

## A Modified BEZIER Algorithm For Controlling and Generating A Third Degree Curves

Dr. A.M.S. Rahma\*

Dr. A. M. Jaber\* &

Ali Adil Saeid\*

Received on: 18/3/2009

Accepted on: 7/10/2010

### Abstract

This paper presents a new method to generate curves, which allows the designer to produce a curve in combinational way allowing him to get the shape that he had in his imagination with keeping the four control point for the curve design.

Basing on the modified BEZIER curve, the upgraded algorithm used de Casteljau algorithm on interval  $[r, s]$  as a mathematical base to upgrade the suggested method.

The suggested method shows a great flexibility at the curve controlling area with changing one, more or no need to change the control points of Bezier curve as it shows in the figures presented in the paper.

### تطویر خوارزمیة (BEZIER) للتحكم وتولید المنحنيات من الدرجة الثالثة

#### الخلاصة

يقم البحث أسلوب جديداً لتوليد المنحنيات والذي يتيح للمصمم توليد المنحني وتطویره بطريقة تفاعلية تمكنه من الحصول على الشكل الذي كونه في مخيلته مع الاحتفاظ بتوافق المنحني مع نقاط السيطرة الأربع لتصميم المنحني.

الخوارزمية المطورة اعتمدت على أسلوب de Casteljau في تطوير منحني BEZIER على المجال  $[r, s]$  كأساس رياضي للتطویر. اثبتت الطريقة المقترنة مرونة عالية في مجال التحكم بالمنحني المصمم بتغيير بعض أوكل أو من دون تغيير نقاط السيطرة كما موضح في النتائج التي عرضها البحث.

### Introduction

Curve generating algorithms by using de Casteljau algorithm. In this proposed work that we modified for controlling and generating Bezier curves of degree three. Polynomial curve is defined in terms of polar forms. Because the polynomial curve involved containing one variable, natural way to polarize polynomial curve. The approach yields polynomial curve. It shows versions of the de-Casteljau algorithm can be turned into subdivisions, by giving an efficient method of performing

subdivision. It is also shown that it is easy to compute a new control net from given net.

For this the paper develops an algorithm which should include:-

1- Original Bezier curve. We will use this curve for  $t \in [0, 1], [1], [2] [3], [4], [5], [6], [7]$ .

2- Gallier Modified Bezier curve. We will use this curve for  $t \in [r, s], [7], [8]$ .

3- Develop Bezier curve. We will use this curve for  $t \in [r, s]$ .

---

\*Computer Department, University of Technology/ Baghdad

**Definition**[4], [7].

A **linear combination** of m vectors  $\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_m$  is a vector, say  $\mathbf{W}$ , of the form

$$\mathbf{W} = a_1 \mathbf{V}_1 + a_2 \mathbf{V}_2 + \dots + a_m \mathbf{V}_m, \quad \dots(1)$$

Where  $a_1, a_2, \dots, a_m$  are scalar.

A linear combination (1) of vectors is called an **affine combination** if

$$a_1 + a_2 + \dots + a_m = 1$$

The coefficients of an affine combination of two vectors  $\mathbf{V}_1$  and  $\mathbf{V}_2$  are often forced to have scalar t for one of them and (1-t) for the other in the form:

$$\mathbf{W} = (1-t)\mathbf{V}_1 + t\mathbf{V}_2.$$

Affine space  $A^n$  is called the real affine space of dimension n.

In most case, we will consider  $n=1, 2, 3$ .

A is defined over the field R (where R is real numbers)

**Polar Form of a Polynomial Curve**  
[7], [8].

A method of specifying polynomial curves that yield very nice geometric constructing the curves is based on polar polynomial form.

To define the polar from of a polynomial of degree three, let

$$X(t) = F_1(t) = a_1 t^3 + b_1 t^2 + c_1 t + d_1 \quad \dots(2)$$

$$Y(t) = F_2(t) = a_2 t^3 + b_2 t^2 + c_2 t + d_2, \quad \dots(3)$$

Where  $a_1, b_1, c_1, d_1, a_2, b_2, c_2$  and  $d_2$  are constants. The polar form of F is a symmetric affine function

$f: A^3 \rightarrow A$  that takes the same value for all permutations of  $t_1, t_2, t_3$ . That is  $f(t_1, t_2, t_3) = f(t_2, t_1, t_3) = f(t_1, t_3, t_2) = f(t_2, t_3, t_1) = f(t_3, t_1, t_2) = f(t_3, t_2, t_1)$ .

Which is affine in each argument and  $f(t) = f(t, t, t)$ , for all  $t \in R$ . It is easily to verify that f must have the form:

$$f(t_1, t_2, t_3) = at_1 t_2 t_3 + b[t_1 t_2 + t_1 t_3 + t_2 t_3]/3 + c[t_1 + t_2 + t_3]/3 + d \quad \dots(4)$$

This is called the polar form of cubic polynomial in (2) or (3). {Note that Polar form a new definition has no relation to polar coordinate. [7], [8].}

**Example:**

Consider the plane cubic define as follows

$$F_1(t) = 30t \quad \dots(5)$$

$$F_2(t) = t^3 - 3t \quad \dots(6)$$

The polar forms of  $F_1 = X(t), F_2 = Y(t)$

Are

$$f_1(t_1, t_2, t_3) = 10[t_1 + t_2 + t_3] \quad \dots(7)$$

$$f_2(t_1, t_2, t_3) = t_1 t_2 t_3 - [t_1 + t_2 + t_3]. \quad \dots(8)$$

It is easy to notice that  $f_1(t_1, t_2, t_3)$  in (7) is polar form of  $F_1 = X(t)$  in (5), and  $f_2(t_1, t_2, t_3)$  in (8) is polar form of  $F_2 = Y(t)$  in (6).

**Note 1**[2], [7], [8].

Using two control points say  $P_0$  and  $P_1$  shows that  $P(t)$  is on the line determined by  $P_0$  and  $P_1$ ,  $P(t)$  is indeed between  $P_0$  and  $P_1$ , at (1-t) of the way from  $P_0$ , every point  $P(t)$  on the line  $P_0 P_1$  is obtained by a single interpolation step,  $P(t) = (1-t)P_0 + tP_1$ . Which linear interpolation at  $P(0) = P_0$  and  $P(1) = P_1$ , for  $0 \leq t \leq 1$ .

**De-Casteljau Algorithms**

De-Casteljau algorithm is an algorithm uses a sequence of control points to construct a well defined curve  $P(t)$  at each value of t from 0 to 1. This provides a way to generate a curve from set of points. Changing the points will change the curve.

Let us illustrate the algorithm of degree three in Table (1).

The de Casteljau algorithm uses a sequence of four control points  $P_0, P_1, P_2$  and  $P_3$  to construct a well-defined curve  $P(t)$  at each value of t from 0 to 1.  $P(t)$  is defined as:

$$P(t) = (1-t)p_0^2(t) + t p_1^2(t) \quad \dots(9)$$

$$p_0^2(t) = (1-t)p_0^1(t) + t p_1^1(t) \quad \dots(10)$$

$$p_1^2(t) = (1-t)p_1^1(t) + t p_2^1(t) \quad \dots(11)$$

$$p_1^1(t) = (1-t)P_1 + tP_2 \quad \dots(12)$$

$$p_0^1(t) = (1-t)P_0 + tP_1 \quad \dots(13)$$

$$p_2^1(t) = (1-t)P_2 + tP_3 \quad \dots(14)$$

Substitution of equations (10, 11, 12, 13), and (14) in (9) gives

$$\begin{aligned} P(t) &= (1-t)^3 P_0 + 3(1-t)^2 t P_1 \\ &+ 3(1-t)t^2 P_2 + t^3 P_3, \end{aligned} \quad \dots(15)$$

Equation (15) called original Bezier curve which is dependent on closed interval  $[0, 1]$ . [3], [5], [6], [7], [9].

#### Note 2.

The coefficients of the control points in the cubic curve (15) are called Bernstein Polynomials, gives as:

$$\begin{aligned} &[(1-t)^3 + 3(1-t)^2 t + 3(1-t)t^2 + t^3] \\ &= [(1-t) + t]^3 = 1 \end{aligned}$$

This equation immediately yields an important property of these polynomials. They add to unit at every  $t$ , mathematically.

#### Gallier Modified cubic Bezier Curves [7], [8].

The modified de-Casteljau algorithm is an algorithm that uses a sequence of control points

$$p_i = f(r^{3-i} s^i), \text{ for } i=0, 1, 2, 3.$$

To construct a well defined curve  $f(t)=f(t, t, t)$ , on a given affine frame  $[r, s]$  for which  $r \neq s$ ,  $t \in [r, s]$ , and  $(r, s \in A)$ . Let us begin with straight lines, given any affine frame  $[r, s]$  for which  $r \neq s$  for  $t \in A$  can be written uniquely as

$$t = [1 - \lambda]r + \lambda s = r + \lambda[s - r]. \quad \dots(16)$$

$$\text{then, } \lambda = \frac{t - r}{s - r},$$

... (17)

$$\text{and } 1 - \lambda = \frac{s - t}{s - r}.$$

... (18)

Since  $F$  is affine then

$$F(t) = F[(1 - \lambda)r + \lambda s].$$

Therefore

$$F(t) = (1 - \lambda)F(r) + \lambda F(s). \quad \dots(19)$$

De Casteljau algorithm uses two control points say  $F(r)$  and  $F(s)$  as in equation (19).

#### Cubic Curve

De Casteljau algorithm of degree three uses a sequence of four control points, which are  $p_0 = f(r^3) = f(r, r, r)$ ,  $p_1 = f(r^2 s) = f(r, r, s)$ ,  $p_2 = f(r s^2) = f(r, s, s)$  and  $p_3 = f(s^3) = f(s, s, s)$ . To construct a well defined curve  $P(t) = F(t)$  at each value of  $t$  from  $r$  to  $s$ . see table (2) fig 2.

From the stage 3,  $f(t, t, t)$  is given by:

$$\begin{aligned} f(t, t, t) &= f(t, t, (1-\lambda)r + \lambda s) \\ &= (1-\lambda)f(t, t, r) \\ &+ \lambda f(t, t, s), \end{aligned} \quad \dots(20)$$

in termes of  $f(t, t, r)$  and  $f(t, t, s)$ .

From the stage 2,  $f(t, t, r)$  and  $f(t, t, s)$  are given by:

$$\begin{aligned} f(t, t, r) &= f(t, (1-\lambda)r + \lambda s, r) \\ &= (1-\lambda)f(t, r, r) + \lambda f(t, s, r) \end{aligned} \quad \dots(21)$$

$$\begin{aligned} f(t, t, s) &= f(t, (1-\lambda)r + \lambda s, s) \\ &= (1-\lambda)f(t, r, s) \\ &+ \lambda f(t, s, s) \end{aligned} \quad \dots(22)$$

From the stage 1,  $f(t, t, t)$  is given by:

$$\begin{aligned} f(t, r, r), f(r, t, s) \text{ and } f(t, s, s) \text{ which are} \\ f(r, r, t) &= f[r, r, (1-\lambda)r + \lambda s] \\ &= (1-\lambda)f(r, r, r) \\ &+ \lambda f(r, r, s) \end{aligned} \quad \dots(23)$$

$$\begin{aligned} f(r, t, s) &= f(r, (1-\lambda)r + \lambda s, s) \\ &= (1-\lambda)f(r, r, s) \\ &+ \lambda f(r, s, s) \end{aligned} \quad \dots(24)$$

$$\begin{aligned} f(t, s, s) &= f((1-\lambda)r + \lambda s, s, s) \\ &= (1-\lambda)f(r, s, s) \end{aligned}$$

$$+ \lambda f(s, s, s). \quad \dots(25)$$

From symmetry  $f(r, r, t) = f(t, r, r)$  and  $f(r, t, s) = f(t, s, r) = f(t, r, s)$ , and substitution of equations (21, 22, 23, 24) and (25) in (20) gives

$$\begin{aligned} f(t, t, t) &= (1-\lambda)^3 f(r, r, r) + 3\lambda(1-\lambda)^2 f(r, r, s) \\ &+ 3(1-\lambda)\lambda^2 f(r, s, s) \\ &+ \lambda^3 f(s, s, s). \end{aligned} \quad \dots(26)$$

Use the de Casteljau algorithm in cubic cases, to define a curve  $P(t)$  at each value of  $t$  from  $r$  to  $s$ , where, ( $r, s \in A$ ) for some set  $A$ . Suppose  $P(t)$  is a Gallier Modified cubic Bezier curve. Since  $f(r, r, r), f(r, r, s), f(r, s, s), f(s, s, s) = P_0, P_1, P_2, P_3$  are control points equation (26) becomes

$$P(t) = (1-\lambda)^3 p_0 + 3\lambda(1-\lambda)^2 p_1 + 3(1-\lambda)\lambda^2 p_2 + \lambda^3 p_3. \quad \dots(27)$$

Equation (27) called original Gallier modified Bezier curve depend on interval  $[r, s]$ .

Substitution of  $\lambda$  and  $1-\lambda$  in (27) where  $\lambda = \frac{t-r}{s-r}$  and  $1-\lambda = \frac{s-t}{s-r}$  gives:

$$\begin{aligned} P(t) &= \left[ \frac{s-t}{s-r} \right]^3 p_0 + 3 \frac{t-r}{s-r} \left[ \frac{s-t}{s-r} \right]^2 p_1 \\ &+ 3 \frac{s-t}{s-r} \left[ \frac{t-r}{s-r} \right]^2 p_2 + \left[ \frac{t-r}{s-r} \right]^3 p_3. \end{aligned} \quad \dots(28)$$

### Note 3

The coefficients of the control points in the cubic curve (28) are called Bernstein Polynomials gives as:

$$\begin{aligned} \left[ \frac{s-t}{s-r} + \frac{t-r}{s-r} \right]^3 &= \left[ \frac{s-t}{s-r} \right]^3 + 3 \frac{t-r}{s-r} \\ &\left[ \frac{s-t}{s-r} \right]^2 + 3 \frac{s-t}{s-r} \left[ \frac{t-r}{s-r} \right]^2 \\ &+ \left[ \frac{t-r}{s-r} \right]^3 = 1. \end{aligned}$$

The cubic Bernstein polynomials are remembered as the terms that get by expanding the expression and then collecting terms in the various powers of  $\frac{s-t}{s-r}$  and  $\frac{t-r}{s-r}$ .

This equation immediately yields an important property of these polynomials. They add to unity at every  $t$ , mathematically

$$\sum_{k=0}^3 \binom{3}{k} \left[ \frac{s-t}{s-r} \right]^{3-k} \left[ \frac{t-r}{s-r} \right]^k = 1 \quad \dots(29)$$

where

$$\binom{3}{k} = \frac{3!}{k!(3-k)!}$$

Equation (29) is called Bernstein polynomial of degree three [7], [8]. These control points play a major role in the de Casteljau algorithm and its extensions. The polynomial curve defined  $F$  passes through two point's which are  $p_0$  and  $p_3$ , but not through the other control points. For  $r=0$  and  $s=1$ , equation (27) becomes: -

$$F(t) = f(t, t, t) = (1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t)t^2 p_2 + t^3 p_3. \quad \dots(30)$$

Equation (30) identical with equation (15).

### The Developed Cubic Bezier Curve

The new mathematical procedure has been introduced to control and generate curves. This new technique gives the designer a flexibility to design and generate curves leaving the main properties for designing curves intact.

Concerning Bezier curves, a new algorithm is developed. In connection with equation (27) a curve depending on four control points and a parameter  $t$  working on interval  $[r, s]$  rather than  $[0, 1]$ ;  $r, s \in A$  for some set  $A$ .

New algorithm for calculating

$f(t, t, t_6)$  at each value of  $t$  where  $r \leq t_j \leq s$  ( $j=1, 2, 3, 4, 5, 6$ ) is given in Table (3). From the stage 3,  $f(t, t, t_6)$  is given by :

$$\begin{aligned} f(t, t, t_6) &= f(t, t, (1-\lambda_6)r + \lambda_6 s) \\ &= (1-\lambda_6)f(t, t_4, r) \\ &\quad + \lambda_6f(t, t_5, s), \end{aligned} \quad \dots(31)$$

In termes of  $f(t, t_4, r)$  and  $f(t, t_5, s)$ . From the stage 2,  $f(t, t_4, r)$  and  $f(t, t_5, s)$  is given by:

$$\begin{aligned} f(t, t_4, r) &= f(t, (1-\lambda_4)r + \lambda_4 s, r) \\ &= (1-\lambda_4)f(t_1, r, r) \\ &\quad + \lambda_4f(t_2, s, r). \end{aligned} \quad \dots(32)$$

$$\begin{aligned} f(t, t_5, s) &= f(t, (1-\lambda_5)r + \lambda_5 s, s) \\ &= (1-\lambda_5)f(t_2, r, s) \\ &\quad + \lambda_5f(t_3, s, s). \end{aligned} \quad \dots(33)$$

From the stage 1,  $f(t_1, r, r)$ ,  $f(t_2, s, r)$  and  $f(t_3, s, s)$  are given by:

$$\begin{aligned} f(t_1, r, r) &= f[r, r, (1-\lambda_1)r + \lambda_1 s] \\ &= (1-\lambda_1)f(r, r, r) \\ &\quad + \lambda_1f(r, r, s) \end{aligned} \quad \dots(34)$$

$$\begin{aligned} f(t_2, s, r) &= f(r, (1-\lambda_2)r + \lambda_2 s, s) \\ &= (1-\lambda_2)f(r, r, s) \\ &\quad + \lambda_2f(r, s, s) \end{aligned} \quad \dots(35)$$

$$\begin{aligned} f(t_3, s, s) &= f((1-\lambda_3)s + \lambda_3 s, s, s) \\ &= (1-\lambda_3)f(r, s, s) \\ &\quad + \lambda_3f(s, s, s) \end{aligned} \quad \dots(36)$$

Then by symmetry,  $f(r, r, t_1) = f(t_1, r, r)$  and  $f(r, t_2, s) = f(t_2, s, r)$ .

Substitution of equations 32, 33, 34, 35 and 36 in 31 gives

$$\begin{aligned} F(t) &= (1-\lambda_6)(1-\lambda_4)(1-\lambda_1)f(r, r, r) \\ &\quad + [\lambda_1(1-\lambda_6)(1-\lambda_4) \\ &\quad + (1-\lambda_2)(1-\lambda_6)\lambda_4 \\ &\quad + (1-\lambda_2)(1-\lambda_5)\lambda_6]f(r, r, s) \\ &\quad + [\lambda_2\lambda_4(1-\lambda_6) \\ &\quad + \lambda_6(1-\lambda_5)\lambda_2]f(r, s, s) \\ &\quad + \lambda_3\lambda_5\lambda_6f(s, s, s) \end{aligned} \quad \dots(37)$$

Where  $\lambda_j = \frac{t_j - r}{s - r}$ ,  $1 - \lambda_j = \frac{s - t_j}{s - r}$ , and  $j=1, 2, 3, 4, 5, 6$  for  $r \leq t_j \leq s$ .

From  $\lambda_6 = \lambda_5 = \lambda_3 = \lambda_1$ , then equation (37) becomes:

$$\begin{aligned} f(t) &= (1-\lambda_4)(1-\lambda_1)^2f(r, r, r) \\ &\quad + [\lambda_1(1-\lambda_4)(1-\lambda_4) + (1-\lambda_1)(1-\lambda_2)\lambda_4 \\ &\quad + (1-\lambda_2)(1-\lambda_1)\lambda_4]f(r, r, s) \\ &\quad + [\lambda_2\lambda_4(1-\lambda_1) + \lambda_1\lambda_2(1-\lambda_1) + \lambda_1^2 \\ &\quad (1-\lambda_1)]f(r, s, s) \\ &\quad + \lambda_1^3f(s, s, s). \end{aligned} \quad \dots(38)$$

Suppose  $P(t) = f(t)$  and  $f(r, r, r)$ ,  $f(r, r, s)$ ,  $f(r, s, s)$ ,  $f(s, s, s) = P_0, P_1, P_2, P_3$  are control points and equation (38) becomes

$$\begin{aligned} P(t) &= (1-\lambda_4)(1-\lambda_1)^2P_0 + [\lambda_1(1-\lambda_4)(1-\lambda_4) \\ &\quad + (1-\lambda_1)(1-\lambda_2)\lambda_4 + (1-\lambda_2)(1-\lambda_1)\lambda_4]P_1 + [\lambda_2\lambda_4(1-\lambda_1) + \lambda_1\lambda_2(1-\lambda_1) + \lambda_1^2(1-\lambda_1)]P_2 + \lambda_1^3P_3. \end{aligned} \quad \dots(39)$$

Equation (39) is a **new cubic Bezier curve**.

Treat the coordinates of each point as a two-component vector and using the symbols  $p_0, p_1, p_2$  and  $p_3$  for control points. Let

$$p_i = (x_i, y_i) \text{ for } i=0, 1, \dots, m.$$

And

$$P_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix}.$$

The set of points, in parametric form is

$$P(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \quad r \leq t \leq s, [3], [7],$$

[8].

The pressing equation (39) represents the pair of equations

$$\begin{aligned} F_1(t) &= x(t) = (1-\lambda_4)(1-\lambda_1)^2x_0 + [\lambda_1(1-\lambda_4)(1-\lambda_4) + (1-\lambda_1)(1-\lambda_2)\lambda_4 + (1-\lambda_2)(1-\lambda_1)\lambda_4]x_1 + [\lambda_2\lambda_4(1-\lambda_1) + \lambda_1\lambda_2(1-\lambda_1) + \lambda_1^2(1-\lambda_1)]x_2 + \lambda_1^3x_3. \end{aligned}$$

$$\begin{aligned} F_2(t) &= y(t) = (1-\lambda_4)(1-\lambda_1)^2y_0 + [\lambda_1(1-\lambda_4)(1-\lambda_4) + (1-\lambda_1)(1-\lambda_2)\lambda_4 + (1-\lambda_2)(1-\lambda_1)\lambda_4]y_1 + [\lambda_2\lambda_4(1-\lambda_1) + \lambda_1\lambda_2(1-\lambda_1) + \lambda_1^2(1-\lambda_1)]y_2 + \lambda_1^3y_3. \end{aligned}$$

The set of points, in parametric form is  
To explain the above new cubic Bezier curve (39) by the following example is given

**Example: -**

Let the four control points as:

$$P_0 = (x_0, y_0) = (100, 300),$$

$$P_1 = (x_1, y_1) = (40, 50),$$

$$P_2 = (x_2, y_2) = (430, 50),$$

$$P_3 = (x_3, y_3) = (270, 300).$$

With respect to the interval  $r=-5$ ,  $s=5$ . For parameter  $t$  where  $-5 \leq t_j \leq 5$ , where  $t$  is increases by ( $t=0.0005$ ).

The following cases have been discussed.

**Case 1**

Let  $\lambda_6=\lambda_5=\lambda_4=\lambda_3=\lambda_2=\lambda_1$ . Then equation (39) reduces to the original Bezier curve in equation (15) see figure 3. Equation (39) is building by mathematical developing of de Casteljau algorithm. This case can be seen by simply mathematical changing of equation (39) which reduces to modified equation (27), that easily reduces to original equation (30) by take special case for  $r=0$  and  $s=1$ . This gives the base classical populates of Bezier curve that are interpolations at the first point  $P_0$  and the last point  $P_3$ . The design obtained does **not change** only from **change at least one or more** of control points of Bezier curve.

**Case 2:**

Let  $\lambda_6=\lambda_5=\lambda_4=\lambda_3=\lambda_1$ , and  $(+\lambda_2)$  is take to be increases ( $t_2 = 0.02$ ). In this case it is found that the design can be moved to **right** in an **interior** and **exterior** manner with **no need to change** any of the control points, see figure 4. New formula of equation (39). In this case which is build by a new change easily done by changing

only part  $\lambda_2$  (*i.e.*  $t_2 = 0.02$ ), which gives the new design and move it to right as see in figure 4.

**Case 3:**

Let  $\lambda_6=\lambda_5=\lambda_4=\lambda_3=\lambda_1$ , and  $(-\lambda_2)$  is taken to be decreased. It is found in this case that the design can be moved to **left** in an **exterior** and **interior** manner with **no need to change** any of the control points, see figure 5. In this case as in case 2 building by new condition in this which give new formula of equation (39). by changing only part  $\lambda_2$  (*i.e.*  $t_2 = -0.02$ ) can change the design to left, see figure 5.

**Case 4:**

Let  $\lambda_6=\lambda_5=\lambda_3=\lambda_2=\lambda_1$ , and  $(+\lambda_4)$  increased. In this case, it is found that the design can be moved **right** to **(exterior manner)** with **no need to change** any of the control points. See figure 6. By variation in last equation only part  $\lambda_4$  (*i.e.*  $t_4 = 0.02$ ) the design can be moved to **right** with out changing any of the control points. See figure 6.

**Case 5:**

Let  $\lambda_6=\lambda_5=\lambda_3=\lambda_2=\lambda_1$ , and  $(-\lambda_4)$  decreased. In this case, it is found that the design can be moved to the **left** to **(interior manner)** with **no need to change** any of the control points. See figure 7. By changing the formula of equation (39) as the condition of this case.

Variation at only part of this mathematically modeling at  $\lambda_4$  (*i.e.*  $t_4 = -0.02$ ), gives the new result is changing the design. See figure 7.

**Case 6:**

Let  $\lambda_6=\lambda_5=\lambda_3=\lambda_2=\lambda_1$ , and  $(+\lambda_2, -\lambda_2, \lambda_4$  and  $-\lambda_4)$  and the original are taken in same design. See fig 8. This again gives another characteristic to design and help the designer to control the generated curve keeping the **control points unchanged**. By

variation of the new formula of Bezier curve in equation (39) gives another modeling of mathematically equation which appear by change all parts Which gives the new formula by move the design up and down with out change the control points, also this result is new, figure 8.

Note that the building of Bezier cubic in equation (27) gives by Gallier modified the de Casteljau algorithm by successive mathematical steps until give the modified formula (39), which difference from initial equation (30) and Gallier modified equation (27).

Equation (39) modifies a curve generation algorithm to be used, which allows the designer to produce a curve and to modify its shape interactively until the desired curve is obtained.

The Bezier curve method is used in the developing curve generating algorithm. Actually the de Casteljau algorithm method for determining the curve

$F(t)=f(t, t, t)$  on the interval  $(r, s)$ , specified by the sequence of its 4 control points. The better of the algorithm is that it does not assume any prior knowledge about the curve.

### Conclusions

This work concludes The Bezier equation basing on a mathematical procedure depending on the linear construction of polynomials and following de Casteljau. This led to a general procedure that can be used easily.

A construction of a modified formula for Bezier curve has been achieved through a procedure followed by de Casteljau algorithm. This procedure has been developed in a sequential and mathematical way as it is obvious

in formula (39). It has the following advantages:

- (i) The modified linear mathematical construction of the equation gives the designer more room to control and construct his design. This is done through controlling certain parts of the equation which has eventually great effect on certain parts of the design or the design in large. This can be seen in the illustrate us in a above cases.
- (ii) The modified equation works on the extended interval  $[r, s]$  for the parameter  $t$  rather than the familiar special interval  $[0, 1]$ .
- (iii) The designer has the advantage of controlling and modifying the design with out changing any of the control points.
- (iv) The original equation named (classical) Bezier curve is a special case of our modified equation (39). That is the new modified equation can be reduced to (30) the original classical one easily.

### References

- [1] Faux I D (1983) Computational Geometry for Design and manufactured, Ellis Hood Ltd.
- [2] John R. Rice (1985) Numerical Methods Software . And Analysis.
- [3] Gerald C. F and Wheatley P. O. (1999) Applied Numerical Analysis. Addison Wesly.
- [4] Hill F. S, Jr (2001) Computer Graphics Using Open GL. Prentice Hall.
- [5] Buss Samuel R. 3-D (2003) Computer Graphics A Mathematical Introduction with Open GL.
- [6] Lengyel Eric (2004) Mathematics for 3D Game Programming and Computer Graphics Charles River Medla. Inc.
- [7] Jaber A. Mohssin (2005) Modeified Curves Mathematical

Models. Ph D thesis. University of Technology.  
**[8]** Gallier Jean (2000) Curves and Surfaces in Geometric Modeling Theory and Algorithms. Morgan Kaufmann publishers.

Modeling Theory and Algorithms.  
**[9]** Watt Alan (2000) ``3D Computer Graphics'', Published by Addison Wesley Company, Inc.

**Table (1) de Casteljau algorithm**

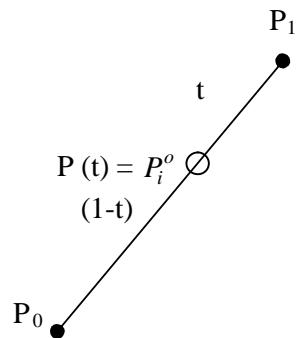
$(i, j)$	$0$	$1$	$2$	$3$
$0$	$P_0$			
		$p_0^1(t)$		
$I$	$P_I$		$p_0^2(t)$	
		$p_I^1(t)$		$p(t)$
$2$	$P_2$		$p_I^2(t)$	
		$p_2^1(t)$		
$3$	$P_3$			

**Table (2) de Casteljau algorithm**

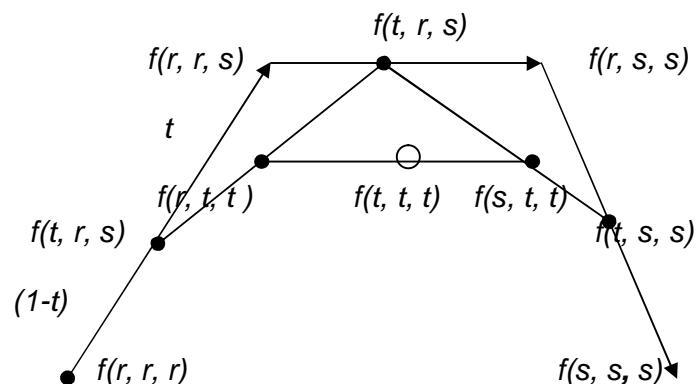
$(i, j)$	$0$	$1$	$2$	$3$
$0$	$f(r, r, r)$			
		$f(r, r, t)$		
$I$	$f(r, r, s)$		$f(t, t, r)$	
		$f(r, t, s)$		$f(t, t, t)$
$2$	$f(r, s, s)$		$f(t, t, s)$	
		$f(t, s, s)$		
$3$	$f(s, s, s)$			

**Table (3) de Casteljau algorithm**

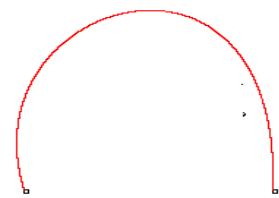
$(i, j)$	$0$	$1$	$2$	$3$
$0$	$f(r, r, r)$			
		$f(r, r, t_1)$		
$I$	$f(r, r, s)$		$f(t, t_4, r)$	
		$f(r, t_2, s)$		$f(t, t, t_6)$
$2$	$f(r, s, s)$		$f(t, t_5, s)$	
		$f(t_3, s, s)$		
$3$	$f(s, s, s)$			



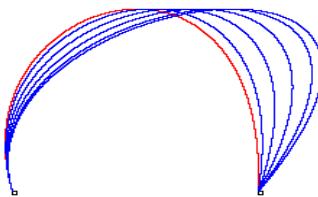
**Figure (1) Linear Curve Interpolation at  $t=0$  then,  $P(t) = P_0$ ,  
and, at  $t= 1$  then  $P(t) = P_1$**



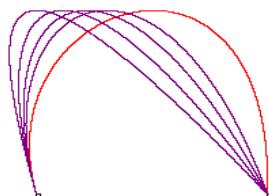
**Figure (2) Cubic curves**



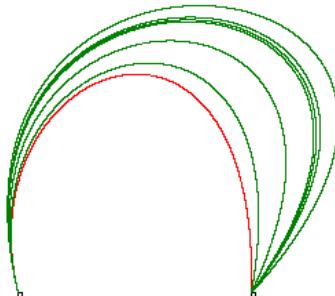
**Figure (3) Bezier Cubic** Which gives the base classical populates of Bezier curve are interpolations at first point  $P_0$  and last point  $P_3$  does not change the design only from change at least one or more of control points of Bezier curve.



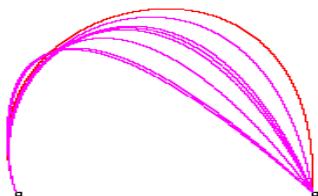
**Figure (4) Bezier cubic when  $(+ \lambda_2)$  increases the design can be moved to right in an interior and exterior manner with no need to change any of the control points.**



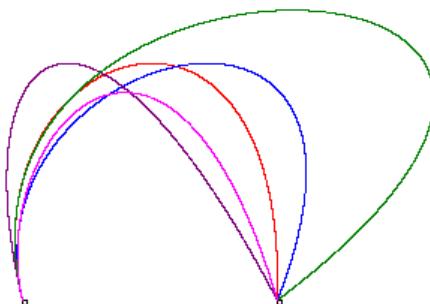
**Figure(5) Bezier cubic when  $(- \lambda_2)$  decreases the design can be moved to left in an exterior and interior manner with no need to change any of the control points**



**Figure (6)** Bezier cubic when  $(+\lambda_4)$  increases the design can be moved right to (exterior manner) with no need to change any of the control points.



**Figure (7)** Bezier cubic when  $(-\lambda_4)$  decreases the design can be moved to the left to (interior manner) with no need to change any of the control points .



**Figure (8)** Bezier cubic at all cases the designer controlling the generating curve keeping the control points unchanged.