PERGAMON

Computers & Graphics 26 (2002) 709–719

COMPUTERS & GRAPHICS

www.elsevier.com/locate/cag

Parametric-surface adaptive tessellation based on degree reduction

Seok-Hyung Bae, Hayong Shin, Won-Hyung Jung, Byoung K. Choi*

Department of Industrial Engineering, KAIST, 373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea

Abstract

Parametric-surface tessellation is one of the most important algorithms for CAGD applications. This paper presents a new parametric-surface tessellation method based on degree reduction: (1) a given parametric surface (or NURBS surface) of degrees (p, q) is decomposed into a set of Bezier surfaces, (2) the Bezier surfaces are converted into a set of bilinear surfaces by applying consecutive stepwise degree reduction processes combined with adaptive subdivision—in each degree reduction step, a Bezier surface is adaptively subdivided until the approximation error from degree reduction is smaller than the corresponding step tolerance, (3) the bilinear surfaces are converted into a triangular net. The proposed method guarantees the resulting piecewise-planar approximant to deviate from the original parametric surface within a pre-defined tolerance, and to form a "topologically" water-tight triangular net.

Keywords: Parametric-surface tessellation; Stepwise degree reduction; Adaptive subdivision; Piecewise-planar approximant; Water-tight triangular net

1. Introduction

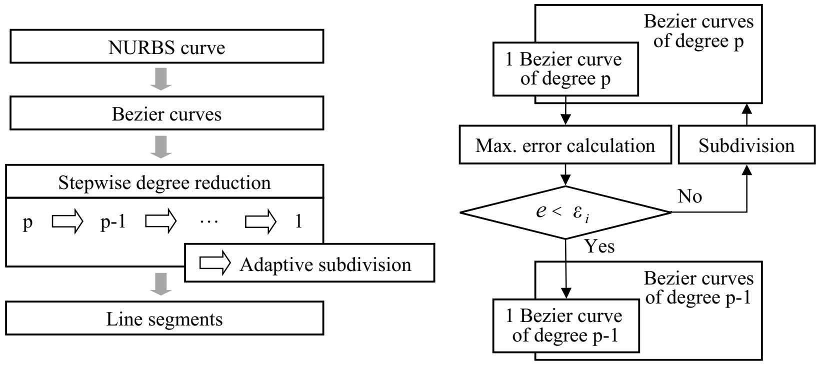
Parametric-surface tessellation, one of the most basic algorithms for CAGD applications such as CG, CAD/CAM, CAE, etc., is the process in which a given parametric surface is approximated as a piecewise-planar approximant within a pre-defined tolerance.

There have been many studies about parametric-surface tessellation, which can be roughly categorized into two as follows: (1) uniform sampling and (2) adaptive sampling. In uniform sampling methods [1–3], the parameter domain of the surface is "uniformly" divided with the grid size (i.e., parametric increment) computed based on 2nd derivative-bound estimation of the surface [1,3–5] so that the resulting approximant deviates from the original surface within a given tolerance. Uniform sampling methods are fast, however, produce too many triangles more than needed because the estimated 2nd derivative bounds are dependent of the most highly curved region (i.e., over-estimated). Adaptive sampling methods produce a piecewise-planar approximant from a parametric surface by repetitively applying "error check" and "subdivision". If the maximum deviation (or approximation error) of each planar segment from the corresponding parametric surface segment exceeds pre-defined tolerance, the parametric surface segment is subdivided into half or quarter. According to approximation-error calculation method, there can be three categories of adaptive sampling as follows: (1) 2nd derivative-bound (2DB) method, (2) control-point derivation (CP-deviation) method, and (3) point-on-surface derivation (POS-deviation) method. The approximation error calculated by using 2DB methods [5,6] is "inevitably" over-estimated, and it leads the resulting approximant to have "unnecessarily" excess triangles. CP-deviation methods [7–9] estimate the maximum deviation by calculating the maximum distance between the control net of the parametric surface (Bezier or B-spline surface) and the corresponding planar segment. CP-deviation methods guarantee the resulting approximant deviates from a given parametric surface within a pre-defined tolerance in virtue of the convex-hull

*Corresponding author. Tel.: +82-42-869-3115; fax: +82-42-869-3110.

E-mail address: bkchoi@vmslab.kaist.ac.kr (B.K. Choi).

PII: S0097-8493(02)00126-7



(a) (b)

Fig. 1. Basic idea of parametric-curve polygonization: (a) overall procedure and (b) adaptive subdivision.

property of Bezier and B-spline surfaces. However, CP-deviation methods also have the same problem—the resulting approximant has excess triangles because of the over-estimation of the maximum deviation. POS-deviation method [10] is to sample points from a parametric surface segment and to calculate distances between the sampled points and the corresponding planar segment, and finally to take the maximum value among them. Unfortunately, there have been little theoretical backgrounds related to sampling density for the resulting approximant to meet a pre-defined tolerance.

To be presented in the paper is a new parametric-surface tessellation method, which is extended from our recent study [11]—the parametric-curve polygonization method based on degree reduction. The proposed method consists of following steps: (1) a given parametric surface is decomposed into a set of Bezier surfaces, (2) each Bezier surface is converted into a set of bilinear surfaces by applying consecutive stepwise degree reduction processes combined with adaptive subdivision, (3) all the bilinear surfaces are converted into a triangular net. The proposed method guarantees the resulting piecewise-planar approximant to deviate from the original parametric surface within a pre-defined tolerance, and to form a "topologically" water-tight triangular net.

The organization of the paper is as follows: summarized in Section 2 is the parametric-curve polygonization method based on degree reduction, which is extended to a surface tessellation in Section 3. Illustrative examples are given in Section 4 followed by discussions and conclusions in Section 5.

2. Parametric-curve polygonization

The overall procedure of the parametric-curve polygonization method proposed by Bae et al. [11] is shown in Fig. 1a. A given parametric curve (commonly, NURBS curve) of degree p is decomposed into a set of Bezier curves. The Bezier curves are converted into a set of line segments by applying successive $p-1$ stepwise degree reduction steps. A given pre-defined tolerance (ε) is distributed to each degree reduction step, and in each step, the degrees of parametric-curve segments are reduced within the corresponding step tolerance (ε_i) combined with adaptive subdivision (see Fig. 1b). Consequently, the resulting piecewise-linear approximant (or a set of line segments, i.e., polygon), of which the approximation error does not exceed the pre-defined tolerance, is obtained.

Now, let us consider the maximum error calculation for adaptive subdivision in each degree reduction step. The stepwise degree reduction of a Bezier curve of degree p

$$\mathbf{C}(t) = \sum_{i=0}^{p} B_{i,p}(t)\mathbf{P}_i \quad (1)$$

is finding the control points $\{\bar{\mathbf{P}}_i\}$ of a Bezier curve of degree $p-1$

$$\bar{\mathbf{C}}(t) = \sum_{i=0}^{p-1} B_{i,p-1}(t)\bar{\mathbf{P}}_i. \quad (2)$$

There are two well-known "recursive" extrapolation formulas as follows [12]:

$$\bar{\mathbf{P}}_0^{\mathrm{I}} = \mathbf{P}_0, \quad \bar{\mathbf{P}}_i^{\mathrm{I}} = \frac{\mathbf{P}_i - \alpha_i \bar{\mathbf{P}}_{i-1}^{\mathrm{I}}}{1-\alpha_i}, \quad i = 1, \ldots, p-1, \quad (3a)$$

$$\bar{\mathbf{P}}_{p-1}^{\mathrm{II}} = \mathbf{P}_p,$$
$$\bar{\mathbf{P}}_i^{\mathrm{II}} = \frac{\mathbf{P}_{i+1} - (1-\alpha_{i+1})\bar{\mathbf{P}}_{i+1}^{\mathrm{II}}}{\alpha_{i+1}}, \quad i = p-2, \ldots, 0, \quad (3b)$$

where $\alpha_i = i/p$. Bae et al., chose the midpoint-blending type stepwise degree reduction algorithm [12] for the

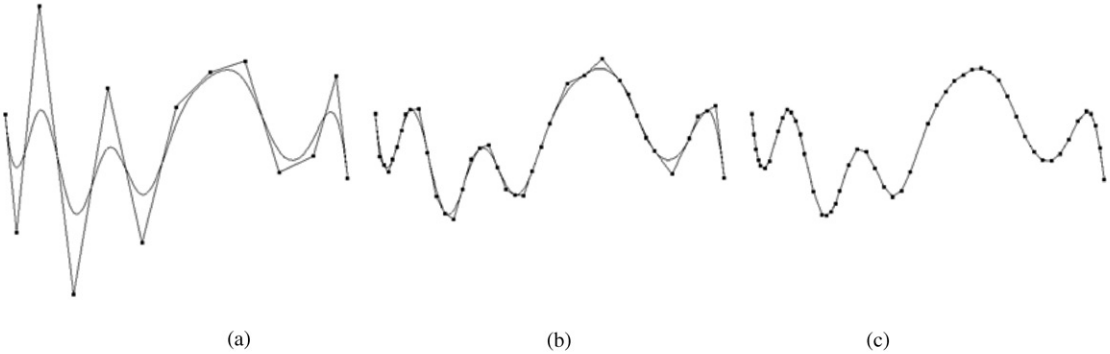


Fig. 2. Parametric-curve polygonization method based on stepwise degree reduction: (a) given cubic B-spline curve, (b) quadratic Bezier curves, and (c) resulting piecewise-linear approximant.

unique solution as given by

$$\bar{\mathbf{P}}_i = \begin{cases} \left.\begin{cases} \bar{\mathbf{P}}_i^{\mathrm{I}} & \text{for } i = 0, \ldots, r \\ \bar{\mathbf{P}}_i^{\mathrm{II}} & \text{for } i = r+1, \ldots, p-1 \end{cases}\right\} & \text{if } p \text{ is even,} \\ \left.\begin{cases} \bar{\mathbf{P}}_i^{\mathrm{I}} & \text{for } i = 0, \ldots, r-1 \\ \frac{1}{2}(\bar{\mathbf{P}}_i^{\mathrm{I}} + \bar{\mathbf{P}}_i^{\mathrm{II}}) & \text{for } i = r \\ \bar{\mathbf{P}}_i^{\mathrm{II}} & \text{for } i = r+1, \ldots, p-1 \end{cases}\right\} & \text{if } p \text{ is odd,} \end{cases} \quad (4)$$

where $r = \lfloor (p-1)/2 \rfloor$. The midpoint-blending method is much easier to implement than the algorithms based on best approximation [13–16] or non-linear optimization [17,18]. Moreover, it provides a simple explicit (parametric) approximation error function as follows [19]:

$$\begin{aligned} e(t) &= \| \mathbf{C}(t) - \bar{\mathbf{C}}(t) \| \\ &= \begin{cases} B_{r+1,p}(t) \left| \mathbf{P}_{r+1} - \frac{1}{2}(\bar{\mathbf{P}}_r^{\mathrm{I}} + \bar{\mathbf{P}}_{r+1}^{\mathrm{II}}) \right| & \text{if } p \text{ is even,} \\ \frac{1}{2}(1-\alpha_r)|B_{r,p}(t) - B_{r+1,p}(t)| \, \| \bar{\mathbf{P}}_r^{\mathrm{I}} - \bar{\mathbf{P}}_r^{\mathrm{II}} \| & \text{if } p \text{ is odd.} \end{cases} \end{aligned} \quad (5)$$

The maximum values of the error function can be precisely calculated: the even-case error function has one maximum at $t = 0.5$, and odd-case error function has two "equal" maximums at parameters satisfying $t^2 - t + (p-1)/4p = 0$.

Fig. 2 shows the step-by-step results of the parametric-curve polygonization method proposed by Bae et al. for a cubic B-spline curve.

3. Extension to surface tessellation

The basic idea of the parametric-curve polygonization method given in the previous section is simply extended to a surface case, but some additional treatments are necessary. First, the maximum approximation error caused by stepwise degree reduction for a parametric surface has to be calculated. Second, the parametric-surface segment of degrees (1,1) obtained from stepwise degree reduction processes, is not a "planar" segment, but a "bilinear" surface. Hence, another approximation step must be included. On the other hand, for some applications, it is necessary to make a "water-tight" triangular net from the bilinear surface set.

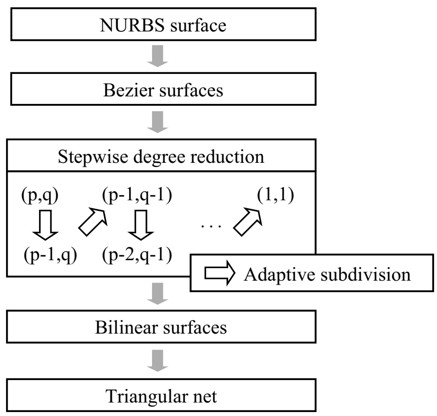


Fig. 3. Overall procedure of parametric-surface tessellation.

3.1. Overall procedure

So far no "explicit" approximation error function when the degrees of a parametric surface are reduced from (p, q) to $(p-1, q-1)$ has been known. Thus, we use "one-directional" stepwise degree reduction. For example, the u-directional degree of a parametric surface is first reduced, and v-directional degree next, and u-directional degree in turn, etc. Fig. 3 shows the overall

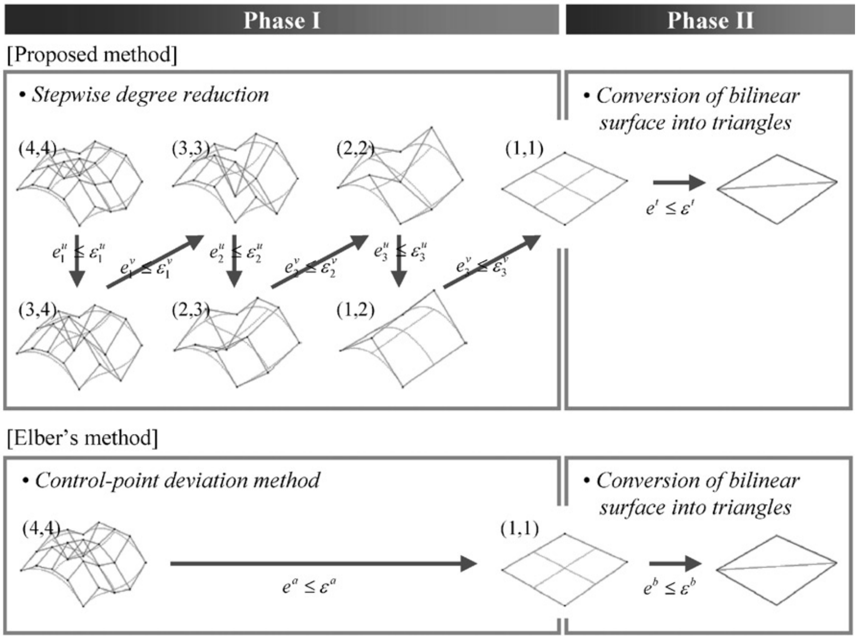


Fig. 4. Two-phase approximation approach.

procedure of the proposed parametric-surface tessellation method.

As mentioned earlier, because a bilinear surface is not planar, the two-phase approximation approach, which is composed of the approximation of a Bezier surface to a bilinear surface and that of a bilinear surface to triangles, is used. Fig. 4 shows the two-phase approximation of the proposed method compared with that of Elber's method [9] (for the sake of explanatory simplicity, we assumed that the both sides of degrees are the same, that is, $p = q$). While Elber's method uses the CP-deviation method to estimate the maximum deviation of each bilinear surface segment from a given parametric surface, the proposed method uses the accurate approximation error function of each u- and v-directional stepwise degree reduction. In Elber's, if the maximum deviation estimated from two phases is greater than pre-defined tolerance ($e^a + e^b > \varepsilon$), the parametric surface segment is split. Whereas, in our proposed method, pre-defined tolerance is distributed into each degree reduction step so that $\sum_{i=1}^{p-1}(\varepsilon_i^u + \varepsilon_i^v) + \varepsilon^t = \varepsilon$, and adaptive subdivision is performed in each step with its corresponding step tolerance.

3.2. Stepwise degree reduction for surface case

Based on the "notational" symmetry of parameters of a Bezier surface (more strictly, tensor-product surface), we will consider only one parameter, u.

The u-directional stepwise degree reduction of a Bezier surface of degrees (p, q)

$$\mathbf{S}(u,v) = \sum_{i=0}^{p}\sum_{j=0}^{q} B_{i,p}(u)B_{j,q}(v)\mathbf{P}_{i,j} \tag{6}$$

is finding the control points $\{\bar{\mathbf{P}}_{i,j}\}$ of a Bezier surface of degrees $(p-1, q)$

$$\bar{\mathbf{S}}(u,v) = \sum_{i=0}^{p-1}\sum_{j=0}^{q} B_{i,p-1}(u)B_{j,q}(v)\bar{\mathbf{P}}_{i,j}. \tag{7}$$

By applying the midpoint-blending method to each row of the control net of the Bezier surface, the new control points are given as

$$\bar{\mathbf{P}}_{i,j} = \begin{cases} \left.\begin{cases} \bar{\mathbf{P}}^{\mathrm{I}}_{i,j} & \text{for } i = 0,\ldots,r \\ \bar{\mathbf{P}}^{\mathrm{II}}_{i,j} & \text{for } i = r+1,\ldots,p-1 \end{cases}\right\} \text{if } p \text{ is even}, \\ \left.\begin{cases} \bar{\mathbf{P}}^{\mathrm{I}}_{i,j} & \text{for } i = 0,\ldots,r-1 \\ \frac{1}{2}(\bar{\mathbf{P}}^{\mathrm{I}}_{i,j} + \bar{\mathbf{P}}^{\mathrm{II}}_{i,j}) & \text{for } i = r \\ \bar{\mathbf{P}}^{\mathrm{II}}_{i,j} & \text{for } i = r+1,\ldots,p-1 \end{cases}\right\} \text{if } p \text{ is odd}, \end{cases} \tag{8}$$

where $r = \lfloor (p-1)/2 \rfloor$ and

$$\bar{\mathbf{P}}^{\mathrm{I}}_{0,j} = \mathbf{P}_{0,j}, \quad \bar{\mathbf{P}}^{\mathrm{I}}_{i,j} = \frac{\mathbf{P}_{i,j} - \alpha_i \bar{\mathbf{P}}^{\mathrm{I}}_{i-1,j}}{1-\alpha_i}, \quad i = 1,\ldots,p-1, \tag{9a}$$

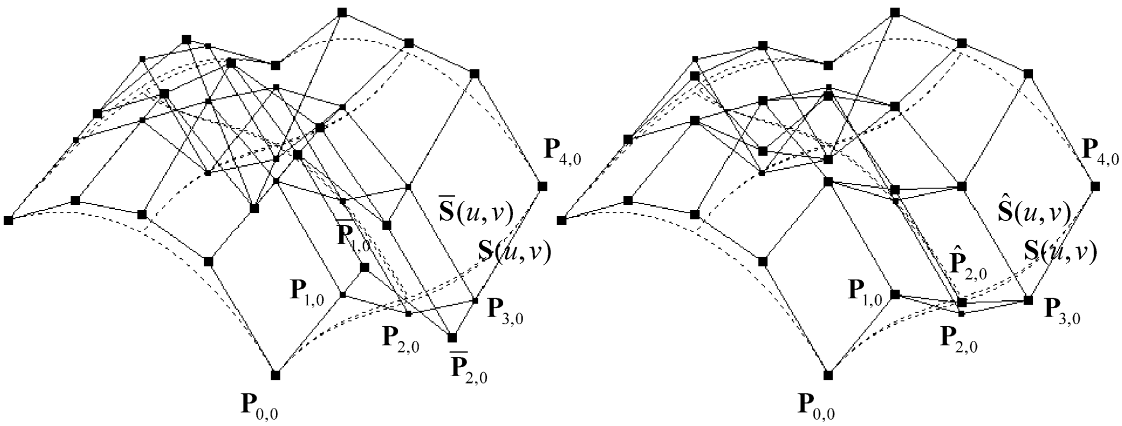


Fig. 5. u-directional stepwise degree reduction of Bezier surface of degrees (4, 4).

$$\bar{\mathbf{P}}^{\mathrm{II}}_{p-1,j} = \mathbf{P}_{p,j},$$
$$\bar{\mathbf{P}}^{\mathrm{II}}_{i,j} = \frac{\mathbf{P}_{i+1,j} - (1-\alpha_{i+1})\bar{\mathbf{P}}^{\mathrm{II}}_{i+1,j}}{\alpha_{i+1}},\ i = p-2, \ldots, 0, \tag{9b}$$

where $\alpha_i = i/p$.

As similar to the curve case, the most important advantage of using the midpoint-blending method is that an explicit (parametric) approximation error function having a simple form is precisely given. Let $\hat{\mathbf{S}}(u,v)$ is the degree-elevated surface of degrees (p,q) from $\bar{\mathbf{S}}(u,v)$ as

$$\hat{\mathbf{S}}(u,v) = \sum_{i=0}^{p}\sum_{j=0}^{q} B_{i,p}(u)B_{j,q}(v)\hat{\mathbf{P}}_{i,j}, \tag{10}$$

where $\{\hat{\mathbf{P}}_{i,j}\}$ are given by well-known degree elevation formula as

$$\hat{\mathbf{P}}_{i,j} = \alpha_i\bar{\mathbf{P}}_{i-1,j} + (1-\alpha_i)\bar{\mathbf{P}}_{i,j}, \quad i = 0, \ldots, p. \tag{11}$$

If p is even, the error function is derived as follows:

$$\begin{aligned}
e(u,v) &= \|\mathbf{S}(u,v) - \bar{\mathbf{S}}(u,v)\| = \|\mathbf{S}(u,v) - \hat{\mathbf{S}}(u,v)\| \\
&= \left| \sum_{i=0}^{p}\sum_{j=0}^{q} B_{i,p}(u)B_{j,q}(v)\mathbf{P}_{i,j} - \sum_{i=0}^{p}\sum_{j=0}^{q} B_{i,p}(u)B_{j,q}(v)\hat{\mathbf{P}}_{i,j} \right| \\
&= \left| \sum_{j=0}^{q} B_{j,q}(v)\left[\sum_{i=0}^{p} B_{i,p}(u)(\mathbf{P}_{i,j} - \hat{\mathbf{P}}_{i,j})\right] \right| \\
&= \left| \sum_{j=0}^{q} B_{j,q}(v)\left[B_{r+1,p}(u)(\mathbf{P}_{r+1,j} - \hat{\mathbf{P}}_{r+1,j})\right] \right| \\
&= B_{r+1,p}(u)\left| \sum_{j=0}^{q} B_{j,q}(v)(\mathbf{P}_{r+1,j} - \hat{\mathbf{P}}_{r+1,j}) \right| \\
&= B_{r+1,p}(u)\left| \sum_{j=0}^{q} B_{j,q}(v)\left[\mathbf{P}_{r+1,j} - \frac{1}{2}(\bar{\mathbf{P}}^{\mathrm{I}}_{r,j} + \bar{\mathbf{P}}^{\mathrm{II}}_{r+1,j})\right] \right|.
\end{aligned} \tag{12}$$

To be remarkable in Eq. (12) is the error function is "parameter-separable"—letting $\tilde{\mathbf{P}}_j \equiv \mathbf{P}_{r+1,j} - (\bar{\mathbf{P}}^{\mathrm{I}}_{r,j} + \bar{\mathbf{P}}^{\mathrm{II}}_{r+1,j})/2$, the error function has the form of a scaled 3D-curve norm, which is the product of a scale function $s(u)$ and 3D-curve norm $\|\tilde{\mathbf{C}}(v)\|$, as follows:

$$e(u,v) = B_{r+1,p}(u)\left| \sum_{j=0}^{q} B_{j,q}(v)\tilde{\mathbf{P}}_j \right| \equiv s(u)\cdot\|\tilde{\mathbf{C}}(v)\|. \tag{13}$$

Obviously, the error function has the maximum value at (u_M, v_M) where $s(u)$ and $\|\tilde{\mathbf{C}}(v)\|$ have maximum values independently as follows:

$$\begin{aligned}
e(u_M, v_M) &\equiv \max_{u,v} e(u,v) = \max_{u,v} s(u)\cdot\|\tilde{\mathbf{C}}(v)\| \\
&= \max_{u} s(u)\cdot \max_{v}\|\tilde{\mathbf{C}}(v)\| \equiv s(u_M)\cdot\|\tilde{\mathbf{C}}(v_M)\|.
\end{aligned} \tag{14}$$

Fig. 5 shows an example of u-directional degree reduction when p is 4. It is noticeable what to affect the approximation error are only the differences of the control points on the $(r+1)$th row of each control net of $\mathbf{S}(u,v)$ and $\hat{\mathbf{S}}(u,v)$.

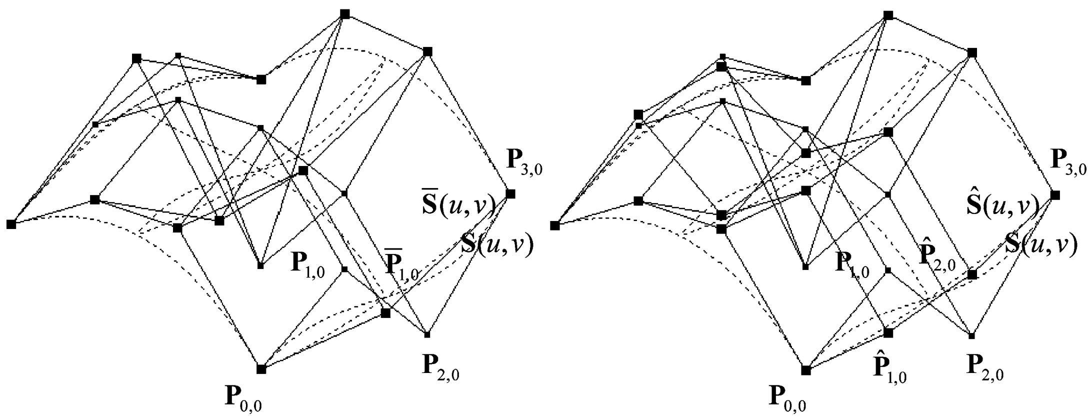


Fig. 6. u-directional stepwise degree reduction of Bezier surface of degrees (3, 3).

If p is odd, the similar result is obtained as follows:

$$
\begin{aligned}
e(u,v) &= \left|\sum_{j=0}^{q} B_{j,q}(v)[B_{r,p}(u)(\mathbf{P}_{r,j}-\hat{\mathbf{P}}_{r,j}) \right.\\
&\quad \left. + B_{r+1,p}(u)(\mathbf{P}_{r+1,j}-\hat{\mathbf{P}}_{r+1,j})]\right|\\
&= \left|\sum_{j=0}^{q} B_{j,q}(v)\left[\frac{1}{2}(1-\alpha_r)(B_{r,p}(u) \right.\right.\\
&\quad \left.\left. -B_{r+1,p}(u))(\bar{\mathbf{P}}^{\mathrm{I}}_{r,j}-\bar{\mathbf{P}}^{\mathrm{II}}_{r,j})\right]\right|\\
&= \frac{1}{2}(1-\alpha_r)|B_{r,p}(u)-B_{r+1,p}(u)|\\
&\quad \left|\sum_{j=0}^{q} B_{j,q}(v)[(\bar{\mathbf{P}}^{\mathrm{I}}_{r,j}-\bar{\mathbf{P}}^{\mathrm{II}}_{r,j})]\right|\\
&\equiv \frac{1}{2}(1-\alpha_r)|B_{r,p}(u)-B_{r+1,p}(u)|\left|\sum_{j=0}^{q} B_{j,q}(v)\tilde{\mathbf{P}}_j\right|\\
&\equiv s(u)\cdot\|\tilde{\mathbf{C}}(v)\|. \qquad (15)
\end{aligned}
$$

The above error function also has a parameter-separable form: $s(u)\equiv(1-\alpha_r)|B_{r,p}(u)-B_{r+1,p}|/2$ and $\|\tilde{\mathbf{C}}(v)\|\equiv\|\sum_{j=0}^{q}B_{j,q}(v)\tilde{\mathbf{P}}_j\|$ where $\tilde{\mathbf{P}}_j\equiv\bar{\mathbf{P}}^{\mathrm{I}}_{r,j}-\bar{\mathbf{P}}^{\mathrm{II}}_{r,j}$. In the case, the error function is affected by the differences of control points on "two" rows of each control net of $\mathbf{S}(u,v)$ and $\hat{\mathbf{S}}(u,v)$ as shown in Fig. 6.

3.3. Maximum deviation calculation

As noted earlier, the maximum value of the approximation error function can be calculated by finding maximum values of the scale function and 3D-curve norm, independently. The scale function's maximum value is already known [19]; the even-case scale function $B_{r+1,p}(u)$ has the maximum at $u=0.5$, and odd-case scale function $(1-\alpha_r)|B_{r,p}(u)-B_{r+1,p}(u)|/2$ has two maximums at the parameters satisfying $u^2-u+(p-1)/4p=0$. Now, let us consider the calculation of the maximum 3D-curve norm $\|\tilde{\mathbf{C}}(v_M)\|$. Following two approaches are applicable to adaptive subdivision: (1) exact-value finding, and (2) practical-bound estimation. To find the "exact" maximum value of $\|\tilde{\mathbf{C}}(v)\|$ can be interpreted as the general point-projection problem, which is searching for the parameter of $\tilde{\mathbf{C}}(v)$ such that the distance between the origin $\mathbf{O}(0,0,0)$ and $\tilde{\mathbf{C}}(v)$ is maximized, or as finding maximum value of a single-valued function (or non-parametric Bezier curve) of degree $2q$, $f(v)\equiv\|\tilde{\mathbf{C}}(v)\|^2=\tilde{\mathbf{C}}(v)\cdot\tilde{\mathbf{C}}(v)$ which can be obtained by applying the symbolic dot-product operation [20]. In general, these approaches require a kind of numerical methods such as the Newton–Raphson iteration [21], except q is extremely low. Thus, it is important to take good "seed" parameters for reliable convergence. Whereas, in a practical viewpoint, the bounds of $\|\tilde{\mathbf{C}}(v_M)\|$ based on the convex-hull property of (refined) control polygon [1,3] can be useful—they give somewhat over-estimated values, but are fast and stable.

3.4. Subdivision strategy

In each degree reduction step, when the degree of a Bezier surface does not be reduced along one parameter direction within the corresponding step tolerance, the Bezier surface is split in half along the same direction. And the degree reduction is applied to the split surfaces again. In this way, we can obtain the degree-reduced Bezier surfaces meet the step tolerance (we call this subdivision strategy binary subdivision).

3.5. Tolerance distribution

How to distribute the pre-defined tolerance to subsequent stepwise degree reduction processes is very important to the performance of the proposed method. Bae et al. [11] suggested the exponential step-tolerance method—as the degree reduction step increases, the step

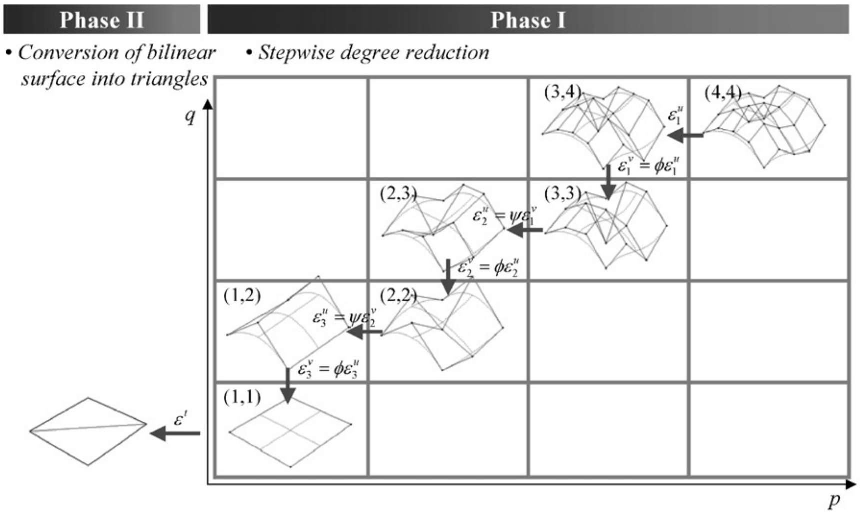


Fig. 7. Schematic diagram of exponential step-tolerance method.

tolerance increases with a given ratio. For a surface case, the relations between the step tolerances are given as $\varepsilon_i^v = \phi\varepsilon_i^u$, $\varepsilon_{i+1}^u = \psi\varepsilon_i^v$, $\varepsilon_{i+1}^u = (\phi\psi)\varepsilon_i^u$, $\varepsilon_{i+1}^v = (\phi\psi)\varepsilon_i^v$; $\phi, \psi \geqslant 1$. Fig. 7 shows the schematic diagram of the exponential step-tolerance method. By some arithmetical manipulations, the explicit formulas for the each step tolerance are derived as follows (see Appendix A for details):

$$\varepsilon_i^u = \frac{(\varepsilon_{rest,i}^u - \varepsilon_t)(\phi\psi - 1)}{(1+\phi)\{(\phi\psi)^{p-i} - 1\}}, \tag{16a}$$

$$\varepsilon_i^v = \frac{(\varepsilon_{rest,i}^v - \varepsilon_t)(\phi\psi - 1)}{\psi(1+\phi)\{(\phi\psi)^{q-i-1} - 1\} + \phi\psi - 1}, \tag{16b}$$

where $\varepsilon_{rest,i}^u$ and $\varepsilon_{rest,i}^v$ are available tolerances for subsequent degree reduction steps, which are calculated based on the cumulative approximation error of each intermediate Bezier surface as $\varepsilon_{rest,i}^u = \varepsilon - \sum_{j=1}^{i-1}(e_j^u + e_j^v)$ and $\varepsilon_{rest,i}^v = \varepsilon - \{\sum_{j=1}^{i-1}(e_j^u + e_j^v) + e_i^u\}$. Fig. 8 shows the results of the first-phase approximation of the proposed method with different ψ values (with $\phi = 1$); the abscissa is ψ, which varies from 1 to 32, and the ordinate is the ratio of the number of bilinear surfaces produced by the proposed method with the corresponding ψ, to the number of bilinear surfaces produced by Elber's method. The test surface is an automatic-gearshift knob surface (see Fig. 13a), which is a bicubic B-spline surface (60 mm × 30 mm × 100 mm). We can observe that the exponential step-tolerance method ($\psi > 1$) gives better results than the uniform step-tolerance method [15,16,22] ($\psi = 1$) in view of both "economics" and "accuracy". In the other words, the accurate approximations of the early steps pay back—the resulting triangular-net has less triangles, and is more close to the original parametric surface. On the other hand, we can also find that the proposed method generates less bilinear surfaces than Elber's method if a proper ψ value is chosen, and works comparatively better as the pre-defined tolerance is smaller.

3.6. Triangular-net construction

When converting bilinear surfaces into a set of triangles, the special attention should be paid to make a water-tight triangular net with no internal cracks for a wide range of applications. Peterson [8] suggested a simple crack prevention method as shown in Fig. 9, but which does not provide a "topologically" valid water-tight triangular net.

For the case, we present a new crack prevention method. If a bilinear surface adjacent to one along a boundary edge like patch *A* in Fig. 10a, it is diagonally split into two triangles. And if not (see patch *B* and *C* in Fig. 10a), split into four triangles with the center point of the patch evaluated at (0.5, 0.5) (see Fig. 10b), then triangles are refined as shown in Fig. 10c.

When a bilinear surface $\mathbf{B}(u, v)$ is converted into two or four triangles, the maximum approximation errors [9] are given as follows (see Fig. 11).

$$e_{M,2} = \frac{1}{4}\max\{dist(\mathbf{P}_{1,1}, \Pi(\mathbf{P}_{0,0}, \mathbf{P}_{1,0}, \mathbf{P}_{0,1})), dist(\mathbf{P}_{0,0}, \Pi(\mathbf{P}_{1,1}, \mathbf{P}_{0,1}, \mathbf{P}_{1,0}))\}, \tag{17a}$$

$$e_{M,4} = \frac{1}{8}\max\left\{\begin{array}{l} dist(\mathbf{P}_{1,0}, \Pi(\mathbf{P}_{0,0}, \mathbf{P}_c, \mathbf{P}_{0,1})), dist(\mathbf{P}_{1,1}, \Pi(\mathbf{P}_{0,0}, \mathbf{P}_c, \mathbf{P}_{0,1})), \\ dist(\mathbf{P}_{1,1}, \Pi(\mathbf{P}_{1,0}, \mathbf{P}_c, \mathbf{P}_{0,0})), dist(\mathbf{P}_{0,1}, \Pi(\mathbf{P}_{1,0}, \mathbf{P}_c, \mathbf{P}_{0,0})), \\ dist(\mathbf{P}_{0,1}, \Pi(\mathbf{P}_{1,1}, \mathbf{P}_c, \mathbf{P}_{1,0})), dist(\mathbf{P}_{0,0}, \Pi(\mathbf{P}_{1,1}, \mathbf{P}_c, \mathbf{P}_{1,0})), \\ dist(\mathbf{P}_{0,0}, \Pi(\mathbf{P}_{0,1}, \mathbf{P}_c, \mathbf{P}_{1,1})), dist(\mathbf{P}_{1,0}, \Pi(\mathbf{P}_{0,1}, \mathbf{P}_c, \mathbf{P}_{1,1})) \end{array}\right\}, \tag{17b}$$

where Π is the plane on which three 3D points in its parenthesis are, and $\mathbf{P}_c$ is the center point on the bilinear surface sampled at (0.5, 0.5).

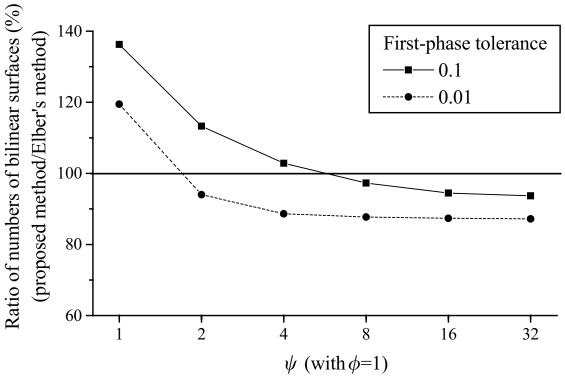


Fig. 8. First-phase approximation test.

For splitting into two triangles, there exist two sets of two triangles, and the set of triangles whose the maximum approximation error calculated with Eq. (17a) is smaller, is selected as a part of the resulting triangular net.

4. Illustrative examples

Fig. 12 shows an example of the proposed parametric-surface tessellation method. The parametric surface to be tessellated in Fig. 12a is a "nautilus-shell" shaped bicubic B-spline surface (310 mm × 270 mm × 80 mm), which is designed by using a "logarithmic spiral" whose the curvature (or radius of curvature) is monotonically changed. Figs. 12b and c are the resulting piecewise-planar approximants with the pre-defined tolerances 2.0 and 1.0 (mm), respectively. The computation time was 6.379 and

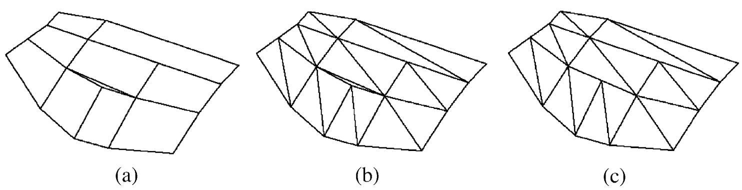


Fig. 9. Crack prevention method proposed by Peterson.

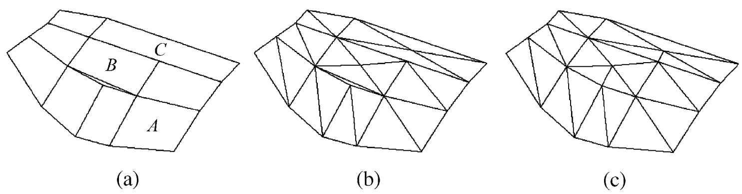


Fig. 10. Water-tight triangular net construction with proposed crack prevention method.

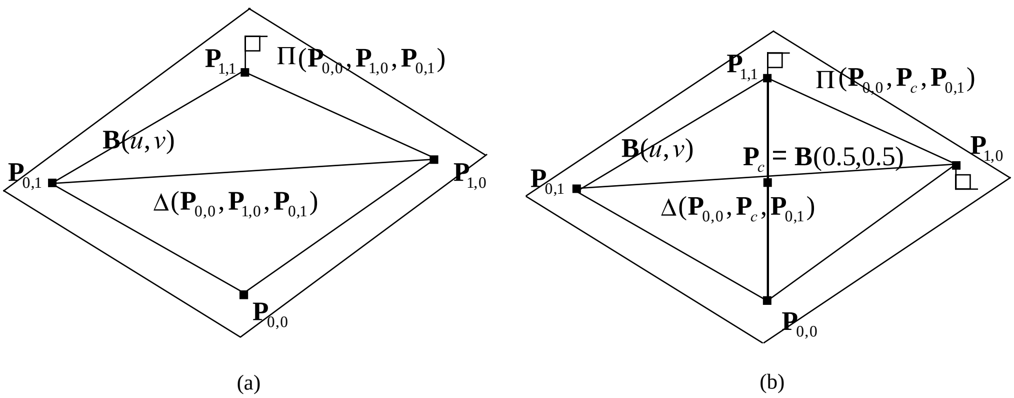


Fig. 11. Conversion of bilinear surface into triangles.

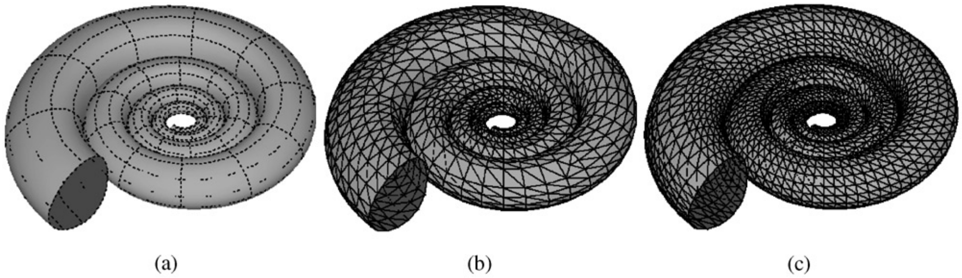


Fig. 12. Parametric-surface tessellation of nautilus-shell shaped surface: (a) bicubic B-spline surface, resulting piecewise-planar approximants with pre-defined tolerance, (b) 2.0, and (c) 1.0.

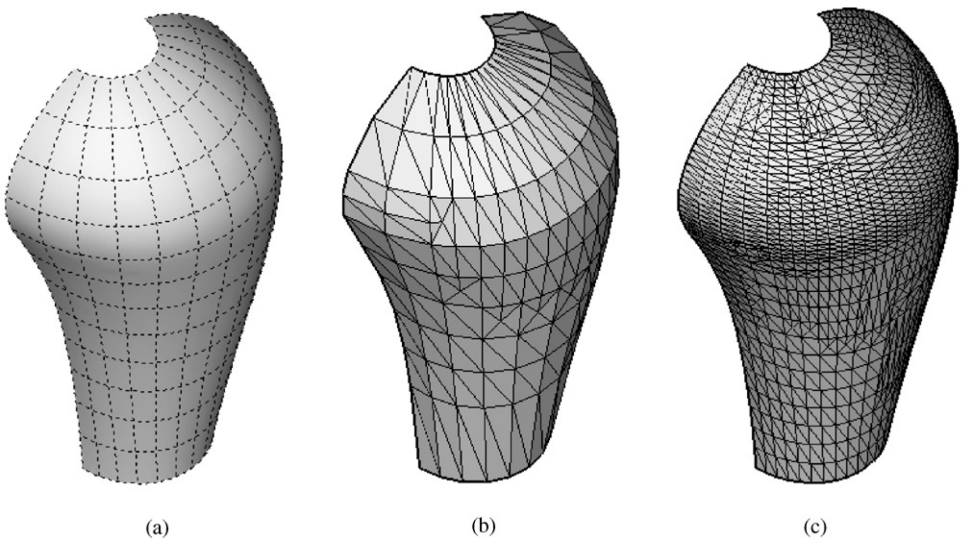


Fig. 13. Parametric-surface tessellation of gearshift-knob surface: (a) bicubic B-spline surface, resulting piecewise-planar approximants with pre-defined tolerance, (b) 1.0, and (c) 0.1.

10.675 (s), respectively (measured on a 800 MHz Intel® Pentium III processor).

Another example is given in Fig. 13: a given bicubic B-spline surface, which is an automatic-gearshift knob surface ($60\,\mathrm{mm} \times 30\,\mathrm{mm} \times 100\,\mathrm{mm}$), is converted into water-tight triangular net with the pre-defined tolerances 1.0 and 0.1 (mm), respectively (computation time was 0.580 and 2.554 (s), respectively).

5. Discussion and conclusions

The proposed parametric-surface tessellation method is extended from our previous work [11]—parametric-curve polygonization whose the basic idea is that a piecewise-linear approximant satisfying a pre-defined tolerance can be obtained by combining stepwise degree reduction with adaptive subdivision. To extend the curve-case result to a surface case, the one-directional stepwise degree reduction strategy is introduced. By applying the midpoint-blending type stepwise degree reduction to each parameter direction of the surface, the approximation error function is factorized into a function of u and a function of v, and the maximum error can be easily calculated because the problem of a surface (error function) becomes that of a curve. For converting each parametric surface of degrees (1, 1), so called bilinear surface, into planar segments (i.e., triangles), Elber's results [9] are adopted, and for constructing a water-tight triangular net, a simple crack prevention method is proposed.

One of the distinct characteristics of the proposed method is a kind of "multi-step" approximation. Namely, the resulting approximant is "recursively" approximated to a given parametric surface. Thus, the resulting triangular-net deviates from a given parametric surface within a pre-defined tolerance, but the vertices of it are not on the original surface (so, the term "approximant" was used instead of "interpolant" in the paper). Also, in the multi-step approximation, the pre-defined tolerance has to be distributed to each step, and it can cause the unwanted waste of the available tolerance. Thus, the proposed method uses the exact calculation of the maximum deviation of approximants and the exponential step-tolerance strategy with proper ratios (ϕ, ψ) to compensate the problem.

Further work can be how to determine the optimal values of the ratios (ϕ, ψ) for the exponential step-tolerance method, which is expected to relate to a size ratio of the parametric surface to be tessellated and the pre-defined tolerance.

Acknowledgements

The Ministry of Science and Technology of Korean government has supported the research.

Appendix A. Exponential Step-tolerance Calculation

The available tolerance for ith u-directional stepwise degree reduction is to be distributed into the step tolerances for subsequent $(p-i)$ degree reduction steps and tolerance for the conversion of a bilinear surface into triangles as follows:

$$\begin{aligned}\varepsilon^u_{rest,i} &= \sum_{j=i}^{p-1}(\varepsilon^u_j + \varepsilon^v_j) + \varepsilon^t = (1+\phi)\sum_{j=i}^{p-1}\varepsilon^u_j + \varepsilon^t \\ &= (1+\phi)\sum_{j=0}^{p-i-1}(\phi\psi)^j\varepsilon^u_i + \varepsilon^t \\ &= \varepsilon^u_i(1+\phi)\sum_{j=0}^{p-i-1}(\phi\psi)^j + \varepsilon^t \\ &= \varepsilon^u_i(1+\phi)\,\frac{(\phi\psi)^{p-i}-1}{\phi\psi-1} + \varepsilon^t. \qquad \text{(A.1)}\end{aligned}$$

Similarly,

$$\begin{aligned}\varepsilon^v_{rest,i} &= \varepsilon^v_i + \sum_{j=i+1}^{q-1}(\varepsilon^u_j + \varepsilon^v_j) + \varepsilon^t \\ &= \varepsilon^v_i + \psi(1+\phi)\sum_{j=i+1}^{q-1}\varepsilon^v_{j-1} + \varepsilon^t \\ &= \varepsilon^v_i + \psi(1+\phi)\sum_{k=i}^{q-2}\varepsilon^v_k + \varepsilon^t \\ &= \varepsilon^v_i + \psi(1+\phi)\sum_{k=0}^{q-i-2}(\phi\psi)^k\varepsilon^v_i + \varepsilon^t \\ &= \varepsilon^v_i + \varepsilon^v_i\psi(1+\phi)\sum_{k=0}^{q-i-2}(\phi\psi)^k + \varepsilon^t \\ &= \varepsilon^v_i + \varepsilon^v_i\psi(1+\phi)\,\frac{(\phi\psi)^{q-i-1}-1}{\phi\psi-1} + \varepsilon^t. \qquad \text{(A.2)}\end{aligned}$$

References

[1] Filip D, Magedson R, Markot R. Surface algorithms using bounds on derivatives. Computer Aided Geometric Design 1986;3:295–311.

[2] Sheng X, Hirsh EE. Triangulation of trimmed surfaces in parametric space. Computer Aided Design 1992;24(8): 437–44.

[3] Piegl L, Richard AM. Tessellating trimmed NURBS surfaces. Computer Aided Design 1995;27(1): 16–26.

[4] Lane JM, Riesenfeld RF. Bounds on a polynomial, BIT 1981;21:112–7.

[5] Dolenc A, Makela I. Optimized triangulation of parametric surfaces. In: Bowyer A, editor. Computer-aided surface geometry and design: the mathematics of surfaces IV. Oxford, UK: Clarendon Press, 1991. p. 1–13.

[6] Lane JM, Carpenter L. A generalized scan line algorithm for the computer display of parametrically defined surfaces. Computer Graphics and Image Processing 1979;11:290–7.

[7] Lane JM, Riesenfeld RF. A theoretical development for computer generation and display of piecewise polynomial surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 1980;2(1):35–46.

[8] Peterson JW. Tessellation of NURBS surfaces. In: Heckbert PS, editor. Graphics gems IV. New York: Academic Press, 1994. p. 286–320.

[9] Elber G. Error bounded piecewise linear approximation of freeform surfaces. Computer Aided Design 1996;28(1): 51–7.

[10] Austin SP, Jerard RB, Drysdale RL. Comparison of discretization algorithms for NURBS surface with application to numerically controlled machining. Computer Aided Design 1997;29(1):71–83.

[11] Bae SH, Shin HY, Choi BK. Parametric curve polygonization based on degree reduction, VMS Lab Technical Report, VMS-01-01. KAIST (in Korean), 2001.

[12] Forrest AR. Interactive interpolation and approximation by Bezier polynomials. The Computer Journal 1972;15(1):71–9.

[13] Lachance MA, Saff EB, Varga RS. Bounds for incomplete polynomials vanishing at both endpoints of an interval. In: Coffman C, Fix G, editors. Constructive approaches to mathematical models. New York: Academic Press, 1979. p. 421–37.

[14] Lachance MA. Chebyshev economization for parametric surfaces. Computer Aided Geometric Design 1988;5:195–208.

[15] Watkins MA, Worsey AJ. Degree reduction of Bezier curves. Computer Aided Design 1988;20(7):398–405.

[16] Eck E. Degree reduction of Bezier curves. Computer Aided Geometric Design 1993;10:237–51.

[17] Eisele EF. Chebyshev approximation of plane curves by splines. Journal of Approximation Theory 1994; 76:133–48.

[18] Degen WLF. Best approximations of parametric curves by splines. In: Lyche T, Schumaker L, editors. Mathematical methods in computer aided geometric design II. New York: Academic Press, 1992. p. 171–84.

[19] Piegl L, Tiller W. Algorithm for degree reduction of B-spline curves. Computer Aided Design 1995;27(2):101–10.

[20] Piegl L, Tiller W. Symbolic operators for NURBS. Computer Aided Design 1997;29(5):361–8.

[21] Piegl L, Tiller W. The NURBS book. New York: Springer, 1995.

[22] Eck M. Least squares degree reduction of Bezier curves. Computer Aided Design 1995;27(11):845–51.