Short Communication

An Algorithm for Computing the Shortest Distance between a Point and Quadratic Bezier Curve

NOURI AHMAD SULEIMAN

Department of Mathematics, Faculty of Science and Literature, Harran University, Turkey

*Corresponding author e-mail: Shamoo1981@yahoo.com

ABSTRACT

The problem of data fitting using Bezier curve is one of the most important problems in the modern science, during this process, the problem of determining the shortest distance between a point and Bezier curve arises, so when we use a second degree Bezier curve for fitting data points, the problem will be changed into solving third degree polynomial, but when we use third degree Bezier curve, the problem is changed into finding roots of polynomial too, but in this case the polynomial is fifth degree, in this paper we will discuss the minimum distance between a point and a Bezier curve with three control points, i.e. quadratic Bezier curve.

Keywords: Bezier curve, Data fitting, Roots finding

INTRODUCTION

Bezier curves [1,2] are widely used in computer graphics to produce smooth curves [3]. The curves are named after the French engineer Pierre Bezier who was an applied mathematician with the car manufacturer Renault. In the early 1960's, he began searching for ways to automate the process of designing cars. His method has been the basis of the modern field of Computer Aided Geometric Design (CAGD).

Using Bezier curves changed the common attitude of graphics and introduced the concept of vector graphics as opposed to raster images. A raster image is described in terms of pixels. Each pixel is coded with a color, depending of various algorithm available (jpg, bmp, etc.). The two major problems of raster images is that they are generally big in term of file size and they are not scalable, that means; if you zoom in the image, you will just see the square pixels getting bigger and bigger. In

this aspect, Bezier curves are the adopted solution to describe an image in terms of its mathematical representation. The vector graphic file contains the coordinates of some basic points (control points) to describe a series of curves, if the user wants to zoom in, it is only a matter of increasing the space between the control points and it can redraw a perfectly smooth curve again.

Let us describe the difference between **Figures 1 and 2:**

In **Figure 1**, Raster image that is composed of grid of pixels (often called bitmap images), Raster images are best used for photographs and images with subtle shading, Raster file formats include TIFF, JPG, GIF, BMP, PNG, PSD, and EPS.

In **Figure 2**, Vector images consist of points, lines and curves based on mathematical definitions, the edges of vector graphically smooth at any size or

resolution. Fonts, line art, charts and graphs are typically vector-based, vector files formats include EMF, EPS, PDF, and PS.

Bezier curve of degree n

Bezier curve is a parametric curve [4]. Bezier curve of degree n is given as:

$$B(t) = \sum_{i=0}^{n} \binom{n}{i} t^i (1-t)^{n-i} P_i, \qquad t \in [0,1]$$

Where $B(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2, \qquad t \in [0,1]$, are the control points?

Quadratic bezier curve

The formula of Quadratic Bezier curve is:

$$B(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2, \qquad t \in [0,1]$$

Where $(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3, \quad t \in [0,$ are the control points?

Cubic bezier curves

Cubic Bezier curve have three four control points, and has the formula:

$$B(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3, \qquad t \in [0,1] \cdot$$

Suppose we have a point $q(\xi, \eta)$. Define a function

$$f(t) \equiv | B(t) - q |^2 . \ (1)$$

Then the problem of finding the shortest distance between the point $q(\xi, \eta)$ and a quadratic Bezier curve is mathematically expressed as follow:

Problem: Find the value $f(t^*) = \min_{0 \le t \le 1} f(t)$ such that:

$$f(t^*) = \min_{0 \le t \le 1} f(t) . \ (2)$$

The value t^* minimizes the distance between q and the curve $B(t^*)$, i.e., $B(t^*)$ is the point located on the curve $B(t)$ which is the closest one to the point q. From (1), it is easily to see that the function $f(t)$ is a polynomial of fourth degree,

$$f(t) = d_4 t^4 + 4d_3 t^3 + 2d_2 t^2 + 4d_1 t + d_0, \qquad (3)$$

Where

$d_4 \equiv (x_0 - 2x_1 + x_2)^2 + (y_0 - 2y_1 + y_2)^2$
$d_3 \equiv (x_0 - 2x_1 + x_2)(x_1 - x_0) + (y_0 - 2y_1 + y_2)(y_1 - y_0)$
$d_2 \equiv (x_0 - 2x_1 + x_2)(x_0 - \xi) + (y_0 - 2y_1 + y_2)(y_0 - \eta) + 2(x_1 - x_0)^2 + 2(y_1 - y_0)^2$
$d_1 \equiv (x_1 - x_0)(x_0 - \xi) + (y_1 - y_0)(y_0 - \eta)$
$d_0 \equiv (x_0 - \xi)^2 + (y_0 - \eta)^2.$

To solve this Problem, we have to find the roots of the equation $f'(t) = 0$. Differentiating $f(t)$ given in (3) yields:

$$f'(t) = 4P(t),$$

Where

$$P(t) \equiv d_4 t^3 + 3d_3 t^2 + d_2 t + d_1 \qquad (4)$$

As we see is a polynomial of third degree.

Thus, the problem of determining the shortest distance between a point and quadratic Bezier curve is reduced to a problem of finding the roots of a third degree polynomial

There are many methods for finding the roots of the polynomial (4), for example Müller's method [5].

After determining the real roots of the polynomial $P(t)$, we select only those roots that lie in the interval $[0,1]$. Let the set of chosen roots be $\{t_\mu\}$. Then t^* is chosen from the set $\{\{t_\mu\}, 0, 1\}$ for which

$$f(t^*) = \min\{\{f(t_\mu)\}, f(0), f(1)\} . \ (5)$$

Algorithm

Inputs $P_1(x_1, y_1), P_1(x_1, y_1), q(\xi, \eta), q(\xi, \eta)$.

Compute the coefficients of the polynomial

$$f(t) = d_4 t^4 + 4d_3 t^3 + 2d_2 t^2 + 4d_1 t + d_0$$

Deriving $f(t)$, we get new polynomial

$$Q_3(t) = c_3 t^3 + c_2 t^2 + c_1 t + c_0$$

Find the roots of $\{t_\mu\}$, where its coefficients are determined

Define a set $\{t_\mu\}$ of real roots that lie in the segment

$\{t_\mu\}, 0, 1\}$

Choose the knot from the set $\{t_\mu\}, 0, 1\}$, such that
$$f(t^*) = \min\{\{f(t_\mu)\}, f(0), f(1)\}$$

If the roots of the polynomial $Q_3(t)$ don't exist in the segment $[0,1]$, then we choose t^* that satisfies

$$f(t^*) = \min\{f(0), f(1)\}$$

Output t^*, $f(t^*)$

Numerical Experiments

Test 1

Suppose we want to expert our algorithm in calculating the minimum distance between a point with coordinates (4,9), and quadratic Bezier curve with the control points: $p_0(0,0), P_1(4,6) and P_2(8,0)$, we run the program of achieved algorithm using Maple [6], we get the value of the parameter t which minimizes distance between the point and the curve mentioned above, $t^* = 0.5$. Evaluating $t^* = 0.5$ at $t^* = 0.5$ we get $f(t^*) \equiv | B(t^*) - q |^2 = 36$, easily we can conclude that the minimum distance between the point and the curve is the square root of $f(t^*)$, so that it is 6.

Simple analyzing of above figure, shows that the parameter t^* corresponding with the point $B(t^*)$ is located in the middle of the segment line $P_0 P_2$ whose long is 1, so t^* is equal to 0.5. Easily we can see that the shortest distance between q and $q(9,0.5)$ is equal to 6, and this harmonizes with the result gotten by the algorithm.

Test 2

Now we will test the algorithm by choosing new point $q(9,0.5)$ and Bezier curve whose control points are $P_0(0,0) P_1(4,6) and P_2(8,0)$.

Running a program of the algorithm gives $t^* = 1.008933271$, this value is out of the interval [0, 1]. In this case according to the algorithm we compute $f(0)$ and $f(1)$; $f(t) \equiv | B(t) - q |^2$, we get $f(0) = 81.25$ and $f(1) = 1.25$, then we choose the minimum value of these two values which is 1.25 and so that the minimum distance between the given point and the curve is $q = \sqrt{1^2 + (0.5)^2} = \sqrt{1.3} = 1.118$.

Analyzing the above figure, the minimum distance between the point q and Bezier curve logically is the line segment aq, we can easily compute it from the triangle aqb, $q = \sqrt{1^2 + (0.5)^2} = \sqrt{1.3} = 1.118$.

So we see that our analyzing does harmonize with the result of the established algorithm.

Test 3

Now if the point is (-2, 1) and the control points of Bezier curve are $P_1(4,6) and P_2(8,0)$, $P_1(4,6) and P_2(8,0)$ in this case we get $t^* = -0.01671531629$, this value doesn't belong to the interval [0, 1] so we have to compute $f(0)$ and $f(0) = 5$.

$f(0) = 5$ And $f(0) = 5$, we have chosen the minimal value 5, so that $f(0) = 5$, taking the square

root of $f(0) = 5$, and so that the minimum distance between the point and the curve in this case is $\sqrt{5} = 2.2360679775$.

Noticing the latest figure, the closest point of Bezier curve to the point q is the point (0,0) so that $bq = \sqrt{1^2 + (-2)^2} = \sqrt{5} = 2.2360679775$, and we get exactly by running the program of the established algorithm.

Test 4

Let us choose the point $q(8,3)$ and the control points of Bezier curve are $P_0(0, 0)$, $P_1(4, 6)$ and $P_2(8, 0)$.

In this case we find that $f(t) \equiv | B(t) - q |^2 = 3.55555559$ and $f(t^*) \equiv | B(t^*) - q |^2 = 3.55555559$, and consequently the minimum distance between the point and the curve is $\sqrt{3.55555559} = 1.88561809$.

In this case, noticing the above figure, we can, theoretically, expert our algorithm by computing the parameter t^*, we first draw the tangent and the line qq^* being orthogonal to each other, then the projection of q^* on the line segment [0, 1] is t^*.

t^* Is close to the value 0.8 on the line segment, and approximately is equal to 0.83. In all cases tests, verify the algorithm does work well **(Figures 3-9).**

Conclusion

This paper presents an improved method for finding the shortest distance between a point and quadratic Bezier curve; this method converts the problem into finding the roots of third degree polynomial, and plays very important role in data fitting using Bezier curves. It is a basic element for curves and surfaces design and for many different fields too. After the theoretic study of algorithm we have changed to write program of it and running it by Maple, we have showed four cases, and analyzed them comparing theoretical solution with programmed one.

REFERENCES

1. Prautsch H, Boehm W, Marco, Paluszuy (2002) Bezier- and B- spline techniques.

2. Iqbal K (2013) Fundamental Engineering Optimization Methods.

3. Shifrin T (2016) Differential Geometry: A First Course in Curves and Surfaces

4. Thomas BG, Thomas Calculus, Massachusetts Institute of Technology.

5. Bin Y (2016) Roots of Polynomials (Com S 477/577 Notes).

6. Bernardin L, Chin P, DeMarco P, Geddes KO (2011) Maple programming Guide.
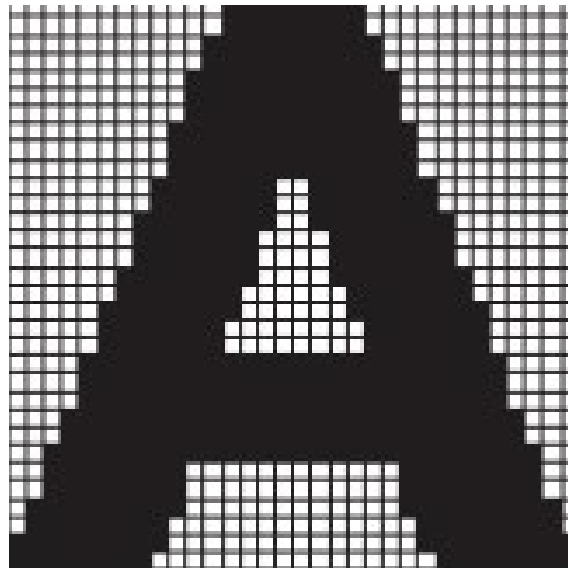
Figure 1. Raster image that is composed of grid of pixels (often called bitmap images), Raster images are best used for photographs and images with subtle shading, Raster file formats include TIFF, JPG, GIF, BMP, PNG, PSD, and EPS
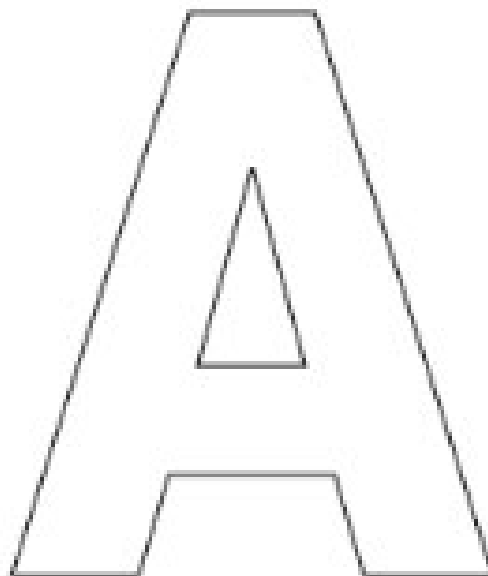
Figure 2. Vector images consist of points, lines and curves based on mathematical definitions, the edges of vector graphically smooth at any size or resolution. Fonts, line art, charts and graphs are typically vector-based, vector files formats include EMF, EPS, PDF, and PSV.
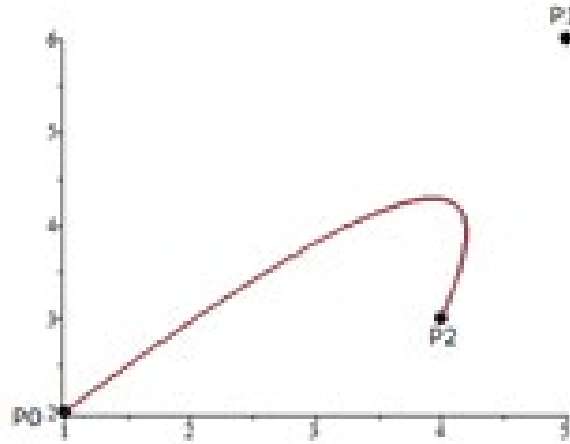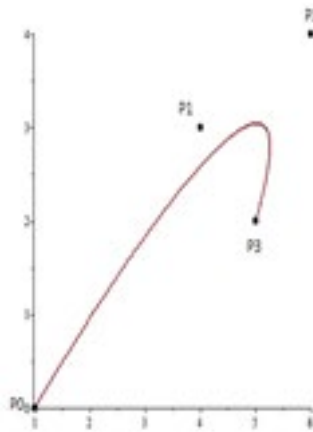
Figure 3: Quadratic Bezier curve.

Figure 4: Cubic Bezier curve.
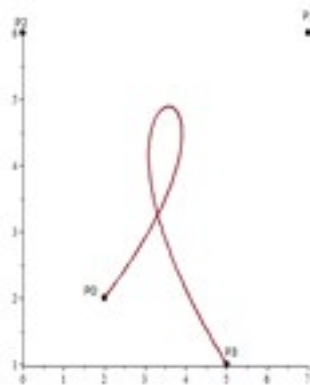
Figure 5: Looped cubic Bezier curve.

Figure 6: Simple analyzing of above figure, shows that the parameter corresponding with the point is located in the middle of the segment line Po P2 whose long is 1, so is equal to 0.5. Easily we can see that the shortest distance between q and is equal to 6, and this harmonizes with the result gotten by the algorithm.

Figure 7: Analyzing the figure below, the minimum distance between the point q and Bezier curve logically is the line segment aq, we easily we compute it from the triangle aqb.

Figure 8: Noticing the latest figure, the closest point of Bezier curve to the point q is the point (0,0), so that , and this what we get exactly by running the program of the established algorithm.

Figure 9: Let us choose the point q (8, 3) and the control points of Bezier curve are P⁰ (0, 0), P_1 (4, 6) and P_2 (8, 0). In this case we find that $\sqrt{3.55555559} = 1.885618092$ and $\sqrt{3.55555559} = 1.885618092$, and consequently the minimum distance between the point and the curve is $\sqrt{3.55555559} = 1.885618092$.