B.A.I.O.C.A.

Bare Attempt to Improve Offset Curve Algorithm

Giandomenico Rossi   Nicola Fontana

April 16, 2013

Computer Aided Design (CAD) and related software is often based on cubic Bézier curves: the Postscript language and consecuentely the PDF file format are two widespread examples of such software. Defining an optimal algorithm for approximating a Bézier curve parallel to the original one at a specific distance (the so called "offset curve") is a big requirement in CAD drafting: it is heavily used while constructing derived entities (e.g., a fillet) or to express machining allowance.

This document describes an algorithm suitable for CAD purposes. In those cases, the starting and ending points of the offset curve **must** have coordinates and slopes coincident with the perfect solution, so the continuity with previous and next offseted entity is preserved.

1 Mathematic

The generic formula for a cubic Bézier curve is

$$\vec{B}(t) = b_0(t)\vec{B}_0 + b_1(t)\vec{B}_1 + b_2(t)\vec{B}_2 + b_3(t)\vec{B}_3$$

where

$$\vec{B}_i \equiv (B_x^i, B_y^i) \in \mathbb{R}; \qquad i = 0, 1, 2, 3$$

$$b_i(t) \equiv \binom{3}{i} t^i (1-t)^{3-i}.$$

Given in $\{t_i\}_{i=0}^{n}$ a set of values for t chosen in some manner with $t_0 = 0, t_n = 1$ and in R the required distance of the offset curve,

$$\vec{C}_i = \vec{B}(t_i) + R \left. \frac{\dot{B}_y, -\dot{B}_x}{\sqrt{\dot{B}_x^2 + \dot{B}_y^2}} \right|_{t=t_i} \qquad \forall t_i \tag{1}$$

is the equation of the offset curve that has in $\{\vec{C}_i\}_{i=0}^{n}$ the set of its points and where $\dot{\vec{B}} \equiv (\dot{B}_x, \dot{B}_y) \equiv \left(\frac{d}{dt}B_x(t), \frac{d}{dt}B_y(t)\right)$.

We must find the Bézier curve

$$\vec{Q}(t) = b_0(t)\vec{Q}_0 + b_1(t)\vec{Q}_1 + b_2(t)\vec{Q}_2 + b_3(t)\vec{Q}_3 \tag{2}$$

where

$$\vec{Q}_i \equiv (Q_x^i, Q_y^i) \in \mathbb{R} \qquad i = 0, 1, 2, 3$$

which best fits (1) within the needed constraints, that is:

1. $\vec{Q}(0) = \vec{C}_0$ and $\vec{Q}(1) = \vec{C}_n$ (interpolation);
2. $\dot{\vec{Q}}(0) = \dot{\vec{B}}(0)$ and $\dot{\vec{Q}}(1) = \dot{\vec{B}}(1)$ (tangents), where $\dot{\vec{Q}} \equiv \frac{d}{dt}\vec{Q}(t)$.

Condition 1 implies that $Q_0 = C_1$ and $Q_3 = C_n$.

Condition 2 implies that $\dot{\vec{Q}}_0 = \dot{\vec{B}}_0$ and $\dot{\vec{Q}}_3 = \dot{\vec{B}}_3$. Imposing by convention

$$\vec{P}_i = \vec{B}_{i+1} - \vec{B}_i; \qquad i = 0, 1, 2; \tag{3}$$

we can calculate $\dot{\vec{B}}_0$ and $\dot{\vec{B}}_3$ directly from the hodograph of $\vec{B}(t)$:

$$\begin{aligned}
&\dot{\vec{B}}(t) = 3\left[(1-t)\vec{P}_0 + 2t(1-t)\vec{P}_1 + t\vec{P}_2\right];\\
&\dot{\vec{B}}(0) = 3\vec{P}_0 \equiv \dot{\vec{B}}_0 = \dot{\vec{Q}}_0;\\
&\dot{\vec{B}}(1) = 3\vec{P}_2 \equiv \dot{\vec{B}}_3 = \dot{\vec{Q}}_3.
\end{aligned}$$

Knowing that one of the properties of a Bézier curve is the start of the curve is tangent to the first section of the control polygon and the end is tangent to the last section, condition 2 is hence equivalent to:

$$\begin{aligned}
&\vec{Q}_1 = \vec{Q}_0 + \frac{r}{3}\dot{\vec{Q}}_0 = \vec{Q}_0 + r\vec{P}_0; \qquad r, s \in \mathbb{R}\\
&\vec{Q}_2 = \vec{Q}_3 + \frac{s}{3}\dot{\vec{Q}}_3 = \vec{Q}_3 + s\vec{P}_2.
\end{aligned} \tag{4}$$

Substituting (4) in (2) we get

$$\vec{Q}(t) = b_0(t)\vec{Q}_0 + b_1(t)\vec{Q}_0 + b_1(t)r\vec{P}_0 + b_2(t)\vec{Q}_3 + b_2(t)s\vec{P}_2 + b_3(t)\vec{Q}_3.$$

Determine the value of r and s that minimizes the quantity $\phi = \sum\left[\vec{C}_i - \vec{Q}(t_i)\right]^2$, equivalent to solve the system

$$\begin{cases} \dfrac{\delta\phi}{\delta r} = 0;\\ \dfrac{\delta\phi}{\delta s} = 0. \end{cases}$$

Now, given the shortcuts[1] $\sum \equiv \sum_{i=1}^{n-1}$ and $b_j \equiv b_j(t_i)$, we can write ϕ as

$$\phi(r,s) = \sum \left[\vec{C}_i - b_0\vec{Q}_0 - b_1\vec{Q}_0 - rb_1\vec{P}_0 - b_2\vec{Q}_3 - sb_2\vec{P}_2 - b_3\vec{Q}_3\right]^2;$$

that, applied to the previous system, bring us to the following linear system

$$\begin{cases} \sum\left(\vec{C}_i - b_0\vec{Q}_0 - b_1\vec{Q}_0 - rb_1\vec{P}_0 - b_2\vec{Q}_3 - sb_2\vec{P}_2 - b_3\vec{Q}_3\right)\left(-2b_1\vec{P}_0\right) & = 0; \\ \sum\left(\vec{C}_i - b_0\vec{Q}_0 - b_1\vec{Q}_0 - rb_1\vec{P}_0 - b_2\vec{Q}_3 - sb_2\vec{P}_2 - b_3\vec{Q}_3\right)\left(-2b_2\vec{P}_2\right) & = 0; \end{cases}$$

from which we get

$$\begin{cases} \sum\begin{pmatrix} b_1\langle\vec{C}_i,\vec{P}_0\rangle - b_0b_1\langle\vec{Q}_0,\vec{P}_0\rangle - b_1b_1\langle\vec{Q}_0,\vec{P}_0\rangle - rb_1b_1\langle\vec{P}_0,\vec{P}_0\rangle \\ -b_1b_2\langle\vec{Q}_3,\vec{P}_0\rangle - sb_1b_2\langle\vec{P}_2,\vec{P}_0\rangle - b_1b_3\langle\vec{Q}_3,\vec{P}_0\rangle \end{pmatrix} & = 0; \\ \sum\begin{pmatrix} b_2\langle\vec{C}_i,\vec{P}_2\rangle - b_0b_2\langle\vec{Q}_0,\vec{P}_2\rangle - b_1b_2\langle\vec{Q}_0,\vec{P}_2\rangle - rb_1b_2\langle\vec{P}_0,\vec{P}_2\rangle \\ -b_2b_2\langle\vec{Q}_3,\vec{P}_2\rangle - sb_2b_2\langle\vec{P}_2,\vec{P}_2\rangle - b_2b_3\langle\vec{Q}_3,\vec{P}_2\rangle \end{pmatrix} & = 0. \end{cases}$$

Given the additional conventions

$$\begin{aligned} D_0 &\equiv \sum b_1\langle\vec{C}_i,\vec{P}_0\rangle; \\ D_2 &\equiv \sum b_2\langle\vec{C}_i,\vec{P}_2\rangle; \\ E_{jk} &\equiv \sum b_jb_k; \end{aligned} \tag{5}$$

we can substitute to get

$$\begin{cases} D_0 - E_{01}\langle\vec{Q}_0,\vec{P}_0\rangle - E_{11}\langle\vec{Q}_0,\vec{P}_0\rangle - rE_{11}\langle\vec{P}_0,\vec{P}_0\rangle - E_{12}\langle\vec{Q}_3,\vec{P}_0\rangle \\ \qquad -sE_{12}\langle\vec{P}_2,\vec{P}_0\rangle - E_{13}\langle\vec{Q}_3,\vec{P}_0\rangle = 0; \\ D_2 - E_{02}\langle\vec{Q}_0,\vec{P}_2\rangle - E_{12}\langle\vec{Q}_0,\vec{P}_2\rangle - rE_{12}\langle\vec{P}_0,\vec{P}_2\rangle - E_{22}\langle\vec{Q}_3,\vec{P}_2\rangle \\ \qquad -sE_{22}\langle\vec{P}_2,\vec{P}_2\rangle - E_{23}\langle\vec{Q}_3,\vec{P}_2\rangle = 0; \end{cases}$$

and derive the following known terms

$$\begin{aligned} A_1 &= D_0 - \langle\vec{Q}_0,\vec{P}_0\rangle(E_{01}+E_{11}) - \langle\vec{Q}_3,\vec{P}_0\rangle(E_{12}+E_{13}); \\ A_2 &= D_2 - \langle\vec{Q}_0,\vec{P}_2\rangle(E_{02}+E_{12}) - \langle\vec{Q}_3,\vec{P}_2\rangle(E_{22}+E_{23}); \\ A_3 &= \langle\vec{P}_0,\vec{P}_0\rangle E_{11}; \\ A_4 &= \langle\vec{P}_0,\vec{P}_2\rangle E_{12}; \\ A_5 &= \langle\vec{P}_2,\vec{P}_2\rangle E_{22}. \end{aligned} \tag{6}$$

[1] Either C_0 and C_n are not considered because they have been already used as Q_0 and Q_3 by the interpolation constraint.

The system is hence reduced to

$$\begin{cases} rA_3 + sA_4 & = A_1; \\ rA_4 + sA_5 & = A_2; \end{cases}$$

from which we can calculate r and s

$$r = \frac{A_1A_5 - A_4A_2}{A_3A_5 - A_4A_4}; \qquad s = \frac{A_3A_2 - A_1A_4}{A_3A_5 - A_4A_4}. \tag{7}$$

2 Algorithm

1. Select $\{t_i\}_{i=0}^n$ as shown in section 3;
2. compute $\{\vec{C}_i\}_{i=0}^n$ with (1): $\vec{Q}_0 = C_0$ and $\vec{Q}_3 = C_n$;
3. calculate $\vec{P}_0$ and $\vec{P}_2$ with (3);
4. calculate D_0, D_2, E_{01}, E_{02}, E_{11}, E_{12}, E_{13}, E_{22} and E_{23} with (5);
5. calculate A_1, A_2, A_3, A_4 and A_5 with (6);
6. calculate r and s with (7);
7. get $\vec{Q}_1$ and $\vec{Q}_2$ from (4).

$\vec{Q}_0$ and $\vec{Q}_3$ are respectively the starting and ending points of the offset Bézier curve while $\vec{Q}_1$ and $\vec{Q}_2$ are its control points.

3 Choosing t_i

To select the $\{t_i\}_{i=0}^{n}$ set of values for t needed by the offsetting algorithm, we can use different methods. Here are some basic ones: no further research is performed to check the quality of the results.

3.1 Method 1: too lazy to think

The most obvious method is to directly use evenly spaced time values:

$$t_i = \frac{i}{n}.$$

3.2 Method 2: squared distances

Let's select some $\{\vec{F}_i\}_{i=0}^{n}$ points on $\vec{B}(t)$, for instance by resolving the t values got from the lazy method. The following formula will partition the Bézier curve proportionally to their squared distances:

$$t_0 = 0; \qquad f = \sum_{i=1}^{n} \left(\vec{F}_i - \vec{F}_{i-1}\right)^2$$
$$t_i = t_{i-1} + \frac{\left(\vec{F}_i - \vec{F}_{i-1}\right)^2}{f}.$$

3.3 Method 3: distances

A variant of the previous method that uses distances instead of squared distances. This is computationally more intensive because the norm of a vector $\|\vec{F}\| \equiv \sqrt{F_x^2 + F_y^2}$ requires a square root.

$$t_0 = 0; \qquad f = \sum_{i=1}^{n} \|\vec{F}_i - \vec{F}_{i-1}\|$$
$$t_i = t_{i-1} + \frac{\|\vec{F}_i - \vec{F}_{i-1}\|}{f}.$$