Journal of Information & Computational Science 2: 2 (2005) 375–384
Available at http://www.joics.com

A Successive Minima Method for Implicit Approximation

Zhouwang Yang*, Jiansong Deng, Changqi Hu

Department of Mathematics, University of Science and Technology of China, Hefei 230026, China

Received 18 February 2005; revised 14 May 2005

Abstract

This paper is concerned with the problem of approximating a collection of unorganized data points by an algebraic tensor-product B-spline curve. The implicit approximation to data points would be ideally based on minimizing the sum of squares of geometric distance. Since the geometric distance from a point to an implicit curve cannot be computed analytically, Sampson distance, which is the first-order approximation of the geometric distance, is introduced via a derivation from the viewpoint of optimization theory. Then, the implicit approximation is modeled as a nonlinear optimization problem by minimizing the Sampson error and the fair term for smoothing effect. By the idea of successive minima technique, we induct a quadratic constraint function of the data at every iteration step, and show that the minimization reduces to a constrained quadratic optimization subproblem, which can be solved as generalized eigenvector fitting. This successive procedure is stable and computationally reasonable. Some examples are implemented in our approach, and the high-quality reconstruction curve is obtained in a robust way.

Keywords: Implicit curve; Geometric distance; Sampson distance; Successive minima technique

1 Introduction

Curves and surfaces in geometric modeling can be described by parametric and implicit representations. Currently, most applications rely on parametric representations [7, 8], since they offer a number of advantages, such as simple sampling techniques, easy visualization. However, parametric representations introduce a parametrization of the geometry, which is often artificial. For instance, in order to fit curves or surfaces to scattered data, one has to associate certain parameter values to the data, and this parametrization would determine the shape and topology of the solution. Using implicit representations, it is possible to avoid this problematic parametrization process. The representation of curves and surfaces in implicit form offers a number of advantages

*The work is supported by the Outstanding Youth Grant of NSF of China (No. 60225002), NSF of China (No. 10201030 and No. 60473132), a National Key Basic Research Project of China (No. 2004CB318000), the TRAPOYT in Higher Education Institute of MOE of China, and SRF for ROCS, SEM.

*Corresponding author.
Email address: yangzw@ustc.edu.cn (Zhouwang Yang).

1548–7741/ Copyright © 2005 Binary Information Press
June 2005

[3]. Recently, the piecewise algebraic curves and surfaces [1, 9, 10] are considered as powerful representation in applications to blending, approximation and reconstruction.

Given a family of implicit functions defined by a finite number of coefficients and a finite set of data points assumed to belong (or be vicinal) to the same curve or surface, we want to estimate the coefficients that minimize the sum of squares of geometric distance from the data points to the implicit curve or surface. Unfortunately, there is no closed-form expression for geometric distance, and iterative methods are required to compute it. In this paper, we introduce Sampson distance [13], a first-order approximation of the geometric distance, via a derivation from the viewpoint of optimization theory. Then, the implicit approximation is modelled as a nonlinear optimization problem by minimizing the Sampson error and the fair term for smoothing effect. Since the Sampson error on a fixed set of data points is not a simple quadratic form in the coefficients, the minimization needs to be solved by some well-established nonlinear optimization techniques.

In the past, researchers [4, 11, 1] have minimized the algebraic error on the data points under different constraints. It is well known that this error function can produce a very biased result sometimes. From studying the poor performance of algebraic error and the geometric conditions under which Sampson distance failed to approximate the geometric distance, we induct a data-dependent quadratic function as constraint and turn the minimization into a constrained quadratic optimization subproblem. We then propose a successive minima algorithm, which is based on generalized eigenvector fitting, to solve the model of implicit approximation. This procedure helps to improve the solution stably at a low cost. In most of the cases, the final result of the successive minima algorithm is a very good approximation to the given data set.

2 Algebraic Tensor-product B-spline Curves and Surfaces

Let $f : \mathbb{R}^\nu \to \mathbb{R}$ be a smooth function with continuous first-order and second-order derivatives. The zeros set of function f is defined by

$$V(f) = \{\mathbf{X} \in \mathbb{R}^n \mid f(\mathbf{X}) = 0\}. \tag{1}$$

We are interested in two particular cases for their applications in computer aided geometric design: the zeros set $V(f)$ is a planar curve if $\nu = 2$, and a surface if $\nu = 3$. It is well known that $V(f)$ is identical to $V(\alpha f)$ for every nonzero α.

Let $\mathbf{P}_i = (x_i, y_i)^\tau, i = 1, \ldots, M$, be the given data points or samples on a given planar-curve. The data set is assumed to represent the shape of some planar curve Γ, which is called a target shape model. We are looking for a piecewise polynomial function, whose zero contour approximates the given target shape model.

Let us consider a bivariate tensor-product B-spline function of bi-degree (l, l)

$$f(x, y) = \sum_{r,s=1}^{m,n} c_{rs} M_r(x) N_s(y), \tag{2}$$

where $\{c_{rs}\}_{m\times n}$ are the real coefficients (called *control coefficients*), $\{M_r(x)\}_{r=1}^m$ and $\{N_s(y)\}_{s=1}^n$ denote the B-spline basis [6] functions of degree l with respect to certain user defined knot vectors $\zeta = \{\zeta_r\}_{r=1}^{m+l+1}$ and $\eta = \{\eta_s\}_{s=1}^{n+l+1}$. Accordingly, the zero set of the function f in domain $\mathcal{D} \subset \mathbb{R}^2$ is defined by

$$V(f) = \{(x, y) \in \mathcal{D} \mid f(x, y) = 0\}. \tag{3}$$

and it is called an *algebraic tensor-product B-spline curve*. For a fixed set of basis functions, the algebraic tensor-product B-spline curve is determined by the control coefficients.

For simplicity, the control coefficients and the basis functions are gathered (in a suitable ordering) into two column vectors, denoted by $\mathbf{f}$ and $\mathbf{q}(x, y)$ respectively. By the notation, $f(x, y)$ is rewritten in form

$$f(x,y) = \sum_{j=1}^{N} f_j B_j(x,y) = \mathbf{q}(x,y)^\tau \mathbf{f}, \tag{4}$$

where $N = mn$, $\mathbf{f} = (f_1, f_2, \ldots, f_N)^\tau = (c_{11}, c_{12}, \ldots, c_{mn})^\tau$ and $\mathbf{q}(x,y) = (B_1(x,y), B_2(x,y), \ldots, B_N(x,y))^\tau = (M_1(x)N_1(y), M_1(x)N_2(y), \ldots, M_m(x)N_n(y))^\tau$. In this case, we can express the value of f at a given point as

$$f(\mathbf{P}_i) = f(x_i, y_i) = \mathbf{q}_i^\tau \mathbf{f}, \tag{5}$$

and the gradient of $f(\mathbf{P}_i)$ as

$$\nabla f(\mathbf{P}_i) = \left(\frac{\partial f}{\partial x}(\mathbf{P}_i), \frac{\partial f}{\partial y}(\mathbf{P}_i)\right)^\tau = \begin{pmatrix} \mathbf{u}_i^\tau \mathbf{f} \\ \mathbf{v}_i^\tau \mathbf{f} \end{pmatrix}, \tag{6}$$

where $\mathbf{q}_i = (B_1(\mathbf{P}_i), \ldots, B_N(\mathbf{P}_i))^\tau$, $\mathbf{u}_i = (\frac{\partial B_1}{\partial x}(\mathbf{P}_i), \ldots, \frac{\partial B_N}{\partial x}(\mathbf{P}_i))^\tau$ and $\mathbf{v}_i = (\frac{\partial B_1}{\partial y}(\mathbf{P}_i), \ldots, \frac{\partial B_N}{\partial y}(\mathbf{P}_i))^\tau$.

The basic problem of describing curves is related to the creation of geometric objects in the context of specific applications. A generic method consists in determining a curve that almost passes through (or approximates) a set of points $\{\mathbf{P}_i\}_{i=1}^M$ and at the same time satisfies some application-dependent criteria. In general, besides obtaining a curve with a given shape, many other conditions that measure the 'quality' of the curve can be imposed. Among those conditions we could mention: Continuity; Fairness. Each condition imposes some restriction on the curve to be constructed. These restrictions can be of analytical, geometrical or topological nature. A convenient way to obtain curves that satisfy some set of conditions is to pose the problem in the context of optimization: define an energy function such that the curves which are minimizers of this functional automatically satisfy the desired criteria. A frequently used example is the simplified thin plate energy [5], a quadratic function in the second partial derivatives,

$$\mathrm{Eng}(\mathbf{f}) = \iint_{\mathcal{D}} (f_{xx}^2(x,y) + 2f_{xy}^2(x,y) + f_{yy}^2(x,y))\, dxdy = \mathbf{f}^\tau \mathbf{H} \mathbf{f}. \tag{7}$$

It is quadratic in the control coefficients vector $\mathbf{f}$, and the symmetrical matrix $\mathbf{H}$ can be computed by Gauss integration.

The use of algebraic tensor-product B-spline curve offers several advantages, including simple implementation, simple evaluation, simple conditions for global smoothness and differentiability, sufficient flexibility and refinability. In what follows, we use algebraic tensor-product B-spline curves as the result curves in implicit approximation.

3 Sampson Distance: First-order Approximation of Geometric Distance

Given a collection of unorganized data points $\{\mathbf{P}_i\}_{i=1}^M$ in 2D-plane, Our aim is to generate a planar algebraic tensor-product B-spline curve $V(f)$ to fit the points. The problem is essentially about the reconstruction of the target shape by an implicit curve. Firstly, one needs to define a meaningful metric

$$\mathrm{Err}(\mathbf{f}) = \mathrm{Err}(\{\mathbf{P}_i\}_{i=1}^M, V(f)), \tag{8}$$

as error function of data points set $\{\mathbf{P}_i\}_{i=1}^M$ to the implicit curve $V(f)$. We will now describe several error functions which may be minimized in order to determine the control coefficients vector $\mathbf{f}$ of the implicit curve $V(f)$.

Generally, the geometric distance from one point to an implicit curve is defined by

$$\mathrm{Dist}(\mathbf{P}, V(f)) = \|\mathbf{X} - \mathbf{P}\|, \tag{9}$$

where $\mathbf{X} \in V(f)$ is the nearest point (foot-point) on the implicit curve. For every point $\mathbf{P}$, the geometric distance $\mathrm{Dist}(\mathbf{P}, V(f))$ assigns it the shortest Euclidean distance to implicit curve $V(f)$. Then, we denote the sum of squares of geometric distance as geometric error

$$\mathrm{Err}_{geo}(\mathbf{f}) = \sum_{i=1}^{M} \mathrm{Dist}(\mathbf{P}_i, V(f))^2. \tag{10}$$

It is ideal to minimizing the geometric error when an implicit curve approximates given data points. Unfortunately, this metric entails an intractable minimization problem whose solution cannot be expressed analytically in closed form.

In contrast with complexity of geometric error function, the algebraic error

$$\mathrm{Err}_{alg}(\mathbf{f}) = \sum_{i=1}^{M} f(\mathbf{P}_i)^2, \tag{11}$$

is straightforward to compute as the sum of squares of algebraic distance [1] on all points. The minimization problem based on the algebraic error function is very simple, but this function can produce a very biased result at some time. All these lead to a further error function that lies between the algebraic error and the geometric error in terms of complexity, but gives a close approximation to geometric error. We will refer to this error function as Sampson distance error since Sampson used it for conic fitting [13]. We get to Sampson distance via a different derivation from the viewpoint of optimization theory.

First, let us recall that the geometric distance $\mathrm{Dist}(\mathbf{P}, V(f))$ is the Euclidean distance from $\mathbf{P}$ to a nearest point on the implicit curve $V(f)$. If let $\mathbf{X}$ be the point on the variety $V(f)$, the geometric distance can be defined identically by a constrained minimization model:

$$\min\{\|\mathbf{X} - \mathbf{P}\| \mid f(\mathbf{X}) = 0\}. \tag{12}$$

From this model, the point which results in the geometric distance, cannot be estimated directly except via iterative, because of the nonlinear nature of the variety $V(f)$. The idea is to assume that the function $f(\mathbf{X})$ can be well approximated linearly in the neighborhood of the estimated point. To first order, the function $f(\mathbf{X})$ may be approximated locally with (Taylor) expansion

$$f(\mathbf{X}) = f(\mathbf{P}) + \nabla f(\mathbf{P})^\tau (\mathbf{X} - \mathbf{P}) + o(\|\mathbf{X} - \mathbf{P}\|). \tag{13}$$

If we write $\delta_\mathbf{P} = \mathbf{X} - \mathbf{P}$ and desire $\mathbf{X}$ to lie on the variety $V(f)$ so that $f(\mathbf{X}) = 0$, then the minimization problem is turned to find the smallest $\delta_\mathbf{P}$ that satisfies linear constraint:

$$\min\{\|\delta_\mathbf{P}\| \mid f(\mathbf{P}) + \nabla f(\mathbf{P})^\tau \delta_\mathbf{P} = 0\}. \tag{14}$$

We solve minimization problem (14) directly, and get

$$\delta_{\mathbf{P}} = \frac{-f(\mathbf{P})}{\|\nabla f(\mathbf{P})\|^2} \cdot \nabla f(\mathbf{P}). \tag{15}$$

This argument suggests that $\|\delta_{\mathbf{P}}\| = \dfrac{|f(\mathbf{P})|}{\|\nabla f(\mathbf{P})\|}$, named Sampson distance, be a first-order approximation of the geometric distance, and the sum of $\|\delta_{\mathbf{P}_i}\|^2$ be Sampson distance error:

$$\mathrm{Err}_{sps}(\mathbf{f}) = \sum_{i=1}^{M} \frac{f(\mathbf{P}_i)^2}{\|\nabla f(\mathbf{P}_i)\|^2} = \sum_{i=1}^{M} \frac{\mathbf{f}^\tau \mathbf{A}_i \mathbf{f}}{\mathbf{f}^\tau \mathbf{B}_i \mathbf{f}}, \tag{16}$$

where $\mathbf{A}_i = \mathbf{q}_i \mathbf{q}_i^\tau$ and $\mathbf{B}_i = \mathbf{u}_i \mathbf{u}_i^\tau + \mathbf{v}_i \mathbf{v}_i^\tau$.

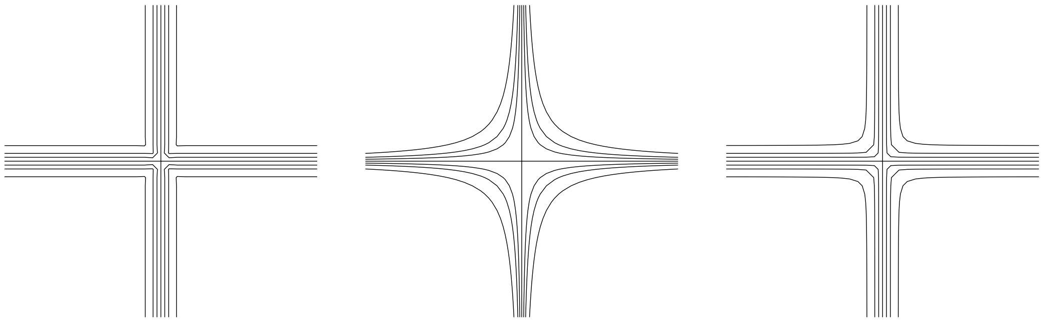


Fig. 1: Contours of constant distance to the curve $V(f) = \{(x, y) \mid xy = 0\}$.

Fig. 1 shows several contours of constant geometric distance, constant algebraic distance and constant Sampson distance to an implicit curve $V(f) = \{(x, y) \mid xy = 0\}$. The Sampson distance also has several interesting geometric properties. It is independent of the homogeneous representation of $V(f)$. If $\alpha \neq 0$ and $\widetilde{f}(\mathbf{X}) = \alpha f(\mathbf{X})$, then

$$\frac{|\widetilde{f}(\mathbf{X})|}{\|\nabla \widetilde{f}(\mathbf{X})\|} = \frac{|\alpha f(\mathbf{X})|}{\|\alpha \nabla f(\mathbf{X})\|} = \frac{|f(\mathbf{X})|}{\|\nabla f(\mathbf{X})\|}. \tag{17}$$

Let $\mathbf{Y} = \dfrac{1}{\alpha}\mathbf{R}\mathbf{X} + \mathbf{T}$ be a similar transformation of the space variables and denote $g(\mathbf{Y}) = g(\dfrac{1}{\alpha}\mathbf{R}\mathbf{X} + \mathbf{T}) \triangleq f(\mathbf{X})$, then $\nabla g(\mathbf{Y}) = \dfrac{1}{\alpha}\mathbf{R}\nabla f(\mathbf{X})$, and therefore

$$\frac{|g(\mathbf{Y})|}{\|\nabla g(\mathbf{Y})\|} = |\alpha| \frac{|f(\mathbf{X})|}{\|\nabla f(\mathbf{X})\|}, \tag{18}$$

where $\mathbf{R}$ is an orthogonal matrix and $\mathbf{T}$ is a translation vector. This derivation shows that the Sampson distance is correspondingly scaling changed under a similar transformation. Specially, it is invariant to rigid body transformation.

4 Implicit Approximation Model via Successive Minima Algorithm

Let $\{\mathbf{P}_i\}_{i=1}^M$ be given data points or samples on a given target curve Γ. We are looking for a curve $V(f)$ with implicit representation to approximate the given data set. Then, the implicit curve approximation is modeled as minimizer of a functional,

$$\min Q(\mathbf{f}) = \sum_{i=1}^{M} \frac{f(\mathbf{P}_i)^2}{\|\nabla f(\mathbf{P}_i)\|^2} + w\mathbf{f}^\tau \mathbf{H}\mathbf{f}. \tag{19}$$

The first part of the object function is the Sampson distance error from given data points $\{\mathbf{P}_i\}_{i=1}^M$ to implicit curve $V(f)$. With the Sampson error $\mathrm{Err}_{sps}(\mathbf{f})$ minimization, one would fit an implicit curve to the target shape of data points. The second part is a fair term $\mathrm{Eng}(\mathbf{f})$ with certain non-negative weight w, and is used to enforce the fairness of the result curve. We add the fair term to pull the approximating curve towards a simpler shape and keep away the unwanted branches.

Because the Sampson error (so the object function) is not a simple quadratic form in the unknown control coefficients vector $\mathbf{f}$, we cannot express the solution to minimization problem (19) analytically in closed form. Usually some well-established nonlinear optimization techniques, such as trust-region methods, are suggested to solve the model [16].

This numerical intractability can be avoided by an idea of successive minima technique, which is similar in spirit to those "iterative weighted least-squares algorithms" and "reweight procedure" that appear in the literature [12, 15]. First, we assume to have got an estimated $\mathbf{f}^{(k)}$, whose gradient gives a guess to $\nabla f(\mathbf{P}_i)$, and denote $\theta_{i,k} = \dfrac{1}{\|\nabla f^{(k)}(\mathbf{P}_i)\|}$. The Sampson error can be replaced with $\sum_{i=1}^{M} \theta_{i,k}^2 f(\mathbf{P}_i)^2$ currently and the object function of problem (19) is turned into quadratic form of the control coefficients vector $\mathbf{f}$. That is

$$Q^{(k)}(\mathbf{f}) = \mathbf{f}^\tau \mathbf{A}^{(k)}\mathbf{f}, \tag{20}$$

where $\mathbf{A}^{(k)} = \sum_{i=1}^{M} \theta_{i,k}^2 \mathbf{A}_i + w\mathbf{H}$ is a positive-definite matrix. Second, since the implicit representation is homogeneous, we induct a quadratic constraint on the control coefficients

$$\mathbf{f}^\tau \mathbf{B}\mathbf{f} = M \tag{21}$$

to obtain a nontrivial solution. The constraint function is data-dependent, with matrix $\mathbf{B} = \sum_{i=1}^{M} \mathbf{B}_i$ non-negative-definite.

Based on aforesaid analysis, we minimize Eq. (20) constrained by Eq. (21) and form a constrained optimization subproblem

$$\begin{array}{ll} \min & Q^{(k)}(\mathbf{f}) = \mathbf{f}^\tau \mathbf{A}^{(k)}\mathbf{f}, \\ \text{s.t.} & \mathbf{f}^\tau \mathbf{B}\mathbf{f} = M \end{array} \tag{22}$$

at every iteration step in the successive minima method. From the principle of Lagrange-multiplier [2], the constrained optimization subproblem (22) reduces to generalized eigenvector fitting

$$(\mathbf{A}^{(k)} - \lambda \mathbf{B})\mathbf{f} = 0, \tag{23}$$

where λ is generalized eigenvalue of $\mathbf{A}^{(k)}$ to $\mathbf{B}$. The procedure of generalized eigenvector fitting would find the solution as generalized eigenvector related to the minimal generalized eigenvalue $\lambda_{\min}$ at a low cost.

With the above analysis, we generate the successive minima algorithm for implicit curve approximation as follows:

Successive Minima Algorithm

(1) Input the given data points $\{\mathbf{P}_i\}_{i=1}^{M}$, generate the matrices $\mathbf{H}$ and $\mathbf{B}$. Given initial guess of the control coefficients vector $\mathbf{f}^{(0)}$, and compute Sampson error $\mathrm{Err}_{sps}(\mathbf{f}^{(0)})$. Set $w > 0$, $\varepsilon > 0$ and $k := 0$.

(2) Compute $\theta_{i,k}$ and synthesize the matrix $\mathbf{A}^{(k)}$. Solve the subproblem (22) based on generalized eigenvector fitting, which gives a solution $\hat{\mathbf{f}}$. Resize the vector $\hat{\mathbf{f}} = (M/\hat{\mathbf{f}}^{\tau}\mathbf{B}\hat{\mathbf{f}})^{1/2}\hat{\mathbf{f}}$.

(3) Calculate the ratio $\rho = \dfrac{\mathrm{Err}_{sps}(\hat{\mathbf{f}})}{\mathrm{Err}_{sps}(\mathbf{f}^{(k)})}$, and set

$$\mathbf{f}^{(k+1)} = \begin{cases} \hat{\mathbf{f}} & \text{if } \rho < 1, \\ \mathbf{f}^{(k)} & \text{otherwise.} \end{cases}$$

(4) If $\rho < 1 - \varepsilon$, set $k := k + 1$ and go to step (2); Otherwise, output the control coefficients vector $\mathbf{f}^{(k+1)}$.

Thus, the control coefficients vector $\mathbf{f}^{(k+1)}$ would determine a result implicit curve

$$V(f^{(k+1)}) = \{(x, y) \in \mathcal{D} \mid f^{(k+1)}(x, y) = \mathbf{q}(x, y)^{\tau}\mathbf{f}^{(k+1)} = 0\},$$

as final approximation to the target shape model Γ, i.e., given data points $\{\mathbf{P}_i\}_{i=1}^{M}$.

5 Implementation and Examples

In the implementation of implicit curves approximation, our first task is to obtain the knots vectors, with which the tensor-product B-spline function is defined. When a set of data points is given as the target shape model, we consider a slightly enlarged area of the data bounding box, and subdivide the area into rectangular cells with equidistant grids on the x and y axis. By adding additional l knots at the beginning and at the end of the grids, we obtain the knots vectors $\zeta = \{\zeta_r\}_{r=1}^{m+l+1}$ and $\eta = \{\eta_s\}_{s=1}^{n+l+1}$.

As mentioned in the previous section, the successive algorithm for implicit approximation is iterative and an initial step is required. In order to begin our calculation, we are optional to make an initial guess of the control coefficients vector $\mathbf{f}^{(0)}$, or to directly set $\theta_{i,0} = 1$ for $i = 1, \cdots, M$. For the former, we give an ordinary specification of initial implicit curve $V(f^{(0)})$, as setting the control coefficients

$$f_j^{(0)} = f_{(r-1)n+s}^{(0)} = c_{rs} = \frac{r(m-r+1)}{m}\frac{s(n-s+1)}{n} - c_0, \quad j = 1, \cdots, N. \tag{24}$$

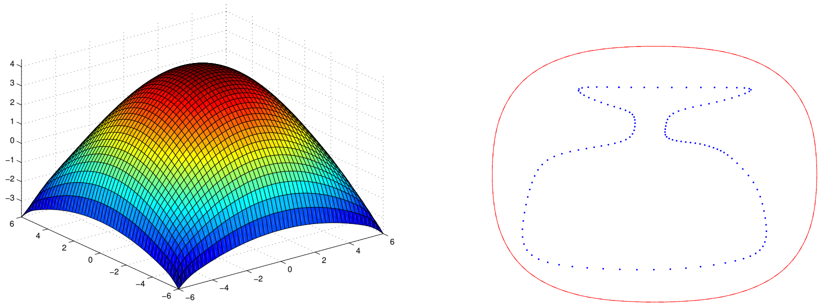
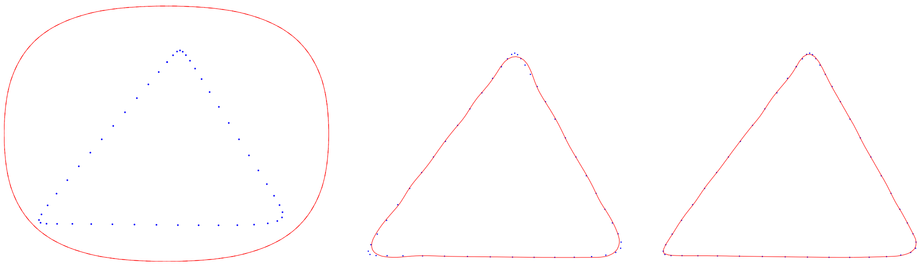


Fig. 2: The graph of $f^{(0)}(x,y)$ and the initial curve $V(f^{(0)})$.

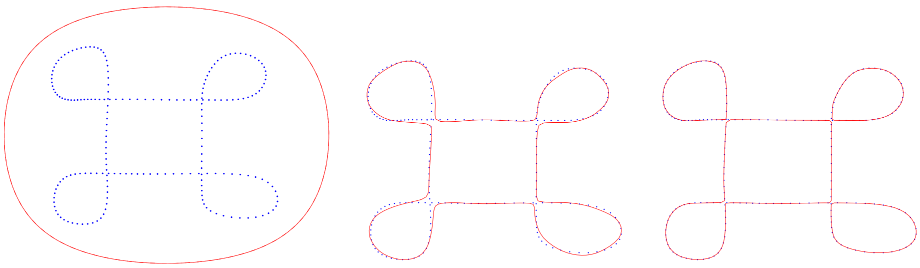
The constant c_0 is properly chosen to make the initial curve $V(f^{(0)})$ encase the target shape model, i.e., given set of data points $\{\mathbf{P}_i\}_{i=1}^M$, as show in Fig. 2.

Various experiment of implicit approximation is implemented in the successive minima method which is described in previous section. The successive procedures of some selected examples are illustrated in Fig. 3- Fig. 6, where we set $l = 2$, $m = 8$ and $n = 8$.



(a) (b) (c)

Fig. 3: Example 1. (a) data points and initial curve; (b) 1st step; (c) 2nd step.



(a) (b) (c)

Fig. 4: Example 2. (a) data points and initial curve; (b) 1st step; (c) 2nd step.

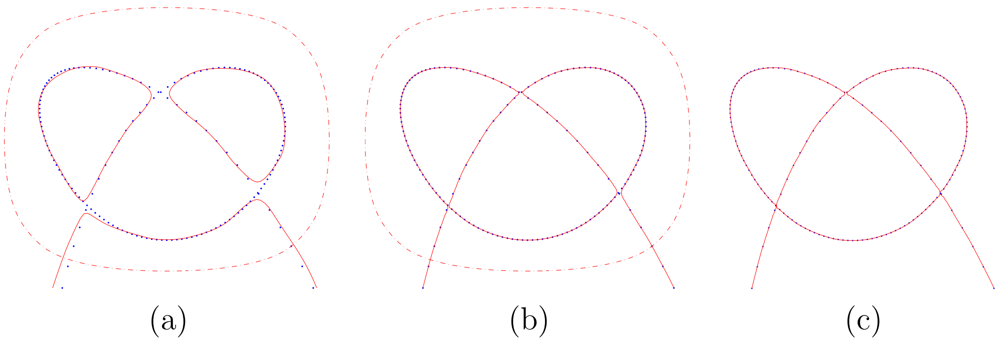


Fig. 5: Example 3. (a) 1st step; (b) 2nd step; (c) 3rd step.

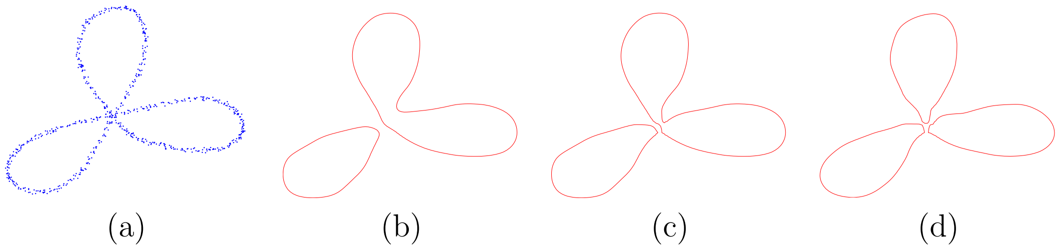


Fig. 6: Example 4. (a) noisy data; (b) 1st step; (c) 2nd step; (d) 3rd step.

6 Conclusions

We have described a successive method for fitting implicitly defined curves to planar scattered data points. The examples in this study demonstrate that the successive minima algorithm for implicit curve approximation is stable and computationally reasonable. In most of the cases, the successive procedure would get a very good approximation of the given data points.

Due to the use of tensor-product B-spline, the implementation of our method is relatively simple. In fact, the successive minima method can be applied to any implicitly defined planar-curve, not only to algebraic curves and surface in tensor-product representation. For instance, one can apply this successive minima algorithm to bivariate piecewise polynomials of total degree l, which is defined on triangular domain or T-mesh [14]. In the future, we will try to extend the method to implicit surface reconstruction. Furthermore, it is worthwhile to explore the applications of implicit B-spline curves and surfaces in geometric modeling and computer graphics.

References

[1] C. L. Bajaj, I. Ihm and J. Warren, Higher-order interpolation and least-squares approximation using implicit algebraic surfaces, ACM Transactions on Graphics 12 (4) (1993) 327-347.

[2] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, 1982.

[3] J. Bloomenthal (Ed.), *Introduction to Implicit Surfaces*, Morgan Kaufmann Publishers, 1997.

[4] F. Bookstein, Fitting conic sections to scattered data, Comput. Graph. Image Process 9 (1979) 56-71.

[5] G. Brunnett, H. Hagen and P. Santarelli, Variational design of curves and surfaces, Surveys on Mathematics for Industry 3 (1) (1993) 1-27.

[6] C. de Boor, *A Practical Guide to Splines*, Springer-Verlag, New York, 2001.

[7] G. Farin, *Curves and Surfaces for CAGD–A Pratical Guide*, 5th Edition, Morgan Kaufmann Publishers, 2002.

[8] J. Hoschek and D. Lasser, *Fundamentals of Computer Aided Geometric Design*, AK Peters, Wellesley, MA, 1993.

[9] B. Jüttler and A. Felis, Least-squares fitting of algebraic spline surfaces, Advances in Computational Mathematics 17 (2002) 135-152.

[10] B. Jüttler, Approximate implicitization via curve fitting, in: Eurographics Symposium on Geometry Processing, 2003, pp. 240-278.

[11] V. Pratt, Direct least-squares fitting of algebraic surface, ACM Comput. Graphics (Siggraph) 21 (1987) 145-152.

[12] J. O. Ramsay, A comparative study of several robust estimates of slope, intercept, and scale in linear regression, Journal of Amer. Stat. Assoc. 72 (1977) 608-615.

[13] P. D. Sampson, Fitting conic sections to very scattered data: An iterative refinement of the Bookstein algorithm, Computer Vision, Graphics and Image Processing 18 (1982) 97-108.

[14] T.W. Sederberg, J. Zheng, A. Bakenov and A. Nasri. T-splines and T-Nurbs, Proceedings of SIGGRAPH'03, 2003.

[15] R. Taubin, Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation, IEEE Trans. Pattern Anal. Mach. Intelligence 13 (1991) 1115-1138.

[16] Z. W. Yang, J. S. Deng and F. L. Chen, Dynamic implicit curve reconstruction based on approximate geometric distance, Journal of Software (Chinese version) 15 (Suppl.) (2004) 264-272.