ELSEVIER

Computer-Aided Design 36 (2004) 461–472

COMPUTER-AIDED DESIGN

www.elsevier.com/locate/cad

Curve fitting and fairing using conic splines

Xunnian Yang*

Department of Mathematics, Zhejiang University, Yuquan, Hangzhou 310027, China

Received 17 December 2002; received in revised form 3 June 2003; accepted 4 June 2003

Abstract

We present an efficient geometric algorithm for conic spline curve fitting and fairing through conic arc scaling. Given a set of planar points, we first construct a tangent continuous conic spline by interpolating the points with a quadratic Bézier spline curve or fitting the data with a smooth arc spline. The arc spline can be represented as a piecewise quadratic rational Bézier spline curve. For parts of the G^1 conic spline without an inflection, we can obtain a curvature continuous conic spline by adjusting the tangent direction at the joint point and scaling the weights for every two adjacent rational Bézier curves. The unwanted curvature extrema within conic segments or at some joint points can be removed efficiently by scaling the weights of the conic segments or moving the joint points along the normal direction of the curve at the point. In the end, a fair conic spline curve is obtained that is G^2 continuous at convex or concave parts and G^1 continuous at inflection points. The main advantages of the method lies in two aspects, one advantage is that we can construct a curvature continuous conic spline by a local algorithm, the other one is that the curvature plot of the conic spline can be controlled efficiently. The method can be used in the field where fair shape is desired by interpolating or approximating a given point set. Numerical examples from simulated and real data are presented to show the efficiency of the new method.

Keywords: Conic spline; NURBS; Data fitting; Conic arc scaling; Fairness

1. Introduction

Data fitting by parametric curves can be used in wide fields such as pattern recognition, image processing, statistical data analysis and many other industrial applications. While most curve fitting algorithms try to construct a smooth curve passing through or near the given points, in the field of computer aided geometric design, it is often desirable to fit a point set by a curve close to the points and which has a please shape [17,26,28]. Even more, the curvature plot of the fitting curve should consist of as few as possible monotone pieces or with prescribed curvatures at selected points [8]. In this paper we address the problem of data fitting by conic spline for which the shape of the result curve can be controlled efficiently.

With the capability of representing conics and freeform curves and surfaces in a unified way, NURBS have become de facto the state of art in computer aided design [7,21]. The problems of data fitting by B-spline curves and surfaces have been studied extensively in the literature. To fit a point set by a B-spline curve [15,18], knots and parameters corresponding to the data points are often given ahead, then least square fitting of the data is often applied and the control points of the curve can be obtained by solving a system of linear equations. There are also several methods to fit the point data by a rational B-spline curve [3,16,27], but how to set the weights for a fair NURBS curve with controllable curvature plot is still a challenging problem.

Data fitting by arc spline curves [12,22,30] is another important method used frequently in tool path description for machining, robot path planning as well as shape modeling. When fitting a point set by an arc spline, every segment can be constructed as an elementary geometric problem. Then the shape and the number of arcs can be controlled efficiently. Even more, an arc spline can be constructed in a fairness manner and the arc segments of the spline can be reduced efficiently within a prescribed tolerance [29]. One main shortcoming of the arc spline is that its curvature is not continuous, so it cannot be used for high quality shape modeling.

With arc spline as a special case, conic spline curves and surfaces own a lot of elegant properties which make them a powerful tool for shape modeling. A conic segment can both

* Tel.: +86-571-87951609; fax: +86-571-87951428.
E-mail address: yxn@zju.edu.cn (X. Yang).

0010-4485/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/S0010-4485(03)00119-2

be represented as an algebraic curve segment and as a rational Bézier curve [14]. The parametrization of a conic segment can be controlled efficiently for high order continuity purpose or surface skinning in homogeneous space [4,5]. The offset and the CNC interpolator for the conics can be computed efficiently [9,10]. Farin [6] and Pottmann [23] have presented a global and a local method for the construction of curvature continuous conic spline curve separately. These two methods either have the limitation that a change of the local parameter may influence the whole shape or the segments number are approximately the twice of the interpolating points. Schaback [25] has presented a method of conic spline interpolation by solving a global system of non-linear equations. Conic spline can also be used to fit a piecewise linear curve or another smooth curve [1,19,20,24], but how to control the curvature plot of the conic spline curve have not been addressed.

Recently, Ahn and Kim [2] and Frey and Field [11] have presented independently the same result that shows whether or not a given conic segment has monotone curvature. This result can be used for the design of a single conic segment. In this paper we will show how to construct a curvature continuous conic spline curve from a tangent continuous conic spline and fair the conic spline by removing unwanted curvature extrema. We assume here that the initial G^1 conic spline curve be a tangent continuous quadratic Bézier spline curve interpolating a given point sequence or an arc spline curve approximating a noisy point set [30]. For the latter case, each arc can be expressed as a quadratic rational Bézier segment or two of them if the angle is larger than π. Then for every two adjacent G^1 connected conic segments without inflection, the tangent at the joint point and the two weights of the Bézier representation can be reset so that the two new conic segments are curvature continuous while the tangents and curvatures at the other two ends are fixed. To obtain a fair conic spline curve, the additional curvature extrema within some conic segments or at the joint point of two adjacent conics can be removed by scaling the weights or moving the joint points of selected conics. By combining the method of tangent displacement, weight scaling and end point moving, an efficient algorithm for fair conic spline construction is presented.

The organization of the paper is as follows. In Section 2 we will give a brief introduction of conics and the conditions for various curvature distribution are also presented. Algorithm for the construction of a curvature continuous conic spline from an initial G^1 spline is presented in Section 3. In Section 4 we will discuss how to achieve a fair conic spline curve by resetting the conic ends and scaling the weights of selected arcs. Examples are presented in Section 5 and we conclude the paper in Section 6.

2. Preliminary information on conics

The standard form for a rational quadratic Bézier curve is

$$R(t) = \frac{R_0(1-t)^2 + 2R_1wt(1-t) + R_2t^2}{(1-t)^2 + 2wt(1-t) + t^2},$$

where R_i $(i = 0,1,2)$ are the control points of the Bézier curve and w is the middle weight. It is well known that any rational quadratic Bézier curve is a conic segment [14]. The curve is a segment of parabola when the weight $w = 1$, and the curve is a segment of ellipse or a segment of hyperbola when w is less than or larger than 1, respectively. If the control polygon $R_0R_1R_2$ forms an isosceles triangle, let the angle $\angle R_1R_0R_2 = \theta$ and set the weight $w = \cos\theta$, then the quadratic rational Bézier curve is a circular arc.

The unsigned curvature for the curve $R(t)$ can be computed as

$$k(t) = \frac{\|R'(t) \times R''(t)\|}{\|R'(t)\|^3},$$

then the curvatures at the two ends of the conic segment are

$$k(0) = \frac{1}{2}\frac{\|(R_1 - R_0) \times (R_2 - R_1)\|}{w^2\|R_1 - R_0\|^3},$$

and

$$k(1) = \frac{1}{2}\frac{\|(R_1 - R_0) \times (R_2 - R_1)\|}{w^2\|R_2 - R_1\|^3},$$

respectively. From the curvature formula, we can see that the two end curvatures will be scaled simultaneously if the weight w has been changed.

Suppose that the three control points R_0, R_1 and R_2 are not collinear and $w > 0$, then the conic segment will not degenerate to a line. As indicated by Ahn and Kim [2] and Frey and Field [11], a quadratic rational Bézier curve can be with monotone curvature plot if and only if the middle control point R_1 lies in a region defined by the boundary points and the middle weight.

Theorem 1. *Let*

$$U = \frac{R_2 - R_0}{\|R_2 - R_0\|},$$

and

$$r = \frac{\|R_2 - R_0\|}{4w^2},$$

then we can define two circles O_0 and O_1 both with radius r and centered at $O_0 = R_0 + rU$ and $O_1 = R_2 - rU$, respectively. If the control point R_1 lies outside both of the two circles or inside both of the circles, then the curvature plot of the conic is with a local maximum value or a local minimum value. If the control point R_1 lies inside one of the two circles but outside the other one, then the curvature plot of the conic segment is monotone when $w^2 > 1/2$ and has *one local maximum value and one local minimum value when* $w^2 < 1/2$.

Proof. See Refs. [2,11] □.

For the convenience of a fair curve design, we can assign an integer sign $Kt[i]$ to indicate the curvature type of conic C_i (see Fig. 1). We set $Kt[i] = 2$ when the control point R_1 lies outside both of the two circles O_0 and O_1. In this case the curvature plot has a local maximum. If R_1 lies inside both of the two circles O_0 and O_1, there is a local minimum value within the curvature plot, then we set $Kt[i] = 0$. If R_1 lies outside O_0 but inside O_1 we set the curvature type $Kt[i] = 1$. If R_1 lies inside O_0 but outside O_1 the curvature type $Kt[i] = -1$. When $w^2 > 1/2$, and if the curvature type of a conic section is 1 or -1, the curvature plot of the conic is monotone increasing or monotone decreasing, respectively.

To classify the curvature type of a conic segment, we will have to judge the relationship between the middle control point R_1 and the two circles O_0 and O_1. When R_1 lies on the circle O_0, we have

$$2\frac{\|R_2 - R_0\|}{4w^2}\cos\alpha_0 = \|R_1 - R_0\|. \qquad (1)$$

If R_1 lies on the circle O_1, we have

$$2\frac{\|R_2 - R_0\|}{4w^2}\cos\alpha_1 = \|R_2 - R_1\|. \qquad (2)$$

When each conic segment is represented as a quadratic rational Bézier curve, the conic spline curve consisting of m segments of smooth connected conic pieces can then be transformed into a rational B-spline curve [21]. The knot vector for the NURBS curve can be set as $\tau = \{0, 1, \ldots, m\}$, where the multiplicity for each interior knot is 2 and the multiplicity of the first and the last knots are both 3. The control polygon of the rational B-spline curve is $P_0P_1P_2\ldots P_{2m-1}P_{2m}$ where $P_{2i-2}P_{2i-1}P_{2i}$ are just the control polygon of the conic C_i $(i = 1,2,\ldots,m)$. Every two adjacent conics C_i and C_{i+1} are jointed at point P_{2i}, and the points P_{2i-1}, P_{2i} and P_{2i+1} are collinear when the two conics are tangent continuous at the joint point. Since the weights associated with the control polygon of the conic C_i are 1, w_{2i-1} and 1, then the weights of the NURBS curve are just the weights of the conics, where w_{2i-1} is the weight corresponding to the control point $P_{2i-1}, (i = 1,2,\ldots,m)$. The rest weights $w_0 = w_2 = \cdots = w_{2m} = 1$.

3. G^2 conic spline fitting

Though there are several algorithms presented in the literature for fitting a set of points by a conic spline curve [6,19,23,25], but it is still a challenging problem to fit the data by a fair and curvature continuous conic spline curve directly. In this section we fit a conic spline curve to a point set by first fitting the points with a tangent continuous quadratic Bézier spline or an arc spline and then construct a curvature continuous conic spline from the original G^1 conic spline.

With a set of ordered planar points, an initial G^1 conic spline can be obtained by interpolating the points with piecewise quadratic Bézier curves. If the points are noisy, we can first fair and fit the data by an arc spline within a prescribed tolerance [30]. When the data is fitted by an arc spline, the number of arcs can be reduced efficiently within another tolerance [29]. For the convenience of representing circular arcs as rational Bézier curves, we assume that the central angle for an arc is less than π. If the central angle for

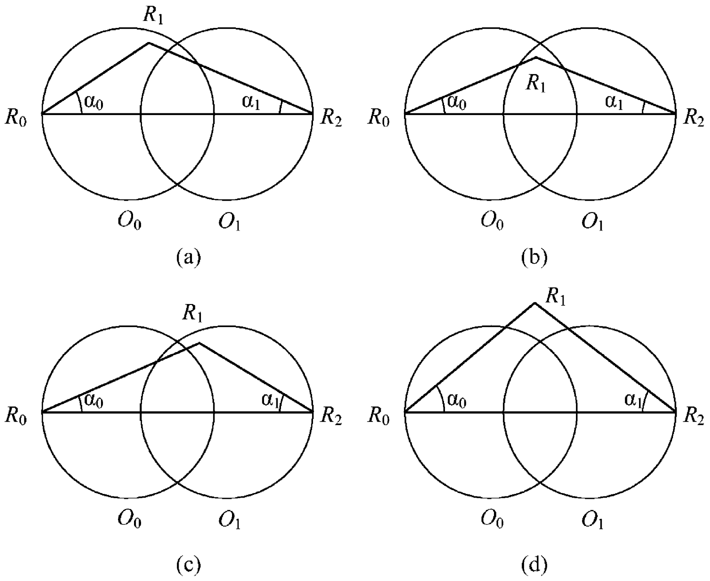

Fig. 1. Curvature type for a conic segment in Bézier form: (a) $Kt[i] = -1$, (b) $Kt[i] = 0$, (c) $Kt[i] = 1$, (d) $Kt[i] = 2$.

an arc segment is larger that π, it can be divided into two segments.

We assume that the G^1 conic spline obtained above consists of m segments, namely, $C_1, C_2, \ldots, C_m$. The center angles for the arc spline are $2\theta_1, 2\theta_2, \ldots, 2\theta_m$, respectively. Then, each conic segment can be expressed as a rational quadratic Bézier curve,

$$R_i(t) = \frac{P_{2i-2}(1-t)^2 + 2P_{2i-1}w_{2i-1}t(1-t) + P_{2i}t^2}{(1-t)^2 + 2w_{2i-1}t(1-t) + t^2},$$

where the weight $w_{2i-1} = \cos\theta_i$ for the arcs. If it is a quadratic Bézier curve, the weight can be chosen as $w_{2i-1} = 1$. Because any two adjacent arcs C_i and C_{i+1} ($i = 1, 2, \ldots, m-1$) are tangent continuous, then the control points P_{2i-1}, P_{2i} and P_{2i+1} are collinear.

To construct a curvature continuous conic spline from the G^1 conic spline, the weights and the tangent at the joint point for every two adjacent arc segments or conic segments can be reset to obtain a new pair of G^2 connected conic segments. As shown in Fig. 2, C_i and C_{i+1} are two adjacent conics which joint at point P_{2i}, then we can move the point $\bar{P}_{2i-1}$ along the line $P_{2i-2}P_{2i-1}$ and move the point $\bar{P}_{2i+1}$ along the line $P_{2i+1}P_{2i+2}$ so that the line connecting the two new points $\bar{P}_{2i-1}$ and $\bar{P}_{2i+1}$ still passes through the joint point P_{2i}. In another point of view, we can obtain two new control points $\bar{P}_{2i-1}$ and $\bar{P}_{2i+1}$ by rotating the line $P_{2i-1}P_{2i+1}$ around the fixed point P_{2i} with an angle θ and compute the intersection points with the lines $P_{2i-2}P_{2i-1}$ and $P_{2i+1}P_{2i+2}$, respectively.

Let the unsigned angle between the vector $P_{2i-2}P_{2i-1}$ and the vector $P_{2i-1}P_{2i}$ be α, the unsigned angle between the vector $P_{2i}P_{2i+1}$ and the vector $P_{2i+1}P_{2i+2}$ be β. Because the new control point $\bar{P}_{2i-1}$ lies on the line $P_{2i-2}P_{2i-1}$, then we can assume that $\bar{P}_{2i-1} - P_{2i-2} = \lambda(P_{2i-1} - P_{2i-2})$. In a similar way, the new control point $\bar{P}_{2i+1}$ lies on the line $P_{2i+2}P_{2i+1}$, and we have $P_{2i+2} - \bar{P}_{2i+1} = \mu(P_{2i+2} - P_{2i+1})$. Then, if we have the scaling coefficients λ and μ, we will obtain the two new control points for the two conics immediately. From the triangle $\Delta P_{2i-1}P_{2i}\bar{P}_{2i-1}$, we have

$$\frac{\|P_{2i-1} - P_{2i-2}\|(\lambda - 1)}{\sin\theta} = \frac{\|P_{2i} - P_{2i-1}\|}{\sin(\alpha+\theta)}. \tag{3}$$

Based on the triangle $\Delta P_{2i}\bar{P}_{2i+1}P_{2i+1}$, we have

$$\frac{\|P_{2i+1} - P_{2i}\|}{\sin(\beta-\theta)} = \frac{\|P_{2i+2} - P_{2i+1}\|(\mu - 1)}{\sin\theta}. \tag{4}$$

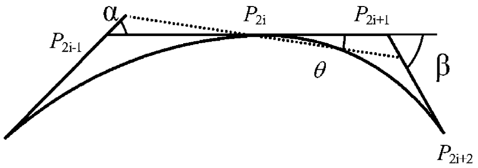

Fig. 2. Construct a pair of G^2 continuous conics by displacing the common tangent.

From Eqs. (3) and (4) we can express the parameters λ and μ as two functions of the parameter θ,

$$\lambda = 1 + \frac{\|P_{2i} - P_{2i-1}\|}{\|P_{2i-1} - P_{2i-2}\|}\frac{\sin\theta}{\sin(\theta+\alpha)}, \tag{5}$$

and

$$\mu = 1 - \frac{\|P_{2i+1} - P_{2i}\|}{\|P_{2i+2} - P_{2i+1}\|}\frac{\sin\theta}{\sin(\beta-\theta)}. \tag{6}$$

If we replace the control point P_{2i-1} of the conic C_i by a new control point $\bar{P}_{2i-1}$ and keep the weights of the curve unchanged, then the end curvatures of the conic can be computed as $\bar{k}_i(0) = (1/\lambda^2)k_i(0)$ and

$$\bar{k}_i(1) = \frac{\lambda\sin^3(\alpha+\theta)}{\sin^3\alpha}k_i(1),$$

where $k_i(0)$ and $k_i(1)$ are the end curvatures of the original conic segment. In a similar way, the new end curvatures for the conic C_{i+1} can be obtained as

$$\bar{k}_{i+1}(0) = \frac{\mu\sin^3(\beta-\theta)}{\sin^3\beta}k_{i+1}(0),$$

and $\bar{k}_{i+1}(1) = (1/\mu^2)k_{i+1}(1)$.

To be sure that the curvature at the first end point of the new conic C_i be fixed or equal to a predefined value such as the second end curvature of its former conic segment, the original weight w_{2i-1} should be reset as a new one $\bar{w}_{2i-1}$. Then we have

$$\frac{1}{\lambda^2}k_i(0)\frac{w_{2i-1}^2}{\bar{w}_{2i-1}^2} = k_{i-1}(1). \tag{7}$$

If $i = 1$ or P_{2i-2} is an inflection, we may set $k(0) = k_i(0)$ for Eq. (7). Then the new weight can be set as

$$\bar{w}_{2i-1} = \sqrt{\frac{k_i(0)}{k_{i-1}(1)}}\frac{w_{2i-1}}{\lambda}.$$

At the same time, the curvature at the other end of the conic C_i has been changed as

$$\frac{\lambda\sin^3(\alpha+\theta)}{\sin^3\alpha}k_i(1)\frac{w_{2i-1}^2}{\bar{w}_{2i-1}^2}.$$

In the same way, the weight w_{2i+1} for the conic C_{i+1} can be changed as $\bar{w}_{2i+1} = (w_{2i+1})/\mu$, and the end curvatures of the new conic are

$$\frac{\mu^3\sin^3(\beta-\theta)}{\sin^3\beta}k_{i+1}(0),$$

and $k_{i+1}(1)$. To be sure that the new end curvatures at the joint point of the two conics C_i and C_{i+1} are equal, we need

$$\frac{\lambda\sin^3(\alpha+\theta)}{\sin^3\alpha}k_i(1)\frac{w_{2i-1}^2}{\bar{w}_{2i-1}^2} = \frac{\mu^3\sin^3(\beta-\theta)}{\sin^3\beta}k_{i+1}(0). \tag{8}$$

Substituting Eq. (7) into Eq. (8), we have

$$\lambda \sin(\alpha + \theta) = \mu \sin(\beta - \theta) \frac{\sin \alpha}{\sin \beta} \left(\frac{k_{i+1}(0) k_i(0)}{k_{i-1}(1) k_i(1)} \right)^{1/3}. \qquad (9)$$

By expressing λ and μ as the functions of θ, Eq. (9) can be changed into

$$\sin(\alpha + \theta) - s_2 \sin(\beta - \theta) + (s_0 + s_1 s_2) \sin \theta = 0, \qquad (10)$$

where

$$s_0 = \frac{\|P_{2i} - P_{2i-1}\|}{\|P_{2i-1} - P_{2i-2}\|},$$

$$s_1 = \frac{\|P_{2i+1} - P_{2i}\|}{\|P_{2i+2} - P_{2i+1}\|},$$

and

$$s_2 = \frac{\sin \alpha}{\sin \beta} \left(\frac{k_{i+1}(0) k_i(0)}{k_{i-1}(1) k_i(1)} \right)^{1/3}.$$

By expanding the sine function, we have $A \sin \theta + B \cos \theta = 0$, where $A = \cos \alpha + s_2 \cos \beta + s_0 + s_1 s_2$ and $B = \sin \alpha - s_2 \sin \beta$. The solution to Eq. (10) is $\theta = \arctan(-B/A)$.

When θ is obtained, the two scaling factors λ and μ are given by Eqs. (5) and (6), and the new control points and the new weights for the conics with G^2 continuity are obtained. If the curvatures $k_i(0)$ and $k_{i+1}(1)$ for the two conics are fixed, the left and the right curvatures at the point P_{2i} will become as

$$\bar{k}_i(1) = \frac{\lambda^3 \sin^3(\alpha + \theta)}{\sin^3 \alpha} k_i(1),$$

and

$$\bar{k}_{i+1}(0) = \frac{\mu^3 \sin^3(\beta - \theta)}{\sin^3 \beta} k_{i+1}(0).$$

It can be easily concluded from the equation $\bar{k}_i(1) = \bar{k}_{i+1}(0)$ that when the inequality $k_i(1) < k_{i+1}(0)$ holds, θ will be a positive number, and then we have $\lambda > 1$ and $\mu < 1$. Consequently, we have $\bar{k}_i(1) > k_i(1)$ and $\bar{k}_{i+1}(0) < k_{i+1}(0)$. When $k_i(1) > k_{i+1}(0)$, the directions of the inequalities should be inversed. With this fact, we can smooth a G^1 conic spline by a curvature continuous conic spline, and we can also obtain a G^2 connected conic spline curve by a local algorithm. Even more, if the initial conic spline interpolates a given point set, the G^2 conic spline also interpolate the same point set.

4. Fair curve construction through conic scaling

Though the conic spline constructed above is G^2 continuous except at inflection points, it may still be possible to reduce the number of curvature extrema. In this section we will discuss how to reduce the curvature extrema by moving the positions of the joint points (Section 4.1) and scaling the weights (Section 4.2).

4.1. Conic arc scaling with control points resetting

In the first case, we show that if the curvature at the joint point of two adjacent conics is an unwanted extremum, it can be removed or smoothed by resetting the position of the joint point. In this paper we define a joint point with an unwanted curvature extremum if the curvature at the point is a local minimum or a local maximum and the two joining conic segments own at least one another curvature extremum within the conic segments or at the other two ends. For example, if two adjacent conic segments both have local maximum curvature extrema within the conics, the curvature at the joint point is a local minimum one. In this case the curvature types of the two conics are $(2,2)$. If the curvature types of a pair of adjacent conics are one of the types $(2,2)$, $(2,1)$, $(-1,2)$, the curvature at the joint point is an unwanted minimum. If the two conics have monotone decreasing and monotone increasing curvature plot like $(-1,1)$ and at least one curvature of the two ends of the conic pair is a local extremum, then the curvature at the joint point is also defined as an unwanted local extremum. In a similar way we can define the joint point with an unwanted maximum curvature just by replacing 2 by 0, 1 by -1 and -1 by 1 of the local minimum curvature patterns.

If the curvature at the joint point of conics C_i and C_{i+1} is a local curvature extremum as defined above, we should then move the joint point P_{2i} along the normal direction at the point (see Fig. 3), so that the number of curvature extrema within the two conics and at the joint point can be reduced or the curvature difference of adjacent curvature extrema can be smoothed. When the point P_{2i} has been moved, the curvatures of the two conic segments at the point are generally not equal to each other any more. We can then construct a G^2 connected conic pair by rotating the common tangent with the new joint point fixed.

Let V be the unit vector paralleling the normal at the joint point P_{2i} and lying at the opposite side of the tangent line with respect to the conic curve itself, we can then push the joint point along the vector V to a new position as $\bar{P}_{2i} = P_{2i} + hV$. Consequently, the control points P_{2i-1} and P_{2i+1} will be pulled to two new positions along the lines $P_{2i-2}P_{2i-1}$ and $P_{2i+2}P_{2i+1}$, respectively. These two new points can also be defined

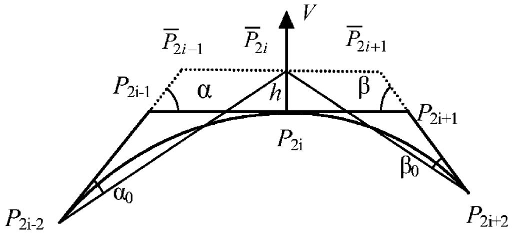

Fig. 3. Fair a pair of conics by disturbing the joint point.

by the equations $\bar{P}_{2i-1} - P_{2i-2} = \lambda_0(P_{2i-1} - P_{2i-2})$ and $\bar{P}_{2i+1} - P_{2i+2} = \mu_0(P_{2i+1} - P_{2i+2})$. Then, the signed distance between points P_{2i} and $\bar{P}_{2i}$ can be obtained as

$$h = \|P_{2i-1} - P_{2i-2}\|(\lambda_0 - 1)\sin\alpha$$
$$= \|P_{2i+1} - P_{2i+2}\|(\mu_0 - 1)\sin\beta. \tag{11}$$

With this definition, if we can obtain one of the two scalar factors μ_0 or λ_0, the other one can be obtained immediately.

When the control points P_{2i-1} and P_{2i} for the conic C_i have been moved to new positions, the lengths of two polygon legs have been changed as $\|\bar{P}_{2i-1} - P_{2i-2}\| = \lambda_0\|P_{2i-1} - P_{2i-2}\|$ and $\|\bar{P}_{2i} - \bar{P}_{2i-1}\| = \delta_0\|P_{2i} - P_{2i-1}\|$, where

$$\delta_0 = 1 - \frac{\|P_{2i-1} - P_{2i-2}\|}{\|P_{2i} - P_{2i-1}\|}(\lambda_0 - 1)\cos\alpha.$$

The end curvatures of the conic will become $(\delta_0/\lambda_0^2)k_i(0)$ and $(\lambda_0/\delta_0^2)k_i(1)$. To be sure that the new conic is still G^2 continuous with its adjacent conic C_{i-1}, the end curvature $k_i(0)$ of the conic C_i should be kept unchanged. Then, we can set the new weight as

$$\bar{w}_{2i-1} = \frac{w_{2i-1}}{\lambda_0}\sqrt{\delta_0},$$

and the two new end curvatures of the conic are $\bar{k}_i(0) = k_i(0)$ and $\bar{k}_i(1) = (\lambda_0^3/\delta_0^3)k_i(1)$. In a similar way we can set the new weight for the conic C_{i+1} as

$$\bar{w}_{2i+1} = \frac{w_{2i+1}}{\mu_0}\sqrt{\delta_1},$$

where

$$\delta_1 = 1 - \frac{\|P_{2i+1} - P_{2i+2}\|}{\|P_{2i} - P_{2i+1}\|}(\mu_0 - 1)\cos\beta,$$

and the two end curvatures are $\bar{k}_{i+1}(0) = (\mu_0^3/\delta_1^3)k_{i+1}(0)$ and $\bar{k}_{i+1}(1) = k_{i+1}(1)$. From Eq. (11), if $\lambda_0 > 1$, then $\mu_0 > 1$ and it can be easily verified that both $\bar{k}_i(1) > k_i(1)$ and $\bar{k}_{i+1}(0) > k_{i+1}(0)$ hold. On the other hand, if the point P_{2i} is pulled in the opposite direction, the left and the right curvatures at the point will both be decreased.

If we want the conic C_i to be with monotone curvature plot, the new control point $\bar{P}_{2i-1}$ should lie inside one but outside the other one of the two circles centered along the chord $P_{2i-2}\bar{P}_{2i}$ as defined in Theorem 1. Let the angle $\angle P_{2i-1}P_{2i-2}\bar{P}_{2i} = \alpha_0$, and because the disturbance h is always very small comparing with the lengths of the control polygon, α_0 can be chosen approximately as the original angle, i.e. $\alpha_0 \approx \angle P_{2i-1}P_{2i-2}P_{2i}$. To compute the permitted range of the scaling factor λ_0, we can just compute the scalars by which the control point $\bar{P}_{2i-1}$ lies on the circle O_0 or on the circle O_1, we have

$$2\frac{\|\bar{P}_{2i} - P_{2i-2}\|}{4\bar{w}_{2i-1}^2}\cos\alpha_0 = \lambda_0\|P_{2i-1} - P_{2i-2}\|, \tag{12}$$

$$2\frac{\|\bar{P}_{2i} - P_{2i-2}\|}{4\bar{w}_{2i-1}^2}\cos(\alpha - \alpha_0) = \|\bar{P}_{2i} - \bar{P}_{2i-1}\|. \tag{13}$$

From the triangle $\Delta P_{2i-2}\bar{P}_{2i}\bar{P}_{2i-1}$, we have

$$\frac{\|\bar{P}_{2i} - P_{2i-2}\|}{\sin\alpha} = \frac{\lambda_0\|P_{2i-1} - P_{2i-2}\|}{\sin(\alpha - \alpha_0)},$$

and by substituting the expression of $\bar{w}_{2i-1}$, Eq. (12) can be changed into

$$a_0\lambda_0^2 + b_0\lambda_0 + c_0 = 0, \tag{14}$$

where $a_0 = (1/2)\cos\alpha_0\sin\alpha$,

$$b_0 = \frac{\|P_{2i-1} - P_{2i-2}\|}{\|P_{2i} - P_{2i-1}\|}w_{2i-1}^2\sin(\alpha - \alpha_0)\cos\alpha,$$

and

$$c_0 = -\left(1 + \frac{\|P_{2i-1} - P_{2i-2}\|}{\|P_{2i} - P_{2i-1}\|}\cos\alpha\right)w_{2i-1}^2\sin(\alpha - \alpha_0).$$

On the other hand, from the triangle $\Delta P_{2i-2}\bar{P}_{2i}\bar{P}_{2i-1}$ we also have an identity

$$\frac{\|\bar{P}_{2i} - P_{2i-2}\|}{\sin\alpha} = \frac{\|\bar{P}_{2i} - \bar{P}_{2i-1}\|}{\sin\alpha_0},$$

then Eq. (13) can be changed into

$$a_1\lambda_0^2 + b_1\lambda_0 + c_1 = 0, \tag{15}$$

where $a_1 = (1/2)\cos(\alpha - \alpha_0)\sin\alpha$,

$$b_1 = \frac{\|P_{2i-1} - P_{2i-2}\|}{\|P_{2i} - P_{2i-1}\|}w_{2i-1}^2\sin\alpha_0\cos\alpha,$$

and

$$c_1 = -\left(1 + \frac{\|P_{2i-1} - P_{2i-2}\|}{\|P_{2i} - P_{2i-1}\|}\cos\alpha\right)w_{2i-1}^2\sin\alpha_0.$$

For most practical cases, $0 < \alpha, \alpha_0 < \pi/2$, then we have $a_0 > 0$, $b_0 > 0$ and $c_0 < 0$. In this case, there is one and only one positive root of Eq. (14) and we choose the solution as

$$\lambda_0^0 = \frac{-b_0 + \sqrt{b_0^2 - 4a_0c_0}}{2a_0}.$$

Similar result also hold for Eq. (15) and the positive solution can be obtained as

$$\lambda_0^1 = \frac{-b_1 + \sqrt{b_1^2 - 4a_1c_1}}{2a_1}.$$

If $Kt[i] = -1$, the control point P_{2i-1} lies inside the circle O_0 and in this case we have

$$2\frac{\|P_{2i} - P_{2i-2}\|}{4w_{2i-1}^2}\cos\alpha_0 > \|P_{2i-1} - P_{2i-2}\|.$$

Substituting

$$\bar{w}_{2i-1} = \frac{w_{2i-1}}{\lambda_0}\sqrt{\delta_0},$$

into Eq. (12), we can then conclude that $\lambda_0^0 < 1$. Similarly, from Eq. (13) we have $\lambda_0^1 > 1$. By the same method, we have $\lambda_0^0 > 1$ and $\lambda_0^1 < 1$ when $Kt[i] = 1$, $\lambda_0^0 > 1$ and $\lambda_0^1 > 1$ when $Kt[i] = 2$, $\lambda_0^0 < 1$ and $\lambda_0^1 < 1$ when $Kt[i] = 0$. In the case that the curvature $k_i(1)$ of point P_{2i} is a local minimum, the curvature type of conic C_i must be $Kt[i] = -1$ or $Kt[i] = 2$. Then the scaling factor λ_0 should be selected bigger than 1 and the curvature $\bar{k}_i(1)$ will be increased. On the other hand, if $Kt[i] = 1$ or $Kt[i] = 0$, λ_0 can be chosen less than 1 and the curvature $\bar{k}_i(1)$ will be decreased. If there is no positive real solution to Eqs. (14) or (15), we can just choose $\lambda_0^0 = \lambda_0^1 = 1$.

The criteria for the choice of λ_0 is that the local curvature extrema should be reduced or smoothed while the curvature difference of the conic at two ends should be made as small as possible. We choose $\tilde{\lambda}_0 = 1.1\lambda_0^1 - 0.1\lambda_0^0$ as the initial value of the scaling factor. With this choice, if $\lambda_0^1 > \lambda_0^0$ we have $\tilde{\lambda}_0 > \lambda_0^1 > \lambda_0^0$, and if $\lambda_0^1 < \lambda_0^0$, we have $\tilde{\lambda}_0 < \lambda_0^1 < \lambda_0^0$. This choice can be used for fast convergence purpose for most practical cases. But, even if λ_0^0 and λ_0^1 are both bigger than 1, it is still possible that $\tilde{\lambda}_0 < 1$ and a local minimum curvature may be decreased further if $\lambda_0 < 1$. So, we just choose $\tilde{\lambda}_0 = \lambda_0^1$ when $(\tilde{\lambda}_0 - 1)(\lambda_0^1 - 1) < 0$. Then the curvature $k_i(1)$ of the conic C_i will be increased if the original curvature at the point P_{2i} is a local minimum value or $k_i(1)$ will be decreased if the original curvature at the point is a local maximum value.

To compute another scaling factor μ_0 for the conic C_{i+1}, we can just replace P_{2i-2}, P_{2i-1}, $\bar{w}_{2i-1}$, α_0 and α within Eqs. (12) and (13) by P_{2i+2}, P_{2i+1}, $\bar{w}_{2i+1}$, β_0 and β, respectively, then we have another two quadratic equations with the unknown μ_0. By the same procedure we can obtain an initial value of μ_0 as $\mu_0 = \tilde{\mu}_0$. According to Eq. (11) the choices of λ_0 and μ_0 are dependent on each other. If we choose $\mu_0 = \tilde{\mu}_0$, we can also compute another value for λ_0 as $\bar{\lambda}_0$ from Eq. (11). Then the final choice for λ_0 can be set as $\lambda_0 = (\tilde{\lambda}_0 + \bar{\lambda}_0)/2$ and the value of μ_0 can be modified from Eq. (11) again.

To be sure that the new conic segments are with the same curvature sign as the original one, the position of the point $\bar{P}_{2i}$ should not only be close to the original position, but also the point $\bar{P}_{2i}$ should be kept in a fixed domain. The two points P_{2i-2}, P_{2i+2} and the intersection point of two tangent lines at these two points forms a triangle (see Fig. 4). Let the intersection points of the triangle with the normal line through the point P_{2i} be P_t and P_b, then the point $\bar{P}_{2i}$ should be moved within the line segment P_bP_t.

Though $\bar{P}_{2i}$ can be moved on the whole line segment P_bP_t theoretically, we set upper and lower bounds for scalars λ_0 or μ_0 for robust numerical computation. If P_t lies on the line $P_{2i-2}P_{2i-1}$, then the upper bound of the scalar λ_0 can be set as

$$\lambda_u = 1 + 0.2\frac{\|P_t - P_{2i-1}\|}{\|P_{2i-1} - P_{2i-2}\|},$$

and the upper bound of μ_0 can be derived from Eq. (11). If P_t lies on the line $P_{2i+2}P_{2i+1}$, we can set the upper bound μ_u for the scalar μ_0 first and compute the upper bound λ_u for λ_0 with μ_u. When λ_0 or μ_0 excels their upper bounds, we can reset λ_0 and μ_0 just as the corresponding bounds. The lower bounds for the scalars λ_0 and μ_0 can be set as a fixed number such as 0.5. If λ_0 or μ_0 is less than 0.5, it can then be set as 0.5 and the other one will be computed from Eq. (11) accordingly.

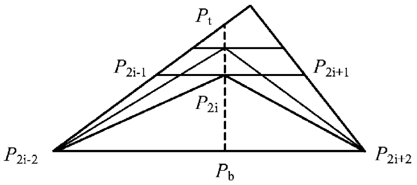

Fig. 4. The scaling bounds for the adjacent conic pair.

4.2. Conic arc fairing by weights scaling

The curvature extrema at the joint points can be reduced and smoothed efficiently by changing the positions of the points. However, the additional curvature extrema within some conic segments may not be removed by this method. To obtain a fair conic spline curve in the end, conic segments with unwanted curvature extrema can be faired by scaling the intermediate weights further.

Before defining the unwanted curvature extremum within a conic segment, we can first define the condition when a conic with one local curvature extremum can be accepted. If neither of the two end curvatures of the conic segment is a local extremum, nor the two adjacent conics have local curvature extrema, the curvature extremum within the conic is defined as an accepted extremum. With this definition, a conic segment with one local maximum curvature plot should be connected to a conic with monotone increasing curvature plot and followed by another conic with monotone decreasing curvature plot. Similarly, the previous conic and the next one to a conic with a local minimum curvature value should have monotone decreasing and monotone increasing curvature plots, respectively. Then any conic segment with a local curvature extremum but its adjacent conics does not satisfy the fairness criterion should be faired.

From Theorem 1, we can check the curvature type of the conic C_i by two circles defined with the control points P_{2i-2}, P_{2i-1}, P_{2i} and the weight w_{2i-1} (see the dashed circles in Fig. 5). Whether the curvature plot of the conic is monotone or not will be determined by the relationship between the point P_{2i-1} and the two circles. To remove the curvature extremum of the conic segment, we can then adjust the two circles by scaling the weight w_{2i-1} with the control polygon of the conic fixed.

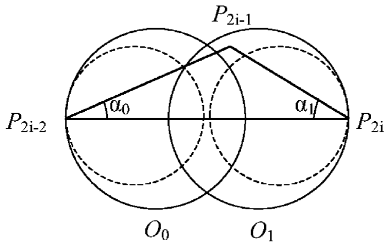

Fig. 5. Fair a conic segment by adjusting the intermediate weight.

Let the angle $\angle P_{2i-1}P_{2i-2}P_{2i} = \alpha_0$ and the angle $\angle P_{2i-2}P_{2i}P_{2i-1} = \alpha_1$, we should determine the permitted range for the weight w_{2i-1} so that the curvature plot of the conic segment is monotone. If $\alpha_0 > \pi/2$ or $\alpha_1 > \pi/2$, the curvature of the conic cannot be made monotone just by scaling the weight, so we will keep the weight unchanged. Otherwise, we will choose a new intermediate weight for the conic so that its curvature extrema will be reduced. Assume that if the weight is scaled as $w_{2i-1}l_0$ the circle O_0 will pass the point P_{2i-1}, then we have

$$2\frac{\|P_{2i} - P_{2i-2}\|}{4(w_{2i-1}l_0)^2}\cos\alpha_0 = \|P_{2i-1} - P_{2i-2}\|. \tag{16}$$

On the other hand, we can set the weight as $w_{2i-1}l_1$ so that the point P_{2i-1} lies on the circle O_1. We have

$$2\frac{\|P_{2i} - P_{2i-2}\|}{4(w_{2i-1}l_1)^2}\cos\alpha_1 = \|P_{2i} - P_{2i-1}\|. \tag{17}$$

From Eq. (16) we have

$$l_0 = \frac{1}{w_{2i-1}}\sqrt{\frac{\|P_{2i} - P_{2i-2}\|}{\|P_{2i-1} - P_{2i-2}\|}\frac{\cos\alpha_0}{2}},$$

and from Eq. (17) we have

$$l_1 = \frac{1}{w_{2i-1}}\sqrt{\frac{\|P_{2i} - P_{2i-2}\|}{\|P_{2i} - P_{2i-1}\|}\frac{\cos\alpha_1}{2}}.$$

Without loss of generality, we can assume that $\alpha_0 < \alpha_1$, then $l_0 > l_1$. If $w_{2i-1}l_1 > \sqrt{2}/2$, then any weight bigger than $w_{2i-1}l_1$ and less than $w_{2i-1}l_0$ can be used to construct a conic with monotone curvature plot. If the curvature type of the conic segment C_i has a local maximum value or $Kt[i] = 2$, we can choose $l = \min(l_0, l_1)$; and if the curvature type $Kt[i] = 0$, we choose $l = \max(l_0, l_1)$. With this choice, the curvatures at the two ends can be deformed as little as possible while the fairness of the conic segment has been improved. Then the new weight for the conic can be chosen as $\bar{w}_{2i-1} = lw_{2i-1}$, and the end curvatures of the conic segment can be obtained as $\bar{k}_i(0) = k_i(0)/l^2$ and $\bar{k}_i(1) = k_i(1)/l^2$. If the weight w_{2i-1} of the conic C_i satisfies the inequality $w_{2i-1} < \sqrt{2}/2$ and $Kt[i] = 1$ or $Kt[i] = -1$, there are one local maximum value and one local minimum value within the curvature plot of the conic. In this case, we can reset the new weight as $\bar{w}_{2i-1} = \sqrt{2}/2$ and the two curvature extrema within the conic segment can then be reduced or removed.

When the weight for the conic C_i has been changed, the curvatures at the two ends are not equal to the curvatures of two adjacent conics any more. To achieve the curvature continuity with the conic C_{i-1}, we can adjust the tangent at point P_{2i-2} with P_{2i-2} fixed as introduced in Section 3. The tangent at point P_{2i} can also be adjusted while fixing the point P_{2i} so that the conic C_i joins with G^2 continuity to the conic C_{i+1} too.

4.3. The algorithm

To obtain a fair conic spline curve fitting a set of ordered points, we can first fit the points by a tangent continuous Bézier spline or an arc spline curve. These initial G^1 conic spline curve will be represented as a quadratic rational B-spline curve. For every two adjacent conic segments without inflection, the common tangent and their intermediate weights will be adjusted. By setting new intermediate control points of every conic in Bézier representation and new weights for these points, the conic spline curve is curvature continuous at convex or concave parts.

The conic spline curves obtained above can be faired further by removing the unwanted curvature extrema. We will reduce the unwanted curvature extrema by adjusting the control points and the weights of conics in two main steps. Firstly, we can check every joint point and try to remove the unwanted curvature extrema at the joint points by moving the positions of the points. After that we can check all conic segments in turn and remove the unwanted curvature extrema within some conic segments by scaling the weights. When a joint point has been moved or a weight has been changed, the new end curvatures of the conic are not equal to the curvatures of its neighboring conics any more. We should then adjust the tangent directions at the joints to make a curvature continuous conic spline curve. But, adjusting a common tangent at a joint point may arouse unwanted curvature extrema within adjacent conic segments. On another hand, the end curvatures of a conic may become local extrema when the curvature type of the conic has been changed too. Then we can fair the conic spline curve by repeating the above two steps. The repeat procedure will not be stopped until all the local curvature extrema are accepted or a prescribed iteration number has been reached. We can accelerate the convergence speed by repeating the first step for joint point resetting with a few times before adjusting the weights for conics with unwanted local curvature extrema. Although convergence cannot be proven, satisfactory convergence was observed in all the examples we tried. It should be noted that different fairness criteria may influence the convergence speed.

For each time of fairing, we need three independent functions, TangentRotating (P_{2i}), PointResetting (P_{2i}) and WeightScaling (C_i). TangentRotating (P_{2i}) is a function to construct a curvature continuous conic pair by rotating the tangent at the joint and resetting two intermediate weights of the conics. To compute how to move the joint point P_{2i}

along the normal direction at the point we use the function PointResetting (P_{2i}). WeightScaling (C_i) can be used to reduce or remove the curvature extrema within a conic segment by adjusting the weight corresponding to the middle control point of the conic C_i. If we want to construct an interpolating curve, we can just omit the point moving step in the following algorithm. For a conic spline with inflection points, we can either divide the curve into convex and concave parts and fair every part independently, or fair the whole curve in a uniform procedure, but joint point moving or weight scaling are only applied for two adjacent conic segments or three adjacent segments without inflection. In this paper we choose the second method because it gives a more compact program.

The algorithm for fair conic spline fitting:

```
Step 1. Fit the data by a $G^1$ conic spline curve consisting of $m$ segments;
Step 2. Construct a $G^2$ continuous conic spline from the initial spline;
Step 3. Compute the curvature type for each conic segment;
Step 4.1. For ($i = 1$; $i < m$; $i++$) {
    If (the curvature at the point $P_{2i}$ is an unwanted local extremum) {
    PointResetting ($P_{2i}$);
    TangentRotating ($P_{2i}$);
    }
}
Step 4.2. Repeat step 4.1 for 3 times;
Step 5. For ($i = 2$; $i < m$; $i++$) {
    If (conic $C_i$ has unwanted local curvature extrema) {
    WeightScaling ($C_i$);
    TangentRotating ($P_{2i-2}$);
    TangentRotating ($P_{2i}$);
    }
}
Step 6. Repeat step4 and step 5 until all the unwanted curvature extrema have been removed or a certain iteration number has been reached.
```

5. Examples and comparison

We have tested the algorithm for many examples and we show a few examples here to illustrate the efficiency of the algorithm.

In example 1 we sample a set of random points from a locally convex plane curve

$$X(t) = \frac{t^2 - 1}{t^2 + 1}\begin{pmatrix} t \\ 1 \end{pmatrix}, \qquad t \in [-2, 2],$$

then these points form a locally convex polygon. At first we interpolate the points by a tangent continuous quadratic Bézier spline curve, where the tangent at every interior point is selected paralleling the line connecting its former and next points and the tangents at two ends are determined by the rule that the control polygon of the first and last Bézier curves form two isosceles triangles (see dashed line in Fig. 6(a)). After that, we have tested two methods to construct curvature

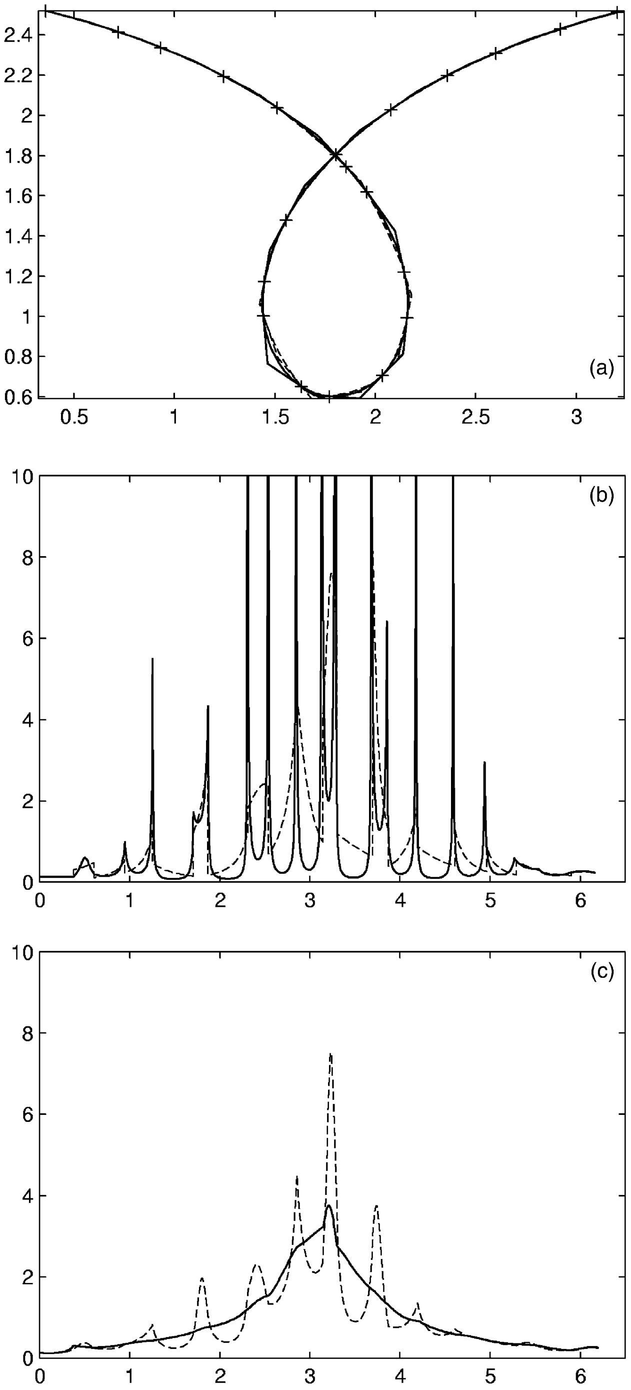

Fig. 6. (a) Interpolating a set of points by locally convex conic splines: the G^1 conic spline (dashed) and a fair, G^2 conic spline (solid); (b) the curvature plot of initial G^1 conic spline (dashed) and of G^2 conic spline with fixed control polygon (solid); (c) the curvature plots of the interpolating G^2 conic spline before (dashed) and after (solid) fairing.

continuous conic spline curves interpolating the sampled points. One method is to keep the control polygon unchanged and compute weights for every conic based on curvature continuity condition [6]. The curvature plot of the G^2 conic spline is shown in Fig. 6(b). The second method is our tangent rotating method presented in Section 3. By computing the new tangent direction and new intermediate weights for every two adjacent conics, we can obtain a curvature continuous conic spline too. One advantage of the second method is that it is a local method. Even more, the curvature extrema within some conic segments can be reduced efficiently by scaling the intermediate weights and rotating the tangents at the joint points, a fair interpolating curve is obtained in the end (see Fig. 6(c)).

In the second example, the original data were sampled from the contour of a bone section. The data points are noisy and irregular. We first approximate the data by an arc spline within tolerance 0.35×10^{-2} and then reduce the arc segments within another tolerance 0.2×10^{-2}. The tiny arcs are also merged and only 34 arcs are left. When the arc spline is obtained, we can construct a fair G^2 conic spline from the arc spline (see Fig. 7(a)). The arc spline and the conic spline in rational B-spline forms are plotted in Fig. 7(b). The curvature plot of an arc spline interpolating the original noisy points and the curvature plot of the final approximating arc spline are shown in Fig. 7(c). For curve parts without inflection, curvature continuous conic spline can be obtained from the arc spline. Even more, the unwanted curvature extrema of the conic spline can be reduced efficiently (see Fig. 7(d)).

In the third example, we fit the profile of a mouse section by an arc spline first and then construct a fair conic spline from the arc spline (Fig. 8(a)). With the same method as example 2, we obtain an arc spline by fitting the data within tolerance 0.5×10^{-2}. Because the arc lengths and arc angles of the initial arc spline are not uniform, the deviation of the final conic spline from the arc spline varies non-uniformly along the whole curve (see Fig. 8(b) and (c)). From the curvature plots in Fig. 8(d) and (e), we obtain a fair conic spline curve fitting the original data in the end.

The algorithm presented in this paper has been implemented on a SGI octane workstation with MIPS R10000. Because we fair a conic spline by a local algorithm and we reduce curvature extrema only by moving selected

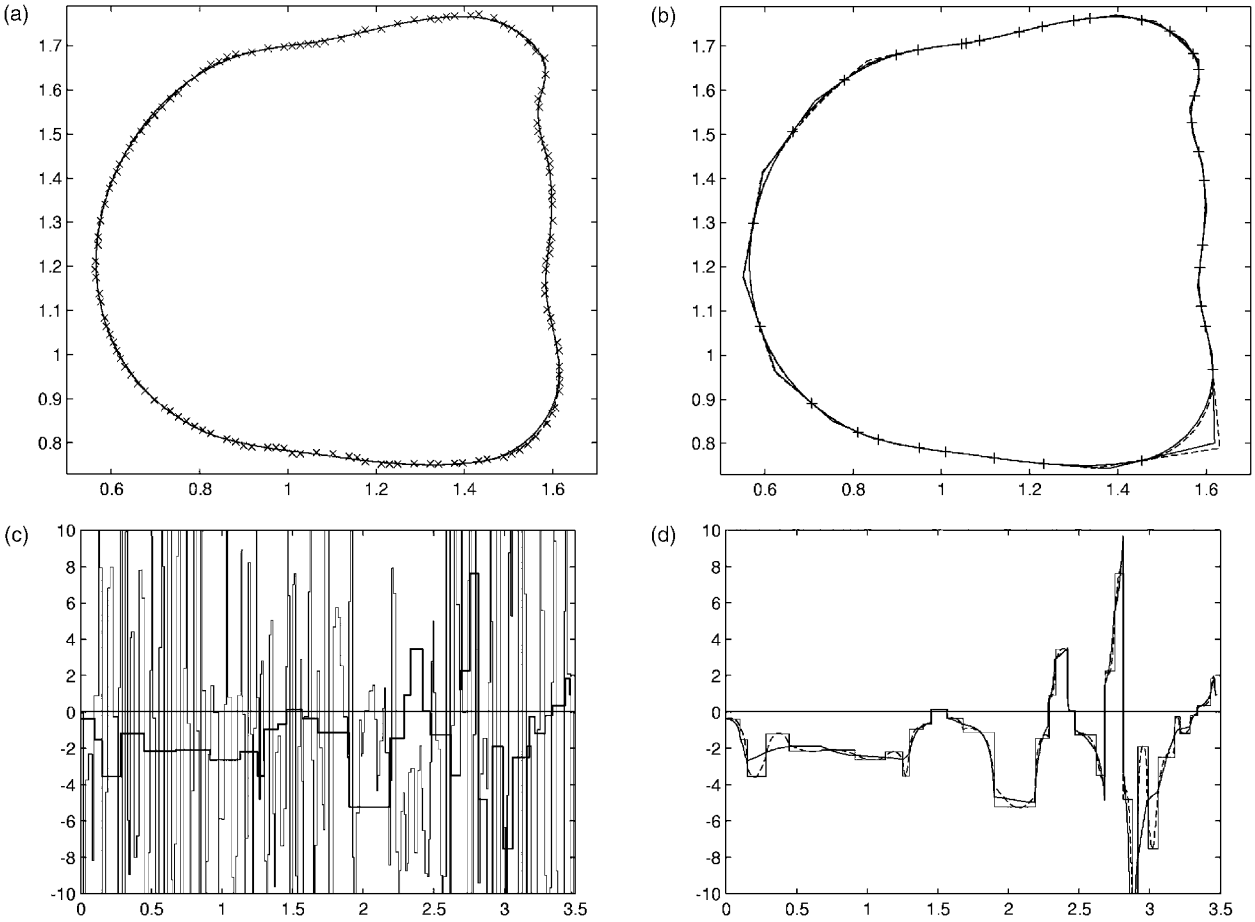

Fig. 7. (a) Approximating the profile of a bone section by an arc spline (dashed) and a fair conic spline (solid); (b) the arc spline (dashed) and the conic spline (solid) with control polygons; (c) curvature plot of an arc spline interpolating the original noisy data (thin) and the curvature plot of a fair fitting arc spline (thick); (d) curvature plots of the G^2 conic spline before (dashed) and after (solid) fairing.

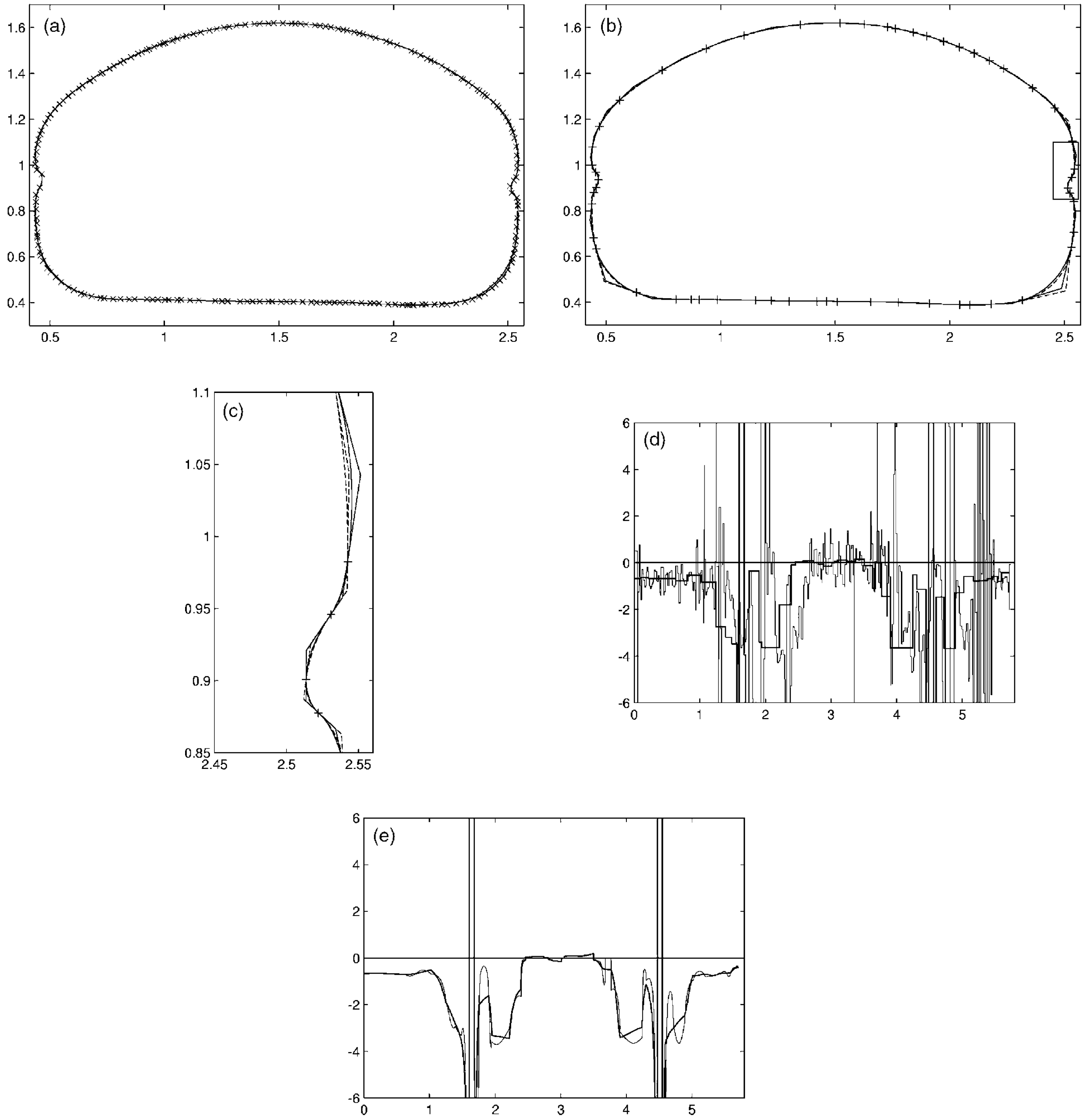

Fig. 8. (a) Approximating a mouse section by an arc spline (dashed) and a fair conic spline (solid); (b) the arc spline (dashed) and the conic spline (solid) with control polygons; (c) zoom in a curve part selected in Fig. 8(b); (d) curvature plot of an arc spline interpolating the original noisy data (thin) and the curvature plot of a fair fitting arc spline (thick); (e) curvature plots of the G^2 conic spline before (thin) and after (thick) fairing.

points and scaling selected weights, the curve can be faired in real time. The conic number, the iteration number, the total number for point move, the total number for weight scaling and the time for conic spline fairing are listed in Table 1.

6. Conclusions

In this paper we have presented a geometric method for constructing curvature continuous conic spline from

Table 1
The iteration number and fairing time, etc. for the three examples

	Points	Conics	Iteration	Point move	Weight scaling	Time (s)
Convex curve	21	20	5	0	47	0.43
Bone section	138	34	28	121	111	0.46
Mouse section	204	54	20	185	195	0.65

an initial tangent continuous conic spline curve and fair the conic spline curve by removing additional curvature extrema within the curve. The original G^1 curve is obtained by interpolating a point set with quadratic Bézier spline or fitting a set of noisy data with an arc spline. By representing the conic spline in piecewise rational quadratic Bézier curves, we can then change the control points and weights of adjacent conic pairs to make curvature continuous conic spline curve. The new conic spline interpolates the same set of points if the original conic spline interpolates the data at the conic ends. To fair a conic spline curve, the curvature extrema can be reduced efficiently by scaling selected conic segments. The final conic spline curves are G^2 continuous at convex or concave parts and G^1 continuous at inflection points. If a curvature continuous curve is desired, the parts containing inflection points can be replaced by cubic curves or other curves with inflections [13,17].

The method presented in this paper can be used in the fields where the shape quality is more desired than the accuracy of the fitting. In this paper we remove the additional curvature extrema for every two or three adjacent conic segments. In fact, this method can be extended to fair a conic spline with some other fairness criteria. For example, there is at most one curvature extremum within a curve part consisting of four, five or even more consecutive segments or a curve part with prescribed length. Then the conic spline can be faired by a similar fairing algorithm.

Acknowledgments

The author owes thanks to referees for their valuable comments and suggestions which have helped to improve the presentation of the paper greatly. This work is supported by 973 project (2002CB312101) and Natural science foundation of China (60073023).

References

[1] Ahn YJ. Conic approximation of planar curves. Comput-Aid Des 2001;33(12):867–72.

[2] Ahn YJ, Kim HO. Curvatures of the quadratic rational Bézier curves. Comput Math Appln 1998;36(9):71–83.

[3] Bae SH, Choi BK. NURBS surface fitting using orthogonal coordinate transform for rapid product development. Comput-Aid Des 2002; 34(10):683–90.

[4] Blanc C, Schlick C. Accurate parametrization of conics by NURBS. IEEE Comput Graph Appln 1996;16(6):64–71.

[5] Fang L. A rational quartic Bézier representation for conics. Comput Aid Geometr Des 2002;19(5):297–312.

[6] Farin G. Curvature continuity and offsets for piecewise conics. ACM Trans Graph 1989;8(2):89–99.

[7] Farin G. From conics to NURBS: a tutorial and survey. IEEE Comput Graph Appln 1992;12(5):78–86.

[8] Farin G, Sapidis N. Curvature and the fairness of curves and surfaces. IEEE Comput Graph Appln 1989;9(2):52–7.

[9] Farouki RT. Conic approximation of conic offsets. J Symbolic Comput 1997;23(2–3):301–13.

[10] Farouki RT, Manni C, Sestini A. Real-time CNC interpolators for Bézier conics. Comput Aid Geometr Des 2001;18(7):639–65.

[11] Frey WH, Field DA. Designing Bézier conic segments with monotone curvature. Comput Aid Geometr Des 2000;17(6):457–83.

[12] Hoschek J. Circular splines. Comput-Aid Des 1992;24(11):611–8.

[13] Juh́csz I. Cubic parametric curves of given tangent and curvature. Comput-Aid Des 1998;30(1):1–9.

[14] Lee ET. The rational Bézier representation for conics. Geometric Modeling: Algorithms and new trends, Philadelphia: Academic Press; 1987. p. 3–19.

[15] Lim CG. A universal parametrization in B-spline curve and surface interpolation. Comput Aid Geometr Des 1999;16(5):407–22.

[16] Ma W, Kruth JP. NURBS curve and surface fitting for reverse engineering. Int J Adv Manufact Technol 1998;14:918–27.

[17] Meek DS, Walton DJ. Planar G^2 Hermite interpolation with some fair, C-shaped curves. J Comput Appl Math 2002;139(1):141–61.

[18] Park H. Choosing nodes and knots in closed B-spline curve interpolation to a point data. Comput-Aid Des 2001;33(13):967–74.

[19] Pavlidis T. Curve fitting with conic splines. ACM Trans Graph 1983; 2(1):1–31.

[20] Piegl LA. Techniques for smoothing scattered data with conic sections. Comput Ind 1987;9(3):223–37.

[21] Piegl LA, Tiller W. Curve and surface constructions using rational B-splines. Comput-Aid Des 1987;19(9):485–98.

[22] Piegl LA, Tiller W. Data approximation using biarcs. Engng Comput 2002;18(1):59–65.

[23] Pottmann H. Locally controllable conic splines with curvature continuity. ACM Trans Graph 1991;10(4):366–77.

[24] Pratt V. Techniques for conic splines. ACM Comput Graph 1985; 19(3):151–9.

[25] Schaback R. Planar curve interpolation by piecewise conics of arbitrary type. Construct Approx 1993;9:373–89.

[26] Seymour C, Unsworth K. Interactive shape preserving interpolation by curvature continuous rational cubic splines. J Comput Appl Math 1999;102(1):87–117.

[27] Tiller W. Rational B-splines for curve and surface representation. IEEE Comput Graph Appln 1983;3(6):61–9.

[28] Walton DJ, Meek DS. Planar G^2 curve design with spiral segments. Comput-Aid Des 1998;30(7):529–38.

[29] Yang XN. Efficient circular arc interpolation based on active tolerance control. Comput-Aid Des 2002;34(13):1037–46.

[30] Yang XN, Wang GZ. Planar point set fairing and fitting by arc splines. Comput-Aid Des 2001;33(1):35–43.

Xunnian Yang is now an associate professor in the department of mathematics and institute of Computer Graphics and Image Processing at Zhejiang Universisity, China. He obtained a BS in applied mathematics from Anhui University and a PhD in CAGD and Computer graphics from Zhejiang University in 1993 and 1998, respectively. His research interests include spans of computer aided geometric design and computer graphics.