在任意拓扑三角形网格上的光滑样条曲面*

冯仁忠+, 王仁宏

(大连理工大学 数学科学研究所,辽宁 大连 116024)

Smooth Spline Surfaces over Arbitrary Topological Triangular Meshes

FENG Ren-Zhong+, WANG Ren-Hong

(Institute of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China)

+ Corresponding author: Phn: 86-411-4708351 ext 850, E-mail: fengrz@sohu.com, fengrz@mail.jl.cn

http://www.dlut.edu.cn

Received 2002-09-24; Accepted 2002-11-20

Feng RZ, Wang RH. Smooth spline surfaces over arbitrary topological triangular meshes. *Journal of Software*, 2003,14(4):830~837.

Abstract: The paper introduces an algorithm for creating smooth spline surfaces over control triangular meshes capable of outlining arbitrary free-form surfaces with or without boundary. The resulting surface has a degree 4 parametric polynomial representation and is represented as a network of tangent plane continuous triangular Bézier patches. The approximation of resulting surface to mesh is controlled by a blend ratio; when the blend ratio is zero, surfaces interpolate meshes. The algorithm is a local method, simple, efficient and fit for appearance design.

Key words: spline surface; control triangular mesh; triangular Bézier patch; geometric continuity; blend ratio

摘 要: 介绍了一种在控制三角形网格上创建光滑样条曲面的算法,该控制网格能够刻画具有或没有边界的任意自由曲面.生成的曲面有一个4次参数多项式表示并且被表示成一个切平面连续的三角形Bézier片网.曲面对网格的逼近程度受到一个混合比控制,当混合比为0时,产生的曲面插值网格.该算法是一种局部方法,简单且效率高,适合于外形设计.

关键词: 样条曲面;控制三角形网格;三角形Bézier片;几何连续性;混合比

中图法分类号: TP391 **文献标识码**: A

Since a single B-spline patch can only represent surfaces of simple topological type (deformed planar regions, cylinders, and torus), a surface of arbitrary topological type must be defined as a network of polynomials. Most methods using piecewise polynomials to construct a surface from a mesh of points fall into one of two categories: global or local. With a global algorithm, a large linear, irregularly sparse system of equations is solved to match

* Supported by the National Natural Science Foundation of China under Grant No.69973010 (国家自然科学基金)

FENG Ren-Zhong was born in 1967. He received his Ph.D. degree in 2000. He is an associate professor at the Institute of Mathematical Sciences, Dalian University of Technology. His research interests are computer aided geometric design and numerical approximation. **WANG Ren-Hong** was born in 1937. He is a professor and doctoral supervisor at the Institute of Mathematical Sciences, Dalian University of Technology. His current research areas are computational geometry, numerical approximation and computer aided geometric design.

data. This makes it more difficult to reason a priori about the shape of the resulting surface. The works on global algorithm are seen in Refs.[1~3]. Local algorithms avoid solving large linear systems of equations and are more geometric in nature. J.PETERS gives a few properties of local algorithm in Ref.[4]. The works on local algorithm are also referred to Refs.[5~8]. The present is a local algorithm modeling surfaces of arbitrary topological type by smoothly approximating a control triangular mesh. The advantage of this technique is

- *Free-form modeling capability* There are no restrictions on the number of triangles meeting at a mesh point.
- *Low-degree parametrization* The surface is parametrized by degree 4 triangular Bèzier patches.
- *Evaluation by averaging* The coefficients of the parametrization in Bernstein-Bèzier form can be obtained by applying mask to the input mesh. Thus the algorithm is local and can be interpreted as a rule for cutting an input polytope such that the limit polytope is the spline surface.
- *Convex hull property* The surface lies locally and globally in the convex hull of the input mesh.
- *Taut interpolation of the control mesh for zero blend ratio* Cut of zero depth result in a singular parametrization at the mesh points analogous to singularities of a spline with repeated knots. The continuity of the surface is reduced, but in return the edges of the input mesh are interpolated and the surface is taut.

1 The G^1 Continuous Conditions Between Two Adjacent Triangular Bézier Patches

1.1 Triangular patches

Triangular polynomial patches can be expressed in a Bernstein- Bézier form

$$\varphi(u,v,w)=\sum_{\substack{i+j+k=n\\ i,j,k\ge 0}} G_{i,j,k}\frac{n!}{i!\,j!\,k!}u^i v^j w^k,\qquad u+v+w=1,\qquad u,v,w\ge 0\,, \tag{1.1}$$

where coefficients $G_{i,j,k}\in R^3$. We use a shorthand notation for the coefficients: $T_i=G_{i,1,n-i-1}, i=0,\dots,n-1, S_i=G_{i,0,n-i}$, $i=0,\dots,n$. We call the coefficients controlling boundary curves except three vertices $G_{0,0,n}, G_{n,0,0}$ and $G_{0,n,0}$ edge coefficients, all interior control vertices face coefficients, such as $T_i, i=1,\dots,n-2$. We shall consider a particular cross-boundary derivative, namely,

$$[D\varphi](u)=(1-u)(\varphi_v-\varphi_u)+u(\varphi_w-\varphi_u)\,.$$

Expressed in terms of Bernstein polynomials,

$$[D\varphi](u)=n(1-u)\sum_{i=0}^{n-1}(T_i-S_i)B_i^{n-1}(u)+nu\sum_{i=0}^{n-1}(T_i-S_{i+1})B_i^{n-1}(u)\,.$$

Simple algebra yields

$$[D\varphi](u)=n\sum_{i=0}^{n}\left(\frac{n-i}{n}T_i+\frac{i}{n}T_{i-1}-S_i\right)B_i^n(u)\,. \tag{1.2}$$

Let us now consider a special case: assume the boundary curve with coefficients $S_i, i=0,\dots,n$, is only of degree $n-1$. This implies the existence of $\widetilde{S}_i, i=0,\dots,n-1$, with

$$S_i=\frac{n-i}{n}\widetilde{S}_i+\frac{i}{n}\widetilde{S}_{i-1},\qquad i=0,\dots,n. \tag{1.3}$$

Invoking the degree elevation of Bernstein polynomial[9] and (1.3), we see that (1.2) is equivalent to

$$[D\varphi](u)=n\sum_{i=0}^{n-1}(T_i-\widetilde{S}_i)B_i^{n-1}(u)\,. \tag{1.4}$$

1.2 The G^1 continuous conditions between two adjacent triangular Bézier patches

Much research has been devoted to this subject, and many approaches to the construction of Bézier surfaces that share tangent planes along their common boundary have been developed[9,10]. Let ϕ and φ be two adjacent triangular patches of degree n, all of whose boundaries are degree $n-1$ and who share a common boundary cure Γ of the form (see Fig.1)

$$\Gamma(u)=\sum_{i=0}^{n}S_iB_i^n(u)=\sum_{i=0}^{n-1}\widetilde{S}_iB_i^{n-1}(u)$$

with the derivative

$$[D\Gamma](u)=(n-1)\sum_{i=0}^{n-2}(\widetilde{S}_{i+1}-\widetilde{S}_i)B_i^{n-2}(u)\ .$$

From (1.4) in section 2.1, we assume that ϕ possesses a cross-boundary derivative of the form

$$[D_1\phi](u)=n\sum_{i=0}^{n-1}(R_i-\widetilde{S}_i)B_i^{n-1}(u)$$

and φ possesses a cross-boundary derivative of the form

$$[D_2\varphi](u)=n\sum_{i=0}^{n-1}(T_i-\widetilde{S}_i)B_i^{n-1}(u)\ .$$

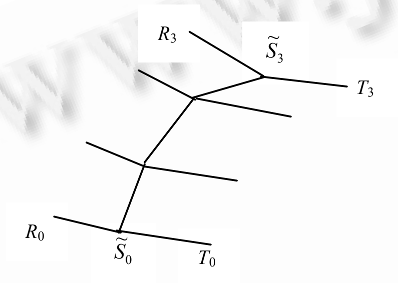

Fig.1 Coefficients for cross-boundary derivatives

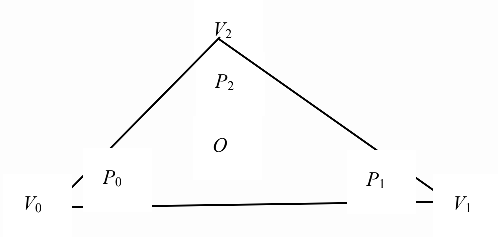

Fig.2 Corner points corresponding to $\Delta V_0V_1V_2$

The G^1 continuous condition [9] is equivalent to

$$\mu(u)[D_1\phi](u)+\alpha(u)[D_2\varphi](u)+\lambda(u)[D\Gamma](u)=0,\qquad \mu,\alpha,\lambda\neq 0\ .$$

In order to arrive at a manageable G^1 construction, we specify that μ and α must be constants while λ must be linear: $\lambda(u)=(1-u)\lambda_0+u\lambda_1$. Since $\alpha\neq 0$, we can assume without loss of generality that $\alpha=1$. This gives the desired G^1 continuous condition

$$T_i=\frac{n-1-i}{n-1}((1+\frac{n-1}{n}\lambda_0+\mu)\widetilde{S}_i-\frac{n-1}{n}\lambda_0\widetilde{S}_{i+1}-\mu R_i)+\frac{i}{n-1}(\frac{n-1}{n}\lambda_1\widetilde{S}_{i-1}+(1-\frac{n-1}{n}\lambda_1+\mu)\widetilde{S}_i-\mu R_i),i=0,\ldots,n-1\ . \qquad (1.5)$$

2 Constructing the Spline

Constructing the spline surface begins with a user-defined control mesh denoted *M*. A control mesh is a collection of vertices, edges, and triangular faces that can intuitively be thought of as a triangular surface that may, or may not, be closed. The term *valance* is used to denote the number of triangles meeting at vertex.

The spline surface is constructed in the following stages:

Input: *a control triangular mesh*

1. create corner points
2. construct edge and vertex coefficients

3. construct face coefficients
4. construct patches

Output: *a network of triangular patches*

The mesh M is passed to the first procedure that creates a set of corner points. The purpose of the first procedure provides initial data for following procedure. We require constructing a triangular Bèzier patch of degree 4 under each triangle on mesh , each boundary of whom is a Bèzier curve of degree 3. After the first step, the set of corner points is used to construct edge and vertex coefficients corresponding to each triangular patch in the second step. Using the edge, vertex coefficients constructed in the second step and the valant value at each vertex to solve a small linear system obtains face coefficients around each vertex. All of Bèzier coefficients controlling each triangular patch are constructed after three steps, a network of triangular Bèzier patches of degree 4 is generated and output. The details of each step are described in the next four sections.

2.1 Create corner points

Let $\Delta V_0V_1V_2$ be one of triangles in mesh M, denote it's barycenter by O, i.e. $O=\frac{1}{3}(V_0+V_1+V_2)$. Points $P_i, i=0,1,2$ are constructed by

$$P_i=(1-\alpha)V_i+\frac{1}{4}\alpha V_{i-1}+\frac{1}{4}\alpha V_{i+1}+\frac{1}{2}\alpha O,\qquad 0\le\alpha\le 1, \tag{2.1}$$

where all subscripts are taken modulo 3 (see Fig.2). We call $P_i, i=0,1,2$ corner point and α blend ratio or shape parameter.

2.2 Construct edge and vertex coefficients

In the second step, 6 edge coefficients and 3 vertex coefficients are constructed corresponding to each triangular patch to be constructed. The labeling scheme of these coefficients is illustrated in Fig.3. V_i and V_j in Fig.3 are two adjacent mesh points. Integer numbers n and m are the numbers of triangles meeting at V_i and V_j, respectively. $\{P_{il}\}_{l=0}^{n-1}$ are corner points constructed in the first step, and $\{E_{il}\}_{l=0}^{n-1}$ edge coefficients around V_i to be constructed. $\{P_{jl}\}_{l=0}^{m-1}$ and $\{E_{jl}\}_{l=0}^{m-1}$ are similar concept. The vertex coefficients at V_i and V_j are denoted by V_i' and V_j', respectively.

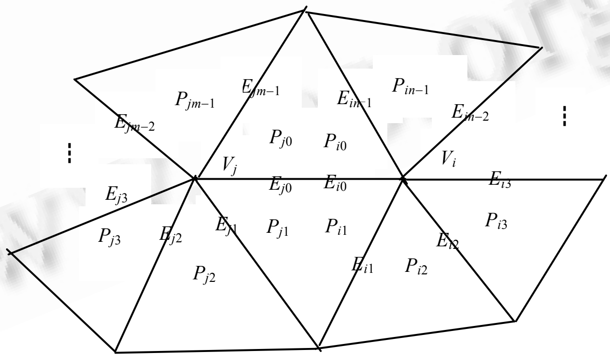

Fig.3 Corner points and edge coefficients around two adjacent mesh vertices V_i and V_j

Suppose that V_j' corresponds to parameter $u=0$ and V_i' to $u=1$ along the common boundary determined by V_j', E_{j0}, E_{i0} and V_i'. According to G^1 constraint (1.5), we have

$$-\lambda_0 E_{j0}+(1+\mu+\lambda_0)V_j'=E_{jm-1}+\mu E_{j1} \tag{2.2}$$

and

$$\lambda_1 E_{i0}+(1+\mu-\lambda_1)V_i'=E_{in-1}+\mu E_{i1}\,. \tag{2.3}$$

Taking

$$\lambda_0=-2\cos\frac{2\pi}{m},\lambda_1=2\cos\frac{2\pi}{n},\mu=1\,. \tag{2.4}$$

This is so that the condition at vertex takes the symmetric form that the sum of the derivatives along the edges meeting there is zero. The discussion on different choice of λ_0,λ_1 and μ is referred to Ref.[11]. Substituting (2.4) into (2.2) and (2.3), we obtain

$$(1-\cos\frac{2\pi}{m})V_j'+\cos\frac{2\pi}{m}E_{j0}=\frac{1}{2}E_{jm-1}+\frac{1}{2}E_{j1} \tag{2.5}$$

and

$$(1-\cos\frac{2\pi}{n})V_i'+\cos\frac{2\pi}{n}E_{i0}=\frac{1}{2}E_{in-1}+\frac{1}{2}E_{i1}\,. \tag{2.6}$$

In order to ensure that the spline surface is G^1, such constraint (2.5) or (2.6) must be satisfied between edge coefficients and vertex coefficients of a pair of adjacent triangular patches surrounding any vertex. Then these constraints at vertex V_i' are as follows:

$$(1-\cos\frac{2\pi}{n})V_i'+\cos\frac{2\pi}{n}E_{il}=\frac{1}{2}E_{il-1}+\frac{1}{2}E_{il+1},\qquad l=0,\dots,n-1\,, \tag{2.7}$$

where all subscripts are taken modulo n.

Above constraint (2.5) implies that all of edge coefficients surrounding V_i' must be co-planar. The following theorem is the key to construct the coefficients that satisfy this requirement:

Theorem 2.1. Let $P_0,\dots,P_{n-1}\in R^3$ be a set of points in general position. The set of points $Q_0,\dots,Q_{n-1}$ found by

$$Q_i=\frac{1}{n}\sum_{j=0}^{n-1}P_j(1+\beta(\cos\frac{2\pi(j-i)}{n}+\tan\frac{\pi}{n}\sin\frac{2\pi(j-i)}{n}))\,, \tag{2.8}$$

satisfy

$$(1-\cos\frac{2\pi}{n})O+\cos\frac{2\pi}{n}Q_i=\frac{1}{2}Q_{i-1}+\frac{1}{2}Q_{i+1}\,, \tag{2.9}$$

where $O=\frac{1}{n}\sum_{j=0}^{n-1}P_j$, and are therefore co-planar.

Consult [6] for the proof of Theorem 2.1. The factor β in equation (2.8) is a free parameter that may be set arbitrarily. Theorem 2.1 applies to the construction at hand by setting $\beta=\frac{3}{2}(1+\cos\frac{2\pi}{n})$, and interpreting the points $P_0,\dots,P_{n-1}$ as the corner points $\{P_{il}\}$ surrounding V_i', the point O as vertex coefficient V_i', and the points $Q_0,\dots,Q_{n-1}$ as the edge coefficients $\{E_{il}\}$ surrounding V_i'. Under this interpretation we immediately obtain the edge coefficients $\{E_{il}\}$ and vertex coefficient V_i' from $\{P_{il}\}$ and Theorem 2.1. Other edge coefficients and vertex coefficients at other vertices can be obtained using similar technique.

2.3 Construct face coefficients

Suppose that V_i' is a vertex coefficient constructed in step 2 and the vertex coefficients adjacent to V_i' are $V_0',\dots,V_{n-1}'$, respectively. Denote the number of triangles meeting at $V_l, l=0,\dots,n-1$ by n_l, the edge coefficients constructed in step two on each common boundary $V_l'\,V_i', l=0,\dots,n-1$ by E_l and E_{il}, $l=0,\dots,n-1$ (see Fig.4).

Because we require that each triangular patch is a polynomial of degree 4, there exist three face coefficients corresponding to each triangular patch. Denote the face coefficients next to V_i' by F_{il}, $l=0,\dots,n-1$ (see Fig.4). From constraint (1.5), in order to ensure two adjacent patches to be G^1, we require that the edge, face and vertex coefficients must satisfy following constrain

$$(4-2\cos\frac{2\pi}{n}-\cos\frac{2\pi}{n_l})E_{il}+2\cos\frac{2\pi}{n}E_l+\cos\frac{2\pi}{n_l}V_i' = 2(F_{il}+F_{il+1}), \quad l=0,\dots,n-1. \tag{2.10}$$

Set

$$S_l=\frac{1}{2}((4-2\cos\frac{2\pi}{n}-\cos\frac{2\pi}{n_l})E_{il}+2\cos\frac{2\pi}{n}E_l+\cos\frac{2\pi}{n_l}V_i'), \quad l=0,\dots,n-1.$$

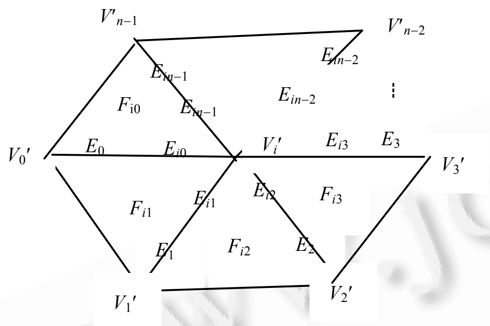

Fig.4 The vertex coefficients adjacent to V_i' and edge coefficients on each edge meeting at V_i'

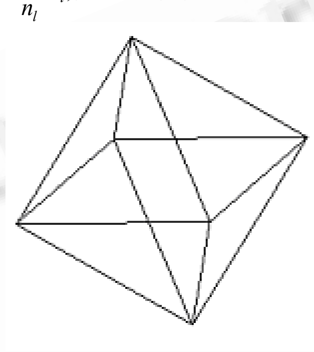

Fig.5 A close initial control triangular mesh

In order to obtain face coefficients $\{F_{il}\}_{l=0}^{n-1}$, we need only solve such a small system of equations

$$AF=S, \tag{2.11}$$

where

$$A=\begin{pmatrix}1&1&0&0&\cdots&0&0\\0&1&1&0&\cdots&0&0\\&&\cdots&\cdots&\cdots&&\\0&0&0&0&\cdots&1&1\\1&0&0&0&\cdots&0&1\end{pmatrix},\quad F=\begin{pmatrix}F_{i0}\\F_{i1}\\\vdots\\F_{in-2}\\F_{in-1}\end{pmatrix},\quad S=\begin{pmatrix}S_0\\S_1\\\vdots\\S_{n-2}\\S_{n-1}\end{pmatrix}.$$

We verify easily that the rank of A is n if n is odd, otherwise is $n-1$. When n is even, the freedom of (2.11) is 1, first we may take

$$F_{i0}=\alpha O_0+(1-\alpha)\frac{1}{2}(E_{i0}+E_{in-1}), \quad 0\le\alpha\le 1, \tag{2.12}$$

where O_0 is the barycenter of triangle $V_0V_iV_{n-1}$, then other face coefficients $F_{il}, l=1,\dots,n-1$ are given by (2.11). When n is odd, all of face coefficients F_{il} is just determined by (2.11).

Remarks. If some V_l lies on boundary of open mesh, then n_l in (2.10) is taken as double of the number of triangles meeting at V_l. The reason is found in Section 3.

2.4 Construct patches

After the third step is finished, all of the Bézier coefficients of each triangular patch are determined, we can output the spline surface. However, the edge, face and vertex coefficients determined in previous three steps are those around interior vertices. Those coefficients around boundary vertices of an open mesh are treated in Section 3.

3 Treatment of Boundaries

3.1 Boundary vertex with 1-valant value

Let V_0 be a 1-valant boundary vertex of mesh M. Let V_1 and V_2 be two vertices adjacent to V_0. Let P_0 be a corner point corresponding to V_0. Two edge coefficients next to V_0 are given by

$$E_{00}=(1+\alpha)V_0+\alpha V_1-2\alpha P_0,\qquad E_{01}=(1+\alpha)V_0+\alpha V_2-2\alpha P_0. \tag{3.1}$$

The vertex coefficient at V_0 is defined as $V_0'=V_0$ (interpolation) or

$$V_0'=(1+3\alpha)V_0-\alpha E_{00}-\alpha E_{01}-3\alpha P_0\,.$$

The face coefficient is determined by (2.12).

3.2 Boundary vertex with valant value more than 2

Let V be a vertex on the boundary of M. Let $k(k\geq 1)$ be the number of triangles meeting at V. Let $P_1,\ldots,P_k$ be the corner points next to V constructed in step 1, and let P_0 and P_{k+1} be the other two points next to V determined by (3.1). A new $n=2k$ corner points $\{P_0,\ldots,P_{n-1}\}$ is given by

$$P_l=2(uQ_0+(1-u)Q_1)-P_{n-l+1},\qquad l=k+2,\ldots,n-1\,,$$

where

$$Q_0=\frac{1}{2}P_0+\frac{1}{2}P_1,\quad Q_1=\frac{1}{2}P_k+\frac{1}{2}P_{k+1},\quad u=\frac{1}{2}\left(1+\cos\frac{2l\pi}{n}+\tan\frac{\pi}{n}\sin\frac{2l\pi}{n}\right).$$

Then we can construct edge coefficients $\{E_l\}_{l=0}^{n-1}$ and vertex coefficient V' using $\{P_l\}_{l=0}^{n-1}$ by Theorem 2.1. However, only $\{E_l\}_{l=0}^{k}$ are required. For face coefficients $\{F_l\}_{l=0}^{k-1}$, we first define F_0 by (2.12), then solve $\{F_l\}_{l=1}^{k-1}$ using (2.11) step by step.

4 Conclusions

An algorithm has been presented for constructing a tangent plane smooth spline surface that approximates an control triangular mesh of arbitrary topological type. The spline surface is a composite of quartic triangular Bézier patches. The algorithm is simple, efficient and generates aesthetically pleasing shapes. We may obtain any approximation to mesh by adjusting the value of blend ratio α. When $\alpha=0$, surface interpolates the mesh.

The spline algorithm as presented was factored into 4 steps. Each of previous three steps was a construction that involved taken weighted averages (affine combinations) of points. Therefore, the spline surface is affine invariant (i.e., independent of any affine transformation applied to the control mesh). It is not clear that the concatenation of the constructions leads to convex combinations in all cases, but according to examination, when $0\leq\alpha\leq 0.7$, the resulting surface preserves convexity.

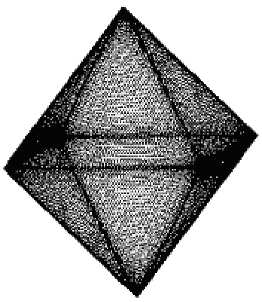

Fig.6 Interpolate mesh in $\alpha=0$

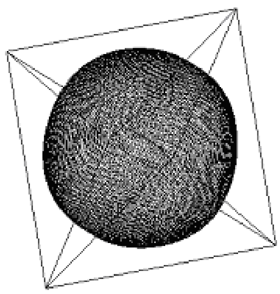

Fig.7 Spline sueface in $\alpha=0.5$

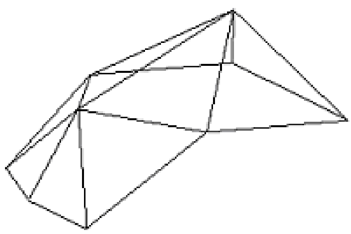

Fig.8 An open control initial mesh

Example. A close control mesh and spline surfaces generated are shown in Figs.5~7; the other open control mesh and spline surfaces are shown in Figs.8~10.

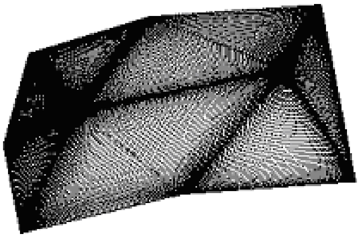

Fig.9 Interpolate mesh in $\alpha = 0$

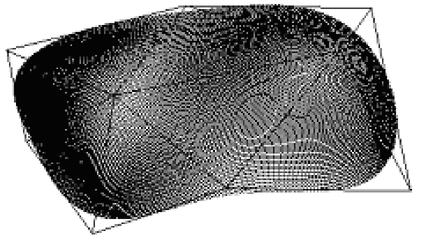

Fig.10 Spline surface in $\alpha = 0.5$

References:

[1] Goodman TNT. Closed surface defined from biquadratic splines. Constructive Approximation, 1991,7(2):149~160.

[2] HÖLLIG K, MÖGERLE H. *G*-Splines. Computer Aided Georemetric Design, 1990,7(1):197~207.

[3] Zeifang R. Local bases of G^2 continuous *G*-Splines. Computer Aided Geometric Design, 1999,16(1):1~20.

[4] Peters J. C^1-Surface splines. SIAM Journal of Numerical Analysis, 1995,32(2):645~666.

[5] Reif U. Biquadratic *G*-spline surfaces. Computer Aided Geometric Design, 1995,12(2):193~205.

[6] Peters J. Biquartic C^1-Surface splines over irregular meshes. Computer-Aided Design, 1995,27(12):895~903.

[7] Loop C. Smooth spline surfaces over irregular meshes. In: Proceedings of the Computer Graphics. Annual Conference Series, 1994. 303~310.

[8] Hahmann S, Bonneau GP. Triangular G^1 interpolation by 4-splitting domain triangles. Computer Aided Geometric Design, 2000,17(8):731~757.

[9] Farin G. Note: A construction for visual C^1 continuity of polynomial surface patches. Computer Graphics and Image Proceeding, 1982,20(3):272~282.

[10] Liu D, Hoschek J. GC^1 continuity conditions between adjacent rectangular and triangular Bézier surface patches. Computer-Aided Design, 1989,21(1):194~200.

[11] Du WH, Schmitt FJM. On the G^1 continuity of piecewise Bézier surface: a review with new results. Computer-Aided Design, 1990,22(9):556~573.