DE GRUYTER OPEN

DOI: 10.1515/ata-2016-0001

Acta Technologica Agriculturae 1
Nitra, Slovaca Universitas Agriculturae Nitriae, 2016, pp. 1–5

APPROXIMATION OF VEHICLE TRAJECTORY WITH B-SPLINE CURVE

Dušan PÁLEŠ*, Veronika VÁLIKOVÁ, Ján ANTL, František TÓTH
Slovak University of Agriculture in Nitra, Slovak Republic

In this contribution, we present the description of a B-spline curve. We deal with creation of its basis function as well as with creation of the curve itself from entered control points. Following the literature, we formed an algorithm for B-spline modelling and we used it for the planar and spatial curve. The planar curve is made of chosen points. The spatial curve approximates the trajectory of a real vehicle, which trajectory was obtained by the set of measured points. The modelled curve very exactly describes the polygon created from the linked control points. With the lowering degree of the curve, this one is more clamping to the given polygon and for the extreme case it is transformed to the polygon itself. The advantage of the B-spline curve use is, for example in comparison with a Bézier curve, high adaptability, which is expressed in its parameters – besides entered control points, these are knots generated on the curve and degree of the curve.

Keywords: B-spline curve; Cox-de Boor formula; curve fitting

The most effective way for determination of curves for practical use is to use a set of control points. These control points can be accompanied by other restriction for the curve, for example boundary conditions or conditions for curve continuity (Sederberg, 2012). When a smooth curve runs only through some control points, we refer to curve approximation. The B-spline curve is one of such approximation curves and is addressed in this contribution. A special case of the B-spline curve is the Bézier curve (Páleš and Rédl, 2014; Rédl et al., 2014).

The B-spline curve is applied to a set of control points in a space, which were obtained by measurement of real vehicle movement on a slope (Rédl, 2007, 2008). Data were processed into the resulting trajectory (Rédl, 2012; Rédl and Kučera, 2008). Except for this, the movement of the vehicle was simulated using motion equations (Rédl, 2003; Rédl and Kročko, 2007).

Material and methods

B-spline basis functions

Bézier basis functions known as Bernstein polynomials are used in a formula as a weighting function for parametric representation of the curve (Shene, 2014). B-spline basis functions are applied similarly, although they are more complicated. They have two different properties in comparison with Bézier basis functions and these are: 1) solitary curve is divided by knots, 2) basis functions are not nonzero on the whole area. Every B-spline basis function is nonzero only on several neighbouring subintervals and thereby it is changed only locally, so the change of one control point influences only the near region around it and not the whole curve.

Let $U = \{u_0, u_1, u_2, \dots , u_m\}$ be non-decreasing sequence of $(m + 1)$ real numbers, thus $u_0 \leq u_1 \leq u_2 \leq \dots \leq u_i \leq u_{i+1} \leq \dots \leq u_m$. These numbers are called knots, the set U is called the knot vector, and the half-opened interval $\langle u_i, u_{i+1})$ is the i-th knot span. Seeing that knots u_i may be equal, some knot spans may not exist, thus they are zero. If the knot u_i appears p times, hence $u_i = u_{i+1} = \dots = u_{i+p-1}$, where $p > 1$, u_i is a multiple knot of multiplicity p, written as $u_i(p)$. If u_i is only a solitary knot, it is also called a simple knot. If the knots are equally spaced, i.e. $(u_{i+1} - u_i)$ = constant, for every $0 \leq i \leq (m - 1)$, the knot vector or knot sequence is said uniform, otherwise it is non-uniform.

Knots can be considered as division points that subdivide the interval $\langle u_0, u_m\rangle$ into knot spans. All B-spline basis functions are supposed to have their domain on $\langle u_0, u_m\rangle$. We will use $u_0 = 0$ and $u_m = 1$.

To define B-spline basis functions, we need one more parameter k, which gives the degree of these basis functions. Recursive formula is defined as follows:

$$N_{i,0}(u) = 1, \quad u_1 \leq u \leq u_{i+1} \qquad (1)$$
$$N_{i,0}(u) = 0, \quad \text{for other } u$$

$$N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} N_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} N_{i+1,k-1}(u) \qquad (2)$$

This definition is usually referred to as the Cox-de Boor recursion formula. If the degree is zero, i.e. $k = 0$, these basis functions are all step functions that follows from Eq. (1). $N_{i,0}(u) = 1$ is only in the i-th knot span $\langle u_i, u_{i+1})$. For example, if we have four knots $u_0 = 0$, $u_1 = 1$, $u_2 = 2$ and $u_3 = 3$, knot spans 0, 1 and 2 are $\langle 0, 1)$, $\langle 1, 2)$ and $\langle 2, 3)$, and the basis functions of degree 0 are $N_{0,0}(u) = 1$ on interval $\langle 0, 1)$

Contact address: *Dušan Páleš, Slovak University of Agriculture in Nitra, Faculty of Engineering, Department of Machine Design, Tr. Andreja Hlinku 2, 949 76 Nitra, Slovakia, e-mail: dusan.pales@uniag.sk

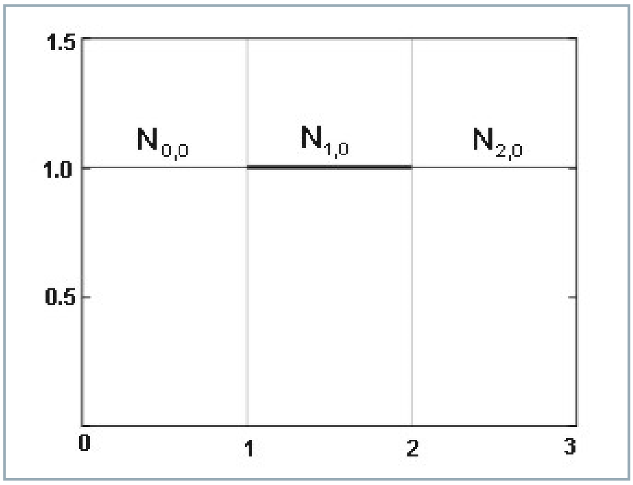

Figure 1 Basis function of zero order

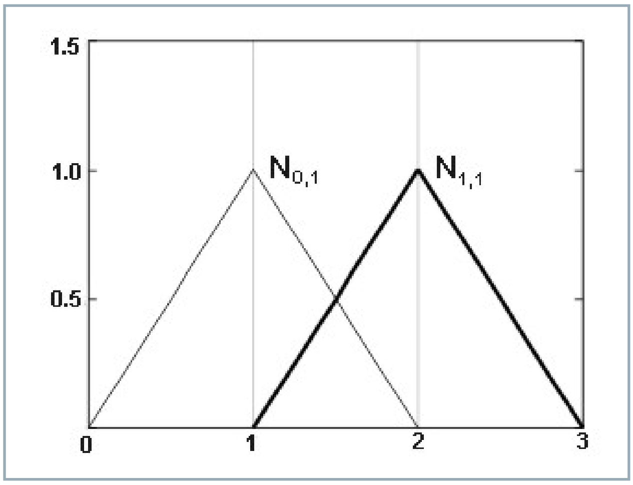

Figure 2 Basis function of first order

and $N_{0,0}$ (u) = 0 elsewhere, $N_{1,0}$ (u) = 1 on interval ⟨1, 2) and $N_{1,0}$ (u) = 0 elsewhere and $N_{2,0}$ (u) = 1 on interval ⟨2, 3) and $N_{2,0}$ (u) = 0 elsewhere, see Figure 1.

To compute $N_{i,1}$ (u), $N_{i,0}$ and $N_{i+1,0}$ are required. Therefore, we can compute $N_{0,1}$ (u), $N_{1,1}$ (u), $N_{2,1}$ (u), $N_{3,1}$ (u) and so on. Next, we compute all $N_{i,2}$ (u), and we continue like this till the computation of final $N_{i,k}$ (u).

In the above, we have obtained $N_{0,0}$ (u), $N_{1,0}$ (u) and $N_{2,0}$ (u) for the knot vector $U = \{0, 1, 2, 3\}$. Let us compute $N_{0,1}$ (u), $i = 0$, $k = 1$ from the definition:

$$N_{i,0}(u) = \frac{u - u_0}{u_1 - u_0} N_{0,0}(u) + \frac{u_2 - u}{u_2 - u_1} N_{1,0}(u) \tag{3}$$

Since $u_0 = 0$, $u_1 = 1$ and $u_2 = 2$, Eq. (3) simplifies to:

$$N_{0,1}(u) = u \cdot N_{0,0}(u) + (2 - u) \cdot N_{1,0}(u) \tag{4}$$

Because $N_{0,0}$ (u) is nonzero on interval ⟨0, 1) and $N_{1,0}$ (u) on interval ⟨1, 2), if $u \in$ ⟨0, 1), only $N_{0,0}$ (u) contributes to $N_{0,1}$ (u) and analogically, only $N_{1,0}$ (u) contributes to $N_{0,1}$ (u) if $u \in$ ⟨1, 2). Therefore, if $u \in$ ⟨0, 1), then $N_{0,1}$ $(u) = u \cdot N_{0,0}$ $(u) = u$, and if $u \in$ ⟨1, 2), then $N_{0,1}$ (u) = (2 - u) · $N_{1,0}$ (u) = (2 - u). Similar computation gives $N_{1,1}$ (u) = (u - 1) for $u \in$ ⟨1, 2) and $N_{1,1}$ (u) = (3 - u) for $u \in$ ⟨2, 3). Figure 2 shows that $N_{0,1}$ (u) is nonzero on intervals ⟨0, 1) and ⟨1, 2), and $N_{1,1}$ (u) is nonzero on interval ⟨1, 2) and ⟨2, 3).

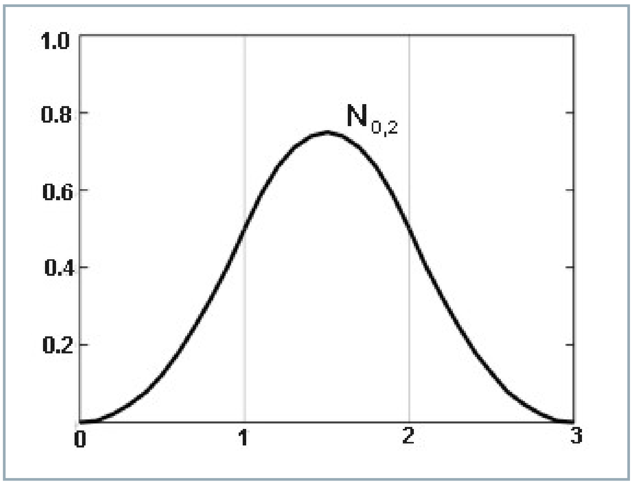

Figure 3 Basis function of second order

In general, for $k = 1$ we obtain the basis function:

$$N_{i,1}(u) = \frac{u - u_i}{u_{i+1} - u_i} N_{i,0}(u) + \frac{u_{i+2} - u}{u_{i+2} - u_{i+1}} N_{i+1,0}(u) \tag{5}$$

As:

$$N_{i,0}(u) = 1, \quad u_1 \le u \le u_{i+1} \quad N_{i+1,0}(u) = 1, \quad u_{i+1} \le u \le u_{i+2} \tag{6}$$
$$N_{i,0}(u) = 0, \quad \text{for other } u \quad N_{i+1,0}(u) = 0, \text{ for other } u$$

hence:

$$N_{i,1}(u) = \frac{u - u_i}{u_{i+1} - u_i}, \quad u_i \le u \le u_{i+1} \tag{7}$$
$$N_{i,1}(u) = \frac{u_{i+2} - u}{u_{i+2} - u_{i+1}}, \quad u_{i+1} \le u \le u_{i+2}$$

When we compute $N_{0,1}$ (u) and $N_{1,1}$ (u), we can express $N_{0,2}$ (u) as:

$$N_{0,2}(u) = \frac{u - u_0}{u_2 - u_0} N_{0,1}(u) + \frac{u_3 - u}{u_3 - u_1} N_{1,1}(u) \tag{8}$$

Similar substitution as in the formula for $N_{0,1}$ (u) for knot values gives us:

$$N_{0,2}(u) = 0.5 \cdot u \cdot N_{0,1}(u) + 0.5(3 - u) \cdot N_{1,1}(u) \tag{9}$$

$N_{0,1}$ (u) is nonzero on intervals ⟨0, 1) and ⟨1, 2), and $N_{1,1}$ (u) is nonzero on intervals ⟨1, 2) and ⟨2, 3); therefore, the basis function has to be divided into three parts, Figure 3:

First: $u \in$ ⟨0, 1), then only $N_{0,1}$ (u) contributes to the function $N_{0,2}$ (u) and hence:

$$N_{0,2} = 0.5 \cdot u^2 \tag{10}$$

Second: $u \in$ ⟨1, 2), $N_{0,1}$ (u) = (2 - u) and $N_{1,1}$ (u) = (u - 1) contribute to the function $N_{0,2}$ (u)

$$N_{0,2} = 0.5 \cdot u(2 - u) + 0.5(3 - u) \cdot (u - 1) = 0.5(-2u^2 + 6u - 3) \quad (11)$$

Third: $u \in \langle 2, 3)$, only $N_{1,1}(u) = (3 - u)$ contributes to the function $N_{0,2}(u)$

$$N_{0,2}(u) = 0.5(3 - u) \cdot (3 - u) = 0.5(3 - u)^2 \qquad (12)$$

B-spline curve

If there is $(n + 1)$ given control points $P_0, P_1, \dots P_n$ and $(m + 1)$ knots $u_0, u_1, \dots u_m$, then the B-spline curve of order k is defined with the formula:

$$C(u) = \sum_{i=0}^{n} N_{i,k}(u) \cdot P_i \qquad (13)$$

where $N_{i,k}(u)$ are the B-spline basis functions of order k explained above. The form of the B-spline curve notation is very similar to the Bézier curve notation. Unlike the Bézier curve, the B-spline curve contains more information, besides the set of $(n + 1)$ control points also the knot vector of $(m + 1)$ knots and degree of the curve k. For n, m and k must hold: $m = n + k + 1$. More precisely, if we want to define the B-spline curve of order k by help of $(n + 1)$ control points, we need to create $(n + k + 2)$ knots $u_0, u_1, \dots u_{n+k+1}$. If we have $(n + 1)$ given control points and $(m + 1)$ knots, then the degree of the B-spline curve is $k = m - n - 1$. The knot points divide the B-spline curve on the parts that are the Bézier curves of order k. While the degree of the Bézier curve depends on the number of control points and holds not to be influenced, the degree of the B-spline curve is the entering parameter for it.

If the knots do not have any ordered structure and they are only regularly placed, an opened B-spline curve is created and it does not run through the first and last point and it also does not have the first and the last flowline of control points as tangent lines. If we want to obtain the so-called clamped B-spline curve, the first and the last knot has to be $(k + 1)$ multiple. At that time, it resembles to the Bézier curve, and we have modelled this type of the clamped B-spline curve. The closed B-spline curve is the last time and it enables creating loops, the first control point of which is equal with the last one.

Results and discussion

We have formed an algorithm for computation of knot points $u_0, u_1, \dots u_{n+k+1}$ (14) according to (Meung, 2014):

```
$UP(n, k) := \text{for } i \in 0 \dots n + k + 1$
    $up_i \leftarrow 0 \text{ if } i < k + 1$
    otherwise                                   (14)
        $up_i \leftarrow n - k + 1 \text{ if } i > n$
        $up_i \leftarrow i - k \text{ otherwise}$
    up
```

Their illustration is shown in Figs 4–7. Another algorithm (15) is used for computation of the basis function of B-spline curves, which are defined by Eqs (1), (2) (Meung, 2014).

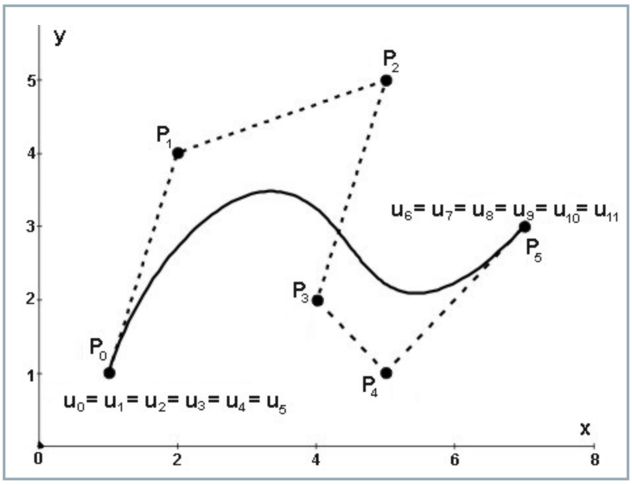

Figure 4 B-spline curve of the degree $k = 5$

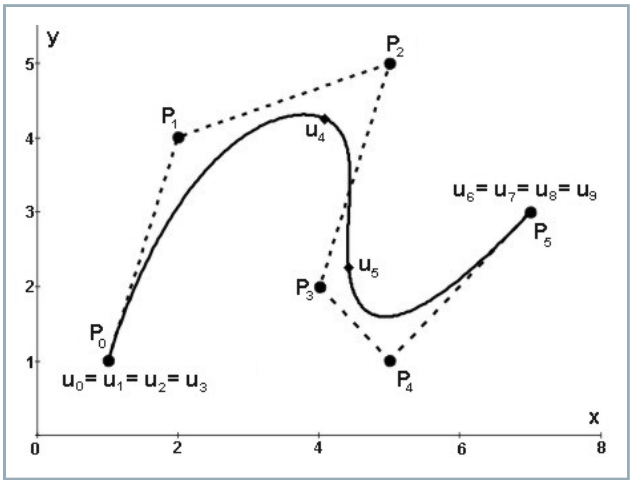

Figure 5 5 B-spline curve of the degree $k = 3$

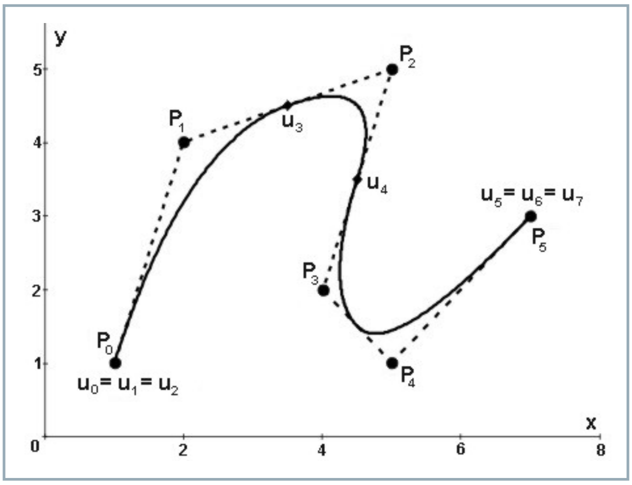

Figure 6 B-spline curve of the degree $k = 2$

The algorithms were debugged with various parameters on a test example, the results of which are shown in Figs 4–7. The control points in the example were defined by the coordinates $P_0[1, 1]$, $P_1[2, 4]$, $P_2[5, 5]$, $P_3[4, 2]$, $P_4[5, 1]$ and $P_5[7, 3]$. As can be seen in the figures, with the lowering degree k, the curve is more clamping to the polygon created

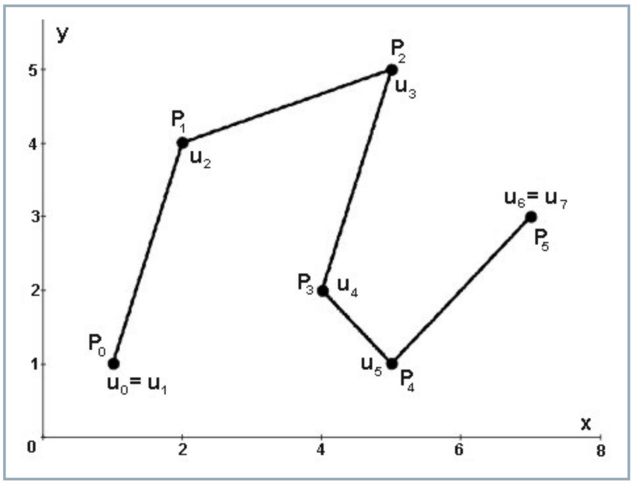

Figure 7 B-spline curve of the degree $k = 1$

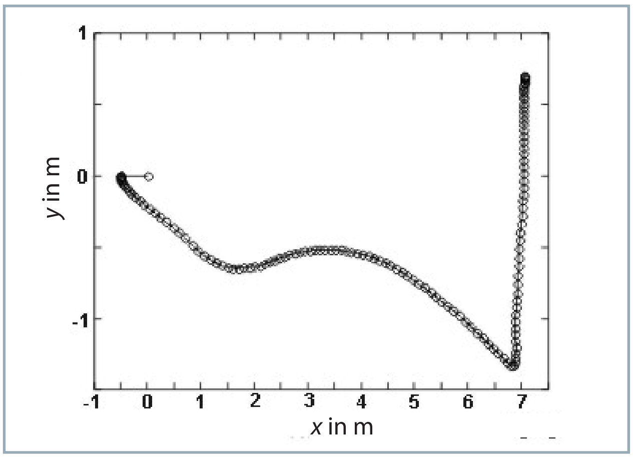

Figure 8 Comparison of measured points of vehicle trajectory with the B-spline curve in the plain *xy*

from the control points. From this follows the essential benefit of the B-spline curve in comparison with the Bézier curve, with the lowering k we can more precisely describe the given polygon, as can be seen when comparing all four figures. An extreme case occurs for the degree of the curve $k = 1$, Figure 7. In this case, the curve correctly copies the polygon of control points and exactly control points are its knots:

```
$N(i, k, n, u, up) :=$ if $k = 0$
    if $i + 1 = n + k + 1$
        if $u \le up_{i+1}$
            sum $\leftarrow$ 1 if $u \ge up_i$
            sum $\leftarrow$ 0 otherwise
        sum $\leftarrow$ 0 otherwise
    otherwise
        if $u < up_{i+1}$
            sum $\leftarrow$ 1 if $u \ge up_i$
            sum $\leftarrow$ 0 otherwise
        sum $\leftarrow$ 0 otherwise
  otherwise
    sum $\leftarrow \frac{(u - up_i)}{(up_{i+k} - up_i)} \cdot B(i, k-1, n, u, up)$ if $up_i + k > up_i$
    sum $\leftarrow$ sum + $\frac{(up_{i+k+1} - u)}{(up_{i+k} - up_{i+1})} \cdot B(i+1, k-1, n, u, up)$ if $up_{i+k+1} > up_{i+1}$
  sum
                                                                        (15)
```

Another issue to note in Figs 4–7 is that this is the clamped B-spline curve, whereby the first and the last knot are $(k + 1)$ multiple. For Figure 4, the degree is $k = 5$ and the first and the last knot are six-fold. For Figure 5, the degree is $k = 3$ and the ending knots are four-fold. Similarly for Figure 6, $k = 2$ and for Figure 7, $k = 1$.

The tested procedure in a plain was applied to a spatial curve (Rusnák et al., 2008). We solved the real vehicle movement on a slope, given by the set of points, each one with three coordinates. The number of control points is $n = 138$, whereby we chose the degree of the curve $k = 5$. The basis functions of the B-spline curve (15) were projected into all three directions *x*, *y* and *z*. In Figs 8–10, we show the projections of control points and of the B-spline curve into all three plains of the space *xy*, *yz*, *zx*. The curve very exactly copies measured control points, illustrated by empty circles, which is obvious from all three figures.

The last Figure 11, represents the spatial B-spline curve created by the trajectory of the vehicle moving on the slope. Seeing that the curve completely approximates the measured coordinates of the control points, these were not individually depicted in the figure.

Conclusion

We applied the algorithm for computation of knot points and basis B-spline functions for creation of the planar and spatial curve. The planar curve was only a test example; the spatial curve was used for the coordinates obtained in real measurement during the vehicle movement on the slope. Following the obtained results, we can state that the B-spline

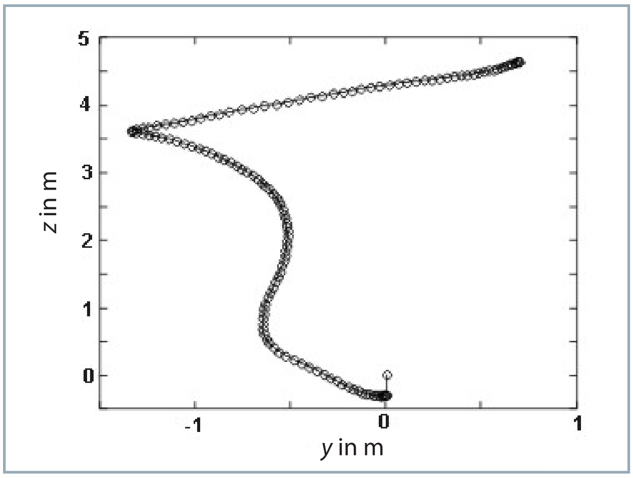

Figure 9 Comparison of measured points of vehicle trajectory with the B-spline curve in the plain *yz*

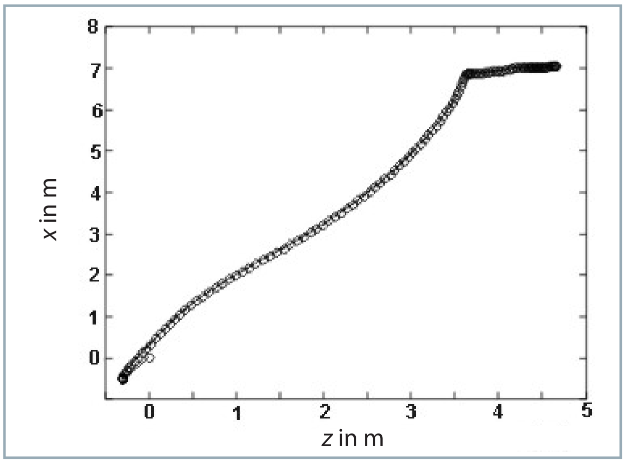

Figure 10 Comparison of measured points of vehicle trajectory with the B-spline curve in the plain *zx*

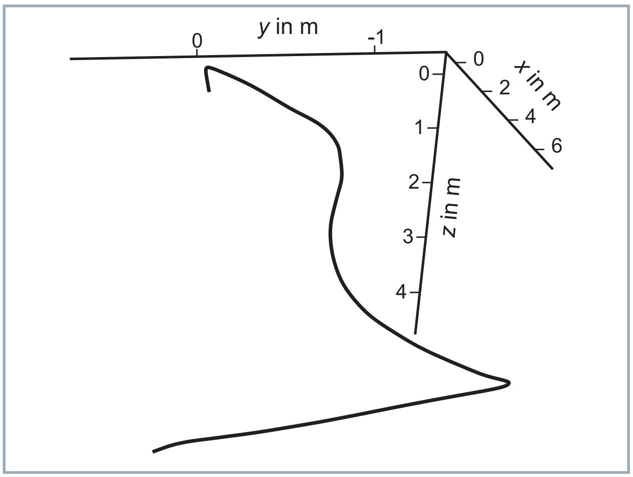

Figure 11 Trajectory of the vehicle in the space modelled through the B-spline curve

curve very exactly describes the continuance of given control points. With the lowering of the curve degree, the curve is more clamping to the polygon created from linked control points. The segments between the knots always create the Bézier curve of corresponding order. From this follows the higher accuracy of approximation with the B-spline curve in comparison with the Bézier curve. Its other advantage is high flexibility because due to its modification we can move the control points, we can change the knots on the curve and last but not least, we can change the degree of the curve. The curve here creates one unit and we do not have to worry about its continuity. The only minus occurring may be longer and more complicated algorithm for formation of basis functions (15) such as the necessity of the next algorithm (14) for knot creation.

Acknowledgment

The research performed at the Department of Machine Design, Faculty of Engineering, Slovak University of Agriculture in Nitra was supported by the Scientific Grant Agency, grant VEGA No 1/0575/14 "Minimising the risks of environmental factors in animal production buildings'.

References

MEUNG, J. KIM. 2014. Geometric Modelling. DeKalb : Department of Mechanical Engineering, Northern Illinois University. Available at: http://www.ceet.niu.edu/faculty/kim/mee430/chapter-5.pdf

PÁLEŠ, D. – RÉDL, J. 2014. Bézier Curve and its Application. Mathematics in Education, Research and Application. Nitra : Slovak University of Agriculture, Faculty of Economics and Management, Department of Mathematics, vol. 1, no. 2, pp. 49–55..

RÉDL, J. 2003. Simulation of moving agricultural aggregate. In International Student Scientific Conference, 1–2 April 2003. Nitra : Slovak University of Agriculture in Nitra, pp. 177–181.

RÉDL, J. 2007. Effect of critical angular velocities on off-road vehicle dynamic stability. In Nové Trendy v Konštruovaní a v Tvorbe Technickej Dokumentácie, 24 May 2007. Nitra : Slovak University of Agriculture in Nitra, pp. 81–87. Available at: http://www.slpk.sk/eldo/2007/024_07/redl.pdf

RÉDL, J. 2008. Effect of dynamic properties of machine on its safe operation. In Technika v Technológiách Agrosektora, 26 November 2008. Nitra : Slovak University of Agriculture in Nitra, pp. 70–78.

RÉDL, J. 2012. Software for determination of side slope stability of agricultural vehicles. In MendelTech. Brno : Mendel University in Brno.

RÉDL, J. – KROČKO, V. 2007. Mathematical model of systemic vehicle movement with supported aggregate. In Acta Facultatis Technicae, vol. 11, no. 1, pp. 135–143.

RÉDL, J. – KUČERA, M. 2008. The spatial identification of selected points of agricultural machine. In Acta Technologica Agriculturae, vol. 11, no. 3, pp. 74–78.

RÉDL, J. – PÁLEŠ, D. – MAGA, J. – KALÁCSKA, G. – VÁLIKOVÁ, V. – ANTL, J. 2014. Technical curve approximation. In Mechanical Engineering Letters, vol. 11, pp. 186–191.

RUSNÁK, J. – KADNÁR, M. – KUČERA, M. 2008. Technical curves realization in pro/engineer. In Acta Facultatis Technicae, vol. 11, no. 1, pp. 153–161.

SEDERBERG, T.W. 2012. Computer Aided Geometric Design Course Notes. Brigham Young University. 262 pp. Available at: http://cagd.cs.byu.edu/~557/text/cagd.pdf

SHENE, C.K. 2014. Introduction to Computing with Geometric Notes. Department of Computer Science, Michigan Technological University. Available at: http://www.cs.mtu.edu/~shene/COURSES/cs3621/NOTES/