

Chapter 2

Splines and Subdivision

Most subdivision schemes are derived from refinement methods of control meshes of spline curves and surfaces. This chapter mainly discusses various refinement methods for control meshes of spline curves and surfaces. Firstly, definitions and basic properties of spline functions are introduced; secondly, refinement rules of spline functions are deduced based on their definitions and basic properties; lastly, refinement rules of control meshes of spline curves and surfaces are deduced based on refinement rules of spline functions. In next chapter, we will generalize these refinement rules of control meshes to arbitrary 2-manifold meshes to construct subdivision surfaces.

2.1 B-Splines

There are many definitions for B-splines. The recursive definition is the most commonly used one in computational field. Its discovery is attributed to de-Boor, Cox, and Mansfield [3, 4].

Definition 2.1 Given a non-decreasing sequence of the real u -axis:

$$U = [\cdots, u_{-2}, u_{-1}, u_0, u_1, u_2, \cdots], \text{ where } u_i \leq u_{i+1}.$$

B-splines (or B-spline basic functions) can be defined as:

$$\begin{cases} N_{i,0}(u) = \begin{cases} 1, & \text{if } u_i \leq u < u_{i+1} \\ 0, & \text{otherwise} \end{cases} \\ N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} N_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} N_{i+1,k-1}(u) \\ \text{assume } 0/0 = 0. \end{cases} \quad (2.1)$$

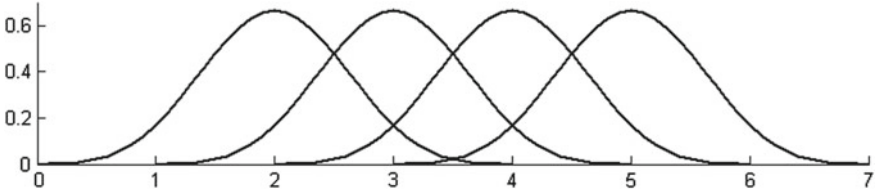


Fig. 2.1 Uniform cubic B-splines

In the above definition, the u_i are called knots and $\mathbf{U}$ is called the knot vector. The half-open interval, $[u_i, u_{i+1})$, is called the i -th knot span. $N_{i,k}(u)$ is the i -th B-spline basic function of k degree (order $k + 1$). From the recursive Formula (2.1), we can find:

- To define a set of B-spline basic functions, we should firstly define a knot vector $\mathbf{U}$, and the degree k .
- To compute $N_{i,k}(u)$, $k + 2$ knots, $u_i, u_{i+1}, \dots, u_{i+k+1}$, have to be used. $[u_i, u_{i+k+1}]$ is called the support span of $N_{i,k}(u)$. The first subscript of $N_{i,k}(u)$ is equal to the subscript of the left end knot of the support interval. The subscript of $N_{i,k}(u)$ indicates its position on the u -axis, as shown in Fig. 2.1.
- The knot vector $\mathbf{U} = [u_0, u_1, \dots, u_{n-k-1}, \dots, u_n]$ can define $n - k$ B-spline basic functions of k -degree: $N_{i,k}(u)$. The last one is $N_{n-k-1,k}(u)$.
- The knot span $[u_i, u_{i+1}]$ can have zero length, since the knots do not need to be distinct.
- The recursive Formula (2.1) can yield the quotient $0/0$, and we define this quotient as zero. If knots of $\mathbf{U}$ are uniformly distributed on the u -axis, i.e.,

$$u_{i+1} - u_i = \text{constant}, \quad i \in \mathbb{Z}.$$

We call $N_{i,k}(u)$ a uniform B-spline. Figure 2.1 gives the shapes of uniform cubic B-spline basic functions. From Formula (2.1), we can find that $\mathbf{U}$ and $t\mathbf{U}$ define the basic functions with same shapes.

We now investigate how to obtain the value or expression of $N_{i,k}(u)$ from Formula (2.1). Obviously,

$$N_{i,1}(u) = \frac{u - u_i}{u_{i+1} - u_i} N_{i,0}(u) + \frac{u_{i+2} - u}{u_{i+2} - u_{i+1}} N_{i+1,0}(u).$$

Based on $N_{i,0}(u) = \begin{cases} 1, & \text{if } u_i \leq u < u_{i+1} \\ 0, & \text{otherwise} \end{cases}$, we can get

$$N_{i,1} = \begin{cases} (u - u_i)/(u_{i+1} - u_i), & \text{if } u_i \leq u < u_{i+1} \\ (u_{i+2} - u)/(u_{i+2} - u_{i+1}), & \text{if } u_{i+1} \leq u < u_{i+2} \\ 0, & \text{otherwise.} \end{cases}$$

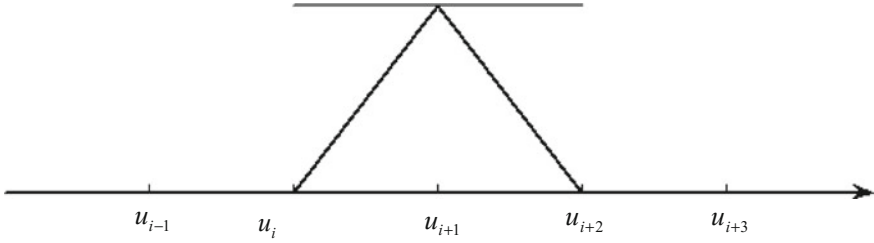


Fig. 2.2 Recursion from zero-degree B-spline to one-degree B-spline

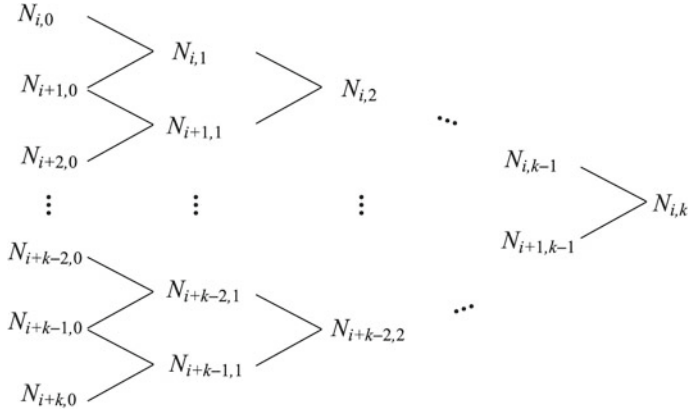


Fig. 2.3 Recursion of $N_{i,k}(u)$

Figure 2.2 shows the recursive process.

To obtain $N_{i,k}(u)$, we may apply the recursive method shown in Fig. 2.3.

From the recursive process of $N_{i,k}(u)$, we can find $N_{i,k}(u)$ is a piecewise polynomial function. It is defined on the whole real span, and

$$N_{i,k}(u) = \begin{cases} > 0 & u \in [u_i, u_{i+k+1}) \\ = 0 & u \notin [u_i, u_{i+k+1}) \end{cases} \quad (k \geq 0).$$

In the recursive process, when there are multiknots in the knot vector U , i.e.,

$$u_i = \cdots = u_{i+r-1} \quad (r > 0), \quad (2.2)$$

quotient $0/0$ may appear. We assume the quotient is zero. Multiknots affect the continuity degree of $N_{i,k}(u)$. For a knot u_i , if Eq. 2.2 exists, we call the knot u_i have the multiplicity r . The continuity degree of $N_{i,k}(u)$ is $k - r$ at the knot u_i . When u_0 has different multiplicities, the shapes of $N_{0,3}(u)$ are shown in Fig. 2.4.

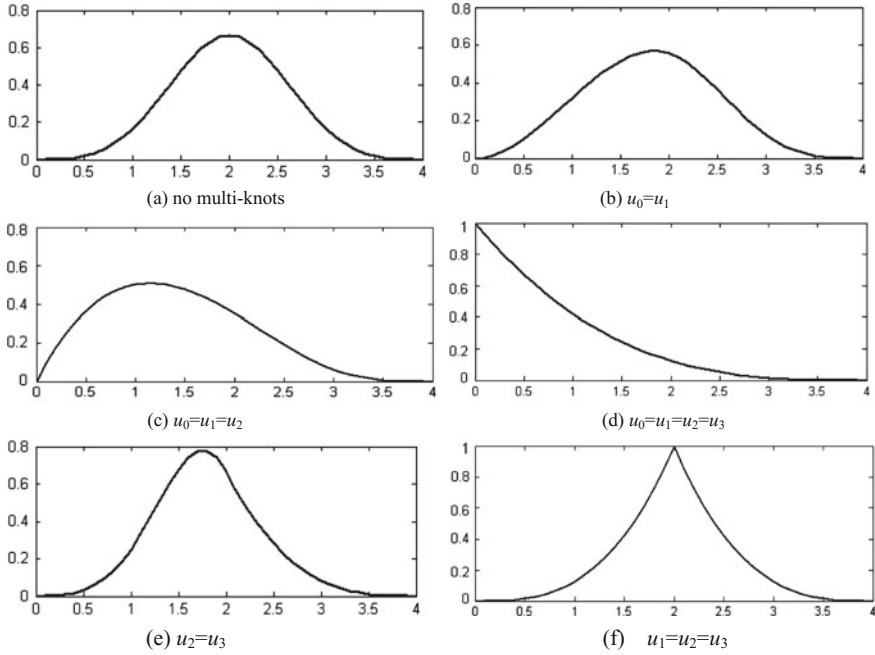


Fig. 2.4 Multiplicity of u_0 effects continuity degree of $N_{0,3}(u)$

From the above discussion, we know that B-splines have the following properties:

- (1) Recursion: $N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} N_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} N_{i+1,k-1}(u)$,
($k > 0$)
- (2) Positivity: $N_{i,k}(u) \geq 0$
- (3) Local support: $N_{i,k}(u) = 0, u \in (-\infty, u_i) \cup [u_{i+k+1}, +\infty)$
- (4) Partition of unity: $\sum_i N_{i,k}(u) = 1$
- (5) Continuity: $N_{i,k}(u)$ is $(k - r)$ times continuously differentiable, where r is the maximum multiplicity of its knots in the support span.

Let us investigate the property of partition of unity. Assume $u \in [u_i, u_{i+1})$, according to the local support property,

$$N_{j,k}(u) = 0 \text{ where } j = \dots, i - k - 2, i - k - 1; j = i + 1, i + 2, \dots$$

Consequently,

$$\sum_j N_{j,k}(u) = \sum_{j=i-k}^i N_{j,k}(u) = 1. \quad (2.3)$$

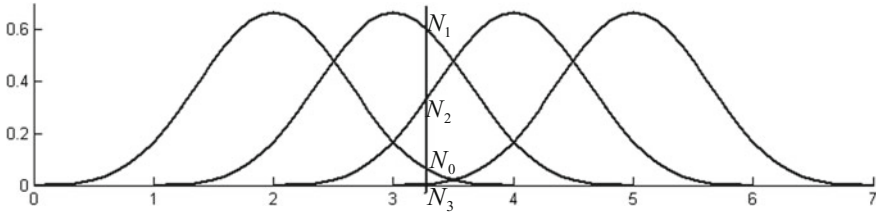


Fig. 2.5 Partition of unity

When $U = [u_0, u_1, \dots, u_{n-k-1}, \dots, u_n]$, there must be $k + 1$ nonzero curve segments in the interval $[u_i, u_{i+1})$. Only in this case can we have Eq. (2.3). The property will be explained in Fig. 2.5.

Except for the recursive definition, the convolution definition and the truncated power definition are always used in theory fields. Starting from the convolution definition, the subdivision rules of uniform B-splines are easy to be obtained. Uniform B-splines can be considered as simple examples of box splines when the convolution definition is used, which will be found in Sects. 2.3 and 2.4.

2.2 B-Spline Curves and Surfaces

2.2.1 B-Spline Curves and Their Properties

A B-spline curve can be defined as:

$$\mathbf{p}(u) = \sum_{i=0}^n \mathbf{P}_i N_{i,k}(u), \quad u \in [u_k, u_{n+1}], \quad (2.4)$$

where the knot vector $U = [u_0, u_1, \dots, u_{n+k+1}]$. $\mathbf{P}_i$ ($i = 0, 1, \dots, n$) are called control points of the B-spline curve $\mathbf{p}(u)$, and the polygon $\mathbf{P}_0 \mathbf{P}_1 \dots \mathbf{P}_n$ is called control polygon of the B-spline curve $\mathbf{p}(u)$. To make ends of the curve $\mathbf{p}(u_k)$ and $\mathbf{p}(u_{n+1})$ identical to ends of the control polygon, we need that:

$$u_0 = \dots = u_k, \quad u_{n+1} = \dots = u_{n+k+1}.$$

In this case,

$$\begin{aligned} \mathbf{p}(u_k) &= \mathbf{P}_0, \quad \mathbf{p}'(u_k) = k \frac{\mathbf{P}_1 - \mathbf{P}_0}{u_{k+1} - u_1}, \\ \mathbf{p}(u_{n+1}) &= \mathbf{P}_n, \quad \mathbf{p}'(u_k) = k \frac{\mathbf{P}_n - \mathbf{P}_{n-1}}{u_{n+k} - u_n}. \end{aligned}$$

Usually, we normalized U , i.e.,

$$u_0 = \cdots = u_k = 0, \quad u_{n+1} = \cdots = u_{n+k+1} = 1.$$

Figure 2.6 gives a uniform cubic B-spline curve and a quasi-uniform cubic B-spline curve. For k -degree quasi-uniform B-spline curve, only end knots have multiplicity $k + 1$ and other knots are uniformly distributed.

$p(u)$ is a segment polynomial curve since $N_{i,k}(u)$ is a segment polynomial. According to the local support property of $N_{i,k}(u)$, the curve segment

$$p_i(u) = \sum_{j=i-k}^i P_j N_{j,k}(u), \quad u \in [u_i, u_{i+1}) \quad (2.5)$$

is a polynomial curve segment and $p_{i+1}(u)$ is another, as shown in Fig. 2.6.

Based on the continuity property of $N_{i,k}(u)$, $p(u)$ is C^{k-1} continuous in a knot interval $[u_i, u_{i+1})$. If the multiplicity of knot u_i is r , $p(u)$ is C^{k-r} at the knot u_i .

Based on the local support property of $N_{i,k}(u)$, if the position of the control point P_i is altered, only $k + 1$ curve segments, $p_i(u)$, $p_{i+1}(u)$, $\dots$, $p_{i+k}(u)$, will change their shapes, as shown in Fig. 2.7.

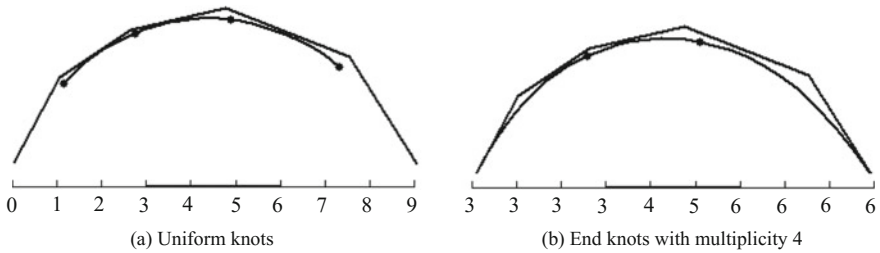


Fig. 2.6 Cubic B-spline curve

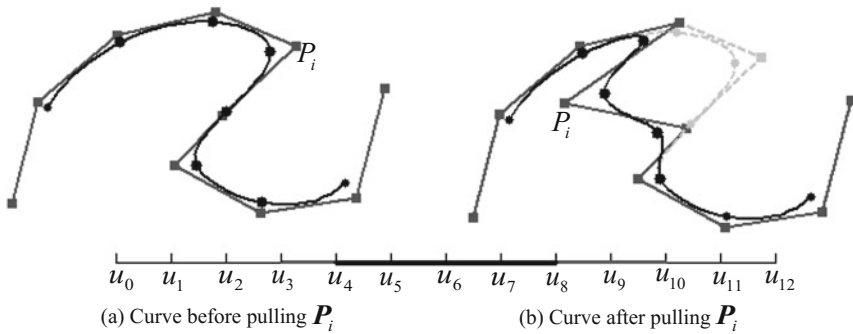


Fig. 2.7 Change of shape of cubic curve after control point P_i is pulled

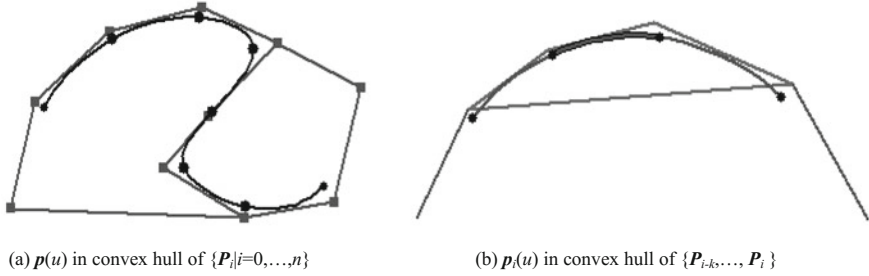


Fig. 2.8 Cubic curve in convex hull formed by its control point set

According to the positivity property and the local support property of $N_{i,k}(u)$, we obtain the convex hull property of B-spline curves, namely, $p(u)$ is in the convex hull formed by $\{P_i\}$, and $P_i(u)$ is in the convex hull formed by $\{P_{i-k}, \dots, P_i\}$, as shown in Fig. 2.8.

According to the recursive property of B-spline basic functions, we can obtain the de-Boor algorithm which calculates points on B-spline curves. For the curve (2.4), let $u \in [u_i, u_{i+1}] \subset [u_k, u_{n+1}]$, then,

$$p(u) = \sum_{j=i-k}^i P_j N_{j,k}(u) = \dots = \sum_{j=i-k}^{i-s} P_j^s N_{j+s,k-s}(u) = \dots = P_{j-k}^k, \quad (2.6)$$

where

$$P_j^s = \begin{cases} P_j, & s = 0 \\ (1 - \alpha_j^s) P_j^{s-1} + \alpha_j^s P_{j+1}^{s-1}, & s > 0 \end{cases} \quad (2.7)$$

$$j = i - k, \dots, i - s; s = 1, 2, \dots, k$$

$$\alpha_j^s = \frac{u - u_{j+s}}{u_{j+k+1} - u_{j+s}}$$

Assume $0/0 = 0$.

In fact,

$$\begin{aligned} p(u) &= \sum_{j=i-k}^i P_j N_{j,k}(u) = \sum_{j=i-k}^i P_j \left[\frac{u - u_j}{u_{j+k} - u_j} N_{j,k-1}(u) + \frac{u_{j+k+1} - u}{u_{j+k+1} - u_{j+1}} N_{j+1,k-1}(u) \right] \\ &= \sum_{j=i-k}^i P_j \frac{u - u_j}{u_{j+k} - u_j} N_{j,k-1}(u) + \sum_{j=i-k}^i P_j \frac{u_{j+k+1} - u}{u_{j+k+1} - u_{j+1}} N_{j+1,k-1}(u) \end{aligned}$$

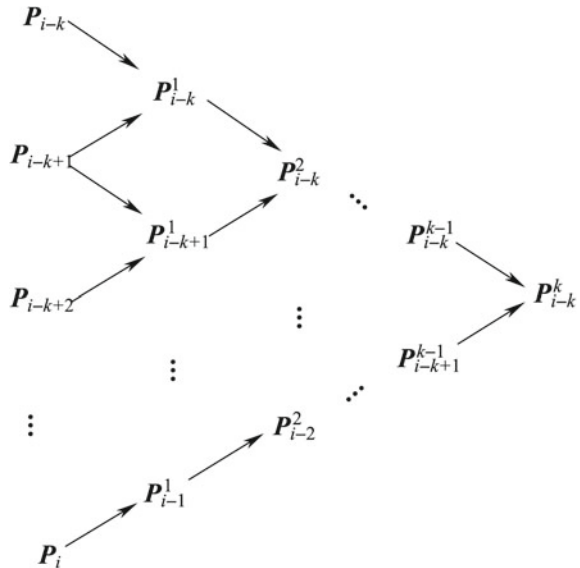
$$\begin{aligned}
&= \sum_{j=i-k-1}^{i-1} P_{j+1} \frac{u - u_{j+1}}{u_{j+k+1} - u_{j+1}} N_{j+1,k-1}(u) + \\
&\quad \sum_{j=i-k}^i P_j \frac{u_{j+k+1} - u}{u_{j+k+1} - u_{j+1}} N_{j+1,k-1}(u) (*).
\end{aligned}$$

Note that $u \in [u_i, u_{i+1}]$, and $N_{i-k,k-1}(u) = 0$, $N_{i+1,k-1}(u) = 0$. Consequently,

$$\begin{aligned}
(*) &= \sum_{j=i-k}^{i-1} \frac{u - u_{j+1}}{u_{j+k+1} - u_{j+1}} P_{j+1} N_{j+1,k-1}(u) + \\
&\quad \sum_{j=i-k}^{i-1} \frac{u_{j+k+1} - u}{u_{j+k+1} - u_{j+1}} P_j N_{j+1,k-1}(u) \\
&= \sum_{j=i-k}^{i-1} \left(\frac{u_{j+k+1} - u}{u_{j+k+1} - u_{j+1}} P_j + \frac{u - u_{j+1}}{u_{j+k+1} - u_{j+1}} P_{j+1} \right) N_{j+1,k-1}(u) \\
&= \sum_{j=i-k}^{i-1} P_j^1 N_{j+1,k-1}(u).
\end{aligned}$$

The recursive process expressed in Formulas (2.6) and (2.7) can be depicted by Fig. 2.9. Figure 2.10 gives the process of computing a point $p(u)$ on a cubic B-spline curve.

Fig. 2.9 Recursive process of de-Boor algorithm



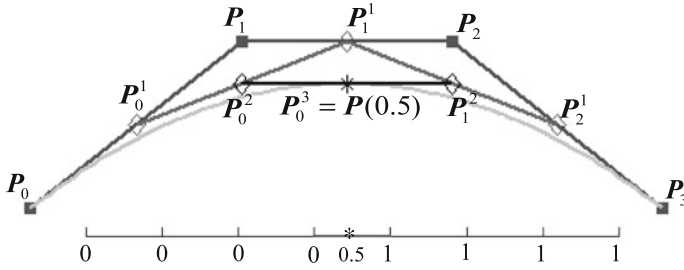


Fig. 2.10 Calculating a point on cubic B-spline curve with de-Boor algorithm

The knot insertion algorithm is an important technique of B-spline curves and surfaces. Its discovery is attributed to Boehm [3, 4]. For the curve expressed by Eq. (2.5), we insert a knot in its definition interval:

$$u \in [u_i, u_{i+1}] \subset [u_k, u_{n+1}].$$

So a new knot vector can be obtained:

$$\bar{U} = [u_0, u_1, \dots, u_i, u, u_{i+1}, \dots, u_{n+k+1}]$$

Readjusting numbers of knots in the knot vector, we have

$$\bar{U} = [\bar{u}_0, \bar{u}_1, \dots, \bar{u}_i, \bar{u}_{i+1}, \bar{u}_{i+2}, \dots, \bar{u}_{n+k+2}].$$

The new vector $\bar{U}$ defines a set of new B-spline basic functions: $\bar{N}_{j,k}(u)$, $j = 0, 1, \dots, n+1$. The B-spline curve can be re-expressed by the set of new B-spline basic functions and new control points $\bar{P}_j$, $j = 0, 1, \dots, n+1$:

$$p(u) = \sum_{j=0}^{n+1} \bar{P}_j \bar{N}_{j,k}(u), \quad u \in [\bar{u}_k, \bar{u}_{n+2}].$$

$\bar{P}_j$ ($j = 0, 1, \dots, n+1$) can be calculated by the following formulas:

$$\begin{aligned} \bar{P}_j &= P_j, \quad j = 0, 1, \dots, i-k \\ \bar{P}_{j+1} &= P_j^1 = (1 - \alpha_j^1)P_j + \alpha_j^1 P_{j+1}, \quad j = i-k, i-k+1, \dots, i-r-1 \\ \alpha_j^1 &= \frac{u - u_{j+1}}{u_{j+k+1} - u_{j+1}}, \quad \text{let } 0/0 = 0 \\ \bar{P}_j &= P_j, \quad j = i-r+1, \dots, n+1. \end{aligned}$$

In above formulas, r is the multiplicity of u in the original knot vector U . If $u_i < u < u_{i+1}$, $r = 0$. When $0 < r < k$, $u = u_i = u_{i-1} = \dots = u_{i-r+1}$. The nature of the algorithm is the first recursion of the de-Boor algorithm when the de-Boor

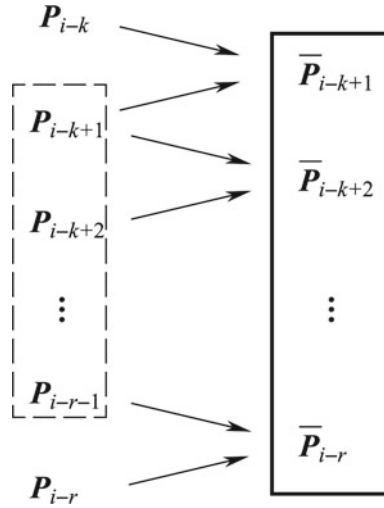
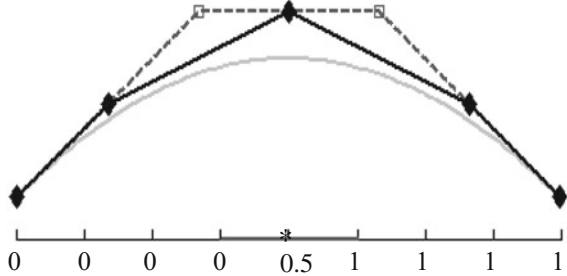


Fig. 2.11 Vertex replacement in knot insertion

Fig. 2.12 Example of knot insertion



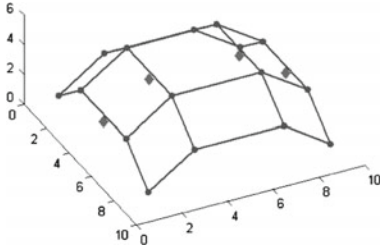
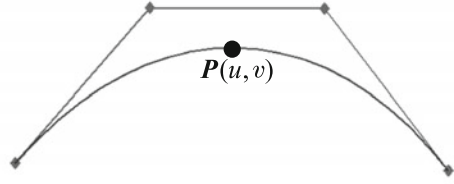
algorithm is applied to calculate a point $p(u)$ of a B-spline curve. Figure 2.11 gives the process of inserting a knot. After those new vertices $\bar{P}_{i-k+1}, \dots, \bar{P}_{i-r}$ replace old vertices $P_{i-k+1}, \dots, P_{i-r-1}$, a knot insertion is completed. Figure 2.12 gives an example of inserting a knot to a cubic B-spline curve.

2.2.2 B-Spline Surfaces and Their Properties

A B-spline surface can be represented as:

$$p(u, v) = \sum_{i=0}^m \sum_{j=0}^n P_{i,j} N_{i,k}(u) N_{j,l}(v), \quad (u, v) \in [u_k, u_{m+1}] \times [v_l, v_{n+1}], \quad (2.8)$$

where $U = [u_0, u_1, \dots, u_{m+k+1}]$, $V = [v_0, v_1, \dots, v_{n+l+1}]$, k and l are knot vectors and degrees of basic functions on the two parameter directions u, v , respectively.

(a) De-Boor algorithm on u parameter(b) De-Boor algorithm on v parameter**Fig. 2.13** Apply de-Boor algorithm on bi-cubic B-spline surface

B-spline surfaces are tensor surfaces, i.e., a B-spline surface can be constructed by applying operators φ, ψ on a point array formed by control points. The two operators are equally applied on u, v , the two parameter directions. Consequently,

$$\mathbf{p}(u, v) = (\psi \cdot \varphi) \mathbf{P} = [N_{0,k}, \dots, N_{m,k}] \begin{bmatrix} \mathbf{P}_{0,0} & \cdots & \mathbf{P}_{0,n} \\ \vdots & \ddots & \vdots \\ \mathbf{P}_{m,0} & \cdots & \mathbf{P}_{m,n} \end{bmatrix} \begin{bmatrix} N_{0,l} \\ \vdots \\ N_{n,l} \end{bmatrix}.$$

A main advantage of tensor surfaces is that many problems about tensor surfaces can be converted to problems about curves, which give great convenience to theory analysis and programming computation. With the property of tensor production, we can generalize the de-Boor algorithm and the knot insertion algorithm from B-spline curves to B-spline surfaces.

Let $(u, v) \in [u_i, u_{i+1}] \times [v_j, v_{j+1}]$ and pick a parameter direction, such as u . Now, we apply the de-Boor algorithm on every column control points. In fact, we easily find that we can only apply curve de-Boor algorithm on $l + 1$ columns: $j - l, \dots, j$. So, there will be $l + 1$ points. On the v parameter direction, after applying the de-Boor algorithm to the $l + 1$ points, we get the point $\mathbf{p}(u, v)$ we need. The calculating process is shown in Fig. 2.13.

For knot insertion, if we insert a knot u for the $\mathbf{U}$ knot vector, we need to apply the curve knot insertion algorithm to every column control points. So a new point array is formed, and it is the vertex array needed by us after inserting the knot u . The calculation is similar when we insert a knot v for the $\mathbf{V}$ knot vector.

2.3 Knot Insertion Algorithm and Refinement of B-Spline Curves and Surfaces

We have pointed out that refinement is subdivision in Chap. 1. Particularly, the subdivision of control meshes of spline surfaces is called refinement. In this chapter, we furthermore point out that the process of doubling knot intervals of B-spline basis

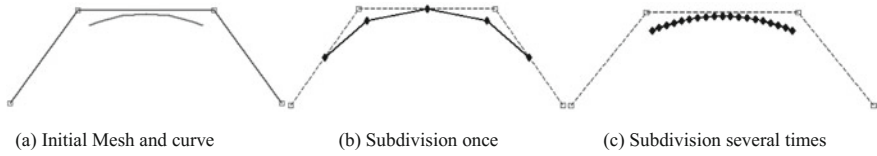


Fig. 2.14 Subdivision of cubic B-spline curve

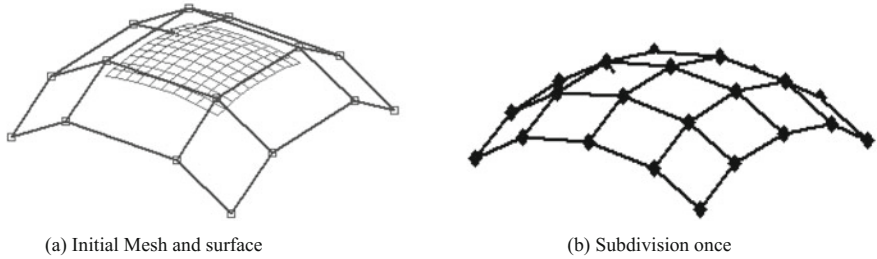


Fig. 2.15 Subdivision of cubic B-spline surface

functions and then obtaining new spline basis functions or new control polygons or control meshes is called refinement. After a B-spline curve or surface is refined, the new control polygon or mesh defines the same curve or surface as the one defined by the former control polygon or mesh. When we continuously refine these polygons or meshes, we can obtain a sequence of polygons or meshes that converge to the B-spline curve or surface defined by it. Figures 2.14 and 2.15 give the process of refinement of a cubic B-spline curve.

Sederberg et al. [25] give non-uniform subdivision schemes. When we apply these subdivision schemes on topology rectangle meshes, we can obtain B-spline surfaces. These subdivision schemes can be derived from knot-doubling algorithms of B-spline curves and surfaces.

2.3.1 Subdivision of B-Spline Curves

Assume that there is no multiplicity knot in knot vector $U = [\dots, u_{-1}, u_0, u_1, \dots]$. Now, we insert a middle point for every knot interval and obtain a new knot vector: $\bar{U} = [\dots, \bar{u}_{-2}, \bar{u}_{-1}, \bar{u}_0, \bar{u}_1, \bar{u}_2, \dots]$. Consider quadric B-spline curve:

$$p(u) = \sum_{i=-\infty}^{+\infty} P_i N_{i,2}(u).$$

Its shape is shown in Fig. 2.16a.

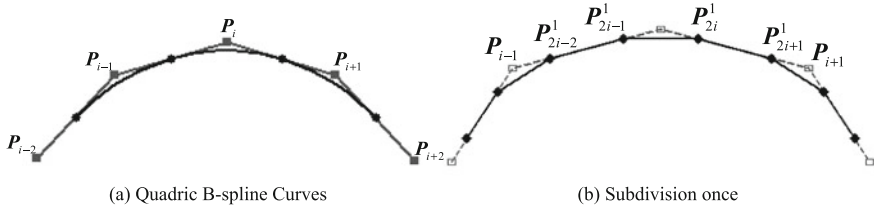


Fig. 2.16 Quadric B-spline curves and their subdivision



Fig. 2.17 Correspondence between knots and control vertices

From Fig. 2.16a, we know that every control vertex P_i corresponds to a curve segment. Set the length of parameter interval of the curve segment as d_i , and parameter of every vertex can be given as $P_i \rightarrow d_i$.

After doubling knots, we can get:

$$\begin{aligned} P_{2i}^1 &= \frac{(d_i + 2d_{i+1})P_i + d_i P_{i+1}}{2(d_i + d_{i+1})} \\ P_{2i+1}^1 &= \frac{d_{i+1}P_i + (2d_i + d_{i+1})P_{i+1}}{2(d_i + d_{i+1})}. \end{aligned} \quad (2.9)$$

As shown in Fig. 2.16b, in order to deduce Formula (2.9), let knots and control vertices correspond as shown in Fig. 2.17:

In this case,

$$d_i = u_{i+2} - u_{i+1}. \quad (2.10)$$

Now consider that how P_{i-1} , P_i , P_{i+1} are replaced in the process of subdivision. It is easy to know that P_{i-1} , P_i , P_{i+1} correspond to the curve segment:

$$p_i(u) = \sum_{j=i-1}^{i+1} P_j N_{j,2}(u), \quad u \in [u_{i+1}, u_{i+2}],$$

where $U_i = [u_{i-1}, u_i, u_{i+1}, u_{i+2}, u_{i+3}, u_{i+4}]$. Since only considering curve segment $p_i(u)$, we can know that new control vertices are only linear combinations of P_{i-1} , P_i , P_{i+1} . Consequently, we insert knots for all knot intervals in U_i by following steps:

Step 1: Insert $u = (u_{i-1} + u_i)/2 \in [u_{i-1}, u_i]$. By algorithm of inserting knots (2.7), to calculate new control vertices, we only need to control the following vertices and knots:

$$P_{i-3}, P_{i-2}, P_{i-1}; \quad u_{i-3}, u_{i-2}, u_{i-1}, u_i, u_{i+1}.$$

So, added vertices are not a linear combination of P_{i-1}, P_i, P_{i+1} . Consequently, we may not insert the knot $u = (u_{i-1} + u_i)/2 \in [u_{i-1}, u_i]$ and may not calculate these new added vertices, either.

Step 2: Insert $u = (u_i + u_{i+1})/2 \in [u_i, u_{i+1}]$. To calculate new control vertices, we only use the following control vertices and knots:

$$P_{i-2}, P_{i-1}, P_i; \quad u_{i-2}, u_{i-1}, u_i, u_{i+1}, u_{i+2}$$

New added vertices are:

$$\begin{aligned} Q_{i-2}^1 &= \frac{u_{i+1} - u_i}{2(u_{i+1} - u_{i-1})} P_{i-2} + \frac{u_{i+1} + u_i - 2u_{i-1}}{2(u_{i+1} - u_{i-1})} P_{i-1} \\ Q_{i-1}^1 &= \frac{2u_{i+2} - u_{i+1} - u_i}{2(u_{i+2} - u_i)} P_{i-1} + \frac{u_{i+1} - u_i}{2(u_{i+2} - u_i)} P_i. \end{aligned}$$

Correspondence between vertices and knots is

$$\begin{array}{cccccccc} P_{i-2} & Q_{i-2}^1 & Q_{i-1}^1 & P_i & P_{i+1} & P_{i+2} & P_{i+3} & P_{i+4} \\ | & | & | & | & | & | & | & | \\ u_{i-2} & u_{i-1} & u_i & (u_i + u_{i+1})/2 & u_{i+1} & u_{i+2} & u_{i+3} & u_{i+4} \end{array}$$

Step 3: Insert $u = (u_{i+1} + u_{i+2})/2 \in [u_{i+1}, u_{i+2}]$. To calculate new control vertices, we only use the following control vertices and knots:

$$\begin{aligned} & Q_{i-1}^1, P_i, P_{i+1}; \quad u_i, (u_i + u_{i+1})/2, u_{i+1}, u_{i+2}, u_{i+3} \\ Q_{i-1}^2 &= \frac{u_{i+2} - u_{i+1}}{2u_{i+2} - u_{i+1} - u_i} Q_{i-1}^1 + \frac{u_{i+2} - u_i}{2u_{i+2} - u_{i+1} - u_i} P_i \\ &= \frac{u_{i+2} - u_{i+1}}{2(u_{i+2} - u_i)} P_{i-1} + \frac{u_{i+2} + u_{i+1} - 2u_i}{2(u_{i+2} - u_i)} P_i \\ Q_i^2 &= \frac{2u_{i+3} - u_{i+2} - u_{i+1}}{2(u_{i+3} - u_{i+1})} P_i + \frac{u_{i+2} - u_{i+1}}{2(u_{i+3} - u_{i+1})} P_{i+1}. \end{aligned}$$

Correspondence between vertices and knots is

$$\begin{array}{cccccccc} Q_{i-2}^1 & Q_{i-1}^1 & Q_{i-1}^2 & Q_i^2 & P_{i+1} & P_{i+2} & P_{i+3} & P_{i+4} \\ | & | & | & | & | & | & | & | \\ u_{i-1} & u_i & (u_i + u_{i+1})/2 & u_{i+1} & (u_{i+1} + u_{i+2})/2 & u_{i+2} & u_{i+3} & u_{i+4} \end{array}$$

Step 4: Insert $u = (u_{i+2} + u_{i+3})/2 \in [u_{i+2}, u_{i+3}]$. To calculate new control vertices, we only use control vertices and knots:

$$\begin{aligned}
& \mathcal{Q}_i^2, \mathbf{P}_{i+1}, \mathbf{P}_{i+2}; u_{i+1}, (u_{i+1} + u_{i+2})/2, u_{i+2}, u_{i+3}, u_{i+4} \\
\mathcal{Q}_i^3 &= \frac{u_{i+3} - u_{i+2}}{2u_{i+3} - u_{i+2} - u_{i+1}} \mathcal{Q}_i^2 + \frac{u_{i+3} - u_{i+1}}{2u_{i+3} - u_{i+2} - u_{i+1}} \mathbf{P}_{i+1} \\
&= \frac{u_{i+3} - u_{i+2}}{2(u_{i+3} - u_{i+1})} \mathbf{P}_i + \frac{u_{i+3} + u_{i+2} - 2u_{i+1}}{2(u_{i+3} - u_{i+1})} \mathbf{P}_{i+1} \\
\mathcal{Q}_{i+1}^3 &= \frac{2u_{i+4} - u_{i+3} - u_{i+2}}{2(u_{i+4} - u_{i+2})} \mathbf{P}_{i+1} + \frac{u_{i+3} - u_{i+2}}{2(u_{i+4} - u_{i+2})} \mathbf{P}_{i+2}.
\end{aligned}$$

Correspondence between vertices and knots is

$$\begin{array}{cccccccc}
\mathcal{Q}_{i-2}^1 & \mathcal{Q}_{i-1}^1 & \mathcal{Q}_{i-1}^2 & \mathcal{Q}_i^2 & \mathcal{Q}_i^3 & \mathcal{Q}_{i+1}^3 & \mathbf{P}_{i+2} & \mathbf{P}_{i+3} \\
| & | & | & | & | & | & | & | \\
u_{i-1} & u_i & (u_i + u_{i+1})/2 & u_{i+1} & (u_{i+1} + u_{i+2})/2 & u_{i+2} & (u_{i+2} + u_{i+3})/2 & u_{i+3}
\end{array}$$

Now, we can find that $\mathbf{P}_{i-1}, \mathbf{P}_i, \mathbf{P}_{i+1}$ have already been replaced by $\mathcal{Q}_{i-1}^1, \mathcal{Q}_{i-1}^2, \mathcal{Q}_i^2, \mathcal{Q}_i^3$. Based on Formula (2.10) and Fig. 2.15b, we can get

$$\begin{aligned}
\mathbf{P}_{2i-2}^1 &= \mathcal{Q}_{i-1}^1 = \frac{(d_{i-1} + 2d_i)\mathbf{P}_{i-1} + d_{i-1}\mathbf{P}_i}{2(d_{i-1} + d_i)}, \\
\mathbf{P}_{2i-1}^1 &= \mathcal{Q}_{i-1}^2 = \frac{d_i\mathbf{P}_{i-1} + (2d_{i-1} + d_i)\mathbf{P}_i}{2(d_{i-1} + d_i)}, \\
\mathbf{P}_{2i}^1 &= \mathcal{Q}_i^2 = \frac{(d_i + 2d_{i+1})\mathbf{P}_i + d_i\mathbf{P}_{i+1}}{2(d_i + d_{i+1})}, \\
\mathbf{P}_{2i+1}^1 &= \mathcal{Q}_i^3 = \frac{d_{i+1}\mathbf{P}_i + (2d_i + d_{i+1})\mathbf{P}_{i+1}}{2(d_i + d_{i+1})}.
\end{aligned}$$

Consequently, we get Formula (2.9). These subdivided new vertices have the following parameters:

$$\mathbf{P}_{2i-1}^1 \rightarrow d_i/2, \quad \mathbf{P}_{2i}^1 \rightarrow d_i/2.$$

Similarly, we consider cubic B-spline curve:

$$p(u) = \sum_{i=-\infty}^{+\infty} \mathbf{P}_i N_{i,3}(u).$$

where $U = [\dots, u_{-1}, u_0, u_1, \dots]$. Assume that its shape is shown in Fig. 2.18a

From Fig. 2.18a, we know that each edge of the control polygon corresponds to a curve segment. Write the length of parameter interval of the curve segment as d_i , and the parameter of each edge can be given as $\mathbf{P}_i \mathbf{P}_{i+1} \rightarrow d_i$.

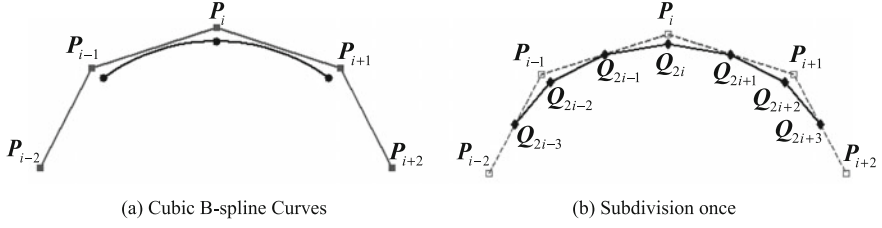


Fig. 2.18 Cubic B-spline curves and their subdivisions

After doubling the knots, we have:

$$\begin{aligned} Q_{2i+1} &= \frac{(d_i + 2d_{i+1})P_i + (d_i + 2d_{i-1})P_{i+1}}{2(d_{i-1} + d_i + d_{i+1})} \\ Q_{2i} &= \frac{d_i Q_{2i-1} + (d_{i-1} + d_i)P_i + d_{i-1} Q_{2i+1}}{2(d_{i-1} + d_i)}. \end{aligned} \quad (2.11)$$

Edges of the new control polygon have parameters:

$$Q_{2i} Q_{2i+1} \rightarrow d_i/2, \quad Q_{2i+1} Q_{2i+2} \rightarrow d_i/2.$$

2.3.2 Subdivision of B-Spline Surface

Firstly, we investigate bi-quadratic B-spline surfaces:

$$p(u, v) = \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} P_{i,j} N_{i,2}(u) N_{j,2}(v),$$

where $U = [\dots, u_{-1}, u_0, u_1, \dots]$, $V = [\dots, v_{-1}, v_0, v_1, \dots]$.

Since B-spline surfaces are tensor surfaces, we can double their knots in two parameter directions, as shown in Fig. 2.19. According to the subdivision Formula (2.9) of quadric B-spline curves, we can refine every column control vertices in u parameter direction:

$$\begin{aligned} Q_{i,2j} &= \frac{(d_j + 2d_{j+1})P_{i,j} + d_j P_{i,j+1}}{2(d_j + d_{j+1})} \\ Q_{i+1,2j} &= \frac{(d_j + 2d_{j+1})P_{i+1,j} + d_j P_{i+1,j+1}}{2(d_j + d_{j+1})}, \end{aligned}$$

where $P_{i,j} \rightarrow d_j$.

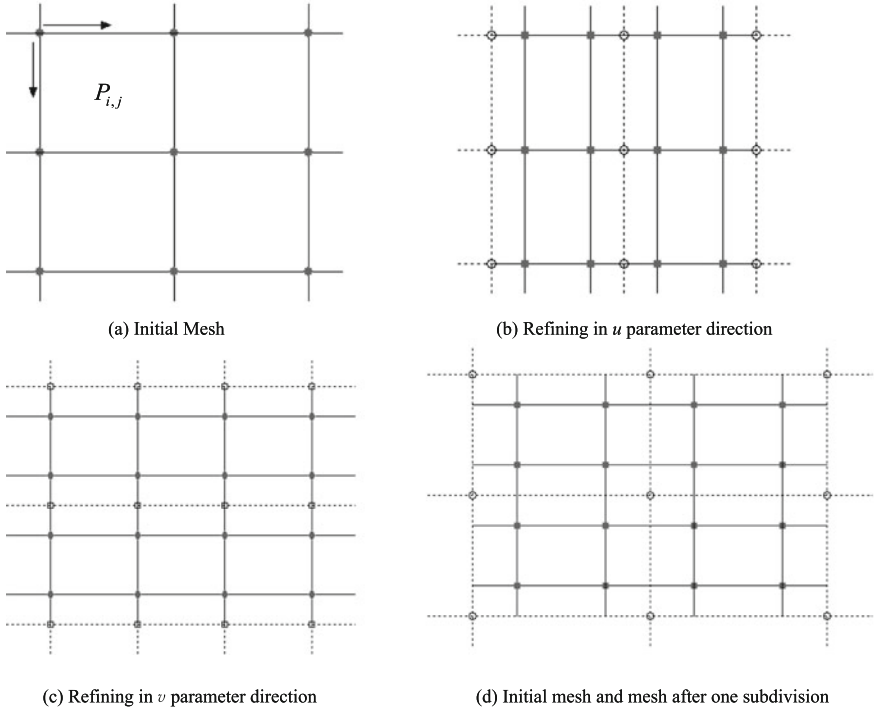


Fig. 2.19 Subdivision of bi-quadric B-spline surface

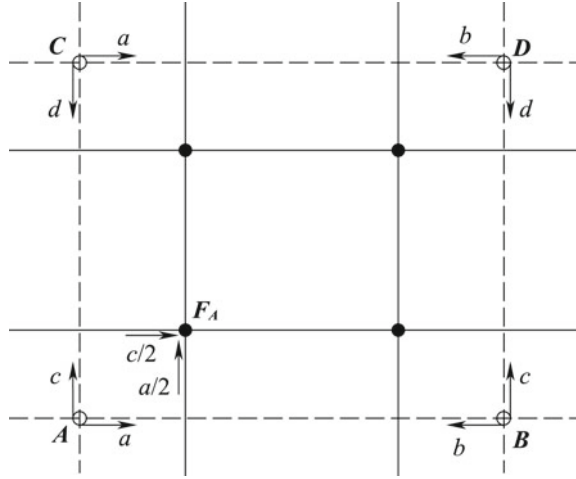
Consequently, we can obtain a new mesh M_Q . We now refine the control vertices in each row in v parameter direction:

$$\begin{aligned}
 P'_{2i,2j} &= \frac{(e_i + 2e_{i+1})Q_{i,2j} + e_i Q_{i+1,2j}}{2(e_i + e_{i+1})} \\
 &= ((e_i + 2e_{i+1})(d_j + 2d_{j+1})P_{i,j} + (e_i + 2e_{i+1})d_j P_{i,j+1} \\
 &\quad + e_i(d_j + 2d_{j+1})P_{i+1,j} + e_i d_j P_{i+1,j+1}) / \\
 &\quad (4(e_i + e_{i+1})(d_j + d_{j+1})) \\
 P'_{2i+1,2j} &= \frac{e_{i+1} Q_{i,2j} + (2e_i + e_{i+1}) Q_{i+1,2j}}{2(e_i + e_{i+1})} \\
 &= (e_{i+1}(d_j + 2d_{j+1})P_{i,j} + e_{i+1}d_j P_{i,j+1} + (2e_i + e_{i+1})(d_j + 2d_{j+1}) \\
 &\quad P_{i+1,j} + (2e_i + e_{i+1})d_j P_{i+1,j+1}) / \\
 &\quad (4(e_i + e_{i+1})(d_j + d_{j+1})),
 \end{aligned}$$

where $Q_{i,j} \rightarrow e_j$.

Assume that $P_{i,j} = C$, $P_{i,j+1} = D$, $P_{i+1,j} = A$, $P_{i+1,j+1} = B$,

Fig. 2.20 Knot doubling of quadratic B-spline surface



$$d_j = a, d_{j+1} = b, e_i = d, e_{i+1} = c,$$

And denoting $P'_{2i+1,2j}$ by F_A , we have

$$\begin{aligned} F_A &= \frac{c(a+2b)C + acD + (2d+c)(a+2b)A + (2d+c)aB}{4(a+b)(c+d)} \\ &= \frac{(ac+2bc+2ad+4bd)A + (2ad+ac)B + (2bc+ac)C + acD}{4(a+b)(c+d)}. \end{aligned} \quad (2.12)$$

Figure 2.20 gives other labels for Formula (2.12). Formula (2.12) is straightforward if we generalize these subdivision rules to any mesh, which can be found in the literature [25].

Now, we research bi-cubic B-spline surfaces and insert midpoint to every knot interval successively in u and v parameter directions, as shown in Fig. 2.21. From Fig. 2.21c, we can know that the process to obtain the mesh M^1 from the mesh M^0 uses the following rules:

- Every face in M^0 has a corresponding new vertex in M^1 , and the new vertex is called the new face point;
- Every edge in M^0 has a corresponding new vertex in M^1 , and the new vertex is called the new edge point; and
- Every vertex in M^0 has a corresponding new vertex in M^1 , and the new vertex is called the new vertex point.

We now deduce the calculating formulas for these three types of new points, which can be obtained from the knot-inserting algorithm of cubic B-spline curves:

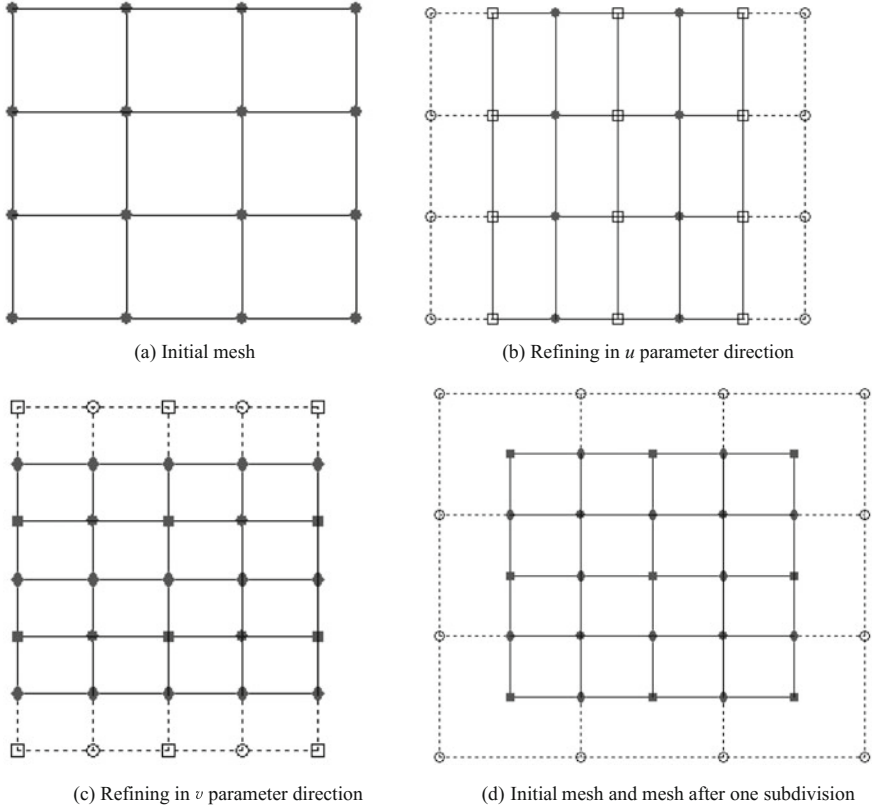


Fig. 2.21 Subdivision of bi-cubic B-spline surface

$$\begin{aligned} \mathbf{Q}_{i,2j+1} &= \frac{(d_j + 2d_{j+1})\mathbf{P}_{i,j} + (d_j + 2d_{j-1})\mathbf{P}_{i,j+1}}{2(d_{j-1} + d_j + d_{j+1})} \\ \mathbf{Q}_{i,2j} &= \frac{d_j \mathbf{Q}_{2j-1} + (d_{j-1} + d_j)\mathbf{P}_j + d_{j-1} \mathbf{Q}_{2j+1}}{2(d_{j-1} + d_j)}, \end{aligned}$$

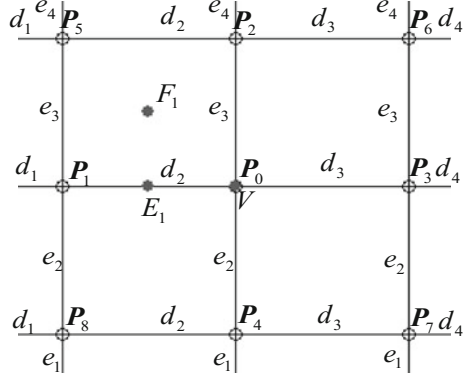
where $\mathbf{P}_{i,j} \mathbf{P}_{i,j+1} \rightarrow d_j$.

While we double knots in v parameter direction, we get

$$\begin{aligned} \mathbf{P}_{2i+1,2j+1}^1 &= \frac{(e_i + 2e_{i+1})\mathbf{Q}_{i,2j+1} + (e_i + 2e_{i-1})\mathbf{Q}_{i+1,2j+1}}{2(e_{i-1} + e_i + e_{i+1})} \\ &= \frac{a\mathbf{P}_{i,j} + b\mathbf{P}_{i,j+1} + c\mathbf{P}_{i+1,j} + d\mathbf{P}_{i+1,j+1}}{4(e_{i-1} + e_i + e_{i+1})(d_{j-1} + d_j + d_{j+1})}, \end{aligned}$$

where $a = (e_i + 2e_{i+1})(d_j + 2d_{j+1})$, $b = (e_i + 2e_{i+1})(d_j + 2d_{j-1})$, $c = (e_i + 2e_{i-1})(d_j + 2d_{j+1})$, $d = (e_i + 2e_{i-1})(d_j + 2d_{j-1})$, and $\mathbf{Q}_{i,j} \mathbf{Q}_{i+1,j} \rightarrow e_j$.

Fig. 2.22 Face, edge, and vertex points



After replacing the variables, we can find that the calculating formula of $P_{2i+1,2j+1}^1$ is in accordance with of the formula for calculating the new face point in the literature [25]:

$$F_1 = [(e_3 + 2e_4)(d_2 + d_1)P_0 + (e_3 + 2e_4)(d_2 + 2d_3)P_1 + (e_3 + 2e_2)(d_2 + 2d_3)P_5 + (e_3 + 2e_2)(d_2 + 2d_1)P_2] / 4(e_2 + e_3 + e_4)(d_1 + d_2 + d_3). \quad (2.13)$$

The variables in above formula are shown in Fig. 2.22.

Similarly, we can get formulas for the new edge point and the new vertex point:

$$E_1 = \frac{e_2F_1 + e_3F_4 + (e_2 + e_3)B_1}{2(e_2 + e_3)}, \quad (2.14)$$

where

$$B_1 = \frac{(2d_1 + d_2)P_0 + (d_2 + 2d_3)P_1}{2(d_1 + d_2 + d_3)}$$

$$V = \frac{P_0}{4} + \frac{d_3e_2F_1 + d_2e_2F_2 + d_2e_3F_3 + d_3e_3F_4}{4(d_2 + d_3)(e_2 + e_3)} + [d_3(e_2 + e_3)B_1 + e_2(d_2 + d_3)B_2 + d_2(e_2 + e_3)B_3 + e_3(d_2 + d_3)B_4] / [4(d_2 + d_3)(e_2 + e_3)]. \quad (2.15)$$

Consequently, subdivision of B-spline surface can be realized by subdivision of B-spline curves in two parameter directions based on tensor product property of B-spline surfaces.

2.4 Uniform Subdivision of B-Spline Curves and Surfaces

These subdivision formulas obtained with knot-inserting algorithm are complex. A majority of existing subdivision rules, such as Doo–Sabin subdivision, Catmull–Clark subdivision, Loop subdivision, do not involve in parameters of vertices and edges of meshes. Since knot distribution of uniform B-spline curves and surfaces is uniform, those parameters of vertices and edges are identical. Consequently, we need not consider parameters when we research subdivision of uniform B-spline curves and surfaces. So, simpler subdivision formulas can be obtained.

2.4.1 Subdivision of Uniform B-Spline Curves

Uniform B-spline has a definition equivalent to the recursive Formula (2.1)—the convolution definition. It is easy to deduce subdivision rules of uniform B-spline curves and surfaces with the convolution definition [8, 86].

Definition 2.2 The convolution of two continuous functions $f(u)$ and $g(u)$ can be defined as:

$$(f \otimes g)(u) = \int_s f(s)g(u-s)ds. \quad (2.16)$$

Theorem 2.1 *The convolution has properties as follows:*

Linearity: $f(u) \otimes (g(u) + h(u)) = f(u) \otimes g(u) + f(u) \otimes h(u)$,

Time shift: $f(u-i) \otimes g(u-j) = p(u-i-j)$,

Time scaling: $f(2u) \otimes g(2u) = \frac{1}{2}p(2u)$,

where $p(u) = f(u) \otimes g(u)$.

Proof We only deduce the time scaling property. The deduction of other two properties is similar to the following deduction.

$$\int_s f(2s)g(2u-2s)ds = \frac{1}{2} \int_s f(2s)g[2(u-s)]d(2s)$$

Let $2s = s'$, $2u = u'$, we have

$$\frac{1}{2} \int_s f(2s)g(2u-2s)d(2s) = \frac{1}{2} \int_{s'} f(s')g(u'-s')ds' = \frac{1}{2}p(u') = \frac{1}{2}p(2u) \quad \#$$

Theorem 2.2 $f(2u-i) \otimes g(2u-j) = \frac{1}{2}p(2u-i-j)$, where $p(u) = f(u) \otimes g(u)$

Proof $f(2u-i) \otimes g(2u-j) = \int_s f(2s-i)g(2u-j-2s)ds$

Assume $s' = 2s - i$, $u' = 2u$, we have

$$f(2u - i) \otimes g(2u - j) = \frac{1}{2} \int_{s'} f(s')g(u' - j - i - s')ds' = \frac{1}{2} p(2u - j - i)$$

Definition 2.3 Let zero-degree B-spline basic function be:

$$N_0(u) = \begin{cases} 1, & 0 \leq u \leq 1 \\ 0, & \text{other.} \end{cases} \quad (2.17)$$

k -degree B-spline basic function can be defined as:

$$N_k(u) = \int_s N_{k-1}(s)N_0(u - s)ds. \quad (2.18)$$

Based on Definition 2.3, when $k = 1$,

$$N_1(u) = \int_0^u N_0(s)N_0(u - s)ds = u, u \in [0, 1].$$

If $u \in [1, 2]$,

$$\begin{aligned} N_1(u) &= \int_{u-1}^1 N_0(s)N_0(u - s)ds = 2 - u, \\ \therefore N_1(u) &= \begin{cases} u & u \in [0, 1] \\ 2 - u & u \in [1, 2]. \end{cases} \end{aligned}$$

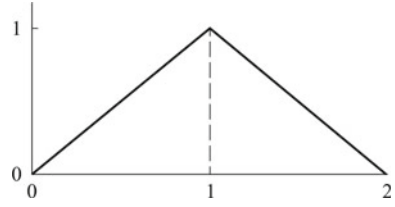
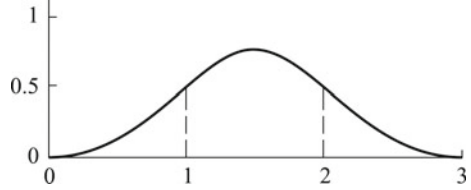
In the case of $k = 2$, we have

If $u \in [0, 1]$,

$$N_2(u) = \int_0^u N_1(s)N_0(u - s)ds = \int_0^s sds = \frac{1}{2}u^2.$$

If $u \in [1, 2]$,

$$\begin{aligned} N_2(u) &= \int_{u-1}^u N_1(s)N_0(u - s)ds = \int_{u-1}^1 N_1(s)N_0(u - s)ds + \\ &\quad \int_1^u N_1(s)N_0(u - s)ds \\ &= \int_{u-1}^1 sds + \int_1^u (2 - s)ds = -u^2 + 3u - \frac{3}{2}. \end{aligned}$$

Fig. 2.23 Shape of $N_1(u)$ **Fig. 2.24** Shape of $N_2(u)$ 

If $u \in [2, 3]$,

$$N_2(u) = \int_{u-1}^2 N_1(s)N_0(u-s)ds = \int_{u-1}^2 (2-s)ds = \frac{1}{2}u^2 - 3u + \frac{9}{2}$$

$$\therefore N_2(u) = \begin{cases} u^2/2, & u \in [0, 1] \\ -u^2 + 3u - 3/2, & u \in [1, 2] \\ u^2/2 - 3u + 9/2, & u \in [2, 3] \\ 0, & \text{other.} \end{cases}$$

The curves of $N_1(u)$ and $N_2(u)$ are shown in Figs. 2.23 and 2.24.

Let $U = [\dots, -2, -1, 0, 1, 2, \dots]$, it can be obtained from the above deduction:

$$\begin{cases} N_k(u) = N_{0,k}(u) \\ N_k(u-i) = N_{i,k}(u). \end{cases}$$

We now double knots of U :

$$U^1 = [\dots, -1, -1/2, 0, 1/2, 1, \dots].$$

Obviously, $N_0(2u)$ is the function defined on U^1 and it satisfies the formula:

$$N_0(u) = N_0(2u) + N_0(2u-1). \quad (2.19)$$

Based on Definitions 2.2, 2.3, and Theorem 2.1, we can know

$$N_k(u) = \bigotimes_{i=0}^k N_0(u) = \bigotimes_{i=0}^k [N_0(2u) + N_0(2u-1)]$$

$$\begin{aligned}
&= \sum_{i=0}^{k+1} \binom{k+1}{i} \left[\bigotimes_{j=0}^{k-i} N_0(2u) \right] \otimes \left[\bigotimes_{j=0}^{i-1} N_0(2u-1) \right] \\
&= \sum_{i=0}^{k+1} \binom{k+1}{i} \left[\frac{1}{2^{k-i}} N_{k-i}(2u) \right] \otimes \left[\frac{1}{2^{i-1}} N_i(2u-i) \right] \\
&= \frac{1}{2^k} \sum_{i=0}^{k+1} \binom{k+1}{i} N_k(2u-i).
\end{aligned}$$

Consequently, uniform B-spline satisfies the following theory,

Theorem 2.3

$$N_k(u) = \frac{1}{2^k} \sum_{i=0}^{k+1} \binom{k+1}{i} N_k(2u-i), \quad (2.20)$$

where the knot vector of $N_k(u)$ is U .

Based on Theorem 2.3, we can know that when $k = 2$, the quadratic uniform B-spline basic function is:

$$\begin{aligned}
N_2(u) &= \frac{1}{4} \sum_{i=0}^3 \binom{3}{i} N_2(2u-i) \\
&= \frac{1}{4} N_2(2u) + \frac{3}{4} N_2(2u-1) + \frac{3}{4} N_2(2u-2) + \frac{1}{4} N_2(2u-3) \\
&= \frac{1}{4} N_2(2u) + \frac{3}{4} N_2 \left[2 \left(u - \frac{1}{2} \right) \right] + \\
&\quad \frac{3}{4} N_2 [2(u-1)] + \frac{1}{4} N_2 \left[2 \left(u - \frac{3}{2} \right) \right] \\
\therefore N_{i,2}(u) &= \frac{1}{4} N_{2i-2}^1(u) + \frac{3}{4} N_{2i-1}^1(u) + \frac{3}{4} N_{2i}^1(u) + \frac{1}{4} N_{2i+1}^1(u). \quad (2.21)
\end{aligned}$$

Positions of $N_{0,2}(u)$ and $N_{j,2}^1(u)$, $j = 0, 1/2, 1, 3/2$ are shown in Fig. 2.25.

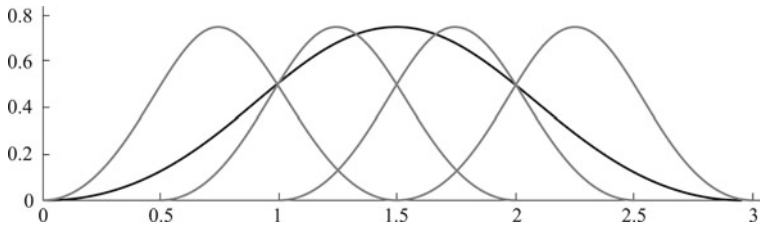


Fig. 2.25 Positions of $N_{0,2}(u)$ and $N_{j,2}^1(u)$ ($j = 0, 1/2, 1, 3/2$)

With Formula (2.21), the quadratic uniform B-spline curve can be represented as:

$$\begin{aligned}
 p(u) &= \sum_{i=-\infty}^{+\infty} P_i N_{i,2}(u) \\
 &= \cdots + P_{i-1} \left[\frac{1}{4} N_{2i-4}^1(u) + \frac{3}{4} N_{2i-3}^1(u) + \frac{3}{4} N_{2i-2}^1(u) + \frac{1}{4} N_{2i-1}^1(u) \right] + \\
 &\quad P_i \left[\frac{1}{4} N_{2i-2}^1(u) + \frac{3}{4} N_{2i-1}^1(u) + \frac{3}{4} N_{2i}^1(u) + \frac{1}{4} N_{2i+1}^1(u) \right] + \\
 &\quad P_{i+1} \left[\frac{1}{4} N_{2i}^1(u) + \frac{3}{4} N_{2i+1}^1(u) + \frac{3}{4} N_{2i+2}^1(u) + \frac{1}{4} N_{2i+3}^1(u) \right] + \cdots
 \end{aligned}$$

Let

$$P_{2i}^1 = \frac{3}{4} P_i + \frac{1}{4} P_{i+1}, \quad P_{2i+1}^1 = \frac{1}{4} P_i + \frac{3}{4} P_{i+1}. \quad (2.22)$$

We have,

$$p(u) = p^1(u) = \sum_{i=-\infty}^{+\infty} P_i^1 N_{i,2}^1.$$

It is easy to know that Formula (2.22) is an especial example of Formula (2.9) while all parameters are equal. When we continuously subdivide the control polygon P by Formula (2.22), we can get the series of polygons:

$$P^0, P^1, \dots, P^i, \dots$$

And,

$$P^\infty = p(u).$$

This is Chaikin algorithm which renders curves by cutting corners of polygons. The process can be represented by Fig. 2.15.

Similarly, the cubic uniform B-spline basic function can be written as:

$$N_{i,3}(u) = \frac{1}{8} N_{2i-2,3}^1(u) + \frac{4}{8} N_{2i-1,3}^1(u) + \frac{6}{8} N_{2i,3}^1(u) + \frac{4}{8} N_{2i+1,3}^1(u) + \frac{1}{8} N_{2i+2,3}^1(u) \quad (2.23)$$

Based on Formula (2.23), we can know that

$$\begin{aligned}
 p(u) &= \sum_{i=-\infty}^{+\infty} P_i N_{i,3}(u) \\
 &= \cdots + P_{i-1} \left[\frac{1}{8} N_{2i-4,3}^1(u) + \frac{4}{8} N_{2i-3,3}^1(u) + \frac{6}{8} N_{2i-2,3}^1(u) + \right. \\
 &\quad \left. \frac{4}{8} N_{2i-1,3}^1(u) + \frac{1}{8} N_{2i,3}^1(u) \right] +
 \end{aligned}$$

$$\begin{aligned}
& \mathbf{P}_i \left[\frac{1}{8} N_{2i-2,3}^1(u) + \frac{4}{8} N_{2i-1,3}^1(u) + \frac{6}{8} N_{2i,3}^1(u) + \right. \\
& \left. \frac{4}{8} N_{2i+1,3}^1(u) + \frac{1}{8} N_{2i+2,3}^1(u) \right] + \\
& \mathbf{P}_{i+1} \left[\frac{1}{8} N_{2i,3}^1(u) + \frac{4}{8} N_{2i+1,3}^1(u) + \frac{6}{8} N_{2i+2,3}^1(u) + \right. \\
& \left. \frac{4}{8} N_{2i+3,3}^1(u) + \frac{1}{8} N_{2i+4,3}^1(u) \right] + \cdots \\
& = \cdots + \frac{1}{2} (\mathbf{P}_{i-1} + \mathbf{P}_i) N_{2i-1,3}^1 + \left(\frac{1}{8} \mathbf{P}_{i-1} + \frac{4}{8} \mathbf{P}_i + \frac{1}{8} \mathbf{P}_{i+1} \right) N_{2i,3}^1 + \\
& \frac{1}{2} (\mathbf{P}_i + \mathbf{P}_{i+1}) N_{2i+1,3}^1 + \cdots
\end{aligned}$$

Let

$$\mathbf{P}_{2i}^1 = \frac{1}{8} \mathbf{P}_{i-1} + \frac{6}{8} \mathbf{P}_i + \frac{1}{8} \mathbf{P}_{i+1}, \quad \mathbf{P}_{2i+1}^1 = \frac{1}{2} \mathbf{P}_i + \frac{1}{2} \mathbf{P}_{i+1}. \quad (2.24)$$

We can get

$$p(u) = \mathbf{p}^1(u) = \sum_{-\infty}^{+\infty} \mathbf{P}_i^1 N_{i,3}^1.$$

Formula (2.24) is a special case of Formula (2.11) while all parameters are equal.

2.4.2 Subdivision of Uniform B-Spline Surfaces

Assume that $\mathbf{U} = \mathbf{V} = [\dots, -2, -1, 0, 1, 2, \dots]$ are knot vectors in two parameter directions. We study the change of control meshes of B-spline surfaces while we double knots of $\mathbf{U}$ and $\mathbf{V}$. We firstly investigate the uniform bi-quadratic B-spline surface:

$$\mathbf{p}(u, v) = \sum_{-\infty}^{+\infty} \mathbf{P}_{i,j} N_{i,2}(u) N_{j,2}(v), \quad (u, v) \in (-\infty, +\infty).$$

When each knot interval is bisected, we can know from Formula (2.21):

$$\begin{aligned}
N_{i,2}(u) N_{j,2}(v) &= \frac{1}{16} \sum_{s=0}^3 \binom{3}{s} N_{2i+s-2,2}^1(u) \sum_{t=0}^3 \binom{3}{t} N_{2i+t-2,2}^1(v) \\
\therefore N_{i,2}(u) N_{j,2}(v) &= \frac{1}{16} \sum_{s=0}^3 \sum_{t=0}^3 \binom{3}{s} \binom{3}{t} N_{2i+s-2,2}^1(u) N_{2i+t-2,2}^1(v).
\end{aligned} \quad (2.25)$$

Let $N_{2i+s-2}^1 N_{2j+t-2}^1 = N_{2i+s-2, 2j+t-2}^1 = (2i+s-2, 2j+t-2)$, basic functions and coefficients in every term of (2.25) have the following correspondence relations:

$$\begin{bmatrix} (2i-2, 2j-2) & (2i-2, 2j-1) & (2i-2, 2j) & (2i-2, 2j+1) \\ (2i-1, 2j-2) & (2i-1, 2j-1) & (2i-1, 2j) & (2i-1, 2j+1) \\ (2i, 2j-2) & (2i, 2j-1) & (2i, 2j) & (2i, 2j+1) \\ (2i+1, 2j-2) & (2i+1, 2j-1) & (2i+1, 2j) & (2i+1, 2j+1) \end{bmatrix} \rightarrow \frac{1}{16} \begin{bmatrix} 1 & 3 & 3 & 1 \\ 3 & 9 & 9 & 3 \\ 3 & 9 & 9 & 3 \\ 1 & 3 & 3 & 1 \end{bmatrix}. \quad (2.26)$$

The matrix on the left of Formula (2.26) is called the subdivision coefficient matrix of $P_{i,j}$. For any term $N_{2i+s-1, 2j+t-1}^1$ in the subdivision coefficient matrix of $P_{i,j}$, there are always subdivision coefficient matrixes of four vertices in which the $N_{2i+s-1, 2j+t-1}^1$ is. Consequently,

$$\begin{aligned} S(u, v) &= \frac{1}{16} \sum_{-\infty}^{+\infty} \sum_{-\infty}^{+\infty} P_{i,j} \left[\sum_{s=0}^3 \sum_{t=0}^3 \binom{3}{s} \binom{3}{t} N_{2i+s-2, 2j+t-2}^1 \right] \\ &= \cdots + \left(\frac{9}{16} P_{i-1, j-1} + \frac{3}{16} P_{i-1, j} + \frac{3}{16} P_{i, j-1} + \frac{1}{16} P_{i, j} \right) N_{2i-2, 2j-2}^1 + \\ &\quad \left(\frac{3}{16} P_{i-1, j-1} + \frac{1}{16} P_{i-1, j} + \frac{9}{16} P_{i, j-1} + \frac{3}{16} P_{i, j} \right) N_{2i-2, 2j-1}^1 + \\ &\quad \left(\frac{1}{16} P_{i-1, j-1} + \frac{3}{16} P_{i-1, j} + \frac{9}{16} P_{i, j-1} + \frac{3}{16} P_{i, j} \right) N_{2i-1, 2j-2}^1 + \\ &\quad \left(\frac{1}{16} P_{i-1, j-1} + \frac{3}{16} P_{i-1, j} + \frac{3}{16} P_{i, j-1} + \frac{9}{16} P_{i, j} \right) N_{2i-1, 2j-1}^1. \end{aligned}$$

Let

$$\begin{aligned} P_{2i-2, 2j-2}^1 &= \frac{9}{16} P_{i-1, j-1} + \frac{3}{16} P_{i-1, j} + \frac{3}{16} P_{i, j-1} + \frac{1}{16} P_{i, j}, \\ P_{2i-2, 2j-1}^1 &= \frac{3}{16} P_{i-1, j-1} + \frac{1}{16} P_{i-1, j} + \frac{9}{16} P_{i, j-1} + \frac{3}{16} P_{i, j}, \\ P_{2i-1, 2j-2}^1 &= \frac{1}{16} P_{i-1, j-1} + \frac{3}{16} P_{i-1, j} + \frac{9}{16} P_{i, j-1} + \frac{3}{16} P_{i, j}, \\ P_{2i-1, 2j-1}^1 &= \frac{1}{16} P_{i-1, j-1} + \frac{3}{16} P_{i-1, j} + \frac{3}{16} P_{i, j-1} + \frac{9}{16} P_{i, j}. \end{aligned}$$

So we can get the subdivision masks of uniform biquadratic B-spline surfaces as shown in Fig. 2.26. A subdivision mask is a mesh formed by old vertices connected with a calculated new vertex. There is the coefficient of the old vertex on the position

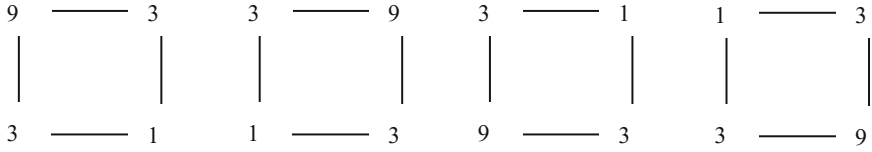


Fig. 2.26 Subdivision masks of uniform biquadratic B-spline surfaces

of the old vertex on the mask. For a mask, all vertices and their coefficients form a linear expression whose value is the geometric position of the corresponding new vertex. A set of subdivision rules is probably described by several masks. The new mesh topology structure is the topology structure shown in Fig. 2.21d.

For the uniform bi-cubic B-spline surface:

$$\mathbf{p}(u, v) = \sum_{-\infty}^{+\infty} \mathbf{P}_{i,j} N_{i,3}(u) N_{j,3}(v), (u, v) \in (-\infty, +\infty).$$

When every knot interval is bisected, we can know through Formula (2.23):

$$\begin{aligned} N_{i,3}(u) N_{j,3}(v) &= \frac{1}{64} \sum_{s=0}^4 \binom{4}{s} N_{2i+s-2,3}^1(u) \sum_{t=0}^4 \binom{4}{t} N_{2j+t-2,3}^1(v) \\ \therefore N_{i,3}(u) N_{j,3}(v) &= \frac{1}{64} \sum_{s=0}^4 \sum_{t=0}^4 \binom{4}{s} \binom{4}{t} N_{2i+s-2,3}^1(u) N_{2j+t-2,3}^1(v). \end{aligned} \quad (2.27)$$

Let $N_{2i+s-2}^1 N_{2j+t-2}^1 = N_{2i+s-2, 2j+t-2}^1 = (2i+s-2, 2j+t-2)$, basic functions and coefficients in each term of (2.22) have the following correspondence relation:

$$\begin{aligned} &\begin{bmatrix} (2i-2, 2j-2) & (2i-2, 2j-1) & (2i-2, 2j) & (2i-2, 2j+1) & (2i-2, 2j+2) \\ (2i-1, 2j-2) & (2i-1, 2j-1) & (2i-1, 2j) & (2i-1, 2j+1) & (2i-1, 2j+2) \\ (2i, 2j-2) & (2i, 2j-1) & (2i, 2j) & (2i, 2j+1) & (2i, 2j+2) \\ (2i+1, 2j-2) & (2i+1, 2j-1) & (2i+1, 2j) & (2i+1, 2j+1) & (2i+2, 2j+1) \\ (2i+2, 2j-2) & (2i+2, 2j-1) & (2i+2, 2j) & (2i+2, 2j+1) & (2i+2, 2j+2) \end{bmatrix} \rightarrow \\ &\frac{1}{64} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}. \end{aligned} \quad (2.28)$$

Through the correspondence relation (2.28), we have known that there are subdivision matrixes of nine vertices which contain the term $N_{2i-2, 2j-2}^1 : \mathbf{P}_{i-2, j-2}, \mathbf{P}_{i-1, j-2},$

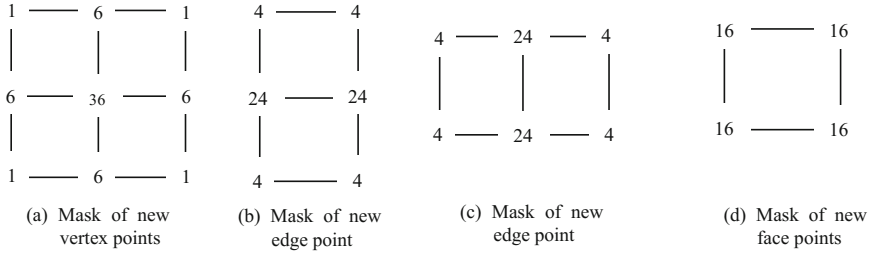


Fig. 2.27 Subdivision masks of uniform bi-cubic B-spline surfaces

$P_{i,j-2}$, $P_{i-2,j-1}$, $P_{i-1,j-1}$, $P_{i,j-1}$, $P_{i-2,j}$, $P_{i-1,j}$, $P_{i,j}$, and coefficient of $N_{2i-2,2j-2}^1$ in each subdivision matrix is shown in Fig. 2.27a:

As for $N_{2i-2,2j-1}^1$, there are subdivision matrixes of six vertices which contain the term: $P_{i-2,j-1}$, $P_{i-1,j-1}$, $P_{i,j-1}$, $P_{i-2,j}$, $P_{i-1,j}$, $P_{i,j}$. Coefficient of $N_{2i-2,2j-1}^1$ in each subdivision matrix is shown in Fig. 2.27b. Similarly, the subdivision mask of $N_{2i-1,2j-2}^1$ is shown in Fig. 2.27c. The subdivision mask of $N_{2i-1,2j-1}^1$ is shown in Fig. 2.27d. Consequently, the subdivision rules of uniform bi-cubic B-spline surface are found.

In the above discussion, we always assume that the control meshes are infinite. The case does not exist at all in engineer application. However, we can apply subdivision rules on finite control meshes, as shown in Figs. 2.19d and 2.21d. So, we can obtain a mesh series which continuously shrinks, and its limitation is a B-spline surface defined by the control mesh. The subdivision of finite polygons is an analogy of infinite meshes.

It is a basic idea of this section to obtain subdivision rules of spline surfaces from subdivision rules of basic functions. The deduction of subdivision rules of box spline surfaces also follows the idea.

2.5 Box Spline

The spline on triangular grids is one of the simplest forms of box splines. It is found by Doo-Sabin and evolved into current box splines due to the researches of de-Boor et al. The uniform B-spline can be regarded as a special case of box splines. Subdivision algorithms on box splines can be founded in the literature [91–93]. These subdivision algorithms can refine control meshes of any box spline surface. This section mainly introduces integral induction definition of bi-variant box splines.

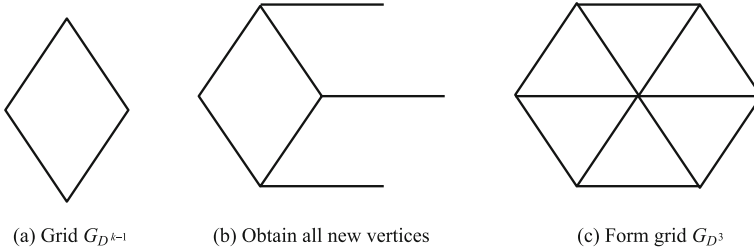
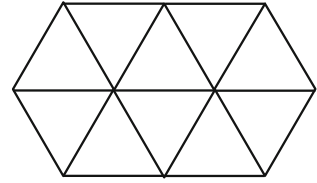


Fig. 2.28 Process from G_D^2 to G_D^3

Fig. 2.29 Triangle grid



2.5.1 Vector Group and Support Mesh

Definition 2.4 Let $s_0 = [1, 0]^T$, $s_1 = [0, 1]^T$, $s_2 = s_0 + s_1$, $s_3 = s_0 - s_1$, and

$$D^2 = \{s_i, s_j\}, (i \neq j, i, j = 0, 1, 2, 3)$$

$$D^k = D^{k-1} \cup \{s_i\}, (i = 0, 1, 2, 3, k \geq 2).$$

We call that $D^k (k \geq 2)$ is a vector group. Without loss of generality, we let $D^2 = \{s_0, s_1\}$ in the latter discussion. The grid generated by D^k is denoted by G_D^k and is called the support grid. Obviously, G_D^2 is a rhombus shown in Fig. 2.28a. To obtain G_D^k from G_D^{k-1} , the following process can be used:

Step 1: Make a unit length extend in direction u_i on boundary vertices of G_D^{k-1} . If a vertex obtained by extend does not superpose any vertex of G_D^{k-1} , a new vertex is obtained.

Step 2: All new vertices and vertices of G_D^{k-1} form vertices of G_D^k . Link vertices of G_D^k with vectors in D^k , and then the grid G_D^k is formed.

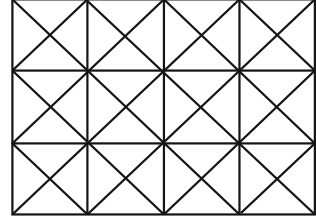
Figure 2.28 gives the process from G_D^2 to G_D^3 , where $D^3 = \{s_0, s_1, s_2\}$.

By translating G_D^k to each vertex of the existing grid, we can obtain an infinite mesh covering the whole plane. As for a vector group D^k , If $\{s_0, s_1\} \subseteq D^k$ and $\{s_0, s_1, s_2\} \not\subseteq D^k$ and $\{s_0, s_1, s_2, s_3\} \not\subseteq D^k$, an infinite quadrilateral grid can be formed by D^k .

If $\{s_0, s_1, s_2\} \subseteq D^k$ and $\{s_0, s_1, s_2, s_3\} \not\subseteq D^k$, an infinite triangle grid can be formed by D^k . The triangle grid usually becomes the shape as shown in Fig. 2.29 by the affine transform.

If $\{s_0, s_1, s_2, s_3\} \subseteq D^k$, an infinite tri-quadrangle mesh can be formed by D^k . It is shown in Fig. 2.30.

Fig. 2.30 Tri-quadrangle grid



It is easy to know that any vertex of a grid can be denoted as:

$$\mathbf{i} = i_0 \mathbf{s}_0 + i_1 \mathbf{s}_1 = (i_1, i_2),$$

where $\{\mathbf{e}_1, \mathbf{e}_2\}$ forms an affine reference frame. $\mathbf{i}$ is called the knot of the grid. In this case, any point on the plane can be denoted as:

$$\mathbf{x} = u \mathbf{s}_0 + v \mathbf{s}_1 = (u, v).$$

Definition 2.5 The support region of a vector group D^k is the region that is enclosed by the boundary of the grid G_D^k and can be denoted as:

$$\text{supp}(D^k) = [t_0, t_1, \dots, t_{k-1}][0, 1)^k,$$

where $D^k = \{t_0, t_1, \dots, t_{k-1}\}$, $t_i \in \{s_0, s_1, s_2, s_3\}$. In the latter discussion, elements in D^k are denoted by $s_0, s_1, \dots, s_{k-1}$ successively, unless otherwise specified. Obviously, if $\mathbf{x} \in \text{supp}(D^k)$,

$$\mathbf{x} = \alpha_0 \mathbf{s}_0 + \dots + \alpha_{k-1} \mathbf{s}_{k-1}, \alpha_i \in [0, 1].$$

Definition 2.6 Translate G_D^k to a knot of the plane grid produced by L_D^k and obtain a new grid $G_{i,D}^k$, and the region that is enclosed by $G_{i,D}^k$ is also called the support region of the vector group D^k and denoted as:

$$\text{supp}(D_i^k) = \mathbf{i} + \text{supp}(D^k).$$

When $D^k = \{s, \dots, s\}$ (where $s \in \{s_0, s_1, s_2, s_3\}$), the discussions in this section and latter sections are also applicable. In this case, uniform B-spline is obtained.

2.5.2 Inductive Definition of Box Splines

Definition 2.7 Let $D^k (k \geq 2)$ be vector groups, the inductive definition of box splines can be given as [94]:

$$N_{D^2}(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \text{supp } D^2 \\ 0, & \text{other} \end{cases} \quad (2.29)$$

$$N_{D^k}(\mathbf{x}) = \int_0^1 N_{D^{k-1}}(\mathbf{x} - t\mathbf{t}_k) dt \quad (2.30)$$

When $D^3 = \{s_0, s_1, s_2\}$, L_D^3 is shown in Fig. 2.29a. Let $\mathbf{x} = (u, v)$, we get

$$\mathbf{x} - ts_2 = (u - t, v - t).$$

Due to Definition 2.7, it is easy to know that when

$$\max\{u - 1, v - 1, 0\} \leq t \leq \min\{u, v, 1\},$$

$N_{D^3}(\mathbf{x}) \neq 0$. Consequently, when $\mathbf{x}$ is in the region enclosed by $(0, 0)$, $(1, 0)$, $(1, 1)$,

$$\begin{aligned} v &\leq u \leq 1, 0 \leq v \leq 1 \\ \therefore 0 &\leq t \leq v \\ \therefore N_{D^3}(\mathbf{x}) &= \int_0^v dt = v. \end{aligned}$$

When $\mathbf{x}$ is in the region enclosed by $(1, 1)$, $(2, 2)$, $(2, 1)$,

$$\begin{aligned} v &\leq u \leq 2, 1 \leq v \leq 2 \\ \therefore N_{D^3}(\mathbf{x}) &= \int_{u-1}^1 dt = 2 - u, \end{aligned}$$

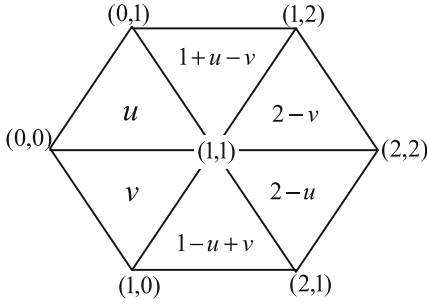
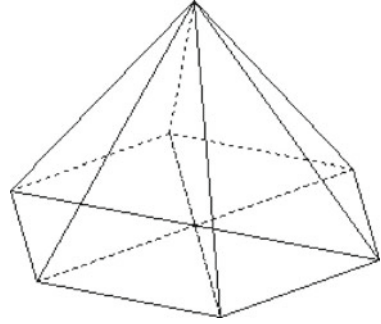
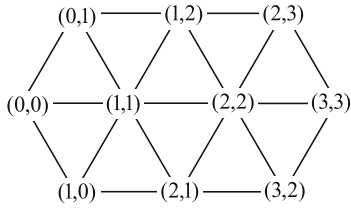
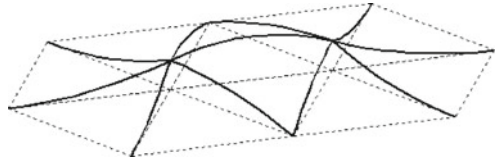
When $\mathbf{x}$ is in the region enclosed by $(1, 0)$, $(1, 1)$, $(2, 1)$,

$$\begin{aligned} 1 &\leq u \leq 1 + v, 0 \leq v \leq 1 \\ N_{D^3}(\mathbf{x}) &= \int_{u-1}^v dt = 1 - u + v, \end{aligned}$$

Similarly, we can deduce those expressions of $N_{D^3}(\mathbf{x})$ when $\mathbf{x}$ is in other regions of $\text{supp } (D^3)$. Those expressions are shown in Fig. 2.31a. Figure 2.31b gives out the shape of the function $N_{D^3}(\mathbf{x})$. When $(u, v) \notin \text{supp}(D^3)$,

$$\begin{aligned} \mathbf{x} - ts_0 &= (u - t, v - t) \notin \text{supp}(D^3) \\ \therefore N_{D^3}(\mathbf{x}) &= \int_0^1 0 dt = 0. \end{aligned}$$

We now investigate the box spline defined by $D^4 = \{s_0, s_1, s_2, s_3\}$. G_D^4 is shown in Fig. 2.32a. Let $\mathbf{x} = (u, v)$, we have

(a) Expressions of $N_{D^3}(\mathbf{x})$ (b) Shape of $N_{D^3}(\mathbf{x})$ **Fig. 2.31** Box spline defined by $D^3 = \{s_0, s_1, s_2\}$ (a) Support region of D^k (b) Shape of $N_{D^k}(\mathbf{x})$ **Fig. 2.32** Box spline defined by $D^k = \{s_0, s_1, s_2, s_3\}$

$$\mathbf{x} - ts_2 = (u - t, v - t)$$

$\therefore \mathbf{x} \in [v_1, v_2, v_3, v_4][0, 1]^3$ where $N_{D^4}(\mathbf{x}) \neq 0$

$\therefore \max\{u - 2, v - 2, 0\} \leq t \leq \min\{u, v, 1\}$.

When $\mathbf{x}$ is in the region enclosed by $(0, 0), (1, 0), (1, 1)$,

$$v \leq u \leq 1, 0 \leq v \leq 1$$

$$\therefore 0 \leq t \leq v$$

$$\therefore N_{D^4}(\mathbf{x}) = \int_0^v (v - t) dt = \frac{1}{2} v^2.$$

When $\mathbf{x}$ is in the region enclosed by $(1, 0), (1, 1), (2, 1)$,

$$1 \leq u \leq 1 + v, 0 \leq v \leq 1$$

$$\therefore 0 \leq t \leq v$$

$$N_{D^4}(\mathbf{x}) = \int_0^{u-1} (1 - u + v) dt + \int_{u-1}^v (v - t) dt$$

$$= (1 - u + v)(u - 1) + \frac{1}{2}(1 - u + v)^2.$$

When $\mathbf{x}$ is in the region enclosed by (1, 1), (2, 2), (2, 1),

$$\begin{aligned} v &\leq u \leq 2, 1 \leq v \leq 2 \\ \therefore 0 &\leq t \leq 1 \\ N_{D^4}(\mathbf{x}) &= \int_0^{v-1} (2 - u + t)dt + \int_{v-1}^{u-1} [1 - (u - t) + (v - t)]dt + \\ &\quad \int_{u-1}^1 (v - t)dt \\ &= (2 - u)(v - 1) + \frac{1}{2}(v - 1)^2 + (1 - u + v)(u - v) + \\ &\quad \frac{1}{2}(1 - u + v)^2 - \frac{1}{2}(v - 1)^2 \\ &= (2 - u)(v - 1) + (1 - u + v)(u - v) + \frac{1}{2}(1 - u + v)^2. \end{aligned}$$

When $\mathbf{x}$ is in the region enclosed by (2,2), (3,2), (2,1),

$$\begin{aligned} 2 &\leq u \leq 1 + v, 1 \leq v \leq 2 \\ \therefore u - 2 &\leq t \leq 1 \\ N_{D^4}(\mathbf{x}) &= \int_{u-2}^{v-1} (2 - u + t)dt + \int_{v-1}^1 (1 - u + v)dt \\ &= (2 - u)(v - u + 1) + \frac{1}{2}(v - 1)^2 - \frac{1}{2}(u - 2)^2 + (1 - u + v)(2 - v). \end{aligned}$$

When $\mathbf{x}$ is in the region enclosed by (2,2), (3,3), (3,2),

$$\begin{aligned} v &\leq u \leq 3, 2 \leq v \leq 3 \\ \therefore u - 2 &\leq t \leq 1 \\ N_{D^4}(\mathbf{x}) &= \int_{u-2}^1 (2 - u + t)dt = (2 - u)(3 - u) + \frac{1}{2} - \frac{1}{2}(u - 2)^2. \end{aligned}$$

Similarly, we can obtain expressions of $N_{D^4}(\mathbf{x})$ in other regions of $\text{supp}(D^4)$. The shape of $N_{D^4}(\mathbf{x})$ is shown in Fig. 2.32b.

Based on Formula (2.30), it is easy to know that it only makes a convolution for $N_{D^{k-1}}(\mathbf{x})$ and $h(t\mathbf{t}_k)$ in the $\mathbf{t}_k$ direction to obtain $N_{D^k}(\mathbf{x})$. The process is shown in Fig. 2.33, where

$$h(t\mathbf{t}_k) = \begin{cases} 1 & t \in [0, 1) \\ 0 & t \in (-\infty, 0) \cup [1, +\infty). \end{cases}$$

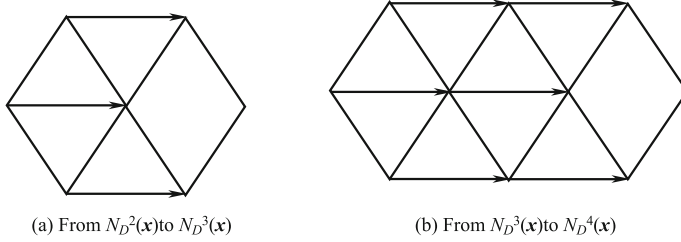


Fig. 2.33 Process of convolution

Actually, when we evaluate a box spline $N_{D^4}(\mathbf{x})$, the following recursive formula is applicable [95]:

$$(k-2)N_{D^k}(\mathbf{x}) = \sum_{i=1}^k [t_i N_{\{D^k \setminus \{t_i\}}(\mathbf{x}) + (1-t_i)N_{\{D^k \setminus \{t_i\}}(\mathbf{x} - \mathbf{t}_i)],$$

where $\{D^k \setminus \{t_i\}$ denotes the vector group obtained by deleting $\mathbf{t}_i$ for D^k . If we only want to show the shape of a box spline $N_{D^k}(\mathbf{x})$, subdivision can be used. How to make a subdivision will be discussed in later sections.

2.5.3 Basic Properties of Box Splines

From Definition 2.7 and the discussion of Sect. 2.5.2, box splines have the following properties [94]:

- (1) It does not depend on the ordering of vectors $\mathbf{t}_i (i = 1, \dots, k)$;
- (2) It is non-negative over the support region $\text{supp}(D^k)$:

$$N_{D^k}(\mathbf{x}) \begin{cases} > 0 & \mathbf{x} \in O(\text{supp}(D^k)) \\ = 0 & \mathbf{x} \notin O(\text{supp}(D^k)), \end{cases}$$

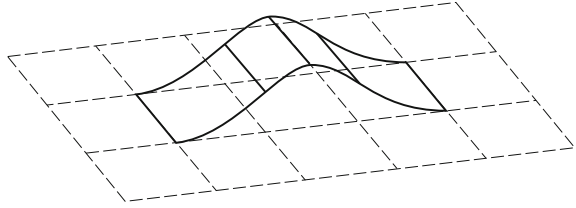
where $O(\text{supp}(D^k))$ is an open set which is the inner region of $\text{supp}(D^k)$.

- (3) It is symmetric with respect to the center of its support region.
- (4) It is $k-2$ degree polynomial over each tile of this partition.

Further, let $N_{D^k}(\mathbf{y}) := N_{D^k}(\mathbf{x} + y\mathbf{t}_r)$. If $\mathbf{t}_r \notin \text{span}\{D^k \setminus \{\mathbf{t}_r\}\}$, $N_{D^k}(\mathbf{y})$ is piece constant in $\mathbf{t}_r$ direction. It is easy to be found in the case that $D^4 = \{\mathbf{t}_0, \mathbf{t}_0, \mathbf{t}_0, \mathbf{t}_1\}$. In the $\mathbf{t}_1$ direction, $N_{D^4}(\mathbf{y})$ is piece constant which is shown in Fig. 2.34.

If $\mathbf{t}_r \in \text{span}\{D^k \setminus \{\mathbf{t}_r\}\}$, then $N_{D^k}(\mathbf{y})$ is continuous since it can be obtained by a convolution from $N_{D^k \setminus \{v_r\}}(\mathbf{y}) = N_{D^k \setminus \{v_r\}}(\mathbf{x} + y\mathbf{t}_r)$,

Fig. 2.34 $N_{D^k}(\mathbf{x} + y\mathbf{t}_r)$ is piece constant when $D^4 = \{s_0, s_0, s_0, s_1\}$



$$\begin{aligned} N_{D^k}(\mathbf{y}) &= \int_0^1 N_{D^k \setminus \mathbf{t}_r}(\mathbf{y} - t) dt = \int_{y-1}^y N_{D^k \setminus \mathbf{t}_r}(t) dt \\ &= \int_{-\infty}^y [N_{D^k \setminus \mathbf{t}_r}(t) - N_{D^k \setminus \mathbf{t}_r}(t-1)] dt. \end{aligned}$$

Further, the directional derivative with respect to $\mathbf{v}_r$ is given by

$$\frac{N_{D^k}(\mathbf{x})}{\partial \mathbf{t}_r} = \frac{N_{D^k}(\mathbf{y})}{\partial y} \Big|_{y=0} = N_{D^k \setminus \mathbf{t}_r}(\mathbf{x}) - N_{D^k \setminus \mathbf{t}_r}(\mathbf{x} - \mathbf{v}_r).$$

If $D^k \setminus \mathbf{t}_r$ spans space $\mathbb{R}^2$ (i.e., plane), then $N_{D^k}(\mathbf{x})$ is continuous and its directional derivatives can be written as a translating linear combination of box splines $N_{D^k \setminus \mathbf{v}_r}(\mathbf{x})$. Applying this argument repeatedly, we can see that

(5) $N_{D^k}(\mathbf{x})$ is $s - 1$ times continuously differentiable in $\mathbf{t}_r$ direction if there are $s\mathbf{t}_r$ vectors in the vector group D^k .

(6) $N_{D^k}(\mathbf{x})$ is normative:

$$\int_{R^s} N_{D^k}(\mathbf{x}) d\mathbf{x} = 1 \quad \text{or} \quad \sum_i N_{D^k}(\mathbf{x} - \mathbf{i}) = 1,$$

where $R^s = \text{span}(D^k)$. Actually,

$$\begin{aligned} \int_{R^s} N_{D^k}(\mathbf{x}) d\mathbf{x} &= \int_{R^s} \int_0^1 D^{k-1}(\mathbf{x} - t\mathbf{v}_k) dt d\mathbf{x} \\ &= \int_0^1 \left[\int_{R^s} N_{D^{k-1}}(\mathbf{x} - t\mathbf{v}_k) d\mathbf{x} \right] dt = \int_0^1 dt = 1. \end{aligned}$$

2.6 Box Spline Surfaces

2.6.1 Definition and Properties

Definition 2.8 Let Γ_{D^k} be the plane grid formed by gradually translating G_{D^k} to each knot. $\mathbf{i}$ denotes the knot of Γ_{D^k} . We call

$$\mathbf{p}(\mathbf{x}) = \sum_i \mathbf{P}_i N_{D^k}(\mathbf{x} - \mathbf{i}) \quad (2.31)$$

the box spline surface, where $\mathbf{P}_i = (\mathbf{P}_{ix}, \mathbf{P}_{iy}, \mathbf{P}_{iz})$ is the point in a three-dimensional space. It is called control vertices of $\mathbf{S}(\mathbf{x})$. The mesh M formed by $\{\mathbf{P}_i\}$ is topology isomorphic to Γ_{D^k} and is the control mesh of $\mathbf{S}(\mathbf{x})$. In the case of not raising confusion, Formula (2.30) is usually denoted as:

$$\mathbf{p}(\mathbf{x}) = \sum_i \mathbf{P}_i N_i(\mathbf{x}) \quad (2.31').$$

Based on basic properties of $N_{D^k}(\mathbf{x})$, we can know that:

- $\mathbf{p}(\mathbf{x})$ is a polynomial over each tile of the given plane partition
- $\mathbf{p}(\mathbf{x})$ is $s - 1$ times continuously differentiable in $\mathbf{t}_r$ direction if there are s $\mathbf{t}_r$ vectors in the vector group D^k .

• $\mathbf{p}(\mathbf{x})$ is in convexity formed by the mesh M .

• Pulling a control vertex $\mathbf{P}_i$, only a local region of $\mathbf{p}(\mathbf{x})$ changes its shape.

Obviously, uniform B-spline surfaces are unique cases of box spline surfaces.

2.6.2 Subdivision of Box Spline Surfaces

Let $D = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_k\}$ and the basic function defined by the vector group be $N^1(\mathbf{x})$. Its support mesh is G_D , and the grid Γ^1 is obtained after translating G_D to each knot. The box spline surface defined by $N^1(\mathbf{x})$ is:

$$\mathbf{p}(\mathbf{x}) = \sum_i \mathbf{P}_i^1 N_i^1(\mathbf{x}) = \sum_i \mathbf{P}_i^1 N^1(\mathbf{x} - \mathbf{i}).$$

We now make a halving refinement, i.e., add a midpoint for each edge of the grid Γ^1 , and then link new vertices and old vertices to form a grid Γ^2 topology isomorphic to Γ^1 . So, we can calculate a new control vertex set $\{\mathbf{P}_j^2 | \mathbf{j} \in Z^2/2\}$ and make

$$\mathbf{p}(\mathbf{x}) = \sum_j \mathbf{P}_j^2 N^1[2(\mathbf{x} - \mathbf{j})], \quad \mathbf{j} \in Z^2/2.$$

Rules according to which we calculate new $\mathbf{P}_j^2$ are as follows: [94]

$$\begin{aligned} d_j^0 &= \begin{cases} 0 & \mathbf{j} \notin Z^2 \\ \mathbf{P}_j^1 & \mathbf{j} \in Z^2 \end{cases} \\ d_j^r &= d_j^{r-1} + d_{j-\mathbf{t}_r/2}^{r-1}, \quad r = 1, \dots, k \\ \mathbf{P}_j^2 &= 2^{-(k-2)} d_j^k. \end{aligned} \quad (2.32)$$

Based on the above recursive refinement formula, we can find the relationship between basic functions $N_D(\mathbf{x} - \mathbf{i})$ and $N_D(2(\mathbf{x} - \mathbf{j}))$, where $\mathbf{i} \in \mathbf{Z}^2$, $\mathbf{j} \in \mathbf{Z}^2/2$. As for a box spline basic function $N^1(\mathbf{x})$, we have

$$N^1(\mathbf{x}) = \sum_i \mathbf{P}_i^1 N_i^1(\mathbf{x}) = \sum_i \mathbf{P}_i^1 N^1(\mathbf{x} - \mathbf{i}),$$

where $\mathbf{P}_i^1 = \begin{cases} 1 & \mathbf{i} = (0, 0) \\ 0 & \text{other} \end{cases}$

By the recursive refinement formula (2.32), we have

$$N_D(\mathbf{x}) = \sum_j c_j N_D[2(\mathbf{x} - \mathbf{j})], \quad (2.33)$$

where $\mathbf{j} \in \text{knot}(G_D)/2$, $\text{knot}(G_D)$ is the knot set of the mesh G_D . When $D = \{s_0, s_1, s_2, s_3\}$, the recursive process is shown in Fig. 2.35.

Based on Fig. 2.35, we know that the recursive process can be executed directly on the initial grid L_D . Figure 2.36 gives the recursive process when $D = \{s_0, s_0, s_1, s_1, s_2, s_2\}$. It should be known that the result is not connected with the direction order adopted. For example, to obtain the result in Fig. 2.36c, we can also execute in the following steps: once in s_2 direction, twice in s_0 direction, twice in s_1 direction, once in s_2 direction.

In Figs. 2.35 and 2.36, those coefficients should lastly multiply $2^{-(k-2)}$, i.e., the third in Formula (2.32) should be executed.

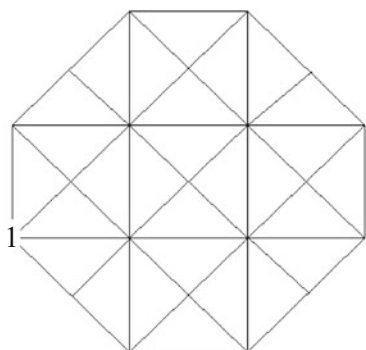
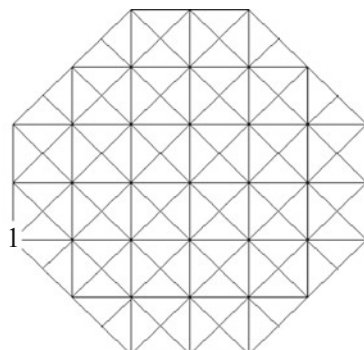
2.6.3 Generating Function of Box Spline

To obtain those coefficients c_j in Formula (2.33), we can use those recursive rules in Formula (2.32) and we can also use the generating function [95] introduced in this section. Assume vector group $D = \{s_0, s_1\}$. Obviously, s_0, s_1 are linear independent. Similar to subdivision Formula (2.21) of the univariate basic function, we can get subdivision formula of bivariate box spline:

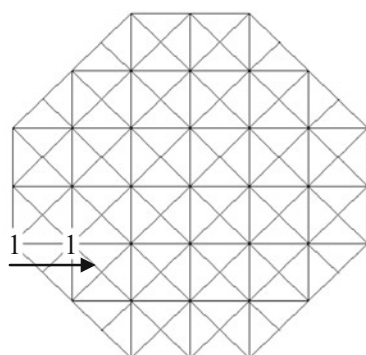
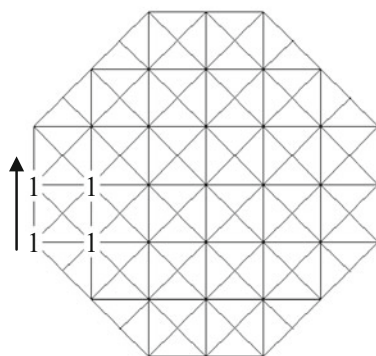
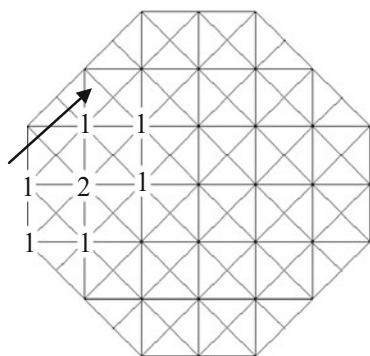
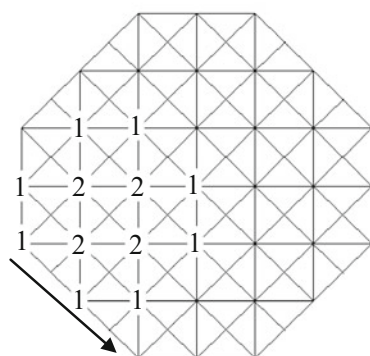
$$\begin{aligned} N_D(\mathbf{x}) &= \sum_{i_1=0}^1 \sum_{i_2=0}^1 N_D(2\mathbf{x} - \mathbf{i}) \\ &= N_D(2\mathbf{x}) + N_D(2\mathbf{x} - s_0) + N_D(2\mathbf{x} - s_1) + N_D(2\mathbf{x} - s_2). \end{aligned} \quad (2.34)$$

So, we can define a generating function for $N_D(\mathbf{x})$:

$$C(\mathbf{z}) = \prod_{j=0}^1 (1 + z_j) = (1 + z_0)(1 + z_1) = 1 + z_0 + z_1 + z_0 z_1.$$

(a) Initial Grid G_D 

(b) Subdivision once

(c) Once in s_0 direction(d) Once in s_1 direction(e) Once in s_2 direction(f) Once in s_3 direction**Fig. 2.35** Recursive process defined by Formula (2.32)

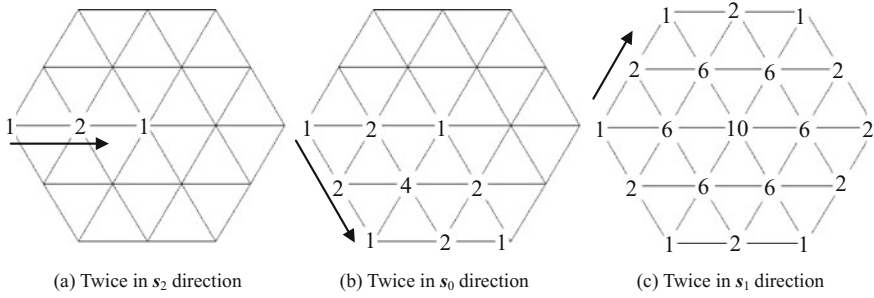


Fig. 2.36 Recursive process on G_D , $D = \{s_0, s_0, s_1, s_1, s_2, s_2\}$

Obviously, coefficients of extension expression of $C(z)$ are consistent with coefficients of expression (2.34), i.e., members z_0 and z_1 of z are grid knots and coefficient of every term is c_j in Formula (2.33). Let

$$z^{t_k} = z_0^{t_k[0]} z_1^{t_k[1]}.$$

We can give the following theorems:

Theorem 2.4 Let $C(z)$ be the generating function of $f(x)$. If

$$g(x) = \int_0^1 f(x - tt_k) dt,$$

then $g(x)$ has the generating function: $\frac{1}{2}C(z)(1 + z^{v_k})$.

Proof By hypothesis,

$$\begin{aligned} g(x) &= \int_0^1 f(x - tt_k) dt \\ &= \int_0^1 \left(\sum_i c_i f[2(x - tt_k) - i] \right) dt \\ &= \frac{1}{2} \int_0^2 \left[\sum_i c_i f(2x - tt_k - i) \right] dt \\ &= \frac{1}{2} \int_0^1 \left[\sum_i c_i f(2x - tt_k - i) + \sum_i c_i f(2x - tt_k - i - t_k) \right] dt \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{2} \int_0^1 \left[\sum_i (c_i + c_{i-t_k}) f(2\mathbf{x} - t\mathbf{t}_k - \mathbf{i}) \right] dt \\
&= \frac{1}{2} \left[\sum_i (c_i + c_{i-t_k}) \int_0^1 f(2\mathbf{x} - t\mathbf{t}_k - \mathbf{i}) dt \right] \\
&= \frac{1}{2} \sum_i [(c_i + c_{i-t_k}) g(2\mathbf{x} - \mathbf{i})]
\end{aligned}$$

So, the generating function of $g(\mathbf{x})$ is exactly $\frac{1}{2} C(\mathbf{z})(1 + \mathbf{z}^{v_k})$. #

Based on Theorem 2.4, we know that the generating function associated with the subdivision Formula (2.33) is:

$$C_D(\mathbf{z}) = 4 \prod_{i=0}^{k-1} \frac{1}{2} (1 + \mathbf{z}^{t_k}).$$

When $D = \{s_0, s_1, s_2\}$,

$$\begin{aligned}
C_D(\mathbf{z}) &= \frac{1}{2} (1 + z_0)(1 + z_1)(1 + z_0 z_1) \\
&= \frac{1}{2} + \frac{1}{2} z_0 + \frac{1}{2} z_1 + z_0 z_1 + \frac{1}{2} z_0^2 z_1 + \frac{1}{2} z_0 z_1^2 + \frac{1}{2} z_0^2 z_1^2.
\end{aligned}$$

2.7 Subdivision Mask of Box Spline Surface

Based on the subdivision Formula (2.33), we can deduce subdivision mask of box spline surfaces. We firstly investigate quadric three-directional box spline surfaces [1] whose basic function is $N_D(\mathbf{x})$ with the vector group $D = \{\mathbf{e}_1, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_3\}$. Coefficients in subdivision Formula (2.33) are shown in Fig. 2.34. We regard the grid start vertex of $N_D(\mathbf{x})$ as (0,0). After translating $N_D(\mathbf{x})$ and let its grid start vertex be (i, j) , we denote $N_D(\mathbf{x})$ by $N_{i,j}(\mathbf{x})$. Obviously,

$$N_D(\mathbf{x}) = N_{0,0}(\mathbf{x}).$$

In this case, box spline surface (2.31') can be written as:

$$p(\mathbf{x}) = \sum_i \sum_j \mathbf{P}_{i,j} N_{i,j}(\mathbf{x}). \quad (2.35)$$

Now, we refine the plane grid formed by G_D and knots of the grid of being refined are still numbered by integers. On the new grid, the box spline whose start vertex is

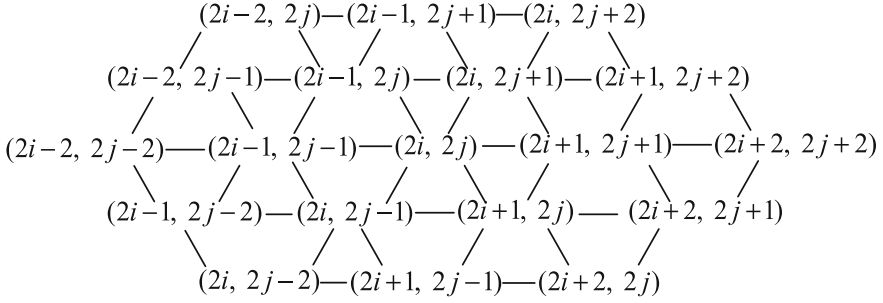


Fig. 2.37 Subdivision coefficient matrix of quadric three-directional box spline surface

(i, j) is denoted by $n_{i,j}^1(\mathbf{x})$. So, Formula (2.33) can be rewritten as:

$$N_{i,j}(\mathbf{x}) = \sum_{s=0}^2 \sum_{t=0}^2 \sum_{l=0}^2 \beta_{s+l,t+l} N_{2i-2+s+l, 2j-2+t+l}^1(\mathbf{x}). \quad (2.36)$$

It is easy to know that, when

$$\mathbf{x} = \alpha_0 \mathbf{t}_0 + \cdots + \alpha_{k-1} \mathbf{t}_{k-1}, \alpha_i \in \{0, 1, 2\},$$

different vectors $[\alpha_0, \cdots, \alpha_{k-1}]$ can correspond to the same $\mathbf{x}$. Based on this reason, in the extended expression of (2.36), if the frequency of $\beta_{s+l,t+l} N_{s+l,t+l}^1(\mathbf{x})$ is a larger one, we only count one for $\beta_{s+l,t+l} N_{s+l,t+l}^1(\mathbf{x})$. Consequently, there are only 19 terms in the extended expression of (2.36). Let

$$N_{2i-2+s+l, 2j-2+t+l}^1 \rightarrow (2i-2+s+l, 2j-2+t+l).$$

Consequently, we can get the subdivision coefficient matrix of the control point $\mathbf{P}_{i,j}$ of the quadric three-directional box spline surfaces (2.34) and the matrix is shown in Fig. 2.37. The coefficient of every basic function in Fig. 2.37 is the value on the corresponding position of Fig. 2.36c.

Based on Fig. 2.37, we can know that there are seven control vertices whose subdivision coefficient matrixes include the basic function $N_{2i-2, 2j-2}^1$:

$$\mathbf{P}_{i,j}, \mathbf{P}_{i-1,j-1}, \mathbf{P}_{i-2,j-2}, \mathbf{P}_{i,j-1}, \mathbf{P}_{i,j-2}, \mathbf{P}_{i-1,j}, \mathbf{P}_{i-2,j-1}.$$

Consequently, the new mesh vertex $\mathbf{P}_{2i-2, 2j-2}^1$ is a linear combination of the above seven control vertices and coefficient of every vertex is shown in Fig. 2.38a. Similarly, $\mathbf{P}_{2i-1, 2j-2}^1$ is a linear combination of four control vertices: $\mathbf{P}_{i,j}, \mathbf{P}_{i-1,j-1}, \mathbf{P}_{i,j-1}, \mathbf{P}_{i-1,j-2}$ and their coefficients are shown in Fig. 2.38b.

It is easy to know that the new vertex corresponding to a basic function in Fig. 2.37 has a subdivision mask in Fig. 2.38. Each vertex in the old mesh corresponds to a

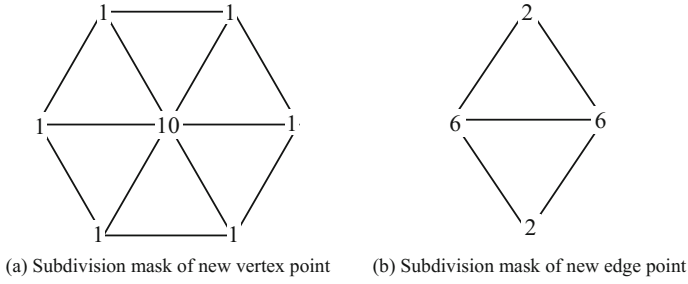


Fig. 2.38 Masks of quadric three-directional box spline surfaces

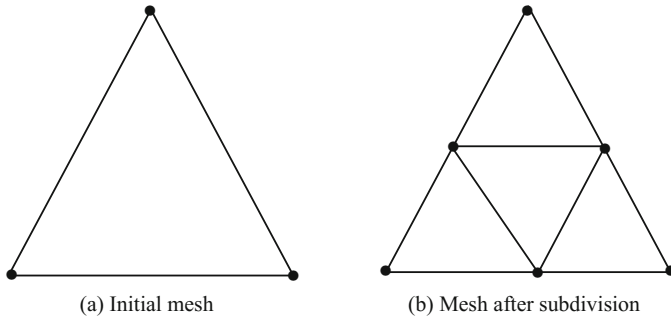


Fig. 2.39 Mesh topology of subdivision about quadric three-directional box spline surface

new vertex point, and each edge in the old mesh corresponds to a new edge point. The topology relation between the new mesh and the old mesh is shown in Fig. 2.39.

We now research the Powell-Sabin spline surface [96]. The vector group of its basic function is $D = \{e_1, e_2, e_3, e_4\}$. In this case, Formula (2.33) can be rewritten as:

$$N_{i,j} = \sum_{s=0}^1 \sum_{t=0}^1 \sum_{m=0}^1 \sum_{n=0}^1 \beta_{s+m+n,t+m-n} N_{2i-2+s+m+n,2j-2+s+m-n}^1. \quad (2.37)$$

In the extended expression of (2.31), if the frequency of $\beta_{s+m+n,t+m-n} N_{2i-2+s+m+n,2j-2+s+m-n}^1$ is a larger one, we only count one for $\beta_{s+m+n,t+m-n} N_{2i-2+s+m+n,2j-2+s+m-n}^1$. Consequently, there are only 12 terms in the extended expression of (2.37). Through the definition of the subdivision coefficient matrix, we can know that the new mesh vertex $P_{2i-2,2j-2}^1$ is a linear combination of these three control vertices: $P_{i,j}$, $P_{i-1,j-1}$, $P_{i-1,j}$. Their coefficients are shown in Fig. 2.40. It is easy to know that the new vertex corresponding to a basic function in right expression of Eq. (2.37) has a subdivision mask in Fig. 2.41.

Only concerned with topology structure, subdivision of the Powell-Sabin spline surface is similar to subdivision of the uniform bi-quadric B-spline surface. The process is shown in Fig. 2.42a, c. Since there are four directions, we can transform

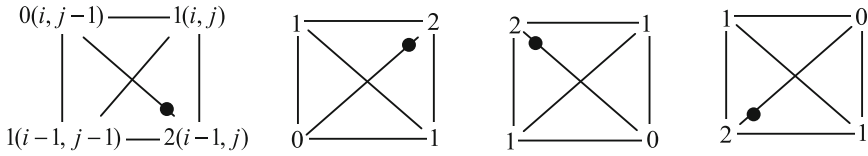


Fig. 2.40 Subdivision masks of Powell-Sabin spline surfaces

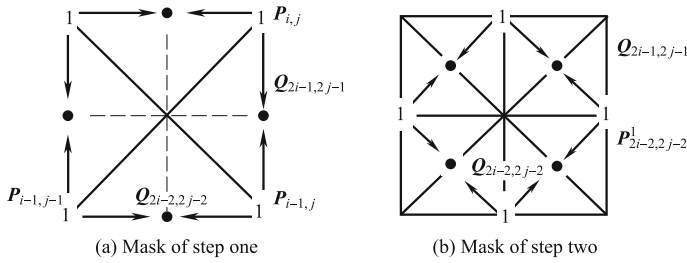


Fig. 2.41 Split of subdivision mask of Powell-Sabin spline surface

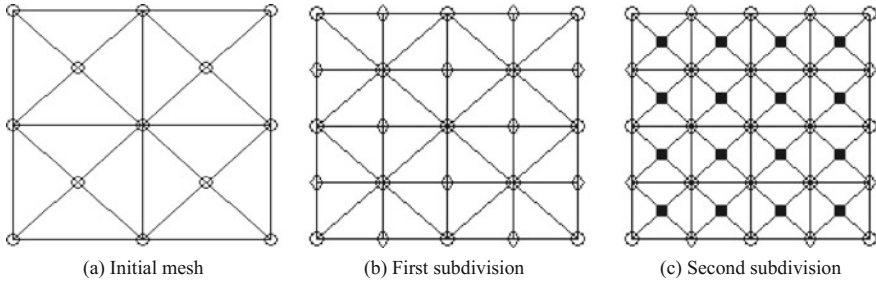


Fig. 2.42 Subdivision by split masks of Powell-Sabin spline surface

the mesh shown in Fig. 2.42a to the mesh shown in Fig. 2.42c by two steps, i.e., a subdivision shown in Fig. 2.40 can be split into two steps shown in Fig. 2.41. Figure 2.42 gives the full process of subdivision by split masks. It should be noticed that every coefficient in Fig. 2.40 should be multiplied by $1/4$ and every coefficient in Fig. 2.31 should be multiplied by $1/2$.

You are probably thinking about why these two subdivision masks in Fig. 2.42 can be united to the subdivision masks in Fig. 2.41. For the new vertex $P_{2i-2,2j-2}^1$ (see Fig. 2.41),

$$\begin{aligned} Q_{2i-2,2j-2} &= \frac{1}{2}P_{i-1,j-1} + \frac{1}{2}P_{i-1,j}, & Q_{2i-1,2j-1} &= \frac{1}{2}P_{i-1,j} + \frac{1}{2}P_{i,j} \\ \therefore p_{2i-2,2j-2}^1 &= \frac{1}{4}P_{i-1,j} + \frac{1}{2}P_{i-1,j} + \frac{1}{4}P_{i,j}. \end{aligned}$$

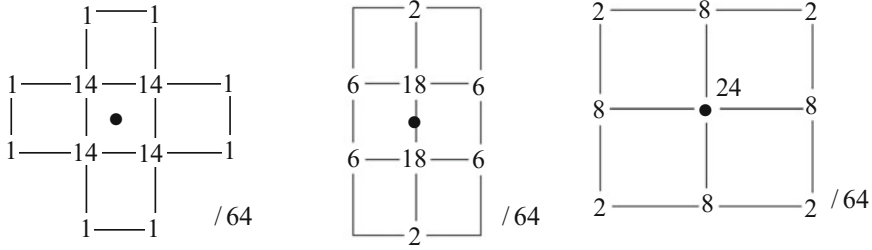


Fig. 2.43 Face, edge, and vertex masks of C^4 box spline surface

Consequently, the two subdivision masks in Fig. 2.41 can indeed be united to the subdivision masks in Fig. 2.40. Masks given in Fig. 2.41 are smaller than masks in Fig. 2.40.

We now research the quadric four-directional box spline surface [97]. The vector group of its basic function is $D = \{s_0, s_0, s_1, s_1, s_2, s_2, s_3, s_3\}$. By the subdivision coefficient matrix, we can get its subdivision masks shown in Fig. 2.43.

Only concerned with topology structure, subdivision of C^4 box spline surface is similar to subdivision of the uniform bi-cubic B-spline surface. However, the large support of the masks makes the implementation of the subdivision of C^4 box spline surface difficult. We decompose the C^4 smoothing operator into two masks as shown in Fig. 2.44. The face mask calculates positions of new added vertices with valance 4 in the new mesh. The vertex mask calculates positions of vertices with valance 8 in the new mesh. In the old mesh, valances of these vertices are 4 or 8. Figure 2.45 gives meshes on three consecutive subdivision levels, $j - 1$, j , and $j + 1$. The subdivision is executed by masks in Fig. 2.44.

Let M be a mesh. We obtain a mesh M^1 after we subdivide M once with the masks in Fig. 2.43.

We obtain another mesh $(M^1)'$ after we subdivide M twice with the masks in Fig. 2.44. Why is $(M^1)'$ the same as M^1 ? In order to ask the question, we label those vertices of the mesh in the subdivision level j in Fig. 2.43. These labels are shown in Fig. 2.46.

At the level $j + 1$, the value of vertex u is computed using the vertex mask in Fig. 2.44:

$$u^{j+1} = \frac{4}{8}u^j + \frac{1}{8}(f^j + g^j + k^j + j^j).$$

The value of u^j is calculated using the face mask in Fig. 2.44:

$$u^j = \frac{1}{4}(f^{j-1} + g^{j-1} + k^{j-1} + j^{j-1}).$$

The value of f^j , g^j , k^j , and j^j is calculated using the vertex mask in Fig. 2.44. For f^j , we have

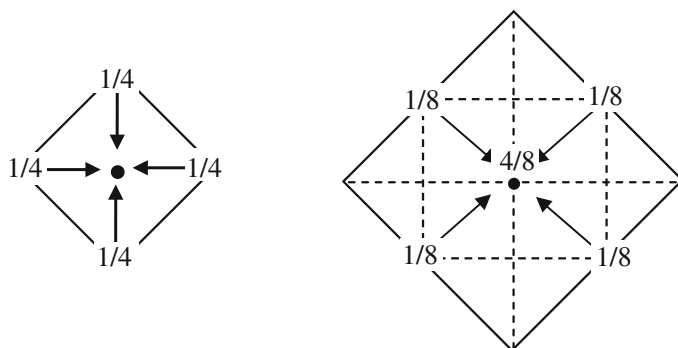


Fig. 2.44 Factorized face and vertex mask of C^4 box spline surface

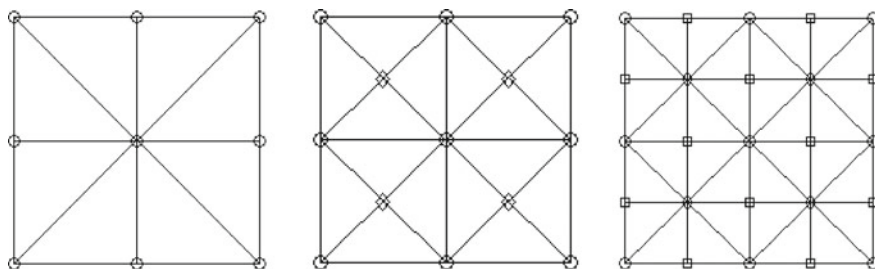
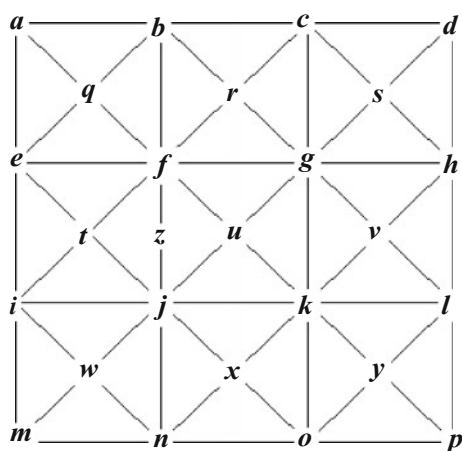


Fig. 2.45 Three consecutive subdivision levels, $j - 1$, j , and $j + 1$

Fig. 2.46 Labels of vertices of the mesh in the subdivision level j



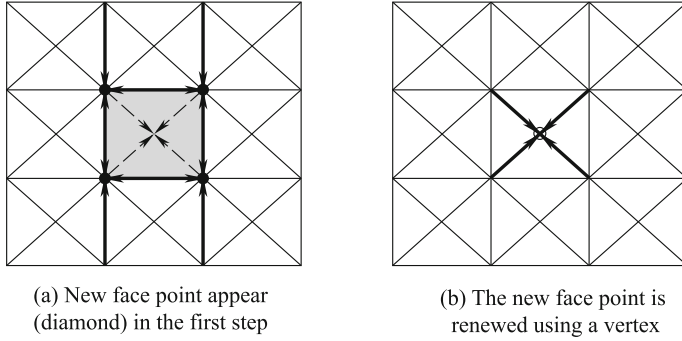


Fig. 2.47 Decomposition of the face mask of C^4 box spline surface

$$f^j = \frac{4}{8}f^{j-1} + \frac{1}{8}(b^{j-1} + g^{j-1} + j^{j-1} + e^{j-1}).$$

Substituting the formulas of vertices u^j , f^j , g^j , k^j , and j^j into the equation for u^{j+1} , we obtain the face mask in Fig. 2.43, which gives the value of u^{j+1} in terms of the values of the vertices of the mesh at level $j - 1$. Consequently, the face mask is decomposed into an application of face and vertex masks at level $j - 1$, followed by an application of vertex mask at level j . Figure 2.47 shows the flow of computations used for updating vertex u .

At level $j + 1$, the value of vertex z is computed using the face mask in Fig. 2.44:

$$z^{j+1} = \frac{1}{4}(f^j + u^j + j^j + t^j).$$

The values of f^j and j^j are calculated using the vertex mask, and the values of u^j and t^j are calculated using the face mask. The calculation for j^j is similar to the calculation of f^j , while the calculation for t^j is similar to the calculation of u^j . Substituting the formulas of f^j , u^j , j^j , and t^j into equation for z^{j+1} , we obtain the edge mask. Consequently, the edge mask is decomposed into an application of face and vertex masks at level $j - 1$, followed by an application of face mask at level j . Figure 2.48 shows the flow of computations used for updating vertex z .

At level $j + 1$, the value of vertex f is calculated using the vertex mask:

$$f^{j+1} = \frac{4}{8}f^j + \frac{1}{8}(r^j + u^j + t^j + q^j).$$

The values of f^j , r^j , u^j , t^j , and q^j are calculated using the vertex mask for calculating f^j and the face mask for calculating r^j , u^j , t^j , and q^j . The calculation for r^j , t^j , and q^j is similar to the calculation of u^j . Again, substituting the formulas of f^j , r^j , u^j , t^j , and q^j into the equation for f^{j+1} , we obtain the vertex mask. Consequently, the vertex mask is decomposed into an application of face and vertex

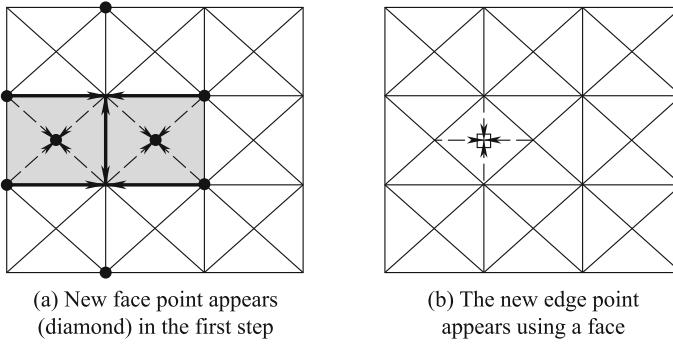


Fig. 2.48 Decomposition of the edge mask of C^4 box spline surface

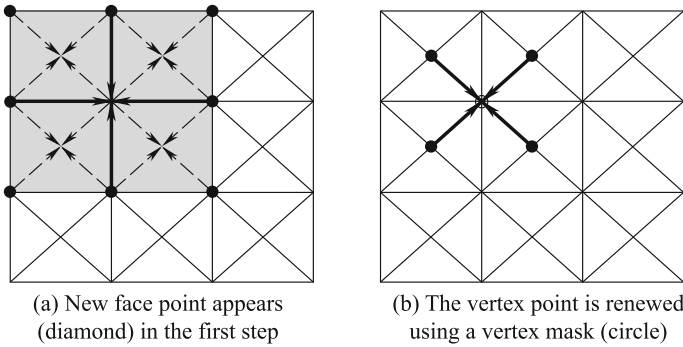


Fig. 2.49 Decomposition of the edge mask of C^4 box spline surface

masks at level $j-1$, followed by an application of vertex mask at level j . Figure 2.49 shows the flow of computations used for updating vertex f .

Remarks

This chapter gives refinement rules for control meshes of spline surfaces. The refinement has two meanings: (1) double knot intervals and obtain a new spline function space; (2) densify vertices of old control meshes and obtain new control meshes. The refinement of control meshes of spline surfaces is also called subdivision of spline surfaces. We firstly obtain refinement rules of basic functions, and then deduce subdivision rules of spline surfaces by refinement rules of basic functions. For four-directional box spline surfaces, we decompose subdivision masks to obtain smaller subdivision masks because of subdivision masks directly derived from refinement rules of basic functions are large. There are several methods to deduce refine-

ment rules of basic functions: the de-Boor algorithm and the convolution definition for B-spline basic functions, the recursive method, and the generating function for box spline basic functions. It should be noticed that B-spline is a unique case of box splines.

Exercises

1. For the knot vector $U = [\dots, -3, -2, -1, 0, 1, 2, 3, \dots]$, please write program codes to compute the B-splines: $N_{0,2}(u)$, $N_{0,3}(u)$, $N_{0,4}(u)$, $N_{0,5}(u)$ using the formula (2.1) and render the curves of these spline functions.
2. Assume that $P_0 = [0, 0]$, $P_1 = [1, 1]$, $P_3 = [3, 0]$. The cubic spline curve is $p(u) = \sum_{i=0}^3 P_i N_{i,3}(u)$, where $U = [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$. Compute its point using the de-Boor algorithm (see Formula (2.7)) and render the curve.
3. For the curve in the above item, double the knot interval $[01]$ as $[0 \ 0.51]$ and then obtain the knot vector $U^1 = [0 \ 0 \ 0 \ 0 \ 0.5 \ 1 \ 1 \ 1 \ 1]$. Please compute the new control vertices for the curve.
4. For the curve in the item 2, double knot intervals with nonzero length k times and then the knot vector $U^k = [0, 0, 0, 0, u_1, u_2, \dots, u_n, 1, 1, 1, 1]$. Please compute the new control vertices after every doubling step.
5. Using Formula (2.11), write program codes to subdivide the control polygon of a non-uniform cubic B-spline curve.
6. Using Formula (2.13)~(2.15), write program codes to subdivide the control mesh of a non-uniform bi-cubic B-spline surface.
7. For uniform cubic B-spline, when double knots, please deduce the formula:

$$N_{i,3}(u) = \frac{1}{8}N_{2i-2,3}^1(u) + \frac{4}{8}N_{2i-1,3}^1(u) + \frac{6}{8}N_{2i,3}^1(u) + \frac{4}{8}N_{2i+1,3}^1(u) + \frac{1}{8}N_{2i+2,3}^1(u).$$

8. See Definition 2.4. Write program codes to construct the grid G_D^3 according to vectors in D^3 .

Subdivision Surface Modeling Technology

Liao, W.; Liu, H.; Li, T.

2017, XVI, 307 p. 193 illus., Hardcover

ISBN: 978-981-10-3514-2