



Boundary Conditions for High-Quality Loop Subdivision Surfaces

Masatake Higashi¹, Shinji Seo¹, Masakazu Kobayashi¹ and Tetsuo Oya²

¹Toyota Technological Institute, higashi_sd04045_kobayashi@toyota-ti.ac.jp

²Keio University, oya@sd.keio.ac.jp

ABSTRACT

In this report, we propose a method which generates a high-quality subdivision surface from mesh data satisfying boundary conditions. We estimate second cross-derivatives at boundaries and construct a control mesh so that the interpolated shape is supposed to be trimmed from an extended shape at the boundary. At sharp features, we introduce a subdivision rule in a similar manner to Hoppe et al., which makes a crease or boundary correspond with a specified curve.

Keywords: boundary condition, subdivision surface, cross derivative.

DOI: 10.3722/cadaps.2011.593-602

1 INTRODUCTION

Tensor product surfaces such as non-uniform B-spline surfaces and Bézier patches have been widely used [6] to represent free-form surfaces which compose industrial products such as automotive bodies and electric appliances. They can represent high-quality surfaces, but to connect them to each other with continuity is rather difficult. On the other hand, subdivision surfaces for arbitrary meshes have been used in the field of computer graphics [5] due to their ability to handle irregular meshes and automatic satisfaction of continuity [4,10,14]. A triangular subdivision surface was first introduced by Loop [12] and subdivision rules were presented. Next, Hoppe et al. [9] proposed a method to automatically determine the topological type of the surface including the presence and location of sharp features such as creases, corners and darts. Then, Levin [11] introduced a method which interpolates nets of curves according to a combined subdivision scheme, and sharp feature control was introduced by Bierman et al. [1].

Tab. 1 shows subdivision rules for a triangle mesh. The first row shows rules for inner vertices proposed by Loop and the second row shows those for vertices on sharp features proposed by Hoppe et al. The main characteristics of Hoppe's approach are as follows.

1. The vertices of the control mesh at sharp features are determined from crease curves and those at corners are set to correspond with the input vertices. As a result, surfaces on the both sides of a crease can share common vertices.
2. The subdivision rule at a crease is different from that at an inner vertex of a surface. Hence, there is no contradiction on the boundary curve; however the boundary condition at the crease is fixed, there is no freedom to determine the cross derivatives, which are used for

	Odd vertex	Even vertex
Inner vertex (Loop)		
Feature Vertex (Hoppe)		

$$\beta = \frac{1}{n} \left(\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right)$$

Tab. 1: Subdivision rules.

tensor surfaces. As a result, this generates a flat shape along the crease and unfavorable effects on the inner surface.

In our approach, to represent a high-quality surface, we set the boundary conditions for a subdivision surface and introduce a method to determine a control mesh to satisfy these conditions, along with a new rule for the crease or boundary curve. The concepts of our approach are:

1. We determine a control mesh at a feature (crease or boundary), such that the mesh can be trimmed at this feature from the mesh representing its extended shape.
2. The mesh corresponding to the extended shape is estimated from the second cross derivative along the feature and represented by extended vertices along the feature.
3. We introduce a subdivision rule at the feature which makes the limit curve at the crease correspond with a specified input curve, due to the curve obtained by the subdivision using the extended mesh not necessarily corresponding with the input curve.

2 CROSS DERIVATIVES AT FEATURE OF SUBDIVISION SURFACE

2.1 Parametric Line and its Cross Derivatives

We first examine a parametric line in the cross boundary direction of a subdivision surface. When the mesh at the boundary is regular (the valence is four along the boundary), an equation for the parametric line can be constructed using the weighting functions of a box spline [2,3]. Fig. 1 shows a

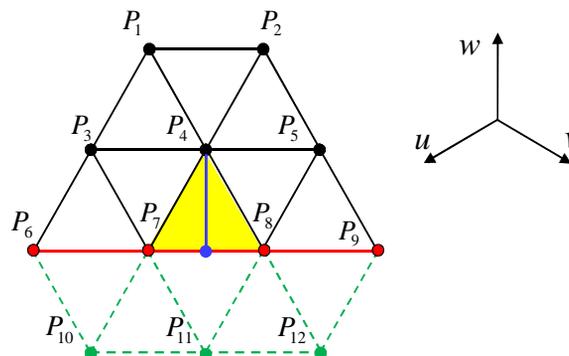


Fig. 1: Parametric line across boundary.

parametric line going across a boundary as a blue line. Here the red line is a boundary and green dashed lines show an extended mesh along the boundary [7,13]. Parametric line $\mathbf{r}(t)$ is represented using control vertices P_i and weighting functions ϕ_i , and the parameter t is set to $t=u=v$ for surface parameters u, v and w :

$$\mathbf{r}(t) = \sum_{i=1}^{12} \phi_i \mathbf{P}_i, \quad (2.1)$$

$$\begin{aligned} \phi_1, \phi_2 &= \frac{1}{12}(1-2t)^3, \\ \phi_3, \phi_5 &= \frac{1}{12} \left(2(1-2t)^3 - (1-2t)^4 + 6(1-2t)^2 t - 2(1-2t)^3 t + 6(1-2t)t^2 + 2t^3 - 2(1-2t)t^3 - t^4 \right), \\ \phi_4 &= \frac{1}{12} \left(1 + 4(1-2t) + 6(1-2t)^2 - 4(1-2t)^3 - (1-2t)^4 + 2t + 6(1-2t)t - 6(1-2t)^2 t \right. \\ &\quad \left. - 2(1-2t)^3 t - 12(1-2t)t^2 - 4t^3 + 4(1-2t)t^3 + 2t^4 \right), \\ \phi_6, \phi_9 &= \frac{1}{12} \left(2(1-2t)t^3 + t^4 \right), \\ \phi_7, \phi_8 &= \frac{1}{12} \left(1 + 2(1-2t) - 4(1-2t)^3 + 2(1-2t)^4 + 4t + 6(1-2t)t - 12(1-2t)^2 t + 4(1-2t)^3 t \right. \\ &\quad \left. + 6t^2 - 6(1-2t)t^2 - 4t^3 - 2(1-2t)t^3 - t^4 \right), \\ \phi_{10}, \phi_{11} &= \frac{1}{12}(1-2t)^3, \\ \phi_{12}, \phi_{13} &= \frac{1}{12} \left(2(1-2t)^3 - (1-2t)^4 + 6(1-2t)^2 t - 2(1-2t)^3 t + 6(1-2t)t^2 + 2t^3 - 2(1-2t)t^3 - t^4 \right). \end{aligned} \quad (2.2)$$

Differentiating Eqn. (2.1), we obtain the first derivative of the parametric line, and at the boundary (when $t=0.5$) represented by a blue dot, the cross derivative is

$$\dot{\mathbf{r}}(0.5) = \frac{1}{4} \left(3\mathbf{P}_{11} + \frac{1}{2}(\mathbf{P}_{10} + \mathbf{P}_{12}) \right) - \frac{1}{4} \left(3\mathbf{P}_4 + \frac{1}{2}(\mathbf{P}_3 + \mathbf{P}_5) \right). \quad (2.3)$$

Additionally at this point, the second derivative is

$$\ddot{\mathbf{r}}(0.5) = 2.5(\mathbf{P}_4 + \mathbf{P}_{11}) + 0.75(\mathbf{P}_3 + \mathbf{P}_5 + \mathbf{P}_{10} + \mathbf{P}_{12}) - 3.75(\mathbf{P}_7 + \mathbf{P}_8) - 0.25(\mathbf{P}_6 + \mathbf{P}_9). \quad (2.4)$$

Replacing the control vertices with new points, \mathbf{a} , \mathbf{b} and \mathbf{c} , we get

$$\begin{aligned} \mathbf{a} &= \frac{5}{8}\mathbf{P}_4 + \frac{3}{8} \left(\frac{\mathbf{P}_3 + \mathbf{P}_5}{2} \right), \\ \mathbf{b} &= \frac{15}{16} \left(\frac{\mathbf{P}_7 + \mathbf{P}_8}{2} \right) + \frac{1}{16} \left(\frac{\mathbf{P}_6 + \mathbf{P}_9}{2} \right), \\ \mathbf{c} &= \frac{5}{8}\mathbf{P}_{11} + \frac{3}{8} \left(\frac{\mathbf{P}_{10} + \mathbf{P}_{12}}{2} \right), \end{aligned} \quad (2.5)$$

and

$$\ddot{\mathbf{r}}(0.5) = 4(\mathbf{a} + \mathbf{c} - 2\mathbf{b}), \quad (2.6)$$

where Fig. 2 shows the relationship between these points and the second derivative. The derivative is obtained as the second difference of the edges multiplied by 4.

This differs to Hoppe's rule where the extended mesh is generated such that the triangle which includes the extended vertex makes a parallelogram with the opposite inner triangle, for example, P_{11} for the quad $P_{11}P_8P_4P_7$ in Fig. 1 [7]. Thus:

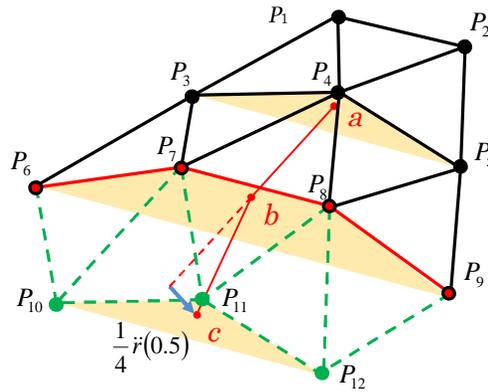


Fig. 2: Second derivative of parametric line.

$$\begin{aligned}
 P_{10} &= P_6 + P_7 - P_3, \\
 P_{11} &= P_7 + P_8 - P_4, \\
 P_{12} &= P_8 + P_9 - P_5,
 \end{aligned}
 \tag{2.7}$$

and the second derivative is obtained by replacing P10, P11 and P12:

$$\ddot{r}(0.5) = \frac{1}{2}(P_6 + P_9) - \frac{1}{2}(P_7 + P_8)
 \tag{2.8}$$

In Eqn. (2.8), the second derivative is determined from the control vertices on the boundary only; it is not possible to control the surface shape at the boundary.

2.2 Relationship between Parametric Lines and a Control Mesh

Next, we discuss the relationship between parametric lines and a control mesh. A parametric line for a box spline, for example $w=0$ in Fig. 1, is a quartic Bezier curve which can be represented as a quasi-quartic B-spline as shown in Fig. 3, because adjacent Bezier curves are C^2 at the junction and the intersection of extended edges which are next to the junction edge generates a new control point q_i .

We obtain a quasi-quartic B-spline using q_i and q_i^M which is the mid control point of the Bezier curve. These control points of the quasi-quartic B-spline are calculated from the vertices of the control mesh:

$$\begin{aligned}
 q_i &= \frac{8}{12}P_i + \frac{1}{12}(P_{i,2} + P_{i,3} + P_{i,5} + P_{i,6}), \\
 q_i^M &= \frac{2}{6}(P_i + P_{i,4}) + \frac{1}{6}(P_{i,3} + P_{i,5}).
 \end{aligned}
 \tag{2.9}$$

Now, let the midpoint of P_i and $P_{i,4}$ be $p_{i,04}$, and similarly $p_{i,23}$ for $P_{i,2}$ and $P_{i,3}$ and $p_{i,56}$ for $P_{i,5}$ and $P_{i,6}$, then we get:

$$\begin{aligned}
 q_i &= \frac{4}{6}P_i + \frac{1}{6}(p_{i,23} + p_{i,56}), \\
 q_i^M &= \frac{4}{6}p_{i,04} + \frac{1}{6}(P_{i,3} + P_{i,5}),
 \end{aligned}
 \tag{2.10}$$

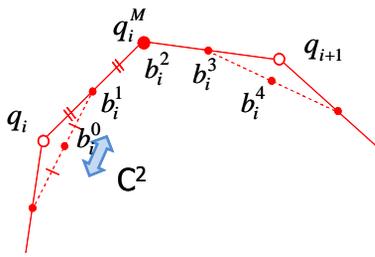


Fig. 3: Parametric line represented as a quasi-quartic B-spline.

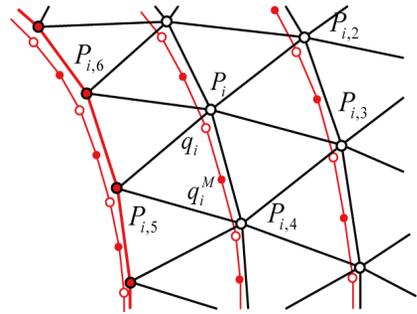
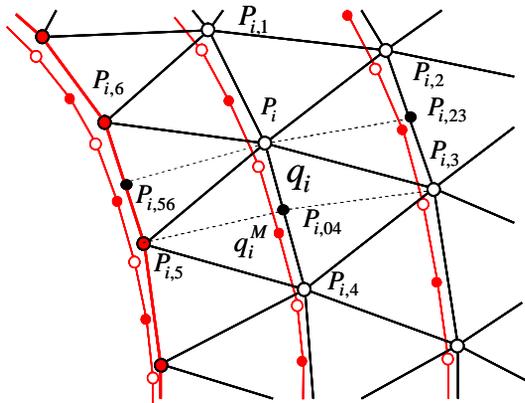
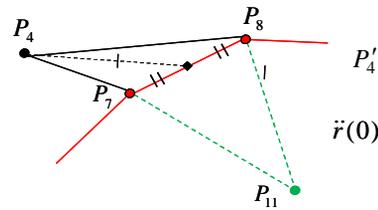


Fig. 4: Relationship between parametric lines and the control mesh.



(a) Parametric lines along the boundary direction



(b) Second derivative

Fig. 5: Cubic B-spline curve for control points of the quasi spline.

and it can be seen that both control points of the quasi-quartic B-spline are generated by summation of control mesh vertices weighted at a ratio of 1:4:1. This ratio is the same as that for cubic B-spline control points, thus, subdivision surfaces can be approximated as cubic splines in the cross boundary direction. However, it is important to note that there is a pair of adjacent B-spline curves with opposite triangle orientation as shown in Fig. 5a, these are not independent of each other. No such relationship exists when using a tensor surface.

As the parametric line moves across the surface in the cross boundary direction, the second derivative becomes

$$\begin{aligned} \ddot{r}(0) &= (P_4 + P_{11}) - (P_7 + P_8) \\ &= \left(P_{11} - \frac{1}{2}(P_7 + P_8) \right) - \left(\frac{1}{2}(P_7 + P_8) - P_4 \right). \end{aligned} \tag{2.11}$$

Fig. 5b shows this relationship where the second derivative of the above B-spline curve corresponds to the second difference of the sequence of the control vertex and the mid point of adjacent control vertices. Using Hoppe's rule at the boundary, the extended vertex is P'_4 (Fig. 5b), and so the second derivative becomes a vector from P'_4 to P_{11} for the extended vertex in Fig. 2.

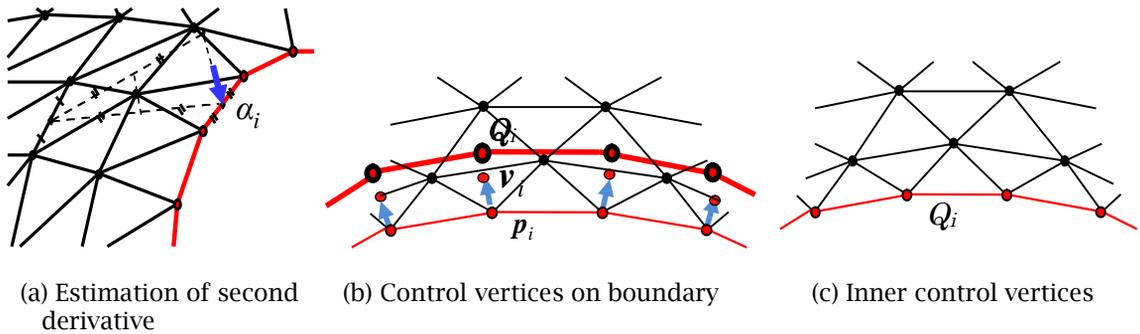


Fig. 6: Steps of determining control vertices.

3 BOUNDARY CONDITIONS AND FITTING OF SUBDIVISION SURFACE

3.1 Determination of the Control Mesh for a Subdivision Surface

Using the relationships described in Section 2, we next determine the control mesh for a Loop subdivision surface according to the boundary conditions at sharp features. The steps for this determination are summarized as:

1. Estimate the second derivative α_i at the mid-point for each edge of a sharp feature (Fig. 6a). Then determine the offset vector \mathbf{v}_i which moves an input vertex P_i , and which also corresponds to the effect of the extended vertex:

$$\mathbf{v}_i = -\frac{1}{12}(\alpha_{i-1} + \alpha_i). \quad (3.1)$$
2. Obtain control vertices Q_i for the features, using the relationship between a limit point and the control vertices, i.e. the control vertices are cubic B-spline control points which are fitted for points $P_i + \mathbf{v}_i$ (Fig. 6b).
3. Calculate inner control vertices by solving a linear equation which relates the limit point and control vertices (Fig. 6c).

In step 1, we estimate a second derivative α_i at the mid point of each edge. Then according to the relationship in Fig. 5b, the extended vertex E_i is allocated at the position translated a distance α_i from Hoppe's extended vertex (Fig. 7a). To determine Hoppe's control vertices Q_i on the boundary, we add the affect of α_i to an input point. The weighting function added to the boundary curve by the extended vertex is expressed in Eqn. (3.2) and shown in Fig. 7b, while the weighting function of Q_i is shown in Fig. 7c.

$$\begin{aligned} \phi_{10}(t) &= \frac{1}{12}(1 - 2t + 2t^3 - t^4), \\ \phi_{11}(t) &= \frac{1}{12}(1 + 2t - 4t^3 + 2t^4), \\ \phi_{12}(t) &= \frac{1}{12}(2t^3 - t^4). \end{aligned} \quad (3.2)$$

When a limit point on the boundary is calculated, the effect of the extended vertex is added via the second derivative. Thus, the input point is translated by vector \mathbf{v}_i according to Eqn. (3.2) as in step 2. In step 3, inner control vertices are calculated such that they satisfy the locations of Q_i on the boundaries and are efficiently solved by the iterative method [8].

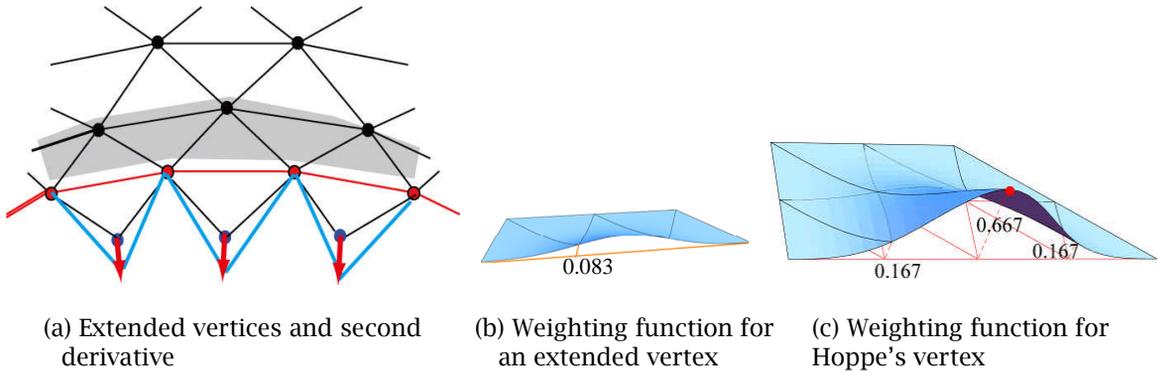


Fig. 7: Weighting functions for boundary and extended vertices.

3.2 Subdivision Rule at Features

We represented a subdivision surface using extended vertices in Section 3.1, but for an input curve, we cannot obtain the boundary curve exactly. Both side surfaces at a feature cannot share a common crease curve. Hence, we have to introduce a new subdivision rule at a feature such as a crease or boundary.

Step 2 in Section 3.1 ensures that the limit points on the boundary correspond with input points. However, as the subdivision is executed, some errors between limit points and points on the input curve appear. Hence, we introduce a subdivision rule which assures the *limit curve* corresponds with the specified input curve. At every subdivision of the feature, we calculate control vertices so that the limit points lie on the input curve. Let v_i^j be an offset vector for the j -th subdivision of a feature, then it is determined using weighting functions and second derivatives at the extended vertices:

$$v_i^j = \left(\frac{1}{2}\right)^j (\alpha_{i-1}\phi_{10}(t) + \alpha_i\phi_{11}(t) + \alpha_{i+1}\phi_{12}(t)), \tag{3.3}$$

where, ϕ_{10} , ϕ_{11} and ϕ_{12} are weighting functions for the crease curve obtained from the weighting function for the extended vertex in Fig. 7b and expressed in Eqn. (3.2).

Q_i^j are found for each subdivision as cubic B-spline control points which are fitted for points $P_i^j + v_i^j$. Here, P_i^j is a point corresponding to parameter t for the subdivision.

Fig. 8 shows the process of subdivision at a feature. As the subdivision advances, the number of control vertices increases, and an offset point $P_i^j + v_i^j$ is calculated from input point P_i^j corresponding to j -th subdivision. From these points, we calculate control vertices Q_i^j as shown in blue on Fig. 8.

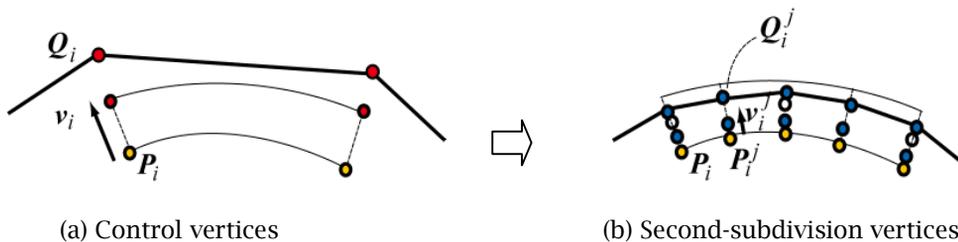


Fig. 8: Subdivision rule at a feature.

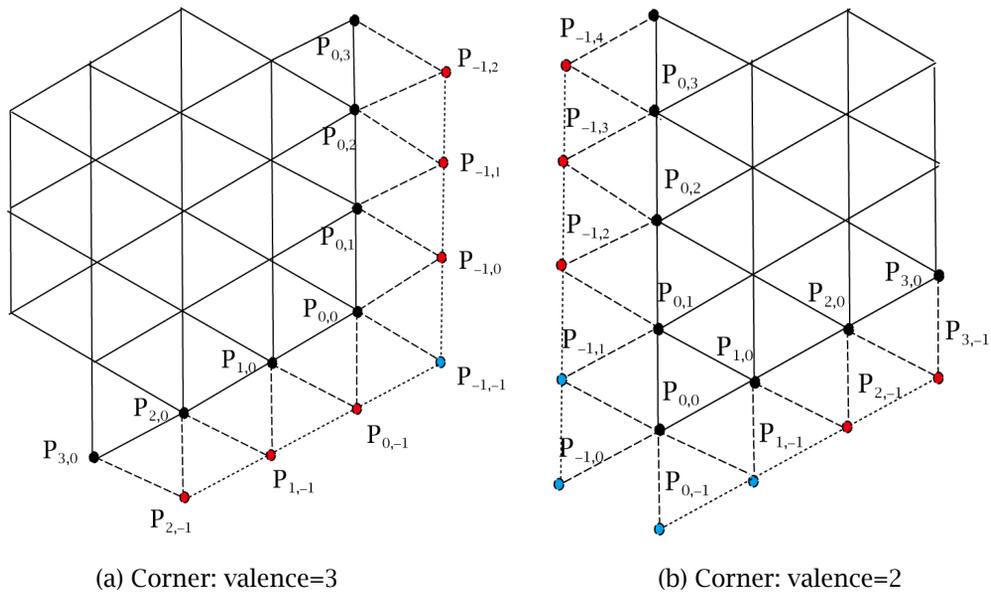


Fig. 9: Control mesh at corner.

3.3 Control Vertices at Corners

Finally, we discuss the control mesh for a corner where multiple features enter. In Fig. 9, for input mesh $P_{i,j}$, two sharp features A and B , $P_{i,0}$ $i=0,m$ and $P_{0,j}$ $j=0,n$, are incident to a corner, $P_{0,0}$. Along these features, we estimate the second derivatives $\alpha_{i,0}$ and $\alpha_{0,j}$, as described in Sec. 3.1. For the corner point where the valence is 3, as shown in Fig. 9a, we estimate the second derivative α_0 from the sequence $P_{0,0}$, $P_{1,1}$ and $P_{2,2}$. For a corner point where the valence is 2, as shown Fig. 9b, four second derivatives, corresponding to the blue extended vertices, are calculated along parametric lines in the same way as $P_{-1,-1}$. Using these derivatives and Eqn. (3.1), we can calculate offset vectors at each input vertex; however at the corner we get two vectors for the two boundaries:

$$\begin{aligned}
 \mathbf{v}_0^A &= -\frac{1}{12}(\alpha_{1,0} + \alpha_0), \\
 \mathbf{v}_0^B &= -\frac{1}{12}(\alpha_{0,1} + \alpha_0).
 \end{aligned}
 \tag{3.4}$$

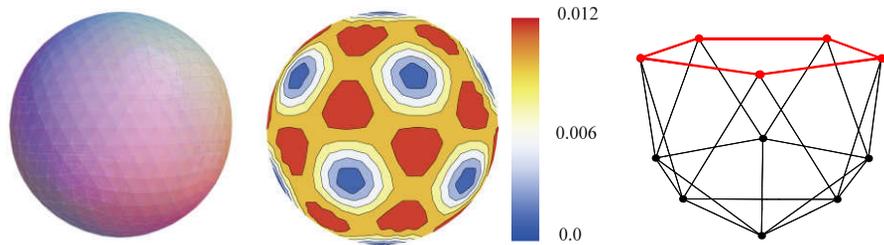
Calculating control vertices $Q_{i,0}$ and $Q_{0,j}$ on the features as in Sec. 3.1, we obtain two vertex values for $Q_{0,0}$ from the two features. As a result, the corner control vertex is taken as the average:

$$Q_{0,0} = \frac{1}{2}(Q_{0,0}^A + Q_{0,0}^B)
 \tag{3.5}$$

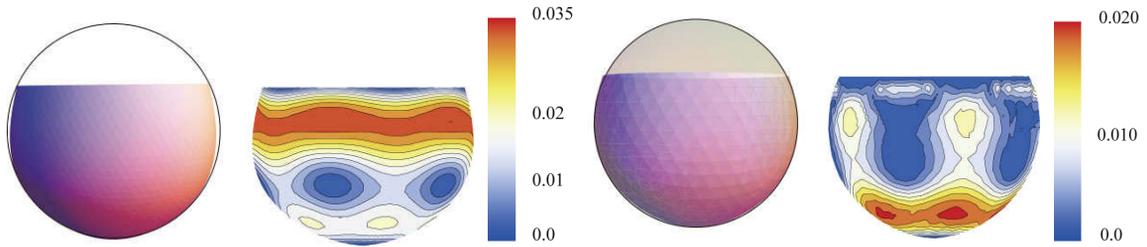
Using these control vertices as boundary conditions, we thus obtain the inner control vertices.

4 EXAMPLES

We now show two examples to which we apply our method. One is a 15-face polyhedron and the other is a free form surface with four boundaries. A 15-face polyhedron is made cutting off 5 faces from an icosahedron whose vertices are on the sphere of radius 1. Using the 11 input vertices shown in Fig. 10a, we determine control vertices for a subdivision surface. We evaluate the quality of the obtained surface



(a) Icosahedron and input vertices for 15-face polyhedron



(b) Hoppe's rule

(c) Our method

Fig. 10: Example of 15-face polyhedron.

by finding the difference between it and the unit sphere. The sphere represented by the icosahedron vertices includes some errors, as shown in the central figure of Fig. 10a. Using Hoppe's rule, the surface becomes smaller near the boundary than the unit sphere as shown in Fig. 10b where the circle on the left of the figure shows this difference clearly. On the other hand, as shown in Fig. 10c, the surface generated by our method has a much smaller error near the boundary and errors similar to the icosahedron around the bottom region. This is because we use the second derivatives as boundary conditions and the boundary curve is specified as a circle, meaning that our method interpolates the surface globally. The average and maximum errors, along with standard deviation, for each result are shown in Tab. 2.

Lastly, we applied our method to a free-form surface that is like a Bezier patch. Fig. 11 shows the obtained subdivision surfaces using Hoppe's rule and from our method. The surface quality is displayed using curvature profiles, which examine the curvature distribution by showing $1/10$ curvature radii along the curve. The curvature profile for the parametric lines of the subdivision surface generated by Hoppe's rule has a large radius near the boundary, while that of our method has a gradual and smooth curvature distribution. This demonstrates that Hoppe's subdivision rule generates a flat surface near the boundary, and that our method generates a more natural surface by specifying second derivatives.

	Maximum error	Average error	Standard deviation
Icosahedron	0.0113	0.0076	0.00311
Hoppe's rule	0.0312	0.0154	0.00849
Our method	0.0187	0.0057	0.00677

Tab. 2: Maximum and average errors of subdivision surface.

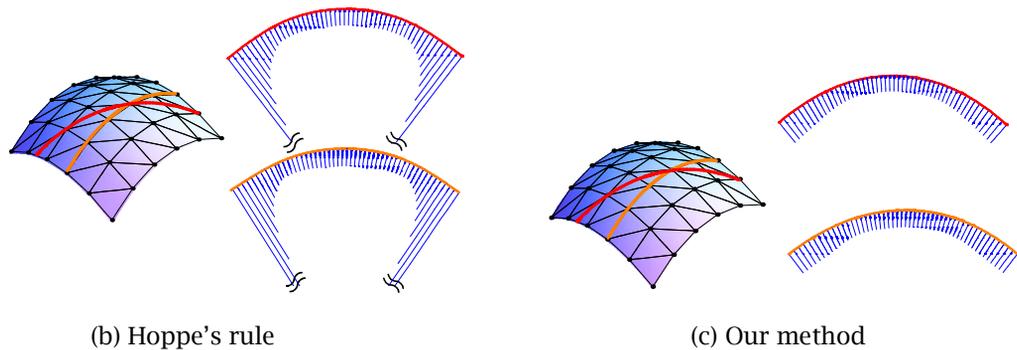


Fig. 11: Example for free form surface.

5 SUMMARY

We have proposed a method which obtains the control vertices for a high-quality subdivision surface using boundary conditions and introduced a new subdivision rule at the boundary by examining the cross boundary derivatives of triangular subdivision surfaces. The method estimates the second cross boundary derivatives, determines control vertices along the boundaries and then obtains inner control vertices. We have shown that the subdivision surface interpolated globally by our method satisfies boundary conditions and has smooth curvature distribution.

Our future research is to apply our method to approximation of subdivision surfaces.

This work was partly supported by KAKENHI (20560139).

REFERENCES

- [1] Biermann, H., Levin, A., Zorin, D.: Piecewise Smooth Subdivision Surfaces with Normal Control, Proc. SIGGRAPH2000, 2000, 113-120.
- [2] Boem, W.: The De Boor Algorithm for Triangular splines, in: Surfaces in Computer Aided Geometric Design, North-Holland Publishing Company, 1983, 109-120.
- [3] Boem, W.: Generating the Bezier Points of Triangular Spline, in: Surfaces in Computer Aided Geometric Design, North-Holland Publishing Company, 1983, 77-91.
- [4] Botsch, M., Pauly, M., Kobbelt, L., Alliez, P., Lévy, B., Bischoff, S., Rössl, C.: Geometric Modeling Based on Polygonal Meshes, SIGGRAPH07 Course a1, 2007.
- [5] DeRose, T., Kass, M., Truong, T.: Subdivision Surfaces in Character Animation, Proc. SIGGRAPH98, 1998, 85-94.
- [6] Farin, G.: Smooth Interpolation to Scattered 3D Data, in: Surfaces in Computer Aided Geometric Design, North-Holland Publishing Company, 1983, 43-63.
- [7] Higashi, M., Inoue, H., Oya, T.: High Quality Sharp Features in Triangular Meshes, Computer-Aided Design & Applications, 4 (1-4), 2007, 227-234.
- [8] Higashi, M., Mori, D., Seo, S., Kobayashi, M.: Efficient Fitting of Subdivision Surfaces by Simplified Newton Method, Computer-Aided Design & Applications, 7 (6), 2010, 821-834.
- [9] Hoppe, H., DeRose, T., Duchamp, T., Halstead, M.: Piecewise Smooth Surface Reconstruction, Proc. SIGGRAPH94, 1994, 295-302.
- [10] Kobbelt, L., Bischoff, S., Botsch, M., Kahler, K., Rossel, C., Schneider, R., Vorsatz, J.: Geometric Modeling Based on Polygonal Meshes, EUROGRAPHICS 2000 Tutorial.
- [11] Levin, A.: Interpolating Nets of Curves By Smooth Subdivision Surfaces, Proc. SIGGRAPH99, 1999, 57-64.
- [12] Loop, C.: Smooth subdivision surfaces based on triangles. Master's thesis, Department of Mathematics, University of Utah 1987.
- [13] Oya, T., Seo, S., Higashi, M.: Generating Sharp Features on Non-regular Triangular Meshes, Proc. ICCS 2008, LNCS 5102, 2008, 66-75, Springer-Verlag.
- [14] Zorin, D.: Modeling Multiresolution Subdivision Surface, SIGGRAPH06 Course, 2006.