Bohumír Bastl

CAGD PACKAGE FOR MATHEMATICA AND ITS USAGE IN THE TEACHING

Abstract

This talk presents a new package for Wolfram's Mathematica which provides functions for finding parametrizations and rendering figures of splines, Bernstein and B-spline basis functions, Bézier and B-spline curves and surfaces together with their rational variants and also functions for basic planar and spatial transformations and surface of revolution as a NURBS surface. Such a package can be used in the teaching of geometric modelling for an interesting demonstration of some properties of Bézier and B-spline objects, e.g. the convex hull property, local modification scheme or an effect of weights of control points to the shape of a rational curve or surface etc.

Keywords

CAGD, Mathematica, NURBS curves, NURBS surfaces.

1 Introduction

Mathematica software, developed by Wolfram Research, is a powerful tool for symbolic and numeric computations. Unfortunately, it provides almost no functions concerning the Computer Aided Geometric Design (CAGD). There are only functions for cubic splines and Bézier curves, but only for visualization of these curves, it is not possible to obtain parametrizations and to work with it further. It means that there are no functions for visualization and computation (obtaining parametrizations) of modern curve and surface representations, such as rational Bézier, B-spline and NURBS curves and surfaces. That's why I have decided to write a new package for Mathematica devoted to CAGD objects to be able to work with these objects and also to demonstrate some interesting properties of these objects in the teaching of geometric modelling on our faculty.

The second reason for writing such a package followed from solving the project "Realization of interactively-information portal for scientific technical applications" which Department of Mathematics on our faculty obtained from Ministry of Education, Youth and Sports in 2004. The aim of this project is to create a new web portal, at this time located on http://webmath.zcu.cz, based on webMathematica software, which will provide different kinds of scientific computations. The computation and visualization of CAGD objects is also integrated to this web portal and anyone can use it.

2 Description of the package

After loading the package to Mathematica kernel by standard Mathematica command `<< CAGD.m`, information about functions contained in the package are printed on the screen. The package contains functions for different objects of CAGD which can be classified into following groups: splines, Bézier and rational Bézier curves and surfaces, B-spline and NURBS curves and surfaces, transformations, surface of revolution.

Now, following paragraphs are devoted to the functions contained in the package in more detail.

In connection with classical interpolating splines, the package contains functions for quadratic and cubic splines. For quadratic spline, supporting points and the boundary condition represented by tangent vector in the first point has to be specified, e.g. the command

```
QuadraticSpline[{{0,0},{-1,5},{3,-2},{5,1},{4,7},{1,-5}},{0,1},t]
```

returns the figure of the corresponding quadratic spline (see Fig. 1 (left)) and also parametrizations of all parts of the spline.

Similar function can be used for computation and visualization of a cubic spline. All possibilities of boundary conditions for cubic spline are implemented (clamped, periodic, natural and spline with given second derivatives in the first and last points) and can be specified by optional parameter of the function, e.g. by the command

```
CubicSpline[{{0,0},{-1,5},{3,-2},{5,1},{4,7},{1,-5}},t,
                 BoundaryConditions->Clamped,{{-10,-10},{-10,0}}]
```

the figure (see Fig. 1 (right)) and the parametrization of the cubic spline for given points and boundary conditions (tangent vectors in the first and last points, here) are obtained.

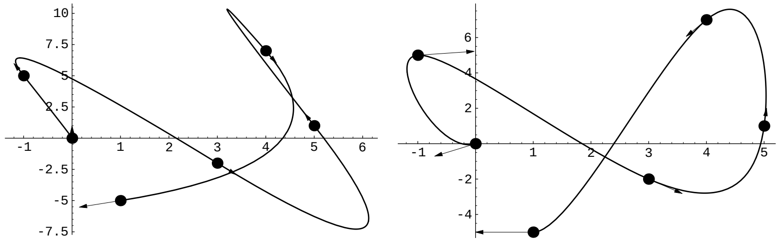

Figure 1: Quadratic and cubic splines.

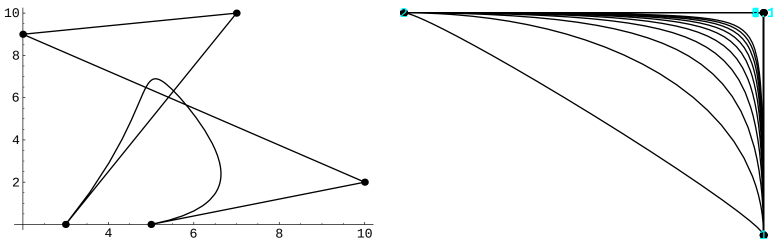

Figure 2: Bézier and rational Bézier curves.

Bézier objects belong to basic and very important CAGD objects. For given control polygon of $n+1$ control points $P_i, i = 0, \ldots, n$, or control net of $(n+1)(m+1)$ control points $P_{ij}, i = 0, \ldots, n, j = 0, \ldots, m$ respectively, the *Bézier curve*, or the *Bézier surface* respectively, is defined in the following way

$$P(t) = \sum_{i=0}^{n} P_i B_i^n(t), \quad P(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} P_{ij} B_i^n(u) B_j^m(v) \qquad (1)$$

where $B_i^n(t)$ are Bernstein polynomials of nth degree. The package provides functions `Bernstein[]` for computation of Bernstein polynomials, `BezierCurve[]` and `BezierSurface[]` for obtaining parametrizations and `PlotBezierCurve[]` and `PlotBezierSurface[]` for visualization of Bézier objects (see Fig. 2 (left) and 3 (left)).

Rational Bézier objects are important for representation of curves or surfaces which cannot be parametrized by polynomial parametrizations, only by rational, e.g. arc of a circle. To each control point P_i

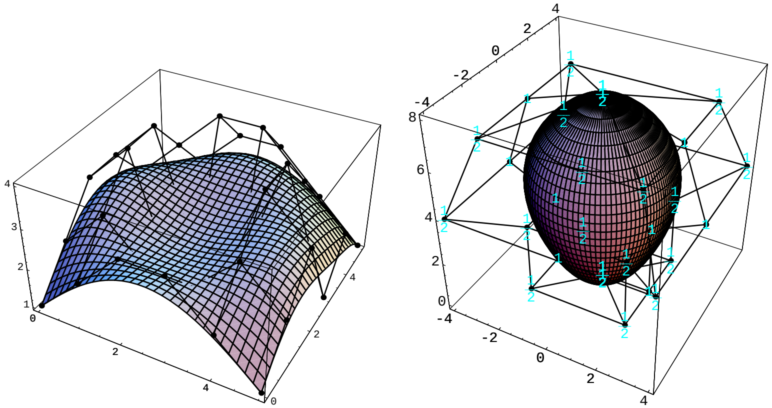

Figure 3: Bézier surface and surface of revolution.

the weight is added as an additional curve (surface) modification tool. Therefore, P_i^w are projective coordinates of control points. The projective definitions of the *rational Bézier curve* and the *rational Bézier surface* are similar to definition of polynomial Bézier objects, i.e.

$$P^w(t) = \sum_{i=0}^{n} P_i^w B_i^n(t), \quad P^w(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} P_{ij}^w B_i^n(u) B_j^m(v). \tag{2}$$

Similarly, functions `RatBezierCurve[]` and `RatBezierSurface[]` for obtaining the parametrizations and `Plot...[]` versions for visualization (see Fig. 2 (right)) are included in the package.

Important object in CAGD are B-spline objects which differ from Bézier objects by another basis functions called *B-spline basis functions* defined recursively on the knot vector $T = \{t_0, \dots, t_m\}$ by

$$\begin{aligned} N_{i,0}(t) &= \begin{cases} 1 & \text{for } t_i \le t \le t_{i+1} \\ 0 & \text{otherwise} \end{cases} \\ N_{i,p}(t) &= \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t). \end{aligned} \tag{3}$$

Using basis (3), B-spline curves and surfaces can be defined similarly as Bézier curves and surfaces in (1), only instead of Bernstein polynomials B-spline basis (3) is used. The change of basis has a lot of important consequences, e.g.

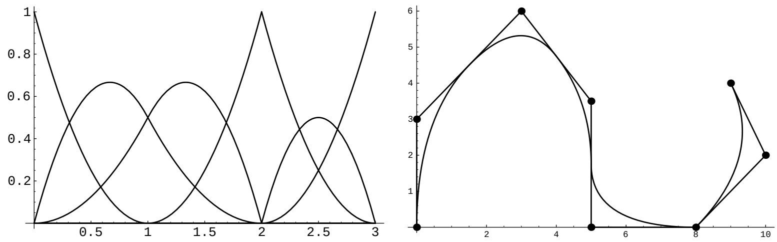

Figure 4: B-spline basis and B-spline curve.

- more possibilities while forming the curve by the help of knot vector — change of the parametrization, change of the degree and even reduction of the continuity,
- local modification scheme — change of position of one point does not affect whole curve but only some part, it can be easily seen from basis functions, e.g. for knot vector $T = \{0, 0, 0, 1, 2, 2, 3, 3, 3\}$ the basis functions are

$$\begin{aligned}&\{\{(1-t)^2, (1-t)\ t + \tfrac{(2-t)\ t}{2}, \tfrac{t^2}{2}, 0, 0, 0\},\\ &\{0, \tfrac{(2-t)^2}{2}, (2-t)\ (-1+t) + \tfrac{(2-t)\ t}{2}, (-1+t)^2, 0, 0\},\\ &\{0, 0, 0, (3-t)^2, 2\ (3-t)\ (-2+t), (-2+t)^2\}\}\end{aligned}$$

  The zero basis function means that the corresponding control point has no effect on this part of the curve.

B-spline basis for the knot vector T is shown on Fig. 4 (left). It can be easily seen from the knot vector T that a corresponding curve will be of degree 2 with three parts and with the possible reduction of continuity by 1 in connection of second and third arcs of the curve. Example of a B-spline curve for knot vector $U = \{0, 0, 0, 1, 2, 3, 4, 4, 5, 5, 5\}$ is then shown on Fig. 4 (right). In connection with B-spline objects, the package contains functions `NBasis[]` for computation of a B-spline basis, `BSplineCurve[]` and `BSplineSurface[]` for computation of parametrizations and `Plot...[]` versions for visualization.

Generalization of B-spline objects to NURBS objects is then similar to obtaining the rational Bézier objects from Bézier objects. Weights are added to control points and corresponding projective coordinates of control points are obtained. Then the definition of NURBS curves and surfaces is similar to the definition of rational Bézier surfaces (2), only Bernstein polynomials are replaced by B-spline basis functions. It can be easily shown using the functions of the package that both

$$\begin{aligned} CP_1 &= \{\{1,0,1\},\{1,1,\tfrac{1}{2}\},\{-1,1,\tfrac{1}{2}\},\{-1,0,1\},\{-1,-1,\tfrac{1}{2}\},\\ &\quad \{1,-1,\tfrac{1}{2}\},\{1,0,1\}\} \\ U_1 &= \{0,0,0,\tfrac{1}{4},\tfrac{1}{2},\tfrac{1}{2},\tfrac{3}{4},1,1,1\} \end{aligned}$$

and

$$\begin{aligned} CP_2 &= \{\{1,0,1\},\{0,1,0\},\{-1,0,1\},\{0,-1,0\},\{1,0,1\}\} \\ U_2 &= \{0,0,0,\tfrac{1}{2},\tfrac{1}{2},1,1,1\} \end{aligned}$$

are NURBS representations of the unit circle centered at the origin. The package contains `NURBSCurve[]` and `NURBSSurface[]` functions for computation of parametrizations and `Plot...[]` versions for visualization.

Finally, the package also contains functions for basic planar and spatial transformations which can be used for demonstration of affine invariance of Bézier and B-spline objects and their rational variants and also the function for computation and visualization of surface of revolution — defining curve is given as a NURBS curve by a control polygon and a knot vector and the function `RevolutionSurface[]` returns the parametrization of all parts, the `Plot...[]` version returns the figure (see Fig. 3 (right)).

3 Conclusion

The paper briefly presented a new package for Mathematica software which is devoted to CAGD. Future work on the package will include implementation of Coons surfaces and probably some other objects often used in CAGD (swung, skinned, swept surfaces).

Acknowledgements

The author has been supported by the research project 1N04078 of Ministry of Education, Youth and Sports of Czech Republic.

References

[1] L. Piegl, W. Tiller: *The NURBS Book,* Monographs in Visual Communications. Springer, Berlin, 1997