Smoothing of Polygonal Chains for 2D Shape Representation Using a G^2-Continuous Cubic A-Spline*

Sofía Behar[1], Jorge Estrada[2], Victoria Hernández[2], and Dionne León[2]

[1] Faculty of Mathematics and Computer Sciences, Havana University, Cuba
[2] Institute of Mathematics and Theoretical Physics, CITMA, Cuba

Abstract. We have developed a G^2-continuous cubic A-spline, suitable for smoothing polygonal chains used in 2D shape representation. The proposed A-spline scheme interpolates an ordered set of data points in the plane, as well as the direction and sense of tangent vectors associated to these points. We explicitly characterize curve families which are used to construct the A-spline sections, whose members have the required interpolating properties and possess a minimal number of inflection points. The A-spline considered here has many attractive features: it is very easy to construct, it provides us with convenient geometric control handles to locally modify the shape of the curve and the error of approximation is controllable. Furthermore, it can be rapidly displayed, even though its sections are implicitly defined algebraic curves.

Keywords: Algebraic cubic splines, polygonal chain, data interpolation and fitting, 2D shape representation.

Mathematics Subject Classification: 65D07(splines), 65D05 (interpolation), 65D17 (Computer Aided Design).

1 Introduction

Several geometry processing tasks use polygonal chains for 2D shape representation. Digital image contouring, snakes, fitting from "noisy" data, interactive shape or font design and level set methods (see for instance [5], [7], [10], [11]) are some illustrating examples. Suppose a curve is sampled within some error band of width 2 ε around the curve. Since the sampled point sequence $\mathcal{S}$ could be dense, a simplification step is often used to obtain coarser or multiresolution representations. A polygonal chain $\mathcal{C}$ approximating the points of $\mathcal{S}$ is constructed, with the property that all points in $\mathcal{S}$ are within an ε-neighborhood of the simplified polygonal chain $\mathcal{C}$.

Prior work on using algebraic curve spline in data interpolation and fitting focus on using bivariate barycentric BB-form polynomials defined on plane triangles ([1], [5],[8],[9]) Some other authors use tensor product A-splines ([2]).

* The results presented in this work were obtained with the support of a FONCI/2003 grant.

M. Lazo and A. Sanfeliu (Eds.): CIARP 2005, LNCS 3773, pp. 42–50, 2005.

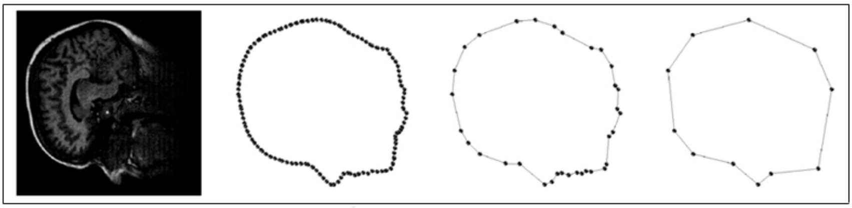

Fig. 1. Sequence of polygonal chains

These A-spline functions are easy to construct. The coefficients of the bivariate polynomial that define the curve are explicitly given. There exist convenient geometric control handles to locally modify the shape of the curve, essential for interactive curve design. Each curve section of the A-spline curve has either no inflection points if the corresponding edge is convex, or one inflection point otherwise, therefore the A-spline sections have a minimal number of inflection points. Since their degree is low, the A-spline sections can be evaluated and displayed very fast. Moreover, some of them are also ε-error controllable.

All that features make these error-bounded A-spline curves promising in the above mentioned applications, which happen to be equivalent to the interpolation and/or approximation a polygonal chain of line segments with error bounds.

Given an input polygonal chain $\mathcal{C}$, we use a cubic A-spline curve $\mathcal{A}$ to smoothly approximate the polygon by interpolating the vertices as well as the direction and sense of the given tangent vectors at the vertices. We also interpolate curvatures at the polygon vertices to achieve G^2-continuity.

The present work is a natural generalization of [5], where once the contour of digital image data has been extracted, the algorithm computes the breakpoints of the A-spline, i.e the junction points for the sections that make up the A-spline curve. Inflection points are also added to the set of junction points of the A-spline. Tangent lines at the junction points are computed using a weighted least square linear fit (fitting line), instead of the classical techniques. This G^1-continuous A-spline scheme interpolates the junction points along with the tangent directions and least-squares approximates the given data between junction points.

2 Some Notations and Preliminaries

The A-spline curve $\mathcal{A}$ discussed in this paper consists of a chain of curve sections $\mathcal{A}_i$. Each section is defined as the zero contour of a bivariate BB-polynomial of degree 3. We show that these curve sections are convex, connected and nonsingular in the interior of the regions of interest.

2.1 Derivative Data

On each vertex Q_i of the polygonal chain $\mathcal{C}$, we assume that the slope of the tangent line t_i as well as the curvature κ_i of $\mathcal{A}$ at Q_i are given. The values t_i can be estimated from the given dense sample data $\mathcal{S}$ by means of a weighted least square linear fit (fitting line) technique, such as proposed in [5], which has a better performance as the ones usually recommended in the literature (see for instance [1] or [2]). Figure 2 illustrates the performance of different methods. The direction of vector $\overrightarrow{v_i}$ may be determined by the estimated value t_i.

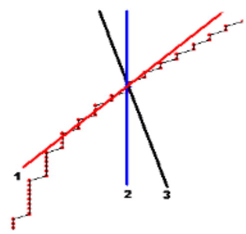

Fig. 2. Selecting the tangent vector using: 1. Fitting line, 2. Interpolation parabola, 3. Fourth degree interpolation polynomial

To compute the curvature values κ_i, we propose the following procedure. Among all (implicitly defined) plane quadratic curves $f(x, y) = 0$ passing through Q_i, such that the tangent line of f at Q_i has slope t_i, compute the quadratic curve with implicit equation $f^*(x, y) = 0$ minimizing the weighted sum

$$W_i := \sum_k \left(\frac{f(P_i^k)}{d_i^k} \right)^2$$

where P_i^k are points in $\mathcal{S}$ which are in a neighborhood of Q_i, $P_i^k \neq Q_i$, and $d_i^k := \|Q_i - P_i^k\|$. Then, set κ_i equal to the curvature of $f^*(x, y) = 0$ at Q_i. The computation of $f^*(x, y) = 0$ reduces to a linear least squares problem, hence it is not expensive.

2.2 Convexity of an Edge

Definition 1. *Given two consecutive vertices of $\mathcal{C}$, we call the edge passing through them* ***convex*** *if the associated tangent vectors point to opposite sides of the edge. Otherwise, we call the edge* ***non convex*** *(see Fig. 3).*

In the non convex case, in an analogous way as explained in [5], we insert to $\mathcal{C}$ a new intermediate vertex for the position of the inflection point and the

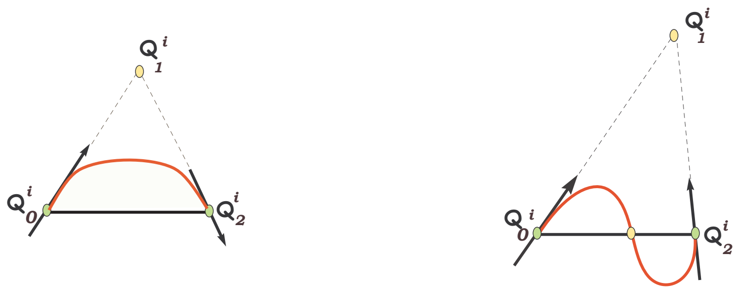

Fig. 3. Examples of convex and non convex cases

tangent line at this new vertex is computed using a weighted least square linear fit. Further, the corresponding curvature value is set equal to 0, since it happens to be an inflection point. In this way we reduce a non convex edge of $\mathcal{C}$ to the union of two consecutive convex edges.

2.3 ε-Error Controllability

Definition 2. *Given a magnitude $\varepsilon > 0$, we call an A-spline ε-**controllable** if the points of each section $\mathcal{A}_i$ are at most at distance ε to the corresponding edge $\mathcal{A}$.*

We show that the proposed A-spline scheme is ε-**controllable**. Note that if we use barycentric coordinates (u, v) with respect to a triangle, such that the edge E_i corresponds to the line $v = 0$, then $\mathcal{A}_i$ is ε-**controllable** iff $|v| \leq \varepsilon_i$, for some $0 \leq \varepsilon_i$ depending on ε and of the geometry of the triangle.

3 Polygonal Chain Approximation by Cubic A-Spline Curves

Given an ordered set of n points in the plane $\mathcal{C}$ and prescribed tangent vectors at these points, we want to construct a cubic G^2-continuous A-spline curve $\mathcal{A}$, interpolating these points, as well as the direction and sense of their prescribed tangent vectors.

3.1 Triangle Chain

Abusing of notation, let us introduce a new sequence of points Q_j^i. First, set $Q_0^i := Q_i$ and $Q_2^i := Q_{i+1}$. Each pair of consecutive points $Q_0^i, Q_2^i \in \mathcal{C}$ with their tangent directions define a triangle T_i, with vertices Q_0^i, Q_1^i, Q_2^i, where Q_1^i

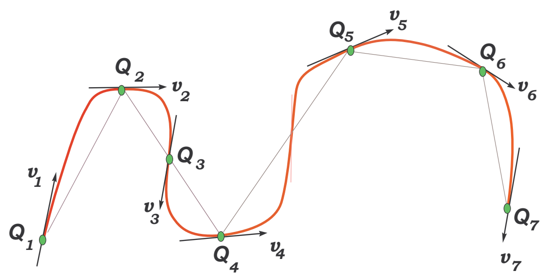

Fig. 4. Interpolating points Q_i with their prescribed tangent vectors $\overrightarrow{v_i}$

is the point of intersection of the tangent directions at Q_0^i and Q_2^i. In order to obtain a continuous curve $\mathcal{A}$, we must require that $Q_2^i = Q_0^{i+1}$ for $i = 1, ...n-1$. Additionally, to construct a closed curve, it is necessary that $Q_2^n = Q_0^1$.

3.2 G^1-Continuity

$\mathcal{A}_i$ may be written in barycentric coordinates (u, v, w), $w = 1 - u - v$ with respect to the vertices of T_i as,

$$\mathcal{A}_i : f_i(u, v) = \sum_{j=0}^{3} \sum_{k=0}^{3-j} a_{kj}^i u^k v^j w^{3-k-j} = 0 \tag{1}$$

Note that after introducing barycentric coordinates the vertex Q_0^i is transformed in the point $(1, 0)$, while the vertex Q_2^i is transformed in the point $(0, 0)$.

It is well known that $\mathcal{A}_i$ interpolates Q_0^i and Q_2^i if the coefficients $a_{0,0}^i$ and $a_{3,0}^i$ in (1) vanish. Furthermore, the tangent lines to $\mathcal{A}_i$ at Q_0^i and Q_2^i are the corresponding sides of the triangle T_i iff $a_{0,1}^i$ and $a_{2,1}^i$ vanish. Assuming that the previous restrictions on the coefficients of are satisfied, then $\mathcal{A}$ is G^1-continuous.

3.3 Explicit Expressions for $\mathcal{A}_i$

Since the section $\mathcal{A}_i$ is traced out from the initial point Q_0^i to the point Q_2^i then, according to the sense of vector $\overrightarrow{v_i}$ associated to Q_0^i we must consider two cases (see Fig. 5):

- **Inner case**: $\overrightarrow{v_i}$ points out to the halfplane containing Q_0^i.
- **Outer case**: $\overrightarrow{v_i}$ does not point out to the halfplane containing Q_0^i.

In the **inner case**, section $\mathcal{A}_i$ is the zero contour of the cubic curve with equation

$$\begin{aligned} I^i(u, v) : \quad & -v^3 + \tfrac{(1-2u_i)^3}{2u_i^3}\, uw^2 + \tfrac{(1-2u_i)^3}{2u_i^3}\, u^2 w - k_2^i \tfrac{(1-2u_i)^3}{2u_i^3}\, v^2 w \\ & - k_0^i \tfrac{(1-2u_i)^3}{2u_i^3}\, uv^2 + (k_2^i + k_0^i) \tfrac{(1-2u_i)^4}{2u_i^4}\, uvw = 0 \end{aligned}$$

(a) (b)

Fig. 5. Interpolating the sense of tangent vectors. (a) **Inner case** (b) **Outer case**.

In the **outer case**, section $\mathcal{A}_i$ is the zero contour of the cubic curve with equation

$$O^i(u,v): \quad -v^3 - \frac{(1-2u_i)^3}{2u_i^3}\, uw^2 - \frac{(1-2u_i)^3}{2u_i^3}\, u^2w + k_2^i \frac{(1-2u_i)^3}{2u_i^3}\, v^2w \\ + k_0^i \frac{(1-2u_i)^3}{2u_i^3}\, uv^2 + (k_2^i + k_0^i)\frac{(1-2u_i)^4}{2u_i^4}\, uvw = 0$$

In the next theorem we show that $\mathcal{A}_i$ is contained in the plane region Ω_i. In the **inner case**, Ω_i is the interior of the triangle $\mathcal{T}$ with vertices $(0,0),(1,0),(0,1)$ otherwise, Ω_i is equal to $\mathcal{R} = \{(u,v) : -0.5 < v < 0, \quad 0 < u < 1-u-v\}$.

Theorem 1. *The plane cubic curves $I^i(u,v), O^i(u,v)$, satisfy the following properties:*

1. *They interpolate the points Q_0^i, Q_2^i. Their tangent lines at Q_0^i and Q_2^i are the corresponding sides of T_i.*
2. *At Q_0^i they have curvature $\kappa_2^i = \frac{k_2^i \Delta_i}{(g_2^i)^3}$ and at Q_2^i have curvature $\kappa_0^i = \frac{k_0^i \Delta_i}{(g_0^i)^3}$. Here $g_j^i = \|Q_j^i - Q_1^i\|$ and Δ_i denotes the area of T_i.*
3. *Geometric handles: The curves I^i interpolate the point with barycentric coordinates $(u_i, 1-2u_i)$ while the curves O^i interpolate the point with barycentric coordinates $(\frac{u_i}{4u_i-1}, \frac{2u_i-1}{4u_i-1})$. Recall that these interpolation points lay on the line $1-2u-v=0$.*
4. *In Ω_i, I^i and O^i are non singular, connected and convex.*
5. *If $\varepsilon_i \geq 1$ then, curves I^i and O^i are ε_i-controllable. Otherwise, for $0 \leq u_i \leq \frac{1-\varepsilon_i}{2-\varepsilon_i}$, I^i are ε_i-controllable and for $0 \leq u_i \leq \frac{\varepsilon_i - 1}{3\varepsilon_i - 2}$, O^i are ε_i-controllable.*

The proof of this theorem is somewhat long and due page limitation could not be completely included. We will present some arguments:

1. See the section G^1**-continuity**.
2. See [6].

3. It is a straightforward computation.

4. For the **inner case**, see [1], [6] and [8]. For the **outer case**, the techniques used in the **inner case** do not apply, furthermore, the curves O^i have not been studied before. Considering the pencil $\mathcal{L}$ of lines passing through Q_1^i, the value of the v-coordinate of the intersection of each line $l \in \mathcal{L}$ with any of the curves O^i satisfies a cubic equation, that rewritten in BB-form permits, using range analysis such as in [4], to ensure that inside of $\mathcal{R}$, l and each curve have only one intersection point, counting multiplicity. Hence these new curves are connected and non singular inside $\mathcal{R}$. Assuming the existence of an inflection point in $\mathcal{R}$, since the curves O^i are connected and additionally they are convex in a neighborhood of Q_0^i as well as of Q_2^i, then there are at least two inflection points in $\mathcal{R}$. Thus considering the line passing through two consecutive inflexion points in $\mathcal{R}$, it is straightforward to show that this line cuts the curve at least in 4 points, but the curves are cubic, a contradiction to Bezout Theorem.

5. For each of the curves I^i, O^i, let us denote them as $f^i(u,v) = 0$, it was computed their partial derivatives with respect to the variable u, $f_u^i(u,v) = 0$ and using elimination theory, we eliminated the variable u from the system of equations $\{f^i(u,v) = 0,\ f_u^i(u,v) = 0\}$, obtaining a polynomial $p^i(v, u_i, k_0^i, k_2^i)$, such that for fixed values of the parameters (u_i, k_0^i, k_2^i), the roots v of $p^i(v, u_i, k_0^i, k_2^i) = 0$ correspond to the v-coordinate of the relative extremes of $v = v(u)$ on the graph of curve $f^i(u,v) = 0$. Considering the limit cases ($k_j^i = 0$ and $k_j^i \to \infty,\ j = 0, 2$), we obtained the above mentioned intervals for the parameter u_i in order to ensure $|v| \leq \varepsilon_i$.

3.4 G^2-Continuity

We already have shown that $\mathcal{A}$ is G^1-continuous. The above proposed cubic sections $\mathcal{A}_i$ have, by construction, free parameters $k_j^i, j = 0, 2$ that permit us to set the curvature value of $\mathcal{A}_i$ at Q_i equal to the curvature values κ_i estimated at each vertex $Q_i \in \mathcal{C}$ in the above section **Derivative data**. Hence, $\mathcal{A}$ is G^2-continuous.

3.5 Shape Control Handles

Given a polygonal chain $\mathcal{C}$, for each section $\mathcal{A}_i$ we have a free parameter, which plays the role of a shape control handle: the selection of an additional point in the interior of the region of interest Ω_i to be interpolated. If one wishes to choose this point (with barycentric coordinates (u_i, v_i)) in a non supervised way, we propose the following procedure: compute the barycentric coordinates (u_i^c, v_i^c) of the center of mass of all points in Ω_i, and set $u_i := \frac{2+u_i^c-2v_i^c}{5}$, hence the interpolating point is the point on the line $1 - 2u - v = 0$ with minimal distance to the points in Ω_i.

3.6 Curve Evaluation and Display

For intensive evaluation of the curve, a quadtree subdivision process on the triangle T_i could be used . On each sub-triangle, by means of blossom principle for triangular BB-functions, the BB-net corresponding to the sub-triangle is computed and we discard those sub-triangles on which the BB-polynomials have only positive or negative coefficients. After few recursion steps, we obtain a set of sub-triangles providing us a set of pixels, whose centers are approximately on the curve. See [4] for more details.

3.7 Numerical Examples

The algorithm proposed in this paper was successfully applied to the approximation of the contours of magnetic resonance images (MRI) of a human head (from Rendering test data set, North Carolina University, ftp.cs.unc.edu). In the same plot, the figure shows the A-splines which approximate 25 contours. Each contour was obtained from a previous processing of a digital image that corresponds to a cross section of the human head. The results were obtained from a MATLAB program that constructs and displays the A-spline curve approximating the contour data.

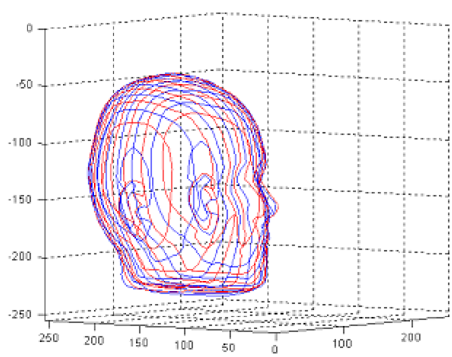

Fig. 6. Approximation of the contours of MRI of a human head

3.8 Conclusions

In comparison to [2], the A-spline proposed in the present work achieves G^2-continuity with the minimal degree (3) and we do not impose restrictions for the interpolation of tangent vectors. On the other hand, we interpolate not only the directions of the tangent vectors but also their sense, which is a completely new feature in this context. Moreover, the high flexibility of our A-spline scheme facilitates, with few adaptations, to solve efficiently another related problems

such as free design of generatrix curves as well as the computation of the structural parameters of the corresponding revolution surfaces (see [3] and [6]) and smoothing and fitting of stream lines from a finite sample of flow data.

References

[1] Bajaj C., Xu G. A-Splines (1999), Local Interpolation and Approximation using G^k-Continuous Piecewise Real Algebraic Curves, Computer Aided Geometric Desing, 16: 557-578.
[2] Bajaj C., Xu G. (2001), Regular algebraic curve sections (III) - Applications in interactive design and data fitting. Computer Aided Geometric Desing, 18: 149-173.
[3] Behar S., Hernández V., Alvarez L., Estrada J., Computing a revolution shell using a G^2-continuous A-spline and a semidiscrete method for the EDPs, Proceedings IV, ITLA, 2001, 241-250, ISBN: 959-7056-13-5.
[4] Estrada, J. , Martínez D., León, D., Theisel, H. , Solving Geometric Problems using Subdivision Methods and Range Analysis, in: Mathematical Methods for Curves and Surfaces: Tromso 2004, M. Daehlen, K. Morken and L.L. Shumaker (eds.), 2005, 101-114, Nashboro Press, Brentwood, TN.
[5] Hernández, V., Martínez D., Estrada J. (2002), Fitting a conic A-spline to contour image data, Revista Investigación Operacional, Vol. 29, 55-64.
[6] Hernández, V., Behar, S., Estrada J., Geometric design by means of a G^2 continuous A-spline, Approximation, Optimization and Mathematical Economics, Physica-Verlag, Heidelberg, 2001, 133-145.
[7] Kass,M., Witkin, A., Terzopoulos, D. (1988), Snakes: active contour models, International. J. Comput. Vision, 321-331.
[8] Paluszny M, Patterson R., G^2-continuous cubic algebraic splines and their efficient display , Curves and Surfaces II , P.J. Laurent , A. Le Méhauté , and L.L. Schumacker (eds.), 1994, 353-359.
[9] Paluszny M, Patterson R. (1998), Geometric control of G^2-cubic A-splines, Computer Aided Geometric Design 15: 261-287.
[10] Ray B., Ray K. (1994), A non-parametric sequential method for polygonal approximation of digital curves, Pattern Recognition Letters 15: 161-167.
[11] Sethian, J., A., Level Set Methods, Cambridge Univ. Press., Cambridge, 1996.