Constructing a constrained C^1 continuous planar interpolated spline curve

Vinod Kumar[1], *Maheshanand*[2], Tushar Patnaik[3]
[1]*SOIT C-DAC Noida*, [1]*IIT Roorkee*, [3]*SOIT C-DAC Noida*

Abstract

The problem of constructing a smooth C^1 continuous planar interpolated spline curve has attracted the attention of many people working in the area of CAD/CAM and its applications such as robotics. In the present paper we propose a method for constructing a C^1 continuous planar interpolated spline using rational quadratic Bezier curve that falls within a closed boundary of straight line segments, which is most frequently used in computer graphics and geometric modelling. The rational quadratic curves are used by CAGD scientists since they do not require complex computations as other higher degree curves do. However, in practice it is desirable to approximate conic sections which cannot be represented by simple Bezier curve. Besides this, we have also presented the some useful properties of the rational quadratic Bezier curve.

Keywords: Rational Quadratic Bezier curve, constrained curve, C^1 continuity, smoothness, interpolation.

1. Introduction

Computer aided geometric design (CAGD) is the science of design. The curve shape will be used to design, such as car, furniture, robot path or other industrial design. The shape may be more accurate if it is design by using computer.

There are several problems whose solution requires this type of method. A user may wish to design a curve that fits on the given data points and falls within the boundary. A user may wish to design a smooth path that follow the given data points, like designing a robot path.

The rational quartic representation of a conic section has been studied in some papers [1, 2, 3, 5]. In [14], Goodman, Ong, Unsworth have presented a construction of a G^2 continuous, shape-preserving curve made of rational cubics that interpolates given points and that lies on one side of a line, or several lines. In[4], Meek, Ong, Walton have given a method for a G^2 continuous curve made of rational cubics that interpolates to given points inside an arbitrary polygon. In [12], interpolation to data points that lie on one side of one or more lines has been considered for generating a G^2 rational cubics spline which also lies on the same side of each of these lines is given by Goodman et al., (1991).

In all the results mentioned above the rational cubics of degree three have been used which are more complex than the rational quadratic. In [6], the non-parametric C^1 rational cubic scheme is extended to include quadratic curves, by relaxing the linear constraints, and the weights of the rational cubic are adjusted so as to satisfy the conditions that a rational cubic curve does not cross a given line. In [1], a method for constructing G^1 quadratic Bezier curves that satisfy given endpoint (positions and arbitrary unit tangent vectors) conditions is described.

In this paper we present a method of constructing a constrained C^1 planar interpolated spline using rational quadratic Bezier curve that falls within a closed boundary of straight line segments. To solve this problem we have used rational quadratic Bezier curve because the space and computation costs of quadratic Bezier curves are both smaller than any other free form curves of degree three or higher. The method used here also gives more localized control on the curve segment.

The paper is organized as follows. In Section 2, we present the rational quadratic B´ezier curves on the *2 “ D* plane and some important properties of the families of curves derived from a rational quadratic. In Section 3 provides an approach to construct the composite C^1 continuous rational quadratic Bezier curves with the endpoint constrains. Section 4 is devoted to determination of the conditions for which the curve that passes through a given point and the given line segment will be tangent to a curve. This will be useful in the construction of the constrained interpolating curve in Section 5. Concluding remarks are presented in the last section.

2. Rational Quadratic Bezier curve

The family of rational quadratic Bezier curves *B (t)* with non-zero area of control triangle $B_0B_1B_2$ is represented by

$$B(t)=\frac{w_0(1-t)^2B_0+2w_1(1-t)tB_1+w_2t^2B_2}{w_0(1-t)^2+2w_1(1-t)t+w_2t^2};0\le t\le 1, B_i\in R^2 \qquad \text{———— (1)}$$

Where B_i $(i = 0, 1, 2)$ are the control points of the curve and w_i are the weights.

Here we list some useful properties of rational quadratic.

2.1 Uniqueness of Weights: For a rational quadratic Bezier curve the value of $w_0 w_2 / 4w_1^2$ remains unchanged, so without loss of generality we may assume that $w_0 = 1$, $w_2 = 1$. We can then rewrite (1) as:

$$B(t)\frac{(1-t)^2 B_0 + 2w_1(1-t)tB_1 + t^2 B_2}{(1-t)^2 + 2w_1(1-t)t + t^2};0 \le t \le 1, B_i \in R^2. \qquad (2)$$

It is called the standard form of rational quadratic Bezier curve. It is well known that the members of this family of curves are segment of conic [2].

2.2 Convex Hull: For $w_1 > 0$ every family of curve segment lies in the convex hull of the control polygon.

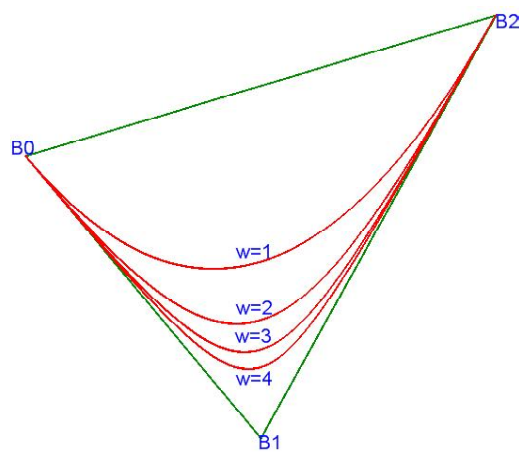

Figure:1 Families of curves derived from a rational quadratic

2.3 Endpoints interpolation: It is easy to see that the curve goes through the both endpoints B_0 and B_2. We have

$$B(0) = B_0 \qquad B(1) = B_2$$

$B'(0)$ has the same direction with the vector $B_1 - B_0$ and $B'(1)$ has the same direction with the vector $B_2 - B_1$.

2.4 Type Parameter: Different conics are uniquely determined by the weight w_1. The curve is a segment of parabola when the weight $w_1 = 1$; $w_1 < 1$ gives a segment of an ellipse, and $w_1 > 1$ gives a segment of a hyperbola. If we change the sign of the weight w_1, then it will represent the complementary segment of the conics.

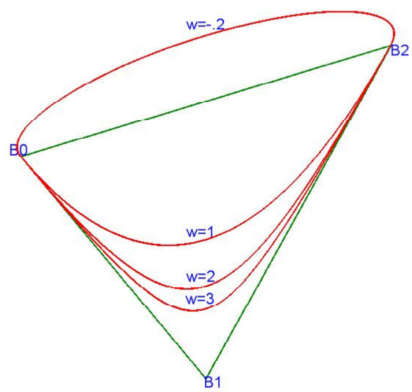

Figure 2: Type parameter

3. Composition of two segments of rational quadratic Bezier curves

If two segments of rational quadratic Bezier curve are given then we can construct a composite curve by imposing some continuity conditions. Here we construct the C^1 continuous curve i.e. the tangent direction and the magnitude for both the curve at joining will be same. The tangent vector at the starting point of the first rational quadratic Bezier curve $B_1(t)$ has the same direction with B_1-B_0, and the tangent vector at the ending point of the second rational quadratic Bezier curve $B_2(t)$ has the same direction with B_4-B_2. We introduce the new point P as $P = (B_1+B_2)/2$, which will be a joining point of two rational quadratic bezier curve. The consecutive segments in a composite Bézier curve are C^1 continuous if the penultimate control vertex of the first curve, the shared endpoint and the second vertex of the next curve are collinear and equally spaced. In Fig.3 the vertices B_1, P and B_2 are collinear and $|| P - B_1|| = || B_2 - P||$ so the composite curve will be C^1 continuous. The composite curve made by $B_1(t)$ and $B_2(t)$ also satisfied the arbitrary end point conditions.

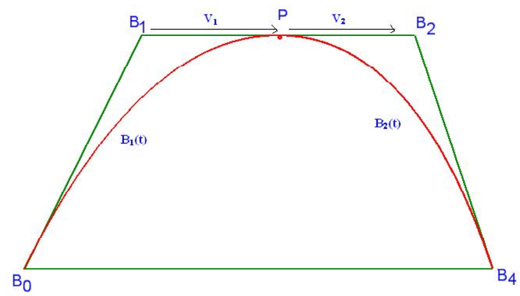

Figure 3. C^1 rational Quadratic Bezier Curves (two segments)

4. Finding curves passing through given points and tangent to given line segments.

4.1 The curve that passes through a given point

The standard form of rational quadratic Bezier curve is given by the Eq.(2).

The points on a rational quadratic Bezier curve are a weighted average of the control points B_0, B_1 and B_2. With the restrictions on t and w_1 in Eq.(2), all of the weights are positive, so all points $B(t, w_1)$ will be inside the control triangle B_0, B_1, B_2.

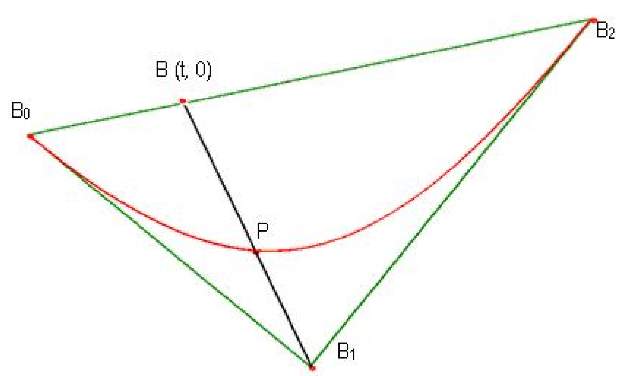

Figure 13: The curve passing through a given point

Given the control points B_0, B_1, B_2 and we know that the curve pass through a point P where the point is inside the $\Delta(B_0, B_1, B_2)$. We need to find out the $t \in (0,1)$ and $w_1 \in (0, \infty)$ such that $B(t, w_1) = P$.

Let the intersection point of the line through B_0 and B_2 with the line through B_1 and P is $S(p_x, p_y)$. The point of intersection for a given t is also on $B(t,0)$ such that $B(t,0) = S(p_x, p_y)$ and this allows the solution of t independently of w_1. The line through B_0 and B_2 is from Eq. (2)

$$B(t,0) = \frac{(1-t)^2 B_0 + t^2 B_2}{(1-t)^2 + t^2} \quad (3)$$

Now from $B(t,0) = S(p_x, p_y)$ we have

$$S(p_x, p_y) = \frac{(1-t)^2 B_0 + t^2 B_2}{(1-t)^2 + t^2} \quad (4)$$

Now from Eq.(4) we have the system equations

$$(1+t^2-2t)X_0 + t^2 X_2 - p_x((1+t^2-2t)+t^2) = 0 \quad (5)$$

And

$$(1+t^2-2t)Y_0 + t^2 Y_2 - p_y((1+t^2-2t)+t^2) = 0 \quad (6)$$

The value of t can be computed either from Eq.(5) or Eq.(6) by solving the quadratic equation in t. We choose the value of $t \in (0,1)$.

Once the t value is calculated from the Eq.(5) or Eq.(6), the corresponding positive value for w_1 is calculated from Eq.(2) as

$$w_1 = \frac{(1-t)^2(B_0-P)(P-B_1) + t^2(B_2-P)(P-B_1)}{2(1-t)t(P-B_1)(P-B_1)} \quad (7)$$

Therefore if the P is inside the Bezier control triangle B_0, B_1, B_2 then there is a unique curve that passes through P. If P is not inside the control triangle then there is no curve that passes through P.

4.2 The curve which is tangent to a given line segment

Given the control points B_0, B_1, B_2 and the line segment L which intersects the polygon at B'_0 and B'_1. We have to find the w_1 value of the curve that is tangent to a given line segment L.

We can define the weight points of a conic section by the using A de Casteljau Algorithm as

$$q_i = \frac{w_i B_i + w_{i+1} B_{i+1}}{w_i + w_{i+1}} \qquad (8)$$

Now the weight point's q_0 and q_1 of a conic section are given by

$$q_0 = \frac{w_0 B_0 + w_1 B_1}{w_0 + w_1}, \text{ and } \quad q_1 = \frac{w_1 B_1 + w_2 B_2}{w_1 + w_2}$$

Where the weight points and the control points are collinear. In this standard form case the weight points q_0 and q_1 are

$$q_0 = \frac{B_0 + w_1 B_1}{1 + w_1}, \text{ and } \quad q_1 = \frac{B_2 + w_1 B_1}{1 + w_1} \qquad \because \{w_0 = w_2 = 1\}$$

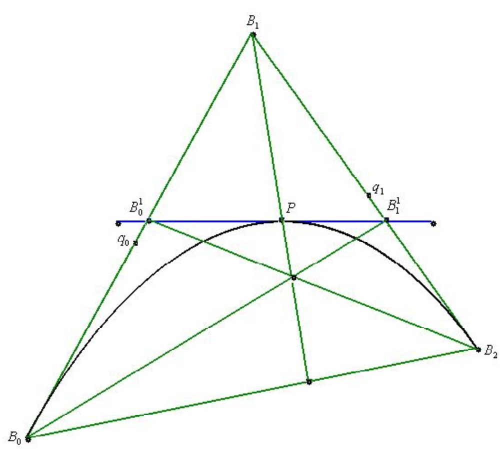

Figure 4: conic constructions: B^0, B^1, B^2, and the tangent are given

The ratio of three collinear points a, b, c is defined by

$$ratio\ (a, b, c) = \frac{volume(a,b)}{volume(b,c)} \quad \text{——— (9)}$$

Where *volume* denotes the one-dimensional *volume,* which is the signed distance between two points.

Now we can compute the ratio of B_0, B_0', B_1 and B_1, B_1', B_2 as follows

$$\left.\begin{array}{l} a = ratio(B_0, B_0', B_1) \\ and \\ b = ratio(B_1, B_1', B_2) \end{array}\right\} \quad \text{——— (10)}$$

Now from the definition of the weight point q_i in Eq. (8), it follows that

$$\left.\begin{array}{l} ratio(B_0, q_0, B_1) = w_1 \\ and \\ ratio(B_1, q_1, B_2) = 1/w_1 \end{array}\right\} \quad \text{——— (11)}$$

The cross ratio *cr* of four collinear points is defined as a ratio of ratios

$$cr(a,b,c,d) = \frac{ratio(a,b,d)}{ratio(a,c,d)} \quad \text{——— (12)}$$

So the *cr* of four points $(B_0, B_0', q_0, B_1) and (B_1, B_1', q_1, B_2)$ is defined as

$$\left.\begin{array}{l} cr(B_0, B_0', q_0, B_1) = \dfrac{ratio(B_0, B_0', B_1)}{ratio(B_0, q_0, B_1)} \\ and \\ cr(B_1, B_1', q_1, B_2) = \dfrac{ratio(B_1, B_1', B_2)}{ratio(B_1, q_1, B_2)} \end{array}\right\} \quad \text{——— (13)}$$

Now by the four tangent theorems we have

$$cr(B_1, B_1', q_1, B_2) = cr(B_0, B_0', q_0, B_1)$$

From equation (13) we have

$$\frac{ratio(B_1, B_1', B_2)}{ratio(B_1, q_1, B_2)} = \frac{ratio(B_0, B_0', B_1)}{ratio(B_0, q_0, B_1)}$$

Now from equation (10) and (11) we have

$$\frac{b}{1/w_1} = \frac{a}{w_1}$$

$$bw_1 = \frac{a}{w_1}$$

$$w^2{}_1 = \frac{a}{b}$$

$$w_1 = \sqrt{\frac{a}{b}} \qquad (14)$$

5. Construction of the constrained interpolating curve

5.1 Initial C^1 continuous interpolating spline curve

Suppose that we are given some planar data points lying within the constrained polyline boundary, and the polyline joining these data points consecutively does not intersect the boundary. We would like to construct a C^1 piecewise rational quadratic interpolating curve that lies within the closed boundary of straight line segments. We assume that no three consecutive data point are collinear.

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

$$\kappa(t) = \frac{\|B'(t) \times B''(t)\|}{\|B'(t)\|^3}$$

then the curvatures at the two ends of the conic segment can be computed as

$$\kappa(0) = \frac{1}{2} \frac{\|(B_1 - B_0) \times (B_2 - B_1)\|}{w^2 \|B_1 - B_0)\|}$$

and $$\kappa(1)=\frac{1}{2}\frac{\|(B_1-B_0)\times(B_2-B_1)\|}{w^2\|B_2-B_1)\|}$$

respectively, where *k(0)* and *k(1)* are the curvature of curve *B(t)* at 0 and 1. From the curvature formula, we conclude that if the weight w_1 will be changed the curvatures will also be scaled simultaneously. Suppose that the three control points B_0, B_1 and B_2 are not collinear and $w_1>0$; then the conic segment will not degenerate to a line. Let the unit vector U in the direction of $B_2 - B_0$ is

$$U=\frac{B_2-B_0}{\|B_2-B_0\|}$$

Now we can define two circles C_1 and C_2 both with radius r and centered at $C_1 = B_0 + rU$ and $C_2 = B_2 - rU$; respectively where the radius r is given by

$$r=\frac{\|B_2-B_0\|}{4w^2}$$

If the control point B_1 lies inside both of the circles then the curvature plot of the conic is with a local minimum value. If the control point B_1 lies outside both of the circles then the curvature plot of the conic is with a local maximum value [2]. If the control point B_1 lies inside one of the two circles but outside the other one, then the curvature plot of the conic segment has one local maximum value and one local minimum value when $w_1^2 <1/2$ and monotone when $w_1^2 >1/2$[2].

We conclude that for the fairness of the curve the control point B_1 lies outside both of the two circles C_1 and C_2. In this case the curvature plot has a local maximum. If B_1 lies inside both of the two circles C_1 and C_2; there is a local minimum value within the curvature plot. If B_1 lies outside C_1 but inside C_2 then the curvature plot of the conic is monotone increasing. When $w^2 >1/2$; and If B_1 lies inside C_1 but outside C_2 then the curvature plot of the conic is monotone decreasing.

Now we will have to find the relationship between the middle control point B_1 and the two circles C_1 and C_2, when B_1 lies on the circle C_1, we have

$$2\frac{\|B_2-B_0\|}{4w^2}\cos\theta_1=\|B_1-B_0\|$$

If B_1 lies on the circle C_2 we have

$$2\frac{\|B_2-B_0\|}{4w^2}\cos\theta_2=\|B_2-B_1\|$$

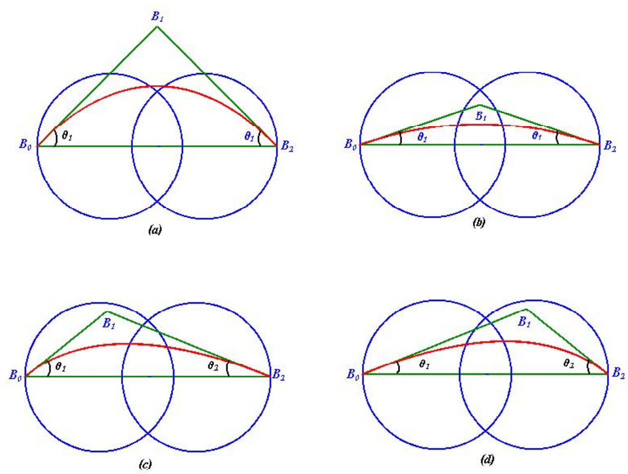

Figure 4. Curvature type for a conic segment in Be´zier form

B_1 lies outside or inside both of the two circles C_1 and C_2 we have

$$2\frac{\|B_2 - B_0\|}{4w^2}\cos\theta_1 = \|B_1 - B_0\| = \|B_2 - B_1\|$$

For constructing the fair curve we have to choose the point B_1 lies outside both of the circles C_1 and C_2 where the curvature plot has a local maximum. Here each conic segment is represented as a quadratic rational Be´zier curve, the conic spline curve consisting of n segments of smooth connected conic pieces can then be transformed into a rational B-spline curve. The control polygon of the rational B-spline curve is $P_0P_1P_2\ldots P_{2n-1}P_{2n}$ where $P_{2i-2}P_{2i-1}P_{2i}$ are just the control polygon of the conic C_i $(i = 1, 2, 3 \ldots\ldots n)$. 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 $P_{2i-1}(i = 1,2,3,\ldots.n)$. The rest weights $w_0 = w_2 = \cdots\cdots = w_{2n} = 1$.

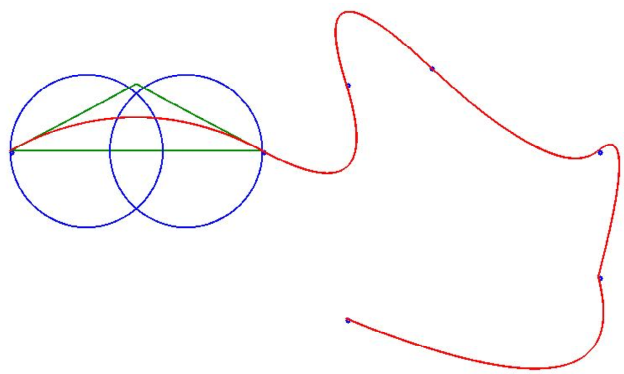

Figure5: Example 5- Constrained interpolating curve with 7 data points

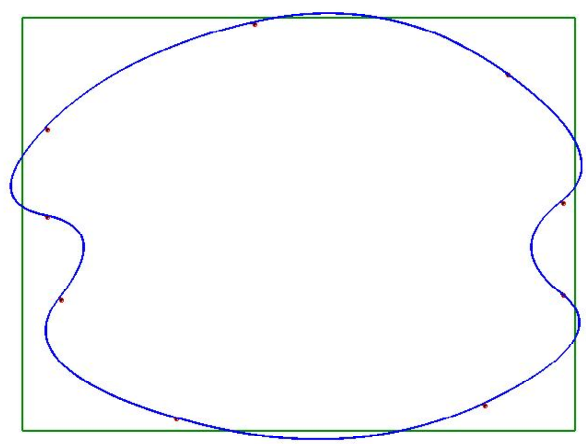

Figure 6: Example 2-Constrained interpolating curve with 9 data points that intersect some boundary segments.

5.2 Algorithm

Based upon the above discussion, we propose an algorithm that could be used to generate a straight line boundary avoiding curve.

Algorithm. Given polyline path (guiding path) segments {I_i: *0d" id"n-1*} where there are no two consecutive coincide points, and a constraint polyline with boundary segments {$\boldsymbol{L}_i$: 0d”*i* d”*m-1*} that do not intersect with the given polyline joining the data points.

(1) For *i* = 0, 1, . . .,*n* “ 1, construct the rational quadratic bezier curve segment *i* of the form B (t, w_1) as described in section 5.

(2) For *i* = 0, 1, . . .,*n* “ 1,

(a) For each of the Be´zier curves, check all boundary segments that are partly, entirely inside or entirely outside the Be´zier control triangle.

(i) If boundary does not enter the control triangle then the initial curve does not intersect the boundary, so the curve with w_1=1 will be the final curve.

(ii) If boundary enters the control triangle then the initial curve may intersect the boundary, so determine the w-value of the curve that avoids all those boundary segments L_i. by performing the necessary operation:

1. Find the rational quadratic $B(t, w_1)$ which passes through the joining point of the concerned two boundary segments by using the results of section 4.1

2. Find the rational quadratic $B(t,w_1)$ which touches the concerned boundary segments using the results of section 4.2

(b) Determine the weight factor w_h which is the largest of all the w_1-values of the curves, then w_1 = max (1, w_h) is the w-value of a curve (2) that does not intersect any of the boundary segments.

(c) The default value of weight factor w_1= 1 is used if the boundary segments allow that value.

6.0 Graphical examples

We shall illustrate our above discussion with two examples. In both examples the data points are lying inside the boundary and marked by "●".

The First example shown in Fig 7 with 9 data in which five initial curve segments crossed the boundary. The resultant curve with w_1 value is determined by using the algorithm described in sections 5.2 such that the final curve avoids all the straight line boundary segments.

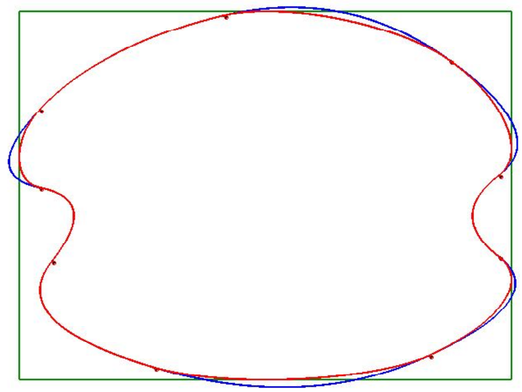

Figure 7: Example 1

The second example shown in Fig 8 is a complex problem in which a portion of curve may intersect two boundary segments. For solution of such complex problem we have determined the resultant curve with value by using the algorithm described in sections 5.2 such that the final curve must passes through the joining point of the boundary segments and it avoids all the boundary segments

Figure 8: Example 2

6. Conclusions

We presented 2D interpolation schemes which all strive to produce spline curves interpolated to set of given data point. The schemes also work for 3D, but this was not tested. It may be more desirable to produce an interpolated spline curve that avoid the given circular arc polygon boundary.

References

[1] He-Jin Gu, Jun-Hai Yong, Jean-Claude Paul, Fuhua (Frank) Cheng, constructing G1 quadratic Bezier curves with arbitrary endpoint tangent vectors. "The 11th IEEE International Conference on CAD/Graphics -CAD/GRAPHICS 2009 (2009) 263-267"

[2] Xunnian Yang*, Curve fitting and fairing using conic splines. Computer-Aided Design 36 (2004) 461–472

[3] Lian Fang, A rational quartic Bézier representation for conics, Computer Aided Geometric Design 19 (2002) 297–312

[4] Constrained interpolation with rational cubics D.S. Meek a, B.H. Ongb,, D.J. Walton Computer Aided Geometric Design 20 (2003) 253–275

[5] Meek, D.S., Ong, B.H., Walton, D.J., A constrained guided G^1continuous spline curve. Computer-Aided Design 35 (2003) 591–599

[6] Ong, B.H., Unsworth, K.K., 1992. On non-parametric constrained interpolation.: in Lyche, T., Schumaker, L.L. (Eds.), Mathematical Methods in Computer Aided Geometric Design II. Academic Press, pp. 419–430.

[7] Interactive shape preserving interpolation by curvature continuous rational cubic splines Chris Seymour, Keith Unsworth ,journal of computational and applied mathematics September 1998.

[8] T.A. Foley, T.N.T. Goodman, K. Unsworth, An algorithm for shape preserving parametric interpolating curves with G^2 continuity, in: T. Lyche, L.L. Schumaker (Eds.), Mathematical Methods in Computer Aided Geometric Design, Academic Press.

[9] T.N.T. Goodman, Shape preserving interpolation by parametric rational cubic splines in: Intnl. Series of Numerical Mathematics, vol. 86, Birkhauser Basel, 1988, pp. 149- 158.

[10] Chan, E.S., Ong, B.H., 2001. Range restricted scattered data interpolation using convex combination of cubic Bézier triangles.J. Comput. Appl. Math. 136, 135–147.

[11] T.N.T. Goodman, B.H. Ong, Shape preserving interpolation by space curves, Computer Aided Geom. Design 15 (1997) 1-17.

[12] Goodman, T.N.T., Ong, B.H., Unsworth, K., 1991. Constrained interpolation using rational cubic splines. In: Farin, G. (Ed.), Nurbs for Curve and Surface Design. SIAM, Philadelphia, pp. 59–74.

[13] McKerrow, P.J., 1991. Introduction to Robotics. Addison- Wesley, New York.

[14] Goodman TNT, Ong BH, Unsworth K. Constrained interpolation using rational cubic splines. In: Farin G, editor. NURBS for curve and surface design. Philadelphia: SIAM; 1991. p. 59–74.