



## ANALYSIS OF B-SPLINE CURVE USING DISCRETE FOURIER TRANSFORM

Ashok Ganguly<sup>1</sup>, Pranjali Arondekar<sup>2\*</sup>

<sup>1</sup>Shri G.S.Institute of Science& Technology 23,Park Road,Indore, India.

<sup>2</sup>Medi Caps Institute of Technology & Management, Indore, India.

[parondekar@gmail.com](mailto:parondekar@gmail.com)

**Abstract-** We apply the Discrete Fourier Transform to the construction of B-Spline curves to gain more insight into their structure. As a B-Spline curve is determined by its control polygon, this analysis is intimately linked to the Fourier analysis of the control polygon. To do this we apply Fast Fourier transform (FFT) algorithm to the structure of B-Spline curve and its rational form. We get inner structure of original B-Spline curve in the transform domain again in the form of B-Spline curve, having control polygon as regular or star polygon. Using the technique mentioned in the paper we get the same curve without change of shape in the transformed case of polygon points. We also extend the idea for the interval form of B-Spline Curve.

**Key Words-** B-Spline, Base B-Spline, Fourier Transform.

### 1. INTRODUCTION

This paper is concerned with analysis of B-Spline curve using Fast Fourier Transform. B-Spline curve is one of the most important curves in Computer aided geometric design (CAGD) used for curve representation.

It is often useful to decompose a given function into components, analyze them, and then reassemble the function again, possibly in a different way. One classical and mathematically very interesting method is to use trigonometric functions. This is the basis for the theory of Fourier analysis.

Our main aim in this paper is to analyze the B-Spline curve to study the inner structure of curve using Fourier Transform. As a B-Spline curve is determined by its control polygon, by transforming the points of control polygon, we transform the B-Spline curve from one domain to another using Discrete Fourier Transform (DFT) through algorithm Fast Fourier Transform (FFT)[1,2,5].

The paper is organized in following sections; we made a basic review in section 2. Then in the next section 3 we analyze the B-Spline curve and its rational form using DFT. We get the inner structure of B-Spline curve, in the form of another B-Spline curve known as base B-Spline curve in the transformed form. We show that the obtained base B-Spline curve have very nice properties with control polygon points as powers of twiddle factor of DFT[3]. Next in section 4, we illustrate results by taking example curves for various cases of B-Spline Curve. The case for, open B-Spline curve having order equal to number of polygon points gives the inner structure of Bézier curve as mentioned by Paul Barry [3]. In fact our paper generalizes the work of [3]. In the last section we also extend the idea for the interval form of B-Spline curve where the control polygon is defined by rectangular or circular interval to capture the uncertainty characteristic of design.

## 2. BASIC REVIEW

### 2.1. B-Spline Curve and Rational B-Spline Form

A B-Spline curve is given by[4]

$$P(t) = \sum_{i=1}^{n+1} N_{i,k}(t)P_i \quad (1)$$

Where,  $P_i$ 's are the position vectors of the  $n+1$  defining polygon vertices and  $N_{i,k}(t)$  is the B-Spline basis functions, given by Cox de-Boor recursive definition, defined for open, uniform and nonuniform knots vector over useful parameter range i.e. over range

for which  $\sum_{i=1}^{n+1} N_{i,k}(t)=1$ .

The rational form of B-Spline curve is given by,

$$R_{n,k}(t) = \frac{\sum_{i=1}^{n+1} P_i h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} \quad (2)$$

where,  $h_i$ 's are the weight assigned to control polygon points  $p_i$ 's respectively for  $i=1, \dots, n+1$ .

### 2.2 Discrete Fourier Transform and FFT

Fourier transform arises from Fourier series by taking more and more frequencies into account. Discrete Fourier transform (DFT) is the attempt to estimate Fourier transform of a function from finite number of sampled point.

The DFT is simply a mapping of one ordered set of  $N$  complex number to a different ordered set. Consider a complex series  $x(k)$  with  $N$  samples of the form  $x_0, x_1, \dots, x_{N-1}$  Where  $x$  is a complex number, with assumption that that the series outside the range  $0, N-1$  is extended  $N$ -periodic, that is,  $x_L = x_{L+N}$  for all  $L$ . The DFT of this series will be denoted by  $X(L)$ . It will also have  $N$  samples. The forward transform will be defined as [1,2]

$$X(L) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi L n}{N}} \quad \text{For } L = 0, \dots, N-1$$

$$\text{Or } X(L) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) w^{Ln} \quad \text{For } L = 0, \dots, N-1 \quad (3)$$

where  $w = e^{-j \frac{2\pi}{N}}$ , is known as twiddle factor.

The DFT series is usually evaluated in practical applications using a Fast Fourier Transformation (FFT), algorithm [5].

We have made programs for transformation of B-Spline using FFT in C++, to analyze the inner structure.

### 3. FOURIER ANALYSIS OF B-SPLINE CURVE

#### 3.1 Fourier Transform of B-Spline Curve

The B-Spline curve of  $n+1$  polygon vertices and order  $k$  is given by definition (1). The polygon points  $P_i$  can be considered as points on complex plane. Thus we can consider  $P_i$  as the inverse transform of points  $\bar{P}_i$ , where  $\bar{P}_i$  is the Fourier transform of points  $P_i$  i.e.,

$$P_i = \sum_{L=0}^n \bar{P}_L w^{-Li} \quad (4)$$

where,  $w = e^{-\frac{j2\pi}{N}}$  is twiddle factor, with  $N=n+1$ .

Thus equation (1) can be written as

$$P(t) = \sum_{i=1}^{n+1} N_{i,k}(t) \sum_{L=0}^n \bar{P}_L w^{-Li}$$

where,  $\bar{P}_L = \frac{1}{n+1} \sum_{i=0}^n P_i w^{Li}$ , written according to definition (3).

As  $\bar{P}_L$ 's are independent of  $i$ , we get

$$P(t) = \sum_{L=0}^n \bar{P}_L \sum_{i=1}^{n+1} N_{i,k}(t) w^{-Li} \quad (5)$$

Finally, we write

$$P(t) = \sum_{L=0}^n \bar{P}_L BB_{L,k}(t) \quad (6)$$

where,  $BB_{L,k}(t) = \sum_{i=1}^{n+1} N_{i,k}(t) w^{-Li}$  for  $L=0, \dots, n$ . (7)

are consider as "Base B-Spline Curve", for the given B-Spline curve of  $n+1$  polygon vertices and order  $k$ . Whose properties we discuss in the next section.

#### 3.2 Properties of Base B-Spline Curve

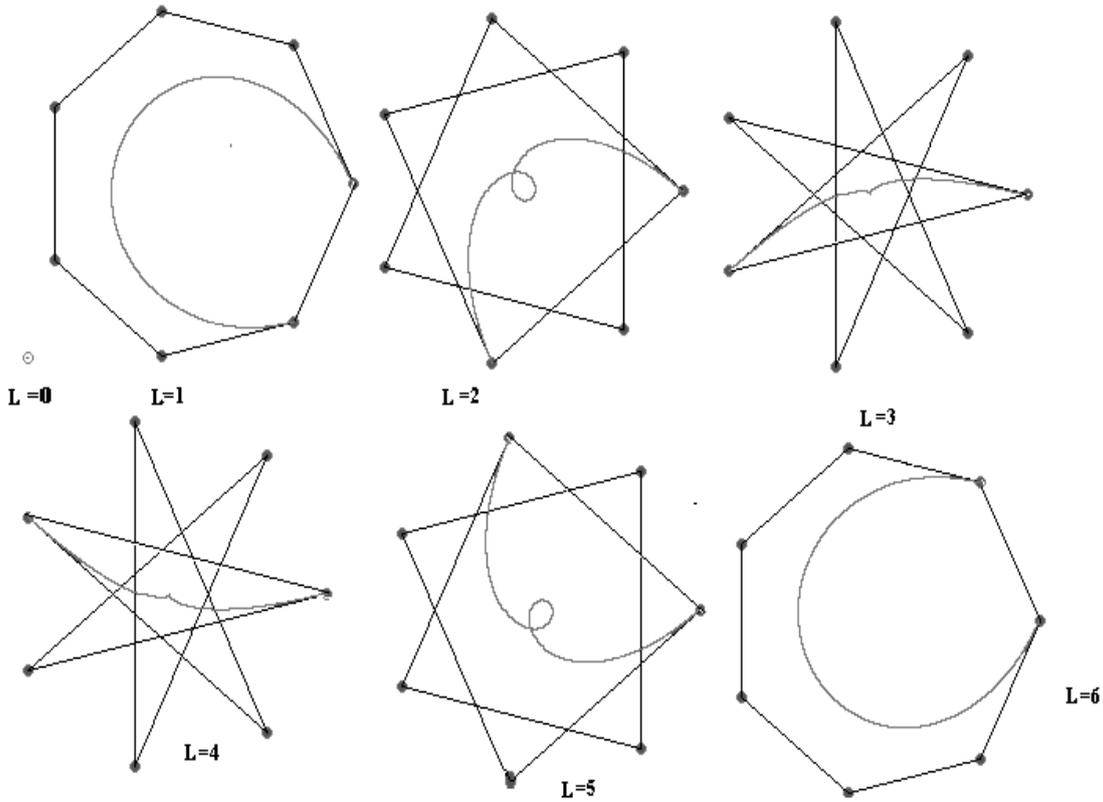
From equation (7), we get, the Base B-Spline curve is the B-Spline curve generated by B-Spline Basis  $N_{i,k}(t)$ , for  $t$  in useful parameter range, and defined over specified type of knot vectors.

Each time they have similar shape for each type of knot vectors.

The control polygon points for the base curves is given by powers of twiddle factor  $w^{-L}$  i.e.  $\{w^{-L}, w^{-2L}, w^{-3L}, \dots, w^{-(n+1)L}\}$ .

The obtained control polygons for Base B-Spline curves are either regular or star polygon and base B-Spline curves are in the convex hull of these regular or star polygons, as seen in figure 1.

Now as even multiple of  $\pi$  in exponential is 1, we have  $W^{((n+1)-L)i} = W^{-Li}$ . Thus obtained Base B-Spline curve  $BB_{L,K}(t)$  and  $BB_{(n+1-L),K}(t)$ , corresponding to order  $k$ , are mirror image of each other about x-axis. It can be easily proved that  $\{BB_{1,k}(t), BB_{2,k}(t), BB_{3,k}(t), \dots, BB_{n+1,k}(t)\}$  behave as basis for B-Spline curve of order  $k$ , in transform form of polygon points. Thus original B-spline curve of order  $k$  is now generated by these Base B-Spline curves. Figure 1 shows the Base B-Spline curves for order 7, satisfying above-mentioned properties.



Base B-Spline curve of order 7, with open Basis

Figure 1

### 3.3 Fourier Transform of Rational Form of B-Spline Curve

A Rational B-Spline Curve is mathematically defined by equation (2). Taking DFT of Polygon points as mentioned in equation (4), we get

$$R_{n,k}(t) = \frac{\sum_{i=1}^{n+1} h_i \sum_{L=0}^n \bar{p}_L w^{-Li} N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)}$$

$$\text{Or, } R_{n,k}(t) = \frac{\sum_{L=0}^n \bar{p}_L \sum_{i=1}^{n+1} h_i N_{i,k}(t) w^{-Li}}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)}$$

$$\text{Finally, } R_{n,k}(t) = \sum_{L=0}^n \bar{P}_L \text{RBB}_{L,k}(t) \tag{8}$$

$$\text{where, } \text{RBB}_{L,k}(t) = \frac{\sum_{i=1}^{n+1} h_i N_{i,k}(t) w^{-Li}}{\sum_{i=1}^{n+1} h_i B_{i,k}(t)} \text{ for } L=0,1,\dots,n \tag{9}$$

It can be easily see that  $\text{RBB}_{L,k}(t)$  's are rational form of Base B-Spline curve mentioned in section 3.1, with the weight assigned to original polygon points get shifted to the new polygon points, i.e. on polygon points obtained by power of twiddle factors.

Thus in transform case of Rational B-Spline curve, it gets generated by rational form of Base B-Spline curve, using knot vectors for defined B-Spline curve

#### 4. EXAMPLE

We are showing the various cases of B-Spline curve and its rational form in transformed domain, on the basis of type of knot vectors, for the B-Spline curve defined by polygon points in Table 1.

**Table 1**

Original polygon points	Polygon points obtained in transformed domain Using FFT
(0,0)	(3.5,4.75)
(2,8)	(-1.250081,1.499977)
(3,1)	(-2,-4.25)
(9,10)	(-0.249919,-1.999977)

**Table 2**

Case 1: Open B-Spline curve of 4 <sup>th</sup> order with 4 polygon vertices. Useful parameter range : 0 to 1				
t	B-Spline curve		Rational B-Spline curve	
	Points on original curve	Points in case of Transformed Domain	Points on original curve	Points in case of Transformed Domain
0	(0,0)	(-2.22e-16, -3.552e-16)	(0,0)	(-2.22e-16,-3.552e-16)
0.25	(1.4062,3.6718)	(1.4063,3.671879)	(1.58241,4.956044)	(1.582642,4.95604)
0.5	(3,4.625)	(3.000168,4.62489)	(2.727273,5.54545)	(2.72750,5.545375)
0.75	(5.3437,5.7656 25)	(5.34416,5.769057)	(4.93150,6.041096)	(4.93191,6.040633)
1	(9,10)	(9,10)	(9,10)	(9,10)
<b>Error</b>	<b>(0.000093118, 0.0006800)</b>		<b>(0.00010484, 0.000093973)</b>	

The Case is shown in Figure (2(a)) for B-Spline curve and its rational form in the given domain and by figure 2(d) for B-Spline curve and its rational form in the transformed

domain. In this case B-Spline curve behaves similar to Bézier curve, so we get the result as that for Bézier curve in transformed domain.

Table 3

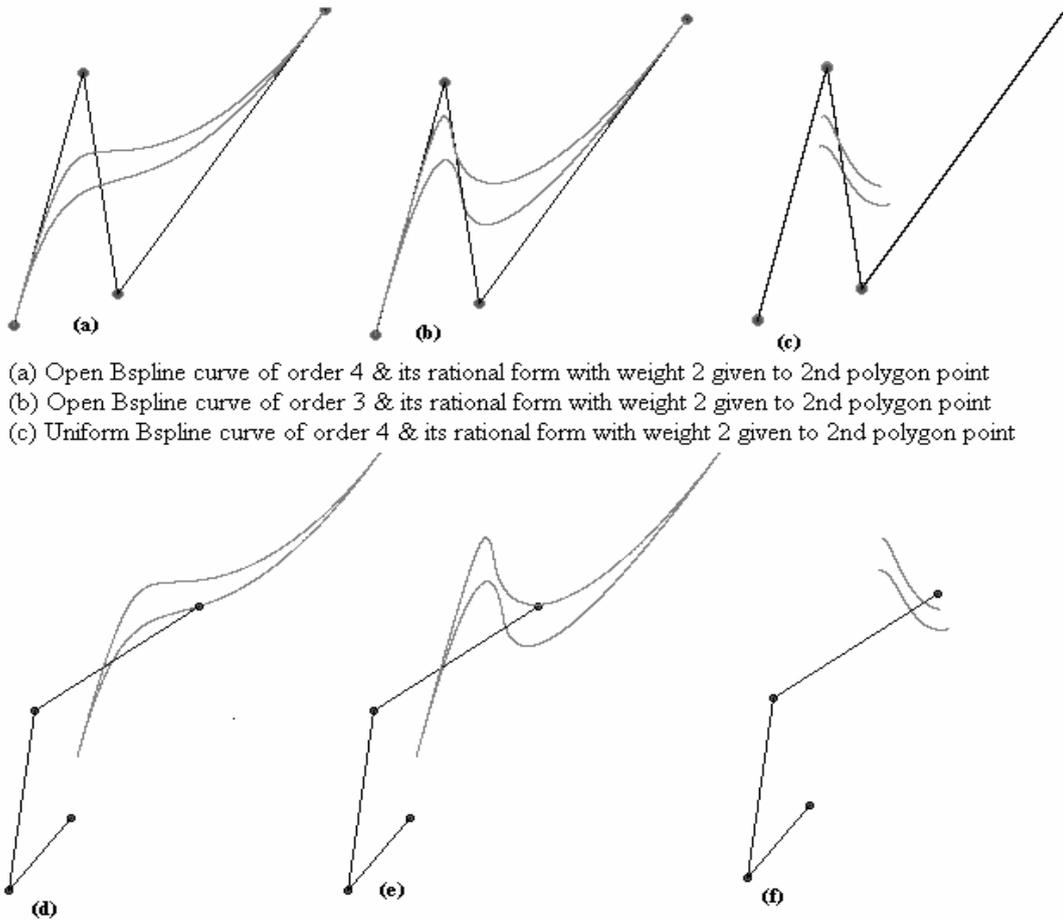
<b>Case 2: Open B-Spline curve of 3<sup>rd</sup> order with 3 polygon vertices .</b>				
<b>Useful parameter range: 0 to 2</b>				
<b>t</b>	<b>B-Spline curve</b>		<b>Rational B-Spline curve</b>	
	<b>Points on original curve</b>	<b>Points in case of Transformed Domain</b>	<b>Points on original curve</b>	<b>Points in case of Transformed Domain</b>
0	(0,0)	(-2.22044e-16, -3.552714e-16)	(0,0)	(-2.22044e-16, -3.552714e-16)
.5	(1.625,5.125)	(1.62521,5.125023)	(1.769231,6.230769)	(1.769523,6.230783)
1	(2.5,4.5)	(2.5000,4.500093)	(2.333333,5.666667)	(2.333488,5.666728)
1.5	(4.375,4.125)	(4.37512,4.124757)	(4.111111,4.555556)	(4.111265,4.555339)
2	(9,10)	(9,10)	(9,10)	(9,10)
<b>Error</b>	<b>(0.000049480905, 0.000192299557)</b>		<b>(0.000073196721, .0000451690159)</b>	

The Case is shown in figure 2(b) as B-Spline curve and it's rational form in the given domain, and figure 2(e) represents B-Spline curve and its rational form in transformed domain.

Table 4

<b>Case 3: Uniform B-Spline curve of 4<sup>th</sup> order with 3 polygon vertices.</b>				
<b>Useful parameter range : 3 to 4</b>				
<b>t</b>	<b>B-Spline curve</b>		<b>Rational B-Spline curve</b>	
	<b>Points on original curve</b>	<b>Points in case of Transformed Domain</b>	<b>Points on original curve</b>	<b>Points in case of Transformed Domain</b>
3	(1.833333,5.5)	(1.83355,5.500031)	( 1.9,6.5)	(1.90029,6.50001)
3.25	(2.1927,5.2369)	(2.19285,5.237034)	(2.119548,6.285945)	(2.11979,6.28597)
3.5	(2.5833,4.5208)	(2.58338,4.520892)	(2.394366,5.647887)	(2.39453,5.64792)
3.75	(3.0989,3.8359)	(3.098948,3.83595)	(2.835644,4.833663)	(2.83573,4.83367)
4	(3.8333,3.6667)	(3.83336,3.666551)	(3.571429,4.285714)	(3.57150,4.28561)
<b>Error</b>	<b>(0.00005298509, 0.0000290289510)</b>		<b>(0.00008730246273, .0000228192900)</b>	

The Case is shown in figure 2(c) as B-Spline curve and it's rational form in the given domain, and figure 2(f) represents B-Spline curve and its rational form in transformed domain.



(a) Open B-spline curve of order 4 & its rational form with weight 2 given to 2nd polygon point  
 (b) Open B-spline curve of order 3 & its rational form with weight 2 given to 2nd polygon point  
 (c) Uniform B-spline curve of order 4 & its rational form with weight 2 given to 2nd polygon point

Curves for figures (a), (b) and (c) in transformed case of polygon points

**Figure 2**

**5. FOURIER ANALYSIS FOR INTERVAL FORM OF BSPLINE CURVE**

During the process of design, various parameters are specified. Instead of simply assigning one real value to a parameter; interval representation captures the uncertainty characteristics of design.

**5.1 Rectangular Interval B-Spline Curve in Transformed Domain**

Vector valued interval [P] in the most general terms is defined as any compact set of points (x,y) in two dimensions as tensor products of scalar intervals[8]:

$$[P]=[a,b] \times [c,d]=\{(x,y) | x \in [a,b] \text{ and } y \in [c,d]\}.$$

Such vector-valued intervals are clearly just rectangular regions in a plane.

An interval B-Spline curve with rectangular vector interval [P<sub>i</sub>] as control polygon vector vertices, is written in the form

$$[P](t) = \sum_{i=1}^{n+1} [P_i] N_{i,k}(t) \tag{10}$$

where N<sub>i,k</sub>(t)'s are the B-Spline basis. For each t defined in given parameter range of knot vector of the given type, the value [P](t) of the interval B-Spline curve is a vector interval that has the following significance: for any B-Spline curve P(t) whose control

points satisfy  $P_i \in [P_i]$  for  $i = 1, \dots, n+1$  we have  $P(t) \in [P](t)$ . Likewise, the entire interval B-Spline curve (10) defines a region in the plane that contains all B-Spline curves of similar type whose control points satisfy  $P_i \in [P_i]$  for  $i=1, \dots, n+1$ .

Interval B-Spline curve defined by (10) can be written in centered form as follows

$$[P](t) = \sum_{i=1}^{n+1} P_{c_i} N_{i,k}(t) + [-1,1] \sum_{i=1}^{n+1} E_i N_{i,k}(t) \quad (11)$$

where,

$$P_{c_k} = \left( \frac{(a_k + b_k)}{2}, \frac{(c_k + d_k)}{2} \right) \text{ and } E_k = \left( \frac{(b_k - a_k)}{2}, \frac{(d_k - c_k)}{2} \right) \text{ are the}$$

center and error of vector interval  $P_k$  respectively.

To analyze the interval form of B-Spline on rectangular domain, we first consider the center  $P_{c_k}$  as points on complex plane i.e.

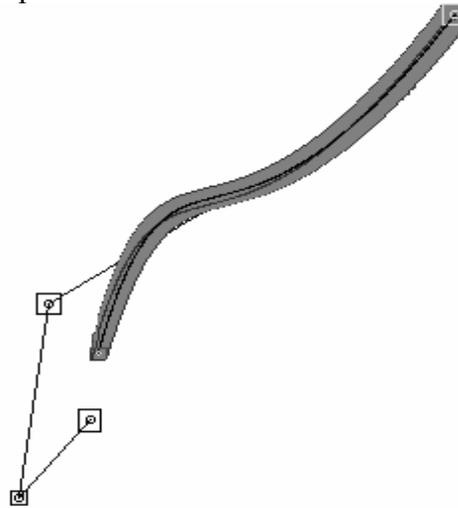
$$P_{c_k} = x_{c_k} + iy_{c_k} = \left( \frac{(a_k + b_k)}{2} + i \frac{(c_k + d_k)}{2} \right) = Z_{c_k}.$$

Using FFT algorithm and the definition (3) we take the DFT of these centered points and obtain,

$$[P](t) = \sum_{L=0}^n \bar{Z}_{c_L} BB_{L,k}(t) + [-1,1] \sum_{i=1}^{n+1} E_i N_{i,k}(t) \quad (12)$$

where  $\bar{Z}_{c_L}$  is the transformed point, of center of rectangular control vector interval.

$BB_{L,k}(t)$  is the base B-Spline Curves given by (7) for  $L=0, \dots, n$ . Thus in transformed domain interval form of B-Spline curve get transformed as a curve, with centered B-Spline curve generated by base B-Spline curve and its error component, which is used to capture uncertainty characteristic of design, is used as it is. Figure 3 shows the rectangular interval B-Spline curve in the transformed domain.



Interval B-Spline curve using rectangular vector interval in transformed domain

**Figure 3**

### 5.2 Circular Interval B-Spline Curve in the Transformed Domain

Interval B-Spline on circular domain is more natural generalization than on rectangular domain. It is defined as [6, 7]

$$(P)t = \sum_{i=0}^n (P_i)N_{i,k}(t) \quad (13)$$

where,

$$(P_i) = (c_{x_i}, c_{y_i}) + \varepsilon_i (\cos\theta, \sin\theta), \text{ and } , 0 \leq \theta \leq 2\pi$$

$(c_{x_i}, c_{y_i})$ 's are the given polygon points, with allowable error tolerance  $\varepsilon_i$  and  $N_{i,k}(t)$ 's are the B-Spline basis function.

To analyze using DFT, we take the circular Interval on complex Plane as

$$(P_i) = \{z \in Z : |z - z_i| \leq \varepsilon_i\}$$

i.e.

$$(P_i) = \{z \in Z : z = z_i + \varepsilon_i e^{i\theta}, 0 \leq \theta \leq 2\pi\}$$

where  $z_i$  is the centre of disk, and  $\varepsilon_i$  is the radius, taken to capture the uncertainty characteristic of design.

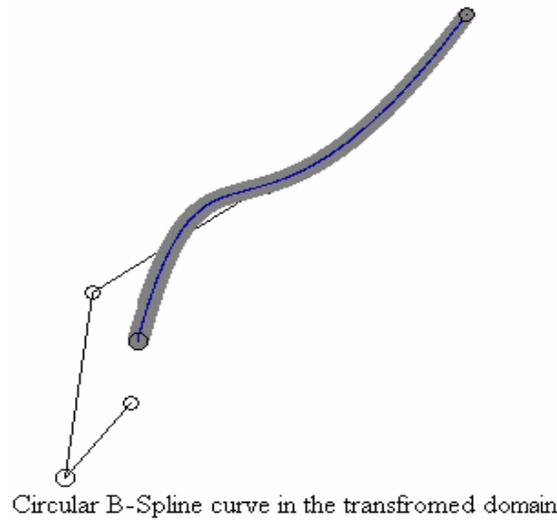
Thus, interval B-Spline curve using circular domain on complex plane, in component form can be written as

$$\begin{aligned} (P)t &= \sum_{i=1}^{n+1} (P_i)N_{i,k}(t) \\ &= \sum_{i=1}^{n+1} (z_i + \varepsilon_i e^{i\theta})N_{i,k}(t) \\ &= \sum_{i=0}^n z_i N_{i,k}(t) + e^{i\theta} \sum_{i=0}^n \varepsilon_i N_{i,k}(t) \quad \text{for } 0 \leq \theta \leq 2\pi \end{aligned} \quad (14)$$

Using FFT algorithm and definition (3) we take the DFT of centered from and obtain

$$(P)(t) = \sum_{L=0}^n \bar{z}_L BB_{L,k}(t) + e^{i\theta} \sum_{i=1}^{n+1} \varepsilon_i N_{i,k}(t) \quad , \text{for } 0 \leq \theta \leq 2\pi \quad (15)$$

Where,  $\bar{z}_L$  is the transformed point of, center of control disk and  $BB_{L,k}(t)$  is the base B-Spline curve given by (7) for  $L=0, \dots, n$ . Thus in transformed domain, original circular B-spline is again the curve generated by base B-Spline curve, with radius of disk used as in original domain, as shown in Figure 4.



**Figure 4**

## 6. CONCLUSION

In this paper, we apply FFT algorithm to analyze the inner structure of B-Spline curve and its rational form along with interval form, given by rectangular or circular vertices. Using this technique, we obtained the same curve in the transformed case of polygon points. The mentioned examples and figures justify the result, that after taking Fourier transform of polygon vertices of the B-Spline curve, original B-Spline gets generated by another form of B-Spline curve, named as base B-Spline curve.

We find by noting the error, which is coming very small around  $10^{-5}$ , we get here a choice of similar and different varieties, i.e., in original and its image shape obtained after transformation. We get a choice of design, whether to select the original curve or its image shape as per domain. The technique can be used in case of image processing too.

## 7. REFERENCES

1. Alan V.Oppenheim, Ronald W.Schafer: Digital Signal Processing, Prentice-Hall of India, 1997.
2. B.P.Lathi: Signal Processing and Linear Systems, Oxford university press, 2004.
3. Barry, Paul, The Fourier Analysis of Bézier Curves, Journal of Visual Mathematics, 5 (4), 2003. <http://www.mi.sanu.ac.yu/vismath/barry>.
4. David F.Rogers, J.Alan Adams: Mathematical Elements For Computer Graphics, Tata McGraw-HILL Edition II, 2002.
5. Douglas F.Elliott, K.Ramamohan Rao: Fast Transforms, Algorithms, Analysis, Application, Academic Press, INC, 2004.
6. Germain E. Randriambelosa: Space curve approximation using G1 ball-spline Curve Applied mathematical research Express 2004, No 5.
7. Lin Q, Rokne J G. Disk Bézier curve [J]. Computer Aided Geometric Design,1998,15(7):721-732.
8. Sederberg TW, Farouki RT. Approximation by interval Bézier curves. IEEE computer Graph Appl, 1992, 15(2):87-95.