13.472J/1.128J/2.158J/16.940J COMPUTATIONAL GEOMETRY

Lecture 8

N. M. Patrikalakis

Massachusetts Institute of Technology
Cambridge, MA 02139-4307, USA

Contents

Reading in the Textbook

- Chapter 11, pp. 353 - 365

Lecture 8

Fitting, Fairing and Generalized Cylinders

8.1 Least Squares Method of Curve Fitting

Example problem

Given N points $\mathbf{P}_i,\ i = 1, 2, ..., N\ (N \geq 4)$, construct an approximating cubic Bézier curve that interpolates $\mathbf{P}_1$ and $\mathbf{P}_N$ (end points).

Solution

1. *Parametrization by chord-length method*
   Let

$$\begin{aligned} \hat{u}_1 &= 0; \\ \hat{u}_{i+1} &= \hat{u}_i + d_{i+1},\ i = 1, 2, ..., N-1 \end{aligned} \tag{8.1}$$

   where $d_{i+1} = |\mathbf{P}_{i+1} - \mathbf{P}_i|$ is the chord length between two consecutive points. The overall chord length is

$$d = \sum_{i=2}^{N} d_i \tag{8.2}$$

   The parametric value associated with point $\mathbf{P}_i$

$$u_i = \hat{u}_i / d \tag{8.3}$$

   which is normalized as $u_i \in [0, 1]$ with $u_1 = 0$ and $u_N = 1$.

2. *Linear equations*
   A cubic Bézier curve is defined as

$$\mathbf{Q}(u) = \sum_{i=0}^{3} \mathbf{Q}_i B_{i,3}(u),\ 0 \leq u \leq 1 \tag{8.4}$$

   where $B_{i,3}(u)$ are the cubic Bernstein polynomials.

Obviously, the boundary conditions require $\mathbf{Q}_0 = \mathbf{P}_1, \mathbf{Q}_3 = \mathbf{P}_N$. The problem is then represented as a linear system with $N-2$ equations and 2 unknowns:

$$\begin{aligned} \sum_{i=1}^{2} \mathbf{Q}_i B_{i,3}(u_j) &= \mathbf{P}_j - \mathbf{P}_1 B_{0,3}(u_j) - \mathbf{P}_N B_{3,3}(u_j) \\ &= \mathbf{L}_j, \quad j = 2, 3, ..., N-1 \end{aligned} \tag{8.5}$$

or in matrix form

$$\mathbf{B}_{(N-2)\times 2} \cdot \mathbf{q}_{2\times 1} = \mathbf{l}_{(N-2)\times 1} \tag{8.6}$$

3. *Least Squares Method*
   Define the mean square error as

$$E^2 = |\mathbf{B} \cdot \mathbf{q} - \mathbf{l}|^2 \tag{8.7}$$

then

$$\begin{aligned} E^2 &= (\mathbf{B} \cdot \mathbf{q} - \mathbf{l})^T (\mathbf{B} \cdot \mathbf{q} - \mathbf{l}) \\ &= \mathbf{q}^T \mathbf{B}^T \mathbf{B} \mathbf{q} - 2\mathbf{q}^T \mathbf{B}^T \mathbf{l} + \mathbf{l}^T \mathbf{l} \end{aligned} \tag{8.8}$$

is a function of $\mathbf{q}$ and is minimized if we set

$$\frac{\partial E^2}{\partial \mathbf{q}} = 0 \quad \Rightarrow \quad \mathbf{B}^T \mathbf{B} \mathbf{q} - \mathbf{B}^T \mathbf{l} = 0 \tag{8.9}$$

$$\Rightarrow \quad \mathbf{B}^T \mathbf{B} \mathbf{q} = \mathbf{B}^T \mathbf{l} \qquad \text{(normal equations)} \tag{8.10}$$

$$\Rightarrow \quad \mathbf{q} = (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \mathbf{l} \quad \text{(formal solution)} \tag{8.11}$$

The extension to fitting with B-splines is similarly formulated.

Notes:

1. The choice of internal knots in the B-spline basis should reflect any knowledge of derivative discontinuity in the data, as shown in Figure 8.1.

2. Greater density of knots is needed in rapidly changing parts of the shape.

3. NAG routines for approximate fitting of cubic B-splines [9]

   (a) Curves: E02BAF

   (b) Surfaces: E02DAF & E02ZAF

4. NAG routines on least square problems provide more flexibility.

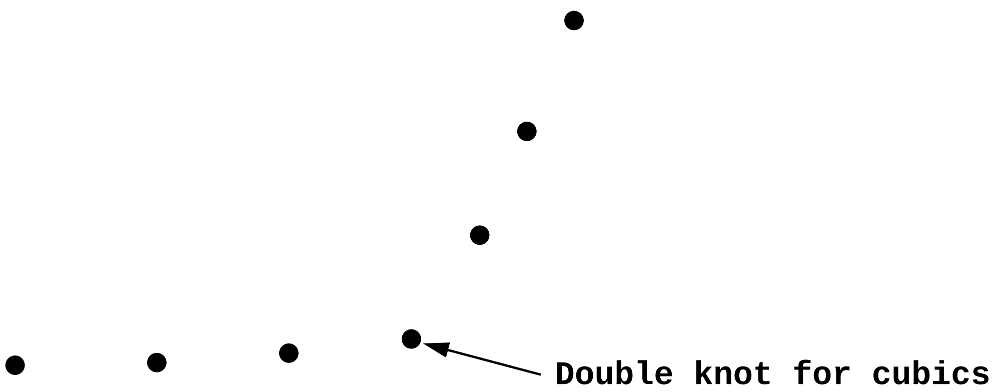


Figure 8.1: Set of data reflecting a possible discontinuity of tangent vector.

8.2 Fairing of Curves and Surfaces

8.2.1 Properties and Definition

Motivation:

1. Spline curves resulting from
   - (a) interpolation of points;
   - (b) manipulation of polygon, usually need fairing.
2. Screen plots (small resolution) are misleading concerning curve quality.
3. Full scale plots.
4. Curvature plots are useful as they allow isolation of problem areas on raster devices.

Properties of fair curves: [3, 4]

1. Curvature continuity (C^2).
2. Curvature is almost piecewise linear with as few spans as possible.

For cubics with simple knots, property (1) is automatically satisfied. If $\mathbf{R}(t)$ is reasonably parametrized, $|\mathbf{R}'(t)|$ is constant, and the curvature

$$\kappa(s) = \frac{|\mathbf{R}'(t) \times \mathbf{R}''(t)|}{|\mathbf{R}'(t)|^3} \simeq |\mathbf{R}''(t)| \tag{8.12}$$

Property (2) thus requires that $|\mathbf{R}''(t)|$ be almost piecewise linear. That means that $\mathbf{R}'''(t)$ needs to be constant, which leads to the following definition of fairness.

Definition: $\mathbf{Q}, \mathbf{R}$ are two C^2 cubic splines in $t \in [a, b]$. $\mathbf{Q}$ is fairer than $\mathbf{R}$ if for $r \in [a, b]$,

$$[\mathbf{Q}'''(r^+) - \mathbf{Q}'''(r^-)]^2 \leq [\mathbf{R}'''(r^+) - \mathbf{R}'''(r^-)]^2 \tag{8.13}$$

If r is a knot at which interpolation of data occurs, the above expression means that reducing “shear” forces from supports increases the fairness of splines, if we consider geometric splines as approximations of physical splines.

8.2.2 Curve Interrogation

We interrogate the fairness of a spline curve by looking into the planar projection of their curvature. The signed curvature is defined as

$$\kappa(u) = \frac{x''y' - x'y''}{(x'^2 + y'^2)^{\frac{3}{2}}} \tag{8.14}$$

and it is easy to point out the changes of sign (inflections). The curve is fair if a plot of $\kappa(u)$, which is made up of a few monotone segments, is continuous. The aim of the fairing process is to locate the places of maximum discontinuity of $\kappa'(u)$ and fair those places (see figure (8.2)).

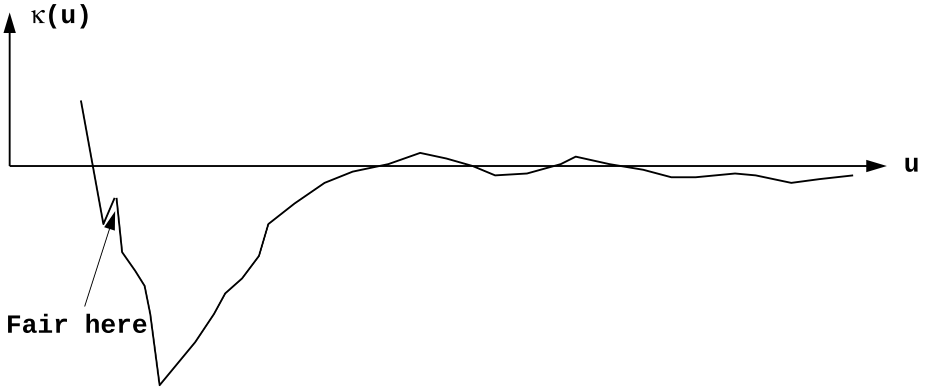


Figure 8.2: Plot of curvature $\kappa(u)$ along curve as a function of u.

8.2.3 Fairing Methods

Assume that we have a spline curve obtained by interpolation of measured data. The curve generally has unwanted behavior. Curve fairing eliminates imperfections by changing data within a measurement tolerance.

1. *Kjellander's Method* [7]

   Procedure

   (a) Obtain data points $\mathbf{P}_j(t_j), 1 \leq j \leq N$.

   (b) Fit the points with a spline $\mathbf{R}(t)$.

   (c) Find the knot where $max|\mathbf{R}'''(t_j^+) - \mathbf{R}'''(t_j^-)|$ occurs and attempt fairing at $\mathbf{P}_J$ where $j = J$ corresponds to the worst jump.

   (d) Using Hermite or Bézier, construct cubic interpolation $\mathbf{Q}(t)$, which interpolates

$$\begin{aligned} \mathbf{R}(t_{J-1}) &= \mathbf{P}_{J-1} && (8.15)\\ \mathbf{R}(t_{J+1}) &= \mathbf{P}_{J+1} && (8.16)\\ \mathbf{R}'(t_{J-1}) &\quad , \quad \mathbf{R}'(t_{J+1}) && (8.17) \end{aligned}$$

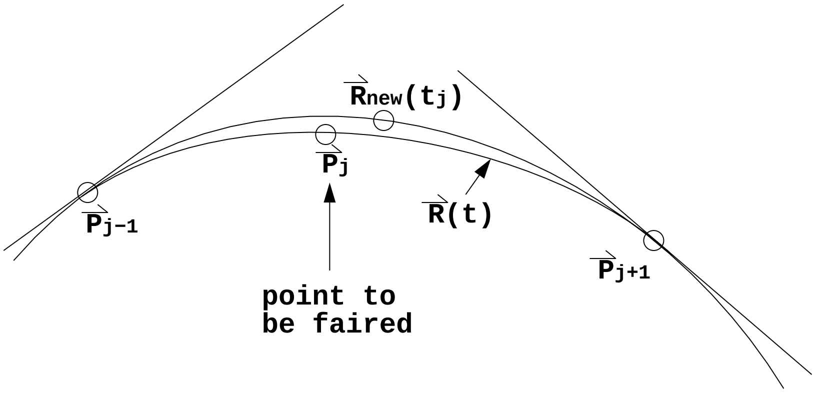


Figure 8.3: Kjellander's fairing method

(e) Determine new curve

$$\mathbf{R}_{new}(t) = \begin{cases} \mathbf{R}(t) & \text{if } t \notin [t_{J-1}, t_{J+1}] \\ \mathbf{Q}(t) & \text{if } t \in [t_{J-1}, t_{J+1}] \end{cases} \tag{8.18}$$

(f) Compute $\mathbf{R}_{new}(t_J)$. Notice that $\mathbf{R}_{new}(t)$ is infinitely differentiable there.

(g) Construct a spline curve on $\mathbf{P}_1, \cdots, \mathbf{P}_{J-1}, \mathbf{R}_{new}(t_J), \mathbf{P}_{J+1}, \cdots, \mathbf{P}_N$.

(h) The resulting curve is usually fairer at t_J.

Disadvantages

- Global scheme.
- Repeated interpolation.

2. *Farin's Method* [4]

Recall Boehm's knot insertion method

$$\sum_{i=0}^{n} \mathbf{P}_i N_{i,4}(t) = \sum_{i=0}^{n+1} \hat{\mathbf{P}}_i \hat{N}_{i,4}(t) \tag{8.19}$$

$$[t_0, t_1, \cdots] \Rightarrow \left[t_0, \cdots, t_l, \hat{t}, t_{l+1}, \cdots\right] \tag{8.20}$$

and the control points are

$$\hat{\mathbf{P}}_i = \alpha_i \mathbf{P}_i + (1 - \alpha_i)\mathbf{P}_{i-1} \tag{8.21}$$

where

$$\alpha_i = \begin{cases} 1 & 0 \le i \le l-3 \\ 0 & l+1 \le i \le n+1 \\ \frac{\hat{t}-t_i}{t_{i+3}-t_i} & l-2 \le i \le l \end{cases} \tag{8.22}$$

Hence

$$\begin{aligned} \hat{\mathbf{P}}_i &= \mathbf{P}_i & 0 \le i \le l-3 \\ \hat{\mathbf{P}}_i &= \mathbf{P}_{i-1} & l+1 \le i \le n+1 \\ \hat{\mathbf{P}}_i &= \alpha_i \mathbf{P}_i + (1-\alpha_i)\mathbf{P}_{i-1} & l-2 \le i \le l \end{aligned} \tag{8.23}$$

- Idea of Farin's method

(a) Remove a knot first to make curve infinitely differentiable at that location such that the curve is fairer now in that area.

(b) Insert the removed knot back so as to have the same knot vector (if needed; not usually necessary).

- Knot removal

Knot removal is an inverse process of knot insertion. From Equations (8.23), we have

$$\begin{aligned} \mathbf{P}_i &= \hat{\mathbf{P}}_i, \quad i = 0, 1, \cdots, l-3 \\ \alpha_i \mathbf{P}_i + (1-\alpha_i)\mathbf{P}_{i-1} &= \hat{\mathbf{P}}_i, \quad i = l-2, l-1, l \\ \mathbf{P}_i &= \hat{\mathbf{P}}_{i+1}, \quad i = l, l+1, \cdots, n \end{aligned} \tag{8.24}$$

Here we have $n+2$ equations and $n+1$ unknowns, therefore, an approximate solution can be obtained by the least squares method.

A sample solution using Farin's method:

$$\begin{aligned} \mathbf{P}_i &= \hat{\mathbf{P}}_i, \quad i = 0, 1, \cdots, l-3 \\ \mathbf{P}_i &= \hat{\mathbf{P}}_{i+1}, \quad i = l, l+1, \cdots, n \end{aligned} \tag{8.25}$$

and from the least squares method, we have the following equations for $\mathbf{P}_{l-2}, \mathbf{P}_{l-1}$

$$\begin{pmatrix} \alpha_{l-2} & 0 \\ 1-\alpha_{l-1} & \alpha_{l-1} \\ 0 & 1-\alpha_l \end{pmatrix} \begin{pmatrix} \mathbf{P}_{l-2} \\ \mathbf{P}_{l-1} \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{P}}_{l-2} - (1-\alpha_{l-2})\hat{\mathbf{P}}_{l-3} \\ \hat{\mathbf{P}}_{l-1} \\ \hat{\mathbf{P}}_l - \alpha_l \hat{\mathbf{P}}_{l+1} \end{pmatrix} \tag{8.26}$$

or in the matrix form

$$\mathbf{A} \cdot \mathbf{p} = \mathbf{f} \Rightarrow \mathbf{A}^T\mathbf{A}\mathbf{p} = \mathbf{A}^T\mathbf{f} \tag{8.27}$$

which yields

$$\mathbf{p} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{f} \tag{8.28}$$

This should be followed by knot insertion to complete the fairing process (if necessary).

- Knot insertion

$$\mathbf{T} = [t_0, \cdots, t_l, \hat{t}, t_{l+1}, \cdots] = [T_0, \cdots, T_l, T_{l+1}, T_{l+2}, \cdots] \tag{8.29}$$

where the removed knot $\hat{t}$ is inserted as the knot T_{l+1}. Hence the control points of the faired curve can be determined by the knot insertion method (Equation 8.23), where

$$\begin{aligned} \alpha_l &= \frac{\hat{t} - t_l}{t_{l+3} - t_l} = \frac{T_{l+1} - T_l}{T_{l+4} - T_l} \\ \alpha_{l-1} &= \frac{\hat{t} - t_{l-1}}{t_{l+2} - t_{l-1}} = \frac{T_{l+1} - T_{l-1}}{T_{l+3} - T_{l-1}} \\ \alpha_{l-2} &= \frac{\hat{t} - t_{l-2}}{t_{l+1} - t_{l-2}} = \frac{T_{l+1} - T_{l-2}}{T_{l+2} - T_{l-2}} \end{aligned} \tag{8.30}$$

If we remove all or many of the knots as in Figure 8.5, the other constraints (such as deviation) will dominate the problem.

Let the curve obtained by either interpolation or approximation be

$$\mathbf{R}(t) = \sum_{i=0}^{n} \mathbf{R}_i N_{i,4}(t) \tag{8.31}$$

and the curve after knot removal be

$$\mathbf{R}_0(t) = \sum_{i=0}^{m} \tilde{\mathbf{Q}}_i \tilde{N}_{i,4}(t) \tag{8.32}$$

where $m < n$. Then, $(n - m)$ knots removed before are inserted to make $\mathbf{R}_0(t)$ have the same knot vector as $\mathbf{R}(t)$, such that

$$\mathbf{R}_0(t) = \sum_{i=0}^{n} \mathbf{Q}_i N_{i,4}(t) \tag{8.33}$$

where $\mathbf{Q}_i, 0 \leq i \leq n$, are new control points determined by knot insertion, and $N_{i,4}(t)$ are B-spline basis function over the same knot vector as that of $\mathbf{R}(t)$.

The deviation of two B-spline curves

$$\begin{aligned}
|\mathbf{R}(t) - \mathbf{R}_0(t)| &= |\sum_{i=0}^{n}(\mathbf{R}_i - \mathbf{Q}_i) N_{i,4}(t)| && (8.34)\\
&\leq \max_i |\mathbf{R}_i - \mathbf{Q}_i| \sum_{i=0}^{n} N_{i,4}(t) && (8.35)\\
&= \max_i |\mathbf{R}_i - \mathbf{Q}_i| && (8.36)
\end{aligned}$$

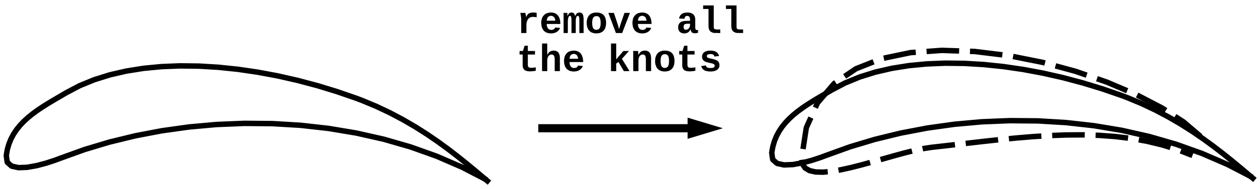


Figure 8.5: Deviation of fair curve.

8.2.4 Surface Fairing

- *Differential Geometry Review*

  Let κ_1, κ_2 be principal curvatures, then

$$\begin{aligned} \text{Gaussian curvature } G &= \kappa_1\kappa_2 && (8.37)\\ \text{mean curvature } H &= \frac{1}{2}(\kappa_1 + \kappa_2) && (8.38)\\ \text{absolute curvature } K &= |\kappa_1| + |\kappa_2| && (8.39) \end{aligned}$$

Local Taylor expansion is

$$z = \frac{1}{2}(\kappa_1 x^2 + \kappa_2 y^2) + h.o.t. \tag{8.40}$$

1. Elliptic case: $G = \kappa_1\kappa_2 > 0$, see Figure 8.6(a).
   - convex surface
   - one side of tangent plane
   - no inflection
2. Hyperbolic case: $G = \kappa_1\kappa_2 < 0$, see Figure 8.6(b).
   - non-convex surface (locally)
   - intersection with tangent plane
   - surface inflection

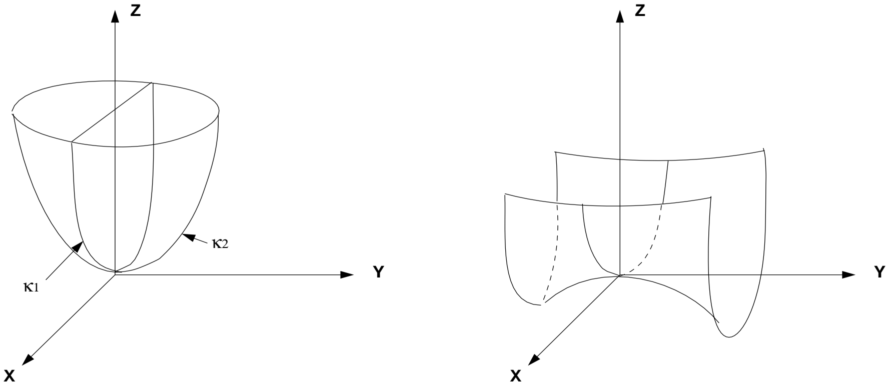


Figure 8.6: (a) Elliptic case; (b) Hyperbolic case.

Definition 1 Surface inflection at a point P exists *iff* the surface crosses the tangent plane at P.

If $G < 0$, surface inflection at P.

If $G = 0$ in a region (e.g., a "cylinder"), then surface inflection exists in the region if H changes sign in the region (not one point).

Definition 2 Surface curve inflection exists at a point P of a surface *iff* a planar surface curve through P changes sign of its curvature at P.

Surface curve inflection $\Rightarrow$ Surface inflection

- *Surface Interrogation*

1. Gaussian curvature is not useful in "cylinders" where $G = 0$.
2. Mean curvature is zero for minimal surfaces where $\kappa_1 = \kappa_2$.
3. Absolute curvature does not have such problems. Its curvature plots are useful in studying surface imperfections (e.g., small oscillations).
4. Reflection lines for parallel line light source (on raster screen, mark points where normal passes through source) are useful for global imperfection detection.

Surface fairing may be performed by fairing auxiliary curves defined for each of the columns and rows of the control polyhedron.

Generalized Cylinders

8.3 Generalized Cylinders: Motivation and Definitions

Generalized cylinders [12, 13, 15, 8] or sweeps provide greater generality for shape representation than tensor product surfaces. A generalized cylinder is a representation of an elongated object viewed as having a main axis (*directrix or spine*) and a smoothly varying cross section (*generatrix*), as defined in Figure 8.7. Both directrix and generatrix can be open or closed (periodic) curves.

8.3.1 Applications

Some of the more common applications of generalized cylinders are

- Representation of measured data (e.g. CAT scans, deformed solids).
- Representation of manufacturing processes.
- Representation of blends.
- Object recognition and scene interpretation in robotics and computer vision.
- Representation of human and animal shapes.

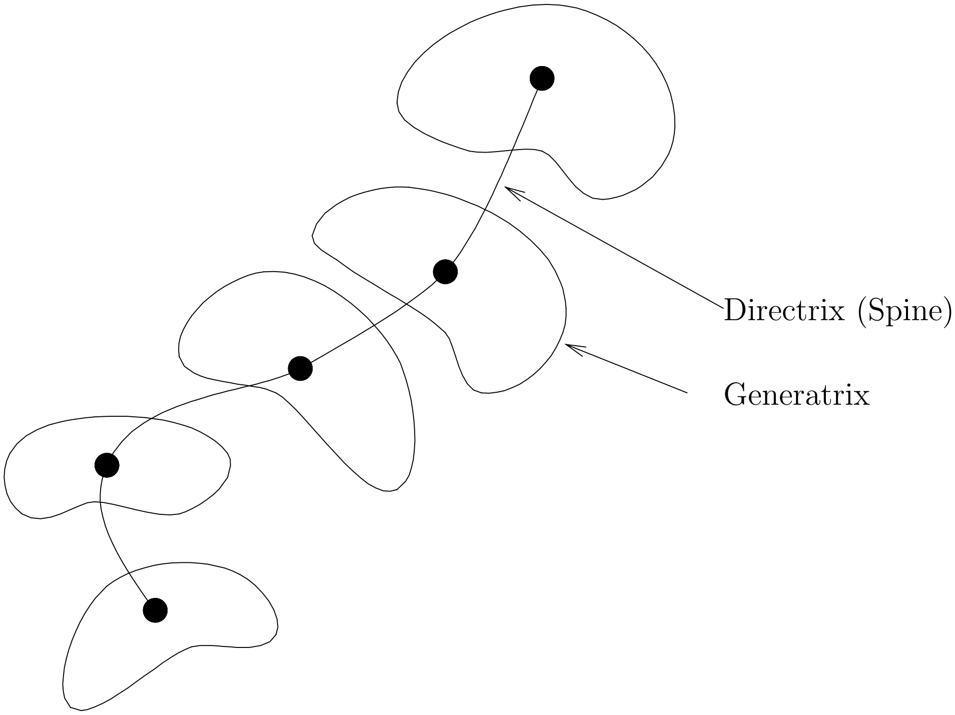


Figure 8.7: Generalized cylinder.

8.3.2 Definition

- Given:
  1. A bounded 3-D curve serving as spine.
  2. A cross-sectional plane swept along the spine perpendicular to it so that the spine passes through the origin of the 2-D coordinate system on the plane.
  3. A cross-sectional curve on the cross-section plane defined locally in the cross-section coordinate system, where the size and shape of the curve may vary with the parameter along the spine curve.
- The surface swept by the curve is a generalized cylinder.
- Some examples of generalized cylinders and the resulting surfaces are:
  1. Spine = straight line, generatrix = circle $\Longrightarrow$ CYLINDER
  2. Spine = circle, generatrix = circle $\Longrightarrow$ TORUS
  3. Spine = straight line segment, generatrix = linearly tapering circles $\Longrightarrow$ CONE

Mathematical description of generalized cylinders:

- Directrix (spine): $\mathbf{A} = \mathbf{A}(s)$, $0 \leq s \leq 1$.

- Generatrix: $\mathbf{C} = \mathbf{C}(t; s) = [x(t; s), y(t; s), 0]$, $0 \leq t \leq 1$.
- Generalized cylinder surface patch: $\mathbf{R}(s, t) = \mathbf{A}(s) + x(t; s)\mathbf{X}(s) + y(t; s)\mathbf{Y}(s)$, where $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$ are orthogonal 3-D unit vectors with the $\mathbf{Z}$ tangent to $\mathbf{A}(s)$, i.e. $\mathbf{Z}(s) = \frac{\mathbf{A}'(s)}{|\mathbf{A}'(s)|}$

As an example, $\mathbf{X}(s)$, $\mathbf{Y}(s)$ could be chosen equal to the normal and binormal vectors of the spine curve $\mathbf{A}(s)$ or by rotation of those by some angle, see Figure 8.8.

Problems with generalized cylinder representation

When $\mathbf{A}(s)$ is straight line, $\mathbf{X}(s)$, $\mathbf{Y}(s)$, $\mathbf{Z}(s)$ should be defined independent of the Frenet trihedron, eg. using $\mathbf{X}(s)$, $\mathbf{Y}(s)$ as constants.

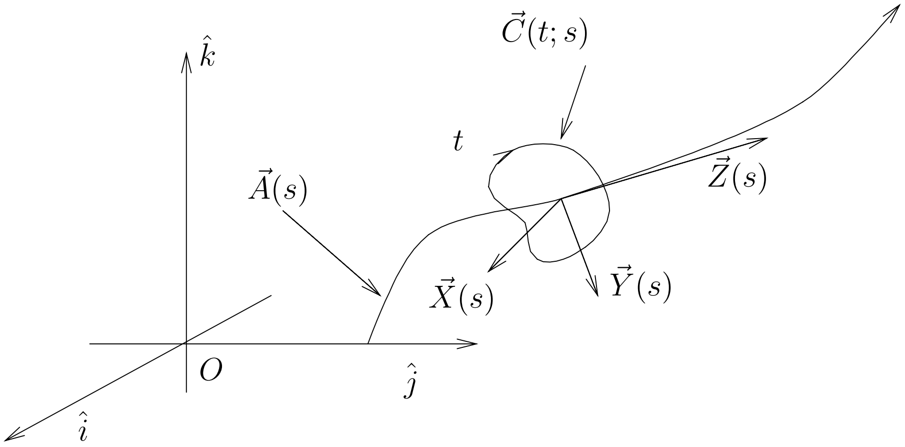


Figure 8.8: Components of a generalized cylinder.

Examples:

a. **Generalized Cylinders with B-Spline Spine and Generatrix Curves**

$$\begin{aligned}\mathbf{A}(s) &= \sum_{i=0}^{k}\mathbf{A}_i N_{i,K}(s)\\ \mathbf{C}(t;s) &= \sum_{i=0}^{l}\sum_{j=0}^{q}\mathbf{C}_{ij}N_{i,L}(t)N_{j,Q}(s)\\ &= [x(t;s), y(t;s), 0]\end{aligned}$$

where $N_{i,K}(s), N_{i,L}(t)$ and $N_{j,Q}(s)$ are B-spline basis functions or the Bernstein specializations.

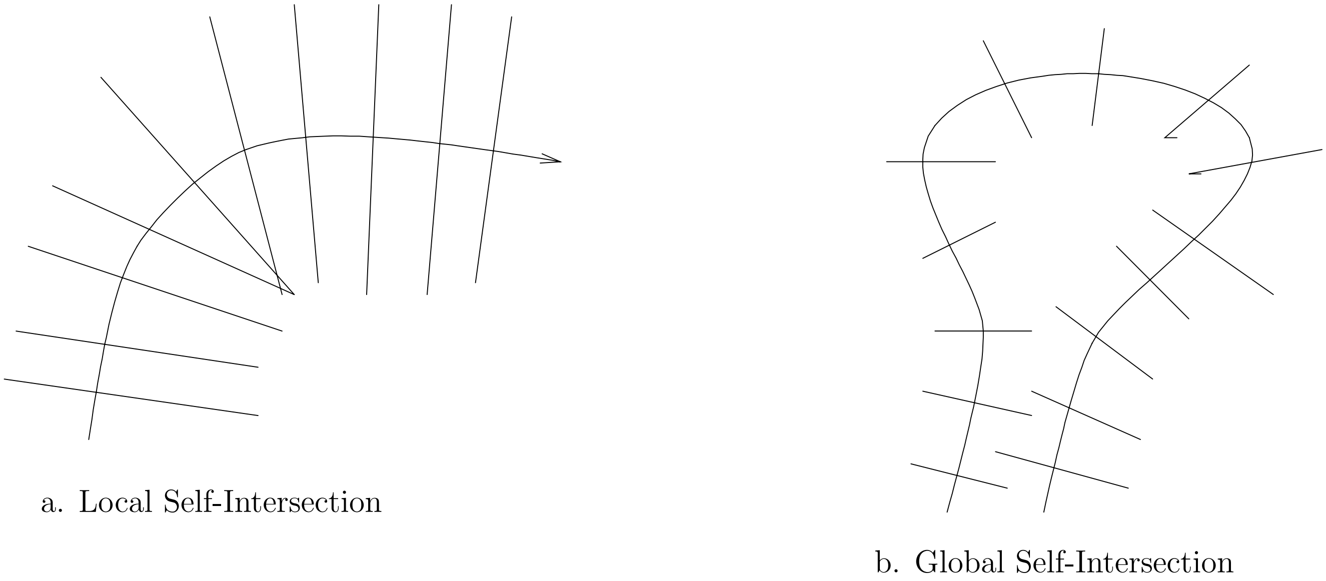
b. **Pipe Surfaces** (See also Section 11.6 of textbook [11])

When the generatrix is a circle, the resulting generalized cylinder is a pipe surface. The pipe surface $\mathbf{P_C}(r)$ with radius r can be parametrized using the Frenet trihedron $(\mathbf{t}(t), \mathbf{n}(t), \mathbf{b}(t))$ of the spine curve $\mathbf{C}(t)$ as follows:

$$\mathbf{P}(t,\theta) = \mathbf{C}(t) + r[\cos\theta\mathbf{n}(t) + \sin\theta\mathbf{b}(t)] \tag{8.41}$$

where $t \in [0,1]$ and $\theta \in [0, 2\pi]$.

8.4 Degeneracies of Generalized Cylinders


a. Local Self-Intersection

b. Global Self-Intersection

Figure 8.9: Types of self-intersection of generalized cylinders.

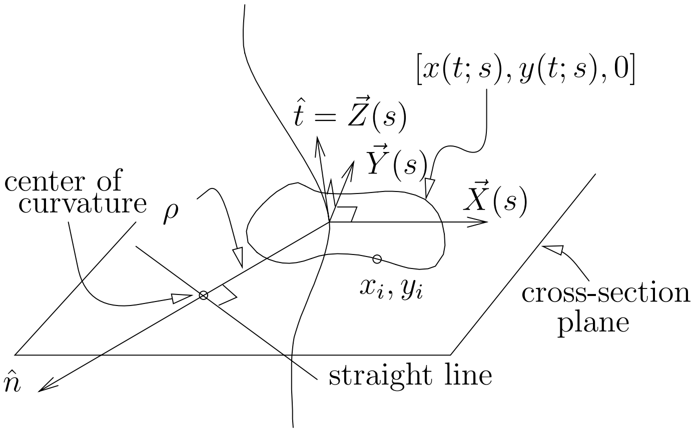


Figure 8.10: Criterion to avoid local self-intersection of generalized cylinders.

There are two types of the degeneracies of generalized cylinders illustrated in Figure 8.9, namely local self-intersection and global self-intersection. A condition to avoid local self-intersection of generalized cylinders is illustrated in Figure 8.10. The condition is

$$max_t(x^2 + y^2) \leq \rho^2(s)$$

for all s, where $\rho(s)$ is the radius of curvature of the spine.

As a special case, we consider local self-intersection of pipe surfaces (see also Maekawa *et al.* (1998) for details). The partial derivative of the pipe surface with respect to t is given by

$$\mathbf{P}_t(t,\theta) = \mathbf{C}'(t) + r[\cos\theta\mathbf{n}'(t) + \sin\theta\mathbf{b}'(t)]. \tag{8.42}$$

Equation (8.42) can be rewritten using the Frenet formulae ($\mathbf{n}'(t) = |\mathbf{C}'(t)|(-\kappa(t)\mathbf{t} + \tau(t)\mathbf{b})$, $\mathbf{b}'(t) = -|\mathbf{C}'(t)|\tau(t)\mathbf{n}$) as

$$\mathbf{P}_t(t,\theta) = |\mathbf{C}'(t)|(1-\kappa(t)r\cos\theta)\mathbf{t} - r|\mathbf{C}'(t)|\sin\theta\tau(t)\mathbf{n} + r|\mathbf{C}'(t)|\cos\theta\tau(t)\mathbf{b} \tag{8.43}$$

where $\kappa(t)$ and $\tau(t)$ are the curvature and torsion of the spine curve. Similarly we can derive $\mathbf{P}_\theta$ as

$$\mathbf{P}_\theta(t,\theta) = r[-\sin\theta\mathbf{n}(t) + \cos\theta\mathbf{b}(t)]. \tag{8.44}$$

The surface normal of the pipe surface can be obtained by taking the cross product of equations (8.43) and (8.44) yielding

$$\mathbf{P}_t \times \mathbf{P}_\theta = -|\mathbf{C}'(t)|r[1-\kappa(t)r\cos\theta][\sin\theta\mathbf{b}(t) + \cos\theta\mathbf{n}(t)]. \tag{8.45}$$

It is easy to observe that the pipe surface becomes singular (the normal vector vanishes) when $1-\kappa(t)r\cos\theta = 0$. Since $\cos\theta$ varies between -1 and 1, there will be no local self-intersection if $\kappa(t)r < 1$. Therefore, to avoid local self-intersection we need to find the largest curvature κ_{max} of the spine curve and set the radius of the pipe surface such that $r < 1/\kappa_{max}$. Figure 8.11 shows an example of local self-intersection.

The curvature $\kappa(t)$ of a space curve $\mathbf{C}(t)$, is given by

$$\kappa(t) = \frac{|\mathbf{C}'(t) \times \mathbf{C}''(t)|}{|\mathbf{C}'(t)|^3}. \tag{8.46}$$

Thus, to find the largest curvature κ_{max} we need to locate the critical points of $\kappa(t)$, i.e. solve the equation $\kappa'(t) = 0$, and decide whether they are local maxima. Then we compare these local maxima with the curvature at the end points, i.e. $\kappa(0)$ and $\kappa(1)$, and obtain the global largest curvature. This problem can be solved by elementary calculus. If the spine curve is given by a rational Bézier curve, equation $\kappa'(t) = 0$ reduces to a single univariate nonlinear polynomial equation. In the case where the spine curve is a rational B-spline, we can extract the rational Bézier segments by knot insertion. Cho *et al.* [1] describe in detail how to obtain $\kappa'(t) = 0$ for integral Bézier curves.

Global self-intersection of a pipe surface involves the following types of intersections:

1. *End circle to end circle*: Two end circles of the pipe surface touch each other, see Figure 8.12.

2. *Body to body*: Two different body portions of the pipe surface touch each other, see Figure 8.13.

3. *End circle to body*: One of the end circles touches the body, see Figure 8.14.

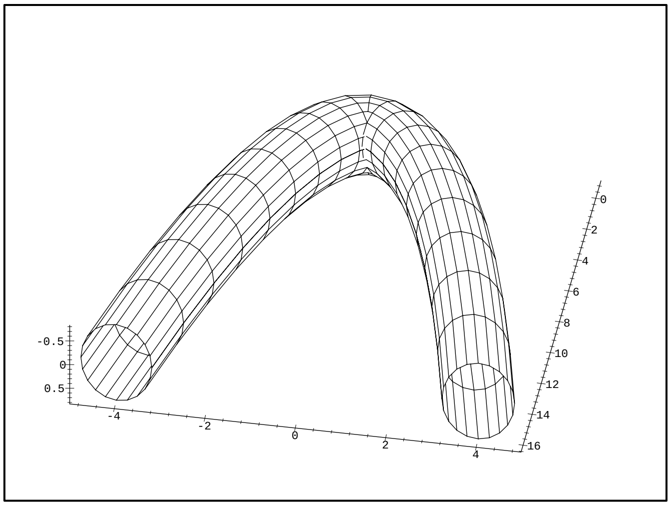


Figure 8.11: Local self-intersection

The theory of intersections and nonlinear solvers is needed to handle these global intersection problems and we will discuss these later.

Let κ_{max} be the maximum curvature of the spine curve, and r_{ee}, r_{bb}, r_{eb} be the maximum possible upper limit radius of the pipe surface such that it does not globally self-intersect between end circle to end circle, body to body and end circle to body of the pipe surface, respectively. Then we have

Theorem *Let $\mathbf{p}(r)$ be the pipe surface with spine curve $\mathbf{c}(t)$ and radius r. Then $\mathbf{p}(r)$ is nonsingular if and only if $r < \delta = min\{1/\kappa_{max}, r_{ee}, r_{bb}, r_{eb}\}$.*

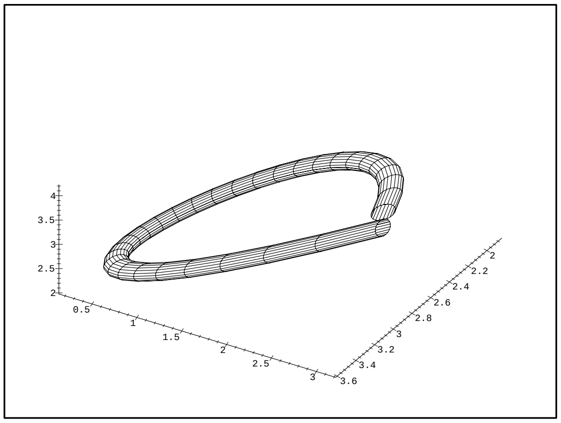


Figure 8.12: End circle to end circle global self-intersection

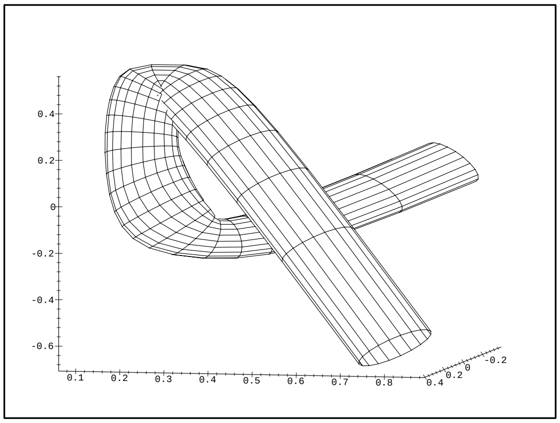


Figure 8.13: Body to body tangential intersection and local self-intersection

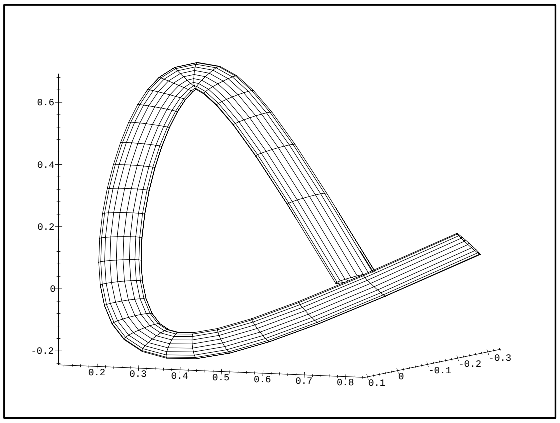


Figure 8.14: End circle to body tangential global and local self-intersections

8.5 Properties of Generalized Cylinders

1. Unit normal vector

$$\hat{n} = \frac{\mathbf{R}_s \times \mathbf{R}_t}{|\mathbf{R}_s \times \mathbf{R}_t|}$$

2. Radius of curvature of spine– $A(s) = [x, y, z]$

$$\begin{aligned} \frac{|\mathbf{A}_s|^3}{|\mathbf{A}_s \times \mathbf{A}_{ss}|} &= \rho(s) \\ &= \frac{(x_s^2 + y_s^2 + z_s^2)^{\frac{3}{2}}}{[(x_s y_{ss} - y_s x_{ss})^2 + (y_s z_{ss} - z_s y_{ss})^2 + (z_s x_{ss} - x_s z_{ss})^2]^{\frac{1}{2}}} \end{aligned}$$

8.6 Discrete Generalized Cylinders

Useful in the interpretation of measured data. The constructive definition for discrete generalized cylinders is:

1. Define a piecewise continuous spine.
2. Obtain point measurements on cross-section curves on planes perpendicular to spine at a discrete set of points on spine.
3. Construct a local system of coordinates on each cross-section with origin on spine.
4. Interpolate each cross-section with splines and establish parametric correspondence between cross-sections, see Figure 8.15.
5. Establish an interpolation rule between cross-sections, $x(t;s_i)$, $y(t;s_i)$, $z(t;s_i)$, $\theta(s_i)$, see Figure 8.15.

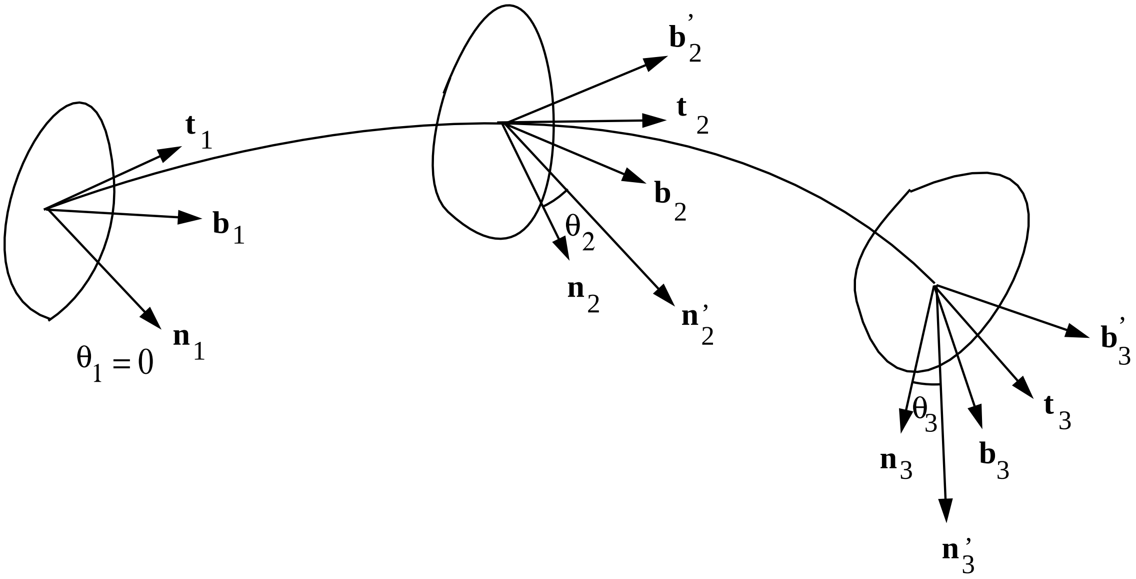


Figure 8.15: Cross sections along the spine curve

Bibliography

[1] W. Cho, T. Maekawa, and N. M. Patrikalakis. Topologically reliable approximation of composite Bézier curves. *Computer Aided Geometric Design*, 13(6):497–520, August 1996.

[2] Q. Ding and B. J. Davies. *Surface Engineering Geometry for Computer-Aided Design and Manufacture*. Ellis Horwood, Chichester, UK, 1987.

[3] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*. Academic Press, Boston, MA, 3rd edition, 1993.

[4] G. Farin, G. Rein, N. Sapidis, and A. J. Worsey. Fairing cubic B-spline curves. *Computer Aided Geometric Design*, 4(1-2):91–103, July 1987.

[5] I. D. Faux and M. J. Pratt. *Computational Geometry for Design and Manufacture*. Ellis Horwood, Chichester, England, 1981.

[6] J. Hoschek and D. Lasser. *Fundamentals of Computer Aided Geometric Design*. A. K. Peters, Wellesley, MA, 1993. Translated by L. L. Schumaker.

[7] J. A. Kjellander. Smoothing of cubic parametric splines. *Computer-Aided Design*, 15:175–179, 1983.

[8] T. Maekawa, N. M. Patrikalakis, T. Sakkalis, and G. Yu. Analysis and applications of pipe surfaces. *Computer Aided Geometric Design*, 15(5):437–458, May 1998.

[9] Numerical Algorithms Group, Oxford, England. *NAG Fortran Library Introductory Guide*, Mark 18 edition, 2000.

[10] J. Owen. *STEP: An Introduction*. Information Geometers, Winchester, UK, 1993.

[11] N. M. Patrikalakis and T. Maekawa. *Shape Interrogation for Computer Aided Design and Manufacturing*. Springer-Verlag, Heidelberg, February 2002.

[12] J. Pegna. *Variable Sweep Geometric Modeling*. PhD thesis, Stanford University, Stanford, CA, 1987.

[13] J. Pegna and D. J. Wilde. Spherical and circular blending of functional surfaces. In *Proceedings of the 7th International Conference on Offshore Mechanics and Arctic Engineering*, volume 7, pages 73–82, Houston, Texas, February 1988. New York:ASME, 1988.

[14] L. A. Piegl and W. Tiller. *The NURBS Book*. Springer, New York, 1995.

[15] U. Shani and D. H. Ballard. Splines as embeddings for generalized cylinders. *Computer Vision, Graphics and Image Processing*, 27:129–156, 1984.

[16] F. Yamaguchi. *Curves and Surfaces in Computer Aided Geometric Design*. Springer-Verlag, NY, 1988.