

J. HOSCHEK

G. SEEMANN

Spherical splines

M2AN. Mathematical modelling and numerical analysis - Modélisation mathématique et analyse numérique, tome 26, n° 1 (1992), p. 1-22

http://www.numdam.org/item?id=M2AN_1992__26_1_1_0

© AFCET, 1992, tous droits réservés.

L'accès aux archives de la revue « M2AN. Mathematical modelling and numerical analysis - Modélisation mathématique et analyse numérique » implique l'accord avec les conditions générales d'utilisation (<http://www.numdam.org/conditions>). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM

Article numérisé dans le cadre du programme
Numérisation de documents anciens mathématiques
<http://www.numdam.org/>



SPHERICAL SPLINES

by J. HOSCHEK ⁽¹⁾, G. SEEMANN ⁽¹⁾

Abstract. — In the present paper we will introduce Bézier curves and Bézier spline curves on algebraic surfaces, especially on a sphere. First we interpolate or approximate spherical point sets by circular splines, then we develop Bézier splines of higher degree on a sphere and on other algebraic surfaces.

Keywords : Rational Bézier curves, interpolation with circles, approximation with circles, spherical Bézier curves, Bézier curves on quadrics.

Résumé. — Splines sphériques. Dans le présent article nous introduisons des courbes de Bézier et des splines de Bézier sur des surfaces algébriques, et plus particulièrement sur une sphère. D'abord nous interpolons ou approximons des ensembles de points sphériques par des splines circulaires, ensuite nous développons les splines de Bézier de degré élevé sur une sphère ou sur d'autres surfaces algébriques.

1. CIRCULAR SPHERICAL BÉZIER CURVES

In the Bézier technique a circular arc can be described by [1, 5, 6]

$$\mathbf{X}(t) = \frac{\mathbf{b}_0 B_0^2(t) + \mathbf{b}_1 \omega B_1^2(t) + \mathbf{b}_2 B_2^2(t)}{B_0^2(t) + \omega B_1^2(t) + B_2^2(t)} \quad (t \in [0, 1]), \quad (1a)$$

where the control points (Bézier points) \mathbf{b}_i must fulfil the circle condition $|\mathbf{b}_0 \mathbf{b}_1| = |\mathbf{b}_1 \mathbf{b}_2|$. $B_i^2(t)$ ($i = 0(1)2$) are the Bernstein polynomials, ω is the weight. We assume that $\mathbf{b}_0, \mathbf{b}_2$ lie on a unit sphere with center O , then the angle 2φ between $\mathbf{b}_0, \mathbf{b}_2$ can be determined by

$$2 \cos \varphi = |\mathbf{b}_0 + \mathbf{b}_2|. \quad (1b)$$

Because of the circle condition, \mathbf{b}_1 must lie on a plane bisecting the angle $(\mathbf{b}_0, \mathbf{b}_2)$ and perpendicular to the plane determined by $(O, \mathbf{b}_0, \mathbf{b}_2)$

⁽¹⁾ Technical University Darmstadt, Fachbereich Mathematik, Technische Hochschule, D-6100 Darmstadt.

Additionally \mathbf{b}_1 must lie in the tangent plane to the sphere at \mathbf{b}_0 and in the tangent plane to the sphere at \mathbf{b}_2 . Thus \mathbf{b}_1 can have the representation (\times denotes the vector product)

$$\mathbf{b}_1 = \frac{\mathbf{b}_0 + \mathbf{b}_2}{2 \cos^2 \varphi} + \lambda \frac{\mathbf{b}_0 \times \mathbf{b}_2}{\sin 2 \varphi}, \quad (1c)$$

where the parameter λ describes the distance of \mathbf{b}_1 from the plane determined by $(O, \mathbf{b}_0, \mathbf{b}_2)$, (see *fig. 1*). If $\lambda = 0$, (1) gives a parametric representation of a spherical great circle, for $\lambda \neq 0$ we get small circles on the sphere (see *fig. 1*). Additionally the weight ω must be adapted: if (1) fulfils the circle equation, we get

$$\omega = \frac{\sin \varphi}{\sqrt{\lambda^2 + \tan^2 \varphi}}. \quad (1d)$$

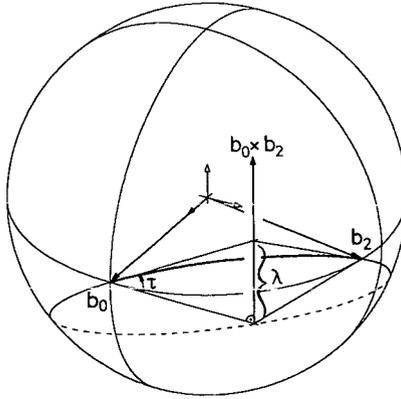


Figure 1. — Representation of a small circle on the sphere.

Instead of the parameter λ we can also introduce the angle τ between the plane $(O, \mathbf{b}_0, \mathbf{b}_2)$ and the Bézier polygon (see *fig. 1*) as measure for the deviation of a small circle from the great circle. For a great circle ($\lambda = 0!$) we have $|\mathbf{b}_0 \mathbf{b}_1| = \tan \varphi$ (see [6, 9]), thus we obtain from figure 1

$$\tan \tau \tan \varphi = \lambda \quad (1e)$$

and for the weight ω with (1d)

$$\omega = \cos \varphi \cos \tau. \quad (1f)$$

For $\tau = \frac{\pi}{2}$, the circular arc is a half circle (see *fig. 2*), while in this case $\omega = 0$ and the control point \mathbf{b}_1 moves to infinity. Therefore we have to split the parametric representation (1a) of the circular arc to (see [5, 8])

$$\mathbf{X}(t) = \frac{\mathbf{b}_0 B_0^2(t) + \mathbf{b}_2 B_2^2(t)}{B_0^2(t) + B_2^2(t)} + \frac{\vec{\mathbf{b}}_1 B_1^2(t)}{B_0^2(t) + B_2^2(t)}, \quad (2)$$

where $\vec{\mathbf{b}}_1$ is the direction which points to the position of \mathbf{b}_1 at infinity. From (1a, c, e) we get

$$\vec{\mathbf{b}}_1 = \frac{\mathbf{b}_0 \times \mathbf{b}_2}{2 \cos^2 \varphi} \sin \tau \quad \text{with (1c) for } \tau \rightarrow \frac{\pi}{2}. \quad (2a)$$

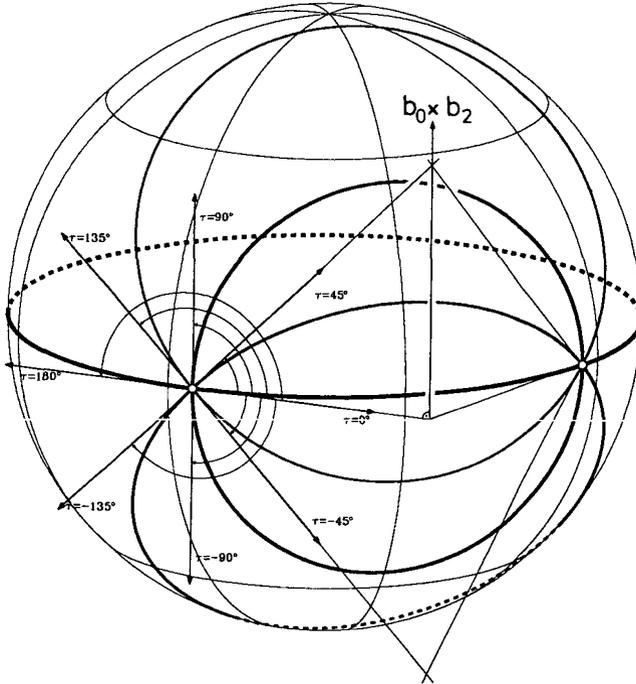


Figure 2. — Different values of parameter τ lead to small or great circles on the sphere.

For $\tau = \pi$ we get the complement arc of the circular arc which we would obtain for $\tau = 0$ (see *fig. 2*).

2. CIRCULAR SPHERICAL SPLINES

Now we will connect circular arcs to circular C^1 spherical spline curves. The i -th spline segment may have the representation

$$\mathbf{X}_i(t) = \frac{\mathbf{b}_{2_i} B_0^2(t) + \mathbf{b}_{2_{i+1}} \omega_i B_1^2(t) + \mathbf{b}_{2_{i+2}} B_2^2(t)}{B_0^2(t) + \omega_i B_1^2(t) + B_2^2(t)} \quad (i = 0(1)n - 1) \quad (3)$$

with $B_k^2(t)$ as Bernstein polynomials over the interval $t \in [0, \mu_i]$ and μ_i as length of the parameter interval of the segment \mathbf{X}_i (see [1, 5]). Two neighboring spline segments are C^1 -continuous if and only if the first derivatives in two corresponding boundary points are equal. Thus we get the condition

$$\omega_{i+1} \mu_{i+1} (\mathbf{b}_{2_{(i+1)+1}} - \mathbf{b}_{2_{(i+1)}}) = \omega_i \mu_{i+1} (\mathbf{b}_{2_{(i+1)}} - \mathbf{b}_{2_{i+1}}). \quad (4a)$$

If we insert (1c, 1e, 1f) this condition reduces to

$$\mu_i \sin \varphi_{i+1} = \mu_{i+1} \sin \varphi_i. \quad (4b)$$

Additionally from (4a) follows that three neighboring Bézier points $\mathbf{b}_{2_{i+1}}, \mathbf{b}_{2_{i+2}}, \mathbf{b}_{2_{(i+1)+1}}$ are collinear. As shown in figure 3, this collinearity can be expressed in terms of the angles τ_i, τ_{i+1} and δ_{i+1} , where δ_{i+1} describes the angle between the planes $(O, \mathbf{b}_{2_i}, \mathbf{b}_{2_{i+2}})$ and $(O, \mathbf{b}_{2_{(i+1)}}, \mathbf{b}_{2_{(i+2)}})$. Thus we have

$$\tau_{i+1} = -(\tau_i + \delta_{i+1}). \quad (4c)$$

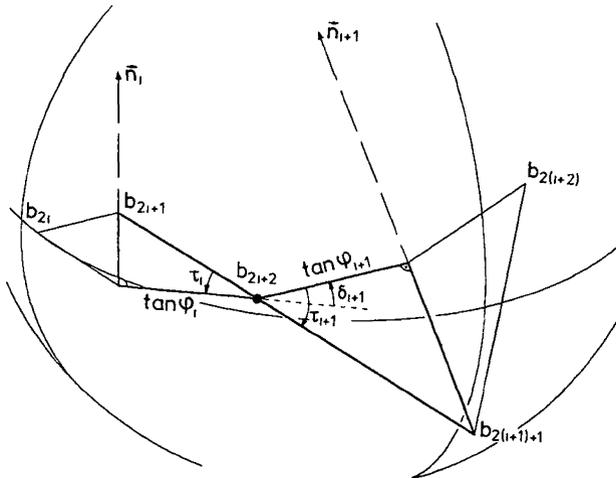


Figure 3. — C^1 -continuity and neighboring control points in the tangent plane at $\mathbf{b}_{2_{i+2}}$.

To evaluate the angle δ_i , we introduce the normal vectors \mathbf{n}_i to the planes through $(O, \mathbf{b}_{2i}, \mathbf{b}_{2(i+1)})$

$$\mathbf{n}_i = \frac{\mathbf{b}_{2i} \times \mathbf{b}_{2(i+1)}}{\sin 2 \varphi_i} . \tag{4d}$$

With help of these normal vectors we get

$$|\sin \delta_{i+1}| = |\mathbf{n}_i \times \mathbf{n}_{i+1}| = \left| \frac{\det (\mathbf{b}_{2i}, \mathbf{b}_{2(i+1)}, \mathbf{b}_{2(i+2)})}{\sin 2 \varphi_i \sin 2 \varphi_{i+1}} \right| , \tag{4e}$$

$$\cos \delta_{i+1} = \mathbf{n}_i \cdot \mathbf{n}_{i+1} = \frac{\cos 2 \varphi_i \cos 2 \varphi_{i+1} - \cos \Omega_i}{\sin 2 \varphi_i \sin 2 \varphi_{i+1}}$$

with Ω_i as angle between \mathbf{b}_{2i} and $\mathbf{b}_{2(i+2)}$. δ_{i+1} is negatively oriented, when the sign of the determinante in (4e) is negative.

With this result — which only fails for diametral points $\mathbf{b}_{2i}, \mathbf{b}_{2(i+1)}$ i.e. $\mathbf{b}_{2i} = -\mathbf{b}_{2(i+1)}$ (see [9]) — two neighboring C^1 spline segments are completely determined.

3. INTERPOLATION WITH SPHERICAL CIRCULAR SPLINES

Now we assume that on the unit sphere an (open) set of $(n + 1)$ points \mathbf{P}_i ($i = 0(1)n$) is given, which may have the spherical coordinates

$$\mathbf{P}_i = (\cos \alpha_i \cos \beta_i, \sin \alpha_i \cos \beta_i, \sin \beta_i) . \tag{5}$$

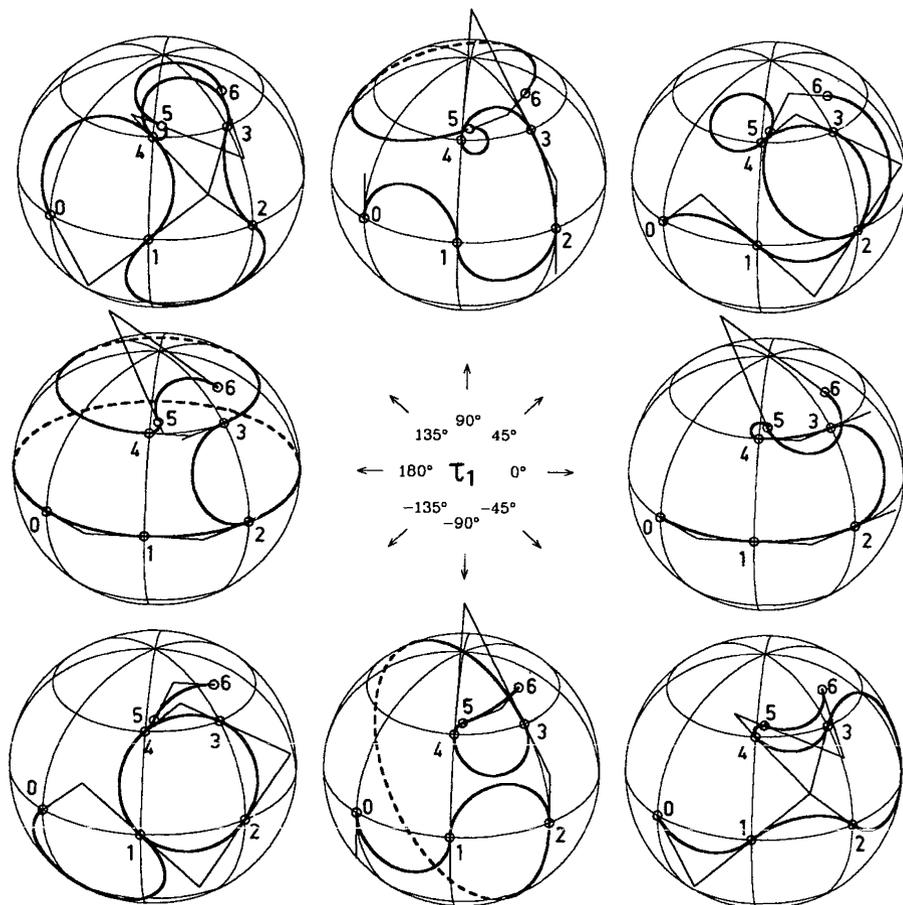
If now we identify these given points as Bézier points \mathbf{b}_{2i} , the interior Bézier points \mathbf{b}_{2i+1} are determined by (1b, c) and (4c, e) when the initial direction τ_1 is given. The interpolation spline curve depends on the choice of this initial direction (see fig. 4).

To reduce this degree of freedom, we can interpolate the given set of points in respect to some constraints. For instance, we can construct circular splines with the shortest arc length : if r_i is the radius of the i -th segment ($i = 1(1)n$) and γ_i its opening angle, we get the objective function

$$L = \sum_{i=1}^n r_i \gamma_i \rightarrow \min . \tag{5a}$$

or after calculation of r_i and γ_i out of (1)

$$L = \sum_{i=1}^n \frac{\sin \varphi_i}{\sqrt{1 - \omega^2}} \arccos \omega_i \rightarrow \min . \tag{5b}$$



P_i	α	β
0	0°	0°
1	45°	0°
2	90°	0°
3	90°	45°
4	45°	45°
5	50°	50°
6	105°	60°

τ_1	arc length
-135°	10.21413
-90°	9.61811
-45°	5.53946
0°	4.66909
45°	7.26881
90°	8.18605
135°	8.17716
180°	16.94670

Figure 4. — Open spherical circular splines through the same points and different starting tangent with direction τ_1 .

where $\omega_i = \omega_i(\tau_i)$ (see (1f)). (5b) is a nonlinear optimization problem which can be solved with help of wellknown optimization algorithms included in software packages like IMSL, NAC. Figure 5 contains the interpolation of the set of points out of figure 4 with circular splines with shortest arc length. The arc lengths of the curves in figure 4 and figure 5 are mentioned in the included table.

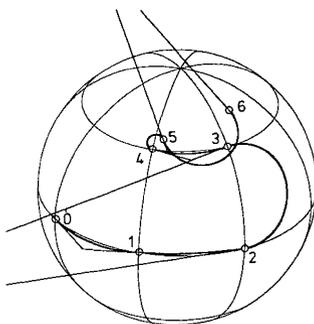


Figure 5. — Shortest circular spherical splines through the set of points out of figure 4 ($\tau_1 = -6.17300^\circ$, arc length = 4.64796).

Special problems arise if we interpolate closed point sets : we have to distinguish between odd and even sets of points P_i ($i = 0(1)n$) with $P_0 = P_n$. For a closed curve we must also have common tangents in $P_0 = P_n$, which lead to the condition

$$\tau_1 = -(\tau_n + \delta_{n+1}). \tag{6}$$

Similar to the interpolating problems with circular splines in the plane (see [4]) we get

- for n odd, there are two solutions of (6) (with the same initial tangent but with different orientation) ;
- for n even, the condition (6) cannot be held generally, therefore one of the given points must be moved to fulfil (6). If (6) is fulfilled randomly, a closed spline curve can be constructed for all initial tangents τ_1 .

In both cases the closed solutions to a given point set can be evaluated in an iterative way : in the odd case by changing τ_1 and in the even case by moving one of the given points (see *fig. 6b, c*). In figure 6a we have again chosen the set of points out of figures 4, 5. The set is odd, therefore a closed spline curve can be found without moving a point. Figure 6b contains an even set of points : the given point P_5 is marked by a square.

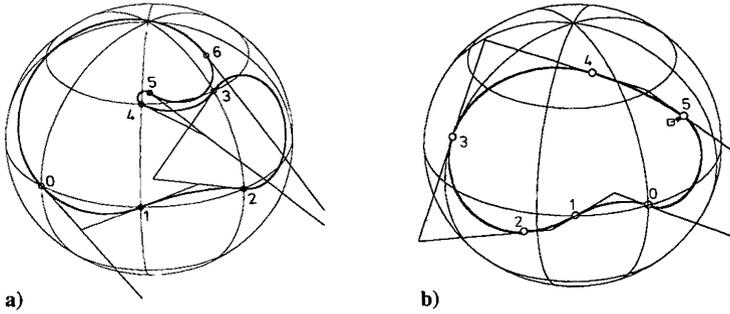


Figure 6a, b. — Closed spherical splines through the (odd) set of points out of figure 4 (fig. 6a); in figure 6b with an even set the given point P_5 is moved to the new position.

The motion of the point P_5 is not arbitrary — the suitable positions of P_5 fulfil a circle as shown in figure 6c.

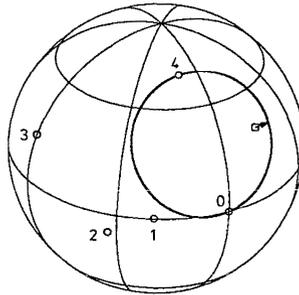


Figure 6c. — Suitable positions of P_5 for a closed circular spline curve through an even set of points.

4. APPROXIMATION WITH CIRCULAR SPLINES

Now we consider a given set of spherical points P_i ($i = 0(1)n$) which should be approximated by a spherical circular spline curve X with N segments and $N < n$. The given points P_i may have the (initial) parameter values t_i , which are found — for instance — about the chordal distances of the given points. Then we have to minimize the distance function

$$d = \sum_{i=0}^n (P_i - X(t_i))^2 \rightarrow \min . \tag{7}$$

For simplicity we assume that the first and last control points of the required

circular spline curve \mathbf{X} coincide with the first and last points of the given set, thus we have

$$\mathbf{b}_0 = \mathbf{P}_0, \quad \mathbf{b}_{2N} = \mathbf{P}_n. \tag{8}$$

The goal of the approximation process is to determine the remaining $(N - 1)$ control points \mathbf{b}_{2k} ($k = 1(1)N - 1$) and the initial value τ_1 . When each of the unknown control points \mathbf{b}_{2k} can be set up about (see (5))

$$\mathbf{b}_{2k} = (\cos u_k \cos v_k, \sin u_k \cos v_k, \sin v_k) \tag{9}$$

we have this result

THEOREM : *A set of $(n + 1)$ points \mathbf{P}_i ($i = 0(1)n$) on a unit sphere can be approximated by N spherical circular spline segments ($N < n$) with help of $2N - 1$ nonlinear unknowns τ_1, u_i, v_i ($i = 1(1)N - 1$).*

To obtain a low number of circular spline segments and an approximation curve with low approximation error e_0 the initial values

- I. of suitable positions of N circular segments,
- II. of the parametrization of the given points

are very important when using nonlinear optimization algorithms.

I. To determine an initial guess for the unknowns we subdivide the given set of points into possible segments so that the points approximately lie on circular arcs. As a spherical circle is an intersection curve of a sphere with an arbitrary plane, we have to calculate the positions of planes ε_k which approximate a part of the given set of points with respect to a given error estimate d_0 . We use three indices j_B, j_M and j_E with $0 \leq j_B < j_M < j_E \leq n$ and test the planes determined by $(P_{j_B}, P_{j_M}, P_{j_E})$ as follows :

We start with $j_B := 0$ and $k = 1$ and set for each further plane $j_E := n$ and $j_M := n - 1$. Now we determine the distances from all points $P_j, j_B < j < j_E$ from a plane ε_k and check whether the distances are less than d_0 . If this fails we first decrease j_M and test again, if the test fails too, we decrease j_E and set $j_M := j_E - 1$. We repeat these steps until the distance test is successful and then we mark the point P_{j_E} as end point of the k -th segment. Now we increase k and start again with $j_B := j_E + 1$. If k is larger than the number N of the required circular spline segments, we have to increase d_0 and repeat the whole algorithm (starting with $k = 1$). Figure 7 shows a set of spherical points approximated by two planes.

Now initial values for the unknowns u_i, v_i (with respect to our theorem) are determined and we can calculate τ_1 so that the first segment of the curve lies in the plane ε_1

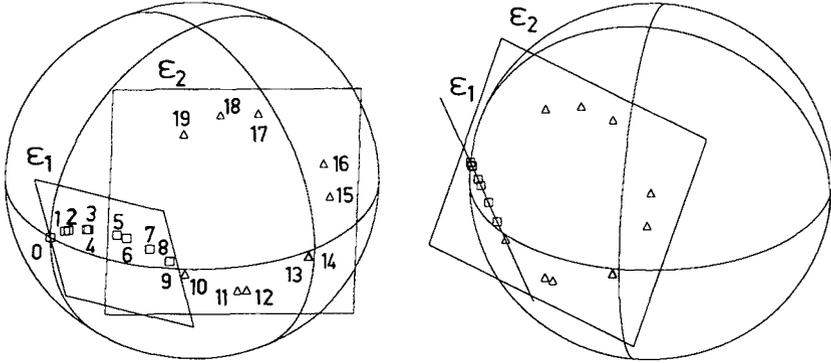


Figure 7. — Planes to determine an initial guess for the optimization algorithm.

II. Additionally we have to choose suitable parameter values t_i for the set of points \mathbf{P}_i . Testing uniform, chordal, centripetal and geometric parametrization, the chordal parametrization turned out to be best in most cases. Difficulties arise if the solution is supposed to have one or more segments spanning more than π — characterized by $w_i < 0$ — (see *fig. 9*). Increasing N may improve the result in some cases (compare *fig. 9b* with *10a*).

We suggest a parametrization relative to the initial guess (see *fig. 10c*): let \mathbf{B}_k denote the initial guess for $\mathbf{b}_{2,k}$ and $\mathbf{B}_0 := \mathbf{P}_0$, $\mathbf{B}_n := \mathbf{P}_n$

$$\Delta_k := \frac{|\mathbf{B}_k - \mathbf{B}_{k-1}|}{N \sum_{j=1} |\mathbf{B}_j - \mathbf{B}_{j-1}|} \cdot s \quad \text{for parameter } t \text{ in } [0, s].$$

Assuming j_k being indices so that $\mathbf{P}_{j_k} = \mathbf{B}_k$, we get the parameter values t_j by

$$t_j - t_{j-1} = \frac{|\mathbf{P}_j - \mathbf{P}_{j-1}|}{\sum_{i=j_{k-1}+1}^{j_k} |\mathbf{P}_i - \mathbf{P}_{i-1}|} \cdot \Delta_k \quad \text{for } j_{k-1} < j \leq j_k. \quad (10)$$

Instead of dividing up the Δ_k according to chord lengths, we could use the shortest distance to the curve of the initial guess, but this works only if the initial guess is a relatively good approximation.

After determining these initial values, we start the nonlinear approximation process to minimize the total error sum (7). If the distances $e_i = |\mathbf{P}_i - \mathbf{X}(t_i)|$ are less than the required maximal error e_0 , the procedure stops — otherwise we introduce a parameter correction ([3], [4]) and start

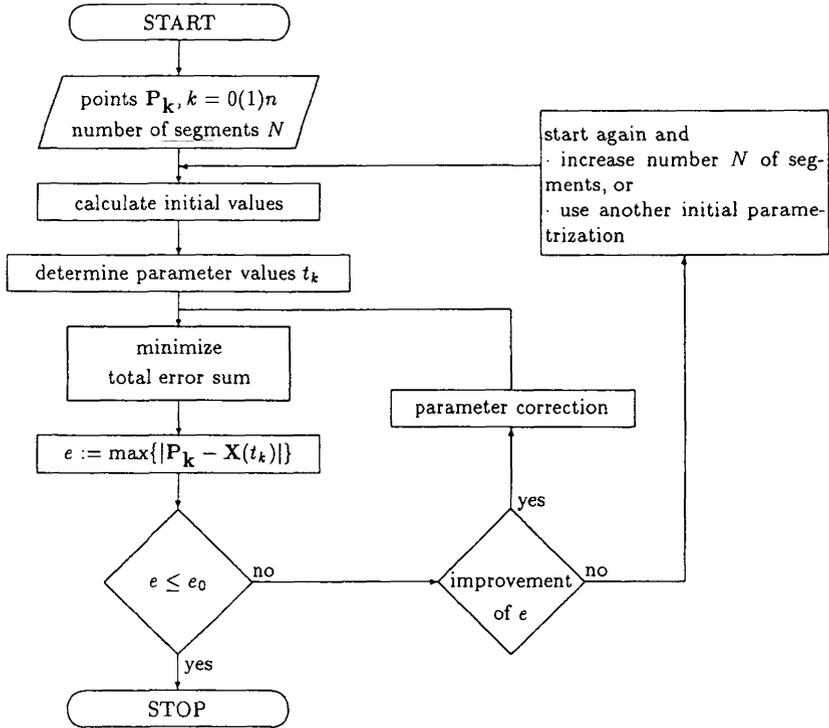


Figure 8. — Algorithm for approximation with spherical splines.

the approximation process again. If the distances e_i do not move below the maximal error e_0 , we can elevate the number N of segments (for instance by subdividing the segment with the largest error e_i at the point with this largest error) or by changing the initial parametrization. The whole approximation process follows the algorithm represented in figure 8.

In the following figures we give some examples about the approximation of spherical set of points. In the figures 9, 10 we demonstrate the effect of the choice of parametrization. First we try to approximate the given set of points with two circular segments and choose chordal parametrization. Figure 9a shows the error vectors for the initial parametrization, figure 9b shows the approximation curve. One can observe, that the approximation process is not successful. Therefore we have to change our strategy : we can elevate the number of splines and obtain the approximation curve in figure 10a. — If we do not want to elevate the number of splines we could change the initial parametrization and hope that other initial values lead to a better result. As shown in figure 10c this strategy is successful if we use the

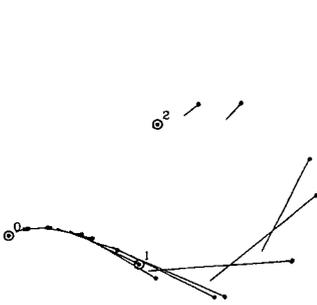


Figure 9a.

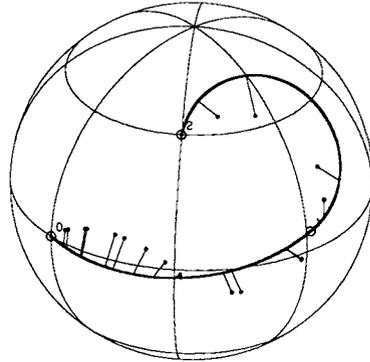


Figure 9b.

Figure 9a. — Error vectors to the initial values for chordal parametrization for two segments.

Figure 9b. — Approximation with the parametrization out of figure 9a.

relative parameter (10) instead of the chordal parametrization. Figure 10b contains the first approximation of the given set of points without parameter correction (observe the effect of the parameter correction!).

Until now we have only approximated open sets of points. Now we will extend our method to a closed set of points and their approximation with closed C^1 circular spherical splines. Similar to the interpolation process we have to distinguish between an odd and an even number N of segments (see chapter 3). For N odd, we can set up the unknown control points

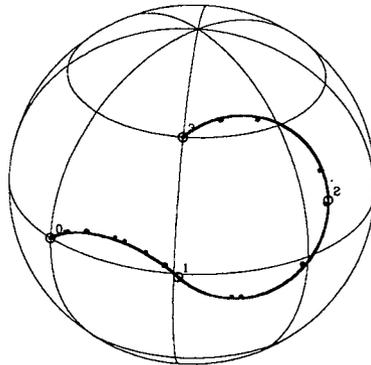


Figure 10a. — Approximation of the set of points out of figure 9 with chordal parametrization but with three circular segments (initial approximation curve).

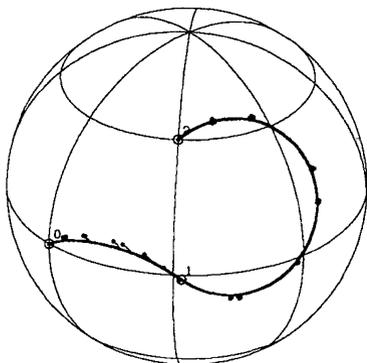


Figure 10b.

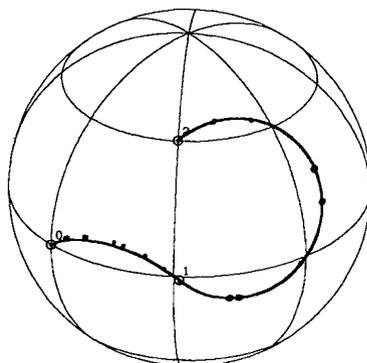


Figure 10c.

Figure 10b. — Approximation of the set of points out of figure 10a with parametrization relative to the initial value (10) and two segments.

Figure 10c. — Approximation and parameter correction of the set of points out of figure 10b, please observe the effect of the parameter correction.

b_{2k} analogously to the open splines (see (9)) and calculate τ_1 with the condition (6) to

$$\tau_1 = \sum_{i=2}^{N+1} (-1)^i \frac{\delta_i}{2}.$$

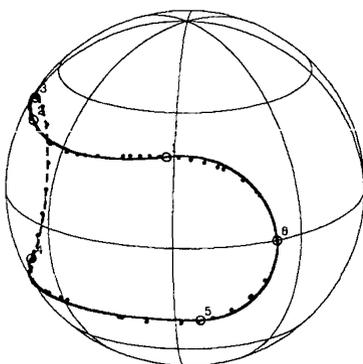


Figure 11a.

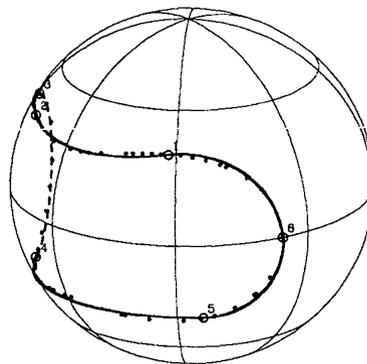


Figure 11b.

Figure 11a. — Approximation of a closed set of points by C^1 spherical splines with 6 segments without parameter correction.

Figure 11b. — Approximation of a set of points out of figure 11a after parameter correction.

For N even, we set up \mathbf{b}_{2k} , $k = 1(1)N - 2$ analogously to the open case, but the point $\mathbf{b}_{2(N-1)}$ is determined by the other points: the Bézier point $\mathbf{b}_{2(N-1)}$ must lie on a circle through the points \mathbf{b}_0 and $\mathbf{b}_{2(N-2)}$. Let $\mathbf{Y}(t)$ be the Bézier representation of the arc from \mathbf{b}_0 to $\mathbf{b}_{2(N-2)}$ and $\bar{\mathbf{Y}}(t)$ the representation of the complementary arc, then we introduce the opening angle Θ to denote the position of $\mathbf{b}_{2(N-1)}$ on this circle:

$$\mathbf{b}_{2(N-1)} = \begin{cases} \mathbf{Y}(\cos \Theta) & \text{if } \cos \Theta \geq 0 \\ \bar{\mathbf{Y}}(-\cos \Theta) & \text{if } \cos \Theta < 0. \end{cases}$$

Thus we get

PROPOSITION: *A closed set of points \mathbf{P}_i ($i = 0(1)n$), with $\mathbf{P}_0 = \mathbf{P}_n$ on a unit sphere can be approximated by N spherical closed C^1 circular spline curves ($N < n$) with help of $2N - 2$ nonlinear unknowns.*

Figure 11 contains the approximation of a set of points lying on the intersection curve of the sphere and a cylinder by closed circular splines.

5. NONCIRCULAR SPHERICAL BÉZIER SPLINES

Now we will construct noncircular spherical Bézier curves and Bézier splines. That means that we are looking for curves represented by rational Bernstein basis functions which are completely lying on a sphere, even if the Bézier control points are in part not on the sphere. The rational spherical curve \mathbf{X} may have the representation

$$\mathbf{X}(t) = \left(\frac{x_1(t)}{x_0(t)}, \frac{x_2(t)}{x_0(t)}, \frac{x_3(t)}{x_0(t)} \right) \quad (11a)$$

or in Bernstein Bézier description of degree n

$$\mathbf{X}(t) = \frac{\sum_{j=0}^n \mathbf{b}_j \beta_j B_j^n(t)}{\sum_{j=0}^n \beta_j B_j^n(t)} \quad (11b)$$

with the Bézier control points \mathbf{b}_j and the weights β_j ($n = 2$, see (1a)).

The components of (11a) must fulfil the equation of a (unit) sphere

$$x_1^2(t) + x_2^2(t) + x_3^2(t) = x_0^2(t). \quad (11c)$$

Therefore the x_i are « pythagorean » functions [2, 7]. If we extend the result of Kubota [7] to the \mathbb{R}^3 , we get a necessary and sufficient representation of spherical rational curves

$$\begin{aligned}
 x_1(t) &= s(t)(u^2(t) - v^2(t) - w^2(t)) \\
 x_2(t) &= 2 s(t) u(t) v(t) \\
 x_3(t) &= 2 s(t) u(t) w(t) \\
 x_0(t) &= s(t) (u^2(t) + v^2(t) + w^2(t))
 \end{aligned}
 \tag{12}$$

with arbitrary functions s, u, v, w .

To get Bézier curves, we have to choose these functions as Bernstein representations and without loss of generality we choose $s(t) = 1$. Thus for quadratic spherical curves (circles), we can set up for instance

$$\begin{aligned}
 u(t) &= u_0 B_0^1(t) + u_1 B_1^1(t), \\
 v(t) &= v_0 B_0^1(t) + v_1 B_1^1(t), \\
 w(t) &= w_0 B_0^1(t) + w_1 B_1^1(t)
 \end{aligned}
 \tag{13a}$$

with

$$\begin{aligned}
 u_0 &= \lambda_0 \cos \varphi_0, & u_1 &= \lambda_1 \cos \varphi_1, \\
 v_0 &= \lambda_0 \sin \varphi_0 \cos \psi_0, & v_1 &= \lambda_1 \sin \varphi_1 \cos \psi_1, \\
 w_0 &= \lambda_0 \sin \varphi_0 \sin \psi_0, & w_1 &= \lambda_1 \sin \varphi_1 \sin \psi_1.
 \end{aligned}
 \tag{13b}$$

If we insert (13) in (12), we additionally need the following *transformation table* for Bernstein polynomials

$$\begin{aligned}
 (B_0^1(t))^2 &= B_0^2(t), & B_0^1(t) B_1^1(t) &= \frac{1}{2} B_1^2(t). \\
 (B_1^1(t))^2 &= B_2^2(t),
 \end{aligned}$$

From (12) and (13) we get parametric representations of quadratic Bézier curves which lie completely on the sphere and therefore these curves are circles (see chapter 2).

Now we choose nonlinear Bézier representations, for instance

$$\begin{aligned}
 u(t) &= u_0 B_0^2(t) + u_1 B_1^2(t) + u_2 B_2^2(t) \\
 v(t) &= v_0 B_0^2(t) + v_1 B_1^2(t) + v_2 B_2^2(t) \\
 w(t) &= w_0 B_0^2(t) + w_1 B_1^2(t) + w_2 B_2^2(t)
 \end{aligned}
 \tag{14a}$$

and have the transformation table

$$\begin{aligned}
 (B_0^2(t))^2 &= B_0^4(t), & B_0^2(t) B_1^2(t) &= \frac{1}{2} B_1^4(t) \\
 B_0^2(t) B_2^2(t) &= \frac{1}{6} B_2^4(t), & B_1^2(t) B_2^2(t) &= \frac{1}{2} B_3^4(t) \\
 (B_1^2(t))^2 &= \frac{2}{3} B_2^4(t), & (B_2^2(t))^2 &= B_4^4(t).
 \end{aligned}
 \tag{14b}$$

Additionally we choose as constants u_i, v_i, w_i

$$\begin{aligned} u_0 &= \lambda_0 \cos \varphi_0, & u_1 &= \lambda_1 \cos \varphi_1, & u_2 &= \lambda_2 \cos \varphi_2 \\ v_0 &= \lambda_0 \sin \varphi_0 \cos \psi_0, & v_1 &= \lambda_1 \sin \varphi_1 \cos \psi_1, & v_2 &= \lambda_2 \sin \varphi_2 \cos \psi_2 \\ w_0 &= \lambda_0 \sin \varphi_0 \sin \psi_0, & w_1 &= \lambda_1 \sin \varphi_1 \sin \psi_1, & w_2 &= \lambda_2 \sin \varphi_2 \sin \psi_2. \end{aligned} \quad (14c)$$

If we insert these assumptions in (12) or the Bézier representation (11b), we get the explicit representation of a quartic spherical Bézier curve. The Bézier control points are

$$\begin{aligned} \mathbf{b}_0 &= \begin{pmatrix} \cos 2 \varphi_0 \\ \sin 2 \varphi_0 \cos \psi_0 \\ \sin 2 \varphi_0 \sin \psi_0 \end{pmatrix} \\ \mathbf{b}_1 &= \begin{pmatrix} \frac{u_0 u_1 - v_0 v_1 - w_0 w_1}{N_1} \\ \frac{u_0 v_1 + u_1 v_0}{N_1} \\ \frac{u_0 w_1 + u_1 w_0}{N_1} \end{pmatrix} \end{aligned} \quad (14d)$$

with $N_1 = u_0 u_1 + v_0 v_1 + w_0 w_1$

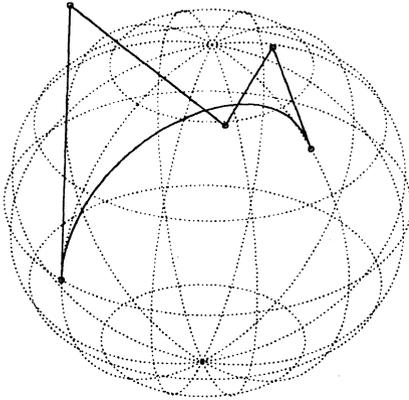
$$\mathbf{b}_2 = \begin{pmatrix} \frac{u_0 u_2 - v_0 v_2 - w_0 w_2 + 4 u_1^2 - 2 \lambda_1^2}{N_2} \\ \frac{u_0 v_2 + 4 u_1 v_1 + u_2 v_0}{N_2} \\ \frac{u_0 w_2 + 4 u_1 w_1 + u_2 w_0}{N_2} \end{pmatrix}$$

with $N_2 = u_0 u_2 + v_0 v_2 + w_0 w_2 + 2 \lambda_1^2$

$$\begin{aligned} \mathbf{b}_3 &= \mathbf{b}_1([u_i, v_i, w_i] \text{ exchange by } [u_{i+1}, v_{i+1}, w_{i+1}] \quad (i = 0, 1)) \\ \mathbf{b}_4 &= \mathbf{b}_0(\varphi_0 \text{ exchange by } \varphi_2, \psi_0 \text{ exchange by } \psi_2). \end{aligned} \quad (14e)$$

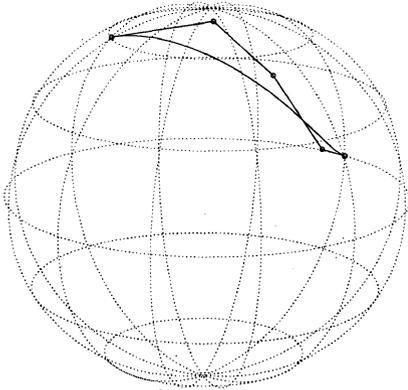
As weights we have $\beta_0 = \lambda_0^2$, $\beta_1 = N_1$, $\beta_2 = N_2$. The weights β_3, β_4 follow from (14e).

These calculations can be easily done with help of a formula manipulator. Figure 12 shows two examples of quartic spherical Bézier curves and the corresponding control polygon.



	0	1	2
φ	0	60°	45°
ψ	60°	90°	45°
λ	1	0.6	1

Fig. 12a.



	0	1	2
φ	0	9°	45°
ψ	60°	0	45°
λ	1	0.6	1

Fig. 12b.

Figure 12a, b. — A convex and a nonconvex quartic spherical Bézier curve and the corresponding Bézier polygon.

If we develop sextic spherical Bézier curves, in the assumption (14a) we have to introduce cubic Bernstein polynomials and constants u_i, v_i, w_i with $i = 0(1) 3$. Analogously to (14b) we get the transformation table

$$\begin{aligned}
 B_0^3 B_0^3 &= B_0^6, & B_0^3 B_1^3 &= \frac{1}{2} B_1^6, & B_0^3 B_2^3 &= \frac{1}{5} B_2^6, & B_0^3 B_3^3 &= \frac{1}{20} B_3^6; \\
 B_1^3 B_1^3 &= \frac{3}{5} B_2^6, & B_1^3 B_2^3 &= \frac{9}{20} B_3^6, & B_1^3 B_3^3 &= \frac{1}{5} B_4^6; \\
 B_2^3 B_2^3 &= \frac{3}{5} B_4^6, & B_2^3 B_3^3 &= \frac{1}{2} B_5^6, & B_3^3 B_3^3 &= B_6^6.
 \end{aligned}
 \tag{15a}$$

In extension to (14c) we have to introduce λ_3 , φ_3 , ψ_3 to describe u_3 , v_3 , w_3 analogously to (14c). With help of corresponding calculations we get the Bézier points

\mathbf{b}_0 and \mathbf{b}_1 in the same representation as in (14d)

$$\mathbf{b}_2 = \left(\begin{array}{c} \frac{2(u_0 u_2 - v_0 v_2 - w_0 w_2) + 6 u_1 - 3 \lambda_1}{N_2} \\ \frac{2(u_0 v_2 + 3 u_1 v_1 + u_2 v_0)}{N_2} \\ \frac{2(u_0 w_2 + 3 u_1 w_1 + u_2 w_0)}{N_2} \end{array} \right)$$

$$\mathbf{b}_3 = \left(\begin{array}{c} \frac{u_0 u_3 - v_0 v_3 - w_0 w_3 + 9(u_1 u_2 - v_1 v_2 - w_1 w_2)}{N_3} \\ \frac{u_0 v_3 + v_0 u_3 + 9(u_1 v_2 + u_2 v_1)}{N_3} \\ \frac{u_0 w_3 + w_0 u_3 + 9(u_1 w_2 + u_2 w_1)}{N_3} \end{array} \right)$$

with

$$N_2 = 2(u_0 u_2 + v_0 v_2 + w_0 w_2) + 3 \lambda_1^2 \quad \text{and}$$

$$N_3 = u_0 u_3 + v_0 v_3 + w_0 w_3 + 9(u_1 u_2 + v_1 v_2 + w_1 w_2)$$

$\mathbf{b}_4 = \mathbf{b}_2$ (exchange $[u_i, v_i, w_i]$ by $[u_{i+1}, v_{i+1}, w_{i+1}]$) with $i = 0(1)2$,
 $\mathbf{b}_5 = \mathbf{b}_1$ (exchange $[u_i, v_i, w_i]$ by $[u_{i+2}, v_{i+2}, w_{i+2}]$) with $i = 0(1)2$, (15c)
 $\mathbf{b}_6 = \mathbf{b}_0$ (exchange φ_0 by φ_3 , exchange ψ_0 by ψ_3).

The denominators N_k of (15b), (15c) lead to the weights β_k .

Two examples of spherical Bézier curves of degree 6 are represented in figure 13.

Now we will construct G^1 -continuous quartic spherical spline curves. With (11b) we denote the i -th segment by

$$\mathbf{X}_i(t) = \frac{\sum_{j=0}^4 \mathbf{b}_j \beta_j B_j^4(t)}{\sum_{j=0}^4 \beta_j B_j^4(t)}. \quad (16)$$

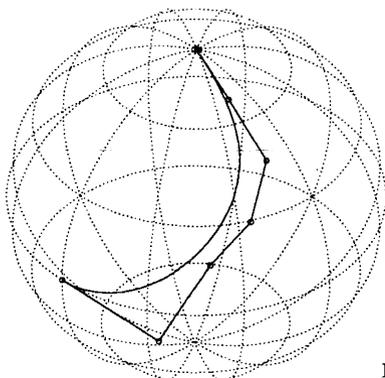


Fig. 13a.

	0	1	2	3
φ	0	36°	45°	45°
ψ	60°	0	45°	90°
λ	1	0.6	1	1

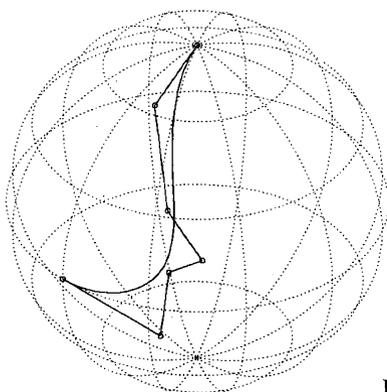


Fig. 13b.

	0	1	2	3
φ	0	36°	22.5°	45°
ψ	60°	0	45°	90°
λ	1	0.6	1	1

Figure 13a, b. — A convex and a nonconvex spherical Bézier curve of degree 6 and the corresponding Bézier polygon.

Additionally the angles in the set up (14c) are denoted by φ_{ki} , ψ_{ki} ($k = 0, 1, 2$). For C^0 -continuity we first obtain

$$\mathbf{b}_{4,i-1} = \mathbf{b}_{0,i} = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}. \tag{17a}$$

If we insert the Bézier points (14d) and (14e) in (17a), we get

$$\begin{aligned} \varphi_{2,i-1} &= \varphi_{0i}, \\ \psi_{2,i-1} &= \psi_{0i}. \end{aligned} \tag{17b}$$

Two Bézier curves are G^1 -continuous iff their common boundary points and the corresponding first neighboring points of the two segments are

linearly dependent, thus we have the wellknown condition (see (4a))

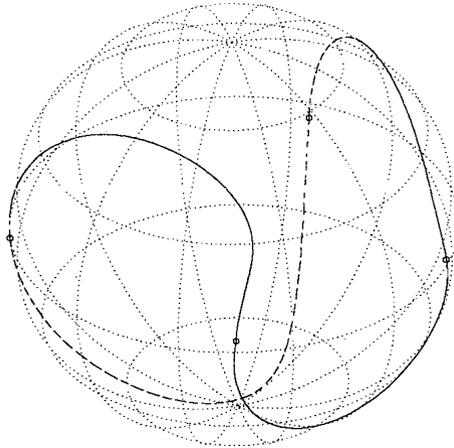
$$(\mathbf{b}_{i-1,4} - \mathbf{b}_{i-1,3}) = \mu (\mathbf{b}_{i1} - \mathbf{b}_{i0}). \tag{17c}$$

We can eliminate the parameter μ suitably and get as sufficient G^1 -continuity condition (with the abbreviations (17a) for the components of the common boundary point)

$$\begin{aligned} \tan \varphi_{1,i-1} &= \frac{1 - x_i}{\sin \psi_{1,i-1} z_i + \cos \psi_{1,i-1} y_i}, \\ \tan \varphi_{1,i} &= \frac{1 - x_i}{\sin \psi_{1,i} z_i + \cos \psi_{1,i} y_i}. \end{aligned} \tag{17d}$$

If we only construct a spline curve with two segments the angles $\psi_{1,i}$ and $\psi_{1,i-1}$ in (17d) can be chosen freely. If more than two segments are required, one has to observe additional compatibility conditions which follow from (17d) by elevating the index i . From the first equation in (17d), we get

$$\tan \varphi_{1,i} = \frac{1 - x_{i+1}}{\sin \psi_{1,i} z_{i+1} + \cos \psi_{1,i} y_{i+1}}. \tag{17e}$$



i	1	2	3	4
φ_{0i}	22.5°	67.5°	-77.5°	-32.5°
φ_{1i}	1.485	-1.546	-1.526	-1.407
φ_{2i}	67.5°	-77.5°	-32.5°	22.5°

i	1	2	3	4
ψ_{0i}	-5°	5°	-5°	5°
ψ_{1i}	1.448	1.597	-1.455	1.552
ψ_{2i}	5°	-5°	5°	-5°

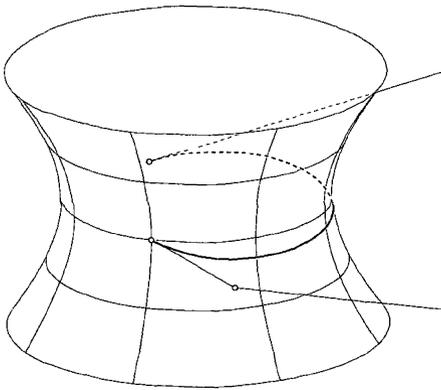
$$\lambda_{ik} = 1(i = 0(1) 2, k = 1(1) 4)$$

Figure 14. — Closed spherical quartic Bézier curve.

(17e) is equivalent to the second condition in (17d), thus we have a joining condition for the angle $\psi_{1,i}$

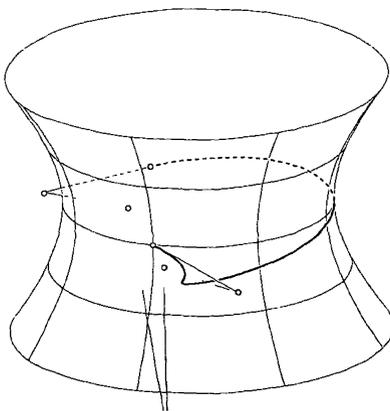
$$\tan \psi_{1,i} = - \frac{y_{i+1}(x_i - 1) - y_i(x_{i+1} - 1)}{z_{i+1}(x_i - 1) - z_i(x_{i+1} - 1)}. \quad (17f)$$

Additionally the first and the last points of the spline curve coincide for closed spline curves. Figure 14 contains a closed quartic spherical Bézier spline curve with four segments. The boundary points of the segments are denoted by circles.



	0	1	2
φ	0°	30°	110°
ψ	0	-0.50	0.10
λ	1	0.60	1.20

Figure 15a. — Bézier curve of degree 4 on a hyperboloid.



	0	1	2	3
φ	0°	30°	-30°	110°
ψ	0	-0.50	0.20	0.10
λ	1	0.60	1	1.20

Fig. 15b. — Bézier curve of degree 6 on a hyperboloid.

6. GENERALIZATIONS

The introduced method for constructing spherical Bézier curves can easily be extended on other quadratic surfaces : for instance for Bézier curves on an ellipsoid only the axes a , b , c must be inserted in the representation (12). For a hyperboloid

$$\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} - \frac{x_3^2}{c^2} = x_0^2$$

instead of (12) we can set up

$$\begin{aligned} x_1(t) &= as(t)(u^2(t) - v^2(t) + w^2(t)) \\ x_2(t) &= 2bs(t)u(t)v(t) \\ x_3(t) &= 2cs(t)u(t)w(t) \\ x_0(t) &= s(t)(u^2(t) + v^2(t) - w^2(t)). \end{aligned} \tag{18}$$

Figure 15 contains a Bézier curve of degree 4 and a Bézier curve of degree 6 on a hyperboloid and the corresponding control polygon.

REFERENCES

- [1] G. FARIN, Curves and surfaces for Computer Aided Geometric Design (sec. ed.) Academic Press, 1990.
- [2] R. D. FAROUKI, T. SAKKALIS, Pythagorean hodographs. Submitted to IBM, *J. Res. Develop.*, 1990.
- [3] J. HOSCHEK, Intrinsic parametrization for approximation. *Comput. Aided Geom. Design* 5 (1988), 27-31.
- [4] J. HOSCHEK, F.-J. SCHNEIDER, P. WASSUM, Optimal approximate conversion of spline surfaces. *Comput. Aided Geom. Design* 6 (1989), 293-306.
- [5] J. HOSCHEK, D. LASSER, Grundlagen der geometrischen Datenverarbeitung. Teubner, 1989.
- [6] J. HOSCHEK, Circular Splines. Submitted to Computer-aided design, 1990.
- [7] K. K. KUBOTA, Pythagorean triples in unique factorization domains. *Amer. Math. Monthly* 79 (1972), 503-505.
- [8] L. PIEGL, On the use of infinite control points in CAGD. *Comput. Aided Geom. Design* 4 (1987), 155-166.
- [9] G. SEEMANN, Interpolation und Approximation mit sphärischen Kreissplines in Bézier-Darstellung. Dipl.-Arbeit Fachbereich Mathematik, Technische Hochschule Darmstadt 1990.