Vratislava Mošová

Bézier curves and their applications

In: Jan Chleboun and Petr Přikryl and Karel Segeth (eds.): Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar. Dolní Maxov, June 6-11, 2004. Institute of Mathematics AS CR, Prague, 2004. pp. 174–179.

Persistent URL: `http://dml.cz/dmlcz/702792`

BÉZIER CURVES AND THEIR APPLICATIONS

Vratislava Mošová

1. Shortly about the history of Bézier polynomials

Bézier polynomials came in the center of interest in the eighties when effective personal computers were developed. They became important tool in computer aided geometric design (CAGD).

The first who used the new type of construction for curves were Frenchmen Paul de Faget de Casteljau and Pierre Bézier. They were employees of Paris car companies. The first one worked for Renault and the second one for Citroën. They found such constructions of curves and surfaces that are proper for the realization of these objects on computer.

P. Bézier presented his "basic curve" as the intersection of two elliptic cylinders situated in the parallelepiped. It has this effect: Affine transformations of the parallelepiped reflect in affine transformations of the curve.

De Casteljau chose another approach. He formed his curve with respect to the given control polygon. He left construction of curve through points situated on it and used the definition of curve through points situated near it. He defined such a curve by means of Bernstein basis. It gave de Casteljau the possibility to develop a system which primarily aimed at the ab initio design of curves.

Later it was proved that curves received by Bézier and de Casteljau are identical. Because P. Bézier, in contrast to de Casteljau, published his results we use the name "Bézier curve" for it. But the algorithm for construction of these curves got its name after P. de Casteljau.

2. Bézier polynomials

We will deal with parametric expression of curves now. Planar or space curves can be represented by a polynomial. For our purposes we consider polynomials that are generated by the Bernstein basis $B_0^n(x), \ldots, B_n^n(x)$.

Definition 1 The *Bernstein polynomial* of degree n is given for $x \in \langle a, b\rangle$ by the formula

$$B_i^n(x) = \frac{1}{(b-a)^n} \binom{n}{i} (x-a)^i (b-x)^{n-i}, \quad i = 0, \ldots, n. \qquad (1)$$

Because $B_i^n(x)$ are invariant to affine transformations we can, without loss of generality, put $t = (x-a)/(b-a)$ and define Bernstein polynomials on interval $\langle 0, 1\rangle$.

Theorem 1 *The Bernstein polynomial has for $t \in \langle 0,1\rangle$ the form*

$$B_i^n(t) = \binom{n}{i} t^i(1-t)^{n-i},\ i=0,\dots,n. \tag{2}$$

Theorem 2 *The $B_i^n(t),\ i=0,\dots,n$, are polynomials of degree n that have a zero in $t \in \langle 0,1\rangle$.*

$$B_i^n(t) \geq 0, \quad t \in \langle 0,1\rangle. \tag{3}$$

$$\sum_{i=0}^{n} B_i^n(t) = 1, \quad t \in \mathbb{R}. \tag{4}$$

$$B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t), \quad t \in \mathbb{R}. \tag{5}$$

$$(B_i^n(t))' = n(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)). \tag{6}$$

We can define the Bézier curve now.

Definition 2 Let $P_0, P_1, \dots, P_n$ be points in $\mathbb{R}^m,\ m=2,3$. Then

$$p(t) = \sum_{i=0}^{n} P_i B_i^n(t),\ t \in \langle 0,1\rangle, \tag{7}$$

is the *Bézier polynomial*, points $P_0, \dots, P_n$ are *control points*, and the polygon determined by these points is the *control polygon.*

We receive properties of the Bézier polynomials from the properties of Bernstein polynomials given above.

Definition 3 Let $\mathcal{P}_n^m$ be the space formed by polynomials

$$p(x) = \sum_{i=0}^{n} a_i x^i,\ x \in \mathbb{R},\ a_0, \dots, a_n \in \mathbb{R}^m.$$

Theorem 3 *Let $p(t)$ be the Bézier polynomial given by the relation* (7)*. Then it has the following properties:*

1. *The polynomial $p(t) \in \mathcal{P}_n^m,\ m=2,3$.*
2. *The polynomial $p(t)$ is invariant to affine transformations.*
3. *At the endpoints,*
$$p(0) = P_1,\ p(1) = P_n. \tag{8}$$
4. *Subpolynomials of the Bézier polynomial defined by the relation*

$$p_i^k(t) = \sum_{l=0}^{k} P_{i+l} B_l^k(t) \tag{9}$$

satisfy the recursion formula

$$p_i^k(t) = (1-t)p_i^{k-1}(t) + t p_{i+1}^{k-1}(t), \ i = 0, \dots, n-k, \ k = 1, \dots, n. \tag{10}$$

5. *The derivatives at the endpoints satisfy*

$$p'(t) = n \sum_{i=0}^{n-1} (P_{i+l} - P_i) B_i^n(t). \tag{11}$$

It means that

$$p'(0) = n(P_1 - P_0), \, p'(1) = n(P_n - P_{n-1}). \tag{12}$$

Graph of the Bézier polygon is often referred to as *Bézier curve.* From the previous theorems we can see that:
The Bézier curve is contained in the convex hull of control points (see (1), (3), (4)).
The Bézier curve and its Bézier polygon have the same endpoints (see (8)).
The tangent lines to the Bézier curve at the endpoints merge with sides of its Bézier polygon (see (12)).
The Bézier curve has only one extremum (see (11)).

The most important relation is the recursion formulae (10). It is the core of *de Casteljau algorithm* that is used to compute points on the Bézier curve. One point on Bézier curve of degree n is received by means of successive convex combinations of control points

$$P_i^k = (1-t)P_i^{k-1} + tP_{i+1}^{k-1}, \ i = k, \dots, n, \ k = 1, \dots, n,$$

where $P_i^k = p_i^k(t)$, $t = const.$
Because de Casteljau algorithm has the form

$$\begin{array}{lllll} P_0^0 = P_0 & & & & \\ P_1^0 = P_1 & P_0^1 & & & \\ P_2^0 = P_2 & P_1^1 & P_0^2 & & \\ \quad \dots & \dots & \dots & \dots & \dots \\ P_n^0 = P_n & P_{n-1}^1 & P_{n-2}^2 & \dots & P_0^n \end{array}$$

the computation cost for de Casteljau algorithm is of order $O(n^2)$.
We demonstrate the basic idea of reconstruction of one point on Bézier curve of degree 2 now. We can write

$$
\begin{aligned}
p(t) &= P_0(1-t)^2 + 2P_1(1-t)t + P_2t^2 \\
&= (1-t)\underbrace{(P_0(1-t)+P_1t)}_{P_0^1} + t\underbrace{(P_1(1-t)t+P_2t)}_{P_1^1} \\
&= \underbrace{(1-t)P_0^1 + tP_1^1}_{P_0^2}.
\end{aligned}
$$

If t is constant then P_0^2 represents one point on Bézier curve.

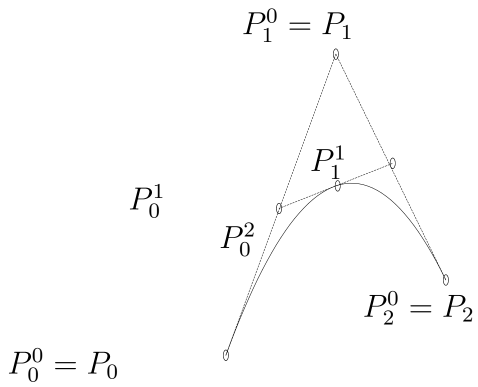

Fig. 1: *Construction of one point on Bézier curve.*

The Bézier curves gained such popularity thanks to the de Casteljau algorithm. On the basis of this algorithm, it is possible to realize simply and quickly various manipulations with Bézier curves.
If we divide the original curve of degree n at the point P_n^n we receive two new control polygons that have vertices $P_0^0, P_0^1, \ldots, P_0^n$ and $P_0^n, P_1^{n-1}, \ldots, P_n^0$. These polygons determine two new curves that have the same degree as the original curve.
We can also connect two Bézier curves in such a way that tangent lines at points of connection coincide. When $p(t)$ is a Bézier polynomial of degree n and $q(t)$ is a Bézier polynomial of degree m, and they are governed by points $P_0, \ldots, P_n$ and $G_0, \ldots, G_m$ then

$$p(1) = g(0) \Rightarrow P_m = G_0,$$

$$p'(1) = g'(0) \Rightarrow n(P_n - P_{n-1}) = m(G_1 - G_0) \Rightarrow G_1 = \frac{n}{m}(P_n - P_{n-1}) + P_n.$$

If we divide the edges of the original polygon of degree n into $n+1$ parts and, at every of the n points, define one new control point

$$P_i^1 = t_i P_{i-1} + (1-t_i)P_i,\ t_i = \frac{i}{n+1},\ i = 0,\dots,n-1,$$

we obtain a new control polygon with $n+1$ vertices and make the degree of Bézier curve higher.

We can do a lot of generalizations.
If we replace the control points $P_0, P_1, \dots, P_n$ by points $w_0P_0, w_1P_1, \dots, w_nP_n$ ($w_i > 0$ are weights) and use the homogenous basis then we receive the *rational Bézier curve*

$$p(t) = \frac{\sum_{i=0}^{n} w_i P_i B_i^n(t)}{\sum_{i=0}^{n} w_i B_i^n(t)} \tag{13}$$

that is invariant to projective transformations.
We can also form two parametrical objects - *Bézier surfaces*. Let the mesh in space be determined by points $P_{ij} \in \mathbb{R}^3,\ i = 0,\dots,n,\ j = 0,\dots,m$. Then we define the Bézier surface

$$p(t) = \sum_{i=0}^{n}\sum_{j=0}^{m} P_{ij} B_i^n(t) B_j^n(t). \tag{14}$$

It is possible to adapt the De Casteljau algorithm and use it to receive points on rational Bézier curve or on Bézier surface.

3. Applications of Bézier polynomials

The Bézier polynomials in connection with the de Casteljau algorithm are predetermined to serve for computer graphic, because they give us possibility to visualize and manipulate geometrical objects on computer effectively and rapidly. But they have their drawbacks. They are not a universal tool. The Bézier polynomials (1) can't model conic curves and surfaces. It is possible to use the rational Bézier polynomials (13) for this purpose. Problems appeared when it was necessary to apply software for the realization of rational Bézier polynomials together with software for the realization of splines on computer. This problem was solved after de Boor's discovery of recursive representation of B-splines. *Non-uniform rational B-splines* (NURBS) were created as a generalization of B-splines and rational Bézier polynomials. They allow a unified geometrical representation because conic or spline curves and spaces can be expressed in the form of piecewise rational polynomials. This is the reason why NURBS form standard equipment of contemporary graphical systems.

In the end we note that Bézier polynomials and their generalizations are also useful for solving differential equations. The rational Bézier polynomials were used for the description of geometrical configuration and for approximation of velocity and preasure functions in the modelling of dynamic behaviour of the rigid rotating shaft in real liquid (see [5]).

References

[1] J. Drdla: *Geometrické modelování křivek a ploch.* UP Olomouc 1993.

[2] G. Farin: *Curves and surfaces for computer aided geometric design.* Academic Press, Boston 1999.

[3] J. Kobza: *Splajny.* UP Olomouc 1993.

[4] R. Kress: *Numerical analysis.* Springer-Verlag, New York 1998.

[5] E. Malenovský, F. Pochylý: *Some results of computational modelling of dynamic behaviour of journal bearings using the Bezier body.* Journal of Mechanical Engineering, **52**, (4), 2001, 235–259.