

COMPARISON OF PARAMETERIZATION METHODS USED FOR B-SPLINE CURVE INTERPOLATION

Sıtkı ÖZTÜRK¹, Cengiz BALTA¹, Melih KUNCAN^{2}*

¹Kocaeli Üniversitesi, Mühendislik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü,
Kocaeli, TÜRKİYE

²Siirt Üniversitesi, Mühendislik-Mimarlık Fakültesi, Mekatronik Mühendisliği Bölümü, Siirt,
TÜRKİYE

E-mail: melihkuncan@siirt.edu.tr

Received: 1 October 2016; Accepted: 4 March 2017

Abstract: In this work we deal with the interpolation of B-spline curves to given data points. B-spline curves are generated and compared with the given data points by various parameterization methods. To perform B-spline curve interpolation on the input data, the parameterization and the node vector must be generated using the input data. For parameterization purposes, uniform, chord length, centripetal, Foley, universal and similar methods have been developed. The uniform method gives good results if the data points are regular. Chord-to-beam parameterization can produce undesirable oscillations in long chords. Therefore, the centripetal method has been developed which operates according to the square root of the chord distance. In this study, these methods were compared with different data sets.

Key words: B-spline, interpolation, parameterization.

1. Introduction

B-spline techniques for curves and surfaces were proposed in the early 1940s and were developed by Rich Riesenfeld in the 1970s. The B-spline structure is a generalized version of the Bezier curves developed by Pierre Bezier. The letter "B" refers to the word "basis" [1].

B-spline curves have been developed to address the following disadvantages in Bezier curves

- The grade of the Bezier curve depends on the number of control points.
- Bezier curves only provide global control, not local control.
- While Bezier is easy to maintain C^1 continuity of curve segments, it is insufficient to maintain continuity C^2 .

B-spline curves provide local control. That is, the shifted control points change the neighborhoods affected by the control point only. In addition, B-spline curves can provide continuity of the curved parts at the desired level [1]. For example, when two curve segments C^2 are combined with continuity, it is desirable that the first and second derivatives of the merging points of the curves are equal. If C^n continuity is desired at the junction of two curves, both curves must be polynomials at n . degree.

B-spline curves are defined by control points and are used to approximate a targeted set of points or surfaces. Also, a parameter space, which is expressed as a node vector that is defined independently of the length of the points, is needed. The node vector may be uniformly uniform, or it may be defined as a nonuniform distribution with a variety of methods to approximate the target points.

2. Recursive Computation of B-Spline Base Functions

Computer-based B-spline curve computations use the Cox-de Boor algorithm, which recursively computes with respect to the input of the node vector. A sample B-spline curve at the p -th order is defined as follows:

$$C_{(u)} = \sum_{i=0}^n N_{i,p}(u)P_i \quad (1)$$

In this definition, P_i denotes control points, whereas $N_{i,p}(u)$ denotes p -th degree B-spline basis functions. The U vector consists of continuously increasing real numbers that operate independently of the object space and are called the node vector.

Each u_i element of the U vector is called a node. Under these definitions, the i -th base function of B-spline at the p -th grade is expressed in a computerized manner with the Cox-de Boor algorithm as follows:

$$N_{i,0}(u) = \begin{cases} 1, & u_i \leq u \leq u_{i+1} \\ 0, & \text{diğer} \end{cases} \quad (2)$$

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

3. B-spline curve interpolation

Bezier and B-spline curves work in computer graphics, animation applications and CAD-CAM surface modeling stages, according to the control points and node vector data from the designer. However, in reverse engineering applications and for data interpolation purposes, this process must be done in a inverse. In this case, it is desirable to generate the control points and the necessary node vector, while providing point data to enter the curve or the surface. At this stage, the operations are reversed. In Fig. 1, the node vector space and the independent state of the object space are given.

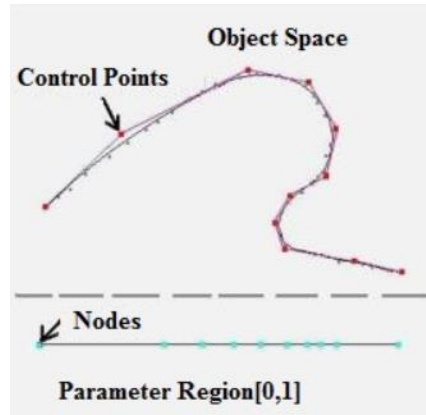


Fig. 1. Parameter space and independence of object space

The equation for the B-spline curve is given as:

$$C_{(u)} = \sum_{i=0}^n N_{i,p}(u)P_i \quad (3)$$

In terms of parameterization, the value of the point to be calculated for each parameter value on the node vector is expressed as follows:

$$D_k = C(t_k) = \sum_{i=0}^n N_{i,p}(t_k)P_i, \quad 0 \leq k \leq n \quad (4)$$

The values of the base functions in the parameter values are written in the matrix N :

$$N = \begin{bmatrix} N_{0,p}(t_0) & N_{1,p}(t_0) & N_{2,p}(t_0) & \cdots & N_{n,p}(t_0) \\ N_{0,p}(t_1) & N_{1,p}(t_1) & N_{2,p}(t_1) & \cdots & N_{n,p}(t_1) \\ \vdots & & & \ddots & \vdots \\ N_{0,p}(t_n) & N_{1,p}(t_n) & N_{2,p}(t_n) & \cdots & N_{n,p}(t_n) \end{bmatrix} \quad (5)$$

The computed D destination points and the P control points are expressed in matrix form as follows:

$$D = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_n \end{bmatrix}, \quad P = \begin{bmatrix} p_0 \\ p_1 \\ \vdots \\ p_n \end{bmatrix} \quad (6)$$

While the control points are given, in the forward direction calculation, the B-spline curve equation is expressed in matrix form as follows:

$$D = N.P \quad (7)$$

For curve interpolation operations where the control points are to be calculated, the following steps are taken to obtain the P control points:

$$D = N.P \quad (8)$$

$$N^T D = N^T N P$$

$$P = (N^T N)^{-1} (N^T D)$$

4. Data Parameterization Methods

In B-spline interpolation and approximation problems, a set of points is given as input. If it is assumed that the node vector changes at $[0,1]$, then some of the cut-off parameters in this range are required for the stoppages to correspond to the data points in the input set. For D_0, \dots, D_n data points, the unit parameter value is defined in the t_0, \dots, t_n region. If $C(u)$ is defined as a curve passing through all the data points, $D_k = C(t_k)$ is for the k values, $0 \leq k \leq n$ at the cut-off points. Fig. 2 shows how parameters are associated with data points.

Selection of parameter values includes uncertainty and there are infinitely many alternatives. However, careless selection of parameter values can cause undesirable shapes and fluctuations to be generated.

During the parameterization phase, various methods called uniform, chord length, centripetal, universal are used.

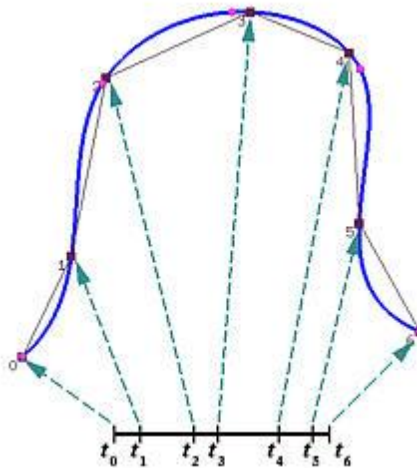


Fig. 2. Parameters and corresponding data points [8]

4.1. Uniformly distributed parameterization

Uniformly distributed parameterization is the simplest method for assigning parameters to given input data points. If it is assumed that parameters are to be distributed in the range $[0,1]$, and if this range of pieces of input is considered to be divisible, then the parameters are defined as follows:

$$t_0 = 0,$$

$$t_i = \frac{i}{n}, \quad 1 \leq i \leq n-1, \quad (9)$$

$$t_n = 1$$

4.2. Parameterization according to chord length

If the distances between input data points are irregularly distributed, the chord-length method works better than uniform parameterization.

Given D_0, D_1, \dots, D_n data points. The distance between point D_{i-1} and point D_i is expressed by $|D_i - D_{i-1}|$. In this case, the sum of the beam distances between all data points is expressed as:

$$L = \sum_{i=1}^n |D_i - D_{i-1}| \quad (10)$$

In this case, the chord length ratio from point D_0 to point D_k is expressed as follows:

$$L_k = \frac{\sum_{i=1}^k |D_i - D_{i-1}|}{L} \quad (11)$$

In the normalized case, assuming that parameterization has been performed in the range [0,1], the distribution is as follows.

$$t_0 = 0, \quad t_k = \frac{1}{L} \sum_{i=1}^k |D_i - D_{i-1}|, \quad t_n = 1 \quad (12)$$

4.3. Centripetal method

Lee proposed a centripetal parameterization. In the case of a car driving on a racetrack, it is desirable that the centrifugal force (or normal force) be very high in sharp turns when driving on a piste. For a safe ride, Lee suggests that along the way the centrifugal force must be proportional to the change in angle. The centripetal method presents this model approach. This model can be seen as an improved version of the chord length method [2].

D_0, D_1, \dots, D_n data points are given. If the base factor is defined as $e = 1/2$, the distance between point D_{i-1} and point D_i is denoted by $|D_i - D_{i-1}|^e$. Kiriş- In the chord method, this expression was used topless. In this case, the length of the entire data polygon is expressed in terms of the centripetal metric as follows:

$$L = \sum_{i=1}^n |D_i - D_{i-1}|^e \quad (13)$$

The ratio of the data beam length from point D_0 to point D_k to the total beam length is expressed as follows.

$$L_k = \frac{\sum_{i=1}^k |D_i - D_{i-1}|^e}{L} \quad (14)$$

In the normalized case, assuming that parameterization is performed in the range [0,1], the values are distributed in the following manner.

$$\begin{aligned}
 t_0 &= 0 \\
 t_k &= \frac{1}{L} \sum_{i=1}^k |D_i - D_{i-1}|^e \\
 t_n &= 1
 \end{aligned} \tag{15}$$

4.4. Universal method

Lim proposed a method that differs from other methods in 1999. In the previous methods, the new proposed method is based on calculating the parameter values from the uniformly distributed clamped node vector, while the node vector is generated from these parameter values after the parameter values are determined [3].

In the Lim method, base functions are generated, starting from a uniformly distributed node vector. The peak values of these base functions are considered as input to the parameter list. Although this method produces more natural-looking curves than previous methods, it produces undesirable oscillations in some cases [3].

Suppose that we wanted to generate the necessary parameter values for $n+1$ data points using B-spline at p . $m = n + p + 1$, the number of elements in the node vector is $m+1$. In this case a uniformly distributed, clamped-clamped node vector is obtained as:

$$u_0 = u_1 = \dots u_p = 0, \quad u_{p+i} = \frac{i}{n-p+1} \quad i = 1, 2, \dots, n-p, \quad u_{m-p} = u_{m-p+1} = \dots u_m = 1 \tag{16}$$

4.5. Foley-Nielson method

This method was proposed by Foley and Nielson. While the previous methods use the beam spacing between data points, the Foley-Nielson method adds an account between the adjacent points. This method uses the Nielson scale to measure the distance between points. On this count, geometry does not change when it enters rotation, translation and scaling operations. The transformed B-spline curve can be obtained from transformed data points [4].

$m = 2$, when working with two-dimensional data points, the coefficient matrix $Q = \{q_{ij}\}$, $i, j = 1, 2$ is defined as:

$$\begin{aligned}
 q_{11} &= \frac{V_y}{g}, \quad q_{22} = \frac{V_x}{g}, \quad q_{12} = q_{21} = -\frac{V_{xy}}{g}, \quad g = V_x V_y - (V_{xy})^2 \\
 V_x &= \frac{\sum_{i=0}^n (x_i - \bar{x})^2}{n+1}, \\
 V_y &= \frac{\sum_{i=0}^n (y_i - \bar{y})^2}{n+1}, \\
 V_{xy} &= \frac{\sum_{i=0}^n (x_i - \bar{x})(y_i - \bar{y})}{n+1}
 \end{aligned} \tag{17}$$

$$\bar{x} = \frac{\sum_{i=0}^n x_i}{n+1}, \quad \bar{y} = \frac{\sum_{i=0}^n y_i}{n+1}$$

In this case, the Neilson distance between points U and V is defined as:

$$M[P](U, V) = \sqrt{(U - V)Q(U - V)^T} \quad (18)$$

According to the computed Neilson distance, if chord distance parameterization is performed, affine-invariant parameterization can be obtained which can be applied to transformations. In the Foley parameterization, the Δt_i parameter step is defined as:

$$\Delta t_0 = d_0 \left[1 + \frac{3\hat{\theta}_1 d_1}{2(d_0 + d_1)} \right], \quad \Delta t_{n-1} = d_{n-1} \left[1 + \frac{3\hat{\theta}_{n-1} d_{n-2}}{2(d_{n-2} + d_{n-1})} \right]$$

$$\Delta t_i = d_i \left[1 + \frac{3\hat{\theta}_i d_{i-1}}{2(d_{i-1} + d_i)} + \frac{3\hat{\theta}_{i+1} d_{i+1}}{2(d_i + d_{i+1})} \right], \quad i = 1, 2, \dots, n-2 \quad (19)$$

$$d_i = M[P](P_i, P_{i+1}), \quad \hat{\theta}_i = \min(\theta_i, \pi/2)$$

$$\theta_i = \pi - \arccos \left[\frac{d_{i-1}^2 + d_i^2 + M^2[P](P_{i-1}, P_{i+1})}{2d_i d_{i-1}} \right]$$

Here, θ_i is also an affine invariant angle, which is calculated by Neilson distance and is not affected when it enters transformations. In Fig. 3, the notations used in the Foley method are given..

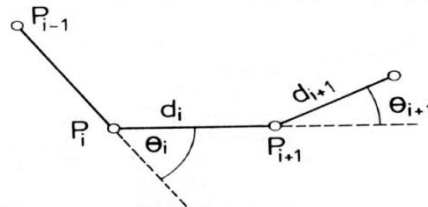


Fig. 3. Parameters used in the Foley method [5]

4.6. Node vector production

After the parameter values are generated, the node vector is generated from these values. Assuming we have $n+1$ parameter values defined in t_0, t_1, \dots, t_n and we use p th degree B-spline fragments, we need $m+1$ node values, defined as $m=n+p+1$ shaped. If the curve is defined as clamped, the first $p+1$ node value is 0. And the last $p+1$ knot value becomes 1. The $n-p$ nodes in the middle are uniformly distributed according to various methods.

If uniform parameterization is used, the median terms are divided into $n-p+1$ pieces.

$$u_0 = u_1 = \dots u_p = 0$$

$$u_{j+p} = \frac{j}{n-p+1}, \quad j = 1, 2, \dots, n-p \quad (20)$$

$$u_{m-p} = u_{m-p+1} = \dots u_m = 1$$

5. Applications

The first dataset interpolated in Fig. 4 is taken from the work of Irvine et al. This data can be problematic for interpolation methods due to sudden directional changes and sparse data points [6].

Table 1. The maximum deviation table from the beam along the curve piece, 1. Dataset

Method	1	2	3	4	5	6	7	Average	RMS	Max. Deviation	Time (ms)
Uniform	0.1918	0.0889	0.4133	0.1647	0.4564	0.1964	0.2765	0.255	0.2843	0.4564	49,860
Chord	1.7113	0.5507	0.0524	2.877	0.3785	0.061	2.304	1.133	1.5569	2.877	50,739
Universal	0.102	0.047	0.4007	0.1642	0.3667	0.2513	0.1360	0.2097	0.2439	0.4007	135,781
Foley	0.0988	0.1354	0.2475	0.728	0.0976	0.0909	0.5904	0.2840	0.3753	0.728	77,506
Centripetal	0.2135	0.1952	0.1777	0.7214	0.0832	0.0906	0.6385	0.3028	0.3889	0.7214	49,789

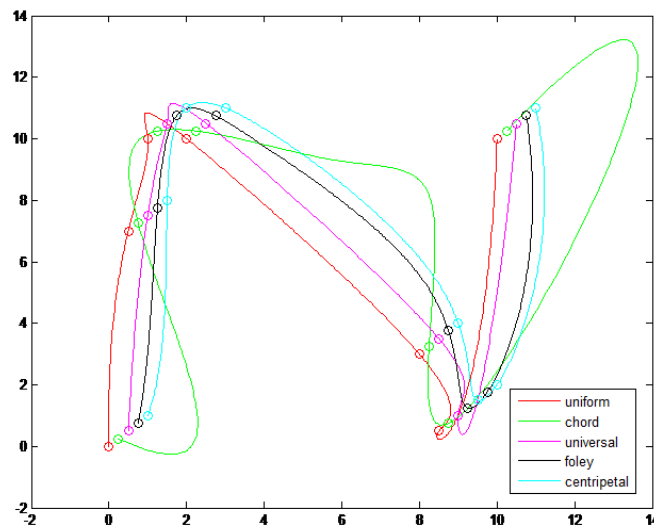


Fig. 4. Cubic Curves Produced by Various Parameterization Methods, 1. Dataset

Fig. 5 shows the cubic B-spline curves generated by the various methods of parameterization for the second dataset. The second data set I in Fig. 5 was taken without the same operation as the first data set. There are also sudden changes in direction and rare data difficulty similar to Larry Irvine et al [6].

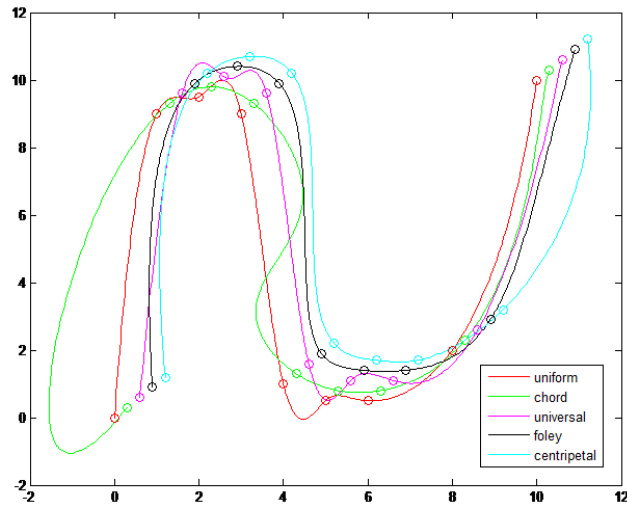


Fig. 5. Cubic Curves Produced by Various Parameterization Methods, 2. Dataset

Table 2. The maximum deviation table from the beam along the curve piece, 2. Dataset

Method	1	2	3	4	5	6	7	8	Average	RMS	Max. Deviation	Time (ms)
Uniform	0.209	0.259	0.473	0.049	0.482	0.152	0.35	0.376	0.2938	0.3271	0.4733	50,461
Chord	1.847	0.118	0.106	0.841	0.067	0.053	0.259	0.431	0.4652	0.7420	1.847	49,805
Universal	0.18	0.442	0.367	0.054	0.476	0.193	0.519	0.256	0.3109	0.3471	0.5187	133,821
Foley	0.59	0.12	0.161	0.269	0.141	0.031	0.345	0.199	0.2319	0.2829	0.5895	76,887
Centripetal	0.667	0.109	0.177	0.231	0.134	0.051	0.191	0.802	0.2952	0.3940	0.8024	52,574

The third set of data in Fig. 6 is taken from the work of Fritsch and Carlson. Sudden distance changes are tested here [7].

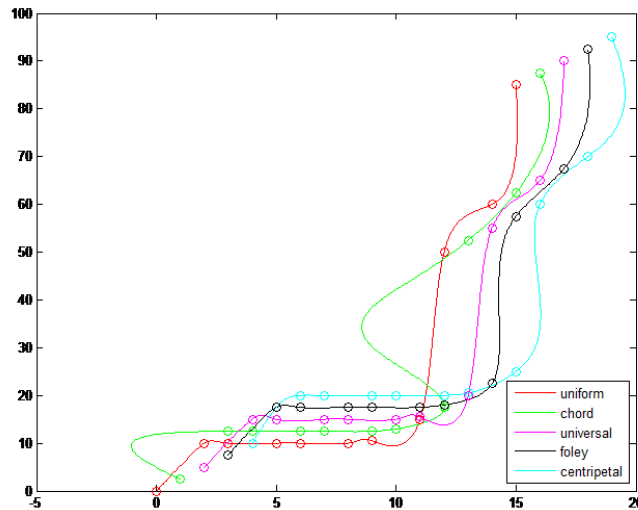


Fig. 6. Cubic Curves Produced by Various Parameterization Methods, 3. Dataset

Table 3. The maximum deviation table from the beam along the curve piece, 3. Dataset

Method	1	2	3	4	5	6	7	8	9	10	Average	RMS	Max. Deviation	Time (ms)
Uniform	0.123	0.359	0.142	0.078	0.185	0.337	1.168	0.19	0.652	0.651	0.3885	0.506	1.168	46,438
Chord	3.347	0.057	0.011	0.008	0.082	0.053	0.583	3.915	0.238	0.806	0.91	1.661	3.915	49,580
Universal	0.169	0.447	0.366	0.162	0.234	0.361	1.246	0.212	0.324	0.469	0.399	0.499	1.246	135,65
Foley	0.164	0.354	0.111	0.131	0.079	0.057	0.516	0.412	0.13	0.597	0.255	0.316	0.597	75,379
Centripetal	0.505	0.045	0.027	0.001	0.04	0.105	0.268	0.741	0.305	1.096	0.3132	0.467	1.096	49,11

The fourth data set in Fig. 7 was taken from the work of Boeing employee ETY Lee [2]. Here, there are more data points near the corners.

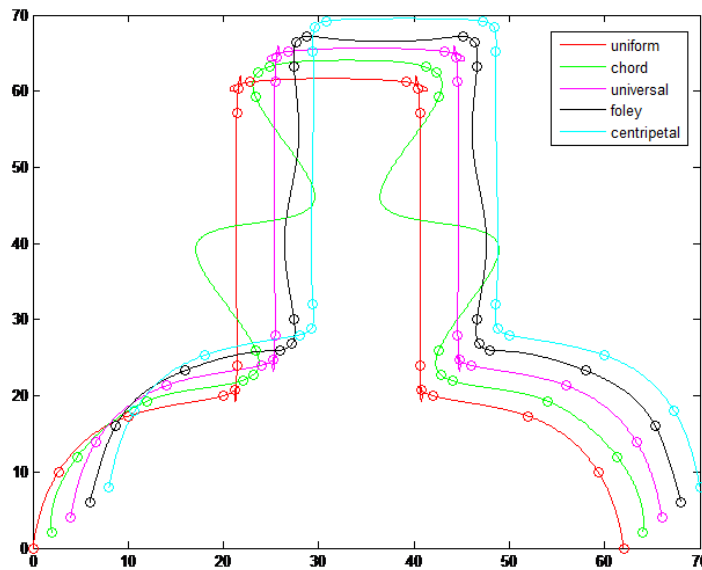


Fig. 7. Cubic Curves Produced by Various Parameterization Methods, 4. Dataset

Table 4. Maximum deviations table from the beam along the curve piece, 4. Dataset

Method	Uniform	Chord	Universal	Foley	Centripetal
1	0.4593	0.8607	0.4602	0.2704	0.3944
2	0.9257	0.6698	0.9127	1.011	0.9574
3	0.225	0.2848	0.2747	0.5958	0.2614
4	0.109	0.8897	0.1558	0.1798	0.1449
5	0.1584	0.4455	0.1461	0.1909	0.1756
6	0.1879	5.224	0.2024	0.999	0.2223
7	0.0998	0.4396	0.1009	0.1437	0.1264
8	0.3779	0.084	0.3907	0.2422	0.1928
9	0.4538	0.8721	0.4598	0.6513	0.3511
10	0.3885	0.08393	0.3903	0.2422	0.1928
11	0.06689	0.4396	0.092	0.1441	0.1278
12	0.1419	5.224	0.2029	0.9916	0.2105
13	0.1518	0.4433	0.1539	0.1937	0.1756
14	0.1074	0.07757	0.1617	0.1655	0.1449
15	0.2171	0.2919	0.2754	0.5958	0.2614
16	0.9257	0.6717	0.9088	1.011	0.9574
17	0.457	0.8607	0.4524	0.2782	0.3944
Average	0.2830	1.0508	0.3377	0.4651	0.2718
RMS	0.4111	1.8705	0.4148	0.5741	0.3991
Max. Deviation.	0.9257	5.224	0.9127	1.011	0.9574
Time (ms)	44,255	43,387	127,132	69,769	43,429

6. Conclusion

In order to be able to interpolate input data with B-spline curves, it is necessary to parameterize the data in the first step. Various methods are used to parameterize the input data. Commonly used methods are uniform uniform parameterization, chord length, universal parameterization, Foley angle parameterization centripetal methods.

The uniform method provides good results if the data points are regular. According to the chord distance, parameterization can produce unwanted oscillations in long beams. Therefore, the centripetal method has been developed which operates according to the square root of the beam spacing. The Foley method, on the other hand, has been successful in providing sharp rotations due to the addition of angle values.

In this study, basic parametrization methods are tested for different data sets and the obtained curves and error measurements are given in Tables 1, 2, 3 and 4 with Figures 4, 5, 6 and 7. From these data sets, it was observed that for the first three data sets, the centripetal-centric method succeeded from the other methods and the performance approaches the Foley angle method. For the fourth data set containing intense data at the corners, the centripetal method was observed to be more successful.

In case of sudden change of the beam length between data points, the centripetal method that works according to the square root of the beam distance is preferred; in the datasets where there are sudden and sharp angular rotations, the Foley method, which adds the angle values, is preferred; In cases where the data are uniformly distributed, the chord-beam distance method would be preferable. Where surface details are insignificant and speed is important, it is seen that uniform methods can be preferred in terms of simplicity in calculations.

Acknowledgment

This project was supported by the Coordinator of BAP Coordination Unit of Scientific Research Projects of Kocaeli University as the project number 2015-018.

References

- [1] Solomon, D. *Curves and Surfaces for Computer Graphics*, Springer, (2006),
- [2] Lee, E. Choosing Nodes in Parametric Curve Interpolation, *Computer Aided Design*, vol. 21, no. 6, (1989), sayfa 363-370.
- [3] Lim, C. G. A Universal Parameterization in B-spline Curve and Surface Interpolation, *Computer Aided Geometric Design* 16, (1999), sayfa 407-422.
- [4] Foley, T.A., & Nielson, G.M. Knot selection for parametric spline interpolation. Mathematical methods in computer aided geometric design. *Academic Press Professional, Inc*, (1989), sayfa 261–72.
- [5] Hoschek, J. & Lasser, D. *Computer Aided Geometric Design*, A K Peter, Ltd. (1993)
- [6] Irvine, L., Marin, S., & Smith, P. Constrained Interpolation and Smoothing, *Constructive Approximation*, Volume 2, Number 1, (1986), sayfa 129-151.



- [7] Fritsch, F., & Carlson R. Monotone Piecewise Cubic Interpolation, *SIAM Journal on Numerical Analysis*, Volume 17, Number 2, (1980), sayfa 238-246.
- [8] Piegl, L., & Tiller, W. The NURBS Book, *Springer*. (1997),