Appl. Math. J. Chinese Univ.
2011, 26(2): 213-229

Cubic Algebraic Spline Curves Design

XU Chen-dong[1] CHEN Fa-lai[2] DENG Jian-song[2] YANG Zhou-wang[2]

Abstract. In this paper we propose a construction method of the planar cubic algebraic spline curve with endpoint interpolation conditions and a specific analysis of its properties. The piecewise cubic algebraic curve has G^2 continuous contact with the control polygon at two endpoints and is G^2 continuous between each segments of itself. The process of this method is simple and clear, and provides a new way of thinking to design implicit curves.

§1 Introduction

Cubic parametric curves play an important role in computer aided geometric design and computer graphics. This is mainly due to that the cubic parametric curve is the curve of the minimal degree which could be singular. Here the singularity includes cusps, inflection points, closed circles, and so on. Furthermore, all quadratic curves are planar, while the cubic curves can express a space curve whose torsion is not always zero. The representations and properties of the parametric curve are discussed and studied in many literatures, see [9,16,22] and references therein. Usually the methods of parametric curve design are intuitive. In other words, we can construct a suitable curve in an interactive environment. For example in order to change the curve's shape, we only need to adjust the position of some control points.

In recent years, (piecewise) implicit/algebraic curves and surfaces have received increasing attentions [17]. To visualize the implicit surfaces, [2] presented a practical algorithm of polygonization and [10] used the ray tracing algorithm. In 1984, Sederberg used piecewise algebraic curve in geometric modeling [20]. [1] presented a sufficient representation of a trivariate polynomial within a tetrahedron (named A-patch). [17] brought forward the concept of functional splines, discussed the properties of the parabolic functional spline, and applied them to surface blending design. As a supplement, [11] discussed the determination method of the contact points or contact curves as using the parabolic in blending design in detail. [12] proved the convexity of this kind of splines. [13] provided systematical analysis on various kinds of functional

Received: 2008-9-22.
MR Subject Classification: 65D17, 65D18, 68U07.
Keywords: CAGD, piecewise algebraic curve, functional spline, geometric continuity.
Digital Object Identifier(DOI): 10.1007/s11766-011-2103-1.
Supported by the National Key Basic Research Project of China (No. 2004CB318 000), the NSF of China (No. 60533060/60872095), the Specialized Research Fund for the Doctoral Program of Higher Education (No. 20060358055), and the Subject Foundation in Ningbo University(No. xkl09046).

spline methods and used them in the smoothing manipulation of G^n blending at cusps. The first method of implicitization of parametric surfaces was given by [19]. [21] first proposed the moving line and moving surface method to make the implicitization process much more efficient, but this method could be theoretically improved. [7] described the method as the μ basis theory which was further improved and expanded in [3–6]. To solve the surface intersection problem, [14] used a projection technique. [8] provided a quite comprehensive approach to solve the blending problem of algebraic surfaces which is first studied by [23]. In addition, there are many other methods in implicit molding such as CSG method, radial basis function method, particle system, and so on.

As known to all, a planar polynomial or rational polynomial parametric curves can be converted into an algebraic curve with the same degree. Compared with parametric curves, algebraic curves of the same degree are with more freedom. For example, a cubic algebraic curve can express a simple close curve with C^∞ continuity, while a cubic polynomial or rational polynomial parametric curve can not express any entire close curve.

It is regrettable that those methods of implicit curve design used in practice are all without intuitive control. In this paper we will propose a construction method of the cubic algebraic spline curve with endpoint interpolation conditions and a specific analysis of its properties. This method can be regarded as a new way of designing implicit curves.

The rest of this paper is organized as follows: first we construct a cubic algebraic interpolation curve from the given four control points and concretely analyze the geometric properties of this curve such as convexity, convex hull and the treatment of some special situations etc; then based on this interpolation curve construction method, we define a piecewise cubic algebraic curve of the given $n > 4$ control points; also we will give some examples to illustrate the effect; the conclusion and analysis of cubic algebraic spline curves are given finally.

§2 Cubic algebraic interpolation curve

In this section, we discuss how to construct a planar cubic algebraic interpolation curve from the given four control points. The geometric properties of this curve are analyzed as well.

2.1 Construction of the algebraic interpolation curve

Given four planar control points $\{\mathbf{p}_i\}_{i=0}^3$, assume that the line equation of the control edge $\mathbf{p}_0\mathbf{p}_1$ is l_1, the line equation of $\mathbf{p}_1\mathbf{p}_2$ is l_2, the line equation of $\mathbf{p}_2\mathbf{p}_3$ is l_3, the line equation of $\mathbf{p}_3\mathbf{p}_0$ is l_0. The positive and negative values of these linear equation in the whole plane are shown in Figure 1. Let

$$L(x,y) = (1-\lambda)l_1l_2l_3 - \lambda l_0^3, \quad \lambda \in (0,1). \tag{1}$$

We can define a cubic algebraic curve as the zero set of $L(x,y)$ on $\mathbb{R}^2$. Here, the parameter λ is used to adjust the shape of the curve $L(x,y)$. For example, to the curve segment inside the control polygon formed by $\{\mathbf{p}_i\}_{i=0}^3$ as shown in the figure, the larger the parameter λ is, the

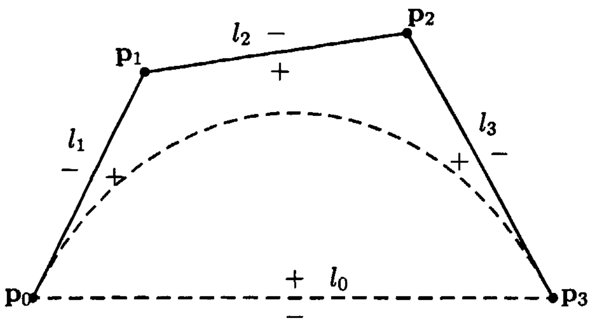

Figure 1: Control points and the positive-negative stipulation of the control edges.

closer the curve is to the control edge $\mathbf{p}_0\mathbf{p}_3$. Here, the role of the parameter λ is similar to the conic shape factor k in the implicit equation of a conic section (see page 292 of [18]).

It is easy to show that the algebraic curve $L(x, y)$ must pass through the two endpoints $\mathbf{p}_0$ and $\mathbf{p}_3$, i.e., interpolate the intersection points of l_0 and l_1, l_0 and l_3. Similarly, if l_2 and l_0 intersect, the intersection point is also on $L(x, y)$. However, here we are mainly concerned about the part of the curve $L(x, y)$ that is inside the convex hull of all the control points. Moreover, according to the definition of the geometric continuity of algebraic curves, we have:

Proposition 2.1. *$L(x, y)$ has G^2 continuous contact with the control edges $\mathbf{p}_0\mathbf{p}_1$ and $\mathbf{p}_2\mathbf{p}_3$ at the endpoints $\mathbf{p}_0$ and $\mathbf{p}_3$ respectively.*

Assume the convex hull of all the control points is Ω. Hereinafter we discuss the properties of the algebraic curve $L(x, y)$ in the region Ω. For the given four control points, there are many ways to form control polygons. According to the positions of the control points, these cases can be classified into two categories:

- All the control points are the vertices of the convex polygon Ω. This kind of polygons is called non-degenerate. See Figure 2;
- One of the control points is inside the convex polygon Ω or on some edge of it. This kind is called degenerate. See Figure 3.

In addition, if some of the control points are superposed, the construction method according to the formula (1) is no longer available and needs additional processes, such as deleting the superposed point, or using some other ways.

In the following subsections, we will show that, for the non-degenerate control polygons, the cubic algebraic interpolation curves constructed as in formula (1) have convexity and convex hull properties.

2.2 Convexity

After extending any edge of the polygon infinitely to become a straight line, if all the other edges are in the same side of the line, then the polygon is called a convex polygon. Of course,

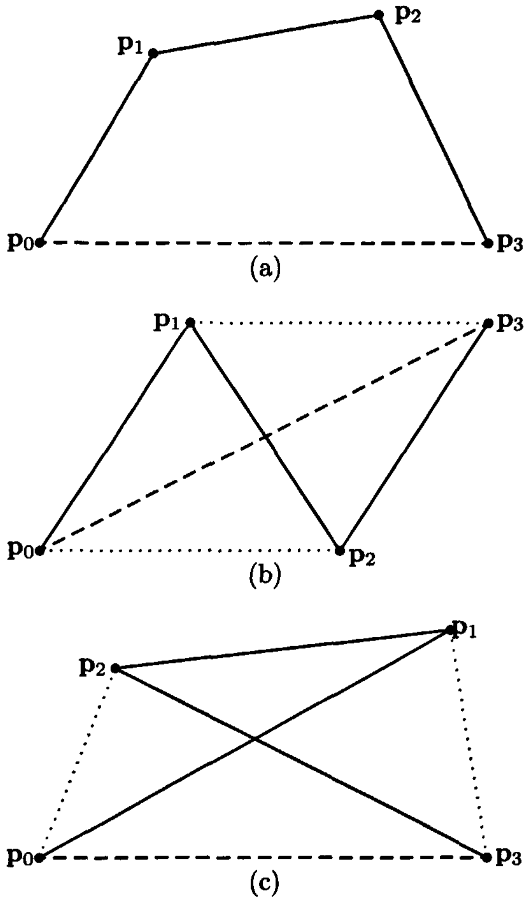

Figure 2: The three cases of the non-degenerate control polygon.

it can also be defined by the magnitude of inner angles: if each interior angle of the polygon is less than π, then the polygon is called a convex polygon. For the control polygon of the given four control points, the first case of the non-degenerate control polygons in Figure 2 is a convex polygon, while the other two cases are not.

The cubic algebraic curve constructed as (1) satisfies the following theorem:

Theorem 2.1. **(Convexity)** *If the four control points* $\{\mathbf{p}_i\}_{i=0}^3$ *form a convex polygon, the cubic algebraic interpolation curve* $L(x, y)$ *defined in the convex hull* Ω *is convex.*

Proof: Without loss of generality, assuming the four control points as Figure 2(a), we only need to prove that the cubic algebraic curves $L(x, y)$ in the convex hull of the control points must be a convex curve.

Through a projective transformation, a general convex quadrangle can be transformed into a rectangle as shown in Figure 4. This transformation does not change the convexity of the

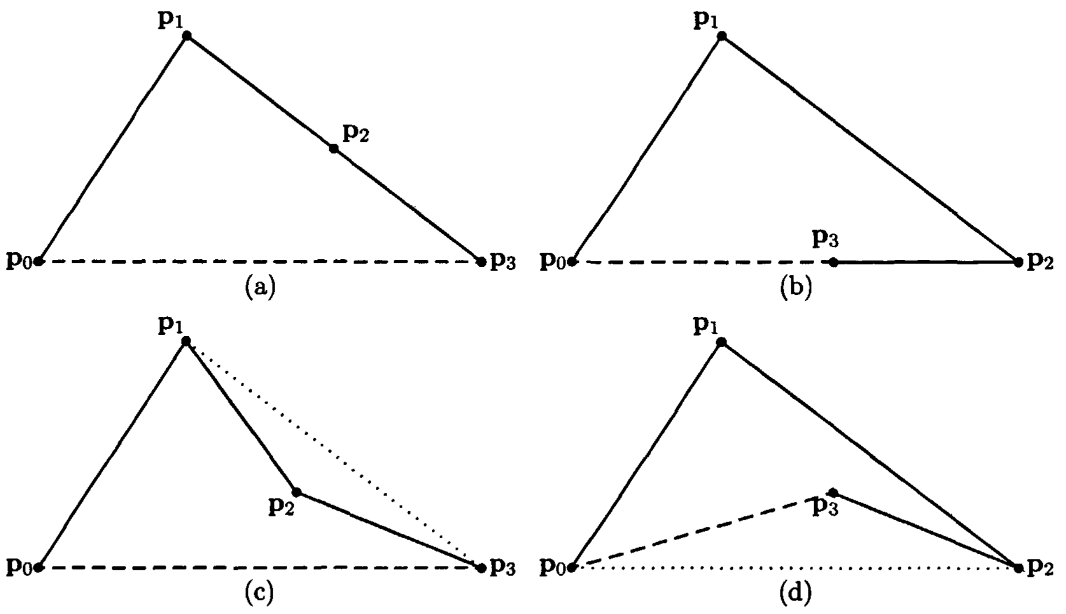

Figure 3: The four cases of the degenerate control polygon.

curve inside the quadrangle, so we only need to prove the theorem for the situation that the control polygon is a rectangle.

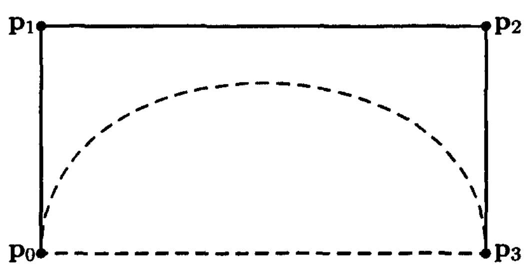

Figure 4: The control polygon after the transformation.

After choosing a suitable coordinate system, assume that the expressions of the straight lines of the control edges are:

$$\begin{cases} l_0(x,y) = y \\ l_1(x,y) = x + a \\ l_2(x,y) = -y + d \\ l_3(x,y) = -x + a \end{cases}$$

Where, $a, d \in \mathbb{R}_+$. Obviously, inside the rectangular, these linear expressions are all positive. Here, the convex hull of the control polygon is $\Omega = \{(x,y) \mid |x| < a, 0 < y < d\}$, and the expression of the cubic algebraic curve is

$$L(x,y) = (1-\lambda)(a-x)(a+x)(d-y) - \lambda y^3$$

Considering y as a function of x determined by $L(x,y) = 0$, we have the second derivative of y as follows:

$$\frac{\partial^2 y}{\partial x^2} = \frac{2L'_x \cdot L'_y \cdot L''_{xy} - (L'_y)^2 \cdot L''_{xx} - (L'_x)^2 \cdot L''_{yy}}{(L'_y)^3}, \quad (L'_y \neq 0)$$

Since $L'_y(x,y) = -3\lambda y^2 - (a-x)(a+x)(1-\lambda)$ is always negative in the region Ω, the denominator of this formula is always negative too. While the numerator can be written as follows.

$$2(d-y)(1-\lambda)^3\left[x^2(2a^2-3x^2) + (a^2+3y^2\frac{\lambda}{1-\lambda})^2 + 6x^2y(2d-y)\frac{\lambda}{1-\lambda}\right]$$

Inside the rectangular region Ω, it is true that the first term in the square brackets is larger than $-a^4$, the second term is larger than a^4 and the third term is larger than or equal to zero. Furthermore when the third term is equal to zero namely $x = 0$, the first term is zero too while the second term is larger than a^4 all the same. In sum, the numerator is always positive in Ω.

Therefore, the second derivative of y about x inside the rectangular is less than zero, namely the curve $L(x,y)$ is convex in this region. ■

2.3 Convex hull property

For the case of non-degenerate control polygon, about the properties of the cubic algebraic curve $L(x,y)$, we also have the following theorem.

Theorem 2.2. (Convex hull property) *If the four control points $\{\mathbf{p}_i\}_{i=0}^3$ form a non-degenerate control polygon, the cubic algebraic interpolation curve $L(x,y)$ defined in the convex hull Ω is continuous and only intersects the edges of Ω at two points $\mathbf{p}_0$ and $\mathbf{p}_3$.*

Proof: There are three cases of the non-degenerate control polygon as (a), (b) and (c) in Figure 2 respectively.

For the first case that the control points form a convex quadrangle, by the previous theorem of the convexity, the cubic algebraic curve segment $L(x,y)$ inside the convex hull of the control points is convex. Furthermore, the points on the edges of the control polygon except $\mathbf{p}_0$ and $\mathbf{p}_3$ are not on this curve, so the theorem holds.

For the second case that all the control points are the vertexes of the convex polygon Ω and the control edges $\mathbf{p}_1\mathbf{p}_2$ and $\mathbf{p}_0\mathbf{p}_3$ have an intersection point (named as $\mathbf{p}_4$), assuming the triangular region determined by the control points $\mathbf{p}_{i_1}$, $\mathbf{p}_{i_2}$ and $\mathbf{p}_{i_3}$ is $\Delta_{i_1 i_2 i_3}$, it is easy to known that the curve $L(x,y)$ certainly pass through $\mathbf{p}_0$, $\mathbf{p}_3$ and $\mathbf{p}_4$ and the points on the edges of the convex quadrangle Ω except $\mathbf{p}_0$ and $\mathbf{p}_3$ are not on this curve. The points inside regions Δ_{143} and Δ_{042} can not satisfy the curve equation $L(x,y) = 0$, so the curve is definitely inside Δ_{041} and Δ_{243}, and these two segments all have G^2 continuous contact with the line l_2 at the point $\mathbf{p}_4$. Further, one can assume that the control edge $\mathbf{p}_0\mathbf{p}_1$ is parallel with $\mathbf{p}_2\mathbf{p}_3$ after a projective transformation. Considering the curve segment in Δ_{041}, assuming that there is a point $\mathbf{q}_1$ on $\mathbf{p}_0\mathbf{p}_4$, and the parallel line of $\mathbf{p}_2\mathbf{p}_3$ that passes through this point intersects the line segment

$\mathbf{p}_1\mathbf{p}_4$ at the point $\mathbf{q}_2$, we have $L(\mathbf{q}_1) > 0$ and $L(\mathbf{q}_2) < 0$. Furthermore, if the point $\mathbf{p}$ moves from $\mathbf{q}_1$ to $\mathbf{q}_2$ along a straight line, the function $L(\mathbf{p})$ decreases monotonously so that the line segment $\mathbf{q}_1\mathbf{q}_2$ has only one intersection point with the curve. For the curve segment in Δ_{243}, the analysis is similar. In sum, we have that the curve $L(x, y) = 0$ does not have multiple segments inside Ω. So the theorem also holds.

For the third case that all the control points are the vertexes of the convex polygon Ω and the control edges $\mathbf{p}_0\mathbf{p}_1$ and $\mathbf{p}_2\mathbf{p}_3$ have a intersection point (also named as $\mathbf{p}_4$), it is easy to known that the curve $L(x, y)$ certainly pass through points $\mathbf{p}_0$ and $\mathbf{p}_3$, and the points on the edges of the convex polygon Ω and the four control edges except $\mathbf{p}_0$ and $\mathbf{p}_3$ are not on this curve. The points in regions Δ_{024} and Δ_{134} can not satisfy the curve equation $L(x, y) = 0$, so the curve is definitely in Δ_{124} and Δ_{034}. Furthermore, the curve segment in the region Δ_{034} is a continuous curve passing through points $\mathbf{p}_0$ and $\mathbf{p}_3$, while the one in Δ_{124} is a close curve if it exists. And the proposition below gives a sufficient condition to determine whether the close curve appears there. So the theorem holds too.

Thus, the theorem holds. ■

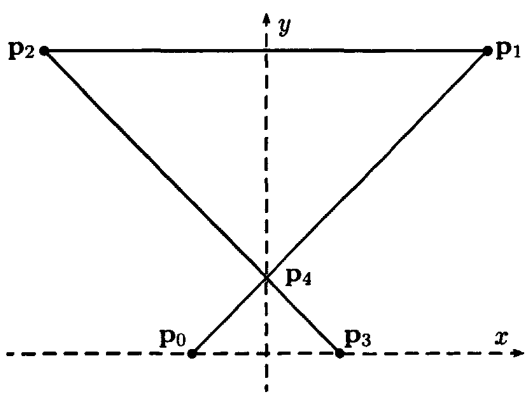

Figure 5: The third non-degenerate control polygon after the transformation.

The non-degenerate control polygon as Figure 2(c) can be transformed into the one as Figure 5 by a projective transformation which keeps the curve's convexity. Using an appropriate coordinate system as shown in the figure, the expressions of those lines of the control edges are:

$$\begin{cases} l_0(x,y) = y \\ l_1(x,y) = -y + x + a \\ l_2(x,y) = -y + a + b \\ l_3(x,y) = -y - x + a \end{cases}$$

Where, $a, b \in \mathbb{R}_+$. Then, the following proposition gives a sufficient condition about the segment of the cubic algebraic curve defined as (1) in the region Δ_{124}, that is to determine whether this close curve exists.

Proposition 2.2. *If the parameter λ satisfies $\frac{b^2}{(3a+b)^2} < \lambda < 1$, then the curve $L(x, y)$ in Ω*

constructed from the four given control points $\{\mathbf{p}_i\}_{i=0}^3$ *as Figure* 5 *merely appears inside the region* Δ_{034}.

Proof: We only need to prove that there is no multiple segment of the curve $L(x, y)$ in the region Δ_{124} when the parameter λ of the algebraic curve construction formula satisfies the condition of the proposition.

For symmetry, if there is a branch of $L(x, y)$ in Δ_{124}, this close curve segment must intersect the y axis. According to the above line equations, we have:

$$L(0, y) = (1-\lambda)(a-y)^2(a+b-y) - \lambda y^3$$

In addition, $L(0,0) = a^2(a+b)(1-\lambda) > 0$ and $L(0, a) = -\lambda a^3 < 0$, thereby the continuous function $L(0, y)$ must have a zero point in the interval $(0, a)$, namely the curve $L(x, y)$ must appear in Δ_{034}.

Considering the properties of the function $L(0, y)$ in the interval $(a, a+b)$, its derivative is:

$$\begin{aligned}\frac{d}{dy}L(0, y) &= -(1-\lambda)(a-y)^2 - 2(1-\lambda)(a+b-y)(a-y) - 3\lambda y^2 \\ &= -3t^2 + (2b - 2b\lambda - 6a\lambda)t - 3a^2\lambda\end{aligned}$$

where $t = y - a \in (0, b)$.

This is a quadratic polynomial of t, and the discriminant is $\Delta = -a^2\lambda + [\frac{1}{3}b(1-\lambda) - a\lambda]^2$. Solving the inequality $\Delta \le 0$, we have $\frac{b^2}{(3a+b)^2} \le \lambda < 1$. So when λ satisfies the condition of the proposition, one have

$$L(0, y) \le L(0, a) < 0, \ \forall y \in (a, a+b)$$

namely the curve $L(x, y)$ does not intersect that part of y axis.

Thus, the proposition holds. ∎

Considering the four degenerate control polygon of Figure 3, one can find that: according to the construction method of the formula (1), the curve of the case (a) is both continuous and convex in Ω and satisfies the endpoint interpolation conditions; the one of case (b) includes the line segment l_0 and a conic passing through $\mathbf{p}_0$ and $\mathbf{p}_2$; curves of cases (c) and (d) are not continuous in Ω and have other intersection points besides $\mathbf{p}_0$ and $\mathbf{p}_3$ with the edges of the convex hull Ω. Specifically, considering case (c) as an example, assuming that the control edge $\mathbf{p}_1\mathbf{p}_2$ intersects the control edge $\mathbf{p}_0\mathbf{p}_3$ at the point $\mathbf{p}_4$ while the control edge $\mathbf{p}_3\mathbf{p}_2$ intersects the control edge $\mathbf{p}_0\mathbf{p}_1$ at the point $\mathbf{p}_5$, the curve $L(x, y)$ defined in Ω must be in Δ_{123} and the quadrangle $\mathbf{p}_0\mathbf{p}_5\mathbf{p}_2\mathbf{p}_4$, pass through points $\mathbf{p}_0$, $\mathbf{p}_4$ and $\mathbf{p}_3$, and have a intersection point with the edge $\mathbf{p}_1\mathbf{p}_3$ of Ω. Later, when designing the cubic algebraic spline curve, we need to deal with these cases (especially the last three ones).

§3 Cubic algebraic spline curve

In this section, we discuss that: given $n > 4$ control points (as Figure 6), how to construct a piecewise cubic algebraic curve (named as cubic A-spline curve), so that it has some fine properties like the classic parametric curve.

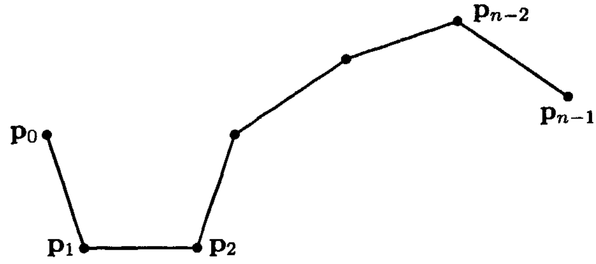

Figure 6: n control points of the A-spline curve.

3.1 Subdivision of the control point sequence

Given n control points $\{\mathbf{p}_i\}_{i=0}^{n-1}$, the final A-spline curve is composed of $n-3$ cubic algebraic curve segments. Here, each segment is constructed from four control points according to the previous construction method of the cubic algebraic interpolation curve. In order that the two adjacent curve segments have G^2 continuous contact with each other, these two segment should have a common control point, namely the last control point of the foregoing segment is exactly the first control point of the posterior segment, and that the third control point of the foregoing segment and the second control point of the posterior segment together with the common point are collinear while these two points are at different sides of the common point. Thus we need $3n-8$ control points in all, written as $\{\hat{\mathbf{p}}_i\}_{i=0}^{3n-9}$. Among these points, there are $n-4$ points named as *virtual* points, which are those common points between each two curve segments, and are easy to be determined for their special locations; the remaining $2n-4$ control points are called *real* points, written as $\{\tilde{\mathbf{p}}_i\}_{i=0}^{2n-5}$, which include the first and last endpoints. Once all the real points have been established, the virtual points can be obtained by using the average of the two corresponding adjacent real points.

First, we introduce a iterative subdivision method of the control points to generate $2n-4$ real points from the n initial control points $\{\mathbf{p}_i\}_{i=0}^{n-1}$. This method is inspired by the de Casteljau algorithm of the Bézier curve and the de Boor algorithm of the B-spline curve. By using the so-called polygon corner cutting idea, the amount of the control points increases by one each time.

Assuming the initial control points are $\{\mathbf{p}_i^0\}_{i=0}^{n_0-1}$, and the control point set after k corner-cutting subdivision is $\{\mathbf{p}_i^k\}_{i=0}^{n_k-1}$, this algorithm can be described as follows:

Algorithm 3.1. (Iterative subdivision of the control points)

1. If $i=0$, $\mathbf{p}_i^k=\mathbf{p}_i^{k-1}$;

2. If $1\le i\le \lceil\frac{n_k-2}{2}\rceil$,

$$\mathbf{p}_i^k=\frac{\mathbf{p}_{i-1}^{k-1}\sum_{j=0}^{i-1}\|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\|+\mathbf{p}_i^{k-1}\sum_{j=i}^{n_{k-1}-2}\|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\|}{\sum_{j=0}^{n_{k-1}-2}\|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\|};$$

3. *If* $\lceil \frac{n_k-2}{2} \rceil \neq \lceil \frac{n_k-1}{2} \rceil$, $i = \lceil \frac{n_k-1}{2} \rceil$,

$$\mathbf{p}_i^k = \frac{\mathbf{p}_{i-1}^{k-1} \sum_{j=0}^{i-1} \|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\| + \mathbf{p}_i^{k-1} \sum_{j=i-1}^{n_{k-1}-2} \|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\|}{\sum_{j=0}^{n_{k-1}-2} \|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\| + \|\mathbf{p}_{i-1}^{k-1}\mathbf{p}_i^{k-1}\|};$$

4. *If* $\lceil \frac{n_k-1}{2} \rceil + 1 \leq i \leq n_k - 2$,

$$\mathbf{p}_i^k = \frac{\mathbf{p}_{i-1}^{k-1} \sum_{j=0}^{i-2} \|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\| + \mathbf{p}_i^{k-1} \sum_{j=i-1}^{n_{k-1}-2} \|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\|}{\sum_{j=0}^{n_{k-1}-2} \|\mathbf{p}_j^{k-1}\mathbf{p}_{j+1}^{k-1}\|};$$

5. *If* $i = n_k - 1$, $\mathbf{p}_i^k = \mathbf{p}_{i-1}^{k-1}$.

This subdivision method takes full advantage of all the edge's length information of the control polygons of the previous step, which is used as the weight value of the new control point. Thus these new control points inherit the properties of the previous control points and have more reasonable position distribution. However, this approach does not convenient for practical applications sometimes, because in order to get $2n-4$ real points from the given n initial given control points, $n-4$ corner-cutting operations are needed to implement which require a lot of computing. So it is inefficient by using this kind of iterative method when the number of the initial given points is quite larger. Hence, we need to turn to other subdivision method which can deal with this situation.

In order to generate all the real points after a few steps for any number of initial given control points, we provide the following subdivision method. This method can be divided into two steps: firstly, using the algorithm 3.2 to get $2n-2$ temporary control points, written as $\{\mathbf{p}_i^*\}_{i=0}^{2n-3}$, from n initial control points $\{\mathbf{p}_i\}_{i=0}^{n-1}$; and then adopting the algorithm 3.3 to get all the real points $\{\tilde{\mathbf{p}}_i\}_{i=0}^{2n-5}$ from these temporary control points at one time.

Algorithm 3.2. (Step 1 of the two-step subdivision method)

1. *If* $i = 0$, $\mathbf{p}_i^* = \mathbf{p}_i$;

2. *If* $i = 1$, $\mathbf{p}_i^* = \dfrac{\mathbf{p}_{i-1} + 2\,\mathbf{p}_i}{3}$;

3. *If i is even and* $2 \leq i \leq 2n-5$, $\mathbf{p}_i^* = \dfrac{5\,\mathbf{p}_{\frac{i}{2}} + \mathbf{p}_{\frac{i+2}{2}}}{6}$;

4. *If i is odd and* $2 \leq i \leq 2n-5$, $\mathbf{p}_i^* = \dfrac{\mathbf{p}_{\frac{i-1}{2}} + 5\,\mathbf{p}_{\frac{i+1}{2}}}{6}$;

5. *If* $i = 2n-4$, $\mathbf{p}_i^* = \dfrac{2\,\mathbf{p}_{\frac{i}{2}} + \mathbf{p}_{\frac{i+2}{2}}}{3}$;

6. *If* $i = 2n-3$, $\mathbf{p}_i^* = \mathbf{p}_{\frac{i+1}{2}}$.

Algorithm 3.3. (Step 2 of the two-step subdivision method)

1. *If* $i = 0$, $\tilde{\mathbf{p}}_i = \mathbf{p}_i^*$;

2. *If* $i = 1$, $\tilde{\mathbf{p}}_i = \mathbf{p}_i^*$;

3. *If* $i = 2$, $\tilde{\mathbf{p}}_i = \dfrac{3\,\mathbf{p}_i^* + 2\,\mathbf{p}_{i+1}^* + \mathbf{p}_{i+2}^*}{6}$;

4. *If* $3 \le i \le 2n-8$, $\tilde{\mathbf{p}}_i = \dfrac{\mathbf{p}_i^* + \mathbf{p}_{i+1}^* + \mathbf{p}_{i+2}^*}{3}$;

5. *If* $i = 2n-7$, $\tilde{\mathbf{p}}_i = \dfrac{\mathbf{p}_i^* + 2\,\mathbf{p}_{i+1}^* + 3\,\mathbf{p}_{i+2}^*}{6}$;

6. *If* $i = 2n-6$, $\tilde{\mathbf{p}}_i = \mathbf{p}_{i+2}^*$;

7. *If* $i = 2n-5$, $\tilde{\mathbf{p}}_i = \mathbf{p}_{i+2}^*$.

By this two-step subdivision method, one can get $2n-4$ real points $\{\tilde{\mathbf{p}}_i\}_{i=0}^{2n-5}$ from the n initial points $\{\mathbf{p}_i\}_{i=0}^{n-1}$, and also the final $3n-8$ control points $\{\hat{\mathbf{p}}_i\}_{i=0}^{3n-9}$ after adding $n-4$ virtual points. The remaining question is how to construct a cubic algebraic spline curve from these control points ?

3.2 Construction of piecewise algebraic curve

Using the previous subdivision method, we can get $3n-8$ control points $\{\hat{\mathbf{p}}_i\}_{i=0}^{3n-9}$ which were further divided into $n-3$ sub-terms, the k-th one of which is composed by those control points from the $(3k-2)$-th to the $(3k+1)$-th. If these four control points form a non-degenerate control polygon or the first case of the degenerate control polygon, one can directly define a cubic algebraic curve segment in the convex hull of these points according to the formula (1), (specially, for the third non-degenerate case, omitting the unwanted close curve); If these four control points form other three cases of the degenerate control polygon, according to the previous analysis, the curve directly constructed by the formula (1) has unreasonable shape, and special treatments are needed. The approach that we would use is to convert the four control points of this sub-term into six new ones by the algorithm 3.2 of the two-step subdivision method, and insert an additional virtual point in the middle position to get a point set of total seven control points, then divide it into two parts respectively and construct the corresponding curve segment in the convex hull of each part according to the formula (1).

Therefore, the generation process of the piecewise algebraic curve can be briefly described as follows.

Algorithm 3.4. (Construction of A-spline curve)

1. *Assume that the amount of the algebraic curve segments is* $m = n-3$, *and* $k = 1$;

2. *Consider the* k*-th sub-term* $P_k = \{\hat{\mathbf{p}}_{3k-3}, \hat{\mathbf{p}}_{3k-2}, \hat{\mathbf{p}}_{3k-1}, \hat{\mathbf{p}}_{3k}\}$ *of the control point set* $\{\hat{\mathbf{p}}_i\}_{i=0}^{3n-9}$, *assume that* $S_{i_1 i_2 i_3}$ *is the area of the triangle that consists of the* i_1*-th,* i_2*-th and* i_3*-th points of this sub-term;*

3. *If* $\mathrm{Min}(S_{123}+S_{134}, S_{124}+S_{234}) \ge \mathrm{Max}(S_{123}, S_{124}, S_{134}, S_{234})$ *and* $S_{124}\cdot S_{134} \ne 0$, *then construct the cubic algebraic curve segment from the point set* P_k *in its convex hull according to the formula (1);*

4. *If* $\mathrm{Min}(S_{123}+S_{134}, S_{124}+S_{234}) < \mathrm{Max}(S_{123}, S_{124}, S_{134}, S_{234})$ *or* $S_{124}\cdot S_{134}=0$, *convert the four control points of the sub-term* P_k *into six new ones by algorithm 3.2, and insert an additional virtual point in the middle position to get a point set of total seven control points, then divide it into two parts respectively and construct the corresponding curve segment in the convex hull of each part according to the formula (1). While the amount of curve segments increases by one, that is* $m \leftarrow (m+1)$;

5. *If* $k = n-3$, *end the process and output the amount* m *of all the curve segments; otherwise let* $k \leftarrow (k+1)$, *go to step 2.*

Here, we did not discuss the situation that some of the control points were superposed. Because the previous construction method of the cubic algebraic interpolation curve is based on control edges, it is not applicable and needs additional processes when there are superposed points. Considering the two-step subdivision method before, one can find that if the given initial points are not superposed, there is no superposed point among the control points $\{\hat{\mathbf{p}}_i\}_{i=0}^{3n-9}$ after the subdivision. In the following examples, we always assume that there is no superposed point in the initial control points .

3.3 Properties of the A-spline curve

Next, we present and analyze some of the properties of the cubic A-spline curve constructed by the above method.

First, it is clear that the cubic A-spline curve constructed by the above method is geometric invariant (affine invariant). Because this method is based on the control edges, as long as the relative position of the control points remains unchanged, the shape of the curve segment in the convex hull of these control points remains unchanged.

Second, this kind of cubic A-spline curve has convex hull property, namely the curve segment must be inside the convex hull of the initial control points $\{\mathbf{p}_i\}_{i=0}^{n-1}$. The reason is that the $3n-8$ final control points $\{\hat{\mathbf{p}}_i\}_{i=0}^{3n-9}$ by the previous two-step subdivision method are all inside the convex hull of the initial control points, and according to the properties of the cubic algebraic interpolation curve, for the $n-3$ sub-term subdivided from the initial control points, each corresponding curve segment is inside the convex hull of the corresponding four control points. Thus the final piecewise cubic algebraic curve is actually composed of those curve segments each one of which is inside the convex hull of the four control points of the corresponding sub-term, so the convex hull of all the sub-terms can be regarded as a more precise definition region of the cubic A-spline curve.

Moreover, this kind of cubic A-spline curve also has local control property, namely changing a initial control point only effects a local part of the final curve. The reason is that according to the two-step subdivision method, when one point of the set $\{\mathbf{p}_i\}_{i=0}^{n-1}$ is changed. At most there are four points of the temporary control points $\{\mathbf{p}_i^*\}_{i=0}^{2n-3}$ are changed correspondingly, and at most six points of the real point set $\{\tilde{\mathbf{p}}_i\}_{i=0}^{2n-5}$ are changed correspondingly, while other points remain unchanged. That is, the change of an initial point only spreads to the corresponding finite neighborhood of it. So at worst situation, four segments of the final A-spline curve are changed because of the change of that initial point.

From the algorithms 3.2 and 3.3, it is easy to see that there exists a linear relationship between the n control points $\{\mathbf{p}_i\}_{i=0}^{n-1}$ and those junction points $\{\hat{\mathbf{p}}_{3i}\}_{i=0}^{3n-9}$. Assuming that the junction points $\{\hat{\mathbf{p}}_{3i}\}_{i=0}^{3n-9}$ are given, we can use our curve construction method in an inverse way to obtain an G^2 piecewise algebraic curve to interpolate these junction points. But some end condition should be added since the number of junction points $\{\hat{\mathbf{p}}_{3i}\}_{i=0}^{3n-9}$ is $n-2$.

Finally, we analyze how to set up the parameter λ to adjust the curve's shape. According to the analysis of the cubic algebraic interpolation curve, for a sub-term of the A-spline curve, the larger λ is, the closer the curve segment's shape is to the straight line. For the sake of simplicity, one can always use the same value of the parameter λ for each curve segment. But in fact, we can use different value of λ for different segment to adjust the A-spline curve's shape more fairing. In the following section, we will determine the shape parameter λ by minimizing the corresponding smoothing energy of each curve segment as follows.

$$E(\lambda) = \frac{1}{2}\int_{\tilde{\Omega}} (L_{xx}^2 + L_{yy}^2 + 2L_{xy}^2)\,dxdy \tag{2}$$

Where, the integral region $\tilde{\Omega}$ is the convex hull of the four control points of that sub-term. In fact, for each sub-term $\{\hat{\mathbf{p}}_{3i-3}, \hat{\mathbf{p}}_{3i-2}, \hat{\mathbf{p}}_{3i-1}, \hat{\mathbf{p}}_{3i}\}$, the energy function $E(\lambda)$ is a quadratic polynomial of λ. Thus it is simple to solve the minimization problem.

§4 Examples

In this section, we provide several examples of the cubic A-spline curve constructed by our method.

Example 1 Assuming the given five control points are:

$$\{(0,0),(1,2),(3,2.6),(5,2),(6,0)\}.$$

The cubic A-spline curve defined in the convex hull of the control points is shown as Figure 7. Here, solid lines denote the initial control polygon, and the dotted lines denote the control polygon after subdivisions.

The curve is divided into two segments, for symmetry, the two shape parameters are all $\lambda = 0.708324$.

The expressions of the two cubic algebraic curves are:

$$\begin{aligned} f_1(x,y) =& 0.159539x^3 - 0.769380yx^2 + 0.333722x^2 + 1.10364y^2x \\ & - 0.857875yx + 0.442405x - 0.462441y^3 + 0.345507y^2 \\ & - 0.221203y + 7.75448\times 10^{-48}, \\ f_2(x,y) =& -0.159539x^3 - 0.769380yx^2 + 3.20543x^2 - 1.10364y^2x \\ & + 10.0904yx - 21.6773x - 0.462441y^3 + 6.96736y^2 \\ & - 33.0661y + 49.1289. \end{aligned}$$

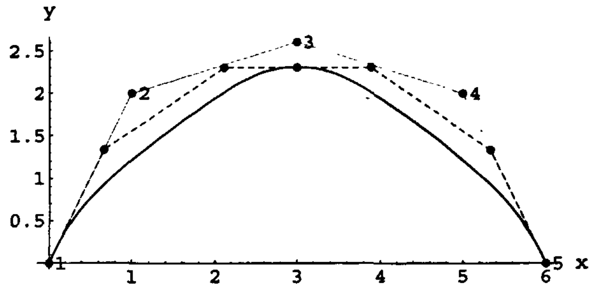

Figure 7: A cubic A-spine curve of five control points.

Example 2 Assuming the given six control points are:

$$\{(0,1.5),(0.5,3),(2.5,3),(2.5,0),(4.5,0),(5,1.5)\}.$$

The cubic A-spline curve defined in the convex hull of the control points is shown as Figure 8.

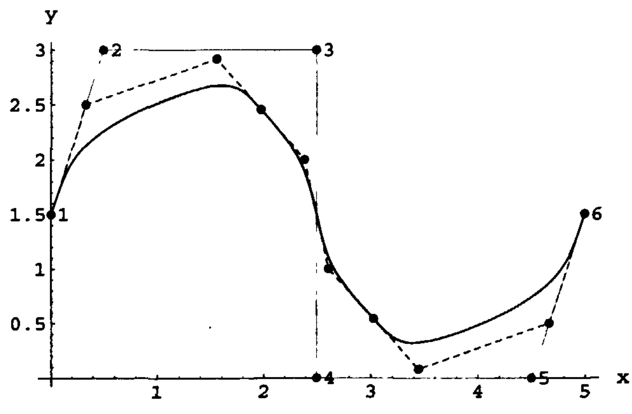

Figure 8: A cubic A-spine curve of six control points.

The whole curve is divided into three segments, and their corresponding control polygon are the first case of the non-degenerate control polygon, the second case of the non-degenerate control polygon and the first case of the non-degenerate control polygon in turn. The corresponding values of λ are 0.435883, 0.486906 and 0.435883. Their expressions are:

$$\begin{aligned} f_1(x,y) =& -0.0913879x^3 + 0.0765831yx^2 - 0.0837695x^2 + 0.716863y^2x \\ & -3.52873yx + 4.62479x - 0.430742y^3 + 2.39426y^2 \\ & -4.59016y + 2.95190, \\ f_2(x,y) =& -0.0530151x^3 + 0.0188762yx^2 + 0.369299x^2 + 0.0396800y^2x \\ & -0.213421yx - 0.795526x - 0.00429716y^3 - 0.0798629y^2 \\ & +0.379383y + 0.534343, \\ f_3(x,y) =& -0.0913879x^3 + 0.0765831yx^2 + 1.22484x^2 + 0.716863y^2x \\ & -1.53829yx - 4.90392x - 0.430742y^3 - 2.10190y^2 \\ & +3.92223y + 6.22234. \end{aligned}$$

Example 3 Assuming the given seven control points are:

$$\{(0,2),(0.5,0),(2.,0),(2.5,2),(4.,3.25),(5.5,3.5),(7,2.5)\}.$$

The cubic A-spline curve defined in the convex hull of these control points is shown as Figure 9.

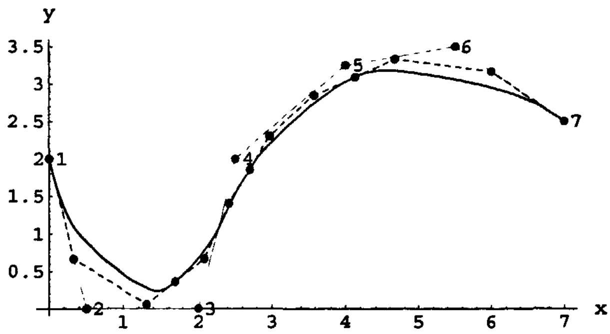

Figure 9: A cubic A-spine curve of seven control points.

Here, the whole curve is divided into five segments, where the control polygon of the second term is the third case of the degenerate control polygon and is further divided into two subterms.

All these above examples use the two-steps subdivision method of the control points. In fact, for the cases of less initial control points, it is more reasonable to select the iterative subdivision method (algorithm 3.1) to make better use of the edge information of the control polygon. This is because: first of all, for less control points, the efficiency of this iterative algorithm can be accepted (since the total number of iteration is $n-4$ here); Secondly, by the two-step subdivision method, the local property and smoothness of the final curve are not good with a few initial control points. In other words, for a relatively small number of control points, we can get a more reasonable final A-spline curve with the acceptable efficiency and non-local property by the iterative subdivision algorithm. For example, the results of example 2 and 3 by using the iterative subdivision algorithm are shown in Figure 10.

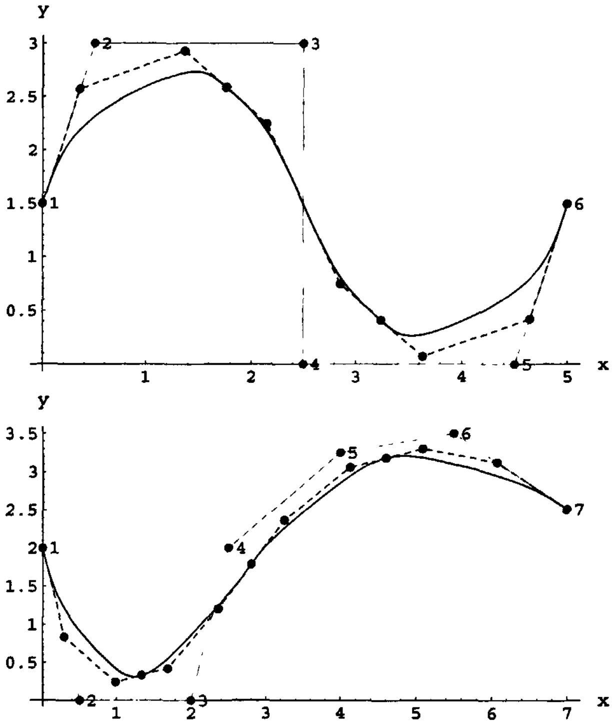

Figure 10: The results by using the iterative subdivision algorithm.

§5 Conclusion

In this paper, we proposed a construction method of the planar cubic algebraic spline curve with endpoint interpolation conditions and a specific analysis of its properties. The piecewise cubic algebraic curve has G^2 continuous contact with the control polygon at two endpoints and is G^2 continuous between each segments of itself. In addition, for each segment of the curve, we modified its shape by using a reasonable value of the parameter λ. The process of this method seems simple and clear, and provides a new way of thinking to design implicit curves.

References

[1] C L Bajaj, J Chen and G L Xu. *Modeling with cubic A-patches*, ACM Trans Graphics, 1995, 14 : 103-133.

[2] J Bloomenthal. *Polygonization of Implicit Surfaces*, CAGD, 1988, 5(4) : 341-355.

[3] F L Chen, J M Zheng and T W Sederberg. *The mu-basis of a rational ruled surface*, CAGD, 2001, 18 : 61-72.

[4] F L Chen and T W Sederberg. *A new implicit representation of a planar rational curve with high order singularity*, CAGD, 2002, 19 : 151-167.

[5] F L Chen. *Reparametrization of a rational ruled surface using μ-basis*, CAGD, 2003, 20(1) : 11-17.

[6] F L Chen and W W Wang. *The μ-basis of a planar rational curve–properties and computation*, Graphical Models, 2003, 64 : 268-381.

[7] D Cox, T W Sederberg and F Chen. *The moving line ideal basis of planar rational curves*, CAGD, 1998, 15 : 803-827.

[8] Y Y Feng, F L Chen, J S Deng, C S Chen and X Tang. *Constructing piecewise algebraic blending surfaces*, in Geometric Computation (F. Chen and D. Wang, eds.), World Scientific, Singapore New Jersey, 2003, 34-64.

[9] G Farin. *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guided*, Academic Press, San Diego, CA, 1997.

[10] P Hanrahan. *Ray Tracing Algebraic Surfaces*, Computer Graphics (Proceedings of SIGGRAPH 83), 1983, 17(3) : 83-90.

[11] E Hartmann. *Blending of implicit surfaces with functional splines*, CAD, 1990, 22 : 500-506.

[12] E Hartmann and Y Y Feng. *On the convexity of functional splines*, CAGD, 1993, 10 : 127-142.

[13] E Hartmann. *Implicit G^n-blending of vertices*, CAGD, 2001, 18 : 267-285.

[14] C Hoffmann. *Geometric and Solid Modeling: An Introduction*, San Francisco, Morgan Kaufmann, 1989.

[15] C Hoffmann. *Implicit Curves and Surfaces in Computer-Aided Geometric Design*, IEEE Computer Graphics and Applications, 1993, 13(1) : 79-88.

[16] J Hoschek and D Lasser. *Fundamentals of Computer Aided Geometric Design*, A K Peters Wellesley, Massachusetts, 1993.

[17] J Li, J Hoschek and E Hartmann. *G^{n-1} functional splines for interpolation and approximation of curves, surfaces and solids*, CAGD, 1990, 7 : 209-220.

[18] L Piegl and W Tiller. *The NURBS book (2nd ed.)*, Springer-Verlag New York Inc, 1997.

[19] T W Sederberg. *Implicit and Parametric Curves and Surfaces for Computer-Aided Geometric Design*, Ph. D. dissertation, Lafayette, IN, Purdue Univ, 1983.

[20] T W Sederberg. *Piecewise Algebraic Curves*, CAGD, 1984, 1 : 241-255.

[21] T W Sederberg and F L Chen. *Implicitization Using Moving Curves and Surfaces*, Computer Graphics, Annual Conference Series (Proceedings of SIGGRAPH 95), 1995, 301-308.

[22] G J Wang, G Z Wang, J M Zheng. *Computer-Aided Geometric Design(in Chinese)*, Higher Education Press & Springer Press, 2001.

[23] J Warren. *Blending algebraic surfaces*, ACM Trans Graphics, 1989, 8(4) : 263-278.

[1] Faculty of Science, Ningbo University, Ningbo, Zhejiang 315211, China. Email: xuchendong@nbu.edu.cn

[2] Department of Mathematics, University of Science and Technology of China, Hefei, Anhui 230026, China