Journal of Computational Mathematics, Vol.12, No.3, 1994, 195–202.

A FAST PARALLEL ALGORITHM OF BIVARIATE SPLINE SURFACES*1)

Sun Jia-chang
(*Computing Center, Academia Sinica, Beijing, China*)

Abstract

A fast algorithm for evaluating and displaying bivariate splines in a three direction is presented based on two-level transfomation of the corresponding B-splines. The efficiency has been shown by experiments of surface modelling design[5].

1. Introduction

It has been shown [1] that bivariate B-spline is a very useful tool for designing surface modelling. One main difficulty in practice, however, is to develop an efficient algorithm for evaluating and displaying the resulting surface. In fact, for a given partition Ω a bivariate spline in the space $S_k^\mu(\Omega)$ is a piecewise bivariate polynomial of total degree k with global continuity degree μ. It means that in each subdomain the surface can be represented as a Bernstein-Bezier form. By using well-known subdivision technique[2] one may give an algorithm for B-B surface in each subtriangle. However, the working amount along this way would still be very large. Because in three direction case there is no analogy of efficient recurrence like so called de Boor-Cox algorithm in univariate case, we have to find another way by using some B-spline properties. In this paper we present a fast algorithm for evaluating and displaying spline surface based on two-level transfomation of B-splines. Numerical tests show it is really very efficient.

2. Two-Level Basis in the Space $S_{3\nu}^{2\nu-1}$

For $\nu = 0$, S_0^{-1} is a space consisting of step functions. It is obvious that after halving the origin mesh the resulting step function is equal to the summation of the four step functions with fine mesh (Fig. 1), i.e.

$$B^0(P,Q;2h) = B^0(P,Q;h) + B^0(P,Q_1;h) + B^0(P,Q_2;h) + B^0(P,Q_3;h).$$

It can be rewritten in terms of shiftor operators

$$B^0(P,Q;2h) = (I + E_1 + E_2 + E_3)(\Delta)B^0(P,Q;h)$$

* Received December 29, 1992.
1) The Project Supported by National Natural Science Fundation of China.

where

$$E_1(\Delta)B^0(P,Q;h) = B^0(P,Q_1;h)$$

etc.

To find the transformation between a finer and a coarser grids in the general spline space of $S_{3\nu}^{2\nu-1}(\nu \geq 1)$, we need to apply the following result [3]:

Theorem 1. *Suppose a piecewise polynomial surface $B^n(P,Q)$ is a B-spline in space $S_{n,\Delta}^{\nu}$ and another piecewise polynomial surface $B^{n+3}(P,Q)$ is defined by the following derivative-difference relation in three directions for each $\Omega_{\lambda\mu\nu}$*

$$D^3_{e_1e_2e_3}B^{n+3}_{rst}(\Omega_{\lambda\mu\nu}) = D_3(\Delta)B^n_{r-1,s-1,t-1}(\Omega_{\lambda\mu\nu})$$

if $rst \neq 0$,

$$D^3_{e_1e_2e_3}B^{n+3}_{rst} = 0$$

if $rst = 0$, where $r+s+t = n+3$

$$D_3 = (I - E_1E_2^{-1})(I - E_2E_3^{-1})(I - E_3E_1^{-1})$$

then $B^{n+3}(P,Q)$ must be a B-spline with the same parameter Q in the space $S^{\nu+2}_{n+3,\Delta}$. Furthermore, there is a difference recurrence in terms of B-nets for each pair of piecewise polynomial surfaces $B^n(P,Q)$ and $B^{n+3}(P,Q)$ [1].

Note that

$$D_3(\Delta_{2h}) = D_3(\Delta_h)G_3(\Delta_h)$$

where

$$G_3 = (I + E_1E_2^{-1})(I + E_2E_3^{-1})(I + E_3E_1^{-1}),$$

$$D_3(\Delta_{2h})B^0(P,Q;2h) = D_3(\Delta_{2h})(I + E_1 + E_2 + E_3)(\Delta_h)B^0(P,Q;h)$$

$$= D_3(\Delta_h)G_3(\Delta_h)B^0(P,Q;h).$$

By communicativity of these operators, hence, we have

$$D^3_{e_1e_2e_3}B^3(P,Q;\Delta_{2h}) = D_3(\Delta_{2h})B^0(P,Q;\Delta_{2h}) = D_3(\Delta_h)G_3(\Delta_h)B^0(P,Q;h)$$

$$= G_3(\Delta_h)D_3(\Delta_h)B^0(P,Q;h) = G_3(\Delta_h)D^3_{e_1e_2e_3}B^3(P,Q;\Delta_h)$$

$$= D^3_{e_1e_2e_3}\{G_3(\Delta_h)B^3(P,Q;\Delta_h)\}.$$

Because the both functions of the above equation under the differentiation in three directions have the same compact support, therefore we have

Theorem 2.

$$B^3(P,Q;\Delta_{2h}) = S_1(\Delta_h)B^3(P,Q;\Delta_h),$$

$$S_1 = G_3(I + E_1 + E_2 + E_3).$$

Example 1. In the space of S_3^1 the operator S_1 contains the following 19 terms:

$$S_1 = 2I + 4(E_1 + E_2 + E_3) + 2(E_1E_2^{-1}E_3 + E_2E_3^{-1}E_1 + E_3E_1^{-1}E_2)$$

$$+ (E_1^2E_2^{-1} + E_1^2E_3^{-1} + E_2^2E_3^{-1} + E_2^2E_1^{-1} + E_3^2E_1^{-1} + E_3^2E_2^{-1})$$

$$+ (E_1E_2^{-1} + E_1E_3^{-1} + E_2E_3^{-1} + E_2E_1^{-1} + E_3E_1^{-1} + E_3E_2^{-1}).$$

In general we may extend the above result into a B-spline of spline space $S_{3\nu}^{2\nu-1}$

Theorem 3.

$$B_{3\nu}^{2\nu-1}(P,Q;\Delta_{2h}) = S_\nu(\Delta_h)B_{3\nu}^{2\nu-1}(P,Q;\Delta_h),$$

where projector

$$S_\nu(\Delta_h) = \{G_3(\Delta_h)\}^\nu(I + E_1 + E_2 + E_3),$$

$$G_3 = (I + E_1E_2^{-1})(I + E_2E_3^{-1})(I + E_3E_1^{-1}).$$

Proof. The conclusion is obvious valid for $\nu = 0$. Suppose by induction that it is true for $\nu = k-1$, hence, from Thereom 1

$$D_{e_1e_2e_3}^3 B_{3k}^{2k-1}(P,Q;\Delta_{2h}) = D_3(\Delta_{2h})B_{3k-3}^{2k-3}(P,Q;\Delta_{2h})$$

$$= D_3(\Delta_h)G_3(\Delta_h)S_{k-1}(\Delta_h)B_{3k-3}^{2k-3}(P,Q;\Delta_h)$$

$$= G_3(\Delta_h)S_{k-1}(\Delta_h)D_3(\Delta_h)B_{3k-3}^{2k-3}(P,Q;\Delta_h)$$

$$= G_3(\Delta_h)S_{k-1}(\Delta_h)D_{e_1e_2e_3}^3 B_{3k}^{2k-1}(P,Q;\Delta_h)$$

$$= D_{e_1e_2e_3}^3 G_3(\Delta_h)S_{k-1}(\Delta_h)B_{3k}^{2k-1}(P,Q;\Delta_h).$$

By using compact support property we complete the proof.

Example 2. In the space of S_6^3 the operator S_2 contains the following 46 non-zero terms:

$$S_2 = G_3S_1 = (I + E_1E_2^{-1})^2(I + E_2E_3^{-1})^2(I + E_3E_1^{-1})^2(I + E_1 + E_2 + E_3)$$

$$= 10I + 6(E_1E_2^{-1} + E_1E_3^{-1} + E_2E_3^{-1} + E_2E_1^{-1} + E_3E_1^{-1} + E_3E_2^{-1})$$

$$+ 2(E_1^2E_2^{-1}E_3^{-1} + E_2^2E_3^{-1}E_1^{-1} + E_3^2E_1^{-1}E_2^{-1} + E_1^{-2}E_2E_3 + E_2^{-2}E_3E_1 + E_3^{-2}E_1E_2)$$

$$+ (E_1^2E_2^{-2} + E_1^2E_3^{-2} + E_2^2E_3^{-2} + E_2^2E_1^{-2} + E_3^2E_1^{-2} + E_3^2E_2^{-2})\}$$

$$+ \{22(E_1 + E_2 + E_3) + 14(E_1E_2^{-1}E_3 + E_2E_3^{-1}E_1 + E_3E_1^{-1}E_2)$$

$$+ 9(E_1^2E_2^{-1} + E_1^2E_3^{-1} + E_2^2E_3^{-1} + E_2^2E_1^{-1} + E_3^2E_1^{-1} + E_3^2E_2^{-1})$$

$$+ 3(E_1^2E_2^{-2}E_3 + E_2^2E_3^{-2}E_1 + E_3^2E_1^{-2}E_2 + E_1^{-2}E_2^2E_3 + E_2^{-2}E_3^2E_1 + E_3^{-2}E_1^2E_2)$$

$$+ 2(E_1^3E_2^{-1}E_3^{-1} + E_2^3E_3^{-1}E_1^{-1} + E_3^3E_1^{-1}E_2^{-1})$$

$$+ (E_1^3E_2^{-2} + E_1^3E_3^{-2} + E_2^3E_3^{-2} + E_2^3E_1^{-2} + E_3^3E_1^{-2} + E_3^3E_2^{-2})\}.$$

3. Two-Level Basis in the Space $S_{3\nu+1}^{2\nu}$

For $\nu = 0$, S_1^0 is a linear space. It is obvious that after halving the original mesh the resulting linear function is a linear combination of the following seven linear function with fine mesh (Fig. 2), i.e.

$$B^1(P,Q;2h) = B^1(P,Q;h) + \frac{1}{2}(B^1(P,Q_1;h) + B^1(P,Q_2;h) + B^1(P,Q_3;h)$$
$$+ B^1(P,Q_4;h) + B^0(P,Q_5;h) + B^0(P,Q_6;h)).$$

It can be rewritten in terms of shiftor operators

$$B^1(P,Q;2h) = \frac{1}{2}G_3(\Delta_h)(\Delta)B^1(P,Q;h).$$

Similarly, we obtain the following relationship between two level B-splines in the space $S^{2\nu}_{3\nu+1}$.

Theorem 4.

$$B^{2\nu}_{3\nu+1}(P,Q;\Delta_{2h}) = \hat{S}_\nu(\Delta_h)B^{2\nu}_{3\nu+1}(P,Q;\Delta_h)$$

where

$$\hat{S}_\nu(\Delta_h) = \frac{1}{2}\{G_3(\Delta_h)\}^{\nu+1}$$

or

$$\hat{S}_\nu = G_3\hat{S}_{\nu-1}.$$

Example 3. In the space of S^2_4 the two-level operator contains the following 19 non-zero terms:

$$G_3^2 = (I + E_1E_2^{-1})^2(I + E_2E_3^{-1})^2(I + E_3E_1^{-1})^2$$
$$= 10I + 6(E_1E_2^{-1} + E_1E_3^{-1} + E_2E_3^{-1} + E_2E_1^{-1} + E_3E_1^{-1} + E_3E_2^{-1})$$
$$+ 2(E_1^2E_2^{-1}E_3^{-1} + E_2^2E_3^{-1}E_1^{-1} + E_3^2E_1^{-1}E_2^{-1} + E_1^{-2}E_2E_3 + E_2^{-2}E_3E_1 + E_3^{-2}E_1E_2)$$
$$+ (E_1^2E_2^{-2} + E_1^2E_3^{-2} + E_2^2E_3^{-2} + E_2^2E_1^{-2} + E_3^2E_1^{-2} + E_3^2E_2^{-2})\}.$$

4. A Parallel Subdivision Algorithm of Bivariate Spline Surfaces

Suppose we have a representation of a bivariate spline in a three direction in terms of a linear combination of B-splines with patition Δ_h as follows

$$S(P) = \sum_Q C(Q)B(P,Q;\Delta_h).$$

What we concern is how to generate and display the surface as fast as possible. A fast display algorithm is very important for numerical simulation. Given a partition in a three direction, by introducing a half mesh we substitute the above two-level relation into the representation of the surface

$$S(P) = \sum_Q C(Q)S(\Delta_{\frac{h}{2}})B(P,Q;\Delta_{\frac{h}{2}})$$

or

$$S(P) = \sum_{Q} C^{[1]}(Q) B(P, Q; \Delta_{\frac{h}{2}}).$$

Repeating the procedure leads to the following formulas

$$S(P) = \sum_{Q} C^{[m]}(Q) B(P, Q; \Delta_{2^{-m}h}).$$

For display purpose it is sufficient to demand $m \le 9$ because general speaking the resolution of a monitor is only 1024×1024.

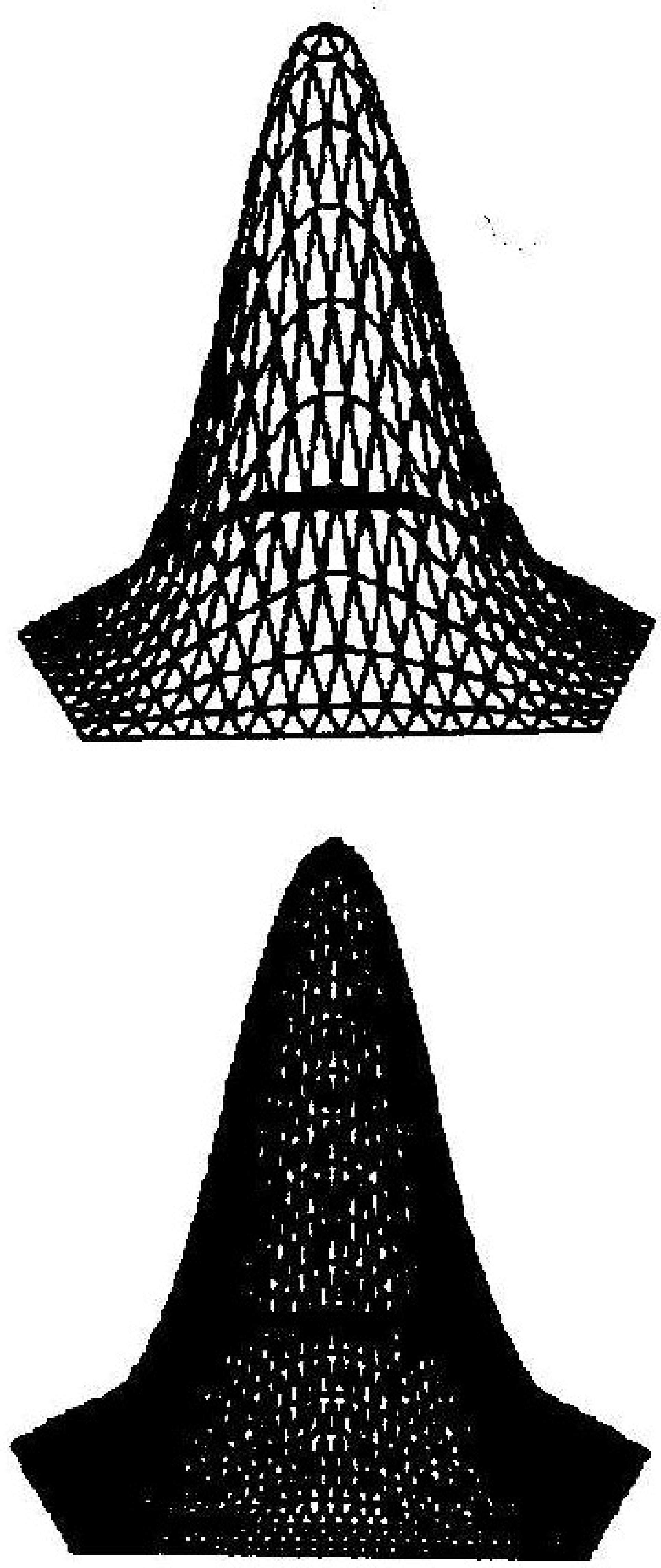

Fig. 1.1

The main idea of our fast algorithm is that for small mesh side h one may approximate the higher degree spline surface by piecewice linear function with second order precision. Comparing with the standard Schoenberg approximation[4] we call this technique as "Inverse Schoenberg Approximation". The key work should be done is to find a recurrence relation between coefficients $C^{[m]}$ and $C^{[m-1]}$ which can be derived from the two-level relation of B-splines.

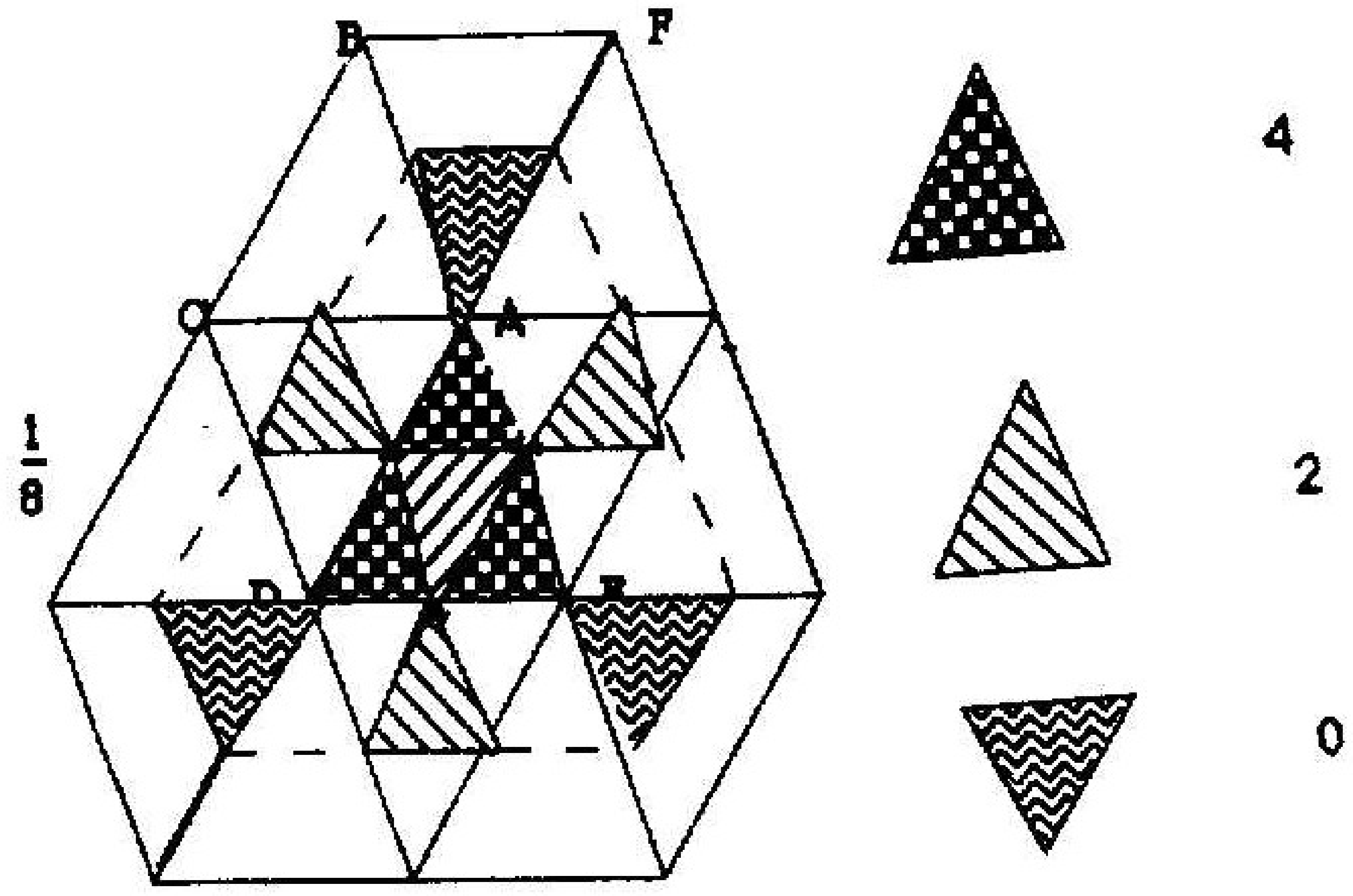

Fig. 1.2

For instance, in the spline space S^1_3 a fast algorithm is as follows (Fig. 1)

$$S(P)=\sum_Q C(Q)S(\Delta_{\frac{h}{2}})B(P,Q;\Delta_{\frac{h}{2}})=\sum_Q C^{[1]}(Q)B(P,Q;\Delta_{\frac{h}{2}}).$$

The formulas evaluating coefficients $C^{[1]}$ is divived into two parts:

$$C^{[1]}_I(Q)=\frac{1}{4}(I+E_1^{-1}+E_2^{-1}+E_3^{-1})C_I(Q),$$

$$C^{[1]}_{II,1}(Q)=\frac{1}{8}(4I+E_1E_2^{-1}+E_1E_3^{-1}+E_2^{-1}+E_3^{-1})C_I(Q),$$

$$C^{[1]}_{II,2}(Q)=\frac{1}{8}(4I+E_2E_3^{-1}+E_2E_1^{-1}+E_3^{-1}+E_1^{-1})C_I(Q),$$

$$C^{[1]}_{II,3}(Q)=\frac{1}{8}(4I+E_3E_1^{-1}+E_3E_2^{-1}+E_1^{-1}+E_2^{-1})C_I(Q).$$

Moreover, in the spline space S^2_4 the fast algorim also consists of two parts (Fig. 2):

At integer points

$$C^{[1]}_I=\frac{1}{16}(10I+E_1E_2^{-1}+E_1E_3^{-1}+E_2E_3^{-1}+E_2E_1^{-1}+E_3E_1^{-1}+E_3E_2^{-1})C(Q).$$

At mid-points

$$C^{[1]}_{II,1}=\frac{1}{8}(3I+3E_2E_3^{-1}+E_1E_3^{-1}+E_1^{-1}E_2)C(Q),$$

$$C^{[1]}_{II,2}=\frac{1}{8}(3I+3E_1E_3^{-1}+E_2E_3^{-1}+E_2^{-1}E_1)C(Q),$$

$$C^{[1]}_{II,3}=\frac{1}{8}(3I+3E_1E_2^{-1}+E_3E_2^{-1}+E_3^{-1}E_1)C(Q),$$

$$C^{[1]}_{II,4}=\frac{1}{8}(3I+3E_3E_2^{-1}+E_1E_2^{-1}+E_1^{-1}E_3)C(Q),$$

$$C^{[1]}_{II,5} = \frac{1}{8}(3I + 3E_3E_1^{-1} + E_2E_1^{-1} + E_2^{-1}E_3)C(Q),$$

$$C^{[1]}_{II,6} = \frac{1}{8}(3I + 3E_2E_1^{-1} + E_3E_1^{-1} + E_3^{-1}E_2)C(Q).$$

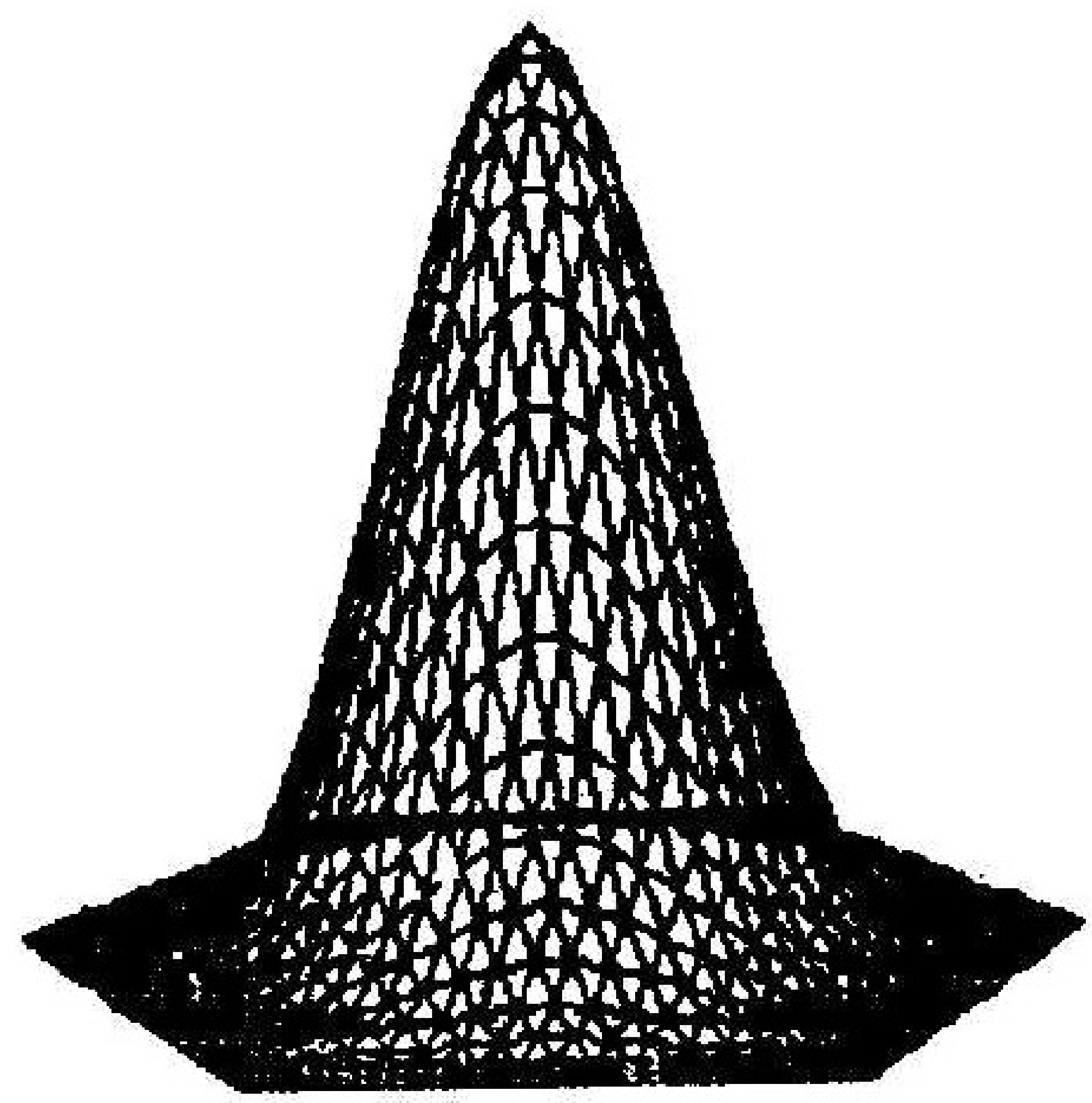

Fig. 2

In general we may construct a corresponding subdivision algorithm for evaluating and displaying spline surface in the space $S^{2\nu-1}_{3\nu}$ and $S^{2\nu}_{3\nu+1}$ for any integer ν. As another example we may present the following fast algorithm for bivariate spline in the space S^3_6.

At the center

$$\begin{aligned} C^{[1]}_{I}(Q) = \frac{1}{64}\{&10I + 14(E_1^{-1} + E_2^{-1} + E_3^{-1})\} \\ &+ (E_1E_2^{-1} + E_1E_3^{-1} + E_2E_3^{-1} + E_2E_1^{-1} + E_3E_1^{-1} + E_3E_2^{-1}) \\ &+ 2(E_1E_2^{-1}E_3^{-1} + E_2E_3^{-1}E_1^{-1} + E_3E_1^{-1}E_2^{-1})\}C(Q). \end{aligned}$$

At the neighbours

$$\begin{aligned} C^{[1]}_{II}(Q) = \frac{1}{64}\{&22I + 6(E_2^{-1} + E_3^{-1}) + 9(E_1E_3^{-1} + E_1E_2^{-1}) + 3(E_3E_2^{-1} + E_2E_3^{-1}) \\ &+ 2(E_1E_2^{-1}E_3^{-1} + E_1^{-1}) + (E_2E_1^{-1} + E_3E_1^{-1})\}C(Q), \end{aligned}$$

$$\begin{aligned} C^{[2]}_{II}(Q) = \frac{1}{64}\{&22I + 6(E_3^{-1} + E_1^{-1}) + 9(E_2E_1^{-1} + E_2E_3^{-1}) + 3(E_3E_1^{-1} + E_1E_3^{-1}) \\ &+ 2(E_2E_3^{-1}E_1^{-1} + E_2^{-1}) + (E_3E_2^{-1} + E_1E_2^{-1})\}C(Q), \end{aligned}$$

$$\begin{aligned} C^{[3]}_{II}(Q) = \frac{1}{64}\{&22I + 6(E_1^{-1} + E_2^{-1}) + 9(E_3E_2^{-1} + E_3E_1^{-1}) + 3(E_1E_2^{-1} + E_2E_1^{-1}) \\ &+ 2(E_3E_1^{-1}E_2^{-1} + E_3^{-1}) + (E_1E_3^{-1} + E_2E_3^{-1})\}C(Q). \end{aligned}$$

References

[1] Sun Jia-chang, The B-net approach and application of bivariate B-spline, University of Minnesota Supercomputer Institute, 88/139, Dec., 1988.

[2] G. Farin, Triangular Bernstei-Bezier patches, *CAGD*, 3(1986), 83–177.

[3] Sun Jia-chang, The B-net structure and recurrence algorithms for B-splines in three directions, *Chinese Journal of Numerical Mathematics and Applications*, 13 (1991), 48–59.

[4] I. J. Schoenberg, On Spline Functions, Inequalities (O. Shisha ed.), Academic Press, New York, 1967, 255-291.

[5] Deng Guo-shun and Sun Jia-chang, An implementation of the fast subdivision algorithm of splines in $S_3^1(\Omega)$ and $S_4^2(\Omega)$, to appear.