partial \phi_{0,0}}{\partial u}=0$ and $\frac{\partial \phi_{0,0}}{\partial v}=0$. So, $\theta_{0,0}$ and $M_{-1,0}$, $\alpha_{0,0}$ and $M_{-1,-1}$ have the following relation

$$\begin{cases}\theta_{0,0}=M_{-1,0},\ \dfrac{\partial\theta_{0,0}}{\partial u}=\dfrac{\partial M_{-1,0}}{\partial u},\ \dfrac{\partial\theta_{0,0}}{\partial v}=\dfrac{\partial M_{-1,0}}{\partial v}\\ \alpha_{0,0}=M_{-1,-1},\ \dfrac{\partial\alpha_{0,0}}{\partial u}=\dfrac{\partial M_{-1,-1}}{\partial u},\ \dfrac{\partial\alpha_{0,0}}{\partial v}=\dfrac{\partial M_{-1,-1}}{\partial v}\end{cases} \tag{2}$$

Similarly, when v=0, $\phi_{1,0}$=1, $\frac{\partial \phi_{1,0}}{\partial u}=0$, and $\frac{\partial \phi_{1,0}}{\partial v}=0$. $\theta_{1,0}$ and $M_{2,0}$, $\alpha_{1,0}$ and $M_{1,-1}$ have the relation

$$\begin{cases}\theta_{1,0}=M_{2,0},\ \dfrac{\partial\theta_{1,0}}{\partial u}=\dfrac{\partial M_{2,0}}{\partial u},\ \dfrac{\partial\theta_{1,0}}{\partial v}=\dfrac{\partial M_{2,0}}{\partial v}\\ \alpha_{1,0}=M_{1,-1},\ \dfrac{\partial\alpha_{1,0}}{\partial u}=\dfrac{\partial M_{1,-1}}{\partial u},\ \dfrac{\partial\alpha_{1,0}}{\partial v}=\dfrac{\partial M_{1,-1}}{\partial v}\end{cases} \tag{3}$$

As $V_{0,0}=\tilde{V}_{1,0}$, $V_{1,0}=\tilde{V}_{0,0}$, from Eqs.(2) and (3) and the lemma, $\beta_{0,0}$ and $\tilde{\beta}_{1,0}$ meet

$$\beta_{0,0}=\tilde{\beta}_{1,0},\ \frac{\partial\beta_{0,0}}{\partial u}=\frac{\partial\tilde{\beta}_{1,0}}{\partial u},\ \frac{\partial\beta_{0,0}}{\partial v}=\frac{\partial\tilde{\beta}_{1,0}}{\partial v} \tag{4}$$

$\beta_{1,0}$ and $\tilde{\beta}_{0,0}$ meet

$$\beta_{1,0}=\tilde{\beta}_{0,0},\ \frac{\partial\beta_{1,0}}{\partial u}=\frac{\partial\tilde{\beta}_{0,0}}{\partial u},\ \frac{\partial\beta_{1,0}}{\partial v}=\frac{\partial\tilde{\beta}_{0,0}}{\partial v} \tag{5}$$

Now S and $\tilde{S}$ can be denoted as

$$\begin{aligned}S=&d_{0,0}(M_{0,0}-\beta_{1,0})+d_{1,0}(M_{1,0}-\beta_{0,0})+d_{1,1}(M_{1,1}-\beta_{0,0}-\beta_{1,0})+d_{0,-1}(M_{0,-1}-\beta_{0,0}-\beta_{1,0})+\\ &[d_{0,1}(M_{0,1}-\beta_{0,0})+A_{0,0}]+[d_{2,1}(M_{2,1}-\beta_{1,0})+A_{1,0}]\end{aligned} \tag{6}$$

and

$$\begin{aligned}\tilde{S}=&\tilde{d}_{0,0}(\tilde{M}_{0,0}-\tilde{\beta}_{1,0})+\tilde{d}_{1,0}(\tilde{M}_{1,0}-\tilde{\beta}_{0,0})+\tilde{d}_{1,1}(\tilde{M}_{1,1}-\tilde{\beta}_{0,0}-\tilde{\beta}_{1,0})+\tilde{d}_{0,-1}(\tilde{M}_{0,-1}-\tilde{\beta}_{0,0}-\tilde{\beta}_{1,0})+\\ &[\tilde{d}_{0,1}(\tilde{M}_{0,1}-\tilde{\beta}_{0,0})+\tilde{A}_{0,0}]+[\tilde{d}_{2,1}(\tilde{M}_{2,1}-\tilde{\beta}_{1,0})+\tilde{A}_{1,0}]\\ =&d_{1,0}(\tilde{M}_{0,0}-\tilde{\beta}_{1,0})+d_{0,0}(\tilde{M}_{1,0}-\tilde{\beta}_{0,0})+d_{0,-1}(\tilde{M}_{1,1}-\tilde{\beta}_{0,0}-\tilde{\beta}_{1,0})+d_{1,1}(\tilde{M}_{0,-1}-\tilde{\beta}_{0,0}-\tilde{\beta}_{1,0})+\\ &[\tilde{d}_{0,1}(\tilde{M}_{0,1}-\tilde{\beta}_{0,0})+\tilde{A}_{0,0}]+[\tilde{d}_{2,1}(\tilde{M}_{2,1}-\tilde{\beta}_{1,0})+\tilde{A}_{1,0}]\end{aligned} \tag{7}$$

Making use of the lemma and Eqs.(4) and (5), let the coefficients in Eqs.(6) and (7) be compared, it is found that the former 4 terms are correspondingly equal, as well as their first derivatives.

Let $C_{0,0}=[d_{0,1}(M_{0,1}-\beta_{0,0})+A_{0,0}]$, $C_{1,0}=[d_{2,1}(M_{2,1}-\beta_{1,0})+A_{1,0}]$. The analysis will proceed under different conditions.

(1) One point of boundary has valence 3.

In Fig.4, for S and $\tilde{S}$ $V_{1,0}=\tilde{V}_{0,0}=3$. Points are noted as in the figure. According to the algorithm,

$$C_{1,0}=d_{2,1}M_{2,1}+d_{2,0}M_{2,0}+d_{1,-1}M_{1,-1}=d_{2,1}M_{2,1}+d_{0,0}M_{2,0}+d_{1,1}M_{1,-1}$$

and

$$\tilde{C}_{0,0}=\tilde{d}_{0,1}\tilde{M}_{0,1}+\tilde{d}_{-1,-1}\tilde{M}_{-1,-1}+\tilde{d}_{-1,0}\tilde{M}_{-1,0}=d_{1,1}\tilde{M}_{0,1}+d_{2,1}\tilde{M}_{-1,-1}+d_{0,0}\tilde{M}_{-1,0}$$

Clearly, $C_{1,0}$ and $\tilde{C}_{0,0}$ are equal, as well as their first derivatives.

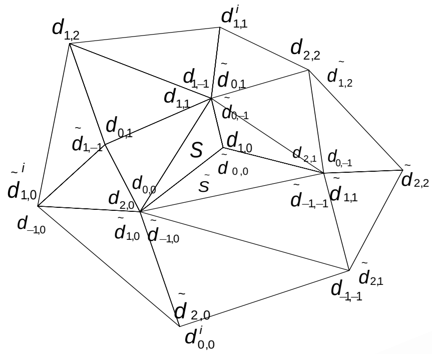


Fig.4 When two points of boundary edge shared by two patches have valences of 3 and >6

(2) One point of boundary has valence 4.

In Fig.5, for S and $\tilde S$ $V_{1,0}=\tilde V_{0,0}=4$.

$$C_{1,0}=d_{2,1}(M_{2,1}-\beta_{1,0})+d_{0,-1}(M_{2,0}+2\beta_{1,0})+d_{2,1}(M_{1,-1}+2\beta_{1,0})=d_{2,1}(M_{2,1}+M_{1,-1}+\beta_{1,0})+d_{0,-1}(M_{2,0}+2\beta_{1,0})$$

and $\tilde C_{0,0}=\tilde d_{0,1}(\tilde M_{0,1}-\tilde\beta_{0,0})+\tilde d_{0,1}(\tilde M_{-1,-1}+2\tilde\beta_{0,0})+\tilde d_{0,-1}(\tilde M_{-1,0}+2\tilde\beta_{0,0})=d_{2,1}(\tilde M_{0,1}+\tilde M_{-1,-1}+\tilde\beta_{0,0})+\tilde d_{0,-1}(\tilde M_{-1,0}+2\tilde\beta_{0,0})$.

Note that: $M_{2,0}+2\beta_{1,0}=M_{2,0}-[(1-\phi_{1,0})M_{1,-1}+\phi_{1,0}M_{2,0}]=(1-\phi_{1,0})(M_{2,0}-M_{1,-1})$ Considering $\phi_{1,0}$ under v=0, $M_{2,0}+2\beta_{1,0}$ and its first derivative is 0. Similarly, $\tilde M_{-1,0}+2\tilde\beta_{0,0}$ has the same results. So, $C_{1,0}$ and $\tilde C_{0,0}$ are equal, as well as to their first derivatives.

(3) One point of boundary has valence 5.

In Fig.5, for S and $\tilde S$ $V_{1,0}=\tilde V_{0,0}=5$. According to the algorithm, $C_{0,0}=d_{0,1}(M_{0,1}-\beta_{0,0})+d_{-1,0}(\alpha_{0,0}-\beta_{0,0})$ and $\tilde C_{1,0}=\tilde d_{2,1}(\tilde M_{2,1}-\tilde\beta_{1,0})+\tilde d_{2,0}(\tilde\alpha_{1,0}-\tilde\beta_{1,0})=d_{-1,0}(\tilde M_{2,1}-\tilde\beta_{1,0})+d_{0,1}(\tilde\alpha_{1,0}-\tilde\beta_{1,0})$.

With the relation between $\alpha_{0,0}$ and $M_{-1,-1}$ in Eq.(2), and that between $\tilde M_{2,1}$ and $M_{-1,-1}$ stated in the lemma, $\alpha_{0,0}$ and $\tilde M_{2,1}$ are equal, as well as their first derivatives. So are $\tilde\alpha_{1,0}$ and $M_{0,1}$ for the same reason. Then $C_{1,0}$ equals to $\tilde C_{0,0}$, and they have the same first derivatives.

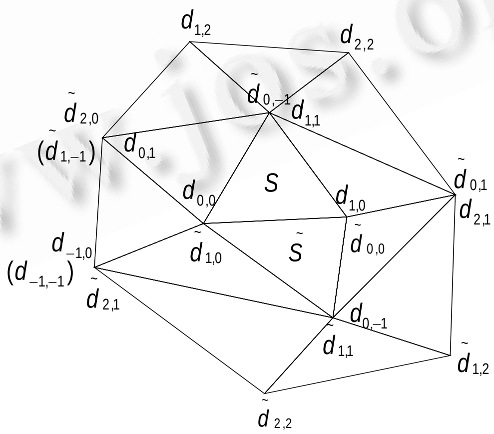


Fig.5 When two points of boundary edge shared by two patches have valences of 4 and 5

(4) One point of boundary has valence >6.

In Fig.4, for S and $\tilde S$, $V_{1,0}=\tilde V_{0,0}>6$. In the light of the algorithm,

$$C_{0,0}=d_{0,1}(M_{0,1}-\beta_{0,0})+\frac{6\beta_{0,0}}{V_{0,0}-6}\sum_{i=1}^{V_{0,0}-6}d_{0,0}^{i}+d_{-1,0}(M_{-1,0}-\beta_{0,0})+d_{-1,-1}(M_{-1,-1}-\beta_{0,0})$$

$$=d_{0,1}(M_{0,1}-\beta_{0,0})+d_{-1,0}(M_{-1,0}-\beta_{0,0})+d_{-1,-1}(M_{-1,-1}-\beta_{0,0})+d_{0,0}^{V_{0,0}-6}\frac{6\beta_{0,0}}{V_{0,0}-6}+\frac{6\beta_{0,0}}{V_{0,0}-6}\sum_{i=1}^{V_{0,0}-5}d_{0,0}^{i}$$

and

$$\tilde{C}_{1,0}=\tilde{d}_{2,1}(\tilde{M}_{2,1}-\tilde{\beta}_{1,0})+\frac{6\tilde{\beta}_{1,0}}{\tilde{V}_{1,0}-6}\sum_{i=1}^{\tilde{V}_{1,0}-6}\tilde{d}_{1,0}^{i}+\tilde{d}_{1,-1}(\tilde{M}_{1,-1}-\tilde{\beta}_{1,0})+\tilde{d}_{2,0}(\tilde{M}_{2,0}-\tilde{\beta}_{1,0})$$

$$=\tilde{d}_{2,1}(\tilde{M}_{2,1}-\tilde{\beta}_{1,0})+\tilde{d}_{1,0}^{1}\frac{6\tilde{\beta}_{1,0}}{\tilde{V}_{1,0}-6}+\frac{6\tilde{\beta}_{1,0}}{\tilde{V}_{1,0}-6}\sum_{i=2}^{\tilde{V}_{1,0}-6}\tilde{d}_{1,0}^{i}+\tilde{d}_{1,-1}(\tilde{M}_{1,-1}-\tilde{\beta}_{1,0})+\tilde{d}_{2,0}(\tilde{M}_{2,0}-\tilde{\beta}_{1,0})$$

$$=d_{0,1}(\tilde{M}_{1,-1}-\tilde{\beta}_{1,0})+d_{-1,0}\frac{6\tilde{\beta}_{1,0}}{\tilde{V}_{1,0}-6}+d_{-1,-1}(\tilde{M}_{2,1}-\tilde{\beta}_{1,0})+d_{0,0}^{V_{0,0}-6}(\tilde{M}_{2,0}-\tilde{\beta}_{1,0})+\frac{6\tilde{\beta}_{1,0}}{\tilde{V}_{1,0}-6}\sum_{i=1}^{V_{0,0}-5}d_{0,0}^{i}$$

With the relation between $\theta_{0,0}$ and $M_{-1,0}$ in Eq.(2), in the formula of $C_{0,0}$, $M_{-1,0}-\beta_{0,0}$ and $M_{-1,0}-\frac{V_{0,0}-6}{V_{0,0}}\theta_{0,0}=(1-\frac{V_{0,0}-6}{V_{0,0}})M_{-1,0}=\frac{6}{V_{0,0}}M_{-1,0}$ have the same relation. With the relation between $\theta_{1,0}$ and $M_{2,0}$ in Eq.(3), in the formula of $\tilde{C}_{1,0}$, $\frac{6\tilde{\beta}_{1,0}}{V_{0,0}-6}=\frac{6}{V_{0,0}-6}\frac{V_{0,0}-6}{V_{0,0}}\tilde{\theta}_{1,0}=\frac{6}{V_{0,0}-6}\tilde{\theta}_{1,0}$ and $\frac{6}{V_{0,0}}\tilde{M}_{2,0}$ have the same relation.

With the relation between $\tilde{M}_{2,0}$ and $M_{-1,0}$ in the lemma, the coefficients of $d_{-1,0}$ are equal in $C_{0,0}$ and $\tilde{C}_{1,0}$, as well as their first derivatives. So are the coefficients of $d_{0,0}^{V_{0,0}-6}$ in $C_{0,0}$ and $\tilde{C}_{1,0}$. Moreover, with the relationship between $\beta_{0,0}$ and $\tilde{\beta}_{1,0}$, and those stated in the lemma, $C_{0,0}$ and $\tilde{C}_{1,0}$ have equal value and first derivatives.

From the analysis, it turns out that no matter what valences the boundary edge have, $C_{0,0}$ and $\tilde{C}_{1,0}$, $C_{1,0}$ and $\tilde{C}_{0,0}$, as well as their first derivatives are correspondingly equal respectively. Therefore, S and $\tilde{S}$ are C^1 at their boundary.

The proof above is only for the boundary of $\tilde{u}=1-u$ and $v=\tilde{v}=0$, but for other kinds of boundary, the sameconclusions can be obtained. So, it can be summarized that: for an arbitrary triangular control mesh, its surface by the algorithm is piecewise C^1.

3.3 Interpolation

It is very easy to apply the algorithm to global or local interpolation. From the algorithm, it is easy to find that point P_i related to de Boor point d_i is determined as

$$P_i=\frac{1}{2}(d_i+\frac{1}{\boldsymbol{V}_i}\sum_{d_i\in I_i}d_i) \tag{8}$$

Where I_i is the 1-neiborhood of d_i. So the equation $A\boldsymbol{D}=\boldsymbol{P}$ is constructed, where $A(i,i)=\frac{1}{2}$ and if $d_j\in I_i$ then $A(i,j)=\frac{1}{2V_i}$, otherwise $(d_j\notin I_i)A(i,j)=0$. It is clear that A^{-1} exsits, so from Eq.(8) the control meshes $\boldsymbol{D}$ for the point-set $\boldsymbol{P}$ can be obtained.

For a local interpolation, i.e. the constructed surfaces interpolate some given points, the control meshes are also easy to compute.

Let the final surface interpolate the point-set $\boldsymbol{P}_I\subset\boldsymbol{P}$. $\boldsymbol{P}\backslash\boldsymbol{P}_I$ is part of the control points. It is necessary to compute the control meshes $\boldsymbol{D}_I$ corresponding to $\boldsymbol{P}_I$. Then the whole control points are $\boldsymbol{D}_I\cup\boldsymbol{P}\backslash\boldsymbol{P}_I$.

For any point $P\in\boldsymbol{P}_I$, an equation can be constructed from Eq.(8). It is esay to get: $A_l d_l=P_l$, where $d_l=[d_1, d_2,\ldots,d_i,\ldots]$, $d_i\in\boldsymbol{D}_I$.

4 Experiments

The first example is shown in Fig.6. Each point in a close mesh of 4 de Boor points have valence 3. The surface comprise of 4 patches. Every two adjacent patches are C^2 continuous at their common boundary.

Another example is shown in Fig.7. The right above is the mesh's projection to a plane. The control mesh is an open one in which the valences are 3, 4, 5 and 6. Every two adjacent patches have continuity of C^1.

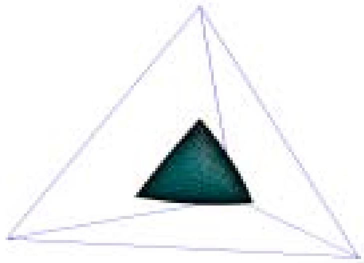


Fig.6 Example 1, all 4 de Boor points with valence 3

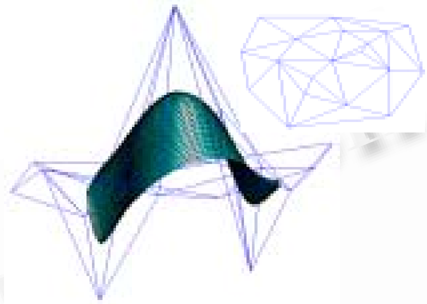


Fig.7 Example 2, de Boor points with arbitrary valences

The third example is for a global interpolation. The original point-set contains 8 102 sampled points from a man face model. Then it is triangulated with the method in Ref.[16]. The resulted triangulation is of arbitrary random point valences. As shown in Fig.8, based on the triangulation, the interpolative surface is reconstructed by the algorithm. The relative error is 0.033. More detailed discussion can be found in Ref.[17].

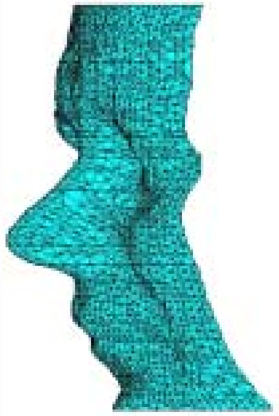


Fig.8 Example 3, A man face reconstructed from an arbitrary triangulation

5 Conclusions

The analysis and experiments indicate that the surface algorithm proposed in this paper can be applied to arbitrary triangular control meshes for approximation, global and local interpolation. Surface generated by the algorithm holds a better smoothness as a whole. It is piecewise C^1 and C^2 in the inner of each patch. In addition, surfaces by the algorithm hold the convex-preserving property.

Future work may lie in algorithm excellence in global or local interpolative surfaces for open triangulations. This is involved in how to build pseudo boundaries.

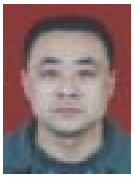
References:

[1] Boehm W. Triangular spline algorithms. Computer Aided Geometric Design, 1985,2(1):61–67.

[2] de Boor C, Hollig K, Riemenschneider S. Box splines. New York: Springer-Verlag, 1993. 1–32.

[3] Boehm W. Calculating with box splines. Computer Aided Geometric Design, 1984,1(2):149−162.
[4] Chui CK, Wang RH. Spaces of bivariate cubic and quartic splines on type-1 triangulations. Journal of Mathematic Analysis & Application, 1984,101(2):540−554.
[5] He WJ, Lai MJ. Construction of bivariate compactly supported biorthogonal box spline wavelets with arbitrarily high regularities. Applied and Computational Harmonic Analysis, 1999,(6):53−74.
[6] Dreger A, Gross MH, Schlegel J. Multiresolution triangular B-spline surfaces. In: Wolter FE, ed. Proc. of the Int'l Computer Graphics. Hannover: IEEE Computer Society, 1998. 166−177.
[7] Zhao HX. The montonicity and convexity of the bivariate box spline surfaces. Aeronautical Computer Technique, 2001,31(3): 22−26 (in Chinese with English abstract).
[8] Loop, C. Smooth subdivision surfaces based on triangles [MS. Thesis]. University of Utah, 1987.
[9] Dyn D, Levin D, Gregory J. A butterfly subdivision scheme for surfaces interpolation with tension control. ACM Trans. on Graphics, 1990,9(2):160−169.
[10] Zorin D, Schroder W, Sweldens W. Interpolating subdivision for meshes with arbitrary topology. In: Rushneier H, ed. Proc. of the SIGGRAPH '96, 23rd Annual Conf. Computer Graphics & Interactive Techniques. New Orleans: ACM Press, 1996. 189−192.
[11] Qu R, Agarwal RP. Smooth surfaces interpolation to scattered data using interpolatory subdivision algorithms. Computer Math. Apllications, 1996,32(3):93−110.
[12] Kolb A, Seidel H. Interpolating scattered data with C^2 surfaces. Computer Aided Design, 1995,27(4):277−282.
[13] Li A. Convexity preserving interpolation. Computer Aided Geometric Design, 1999,16(7):127−147.
[14] Ueshiba T, Roth G. Generating smooth surfaces with bicubic splines over triangular meshes: Toward automatic model building from unorganized 3D points. In: Werner B, ed. Proc. of the 2nd Int'l Conf. 3D Imaging and Modeling. Ottawa: IEEE Computer Society, 1999. 302−311.
[15] Loop C. A G^1 triangular spline surface of arbitrary topological type. Computer Aide Geometric Design, 1994,11(3):303−330.
[16] Zhang YC, Da FP, Song WZ. Triangulations based on a criterion of minimized curvature for scattered point-sets. Journal of Southeast University (Natural Science Edition), 2004,34(6): 851−856 (in Chinese with English abstract).
[17] Zhang YC. Sampling and reconstruction for 3D surfaces [Ph.D. Thesis]. Nanjing: Southeast University, 2005. 87−90 (in Chinese with English abstract).

附中文参考文献:
[7] 赵红星.二元箱样条曲面的凸性和单调性.航空计算技术,2001,31(3):22−26.
[16] 张永春,达飞鹏,宋文忠.基于一种曲率最小优化准则的散乱点三角剖分.东南大学学报(自然科学版),2004,34(6):851−856.
[17] 张永春.三维曲面的采样与重构[博士学位论文].南京:东南大学, 2005. 87−90.



ZHANG Yong-Chun was born in 1968. He is a Ph.D. student at the Southeast University. His current research area is computer graphics.

SONG Wen-Zhong was born in 1936. He is a professor at the Southeast University. His research areas are 3D model reconstruction and advanced control theory.

DA Fei-Peng was born in 1968. He is a professor at the Southeast University. His current research areas are 3D reconstruction and intelligent control.