

INTERSECTION TEST AND BLOSSOMING PERTURBATION FOR DISK PARAMETRIC CURVES AND BALL PARAMETRIC SURFACES

QUN LIN AND J. G. ROKNE

Received 13 December 2004; Accepted 5 October 2005

Errors in curve and surface representation due to inaccuracies in the data are considered and accounted for by introducing disk parametric curves and ball parametric surfaces. Intersection test algorithms and interval extensions using blossoming are discussed for each of the three cases of Bézier curves, tensor product surfaces, and triangular patches. A stability analysis is also performed for each of the three cases. It is shown that under certain restrictions disk Bézier curves and triangular ball Bézier patches are stable with respect to perturbations of the control disks (balls); whereas tensor product ball Bézier surfaces are in general not.

Copyright © 2006 Q. Lin and J. G. Rokne. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

The curves and surfaces employed in classical geometric modeling represent the shapes of objects. This representation is an abstraction where it is assumed that the curves have no width and that the surfaces have zero thickness. This idealization does not carry over to the real world. There are restrictions on how precisely a curve can be rendered, there are restrictions on possible tool movements when a physical implementation is machined and there are restrictions on material forming to mention a few of the sources of real world problems in implementing abstract designs.

As an example consider the effects of drift and restriction of movement of a tool when the tool is used to machine an object whose boundary is defined by a mathematical curve or surface. As the tool operates, a small change in local position may result in a global deviation from the intended object. As a result the topology of the object might change, the object cannot be integrated used for its intended purpose and the object might not have the intended mechanical properties. A more extensive discussion of problems with machining accuracy is found in [18] and especially in the references therein.

2 Intersection test and blossoming perturbation

A second example is the problem of how to handle the modeling of an object by means of curves and surfaces given that imprecise information about the object has been provided. If the curve is defined by control points, then how should these be adjusted to take the imprecise information into account in the best possible manner and at the same time to take into account the offset problem occurring in practical computations and operations?

In both of the above examples, curves of nonzero width, and surfaces having nonzero thickness can be employed. They can also be used to fit double boundary lines and double boundary faces. Further applications of such curves and surfaces can be found in [7, 13, 19, 20].

In this paper previous work by the authors on fitting of parametric curves [10], fitting fat curves [9], and disk Bézier curves [8] is augmented by further results on disk curves, tensor product ball surfaces, and triangular ball patches. For each case, bisection algorithms suitable for intersection tests have been included (Sections 3, 6, 9). Complexity analyses and implementation details are not included for the algorithms. Interval extensions are, however, defined (Sections 4, 7, 10) as stability analyses are (Sections 5, 8, 11).

2. Disk and ball arithmetics

Interval analysis is a tool for inclusion and estimation of errors and uncertainties in numerical calculations [1]. In one dimension, intervals are closed bounded subsets of the real line \mathbb{R} and the set of such intervals is denoted by $I(\mathbb{R})$. An arithmetic can be defined on $I(\mathbb{R})$ and certain algebraic properties are observed (see, e.g., [16]) and an analysis can be developed; see Moore [11]. The extension of one-dimensional intervals to higher dimensions in the space $I^m(\mathbb{R})$ is usually done componentwise in order to take advantage of the properties in one dimension as far as possible. An m -dimensional interval in $I^m(\mathbb{R})$ (also called a box) is therefore described by $2m$ real parameters (the componentwise lower and upper bounds).

An alternate way of extending intervals to higher dimensions is given by disks in \mathbb{R}^2 and balls in \mathbb{R}^m ($m > 2$). Denoting the set of nonnegative reals by \mathbb{R}^+ , a ball(disk) in \mathbb{R}^m is defined to be the set

$$B := \{x \mid |x - r| \leq \rho, r \in \mathbb{R}^m, \rho \in \mathbb{R}^+\} \quad (2.1)$$

described by $m + 1$ real parameters. The set of balls(disks) in \mathbb{R}^m is denoted by $D^m(\mathbb{R})$. If $B \in D^m(\mathbb{R})$, then we also write

$$B = \langle r, \rho \rangle, \quad (2.2)$$

where $r = \text{mid}B$ is the center of the ball(disk), and $\rho = \text{rad}B$, the radius of the ball (disk).

If $B_1 = \langle r_1, \rho_1 \rangle, B_2 = \langle r_2, \rho_2 \rangle \in D^m(\mathbb{R})$ $r_1, r_2 \in \mathbb{R}^m, \rho_1, \rho_2 \in \mathbb{R}$ are two disks (balls), then

$$\begin{aligned} B_1 + B_2 &= \langle r_1 + r_2, \rho_1 + \rho_2 \rangle, \\ B_1 \cdot B_2 &= \langle r_1 r_2, \|r_1\| \rho_2 + \|r_2\| \rho_1 + \rho_1 \rho_2 \rangle. \end{aligned} \quad (2.3)$$

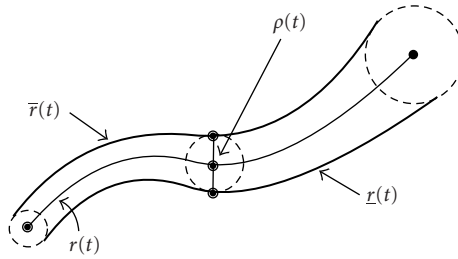


Figure 3.1. A fat curve.

If additionally $T = [t - \delta, t + \delta] \subseteq I(\mathbb{R})$ is a one-dimensional disk, then

$$TB_1 = \langle tr_1, |t|\rho_1 + \|r_1\|\delta + \rho_1\delta \rangle. \tag{2.4}$$

Real ball(disk) arithmetic was discussed in [6] under the name ‘‘Hypernormballe.’’ Further details of this arithmetic are found in [8].

3. The intersection of disk curves

In many applications it is important to know if two curves intersect or not. As an example consider a piece of cloth which is intended for a garment. The garment outlines are defined by curves on an abstraction of the piece of cloth. If the outlines do not intersect, the cutting process is deemed feasible and when the curves are placed on the cloth it is assumed that the garment pieces can be cut out of the cloth. In reality, there are several sources of error such as tool movement, workpiece movement, and numerical evaluation errors that may invalidate the cutting process; even though the abstract design was feasible. In order to guard against such problems, the intersection test for the curves is normally made with a tolerance.

Two curves $p_1(t)$ and $p_2(t)$ are therefore considered, and it is assumed that both curves are defined over $t \in [0, 1]$ without loss of generality. In order to guarantee that the curves do not intersect, we require that the minimum distance between the curves satisfies

$$\min_{\forall s, t \in [0, 1]} \|p_1(s) - p_2(t)\| \geq 2\delta^*, \tag{3.1}$$

where δ^* is a tolerance parameter.

This can then be rephrased in terms of disk parametric curves. A disk parametric curve consists of a parametric axis curve $r(t)$ and an associated variable radius $\rho(t)$, and it is written as $B(t) = \langle r(t), \rho(t) \rangle$. As t varies, the disk $\langle r(t), \rho(t) \rangle$ sweeps out a domain whose inner and outer envelopes are $\underline{r}(t)$ and $\bar{r}(t)$ as shown in Figure 3.1. The interval extension of $B(t)$ to a parametric interval T is written as $B(T)$ where it is assumed that the extension is computed as, for example, a centered form (see [11, 17]).

The curve intersection problem considered above can then be stated in terms of parametric disk curves $B_1(t) = \langle p_1(t), \delta^* \rangle$, $B_2(t) = \langle p_2(t), \delta^* \rangle$, $t \in [0, 1]$, and requiring that

4 Intersection test and blossoming perturbation

these disk curves do not intersect. If the disk curves do not intersect, then we are guaranteed that the axis curves do not intersect up to the assumption on the possible errors of representation.

We cast the intersection problem in a slightly more general setting where the two disk curves are allowed to have width as a function of the parameter t . Let therefore $B_i(t) = \langle r_i(t), \rho_i(t) \rangle$, $i = 1, 2$, $t \in [0, 1]$, where $r_i(t) \in B_i(t)$ is the axis curve and $\rho_i(t)$ is the variable perturbation of the axis curve due to an error, $i = 1, 2$. We will test to see whether or not

$$B_1(s) \cap B_2(t) = \emptyset, \quad s, t \in [0, 1], \quad (3.2)$$

holds. If the property does not hold, then we cannot guarantee that the nonintersection of the axis curves will remain valid when the error perturbations are taken into account and hence the topology of the curves might be changed by the perturbations.

From the above, we can see that our problem can be viewed as an intersection test between two disk curves. This is shown in Figure 3.2. In the following, we will use circular arithmetic to construct a test algorithm for the intersection of disk curves.

Note that

$$\begin{aligned} I &= [0, 1], \\ B_1(s) &= \langle r_1(s), \rho_1(s) \rangle, \quad s \in I, \\ S &= [\underline{s}, \bar{s}] \subseteq I, \quad w(S) = \bar{s} - \underline{s}, \\ B_2(t) &= \langle r_2(t), \rho_2(t) \rangle, \quad t \in I, \\ T &= [\underline{t}, \bar{t}] \subseteq I, \quad w(T) = \bar{t} - \underline{t}. \end{aligned} \quad (3.3)$$

Furthermore for any subintervals $S \subseteq \mathcal{S}$, $T \subseteq \mathcal{T}$, it is assumed that a computational procedure exists for finding inclusions for

$$B_1(S) = \{B_1(s), s \in S\}, \quad B_2(T) = \{B_2(t), t \in T\}. \quad (3.4)$$

The proposed algorithm is based on an exhaustion method, that is, bisection together with interval extension evaluations are used to progressively delete subintervals of $[0, 1]$ which correspond to disk segments guaranteed not to intersect. The algorithm is included here for completeness even though the principle of the algorithm is well known.

Because of finite machine precision, we have to prescribe an admissible error bound EPS which is used to terminate the algorithm when the width of subintervals is less than EPS.

We use lists \mathcal{S} and \mathcal{T} to store the subintervals of $[0, 1]$ for $B_1(s)$ and $B_2(t)$ that must be processed further.

After each bisection, we put the uncertain subintervals that must be tested further at the end of the list. This means that the widest subinterval remaining is always the first on each list.

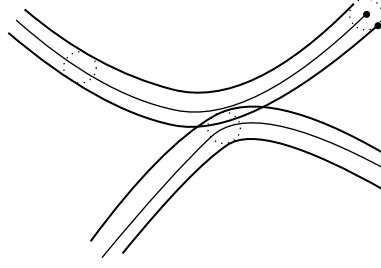


Figure 3.2. Intersection computation.

For convenience, we let

$$\mathcal{B}_1(\mathcal{S}) = \{B_1(S), \forall S \in \mathcal{S}\}, \quad \mathcal{B}_2(\mathcal{T}) = \{B_2(T), \forall T \in \mathcal{T}\} \quad (3.5)$$

and for any $B_1 \in \mathcal{B}_1(\mathcal{S})$, $B_2 \in \mathcal{B}_2(\mathcal{T})$, we define

$$\begin{aligned} B_1 \cap \mathcal{B}_2(\mathcal{T}) &= \{B_1 \cap B, \forall B \in \mathcal{B}_2(\mathcal{T})\}, \\ \mathcal{B}_1(\mathcal{S}) \cap B_2 &= \{B \cap B_2, \forall B \in \mathcal{B}_1(\mathcal{S})\}. \end{aligned} \quad (3.6)$$

Algorithm 3.1. (1) Initialize lists $\mathcal{S} = \{[0, 1]\}$, $\mathcal{T} = \{[0, 1]\}$.

- (2) Set $S =$ first item on \mathcal{S} , $T =$ first item on \mathcal{T} .
- (3) If $w(S) < \text{EPS}$ then go to (7).
- (4) Delete S from \mathcal{S} .
- (5) Bisect S so that $S = S_1 \cup S_2$.
- (6) For $i = 1, 2$ do
 - (i) Calculate $B_1(S_i)$.
 - (ii) If $B_1(S_i) \cap \mathcal{B}_2(\mathcal{T}) \neq \emptyset$ then enter S_i onto the list \mathcal{S} as the last item.
- (7) If $w(T) < \text{EPS}$ then go to (11).
- (8) Delete T from \mathcal{T} .
- (9) Bisect T so that $T = T_1 \cup T_2$.
- (10) For $j = 1, 2$ do
 - (i) Calculate $B_2(T_j)$.
 - (ii) If $\mathcal{B}_1(\mathcal{S}) \cap B_2(T_j) \neq \emptyset$ then enter T_j onto the list \mathcal{T} as the last item.
- (11) If $w(S)$ or $w(T) > \text{EPS}$ then go to (2).
- (12) End.

If the algorithm terminates with either \mathcal{S} or \mathcal{T} being empty, then the two disk curves $B_1(s)$, $B_2(t)$ do not intersect. Conversely if the algorithm terminates with both \mathcal{S} and \mathcal{T} being nonempty, then $B_1(s) \cap B_2(t) \neq \emptyset$ holds to the precision EPS, that is, the two disk curves $B_1(s)$, $B_2(t)$ intersect within that precision. In this case it might be necessary to rerun the algorithm with smaller EPS in order to possibly get a more precise result.

4. The interval extension of a disk Bézier curve

In this section the effect of errors in curve representation is considered for a Bézier curve.

Let $\{b_i\}$, $i = 1, 2, \dots, n$ be a series of control points. A planar Bézier curve defined on these control points can be written as

$$b(t) = \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j b_i, \quad t \in [0, 1], \quad (4.1)$$

where

$$\binom{n}{i \ j} = \frac{n!}{i!j!}; \quad (4.2)$$

see, for example, [3].

There are two sources of errors in the computation of a Bézier curve. These are perturbations of the control points b_i and perturbations of the parameter t .

In order to consider the perturbations of control points b_i , we use control disks $B_i = \langle b_i, \rho_i \rangle$ instead of the original b_i 's obtaining a disk Bézier curve:

$$\begin{aligned} B(t) &= \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j B_i \\ &= \left\langle \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j b_i, \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j \rho_i \right\rangle \\ &= \left\langle b(t), \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j \rho_i \right\rangle, \quad t \in [0, 1]. \end{aligned} \quad (4.3)$$

This means that a disk Bézier curve can be computed from two scalar Bézier curves forming the center and radius of the disk curve, respectively.

In order to consider perturbations of the parameter t , we have to consider the interval extension of a disk curve, and we use a one-dimensional disk parameter $T = [t - \delta, t + \delta]$ instead of the original t , where $\delta > 0$ is an error radius.

The polar form of the curve [2] is an efficient approach for computing the value of a Bézier curve. This can be extended to the computation of a disk curve via the polar form of the disk curve. If F is the polar form of $b(t)$ and if t is written as $t = (1-t) \cdot 0 + t \cdot 1$, we have

$$\begin{aligned} F(t, \dots, t) &= (1-t) \cdot F(t, \dots, t, 0) + t \cdot F(t, \dots, t, 1) \\ &= \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j F \left(\underbrace{0, \dots, 0}_j, \underbrace{1, \dots, 1}_i \right) = b(t), \end{aligned} \quad (4.4)$$

where $b_i = F(\underbrace{0, \dots, 0}_j, \underbrace{1, \dots, 1}_i)$.

Consider now the extension $F(T, \dots, T)$ of the polar form F to an interval $T \subseteq [0, 1]$. This is a type of nested form which is an improvement over the extension of the Bézier form which is defined by

$$B(T) = \sum_{i+j=n} \binom{n}{i \ j} T^i (1-T)^j B_i. \quad (4.5)$$

Due to the subdistributivity of interval or disk arithmetic, we have

$$F(T, \dots, T) \subseteq \sum_{i+j=n} \binom{n}{i \ j} T^i (1-T)^j B_i = B(T) \quad (4.6)$$

and it is also of lower computational complexity when evaluated in a nested fashion.

The de Casteljau recursive process described in, for example, [3, page 45] can be used to compute $F(T, \dots, T)$:

$$\begin{aligned} &\text{Input } B_i^0 = B_i, \ i = 0, 1, \dots, n, T, \\ &\quad \text{For } r = 1, \dots, n, \\ &\quad \quad \text{For } i = 0, \dots, n-r, \\ &\quad \quad \quad B_i^r = (1-T)B_i^{r-1} + TB_{i+1}^{r-1}, \\ &\quad \text{Output } B_0^n, \\ &\quad \text{Return.} \end{aligned} \quad (4.7)$$

In the above process, T is an interval and B_i^r is a disk; hence the result B_0^n is a disk, that is,

$$B_0^n = F(T, \dots, T). \quad (4.8)$$

The polar form extension $F(T, \dots, T)$ is an inclusion for the disk Bézier curve which in turn includes the Bézier curve perturbed with errors. Letting

$$\tilde{T}_{ij} = \binom{n}{i \ j} T^i (1-T)^j, \quad T \subseteq [0, 1], \quad (4.9)$$

we have

$$\begin{aligned} B(T) &= \sum_{i+j=n} \tilde{T}_{ij} \cdot \langle b_i, \rho_i \rangle \\ &= \sum_{i+j=n} \langle \text{mid } \tilde{T}_{ij} \cdot b_i, \mid \text{mid } \tilde{T}_{ij} \mid \rho_i + \|\tilde{T}_{ij}\| \cdot \text{rad } \tilde{T}_{ij} + \rho_i \cdot \text{rad } \tilde{T}_{ij} \rangle \\ &= \left\langle \sum_{i+j=n} \text{mid } \tilde{T}_{ij} \cdot b_i, \sum_{i+j=n} \mid \text{mid } \tilde{T}_{ij} \mid \rho_i + \sum_{i+j=n} \|\tilde{T}_{ij}\| \cdot \text{rad } \tilde{T}_{ij} + \sum_{i+j=n} \rho_i \cdot \text{rad } \tilde{T}_{ij} \right\rangle. \end{aligned} \quad (4.10)$$

8 Intersection test and blossoming perturbation

We can also view T as a disk so that it can be expressed as $\langle t, \delta \rangle$; thus

$$\begin{aligned}
 \tilde{T}_{ij} &= \binom{n}{i \ j} T^i (1-T)^j \\
 &= \binom{n}{i \ j} \cdot \langle t^i, (t+\delta)^i - t^i \rangle \cdot \langle (1-t)^j, (1-t+\delta)^j - (1-t)^j \rangle \\
 &= \binom{n}{i \ j} \cdot \langle t^i (1-t)^j, t^i [(1-t+\delta)^j - (1-t)^j] \\
 &\quad + (1-t)^j [(t+\delta)^i - t^i] + [(t+\delta)^i - t^i] [(1-t+\delta)^j - (1-t)^j] \rangle \\
 &= \left\langle \binom{n}{i \ j} t^i (1-t)^j, \binom{n}{i \ j} [(t+\delta)^i (1-t+\delta)^j - t^i (1-t)^j] \right\rangle
 \end{aligned} \tag{4.11}$$

using (2.4) and where $0 \leq t + \delta, 1 - t + \delta \leq 1$. Therefore

$$\begin{aligned}
 F(T, \dots, T) \subseteq B(T) &= \left\langle \sum_{i+j=n} \binom{n}{i \ j} t^i (1-t)^j b_i, \right. \\
 &\quad \sum_{i+j=n} \binom{n}{i \ j} [(t+\delta)^i (1-t+\delta)^j - t^i (1-t)^j] \\
 &\quad \left. \times \|b_i\| + \sum_{i+j=n} \binom{n}{i \ j} (t+\delta)^i (1-t+\delta)^j \rho_i \right\rangle.
 \end{aligned} \tag{4.12}$$

When the perturbations ρ_i is the control points and the perturbation δ of the evaluation point tends to zero, we would expect that the value of the Bézier curve be obtained. This is indeed the case as can be seen from (4.12) and we have the following theorem for the extension of the polar form.

THEOREM 4.1.

$$\text{rad}(F(T, \dots, T) - b(t)) \longrightarrow 0 \quad (\text{as } \rho, w(T) \longrightarrow 0), \tag{4.13}$$

where $\rho = \max_{0 \leq i \leq n} \{\rho_i\}$, $t = \text{mid } T$.

This means that $F(T, \dots, T)$ converges to $b(t)$.

5. Disk analysis for error generated from blossoming

Blossoming is a technique popularized in [2, 14, 15] for studying polynomials in computer-aided geometric design by replacing them by simple multilinear functions.

A blossoming algorithm is now used to estimate the error propagation in computation of a Bézier curve in order to analyze its computational stability. The stability of polynomials expressed in the Bernstein form, which forms the basis for Bézier curves, was shown to be computationally stable in [4, 5].

A general disk curve can be generated by means of the following formula:

$$B_i^r = \alpha_i^{r-1} B_i^{r-1} + \beta_i^{r-1} B_{i+1}^{r-1}, \quad r = 1, \dots, n, \quad i = 0, \dots, n-r, \quad (5.1)$$

where $B_i^0 = B_i = \langle b_i, \rho_i \rangle$, $i = 0, \dots, n$, $\alpha_i^{r-1}, \beta_i^{r-1} \in [0, 1]$. The result is the disk blossoming B_0^n .

Let

$$\Lambda_i^r = \langle \alpha_i^r, \epsilon_i^r \rangle, \quad \Delta_i^r = \langle \beta_i^r, \delta_i^r \rangle, \quad B_i^r = \langle b_i^r, \rho_i^r \rangle \quad (5.2)$$

be the disk perturbations of α_i^r, β_i^r . We get

$$\begin{aligned} \langle b_i^r, \rho_i^r \rangle &= \Lambda_i^{r-1} \cdot \langle b_i^{r-1}, \rho_i^{r-1} \rangle + \Delta_i^{r-1} \cdot \langle b_{i+1}^{r-1}, \rho_{i+1}^{r-1} \rangle \\ &= \langle \alpha_i^{r-1} b_i^{r-1}, \alpha_i^{r-1} \rho_i^{r-1} + \|b_i^{r-1}\| \epsilon_i^{r-1} + \epsilon_i^{r-1} \rho_i^{r-1} \rangle \\ &\quad + \langle \beta_i^{r-1} b_{i+1}^{r-1}, \beta_i^{r-1} \rho_{i+1}^{r-1} + \|b_{i+1}^{r-1}\| \delta_i^{r-1} + \delta_i^{r-1} \rho_{i+1}^{r-1} \rangle. \end{aligned} \quad (5.3)$$

Considering only the error radii, we have

$$\rho_i^r = \alpha_i^{r-1} \rho_i^{r-1} + \beta_i^{r-1} \rho_{i+1}^{r-1} + (\|b_i^{r-1}\| + \rho_i^{r-1}) \epsilon_i^{r-1} + (\|b_{i+1}^{r-1}\| + \rho_{i+1}^{r-1}) \delta_i^{r-1}. \quad (5.4)$$

In order to estimate the error bound further, we assume that

$$\begin{aligned} (\alpha_i^{r-1} + \beta_i^{r-1}) &\leq 1, & \|b_i^{r-1}\| + \|b_{i+1}^{r-1}\| &\leq K, \\ \rho_i^r &= \max_{0 \leq i \leq n} \{\rho_i^r\}, & \tau &= \max_{\substack{1 \leq r \leq n \\ 0 \leq i \leq n}} \{\epsilon_i^r, \delta_i^r\}. \end{aligned} \quad (5.5)$$

We then have

$$\begin{aligned} \rho^r &\leq (\alpha_i^{r-1} + \beta_i^{r-1}) \rho^{r-1} + \tau (\rho_i^{r-1} + \rho_{i+1}^{r-1}) + \tau (\|b_i^{r-1}\| + \|b_{i+1}^{r-1}\|) \\ &= (1 + 2\tau) \rho^{r-1} + K\tau \\ &\leq (1 + 2\tau)^2 \rho^{r-2} + \tau [K + K(1 + 2\tau)] \\ &\leq \dots \leq (1 + 2\tau)^r \rho^0 + \frac{1}{2} K [(1 + 2\tau)^r - 1] \\ &\leq e^{2rt} \rho^0 + \frac{1}{2} K (e^{2rt} - 1). \end{aligned} \quad (5.6)$$

This shows that the perturbation error vanishes when ρ^0 and $\tau \rightarrow 0$.

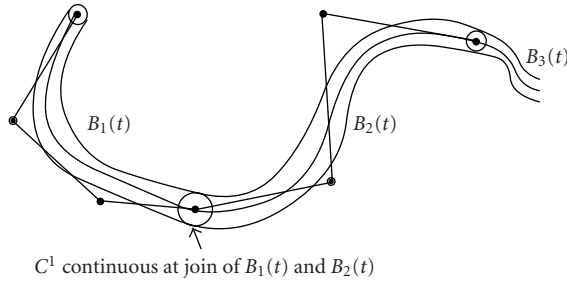


Figure 5.1. The error propagation.

We now consider error propagation for a disk Bézier spline. Let the disk Bézier spline

$$\tilde{B}(t) = \{B_k(t), t \in [0, 1], k = 1, \dots, m\} \tag{5.7}$$

be given, where

$$B_k(t) \subseteq \left\langle \sum_{i+j=n} \binom{n}{i} \binom{n}{j} t^i (1-t)^j b_i^{(k)}, \sum_{i+j=n} \binom{n}{i} \binom{n}{j} t^i (1-t)^j \rho_i^{(k)} \right\rangle \tag{5.8}$$

and where the axis lines of $B_{k-1}(t), B_k(t)$ ($k = 2, \dots, m$) are C^1 continuous; see Figure 5.1.

We now consider how the error perturbation of $B_1(t)$ will affect the other $B_k(t)$ ($k \neq 1$). When the error of $B_1(t)$ is evaluated at the join with $B_2(t)$, the value $\rho_0^{(2)}$ will change from the original value to $\rho_0^{(2)} + \Delta\rho_0$ which gives rise to an additional error radius $(1-t + \delta)^n \Delta\rho_0$ for $B_2(t)$. It can be seen that this extra error increases further to $(1+\delta)^n \Delta\rho_0$ at the initial point of $B_2(t)$ ($t = 0$) due to floating point computational errors. This extra error Δ_0 is bounded by $\delta^n \Delta\rho_0$ ($0 < \delta < 1$) at the terminal of $B_2(t)$ ($t = 1$). This means that although the perturbation increases at the initial point, it is damped during the computational process. In a similar manner this error is transmitted to $B_3(t)$ ($t = 1$) and so on. At the terminal segment $B_m(t)$, the extra error quantity $\Delta\rho_0$ generated from $B_1(t)$ results in $\delta^{mn} \Delta\rho_0$.

As we can see from above, for the disk Bézier curves, the extra perturbation generated from a segment of a curve will be propagated to each of the following segments of the curve along with the control points. This perturbation is attenuated as long as the computation is precise enough, that is, if $0 < \delta < 1$. The construction of a disk Bézier curve is therefore stable.

6. The intersection ball surfaces

The results in Sections 3–5 are now first generalized to ball surfaces and then later to triangular ball patches.

The topological relation between two given surfaces may be changed by a perturbation. In practise we therefore often require that the minimum distance between two surfaces $p_1(s, t)$, $p_2(s, t)$, $s, t \in [0, 1]$, satisfies

$$\min_{\forall s, t, u, v \in [0, 1]} \|p_1(s, t) - p_2(u, v)\| \geq 2\delta^* \quad (6.1)$$

and hence we need to consider the intersection of two ball surfaces

$$B_1(s, t), \quad B_2(u, v), \quad s, t, u, v \in [0, 1], \quad (6.2)$$

with thickness δ^* .

In this section we consider a more general version of the problem. Two ball surfaces $B_i(s, t) = \langle r_i(s, t), \rho_i(s, t) \rangle$, $i = 1, 2$, $s, t \in [0, 1]$, where each $\rho_i(s, t) \in B_i(s, t)$ has to be understood as the error perturbation applied to the surfaces $r_i(s, t)$. We want to check if

$$B_1(s, t) \cap B_2(u, v) = \emptyset, \quad s, t, u, v \in [0, 1], \quad (6.3)$$

holds or not. If the relationship does not hold, then $r_1(s, t)$ and $r_2(s, t)$ may intersect.

From above we can see that our problem is equivalent to the problem of testing for intersection of two ball surfaces. In the following we will use the ball arithmetic to construct a test algorithm for this intersection computation.

Note that

$$\begin{aligned} I^2 &= [0, 1] \times [0, 1], \\ B_1(s, t) &= \langle r_1(s, t), \rho_1(s, t) \rangle, \quad s, t \in I, \\ B_2(u, v) &= \langle r_2(u, v), \rho_2(u, v) \rangle, \quad u, v \in I, \\ X \triangleq (S, T) &= ([\underline{s}, \bar{s}], [\underline{t}, \bar{t}]), \quad S, T \subseteq I, \\ w(X) &= \max \{w(S), w(T)\}, \end{aligned} \quad (6.4)$$

where the notation $X \triangleq (S, T)$ is used to define the variable on the left side of the equation.

Let $B_i(X) = B_i(S, T)$, $S, T \subseteq I$, $i = 1, 2$, be interval extensions of the ball surfaces $B_i(s, t)$, $s, t \in I$, $i = 1, 2$. The proposed algorithm, along the lines of Algorithm 3.1, is based on the exhaustion method which consists of bisection together with interval extension evaluations which are used to progressively delete rectangular subregions of I^2 which correspond to tensor product ball surface pieces guaranteed not to intersect.

Because of the finite machine precision, we also have to prescribe an error bound EPS used to terminate the process.

12 Intersection test and blossoming perturbation

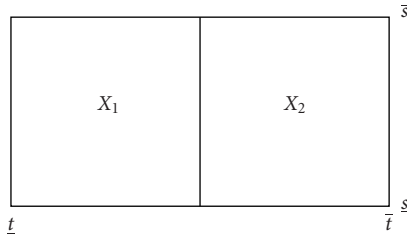


Figure 6.1. Intersection computation.

Given a rectangular subregion $X = (S, T)$ of I^2 as in Figure 6.1, the bisection method consists of subdividing X into two equal rectangles $X_1 \cup X_2 = X$ at the midpoint $\max\{\bar{s} - s, \bar{t} - t\}$ of the longer side of the rectangle.

We use lists \mathcal{X} and \mathcal{Y} to store subregions of I^2 that must be processed further. After each bisection, we put the uncertain subregions that must be tested further at the end of their list. This means that the largest subregion remaining is always first on a list.

For convenience, we let

$$\begin{aligned} \mathcal{B}_1(\mathcal{X}) &= \{B_1(X), \forall X \in \mathcal{X}\}, \\ \mathcal{B}_2(\mathcal{Y}) &= \{B_2(Y), \forall Y \in \mathcal{Y}\} \end{aligned} \quad (6.5)$$

and for any $B_1 \in \mathcal{B}_1(\mathcal{X}), B_2 \in \mathcal{B}_2(\mathcal{Y})$, define

$$\begin{aligned} B_1 \cap \mathcal{B}_2(\mathcal{Y}) &= \{B_1 \cap B, \forall B \in \mathcal{B}_2(\mathcal{Y})\}, \\ \mathcal{B}_1(\mathcal{X}) \cap B_2 &= \{B \cap B_2, \forall B \in \mathcal{B}_1(\mathcal{X})\}. \end{aligned} \quad (6.6)$$

The test algorithm is as follows.

Algorithm 6.1. (1) Initialize lists $\mathcal{X} = \{I^2\}, \mathcal{Y} = \{I^2\}$.

(2) Set $X =$ first item of $\mathcal{X}, Y =$ first item of \mathcal{Y} .

(3) If $w(X) < \text{EPS}$ then go to (7).

(4) Delete X from \mathcal{X} .

(5) Bisect X so that $X = X_1 \cup X_2$.

(6) For $i = 1, 2$ do

(i) Calculate $B_1(X_i)$.

(ii) If $B_1(X_i) \cap \mathcal{B}_2(\mathcal{Y}) \neq \emptyset$ then enter X_i onto the list \mathcal{X} as the last item.

(7) If $w(Y) < \text{EPS}$ then go to (11).

(8) Delete Y from \mathcal{Y} .

(9) Bisect Y so that $Y = Y_1 \cup Y_2$.

(10) For $j = 1, 2$ do

(i) Calculate $B_2(Y_j)$.

(ii) If $\mathcal{B}_1(\mathcal{X}) \cap B_2(Y_j) \neq \emptyset$ then enter Y_j onto the list \mathcal{Y} as the last item.

(11) If $w(X)$ or $w(Y) > \text{EPS}$ then go to (2).

(12) End.

If the algorithm terminates with \mathcal{X} or \mathcal{Y} empty, then the two ball surfaces $B_1(s, t)$, $B_2(u, v)$ do not intersect (to the precision EPS). Conversely, if both \mathcal{X} and \mathcal{Y} are not empty, then $B_1(s, t) \cap B_2(u, v) \neq \emptyset$ holds to the precision EPS, that is, the two ball surfaces intersect. In this case it is necessary to return to the algorithm with a smaller EPS if a more precise result is required.

7. The interval extension of a tensor product ball Bézier surface

In this section we consider the particular case of tensor product ball surfaces.

Let $\{b_{ij}\}, i, j = 0, 1, \dots, n$ be a series of 3D-control points. A tensor product Bézier surface defined on these control points is expressed as

$$b(s, t) = \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q b_{ij}, \quad s, t \in [0, 1]. \quad (7.1)$$

As in the case of Bézier curves, we consider errors both due to perturbations of the control points b_{ij} and due to the perturbations of the parameters s and t .

In order to reflect the perturbations of the control points b_{ij} , we use a control ball $B_{ij} = \langle b_{ij}, \rho_{ij} \rangle$ instead of the original b_{ij} . This results in a ball Bézier surface:

$$\begin{aligned} B(s, t) &= \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q B_{ij} \\ &= \left\langle \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q b_{ij}, \right. \\ &\quad \left. \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q \rho_{ij} \right\rangle \\ &= \left\langle b(s, t), \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q \rho_{ij} \right\rangle, \quad (s, t) \in I^2. \end{aligned} \quad (7.2)$$

This means that a ball Bézier surface can be computed from two scalar Bézier surfaces forming the center and radius of the ball surface, respectively.

In order to include the perturbation of the parameters s, t , we have to consider the interval extension of a ball surface. We use $S = [s - \epsilon, s + \epsilon]$ and $T = [t - \delta, t + \delta]$ instead of the original s, t , where $\epsilon, \delta > 0$ are the error radii.

Let F be the polar form of $b(s, t)$, and let s be written as $s = (1-s) \cdot 0 + s \cdot 1$ and t as $t = (1-t) \cdot 0 + t \cdot 1$, then we obtain

$$\begin{aligned} F(s, \dots, s; t, \dots, t) &= (1-s) \cdot F(s, \dots, s, 0; t, \dots, t) + s \cdot F(s, \dots, s, 1; t, \dots, t) \\ &= (1-s)(1-t) \cdot F(s, \dots, s, 0; t, \dots, t, 0) + (1-s)t \cdot F(s, \dots, s, 0; t, \dots, t, 1) \\ &\quad + s(1-t) \cdot F(s, \dots, s, 1; t, \dots, t, 0) + st \cdot F(s, \dots, s, 1; t, \dots, t, 1) \end{aligned}$$

14 Intersection test and blossoming perturbation

$$\begin{aligned}
 &= \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i \quad p} \binom{n}{j \quad q} s^i (1-s)^p t^j (1-t)^q F \\
 &\quad \times \left(\underbrace{0, \dots, 0}_p, \underbrace{1, \dots, 1}_i; \underbrace{0, \dots, 0}_q, \underbrace{1, \dots, 1}_j \right), \tag{7.3}
 \end{aligned}$$

where $b_{ij} = F(\underbrace{0, \dots, 0}_p, \underbrace{1, \dots, 1}_i; \underbrace{0, \dots, 0}_q, \underbrace{1, \dots, 1}_j)$.

Consider now the interval extension $F(S, \dots, S; T, \dots, T)$ of the polar form F over (S, T) , $S, T \subseteq I$. This is a type of nested form which is an improvement of the Bézier form which is defined as

$$B(S, T) = \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i \quad p} \binom{n}{j \quad q} S^i (1-S)^p T^j (1-T)^q B_{ij}. \tag{7.4}$$

Due to the subdistributivity of interval and ball arithmetic, we have

$$F(S, \dots, S; T, \dots, T) \subseteq B(S, T), \quad S, T \subseteq [0, 1]. \tag{7.5}$$

In practical computations, we use the following de Casteljau recursive process (see [3]) to obtain $F(S, \dots, S; T, \dots, T)$:

$$\begin{aligned}
 &\text{Input } B_{i,j}^{0,0} = B_{ij}, \quad i, j = 0, 1, \dots, n, S, T, \\
 &\quad \text{For } r = 1, \dots, n, \\
 &\quad \quad \text{For } i, j = 0, \dots, n-r, \\
 &\quad \quad \quad B_{i,j}^{r,r} = (1-SS) \begin{pmatrix} B_{i,j}^{r-1,r-1} & B_{i,j+1}^{r-1,r-1} \\ B_{i+1,j}^{r-1,r-1} & B_{i+1,j+1}^{r-1,r-1} \end{pmatrix} \begin{pmatrix} 1-T \\ T \end{pmatrix}. \tag{7.6} \\
 &\quad \quad \text{Output } B_{0,0}^{n,n}. \\
 &\quad \text{Return.}
 \end{aligned}$$

In the above process, S, T are symmetric intervals and $B_{i,j}^{r,r}$ is a ball; hence the result $B_{0,0}^{n,n}$ is a ball, that is, $B_{0,0}^{n,n} = F(S, \dots, S; T, \dots, T)$. Let

$$\begin{aligned}
 \tilde{S}_{ip} &= \binom{n}{i \quad p} S^i (1-S)^p, \quad S \subseteq [0, 1], \\
 \tilde{T}_{jq} &= \binom{n}{j \quad q} T^j (1-T)^q, \quad S \subseteq [0, 1]. \tag{7.7}
 \end{aligned}$$

We have

$$\begin{aligned}
 B(S, T) &= \sum_{i+p=n} \sum_{j+q=n} \tilde{S}_{ip} \tilde{T}_{jq} \cdot \langle b_{ij}, \rho_{ij} \rangle \\
 &= \sum_{i+p=n} \sum_{j+q=n} \left\langle \text{mid}(\tilde{S}_{ip} \tilde{T}_{jq}) \cdot b_{ij}, \text{mid}(\tilde{S}_{ip} \tilde{T}_{jq}) \mid \rho_{ij} \right. \\
 &\quad \left. + \|b_{ij}\| \cdot \text{rad}(\tilde{S}_{ip} \tilde{T}_{jq}) + \rho_{ij} \cdot \text{rad}(\tilde{S}_{ip} \tilde{T}_{jq}) \right\rangle \\
 &= \left\langle \sum_{i+p=n} \sum_{j+q=n} \text{mid}(\tilde{S}_{ip} \tilde{T}_{jq}) \cdot b_{ij}, \sum_{i+p=n} \sum_{j+q=n} \text{mid}(\tilde{S}_{ip} \tilde{T}_{jq}) \mid \rho_{ij} \right. \\
 &\quad \left. + \sum_{i+p=n} \sum_{j+q=n} \|b_{ij}\| \text{rad}(\tilde{S}_{ip} \tilde{T}_{jq}) + \sum_{i+p=n} \sum_{j+q=n} \rho_{ij} \cdot \text{rad}(\tilde{S}_{ip} \tilde{T}_{jq}) \right\rangle.
 \end{aligned} \tag{7.8}$$

Since S, T are disks, they can be written as $\langle s, \epsilon \rangle, \langle t, \delta \rangle$, and we have

$$\begin{aligned}
 \tilde{S}_{ip} \tilde{T}_{jq} &= \binom{n}{i \ p} \binom{n}{j \ q} S^i (1-S)^p T^j (1-T)^q \\
 &= \binom{n}{i \ p} \binom{n}{j \ q} \cdot \langle s^i, (s+\epsilon)^i - s^i \rangle \cdot \langle (1-s)^p, (1-s+\epsilon)^p - (1-s)^p \rangle \\
 &\quad \cdot \langle t^j, (t+\delta)^j - t^j \rangle \cdot \langle (1-t)^q, (1-t+\delta)^q - (1-t)^q \rangle \\
 &= \left\langle \binom{n}{i \ p} \binom{n}{j \ q} \cdot s^i (1-s)^p t^j (1-t)^q, \binom{n}{i \ p} \binom{n}{j \ q} s^i (1-s)^p \right. \\
 &\quad \times [(t+\delta)^j (1-t+\delta)^q - t^j (1-t)^q] \\
 &\quad \left. + \binom{n}{i \ p} \binom{n}{j \ q} [(s+\epsilon)^i (1-s+\epsilon)^p - s^i (1-s)^p] t^j (1-t)^q + \binom{n}{i \ p} \binom{n}{j \ q} \right. \\
 &\quad \left. \times [(s+\epsilon)^i (1-s+\epsilon)^p - s^i (1-s)^p] \cdot [(t+\delta)^j (1-t+\delta)^q - t^j (1-t)^q] \right\rangle \\
 &= \left\langle \binom{n}{i \ p} \binom{n}{j \ q} s^i (1-s)^p t^j (1-t)^q, \binom{n}{i \ p} \binom{n}{j \ q} \right. \\
 &\quad \left. \times [(s+\epsilon)^i (1-s+\epsilon)^p (t+\delta)^j (1-t+\delta)^q - s^i (1-s)^p t^j (1-t)^q] \right\rangle,
 \end{aligned} \tag{7.9}$$

where $0 \leq s + \epsilon, 1 - s + \epsilon, t + \delta, 1 - t + \delta \leq 1$. Therefore

$$\begin{aligned}
F(S, \dots, S; T, \dots, T) &\subseteq B(S, T) \\
&= \left\langle \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q b_{ij}, \right. \\
&\quad \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} [(s+\epsilon)^i (1-s+\epsilon)^p (t+\delta)^j \\
&\quad \times (1-t+\delta)^q - s^i (1-s)^p t^j (1-t)^q] \cdot \|b_{ij}\| \\
&\quad + \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} \\
&\quad \left. \times (s+\epsilon)^i (1-s+\epsilon)^p (t+\delta)^j (1-t+\delta)^q \cdot \rho_{ij} \right\rangle. \tag{7.10}
\end{aligned}$$

When the perturbations of the control points and the error radii ϵ, δ of the evaluation point tend to zero, we would expect that the value of the tensor product Bézier surface to be obtained. This is indeed the case as can be seen from (7.10), and we have the following theorem for the extension of the polar form.

For the extension to polar form the following theorem follows.

THEOREM 7.1.

$$\text{rad}(F(S, \dots, S; T, \dots, T) - b(s, t)) \longrightarrow 0 \quad (\text{as } \rho, w(S, T) \longrightarrow 0), \tag{7.11}$$

where $\rho = \max_{i,j} \{\rho_{ij}\}, (s, t) = \text{mid}(S, T)$.

8. Ball analysis of error generated from blossoming for tensor product surfaces

A general ball surface generated by means of the following recursive formula:

$$B_{i,j}^{r,r} = \left(\alpha_i^{r-1} \beta_i^{r-1} \right) \begin{pmatrix} B_{i,j}^{r-1,r-1} & B_{i,j+1}^{r-1,r-1} \\ B_{i+1,j}^{r-1,r-1} & B_{i+1,j+1}^{r-1,r-1} \end{pmatrix} \begin{pmatrix} \gamma_j^{r-1} \\ \omega_j^{r-1} \end{pmatrix}, \quad r = 1, \dots, n, i, j = 0, \dots, n-r, \tag{8.1}$$

where

$$B_{i,j}^{0,0} = B_{ij} = \langle b_{ij}, \rho_{ij} \rangle, \quad i, j = 0, \dots, n, \alpha_i^{r-1}, \beta_i^{r-1}, \gamma_j^{r-1}, \omega_j^{r-1} \in [0, 1] \tag{8.2}$$

is considered, where the ball blossoming is $B_{0,0}^{n,n}$.

The blossoming algorithm is used to account for the error propagation in the computation of a ball Bézier surface. To analyze the errors in the algorithm the following balls

are defined:

$$\begin{aligned}\Lambda_i^r &= \langle \alpha_i^r, \epsilon_i^r \rangle, & \Delta_i^r &= \langle \beta_i^r, \delta_i^r \rangle, & \Gamma_j^r &= \langle \gamma_j^r, \eta_j^r \rangle, \\ \Omega_j^r &= \langle \omega_j^r, \sigma_j^r \rangle, & B_{i,j}^{r,r} &= \langle b_{i,j}^{r,r}, \rho_{i,j}^{r,r} \rangle.\end{aligned}\quad (8.3)$$

When the above perturbations are entered into (8.1), we get

$$\begin{aligned}\langle b_{i,j}^{r,r}, \rho_{i,j}^{r,r} \rangle &= \left(\Lambda_i^{r-1} \Delta_i^{r-1} \right) \begin{pmatrix} \langle b_{i,j}^{r-1,r-1}, \rho_{i,j}^{r-1,r-1} \rangle & \langle b_{i,j+1}^{r-1,r-1}, \rho_{i,j+1}^{r-1,r-1} \rangle \\ \langle b_{i+1,j}^{r-1,r-1}, \rho_{i+1,j}^{r-1,r-1} \rangle & \langle b_{i+1,j+1}^{r-1,r-1}, \rho_{i+1,j+1}^{r-1,r-1} \rangle \end{pmatrix} \begin{pmatrix} \Gamma_j^{r-1} \\ \Omega_j^{r-1} \end{pmatrix} \\ &= \Lambda_i^{r-1} \Gamma_j^{r-1} \cdot \langle b_{i,j}^{r-1,r-1}, \rho_{i,j}^{r-1,r-1} \rangle + \Delta_i^{r-1} \Gamma_j^{r-1} \cdot \langle b_{i+1,j}^{r-1,r-1}, \rho_{i+1,j}^{r-1,r-1} \rangle \\ &\quad + \Lambda_i^{r-1} \Omega_j^{r-1} \cdot \langle b_{i,j+1}^{r-1,r-1}, \rho_{i,j+1}^{r-1,r-1} \rangle + \Delta_i^{r-1} \Omega_j^{r-1} \cdot \langle b_{i+1,j+1}^{r-1,r-1}, \rho_{i+1,j+1}^{r-1,r-1} \rangle \\ &= \langle \alpha_i^{r-1} \gamma_j^{r-1} b_{i,j}^{r-1,r-1}, \alpha_i^{r-1} \gamma_j^{r-1} \rho_{i,j}^{r-1,r-1} \rangle \\ &\quad + (\alpha_i^{r-1} \eta_j^{r-1} + \gamma_j^{r-1} \epsilon_i^{r-1} + \eta_j^{r-1} \epsilon_i^{r-1}) (\|b_{i,j}^{r-1,r-1}\| + \rho_{i,j}^{r-1,r-1}) \\ &\quad + \langle \beta_i^{r-1} \gamma_j^{r-1} b_{i+1,j}^{r-1,r-1}, \beta_i^{r-1} \gamma_j^{r-1} \rho_{i+1,j}^{r-1,r-1} \rangle \\ &\quad + (\beta_i^{r-1} \eta_j^{r-1} + \gamma_j^{r-1} \delta_i^{r-1} + \eta_j^{r-1} \delta_i^{r-1}) (\|b_{i+1,j}^{r-1,r-1}\| + \rho_{i+1,j}^{r-1,r-1}) \\ &\quad + \langle \alpha_i^{r-1} \omega_j^{r-1} b_{i,j+1}^{r-1,r-1}, \alpha_i^{r-1} \omega_j^{r-1} \rho_{i,j+1}^{r-1,r-1} \rangle \\ &\quad + (\alpha_i^{r-1} \sigma_j^{r-1} + \omega_j^{r-1} \epsilon_i^{r-1} + \sigma_j^{r-1} \epsilon_i^{r-1}) (\|b_{i,j+1}^{r-1,r-1}\| + \rho_{i,j+1}^{r-1,r-1}) \\ &\quad + \langle \beta_i^{r-1} \omega_j^{r-1} b_{i+1,j+1}^{r-1,r-1}, \beta_i^{r-1} \omega_j^{r-1} \rho_{i+1,j+1}^{r-1,r-1} \rangle \\ &\quad + (\beta_i^{r-1} \sigma_j^{r-1} + \omega_j^{r-1} \delta_i^{r-1} + \sigma_j^{r-1} \delta_i^{r-1}) (\|b_{i+1,j+1}^{r-1,r-1}\| + \rho_{i+1,j+1}^{r-1,r-1}).\end{aligned}\quad (8.4)$$

If we only consider the error radius, we have

$$\begin{aligned}\rho_{i,j}^{r,r} &= \alpha_i^{r-1} \gamma_j^{r-1} \rho_{i,j}^{r-1,r-1} + \beta_i^{r-1} \gamma_j^{r-1} \rho_{i+1,j}^{r-1,r-1} + \alpha_i^{r-1} \omega_j^{r-1} \rho_{i,j+1}^{r-1,r-1} + \beta_i^{r-1} \omega_j^{r-1} \rho_{i+1,j+1}^{r-1,r-1} \\ &\quad + (\|b_{i,j}^{r-1,r-1}\| + \rho_{i,j}^{r-1,r-1}) (\alpha_i^{r-1} \eta_j^{r-1} + \gamma_j^{r-1} \epsilon_i^{r-1} + \eta_j^{r-1} \epsilon_i^{r-1}) \\ &\quad + (\|b_{i+1,j}^{r-1,r-1}\| + \rho_{i+1,j}^{r-1,r-1}) (\beta_i^{r-1} \eta_j^{r-1} + \gamma_j^{r-1} \delta_i^{r-1} + \eta_j^{r-1} \delta_i^{r-1}) \\ &\quad + (\|b_{i,j+1}^{r-1,r-1}\| + \rho_{i,j+1}^{r-1,r-1}) (\alpha_i^{r-1} \sigma_j^{r-1} + \omega_j^{r-1} \epsilon_i^{r-1} + \sigma_j^{r-1} \epsilon_i^{r-1}) \\ &\quad + (\|b_{i+1,j+1}^{r-1,r-1}\| + \rho_{i+1,j+1}^{r-1,r-1}) (\beta_i^{r-1} \sigma_j^{r-1} + \omega_j^{r-1} \delta_i^{r-1} + \sigma_j^{r-1} \delta_i^{r-1}).\end{aligned}\quad (8.5)$$

In order to further estimate the error bound, the quantities in the error radius have to be bounded as follows:

$$\begin{aligned}(\alpha_i^{r-1} + \beta_i^{r-1}) &\leq 1, & (\gamma_j^{r-1} + \omega_j^{r-1}) &\leq 1, \\ \|b_{i,j}^{r-1,r-1}\| + \|b_{i+1,j}^{r-1,r-1}\| + \|b_{i,j+1}^{r-1,r-1}\| + \|b_{i+1,j+1}^{r-1,r-1}\| &\leq K, \\ \rho^r &= \max_{i,j} \{\rho_{i,j}^{r,r}\}, & \tau &= \max_{i,j,r} \{\epsilon_i^r, \delta_i^r, \eta_j^r, \sigma_j^r\}.\end{aligned}\quad (8.6)$$

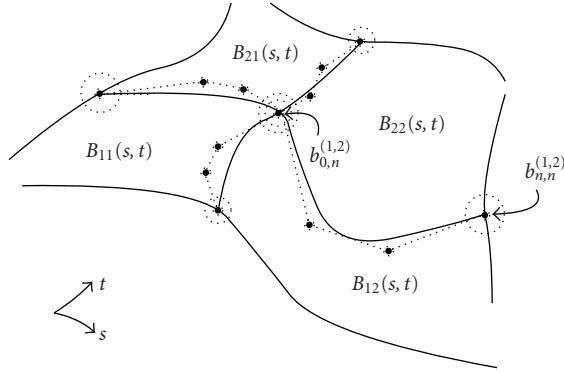


Figure 8.1. Perturbation for tensor product surface.

The error radius can then be estimated as

$$\begin{aligned}
 \rho^r &\leq (\alpha_i^{r-1} + \beta_i^{r-1})(\gamma_j^{r-1} + \omega_j^{r-1})\rho^{r-1} \\
 &\quad + 3\tau^2(\rho_{i,j}^{r-1,r-1} + \rho_{i+1,j}^{r-1,r-1} + \rho_{i,j+1}^{r-1,r-1} + \rho_{i+1,j+1}^{r-1,r-1}) \\
 &\quad + 3\tau^2(\|b_{i,j}^{r-1,r-1}\| + \|b_{i+1,j}^{r-1,r-1}\| + \|b_{i,j+1}^{r-1,r-1}\| + \|b_{i+1,j+1}^{r-1,r-1}\|) \\
 &= (1 + 12\tau^2)\rho^{r-1} + 3K\tau^2 \\
 &\leq \dots \leq (1 + 12\tau^2)^r \rho^0 + 3K\tau^2[1 + (1 + 2\tau^2) + \dots + (1 + 2\tau^2)^{r-1}] \\
 &= (1 + 12\tau^2)^r \rho^0 + \frac{1}{4}K[(1 + 12\tau^2)^r - 1] \\
 &\leq e^{12r\tau^2} \rho^0 + \frac{1}{4}K(e^{12r\tau^2} - 1).
 \end{aligned} \tag{8.7}$$

The error propagation among tensor product ball Bézier surfaces is considered next. Let the ball Bézier surface

$$\tilde{B}(s, t) = \{B_{k,l}(s, t), s, t \in [0, 1], k, l = 1, \dots, m\} \tag{8.8}$$

be given, where

$$\begin{aligned}
 B_{k,l}(s, t) &\subseteq \left\langle \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q b_{i,j}^{(k,l)}, \right. \\
 &\quad \left. \sum_{i+p=n} \sum_{j+q=n} \binom{n}{i} \binom{n}{p} \binom{n}{j} \binom{n}{q} s^i (1-s)^p t^j (1-t)^q \rho_{i,j}^{(k,l)} \right\rangle.
 \end{aligned} \tag{8.9}$$

Moreover, we require that $B_{k-1,l}(s, t)$ and $B_{k,l}(s, t)$ ($\forall l$) are C^1 continuous on their boundary curve, and that $B_{k,l-1}(s, t)$ and $B_{k,l}(s, t)$ ($\forall k$) are C^1 continuous on their boundary curve; see also Figure 8.1.

We wish to assess the effect of an error perturbation of $B_{11}(s, t)$ on the other $B_{k,l}(s, t)$ (k, l not equal to 1 at the same time).

When the error of $B_{11}(s, t)$ is transmitted to the curve

$$B_{1,2}(0, t) = \left\langle \sum_{j+q=n} \binom{n}{j \quad q} t^j (1-t)^q b_{0,j}^{(1,2)}, \sum_{j+q=n} \binom{n}{j \quad q} t^j (1-t)^q \rho_{0,j}^{(1,2)} \right\rangle \quad (8.10)$$

linked to $B_{12}(s, t)$, the value of $\rho_{0,j}^{(1,2)}$ changes from its original value to $\rho_{0,j}^{(1,2)} + \Delta\rho_0$, which then gives rise to a change of the global error radius of $B_{12}(s, t)$. In this case the change of the error of $B_{11}(s, t)$ will add an extra quantity that is no larger than

$$\begin{aligned} & \sum_{j+q=n} \binom{n}{j \quad q} (1-s+\epsilon)^n (t+\delta)^j (1-t+\delta)^q \Delta\rho_0 \\ &= (1-s+\epsilon)^n \sum_{j+q=n} \binom{n}{j \quad q} (t+\delta)^j (1-t+\delta)^q \Delta\rho_0 \\ &= (1-s+\epsilon)^n (1+2\delta)^n \Delta\rho_0 \end{aligned} \quad (8.11)$$

to the error radius of $B_{12}(s, t)$. As this extra error is transmitted to the two boundary curves ($s = 1, t = 1$) of $B_{12}(s, t)$, it will give rise to an increase in the error radii of those curves bounded by

$$\epsilon^n \cdot (1+2\delta)^n \cdot \Delta\rho_0. \quad (8.12)$$

When ($s = 1, t = 1$), that is, at the control ball $b_{n,n}^{(1,2)}$, the extra error quantity is

$$\epsilon^n \cdot (1+2\delta)^n \cdot \Delta\rho_0 < \Delta\rho_0 \quad (8.13)$$

as long as the computational precision $\epsilon < (1+2\delta)^{-1}$ satisfies $\epsilon < 1/3$.

It should be noted that when $s = 0, t = 1$, the extra error quantity at the control ball $b_{0,n}^{(1,2)}$ is

$$(1+\epsilon)^n \cdot (1+2\delta)^n \cdot \Delta\rho_0 > \Delta\rho_0, \quad \forall \epsilon, \delta > 0. \quad (8.14)$$

As can be seen from the above, the extra perturbation generated from a ball Bézier surface patch will be propagated to each of the following surface patches along with the control balls. In the general case, if the computation is not precise enough, that is, ϵ, δ are not sufficiently small, we can find a control ball $b_{n,n}^{(1,2)}$, where the extra error tends to enlarge at this ball. Therefore the construction of tensor product ball Bézier surfaces may be computationally unstable, that is, the construction for curves is absolutely stable but that for surfaces is not.

9. The intersection of triangular ball patches

We sometimes have to consider the intersection of triangular ball patches in the applications where triangular patches are defined in, for example, [3].

Letting the parameters (s, t) be the barycentric coordinates with regard to some domain triangle, we require that the minimum distance between two triangular patches $p_1(s, t), p_2(s, t), s, t \in [0, 1]$, satisfies

$$\min_{\forall s, t, u, v \in [0, 1]} \|p_1(s, t) - p_2(s, t)P\| \geq 2\delta^*. \quad (9.1)$$

We must therefore study the intersection of two triangular ball patches $B_1(s, t), B_2(s, t), s, t \in [0, 1]$, both having thickness δ^* .

We are given $B_i(s, t) = \langle r_i(s, t), \rho_i(s, t) \rangle, i = 1, 2, s, t \in [0, 1]$, where each $\rho_i(s, t) \in B_i(s, t)$ will be understood as a deformation of $r_i(s, t)$ due to an error, and we are going to consider whether

$$B_1(s, t) \cap B_2(u, v) = \emptyset, \quad s, t, u, v \in [0, 1], \quad (9.2)$$

holds. If not, it is possible that $r_1(s, t)$ and $r_2(s, t)$ will intersect because of the error perturbation.

From the above we can see that the problem can be viewed as one of intersecting two triangular ball patches. In the following we will therefore construct a test algorithm for the intersection of two triangular ball patches.

Note that

$$\begin{aligned} B_1(s, t) &= \langle r_1(s, t), \rho_1(s, t) \rangle, \quad s, t \in I, \\ B_2(u, v) &= \langle r_2(u, v), \rho_2(u, v) \rangle, \quad u, v \in I, \\ X &\triangleq (S, T) = ([\underline{s}, \bar{s}], [\underline{t}, \bar{t}]), \quad S, T \subseteq I, \\ w(X) &= \max \{w(S), w(T)\}. \end{aligned} \quad (9.3)$$

Let $B_i(S, T), S, T \subseteq I, i = 1, 2$, be an interval extension of the triangular ball patches $B_i(s, t), s, t \in I, i = 1, 2$. The proposed algorithm, which is another generalization of Algorithm 3.1, is based on bisection together with interval extension evaluations which are used to delete subregions of (I, I) which correspond to the two pieces of ball patches guaranteed not to intersect. For a subregion $X = (S, T)$ of (I, I) , the bisection method is the same as for tensor product surfaces, that is, X is partitioned into two subregions $X_1 \cup X_2 = X$ at the midpoint $\max\{\bar{s} - \underline{s}, \bar{t} - \underline{t}\}$ of the interval component with the longer length. For instance, if $\bar{s} - \underline{s} < \bar{t} - \underline{t}$, then

$$\left(S, \left[\underline{t}, \frac{\bar{t} + \underline{t}}{2} \right] \right) \cup \left(S, \left[\frac{\bar{t} + \underline{t}}{2}, \bar{t} \right] \right) = X. \quad (9.4)$$

We use a list \mathcal{X} to store the subregions of (I, I) for $B_1(X)$ that must be processed further, and use another list \mathcal{Y} to store the subregions of (I, I) for $B_2(Y)$ that must be processed further.

After each bisection we put the uncertain subregions that must be tested further at the end of their list.

For convenience, we let

$$\mathfrak{B}_1(\mathcal{X}) = \{B_1(X), \forall X \in \mathcal{X}\}, \quad \mathfrak{B}_2(\mathcal{Y}) = \{B_2(Y), \forall Y \in \mathcal{Y}\} \quad (9.5)$$

and for any $B_1 \in \mathfrak{B}_1(\mathcal{X}), B_2 \in \mathfrak{B}_2(\mathcal{Y})$, we define

$$\begin{aligned} B_1 \cap \mathfrak{B}_2(\mathcal{Y}) &= \{B_1 \cap B, \forall B \in \mathfrak{B}_2(\mathcal{Y})\}, \\ \mathfrak{B}_1(\mathcal{X}) \cap B_2 &= \{B \cap B_2, \forall B \in \mathfrak{B}_1(\mathcal{X})\}. \end{aligned} \quad (9.6)$$

The test algorithm is as follows.

Algorithm 9.1. (1) Initialize lists $\mathcal{X} = \{(I, I)\}, \mathcal{Y} = \{(I, I)\}$.

(2) Set $X =$ first item on $\mathcal{X}, Y =$ first item on \mathcal{Y} .

(3) If $w(X) < \text{EPS}$ then go to (7).

(4) Delete X from \mathcal{X} .

(5) Bisect X so that $X = X_1 \cup X_2$.

(6) For $i = 1, 2$ do

(i) Calculate $B_1(X_i)$.

(ii) If $B_1(X_i) \cap \mathfrak{B}_2(\mathcal{Y}) \neq \emptyset$ then enter X_i onto the list \mathcal{X} as the last item.

(7) If $w(Y) < \text{EPS}$ then go to (11).

(8) Delete Y from \mathcal{Y} .

(9) Bisect Y so that $Y = Y_1 \cup Y_2$.

(10) For $j = 1, 2$ do

(i) Calculate $B_2(Y_j)$.

(ii) If $\mathfrak{B}_1(\mathcal{X}) \cap B_2(Y_j) \neq \emptyset$ then enter Y_j onto the list \mathcal{Y} as the last item.

(11) If $w(X)$ or $w(Y) > \text{EPS}$ then go to (2).

(12) End.

10. The interval extension of a triangular ball Bézier patch

Let $\{b_{ijk}\}, i + j + k = n$, be a set of 3D-control points. A triangular Bézier patch defined by these control points is expressed as

$$b_{\Delta}(s, t) = \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k b_{ijk}, \quad s, t, (1-s-t) \in [0, 1], \quad (10.1)$$

where

$$\binom{n}{i \quad j \quad k} = \frac{n!}{i!j!k!}. \quad (10.2)$$

In order to consider the error perturbation of the control points b_{ijk} , we use a control ball $B_{ijk} = \langle b_{ijk}, \rho_{ijk} \rangle$ instead of the original b_{ijk} . This results in a ball Bézier patch

$$\begin{aligned}
 B_{\Delta}(s, t) &= \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k B_{ijk} \\
 &= \left\langle \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k b_{ijk}, \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k \rho_{ijk} \right\rangle \\
 &= \left\langle b_{\Delta}(s, t), \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k \rho_{ijk} \right\rangle, \quad s, t \in I.
 \end{aligned} \tag{10.3}$$

In order to further consider the error perturbation of the parameter s, t , we consider the interval extension of a ball patch. We use $S = [s - \epsilon, s + \epsilon]$ and $T = [t - \delta, t + \delta]$ instead of the original s and t . Since the parameters (s, t) are the barycentric coordinates with regard to some domain triangle $\Delta = \Delta(\lambda, \mu, \theta)$, a point z on Δ can be represented as

$$z = s\lambda + t\mu + (1-s-t)\theta. \tag{10.4}$$

Let F_{Δ} be the polar form of b_{Δ} , then we have

$$\begin{aligned}
 F_{\Delta}(z, \dots, z) &= sF_{\Delta}(z, \dots, z, \lambda) + tF_{\Delta}(z, \dots, z, \mu) + (1-s-t)F_{\Delta}(z, \dots, z, \theta) \\
 &= \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k F_{\Delta} \left(\underbrace{\lambda, \dots, \lambda}_i, \underbrace{\mu, \dots, \mu}_j, \underbrace{\theta, \dots, \theta}_k \right) \\
 &= b_{\Delta}(s, t),
 \end{aligned} \tag{10.5}$$

where $b_{\Delta}(s, t) = F_{\Delta}(\underbrace{\lambda, \dots, \lambda}_i, \underbrace{\mu, \dots, \mu}_j, \underbrace{\theta, \dots, \theta}_k)$.

Consider the interval extension $F_{\Delta}(S, \dots, S, T, \dots, T)$ of the polar form F_{Δ} on the (S, T) , $S, T \subseteq I$. This is a type of nested form which is an improvement of the following Bézier extension:

$$B_{\Delta}(S, T) = \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} S^i T^j (1-S-T)^k B_{ijk}. \tag{10.6}$$

Due to the subdistributivity of interval or ball arithmetic, we have

$$F_{\Delta}(S, \dots, S, T, \dots, T) \subseteq B_{\Delta}(S, T), \quad S, T \subseteq [0, 1]. \tag{10.7}$$

In practical computations we use the following de Casteljau recursive process (see [3]) to

obtain $F_\Delta(S, \dots, S, T, \dots, T)$:

$$\begin{aligned}
 &\text{Input } B_{i,j,k}^0 = B_{ijk}, i = 0, 1, \dots, n, S, T, \\
 &\text{For } r = 1, \dots, n, \\
 &\quad \text{For } i + j + k = n - r, \\
 &\quad \quad B_{i,j,k}^r = SB_{i+1,j,k}^{r-1} + TB_{i,j+1,k}^{r-1} + (1 - S - T)B_{i,j,k+1}^{r-1}, \\
 &\quad \text{Output } B_{0,0,0}^n, \\
 &\text{Return.}
 \end{aligned} \tag{10.8}$$

That is, $B_{0,0,0}^n = F_\Delta(S, \dots, S, T, \dots, T)$.

Letting

$$\widetilde{W}_{ijk} = \sum_{i+j+k=n} \binom{n}{i \ j \ k} S^i T^j (1 - S - T)^k, \quad S, T \in [0, 1], \tag{10.9}$$

we have

$$\begin{aligned}
 B_\Delta(S, T) &= \sum_{i+j+k=n} \widetilde{W}_{ijk} \cdot \langle b_{ijk}, \rho_{ijk} \rangle \\
 &= \sum_{i+j+k=n} \left\langle \text{mid } \widetilde{W}_{ijk} \cdot b_{ijk}, \left| \text{mid } \widetilde{W}_{ijk} \right| \cdot \rho_{ijk} + \|b_{ijk}\| \cdot \text{rad } \widetilde{W}_{ijk} + \rho_{ijk} \cdot \text{rad } \widetilde{W}_{ijk} \right\rangle \\
 &= \left\langle \sum_{i+j+k=n} \text{mid } \widetilde{W}_{ijk} \cdot b_{ijk}, \sum_{i+j+k=n} \left| \text{mid } \widetilde{W}_{ijk} \right| \cdot \rho_{ijk} \right. \\
 &\quad \left. + \sum_{i+j+k=n} \|b_{ijk}\| \text{rad } \widetilde{W}_{ijk} + \sum_{i+j+k=n} \rho_{ijk} \cdot \text{rad } \widetilde{W}_{ijk} \right\rangle.
 \end{aligned} \tag{10.10}$$

By considering S, T as disks they can be expressed as $\langle s, \epsilon \rangle, \langle t, \delta \rangle$, and we have

$$\begin{aligned}
 \widetilde{W}_{ijk} &= \binom{n}{i \ j \ k} \cdot \langle s^i, (s + \epsilon)^i - s^i \rangle \cdot \langle t^j, (t + \delta)^j - t^j \rangle \\
 &\quad \cdot \langle (1 - s - t)^k, (1 - s - t + \epsilon + \delta)^k - (1 - s - t)^k \rangle \\
 &= \binom{n}{i \ j \ k} \cdot \langle s^i t^j, (s + \epsilon)^i (t + \delta)^j - s^i t^j \rangle \\
 &\quad \cdot \langle (1 - s - t)^k, (1 - s - t + \epsilon + \delta)^k - (1 - s - t)^k \rangle \\
 &= \left\langle \binom{n}{i \ j \ k} s^i t^j (1 - s - t)^k, \right. \\
 &\quad \left. \binom{n}{i \ j \ k} [(s + \epsilon)^i (t + \delta)^j (1 - s - t + \epsilon + \delta)^k \right. \\
 &\quad \left. - s^i t^j (1 - s - t)^k] \right\rangle,
 \end{aligned} \tag{10.11}$$

where it is assumed that $0 \leq s + \epsilon, t + \delta, 1 - s - t + \epsilon + \delta \leq 1$. Therefore

$$\begin{aligned}
 & F_{\Delta}(S, \dots, S, T, \dots, T) \\
 & \subseteq B_{\Delta}(S, T) \left\langle \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} s^i t^j (1-s-t)^k b_{ijk}, \right. \\
 & \quad \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} [(s+\epsilon)^i (t+\delta)^j (1-s-t+\epsilon+\delta)^k - s^i t^j (1-s-t)^k] \\
 & \quad \cdot \|b_{ijk}\| + \left. \sum_{i+j+k=n} \binom{n}{i \quad j \quad k} (s+\epsilon)^i (t+\delta)^j (1-s-t+\epsilon+\delta)^k \cdot \rho_{ijk} \right\rangle. \tag{10.12}
 \end{aligned}$$

For the extension of polar form the following theorem follows.

THEOREM 10.1.

$$\text{rad}(F_{\Delta}(S, \dots, S, T, \dots, T) - b_{\Delta}(s, t)) \longrightarrow 0 \quad (\text{as } \rho, w(S, T) \longrightarrow 0), \tag{10.13}$$

where $\rho = \max_{i+j+k=n} \{\rho_{ijk}\}$, $s = \text{mid} S$, $t = \text{mid} T$.

11. Ball analysis of error generated from blossoming for triangular patches

We first use the blossoming algorithm to take account of the error propagation in the computation of a triangular ball Bézier patch so as to analyze its computational stability.

A general triangular ball patch can be generated by means of the following recursive formula:

$$B_{i,j,k}^r = \alpha_i^{r-1} B_{i+1,j,k}^{r-1} + \beta_j^{r-1} B_{i,j+1,k}^{r-1} + \gamma_k^{r-1} B_{i,j,k+1}^{r-1}, \quad r = 1, \dots, n, i + j + k = n - r, \tag{11.1}$$

where

$$B_{i,j,k}^0 = B_{ijk} = \langle b_{ijk}, \rho_{ijk} \rangle, \quad i + j + k = 0, \dots, n, \alpha_i^{r-1}, \beta_j^{r-1}, \gamma_k^{r-1} \in [0, 1]. \tag{11.2}$$

The ball blossoming $B_{0,0,0}^n$ is obtained from this computation.

Letting

$$\Lambda_i^r = \langle \alpha_i^r, \epsilon_i^r \rangle, \quad \Delta_i^r = \langle \beta_i^r, \delta_i^r \rangle, \quad \Gamma_i^r = \langle \gamma_i^r, \eta_i^r \rangle, \quad B_{i,j,k}^r = \langle b_{i,j,k}^r, \rho_{i,j,k}^r \rangle, \tag{11.3}$$

it follows from (11.1) that

$$\begin{aligned}
 \langle b_{i,j,k}^r, \rho_{i,j,k}^r \rangle &= \Lambda_i^{r-1} \cdot \langle b_{i+1,j,k}^{r-1}, \rho_{i+1,j,k}^{r-1} \rangle + \Delta_j^{r-1} \cdot \langle b_{i,j+1,k}^{r-1}, \rho_{i,j+1,k}^{r-1} \rangle + \Gamma_k^{r-1} \cdot \langle b_{i,j,k+1}^{r-1}, \rho_{i,j,k+1}^{r-1} \rangle \\
 &= \langle \alpha_i^{r-1} b_{i+1,j,k}^{r-1}, \alpha_i^{r-1} \rho_{i+1,j,k}^{r-1} + (\|b_{i+1,j,k}^{r-1}\| + \rho_{i+1,j,k}^{r-1}) \epsilon_i^{r-1} \rangle \\
 &\quad + \langle \beta_j^{r-1} b_{i,j+1,k}^{r-1}, \beta_j^{r-1} \rho_{i,j+1,k}^{r-1} + (\|b_{i,j+1,k}^{r-1}\| + \rho_{i,j+1,k}^{r-1}) \delta_j^{r-1} \rangle \\
 &\quad + \langle \gamma_k^{r-1} b_{i,j,k+1}^{r-1}, \gamma_k^{r-1} \rho_{i,j,k+1}^{r-1} + (\|b_{i,j,k+1}^{r-1}\| + \rho_{i,j,k+1}^{r-1}) \eta_k^{r-1} \rangle.
 \end{aligned} \tag{11.4}$$

Considering only the error radius, we have

$$\begin{aligned}
 \rho_{ijk} &= \alpha_i^{r-1} \rho_{i+1,j,k}^{r-1} + \beta_j^{r-1} \rho_{i,j+1,k}^{r-1} + \gamma_k^{r-1} \rho_{i,j,k+1}^{r-1} \\
 &\quad + (\|b_{i+1,j,k}^{r-1}\| + \rho_{i+1,j,k}^{r-1}) \epsilon_i^{r-1} + (\|b_{i,j+1,k}^{r-1}\| + \rho_{i,j+1,k}^{r-1}) \delta_j^{r-1} \\
 &\quad + (\|b_{i,j,k+1}^{r-1}\| + \rho_{i,j,k+1}^{r-1}) \eta_k^{r-1}.
 \end{aligned} \tag{11.5}$$

In order to further estimate the error bound, let

$$\begin{aligned}
 (\alpha_i^{r-1} + \beta_j^{r-1} + \gamma_k^{r-1}) &\leq 1, \\
 (\|b_{i+1,j,k}^{r-1}\| + \|b_{i,j+1,k}^{r-1}\| + \|b_{i,j,k+1}^{r-1}\|) &\leq K, \\
 \rho^r &= \max_{i+j+k=n} \{\rho_{i,j,k}^r\}, \quad \tau = \max_{i,j,k,r} \{\epsilon_i^r, \delta_j^r, \eta_k^r\}.
 \end{aligned} \tag{11.6}$$

Then we have

$$\begin{aligned}
 \rho^r &\leq (\alpha_i^{r-1} + \beta_j^{r-1} + \gamma_k^{r-1}) \rho^{r-1} + \tau (\rho_{i+1,j,k}^{r-1} + \rho_{i,j+1,k}^{r-1} + \rho_{i,j,k+1}^{r-1}) \\
 &\quad + \tau (\|b_{i+1,j,k}^{r-1}\| + \|b_{i,j+1,k}^{r-1}\| + \|b_{i,j,k+1}^{r-1}\|) \\
 &= (1 + 3\tau) \rho^{r-1} + K\tau \\
 &\leq \dots \leq (1 + 3\tau)^r \rho^0 + K\tau [1 + (1 + 3\tau) + \dots + (1 + 3\tau)^{r-1}] \\
 &= (1 + 3\tau)^r \rho^0 + \frac{1}{3} K [(1 + 3\tau)^r - 1] \leq e^{3r\tau} \rho^0 + \frac{1}{3} K (e^{3r\tau} - 1).
 \end{aligned} \tag{11.7}$$

Next we consider error propagation among triangular ball Bézier patches as indicated in Figure 11.1.

Let the triangular ball Bézier patch:

$$\tilde{B}_\Delta(s, t) = \{B_l(s, t), s, t \in [0, 1], l = 1, \dots, m\}, \tag{11.8}$$

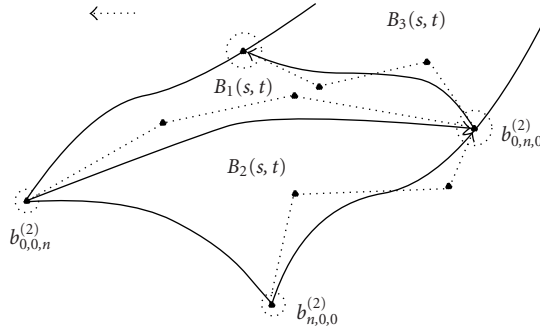


Figure 11.1. Perturbation of triangular patch.

where

$$B_l(s,t) \subseteq \left\langle \sum_{i+j+k=n} \binom{n}{i \ j \ k} s^i t^j (1-s-t)^k b_{ijk}^{(l)}, \right. \\ \left. \sum_{i+j+k=n} \binom{n}{i \ j \ k} s^i t^j (1-s-t)^k \rho_{ijk}^{(l)} \right\rangle, \quad (11.9)$$

be given.

Moreover, two adjacent patches $B_{v_1}(s,t)$ and $B_{v_2}(s,t)$ ($v_1 \neq v_2$) are assumed to be C^1 continuous on their common boundary curve. We now wish to discuss how the error perturbation of $B_1(s,t)$ disrupts the other $B_l(s,t)$ ($l \neq 1$).

When the error of $B_1(s,t)$ is transmitted to the curve

$$B_2(0,t) \subseteq \left\langle \sum_{j+k=n} \binom{n}{i \ j \ k} t^j (1-t)^k b_{0jk}^{(2)}, \sum_{j+k=n} \binom{n}{i \ j \ k} t^j (1-t)^k \rho_{0jk}^{(2)} \right\rangle \quad (11.10)$$

which is linked up, $B_2(s,t)$ the value of $\rho_{0,j,k}^{(2)}$ will change from the original value to $\rho_{0,j,k}^{(2)} + \Delta\rho_0$ which then gives rise to a global change in the error radius of $B_2(s,t)$. In this case the change in the error radius of $B_1(s,t)$ will result in a quantity that is no more than

$$\sum_{j+k=n} \binom{n}{i \ j \ k} (t+\delta)^j (1-s-t+\epsilon+\delta)^k \cdot \Delta\rho_0 = (1-s+\epsilon+2\delta)^n \cdot \Delta\rho_0 \quad (11.11)$$

added to the error radius of $B_2(s,t)$. As this extra error is transmitted to the control point $b_{n,0,0}^{(2)}$ ($s=1, t=0$) of $B_2(s,t)$, this will give rise to changes in the error radius at that point, which are no more than $(\epsilon+\delta)^n \cdot \Delta\rho_0$. It is easy to see that we can guarantee

$$(\epsilon+\delta)^n \cdot \Delta\rho_0 < \Delta\rho_0 \quad (11.12)$$

as long as $\epsilon+\delta < 1$ is valid.

As we can see from above, in the triangular ball Bézier patches, the perturbation generated from a piece of ball patch will be transmitted to each of the following pieces of patches along with the control points. This perturbation is attenuated, however. Therefore the construction of triangular ball Bézier patches is stable.

12. Conclusion

In this paper disk parametric curves, tensor ball surfaces, and triangular ball patches were considered from the point of including errors in representation. Algorithms for intersection testing were given for each case, and analyses for errors were considered. It was found that disk parametric curves and triangular ball patches were stable under bounded error perturbations, whereas tensor ball surfaces were not.

References

- [1] G. Alefeld and J. Herzberger, *Introduction to Interval Computations*, Computer Science and Applied Mathematics, Academic Press, New York; Harcourt Brace Jovanovich, London, 1983, translated by J. Rokne from Einführung in die Intervallrechnung. Bibliographisches Institut, Mannheim.
- [2] T. DeRose, M. Lounsbery, and R. Goldman, *A tutorial introduction to blossoming*, Geometric Modeling (Böblingen, 1990) (H. Hagen and D. Roller, eds.), Computer Graphics—Systems and Applications, Springer, Berlin, 1991, pp. 267–286.
- [3] G. Farin, *Curves and Surfaces for CAGD. A Practical Guide*, 5th ed., Morgan Kaufmann, California, 2002.
- [4] R. T. Farouki and V. T. Rajan, *On the numerical condition of polynomials in Bernstein form*, Computer Aided Geometric Design **4** (1987), no. 3, 191–216.
- [5] ———, *Algorithms for polynomials in Bernstein form*, Computer Aided Geometric Design **5** (1988), no. 1, 1–26.
- [6] H. Fischer, *Ein Versuch zur Verallgemeinerung der Intervallrechnung*, Interval Mathematics, Lecture Notes in Computer Science, vol. 29, Springer, Berlin, 1975, pp. 184–190.
- [7] P. K. Ghosh and S. P. Mudur, *The brush-trajectory approach to figure specification: some algebraic solutions*, ACM Transactions on Graphics **3** (1984), 110–134.
- [8] Q. Lin and J. G. Rokne, *Disk Bézier curves*, Computer Aided Geometric Design **15** (1998), no. 7, 721–737.
- [9] ———, *Fitting fat curves*, Mathematical and Computer Modelling **27** (1998), no. 6, 1–12.
- [10] ———, *Fitting parametric disk curves*, International Journal of Computer Mathematics **71** (1999), no. 2, 173–182.
- [11] R. E. Moore, *Interval Analysis*, Prentice-Hall, New Jersey, 1966.
- [12] K. Nickel, *Interval Mathematics*, Lecture Notes in Computer Science, vol. 29, Springer, Berlin, 1975.
- [13] K. C. Posch and K. C. Fellner, *The circle-brush algorithm*, ACM Transactions on Graphics **8** (1989), no. 1, 1–24.
- [14] L. Ramshaw, *A connect-the-dots approach to splines*, Tech. Rep. 19, Digital Systems Research Center, California, 1987.
- [15] ———, *Béziers and B-splines as multiaffine maps*, Theoretical Foundations of Computer Graphics and CAD (Il Ciocco, 1987) (R. A. Earnshaw, ed.), NATO Adv. Sci. Inst. Ser. F Comput. Systems Sci., vol. 40, Springer, Berlin, 1988, pp. 757–776.
- [16] H. Ratschek, *Nichtnumerische Aspekte der Intervallmathematik*, Interval Mathematics, Lecture Notes in Computer Science, vol. 29, Springer, Berlin, 1975, pp. 48–74.

- [17] H. Ratschek and J. G. Rokne, *Computer Methods for the Range of Functions*, Ellis Horwood Series: Mathematics and Its Applications, Ellis Horwood, Chichester, 1984.
- [18] Web Page, *Estimating machining accuracy*, <http://www.isr.umd.edu/Labs/CIM/vm/ama/node19.html>, 2004.
- [19] T. Whitted, *Anti-aliased line drawing using brush extrusion*, *Computer Graphics* **17** (1983), 151–156.
- [20] C. Yao and J. G. Rokne, *Fat curves*, *Computer Graphics Forum* **10** (1991), no. 3, 237–248.

Qun Lin: Department of Statistics, Xiamen University, Xiamen 361005, China
E-mail address: zl873@xmu.edu.cn

J. G. Rokne: Department of Computer Science, The University of Calgary,
2500 University Drive N. W., Calgary, Canada AB T2N-1N4
E-mail address: rokne@cpsc.ucalgary.ca



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

