Algorithms on cone spline surfaces and spatial osculating arc splines

Stefan Leopoldseder

Technische Universität Wien, Institut für Geometrie,
Wiedner Hauptstrasse 8-10, A-1040 Wien, Austria

Developable surfaces are of considerable importance to many industry applications, e.g. sheet metal forming processes. The objective of this paper is to provide algorithms on the approximation of developable surfaces with pieces of right circular cones. Special emphasis is devoted to practical choices of free parameters and to error estimation. Furthermore, a new algorithm for the approximation of spatial curves with a circular arc spline is presented which stands in close relation to above algorithms on developable surfaces. The proposed arc spline has contact of order 2 to the given curve in a series of curve points. The investigation includes a segmentation algorithm and error estimation.

Keywords: developable surface, spatial arc spline, circular spline

1   Introduction

Surfaces which are composed of (general) cylinders, (general) cones, and tangent surfaces of space curves are called *developable surfaces*. These are the only surfaces that can be unfolded (developed) into a plane without stretching or tearing. Therefore they are of particular interest in the modeling of surfaces of approximately unstretchable materials, such as paper, leather or thin sheets of metal (e.g. Mancewicz and Frey, 1992). Even in the design and engineering of *double curved* ship surfaces developable surfaces appear in an important intermediate stage of the manufacturing process (Randrup, 1996).

The first part of the present paper — which is based on the author's PhD thesis (Leopoldseder, 1998) — deals with the approximation of a given developable surface with a *cone spline surface*, that is a G^1-surface composed of segments of right circular cones. Subsequent cone segments possess the same tangent plane along their common ruling. Cone spline surfaces possess parametric and implicit representations of degree two, and their development and bending into other developable shapes is elementary.

Local algorithms for this problem have first been introduced by Leopoldseder and Pottmann (1998). There a Hermite-like interpolation scheme is presented:

Choose a set of rulings of a developable surface, together with the tangent planes along these rulings. Then, it is possible to interpolate each pair of consecutive rulings plus tangent planes with a pair of smoothly joined right circular cone segments. In this procedure there is still one free parameter which was proposed to be eliminated by minimizing the distance of the vertices (apices) of the cone pair.

Here, the paper mentioned above is extended by some error analysis. The used methods are based on differential geometry of developable surfaces and the Taylor expansion of the curve of regression (cuspoidal curve). A proof of the important fact will be given that the cone pairs interpolating two rulings plus tangent planes of a given developable surface converge nicely as the two rulings are moved together. Furthermore we suggest an alternative to the choice of the free parameter which will provide certain advantages in the implementation.

The second part of this paper — beginning with section 6 — deals with the approximation of space curves with circular arc splines. Its close relationship to developable surfaces and cone spline surfaces will be outlined in section 6 and is dealt with in greater detail in a seperate paper (Leopoldseder, 2000). There, sphere geometry is used to obtain some important results on cone spline surface algorithms.

Planar arc splines are desirable paths for numerically controlled cutting machines as their offsets are easy to find. Therefore there is a rich variety of literature on arc spline approximations of planar curves, mostly with biarc methods (e.g. Bolton, 1975; Nutbourne and Martin, 1988; Piegl, 1986; Sabin, 1977). The approximation of twisted curves in 3-space with *spatial biarcs* has been analyzed with geometric and analytical methods (Fuhs and Stachel, 1988; Hoschek, 1992; Nutbourne and Martin, 1988; Sharrock, 1986).

Here, a geometric method of constructing a *spatial osculating arc spline* is introduced: First, compute the osculating circles of a set of discrete curve points. Then, subsequent osculating circles are joined tangent continuously. Unlike the planar case where there exists a one parameter set of solution arcs joining two given circles (Meek and Walton, 1996), there are, in general, only two complex solution arcs in the 3-dimensional case. A segmentation algorithm of the given curve is presented that is based on geometric properties of the target curve. It guarantees real solutions for the joining arcs and approximation errors within a given tolerance. The practicality of the proposed methods is discussed for several examples.

2   Fundamentals on cone spline surfaces

Let us first sketch the 'cone pair algorithm' presented in Leopoldseder and Pottmann (1998). The following short summary will provide the neccessary background for the error estimates of section 4.

The basic idea is to choose a sequence of rulings e_i of a developable surface Γ, compute their tangent planes τ_i and interpolate consecutive G^1-Hermite elements (e_i, τ_i) with a *cone pair*, i.e. two segments of cones of revolution (right circular cones) that have the same tangent plane along their common ruling. Note that throughout this paper we will tacitly allow that a cone may degenerate to a cylinder.
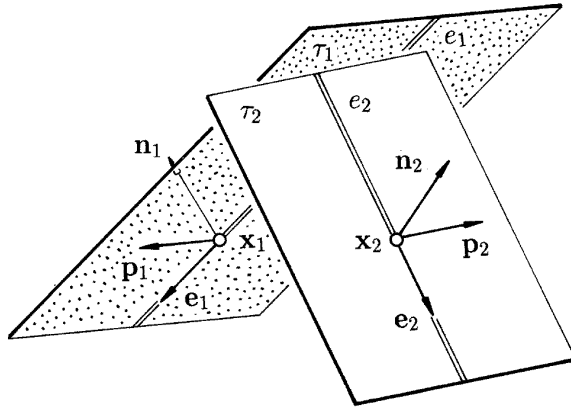

Figure 1: G^1-Hermite data

Figure 1 gives two Hermite elements (e_1, τ_1), (e_2, τ_2) which are interpolated with a cone pair in Figure 2. The spatial position and orientation of (e_i, τ_i) is determined by an associated orthonormal coordinate frame $(\mathbf{x}_i, \mathbf{e}_i, \mathbf{p}_i, \mathbf{n}_i)$. $\mathbf{x}_i$ denotes a point on the ruling e_i with unit direction vector $\mathbf{e}_i$. The vector $\mathbf{p}_i$ is orthogonal to $\mathbf{e}_i$ in τ_i and indicates the direction in which the interpolant between e_i and e_{i+1} has to connect. The unit normal $\mathbf{n}_i = \mathbf{e}_i \times \mathbf{p}_i$ of τ_i then always points to the same side of the surface Γ.

Two elements (e_1, τ_1), (e_2, τ_2) are in general position if $\det(\mathbf{e}_1, \mathbf{e}_2, \mathbf{x}_1 - \mathbf{x}_2) \neq 0$ and $\det(\mathbf{p}_1, \mathbf{p}_2, \mathbf{n}_1 - \mathbf{n}_2) \neq 0$. We will not treat the special cases $\mathbf{e}_1 = \mathbf{e}_2$ (cylindrical case), $\det(\mathbf{e}_1, \mathbf{e}_2, \mathbf{x}_1 - \mathbf{x}_2) = 0, \mathbf{e}_1 \neq \mathbf{e}_2$ (conical case), and $\det(\mathbf{p}_1, \mathbf{p}_2, \mathbf{n}_1 - \mathbf{n}_2) = 0, \mathbf{e}_1 \neq \mathbf{e}_2$ (surface of constant slope case) here. These cases all lead to a planar or spherical biarc scheme (Leopoldseder and Pottmann, 1998). All three cases allow a simple error estimation which is in simple relation to the approximation errors of the biarcs: In case of the approximation of a (general) cylinder or a surface of constant slope with cone pairs, the approximation error is restricted with the approximation error of the planar biarc. In the cone case the approximation
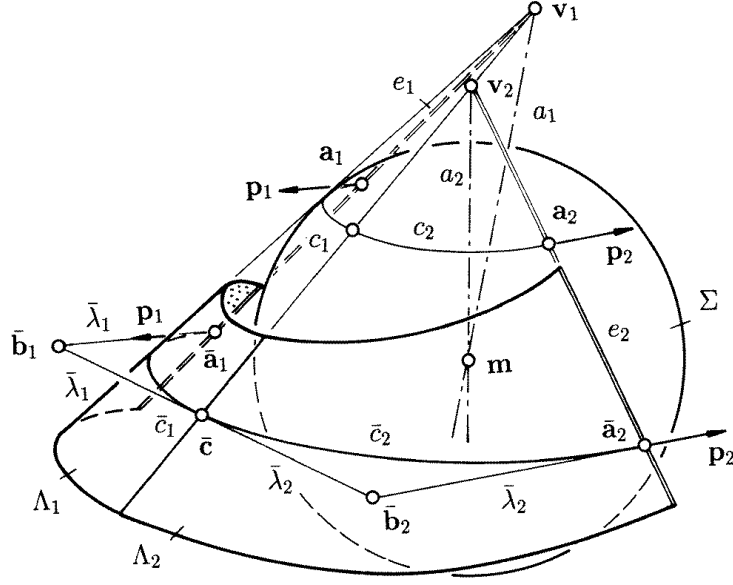
Figure 2: Cone pair with common inscribed sphere

error increases linearly with the distance to the cone vertex. For literature on approximation errors of planar biarc approximation we refer to Meek and Walton (1995).

In the general case there exists a unique sphere Σ that touches τ_i in points $\mathbf{a}_i$ of $e_i, i = 1, 2$. Its midpoint $\mathbf{m}$ can be found as the intersection point of three planes, namely the two normal planes through e_i to τ_i and the bisector plane of τ_1 and τ_2.

An admissible cone pair Λ_1, Λ_2 joining (e_i, τ_i) has to touch this sphere Σ along a biarc c_1, c_2 (see Figure 2). It is well known (see e.g. Sharrock, 1986; Fuhs and Stachel, 1988) that there exists a one parameter family of biarcs c_1, c_2 on Σ joining $(\mathbf{a}_1, \mathbf{p}_1)$ and $(\mathbf{a}_2, \mathbf{p}_2)$. The vertices $\mathbf{v}_1 = \mathbf{a}_1 + \mu_1 \mathbf{e}_1$ and $\mathbf{v}_2 = \mathbf{a}_2 - \mu_2 \mathbf{e}_2$ of the corresponding cone segments have to satisfy $\|\mathbf{v}_2 - \mathbf{v}_1\| = |\mu_1 + \mu_2|$ leading to the bilinear equation

$$(\mathbf{a}_2 - \mathbf{a}_1)^2 - 2\mu_1(\mathbf{a}_2 - \mathbf{a}_1)\mathbf{e}_1 - 2\mu_2(\mathbf{a}_2 - \mathbf{a}_1)\mathbf{e}_2 + 2\mu_1\mu_2(\mathbf{e}_1\mathbf{e}_2 - 1) = 0. \quad (1)$$

Figure 3 shows a typical function of the vertex distance $\|\Delta\mathbf{v}\| = \|\mathbf{v}_2 - \mathbf{v}_1\|$ with respect to parameter μ_1. This (positive) function is of the type $\|\Delta\mathbf{v}\| = \left|\frac{A\mu_1 + B}{C\mu_1 + D}\right|$ with $A, B, C, D \in \mathbb{R}$. Minimizing the vertex distance leads to a quadratic equation in μ_1. One has to take attention that only one of the two solutions μ_1, $\bar{\mu}_1$ leads to a cone pair useful for applications. In section 4 it will be shown that in general the practical solution corresponds to the local minimum μ_1 that is *not* the global minimum. The global minimum $\bar{\mu}_1$ would produce a loop in the cone spline surface.
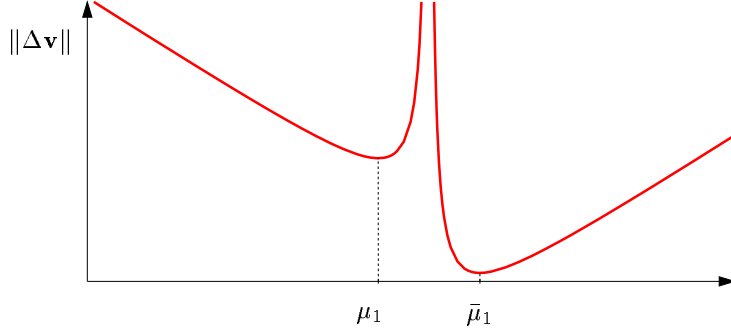
4

Figure 3: Vertex distance function

The chosen parameter μ_1 determines $\mathbf{v}_1$, and $\mathbf{v}_2$ via (1). Thus, we know the vertices $\mathbf{v}_i$, axes $a_i = \mathbf{v}_i \vee \mathbf{m}_i$, boundary rulings e_i and the junction ruling $e = \mathbf{v}_1 \vee \mathbf{v}_2$ of the cone segments Λ_i.

In applications one generally has a prescribed region of interest Ω which must be kept free of singularities. If one of the vertices, say $\mathbf{v}_1$, lies inside of Ω, one can easily move $\mathbf{v}_1$ to the nearer boundary of Ω via the parameter μ_1. One has to check, however, that the corresponding vertex $\mathbf{v}_2$ stays outside of Ω.

In section 4 we will motivate this minimization of the vertex distance. If the Hermite elements (e_i, τ_i) stem from a developable surface then in the limit case of $e_2 \to e_1$ the chosen cone pair is a practical one. One has to note however, that minimizing the vertex distance cannot be the best choice in situations where the original developable surface has a cylindrical ruling. There the singular point is at infinity and in general the singular points of neighboring rulings lie on different sides of the surface patch.

Therefore it is more stable to implement the following: Let $\bar{\mathbf{a}}_1 = \mathbf{a}_1 + \varepsilon \mathbf{e}_1$ be a point of the ruling e_1 which lies well inside the region of interest, and let $\bar{\mathbf{a}}_2 = \mathbf{a}_2 + \varepsilon \mathbf{e}_2$ be the point on e_2 to equal parameter ε. Each biarc $\bar{c}_1, \bar{c}_2$ joining $(\bar{\mathbf{a}}_i, \mathbf{p}_i)$ determines a suitable cone pair. Figure 2 gives the control points $\bar{\mathbf{a}}_1, \bar{\mathbf{b}}_1, \bar{\mathbf{c}}, \bar{\mathbf{b}}_2, \bar{\mathbf{a}}_2$ of the biarc $\bar{c}_1, \bar{c}_2$. The control points $\bar{\mathbf{b}}_1 = \bar{\mathbf{a}}_1 + \bar{\lambda}_1 \mathbf{p}_1$, $\bar{\mathbf{b}}_2 = \bar{\mathbf{a}}_2 + \bar{\lambda}_2 \mathbf{p}_2$ have to satisfy $\|\bar{\mathbf{b}}_2 - \bar{\mathbf{b}}_1\| = |\bar{\lambda}_1 + \bar{\lambda}_2|$ which, similar to (1), leads to the bilinear equation

$$(\bar{\mathbf{a}}_2 - \bar{\mathbf{a}}_1)^2 - 2\bar{\lambda}_1(\bar{\mathbf{a}}_2 - \bar{\mathbf{a}}_1)\mathbf{p}_1 - 2\bar{\lambda}_2(\bar{\mathbf{a}}_2 - \bar{\mathbf{a}}_1)\mathbf{p}_2 + 2\bar{\lambda}_1\bar{\lambda}_2(\mathbf{p}_1\mathbf{p}_2 - 1) = 0. \qquad (2)$$

Nutbourne and Martin (1988) have shown how to minimize the angle between the planes of $\bar{c}_1$ and $\bar{c}_2$ which is equivalent to minimizing the angle between the cone axes a_1 and a_2. This can be achieved by minimizing the distance $|\bar{\lambda}_1 + \bar{\lambda}_2|$ of the control points $\bar{\mathbf{b}}_1, \bar{\mathbf{b}}_2$. Additionally, the winding angles of $\bar{c}_1$ and $\bar{c}_2$ are equal then. Another good choice for $\bar{\lambda}_i$ is obtained by $\bar{\lambda}_1 = \bar{\lambda}_2, \bar{\lambda}_i > 0$. Here one has to compute the positive root of a quadratic equation.

5

As one could expect, in the limit case $e_2 \to e_1$ the cone pairs determined by $\bar{c}_1, \bar{c}_2$ converge similar to the cone pairs obtained by minimizing the vertex distance (see section 4).

3 Differential geometric fundamentals of developable surfaces

Developable surfaces are surfaces that can be isometrically mapped into the plane. Assuming sufficient differentiability, they are characterized by the property of possessing vanishing Gaussian curvature. All nonflat developable surfaces are envelopes of one parameter sets of planes. It is a well-known result of differential geometry (see e.g. Kruppa, 1957) that such a developable surface is either a conical surface, a cylindrical surface, the tangent surface of a twisted curve or a composition of these three surface types. Thus, developable surfaces are ruled surfaces, but with the special property that they possess the same tangent plane at all points of the same generator.

The parametric representation of a ruled surface Γ is

$$\mathbf{g}(u, v) = \mathbf{l}(u) + v\mathbf{e}(u), \tag{3}$$

where $\mathbf{l}(u)$ represents a curve on Γ with respect to arc length u and $\mathbf{e}(u)$ are unit vectors of the generator lines. For a cylinder $\mathbf{e}$ is constant, and for a cone we may choose $\mathbf{l} = const$ as the vertex. In a differential geometric treatment, a tangent surface is written in the form (3) with u as arc length of the line of regression $\mathbf{l}$, and $\mathbf{e} = \mathbf{l}'(u)$ with $\mathbf{l}' = dl/du$. Let $\mathbf{e}, \mathbf{p}, \mathbf{n}$ be the Frenet frame vectors of $\mathbf{l}$, with $\mathbf{p}$ and $\mathbf{n}$ as principal normal and binormal, respectively, and let $\varkappa$ and τ be curvature and torsion of $\mathbf{l}$. Then the *Darboux vector*

$$\mathbf{d}(u) = \tau(u)\mathbf{e}(u) + \varkappa(u)\mathbf{n}(u) \tag{4}$$

defines the axis $a(u)$: $\mathbf{l}(u) + \lambda\mathbf{d}(u)$ of a cone of revolution $\Delta(u)$ with vertex $\mathbf{l}(u)$, which touches Γ along the generator $e(u)$. Δ is called *osculating cone*, since it has contact of order 2 with Γ at all regular points of the common ruling. This cone may be considered as the counterpart of the osculating circle of a curve; it determines the curvature behavior of a developable surface along a ruling. The conical curvature

$$k(u) = \tau(u)/\varkappa(u) \tag{5}$$

equals $k = cot(\alpha)$ where α denotes half of the opening angle of Δ. This follows from (4).

The rectifying planes $\rho(u) = \mathbf{l}(u) + v_1\mathbf{e}(u) + v_2\mathbf{n}(u)$ of $\mathbf{l}(u)$ envelop another developable surface Γ^*. Its generators are the axes $a(u)$ of $\Delta(u)$ (Figure 4). The
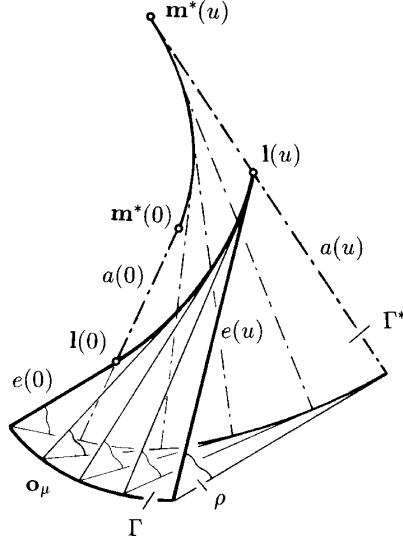
Figure 4: Kinematic generation of Γ as a moulding surface

line of regression of Γ^* is composed of the cuspoidal points

$$\mathbf{m}^*(u) = \mathbf{l}(u) - \frac{\varkappa}{\varkappa\tau' - \varkappa'\tau}(u)\left(\tau(u)\mathbf{e}(u) + \varkappa(u)\mathbf{n}(u)\right) \tag{6}$$

on $a(u)$, see e.g. Kruppa (1957). By using equation (5) this can be simplified to

$$\mathbf{m}^*(u) = \mathbf{l}(u) - \frac{1}{k'(u)}\left(k(u)\mathbf{e}(u) + \mathbf{n}(u)\right). \tag{7}$$

If a plane ρ rolls on a developable surface Γ^* then a curve e in ρ will trace out a so-called *moulding surface*. Recently, approximation algorithms for moulding surfaces have been proposed in Pottmann et al. (1998). Figure 4 identifies developable surfaces Γ to be special moulding surfaces where the generating curve e is a straight line. During this motion each point $\mathbf{l}(0) + \mu\mathbf{e}(0)$ on e traces out a surface curve

$$\mathbf{o}_\mu(u) = \mathbf{l}(u) + (\mu - u)\mathbf{e}(u), \qquad \mu \in \mathbb{R} \tag{8}$$

of Γ. Such a curve intersects all generators of Γ orthogonally as $\mathbf{o}_\mu' \cdot \mathbf{l}' = 0$ is satisfied. These *orthogonal trajectories* of Γ or *filar involutes* of $\mathbf{l}(u)$ are a spatial counterpart of involutes of planar curves and form an orthogonal parameter net on Γ together with the generators $e(u)$. For fixed parameter u the osculating circles of $\mathbf{o}_\mu(u), \mu \in \mathbb{R}$ have a common rotation axis and lie on the osculating cone $\Delta(u)$.

7

4 Approximation quality

Let the given developable surface Γ be the tangent surface of a twisted curve $\mathbf{l}(u)$ which is sufficiently differentiable. Let u denote its arc length, $\varkappa(u)$ and $\tau(u)$ its curvature and torsion. After choosing two Hermite elements $(e_i, \tau_i), i = 1, 2$ of Γ our algorithm is based on a sphere Σ which touches both Hermite elements. On Σ we obtain biarcs which correspond to cone pairs joining the Hermite elements. We now want to analyze our algorithm for Hermite elements (e_i, τ_i) which lie 'close' to each other.

By using the Taylor expansion of the line of regression $\mathbf{l}(u)$ at the point $\mathbf{l}(0)$ we will look into the limit case of joining Hermite elements $(e(0), \tau(0))$ and $(e(u), \tau(u))$ for $u \to 0$. The spheres $\Sigma(u)$ touching these Hermite elements will turn out to converge to a limit sphere which does not possess vanishing radius. A geometric interpretation of this limit sphere will be given.

There is a one parameter set of cone pairs joining the Hermite elements $(e(0), \tau(0))$ and $(e(u), \tau(u))$. In section 4.1 we take that cone pair that minimizes the cone vertex distance. In section 4.2 we eliminate the free parameter via a local biarc $\bar{c}_1, \bar{c}_2$ as discussed in section 2. The feasibility of these choices of the free parameter will be derived from the following calculations.

The Taylor expansion of the line of regression $\mathbf{l}(u)$ of Γ at the point $\mathbf{l}(0)$ is

$$\mathbf{l}(u) = \mathbf{l}(0) + \mathbf{l}'(0)u + \frac{\mathbf{l}''(0)}{2!}u^2 + \frac{\mathbf{l}^{(3)}(0)}{3!}u^3 + \frac{\mathbf{l}^{(4)}(0)}{4!}u^4 + O(u^5). \qquad (9)$$

Denoting tangent vector, principal normal vector and binormal vector in $\mathbf{l}(0)$ with $\mathbf{e} = \mathbf{e}(0)$, $\mathbf{p} = \mathbf{p}(0)$, $\mathbf{n} = \mathbf{n}(0)$ and employing the Frenet formulae for spatial curves the derivatives of $\mathbf{l}(u)$ with respect to arc length u are given by

$$\begin{aligned}
\mathbf{l}'(0) &= \mathbf{e}, \\
\mathbf{l}''(0) &= \varkappa\mathbf{p}, \\
\mathbf{l}^{(3)}(0) &= -\varkappa^2\mathbf{e} + \varkappa'\mathbf{p} + \varkappa\tau\mathbf{n}, \\
\mathbf{l}^{(4)}(0) &= -3\varkappa\varkappa'\mathbf{e} + (\varkappa'' - \varkappa^3 - \varkappa\tau^2)\mathbf{p} + (2\varkappa'\tau + \varkappa\tau')\mathbf{n}, \quad \dots
\end{aligned} \qquad (10)$$

where $\varkappa = \varkappa(0)$ and $\tau = \tau(0)$ are curvature and torsion at $\mathbf{l}(0)$ and $\varkappa' = (d\varkappa/du)(0)$, $\varkappa'' = (d^2\varkappa/du^2)(0)$, $\tau' = (d\tau/du)(0)$, $\dots$ the derivatives of $\varkappa(u)$ and $\tau(u)$ with respect to arc length in $\mathbf{l}(0)$. With $\mathbf{l}(0)$ as origin and the Frenet vectors $\mathbf{e}, \mathbf{p}, \mathbf{n}$ as axes of a coordinate system (9) and (10) give

$$\mathbf{l}(u) = \begin{pmatrix} u & - & \varkappa^2\frac{u^3}{3!} & - & & 3\varkappa\varkappa'\frac{u^4}{4!} & + & O(u^5) \\ \varkappa\frac{u^2}{2!} & + & \varkappa'\frac{u^3}{3!} & + & (\varkappa'' - \varkappa^3 - \varkappa\tau^2)\frac{u^4}{4!} & + & O(u^5) \\ & & \varkappa\tau\frac{u^3}{3!} & + & (2\varkappa'\tau + \varkappa\tau')\frac{u^4}{4!} & + & O(u^5) \end{pmatrix}. \qquad (11)$$

The tangent vector

$$
\mathbf{l}'(u) = \begin{pmatrix} 1 & - & \varkappa^2 \frac{u^2}{2!} & - & 3\varkappa\varkappa'\frac{u^3}{3!} & + & O(u^4) \\ \varkappa u & + & \varkappa'\frac{u^2}{2!} & + & (\varkappa'' - \varkappa^3 - \varkappa\tau^2)\frac{u^3}{3!} & + & O(u^4) \\ & & \varkappa\tau\frac{u^2}{2!} & + & (2\varkappa'\tau + \varkappa\tau')\frac{u^3}{3!} & + & O(u^4) \end{pmatrix} \tag{12}
$$

denotes the direction of the generator $e(u)$ of Γ. The normal vector of the tangent plane $\tau(u)$ along $e(u)$ is given by

$$
\mathbf{n}(u) = \frac{\mathbf{l}'(u) \times \mathbf{l}''(u)}{||\mathbf{l}'(u) \times \mathbf{l}''(u)||} \tag{13}
$$

which leads to

$$
\mathbf{n}(u) = \begin{pmatrix} & & \varkappa\tau\frac{u^2}{2!} & + & (2\varkappa\tau' + \tau\varkappa')\frac{u^3}{3!} & + & O(u^4) \\ -\tau u & - & \tau'\frac{u^2}{2!} & + & (-\tau'' + \tau\varkappa^2 + \tau^3)\frac{u^3}{3!} & + & O(u^4) \\ 1 & - & \tau^2\frac{u^2}{2!} & - & 3\tau\tau'\frac{u^3}{3!} & + & O(u^4) \end{pmatrix}. \tag{14}
$$

We will now use these equations to join the Hermite elements $(e(0), \tau(0))$ and $(e(u), \tau(u))$ by a cone pair. First we compute the sphere $\Sigma(u)$ which touches both Hermite elements. The center $\mathbf{m}(u)$ can be found as intersection of the normal planes

$$
\begin{aligned}
\nu_0 : & \quad (\mathbf{x} - \mathbf{l}(0)) \cdot \mathbf{l}''(0) = 0, \\
\nu_u : & \quad (\mathbf{x} - \mathbf{l}(u)) \cdot \mathbf{l}''(u) = 0
\end{aligned}
$$

with the tangent planes' bisector plane

$$
\sigma : \quad \mathbf{x} \cdot (\mathbf{n}(0) - \mathbf{n}(u)) - \mathbf{l}(0) \cdot \mathbf{n}(0) + \mathbf{l}(u) \cdot \mathbf{n}(u) = 0.
$$

Inserting (5), (12) and (14) one obtains

$$
\mathbf{m}(u) = \begin{pmatrix} -\frac{k}{k'} & + & \frac{kk''}{2k'^2}u & + & O(u^2) \\ & & 0 & & \\ -\frac{1}{k'} & + & \frac{k''}{2k'^2}u & + & O(u^2) \end{pmatrix}. \tag{15}
$$

The third coordinate of $\mathbf{m}(u)$ denotes the distance to generator $e(0)$ and is therefore the radius

$$
r(u) = -\frac{1}{k'} + \frac{k''}{2k'^2}u + O(u^2) \tag{16}
$$

of the sphere $\Sigma(u)$. With (4) we state

Theorem 4.1 *The unique sphere that touches two tangent planes $\tau(u_0), \tau(u_1)$ of a developable surface Γ in points of the rulings $e(u_0), e(u_1)$ converges to a limit sphere $\Sigma(u_0)$ when $u_1 \to u_0$. The midpoint $\mathbf{m}(u_0)$ of this limit sphere lies on the Darboux axis $a(u_0)$ of the ruling $e(u_0)$ and its radius equals $r(u_0) = -1/k'(u_0)$.*

With $\mathbf{m}^*(u)$ from formula (7) and its derivative

$$(\mathbf{m}^*)'(u) = \frac{kk''}{k'^2}(u)\mathbf{e}(u) + \frac{k''}{k'^2}(u)\mathbf{n}(u)$$

(15) simplifies to

$$\mathbf{m}(u) = \mathbf{m}^*(0) + \frac{1}{2}(\mathbf{m}^*)'(0)u + O(u^2). \tag{17}$$

Thus, in the first order approximation the midpoint $\mathbf{m}$ of Σ lies halfway between $\mathbf{m}^*(0)$ and $\mathbf{m}^*(u) = \mathbf{m}^*(0) + (\mathbf{m}^*)'(0)u + O(u^2)$.

The geometric interpretation of this property is illustrated in Fig. 5. Similar to
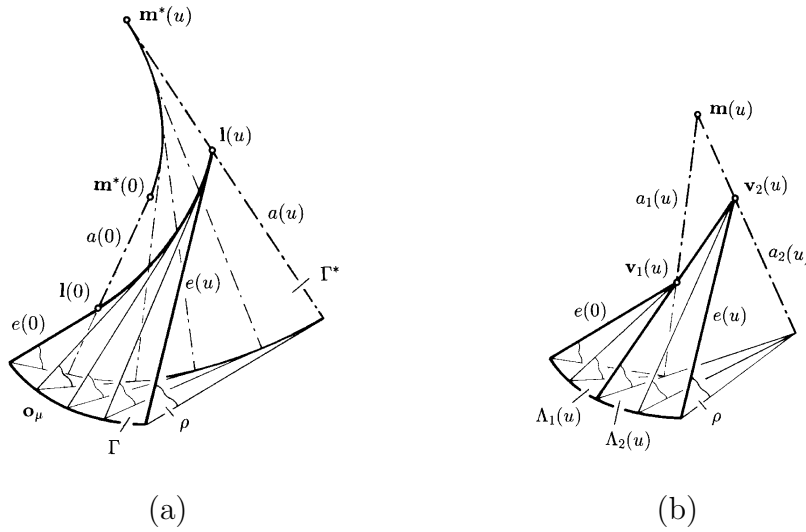
Figure 5: Kinematic generation of (a) a developable surface and (b) an approximating cone pair

the kinematic generation of a developable surface as a moulding surface (Fig. 5(a), see section 3) one can give a kinematic generation of its cone pair approximation. Let a plane ρ rotate around two intersecting axes a_1 and a_2 as in Figure 5(b). Then a line e in ρ will trace out a cone pair. Comparing Fig. 5(a) and (b), the set of Darboux axes $a(u)$ of Γ is replaced by two rotation axes $a_1(u)$ and $a_2(u)$ which intersect in the midpoint $\mathbf{m}(u)$ of $\Sigma(u)$ (see Figure 2). Formula (17) reveals this geometric property.

10

4.1 Minimal vertex distance

Let us assume fixed parameter value u for the moment. $\Sigma(u)$ and its midpoint $\mathbf{m}(u)$ are uniquely determined by the Hermite elements $(e(0), \tau(0))$, $(e(u), \tau(u))$. There is still a free parameter for the choice of cone pairs $\Lambda_1(u)$, $\Lambda_2(u)$ which we will now eliminate by minimizing the vertex distance $\|\mathbf{v}_2(u) - \mathbf{v}_1(u)\|$. Then, the vertices $\mathbf{v}_1(u)$ on $e_1 := e(0)$ and $\mathbf{v}_2(u)$ on $e_2 := e(u)$ merely depend on u. For the computation of the vertices $\mathbf{v}_1(u), \mathbf{v}_2(u)$ will minimize their distance, see section 2. One will expect $\mathbf{v}_1(u)$, $\mathbf{v}_2(u)$ to approximate $\mathbf{l}(0)$, $\mathbf{l}(u)$ as the vertex polygon of the cone spline surface shall approximate the line of regression.

First we need to calculate the points $\mathbf{a}_1(u)$ and $\mathbf{a}_2(u)$ where $\Sigma(u)$ touches $e_1 = e(0)$ and $e_2 = e(u)$ (see Figure 6 for $\mathbf{a}_1(u)$). With (15) the parameter $\lambda_1(u)$
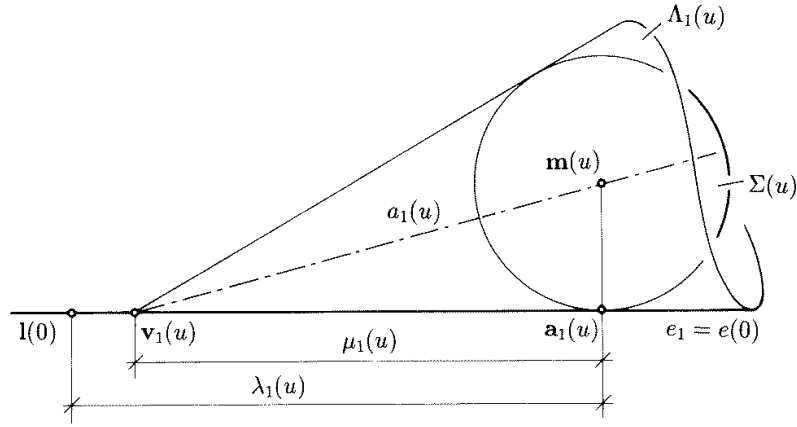

Figure 6: Sphere $\Sigma(u)$ touching $(e(0), \tau(0))$ and $(e(u), \tau(u)$

in $\mathbf{a}_1(u) = \mathbf{l}(0) + \lambda_1(u)\mathbf{l}'(0)$ equals

$$\lambda_1(u) = -\frac{k}{k'} + \frac{kk''}{2k'^2}u + O(u^2) \tag{18}$$

and this leads to

$$\mathbf{a}_1(u) = \begin{pmatrix} -\frac{k}{k'} & + & \frac{kk''}{2k'^2}u & + & O(u^2) \\ & & 0 & & \\ & & 0 & & \end{pmatrix}. \tag{19}$$

The second generator $e(u)$ is touched by $\Sigma(u)$ in $\mathbf{a}_2(u) = \mathbf{l}(u) + \lambda_2(u)\mathbf{l}'(u)$ to parameter

$$\lambda_2(u) = (\mathbf{m}(u) - \mathbf{l}(u)) \cdot \mathbf{l}'(u).$$

With (11), (12) and (15) this simplifies to

$$\lambda_2(u) = -\frac{k}{k'} + (\frac{kk''}{2k'^2} - 1)u + O(u^2) \tag{20}$$

11

and gives

$$\mathbf{a}_2(u) = \begin{pmatrix} -\frac{k}{k'} & + & \frac{kk''}{2k'^2}u & + & O(u^2) \\ & - & \frac{\tau}{k'}u & + & O(u^2) \\ & & & & O(u^2) \end{pmatrix}. \tag{21}$$

A possible vertex pair $\mathbf{v}_1(u) = \mathbf{a}_1(u) + \mu_1(u)\mathbf{e}_1(u)$, $\mathbf{v}_2(u) = \mathbf{a}_2(u) - \mu_2(u)\mathbf{e}_2(u)$ where $\mathbf{e}_1(u) = \mathbf{l}'(0)$, $\mathbf{e}_2(u) = \mathbf{l}'(u)$ has to fulfill the bilinear relation (1) for $\mu_1(u), \mu_2(u)$. Inserting (19) and (21) in the quadratic equation for the minimization of the vertex distance one obtains the two solutions

$$\mu_1(u) = \frac{k}{k'} + (-\frac{kk''}{2k'^2} + \frac{1}{3} - \frac{1}{6})u + O(u^2),$$
$$\bar{\mu}_1(u) = \frac{k}{k'} + (-\frac{kk''}{2k'^2} + \frac{1}{3} + \frac{1}{6})u + O(u^2). \tag{22}$$

Note, that one has to evaluate (19) and (21) up to third order in u to obtain (22).

In the following we will identify $\mu_1(u)$ of (22) as the useful solution while $\bar{\mu}_1(u)$ will not lead to a practically useful cone pair. Let us analyze the good solution first. With (18) and (22) the vertex $\mathbf{v}_1(u) = \mathbf{a}_1(u) + \mu_1(u)\mathbf{l}'(0) = \mathbf{l}(0) + (\lambda_1 + \mu_1)\mathbf{l}'(0)$ simplifies to

$$\mathbf{v}_1(u) = \mathbf{l}(0) + (\frac{1}{6}u + O(u^2))\mathbf{l}'(0). \tag{23}$$

Similarly the vertex $\mathbf{v}_2(u)$ of $\Lambda_2(u)$ equals

$$\mathbf{v}_2(u) = \mathbf{l}(u) - (\frac{1}{6}u + O(u^2))\mathbf{l}'(u). \tag{24}$$

The vector $\Delta\mathbf{v}(u) = \mathbf{v}_2(u) - \mathbf{v}_1(u)$ (Fig. 7) which gives the direction of the
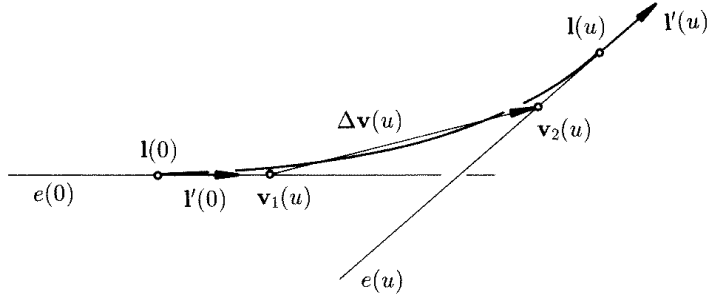

Figure 7: Line of regression $\mathbf{l}(u)$ with approximating vertex polygon

common generator $\mathbf{v}_1(u)\mathbf{v}_2(u)$ of $\Lambda_1(u)$ and $\Lambda_2(u)$ simplifies to

$$\Delta\mathbf{v}(u) = \frac{2}{3}u \begin{pmatrix} 1 & + & O(u^2) \\ \frac{\varkappa}{2}u & + & O(u^2) \\ & & O(u^2) \end{pmatrix}. \tag{25}$$

Thus, the junction generator $\mathbf{v}_1\mathbf{v}_2$ approximates the generator $e(u/2)$: $\mathbf{l}(u/2) + \lambda \mathbf{l}'(u/2)$ of Γ, since

$$\mathbf{l}'(u/2) = \begin{pmatrix} 1 & + & O(u^2) \\ \frac{\varkappa}{2}u & + & O(u^2) \\ & & O(u^2) \end{pmatrix}.$$

Equations (23) and (24) give

$$\|\mathbf{v}_1(u) - \mathbf{l}(0)\| = \|\mathbf{v}_2(u) - \mathbf{l}(u)\| = \frac{1}{6}u + O(u^2)$$

which, according to (25), asymptotically equals a quarter of the vertex distance

$$\|\mathbf{v}_2(u) - \mathbf{v}_1(u)\| = \frac{2}{3}u + O(u^2).$$

This shows that a cone spline surface generated with the cone pair algorithm possesses a vertex polygon $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \ldots$ such that asymptotically

$$\|\mathbf{v}_2 - \mathbf{v}_1\| + \|\mathbf{v}_4 - \mathbf{v}_3\| + \cdots =$$
$$= 2\left(\|\mathbf{v}_1 - \mathbf{l}(0)\| + \|\mathbf{v}_3 - \mathbf{v}_2\| + \|\mathbf{v}_5 - \mathbf{v}_4\| + \cdots\right)$$

holds true. This uneven distribution of the vertices can be noticed in Figure 8(b) which shows the development of a cone spline surface computed with minimizing the vertex distances.
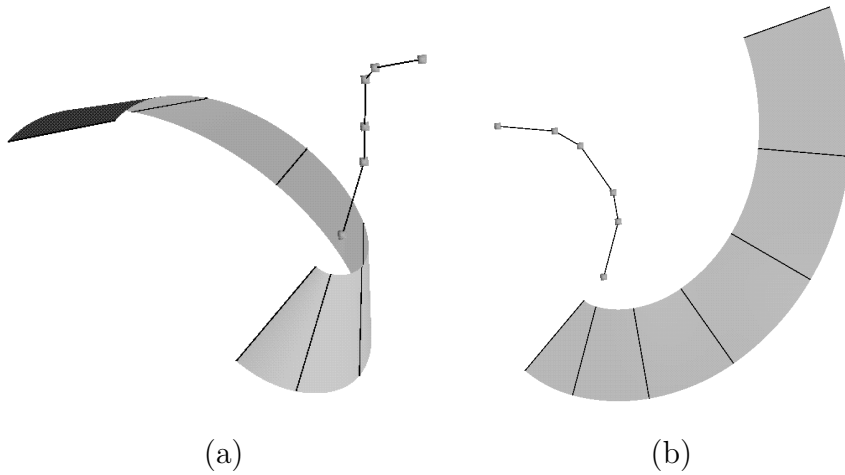

(a)   (b)

Figure 8: Cone spline surface (a) and its development (b)

Finally, let us focus on the second solution $\bar{\mu}_1(u)$ of (22). Completely similar to the calculations above we obtain vertices

$$\begin{aligned} \bar{\mathbf{v}}_1(u) &= \mathbf{l}(0) + (\tfrac{1}{2}u + O(u^2))\mathbf{l}'(0), \\ \bar{\mathbf{v}}_2(u) &= \mathbf{l}(u) - (\tfrac{1}{2}u + O(u^2))\mathbf{l}'(u). \end{aligned}$$

13

The vector

$$\Delta\bar{\mathbf{v}}(u) = \bar{\mathbf{v}}_2(u) - \bar{\mathbf{v}}_1(u) = \frac{1}{12}u^3 \begin{pmatrix} \varkappa^2 & + & O(u) \\ \varkappa & + & O(u) \\ \varkappa\tau & + & O(u) \end{pmatrix}$$

does not approximate the direction of a generator e of Γ. The resulting cone pair is not practical for the approximation of Γ.

With both solutions μ_1 and $\bar{\mu}_1$ the cones $\Lambda_1(u), \Lambda_2(u)$ converge to the osculating cone $\Delta(0)$ at ruling $e(0)$ as $u \to 0$. This can be easily seen, as the vertices $\mathbf{v}_1(u), \mathbf{v}_2(u)$ converge to the point $\mathbf{l}(0)$ and the sphere $\Sigma(u)$ — which is inscribed to $\Lambda_i(u)$ — converges to a sphere that is inscribed $\Delta(0)$, according to Theorem 4.1. The 'bad' solution $\bar{\mu}_1$ will produce a loop in the cone spline surface, which describes $\Delta(0)$ in its limit case.

Note, that the length of $\Delta\bar{\mathbf{v}}(u)$ is of third order in u in contrast with the length of $\Delta\mathbf{v}(u)$ in (25) which is of first order in u. Thus the solution $\bar{\mu}_1$ will — for sufficiently small u — lead to the *global* minimum of the vertex distance function.

We summarize:

Theorem 4.2 *Let Γ be the tangent surface of a twisted curve $l(u)$ and $(e, \tau)(u_i)$, $i = 1, 2$ two of its rulings plus tangent planes. When sufficiently close rulings $e(u_1), e(u_2)$ are joined with a G^1-cone pair while minimizing the distance of the cone vertices, there are two solutions: The global minimum will produce a loop, whereas the second local minimum provides a practical solution. In both cases the cone vertices — the singular points of the cone pair — lie close to the line of regression of Γ.*

4.2   Cone pairs via local biarc $\bar{c}_1, \bar{c}_2$

Here we will choose the free parameter not by minimizing the vertex distance $\|\mathbf{v}_2 - \mathbf{v}_1\|$ as in section 4.1 but via a biarc $\bar{c}_1, \bar{c}_2$ to parameters $\bar{\lambda}_1 = \bar{\lambda}_2$ (compare with Fig. 2 in section 2). $\bar{c}_1, \bar{c}_2$ will interpolate the G^1-data $(\bar{\mathbf{a}}_1, \mathbf{p}_1), (\bar{\mathbf{a}}_2, \mathbf{p}_2)$ where

$$\begin{aligned} \bar{\mathbf{a}}_1(u) &= \mathbf{a}_1(u) + (1 - \varepsilon)\frac{k}{k'} \cdot \mathbf{l}'(0), \\ \bar{\mathbf{a}}_2(u) &= \mathbf{a}_2(u) + (1 - \varepsilon)\frac{k}{k'} \cdot \mathbf{l}'(u) \end{aligned} \tag{26}$$

denote points on the rulings $e(0)$ and $e(u)$. The variable ε is chosen such that $\bar{\mathbf{a}}_i$ lie in the region of interest Ω. Note the constant $\frac{k}{k'}$ we have added in (26), thus ε gives a linear scaling on $e(0)$ with origin in $\mathbf{l}(0)$ for $\varepsilon = 0$ and unit point in $\mathbf{a}_1$ for $\varepsilon = 1$.

The direction vectors

$$\begin{aligned}
\mathbf{p}_1(u) &= \pm(0,1,0)^T, \\
\mathbf{p}_2(u) &= \pm\mathbf{n}(u) \times \mathbf{l}'(u)
\end{aligned} \tag{27}$$

are oriented such that they indicate the direction of the biarc $\bar{c}_i$ from $\bar{\mathbf{a}}_1$ to $\bar{\mathbf{a}}_2$.

Admissible Bézier control points $\bar{\mathbf{b}}_1(u) = \bar{\mathbf{a}}_1(u) + \bar{\lambda}_1(u)\mathbf{p}_1(u)$ and $\bar{\mathbf{b}}_2(u) = \bar{\mathbf{a}}_2(u) - \bar{\lambda}_1(u)\mathbf{p}_2(u)$ are described by the bilinear equation (2) in $\bar{\lambda}_1, \bar{\lambda}_2$. Letting $\bar{\lambda}_1 = \bar{\lambda}_2$ one obtains

$$\bar{\lambda}_1(u) = \frac{-N \pm \sqrt{N^2 - 4MO}}{2O}$$

with $M = (\bar{\mathbf{a}}_2(u) - \bar{\mathbf{a}}_1(u))^2$, $N = -2(\bar{\mathbf{a}}_2(u) - \bar{\mathbf{a}}_1(u))(\mathbf{p}_1(u) + \mathbf{p}_2(u))$ and $O = 2(\mathbf{p}_1(u)\mathbf{p}_2(u) - 1)$. Equations (19), (21) and (27) give the solution

$$\bar{\lambda}_1 = \bar{\lambda}_2 = \frac{1}{4}\left|\frac{\tau\varepsilon}{k'}\right| u + O(u^2).$$

Comparison with $\|\bar{\mathbf{a}}_2(u) - \bar{\mathbf{a}}_1(u)\| = \left|\frac{\tau\varepsilon}{k'}\right| u + O(u^2)$ identifies above solution as a practical one.

The plane of circle $\bar{c}_1$ is spanned by $\bar{\mathbf{a}}_1(u)$, $\bar{\mathbf{b}}_1(u) = \bar{\mathbf{a}}_1(u) + \bar{\lambda}_1\mathbf{p}_1(u)$ and $\bar{\mathbf{b}}_2(u) = \bar{\mathbf{a}}_2(u) - \bar{\lambda}_2\mathbf{p}_2(u)$. Its normal vector equals $\mathbf{d}(u) = (\tau + O(u), O(u), \varkappa + O(u))^T$ which is, for $u \to 0$, identical with the Darboux vector to generator $e(0)$, see equation (4). The axis of cone $\Lambda_1(u)$ passes through the midpoint $\mathbf{m}(u)$ (see equation (15)) of $\Sigma(u)$ and has direction vector $\mathbf{d}(u)$. Intersection of this axis with the generator $e(0)$ gives the cone vertex $\mathbf{v}_1(u)$ which again has representation (23). Analogous considerations lead to equation (24) for the second vertex $\mathbf{v}_2(u)$.

We conclude: In the limit $u \to 0$ the choice of the free parameter via a local biarc $\bar{c}_1, \bar{c}_2$ leads to the same cone pair solutions that can be obtained by minimizing the vertex distance as described in section 4.1.

5   Example and error diagrams

Figure 9 illustrates our 'cone pair algorithm': From a developable surface Γ (see Fig. 9(a)) four rulings plus tangent planes to parameters t_0, t_1, t_2, t_3 are chosen (see Fig. 9(b)). This data is interpolated by a cone spline surface. In Fig. 9(c) the six circular cone segments are displayed together with the corresponding cone vertices. In each of the three segments the free parameter was chosen via a local biarc $\bar{\mathbf{c}}_1, \bar{\mathbf{c}}_2$ with $\bar{\lambda}_1 = \bar{\lambda}_2$ (see Figure 2).

Figure 10 compares the conical curvature diagram $k(t)$ of above developable surface Γ to that of its approximating cone spline surface, which is piecewise constant, obviously. See equation (5) for the definition of the conical curvature.
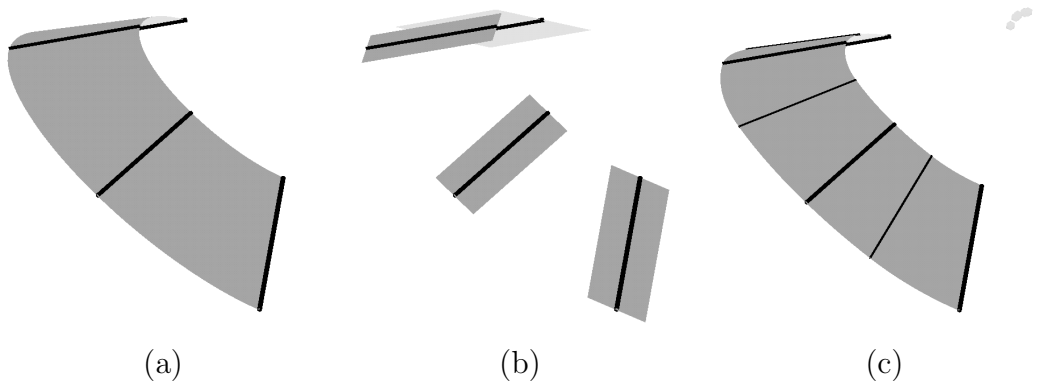
(a)   (b)   (c)

Figure 9: Developable surface (a), G^1-Hermite data (b) and approximating cone spline surface (c)
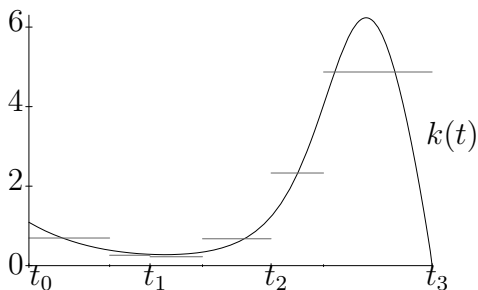

Figure 10: Conical curvature diagrams

Finally, Figure 11 gives the approximation error diagram plotted with contour lines over the parameter domain of Γ. Clearly, errors are zero along the chosen rulings $e(t_i), i = 0, \ldots, 3$. The sign of the approximation error indicates on which side of Γ the circular cone patches are lying. Here the maximum approximation error is smaller than 0.01 where the unit length 1 was defined as the length of the first ruling of Γ.

6 Sphere geometric connections between developable surfaces and spatial arc splines

In the algorithms of section 2 biarc methods were used to describe circular arc splines on cone spline surfaces. There is another close relationship to spatial arc splines, though.

Developable surfaces are the envelopes of their one parameter set of tangent planes, i.e. they are dual to a spatial curve. Analogously, a cone spline surface is dual to a G^1-spline curve whose segments are conics. We are specially interested
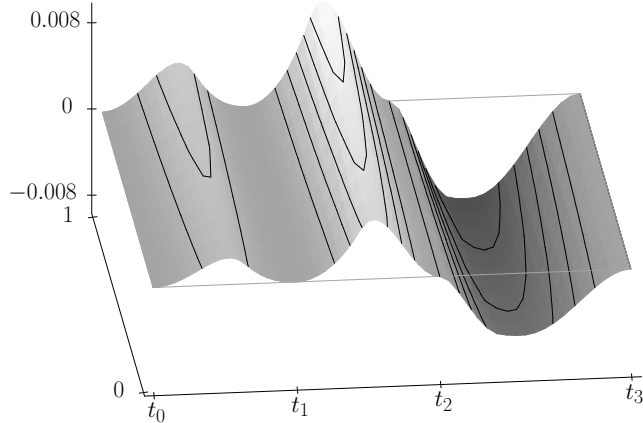
16

Figure 11: Approximation errors of example in Figure 9

in cones of revolution which are special examples of developable surfaces whose tangent planes all touch a one parameter set of spheres, additionally.

This property motivates using 3-dimensional Euclidean sphere geometry in which the elements are *oriented spheres* and *oriented planes* of Euclidean 3-space. The so-called isotropic model of this geometry can be obtained with a standard map of the oriented planes to points of a 3-dimensional space. The tangent planes of a cone of revolution are mapped to *isotropic circles*. These are, in general, ellipses whose top projection are Euclidean circles. The 'cone pair algorithm' summarized in section 2 is transformed to an isotropic biarc scheme that allows similar analytic treatment to the well-known Euclidean biarc scheme. For more details we refer to Leopoldseder (2000).

Also, the algorithm of *osculating arc splines* in Euclidean 3-space described in the next section can be applied to isotropic 3-space. There, selected isotropic osculating circles of a given curve are smoothly joined with isotropic arcs to an isotropic arc spline that osculates (is in second order contact with) the given curve in the selected data points. Translated back to the standard model of 3-dimensional Euclidean sphere geometry we obtain an approximation scheme for developable surfaces Γ where selected osculating cones of Γ are smoothly joined with cone segments to a G^1-cone spline surface (Leopoldseder and Pottmann, 1998).

We summarize: The sphere geometric approach enables to interpret developable surfaces as curves in an isotropic 3-space and cone spline surfaces as isotropic arc splines. This curve interpretation gives deeper insight in cone spline surfaces and enables to find important properties on cone spline algorithms that

17

could not easily be proved directly in Euclidean 3-space (see Leopoldseder, 2000)

7 Algorithms on Spatial Arc Splines

Let $g = \mathbf{g}(t)$ be a spatial curve in E^3 and $c_1, \ldots, c_n$ the osculating circles of g to parameters $t_1, \ldots, t_n$. Now each two consecutive osculating circles c_i, c_{i+1} shall be joined by an arc with G^1 continuity at the junction points. Thus, the given curve $g \in E^3$ is approximated by an arc spline where every second arc is an osculating arc of g. We will call such an arc spline *spatial osculating arc spline* henceforth.

Note that the case of g being a *planar* curve has been analyzed by Meek and Walton (1996). The case of g being a *spherical* curve has been treated by Leopoldseder and Pottmann (1998). Obviously, results on spherical curve approximation can be derived from the planar case by stereographically projecting the plane onto a sphere. We know that in both cases there exists a one parameter set of solution arcs joining two given (oriented) circles. In the following we will restrict ourselves to the non-planar and non-spherical case.

In section 8 we will first present a geometric method of finding an arc c connecting two oriented circles in E^3. Note, that we will tacitly allow a circle to degenerate to a straight line which one can interpret as a circle with infinite radius. We will briefly consider all the special cases and see that there are two complex solutions in the general case.

In section 9 we will take a closer look at the approximation errors and propose an algorithm for a good segmentation of the given curve g. Choosing appropriate initial osculating circles c_i of g clearly has great influence on the quality of the approximation. Finally, in section 10 these results are applied to several examples.

8 Method

Let σ_1, σ_2 be the planes containing the oriented Euclidean circles c_1, c_2 in E^3. We first assume σ_i not to be parallel and treat special cases later. Our goal is to find control points $\mathbf{c}_1, \mathbf{d}, \mathbf{c}_2$ of an arc c joining c_1, c_2 while preserving the circles' orientation (Figure 12).

Obviously, the middle control point $\mathbf{d}$ has to lie on the line $s = \sigma_1 \cap \sigma_2$ and possess equal tangential distance to c_1 and c_2. To determine the location of $\mathbf{d}$ we will first rotate σ_2 around s such that σ_2 and σ_1 coincide. The chordal line d of c_1 and the rotated circle $\bar{c}_2$ in σ_1 contains all points of σ_1 with equal tangential distance to c_1 and $\bar{c}_2$. Its equation is

$$d : 2\mathbf{x}(\mathbf{m}_2 - \mathbf{m}_1) - (r_1^2 - r_2^2) + (\mathbf{m}_1^2 + \mathbf{m}_2^2) = 0$$
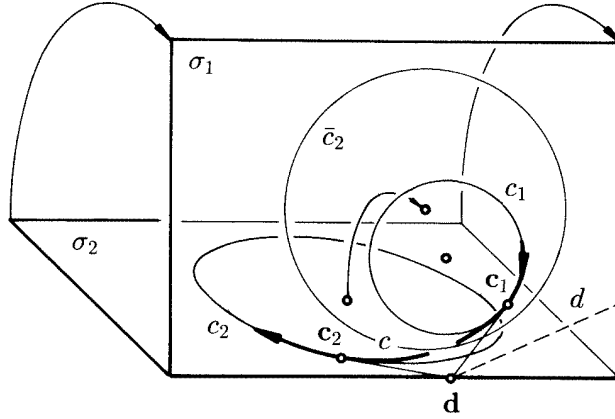
Figure 12: Spatial osculating arc spline

where r_i denote the radii and $\mathbf{m}_i$ the midpoints of the circles. The only possible location of $\mathbf{d}$ is the intersection point of d with s. If $\mathbf{d}$ lies outside of c_1 and c_2 the missing two control points $\mathbf{c}_1$ and $\mathbf{c}_2$ of c can be found by laying tangents from $\mathbf{d}$ to c_1 and c_2. We obtain two solutions if the orientation of c_i is taken into consideration. These solutions are not real if $\mathbf{d}$ lies in the interior of c_1 and c_2. Note, that both possibilities to rotate σ_2 into σ_1 lead to the same point $\mathbf{d}$.

Special cases that have to be treated separately include the spherical case ($s = d$) and the planar case ($\sigma_1 = \sigma_2$), both of them leading to a one parameter set of solutions. If $\sigma_1 \parallel \sigma_2$, $\sigma_1 \neq \sigma_2$ the method given above does not work. Here, the junction points of a solution arc c lie in the common plane of symmetry of c_1 and c_2 (Figure 13).
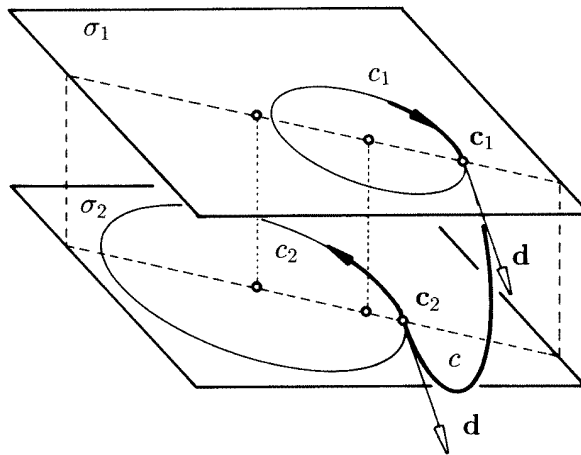
Figure 13: Special case: $\sigma_1 \parallel \sigma_2$

19

If one of the two circles, say c_1, is a line, the intersection of c_1 with σ_2 gives **d**. Figure 14 shows one of the solution arcs which is real because **d** is lying on the outside of c_2. Last but not least, it is obvious that there are no connecting
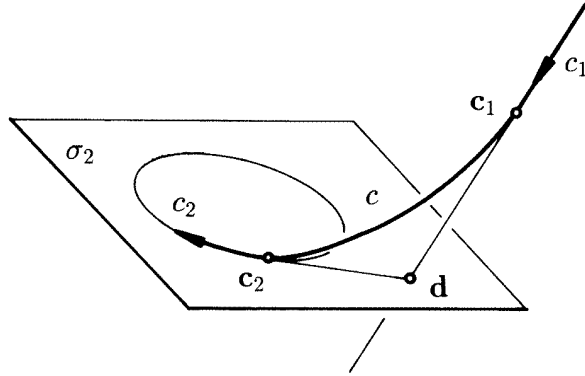

Figure 14: Special case: c_1 is a straight line

arcs in case of two skew lines c_1 and c_2.

Theorem 8.1 *Let c_1, c_2 denote two oriented circles in Euclidean 3-space E^3, lying in general spatial position. There exist two complex circular arcs c which join c_i with G^1-continuity and preserve the orientation of c_i. If the two circles c_i are lying in a plane or on a common sphere there is a one parameter family of solution arcs.*

For our algorithm it is necessary that the solution arcs are real, obviously. This can be guaranteed as we are not interested in joining arcs of two circles with arbitrary spatial position. Our circles to be joined are osculating circles of a given spatial curve, and here we can state:

Theorem 8.2 *Let $\mathbf{g}(t)$ be a piecewise C^∞ curve in Euclidean 3-space E^3. To any point $\mathbf{g}(t_1)$ there exists a parameter interval $U =]t_1, t_1 + \Delta t] \subset \mathbb{R}$ such that the points $\mathbf{g}(t_1)$ and $\mathbf{g}(t_2), t_2 \in U$ can be joined with a Euclidean triarc in the following way: the first and the third arc of this triarc lie on the Euclidean osculating circles c_1 and c_2 of $\mathbf{g}(t)$ to parameters t_1 and t_2. The joining Euclidean arc c is real and joins c_1 and c_2 with G^1-continuity while preserving the orientation of c_i.*

The proof can be found in a separate paper (Leopoldseder, 2000) since here we do not want to deal with the sphere geometric methods necessary for this proof.

9 Segmentation

9.1 Error estimates

The approximation errors of a spatial arc spline can be judged easily. Let $\mathbf{x}$ be a point of the given curve $\mathbf{g}(t)$ and c one of the arc splines' segments with Bézier points $\mathbf{b}_i$ and midpoint $\mathbf{m}$ (Figure 15). It is computationally easy to check if the
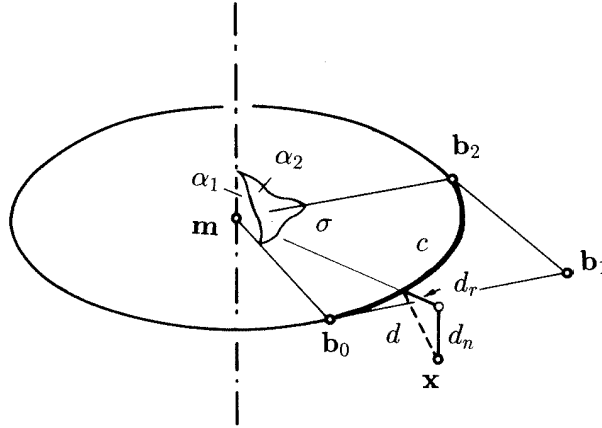

Figure 15: Distance of a point $\mathbf{x}$ to an arc c

point $\mathbf{x}$ lies in the wedge between the planes α_1 and α_2 through the rotation axis of c. The Euclidean distance of $\mathbf{x}$ to the circle segment c then is

$$d(\mathbf{x}, c) = \sqrt{d_n^2 + d_r^2}. \tag{28}$$

where d_n denotes the component lying normal to the circles' plane σ and d_r is the radial distance. In the following we will orient d_r such that $d_r > 0$ indicates that the normal projection of $\mathbf{x}$ onto σ lies outside of c. Also the orientation of c determines an orientation of σ and thus an orientation of d_n. In the degenerating case of c being a straight line segment the distance $d(\mathbf{x}, c)$ again is well-defined but we will not distinguish between radial and normal components.

If we choose a segmentation $\mathbf{g}(t_1), \ldots, \mathbf{g}(t_n)$ of the given curve $\mathbf{g}(t)$ the osculating circles to the parameters $t_1, \ldots, t_n$ will be used in the approximating arc spline. It is natural to select those points where the osculating circle approximates $\mathbf{g}(t)$ well. Let

$$\mathbf{g}(u) = \begin{pmatrix} u & - & \varkappa^2 \frac{u^3}{3!} & + & O(u^4) \\ \varkappa \frac{u^2}{2!} & + & \varkappa' \frac{u^3}{3!} & + & O(u^4) \\ & & \varkappa\tau \frac{u^3}{3!} & + & O(u^4) \end{pmatrix} \tag{29}$$

be the Taylor expansion of $\mathbf{g}(t)$ at the point $\mathbf{g}(0)$ with respect to arc length u (compare with (11)). The Frenet frame in $\mathbf{g}(0)$ is used as coordinate system and

21

$\varkappa = \varkappa(0)$, $\varkappa' = \frac{d\varkappa}{du}(0)$ and $\tau = \tau(0)$ denote the differential invariants evaluated at $\mathbf{g}(0)$. The osculating arc c_0 at $\mathbf{g}(0)$ has midpoint

$$\mathbf{m}_0 = (0, \frac{1}{\varkappa}, 0)^T. \tag{30}$$

The third coordinate in (29) gives the normal component

$$d_n(u) = \frac{1}{6}\varkappa\tau u^3 + O(u^4) \tag{31}$$

of the distance $d(u)$ between $\mathbf{g}(u)$ and c_0. The radial component is

$$d_r(u) = \|\widetilde{\mathbf{g}}(u) - \mathbf{m}_0\| - r_0$$

where $\widetilde{\mathbf{g}}(u)$ is the normal projection of $\mathbf{g}(u)$ into the xy-plane and r_0 the radius of c_0. With (29), (30) and $r_0 = \frac{1}{\varkappa}$ its Taylor expansion simplifies to

$$d_r(u) = -\frac{1}{6}\varkappa' u^3 + O(u^4) \tag{32}$$

and together with (28) and (31) we have

$$d(u) = \frac{1}{6}\sqrt{\tau^2\varkappa^2 + \varkappa'^2}u^3 + O(u^4). \tag{33}$$

for the distance between $\mathbf{g}(u)$ and c_0.

The leading term of (33) provides us with a function

$$F(t) = \frac{1}{6}\sqrt{\tau(t)^2\varkappa(t)^2 + \varkappa'(t)^2} \tag{34}$$

such that $F(t_i)$ indicates the approximation quality of the osculating circle $c(t_i)$ for $t \to t_i$. $F(t_i) = 0$ is equivalent to $\varkappa'(t_i) = 0, \tau(t_i) = 0$ or $\varkappa'(t_i) = 0, \varkappa(t_i) = 0$. In the second case $c(t_i)$ degenerates to a straight line, in both cases $c(t_i)$ and $g = \mathbf{g}(t)$ hyperosculate, i.e. they are in contact of order 3.

9.2   Segmentation algorithm

A segmentation $t_0 = t_a, t_1, \ldots, t_n$ of the curve $\mathbf{g}(t), t \in [t_a, t_b]$ is practicable if the following three criteria hold true for each segment:

Criterion 1: The connecting circle of $c(t_i)$ and $c(t_{i+1})$ is real.

Criterion 2: The maximal error between the curve $\mathbf{g}(t), t \in [t_i, t_{i+1}]$ and the resulting triarc does not exceed a chosen error tolerance.

Figure 16: Violation of Criterion 3

Criterion 3: Both of the two inner joining points of the triarc lie on the right side of the oriented osculating circles $c(t_i)$ and $c(t_{i+1})$ (compare with the bad case in Figure 16).

Note, that in some cases a useful arc spline can be constructed, although criterion 3 is not fulfilled (see example 2, for instance). This depends on the joining point of the next arc segment, though.

Possible segmentation algorithms include

- **The bisection method:** One chooses an evenly distributed segmentation with respect to arc length first. Each segment is bisected for which one of the criteria given above does not hold true. Clearly, this method is easy to implement but will not produce evenly distributed segmentations, in general.

- **Longest triarc method:** Let $t_0 = t_a$ be the first segmentation point. The next one, t_1, will be chosen as largest parameter value such that criteria 1 to 3 are still fulfilled. This procedure is repeated for the next segments. Thus, one minimizes the number of arc segments involved but completely ignores the geometric properties of the given curve. The approximation of a closed curve would give different results depending on the starting point, for instance.

Here a different approach will be proposed. An initial segmentation will be performed at the local minimas of the function $F(t)$ of (34). Figure 17 shows $F(t)$ for a polynomial curve of degree 4 which is taken from example 2 of section 10. The osculating arcs to parameters t_0, t_2, t_4, t_6, t_9 well approximate the curve $\mathbf{g}(t)$. These five segmentation points give four triarc segments. As each of the osculating circles $c(t_2), c(t_4), c(t_6)$ is used in two triarc segments the computed arc spline is composed of nine arc segments (see Figure 23). Figure 18 shows the approximation error of this initial approximation. Note, that in the third triarc segment $[t_4, t_6]$ criterion 3 is violated. The resulting arc spline still is useful although the segment on the osculating arc $c(t_6)$ is very short. In order
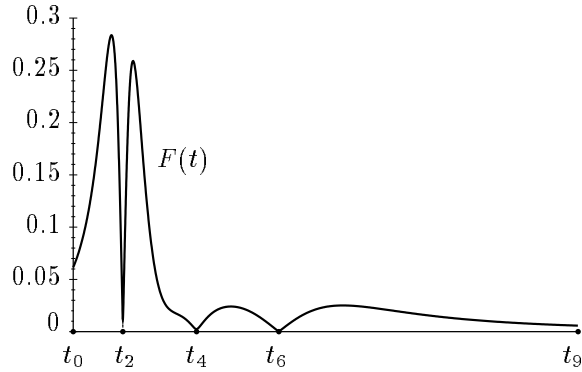
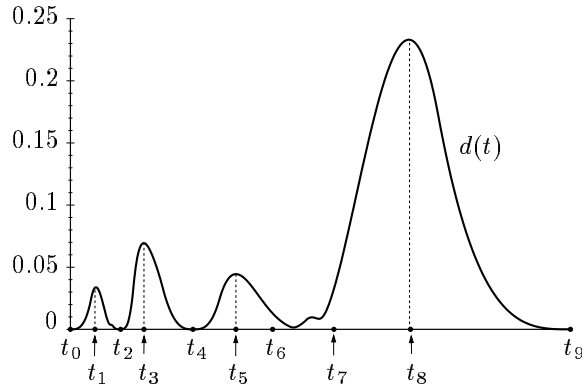Figure 17: Function $F(t)$ for a polynomial curve of degree 4.

Figure 18: Approximation error $d(t)$ after initial segmentation

to avoid such situations and in order to reduce the error to the desired accuracy one has to subdivide segments.

For the subdivision point we choose the peak value of the error function $d(t)$ within a segment (t_1, t_3, t_5, t_7). If $d(t)$ has two peaks of approximately the same value (see for instance Fig. 20 (a), (b) or Fig. 25) one either splits in between or divides into three segments.

After computing the triarc of the segment $[t_6, t_8]$ still the error tolerance of 0.01 is exceeded at this segment (the corresponding figure for $d(t)$ is not given). Subdivision at the maximum error gives t_7. The final arc spline is presented in Figure 24 of the next section.

10   Examples and error diagrams

Example 1: Helical curve Figure 19 (a) shows the approximation of a helical curve (thin curve) by one triarc segment (thick curve) in top view and front
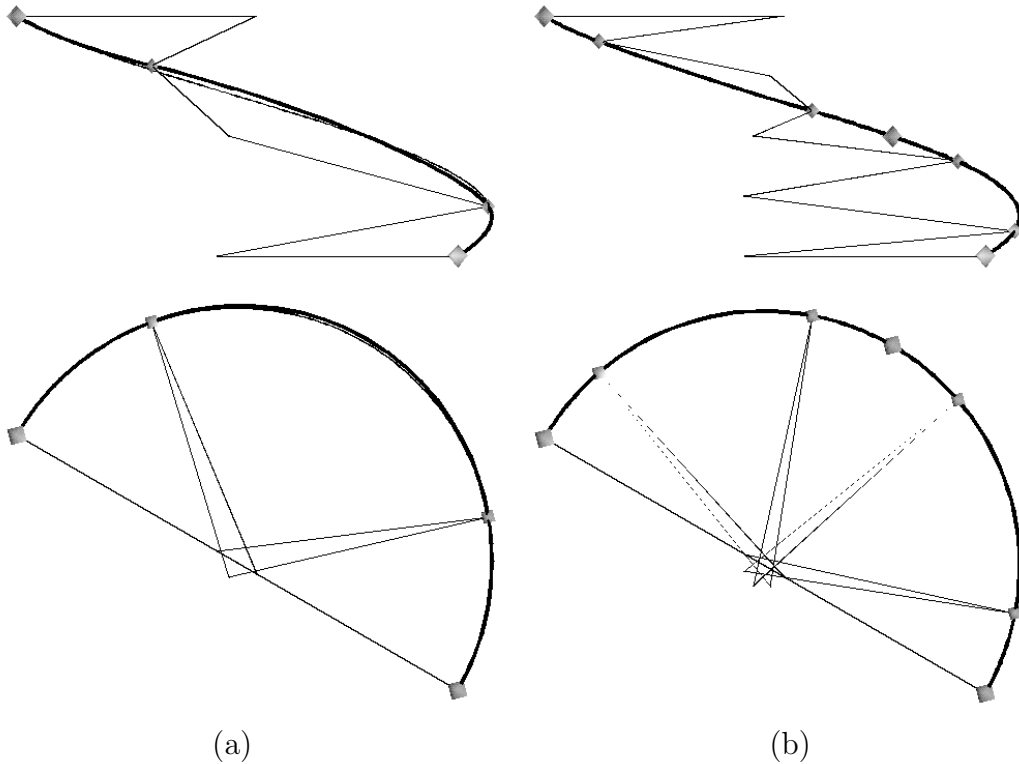
24

Figure 19: Approximation with (a) one, (b) two triarc segments

view. Figure 19 (b) shows the approximation of the same curve with two triarc segments. The big octahedrons indicate the curve points whose osculating circles were computed. The smaller octahedron are the joining points of different arc segments. In order to better illustrate the spatial position of the arc segments their end points are connected to their midpoint with thin lines.

Figure 20 (a) and (b) contain diagrams showing the approximation error $d(u)$ of the helical curve to the arc splines. Here u is the arc length of the helix. The occurring error has its peaks at the middle arc segment which can also be clearly seen in Figure 19 (a). Note the different scalings in the diagrams Fig. 20 (a) and (b) which tend to disguise the fact that the error has decreased approximately by factor 1/8 because the segment has been bisected.

In Figure 21 the error $d(u)$ of Fig. 20(a) is split into its normal and radial components $d_n(u)$ and $d_r(u)$. The function $d_n(u)$ shows that the helical curve lies on both sides of the plane of the middle arc segment. At the point of vanishing d_n the radial distance d_r reaches is maximum. While $d(u)$ is smooth the functions $d_n(u)$ and $d_r(u)$ have discontinuities at the junction points. This occurs because of the twist angle between the planes containing the adjacent arc segments. In the present example the angle between two consecutive planes equals 25.8°.

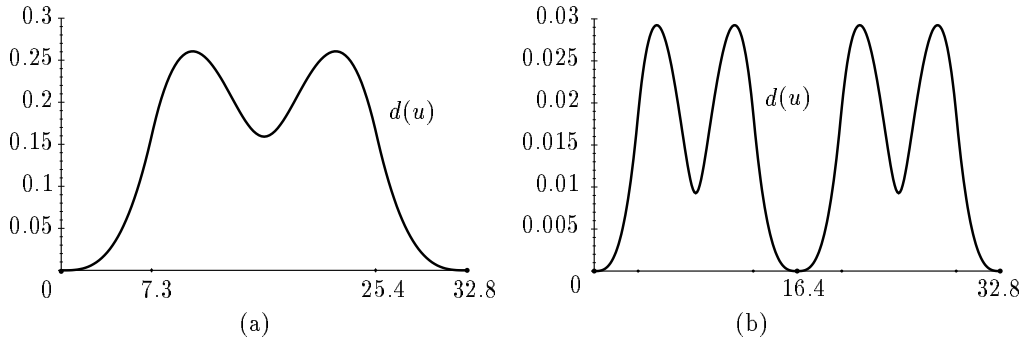Finally, in Figure 22 we look at curvature and torsion diagrams $\varkappa(u)$, $\tau(u)$

25

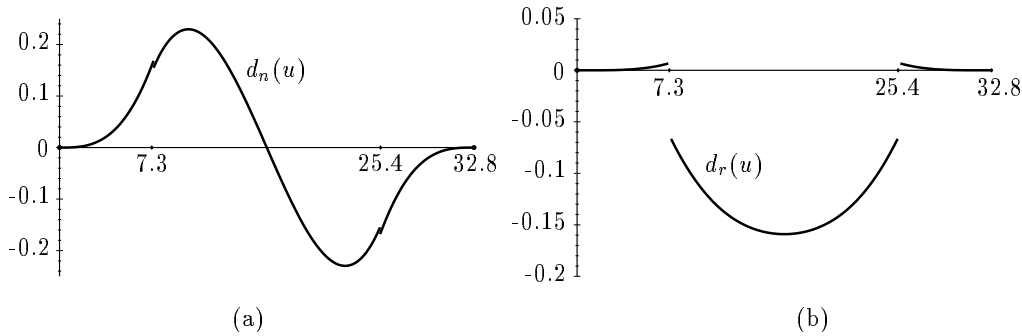Figure 20: Approximation error $d(u)$ for (a) one, (b) two triarc segments

Figure 21: (a) Normal component $d_n(u)$, (b) radial component $d_r(u)$ of the approximation error $d(u)$ of Fig. 20(a)

of the helical curve, both of them being constant functions. The dotted lines indicate the curvature and torsion profile of the approximating arc spline. The small difference in arc length between curve and arc spline has been treated by scaling. Obviously the curvature of each arc segment is constant but the torsion profile of the arc spline needs to be explained. As the arcs are planar curves the torsion is zero at all points except at the junction points where the torsion is not defined. For a sufficiently smooth curve the torsion $\tau(t_0)$ at $\mathbf{g}(t_0)$ can be defined by the limes

$$\tau(t_0) = \lim_{\Delta t \to 0} \frac{\angle \sigma(t_0)\sigma(t_0 + \Delta t)}{\|\dot{\mathbf{g}}(t_0)\|} \tag{35}$$

where $\sigma(t)$ denotes the osculating plane at $\mathbf{g}(t)$. At the junction points of the arc spline there is a sudden rotation of the osculating plane around the tangent vector without a change in arc length. We will take over the term *torsion impulse* from Nutbourne and Martin (1988) for this behavior. We will define such a torsion impulse T at the joining point of two arc segments c_1, c_2 by

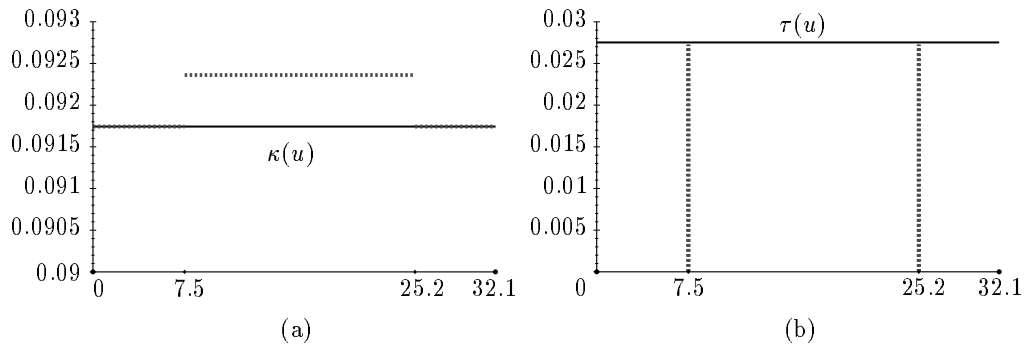$$T = \frac{\angle \sigma_0 \sigma_1}{\frac{1}{2}(l(c_1) + l(c_2))} \tag{36}$$

Figure 22: (a) Curvature and (b) torsion diagrams of a helical curve and its approximation with one triarc

where σ_i are the circles' planes and $l(c_i)$ the arc lengths of the segment c_i.
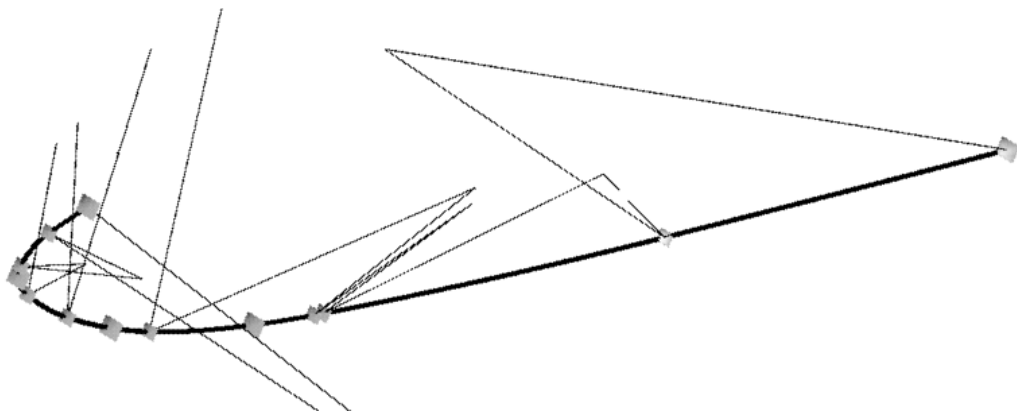
Example 2: Polynomial curve of degree 4

Figure 23: Approximation of a polynomial curve of degree 4 with four triarc segments (9 arc segments)

Figure 23 shows the arc spline approximation of a polynomial curve of degree 4. The bigger octahedra indicate the position of the segmentation points. These have been chosen at the local minima of the function $F(t)$ (Figure 17). After further subdivision of the segments as described in section 9.2 one obtains an osculating arc spline composed of 19 arc segments. Figure 24 gives the final result.

The approximation errors which go with this arc spline are given in Figure 25 Note the different scaling of the diagram compared to Fig. 18. The maximum absolute error of 0.01 gives a maximum relative error of 0.00025 with respect to the arc length of 40.8 of the whole curve.

Figure 26(a) gives the curvature diagram of the polynomial curve and the
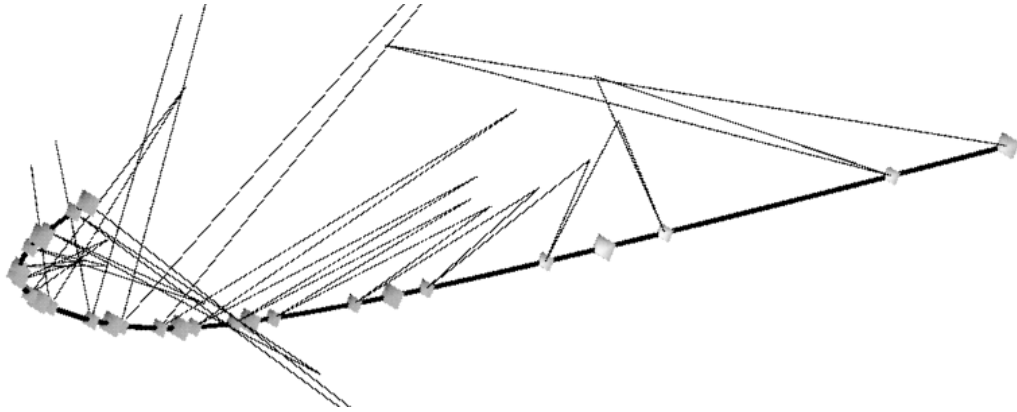
27

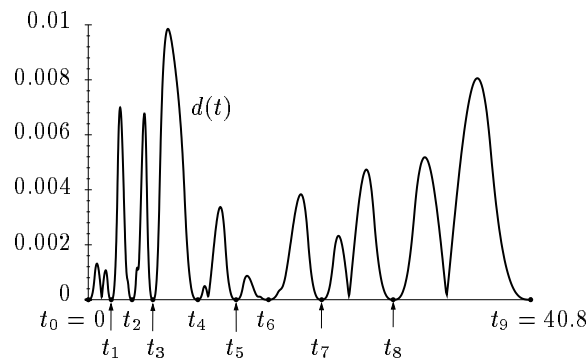Figure 24: Approximation with nine triarc segments (19 arc segments)

Figure 25: Approximation error $d(t)$ for the arc spline approximation of Figure 24

piecewise constant curvature function of the arc spline. Figure 26(b) shows the torsion function $\tau(t)$ and the torsion impulses of the approximating arc spline.

11 Future Research

Our Hermite-type approximation algorithms with cone spline surfaces are based on data coming from a developable surface. In applications, however, one will usually have discrete data (e.g. a point cloud) which should be approximated by a developable surface. Also a (closed) curve might be given as boundary curve of the surface.

It will be a topic of future research to directly compute a cone spline surface from such data and omit the intermediate stage of finding a developable surface to the data first, and approximating it with a cone spline surface afterwards.
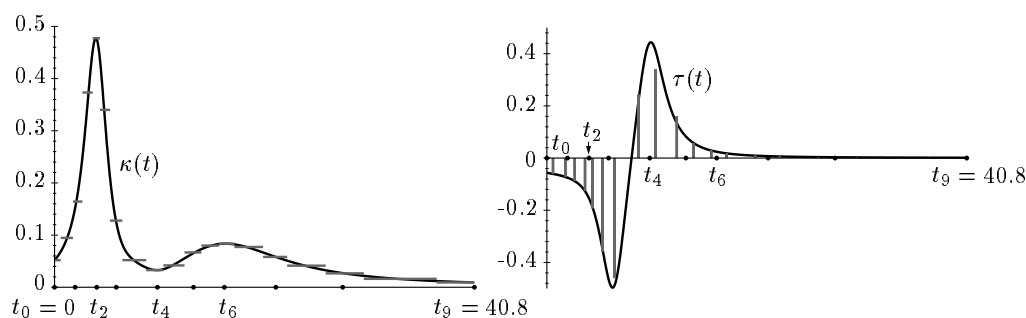
28

Figure 26: (a) Curvature and (b) torsion diagrams of a polynomial curve $\mathbf{g}(t)$ and its arc spline approximation

Acknowledgements

References

Bolton, K.M. (1975), Biarc curves, Computer-Aided Design 7, 89–92.

Fuhs, W. and Stachel, H. (1988), Circular pipe connections, Computers & Graphics 12, 53–57.

Hoschek, J. (1992), Circular splines, Computer-Aided Design 24, 611–618.

Kruppa, E. (1957), Analytische und konstruktive Differentialgeometrie, Springer-Verlag, Wien.

Leopoldseder, S. and Pottmann, H. (1998), Approximation of developable surfaces with cone spline surfaces, Computer-Aided Design 30, 571–582.

Leopoldseder, S. (1998), Cone spline surfaces and spatial arc splines, Dissertation, TU Wien.

Leopoldseder, S. (2000), Cone spline surfaces and spatial arc splines — a sphere geometric approach, Technical report 77, Institut für Geometrie, TU Wien, 2000.

Mancewicz, M.J. and Frey, W.H. (1992), Developable surfaces: properties, representations and methods of design, General Motors R&D Publication 7637.

Meek, D.S. and Walton, D.J. (1995), Approximating smooth planar curves by arc splines, J. Computational and Applied Mathematics 59, 221–231.

29

Meek, D.S. and Walton, D.J. (1996), Planar osculating arc splines, Computer Aided Geometric Design 13, 653–671.

Nutbourne, A.W. and Martin, R.R. (1988), Differential Geometry Applied to Curve and Surface Design. Vol. 1: Foundations, Ellis Horwood.

Piegl, L. (1986), Curve fitting algorithm for rough cutting, Computer-Aided Design 18, 79–82.

Pottmann, H., Chen, H.-Y. and Lee, I.-K. (1998), Approximation by profile surfaces, in: The Mathematics of Surfaces VIII, A. Ball et al., eds, Information Geometers Ltd.

Randrup, T. (1996), Design and engineering of double-curved ship surfaces, ECMI Newsletter 20, 18–20.

Sabin, M.A. (1977), The use of piecewise forms for the numerical representation of shape, Report 60, Computer and Automation Institute, Hungarian Academy of Sciences.

Sharrock, T.J. (1986), Biarcs in three dimensions, in: Mathematics of Surfaces II, R.R. Martin, ed., Oxford University Press.