UNIT-4

Polygon Meshes

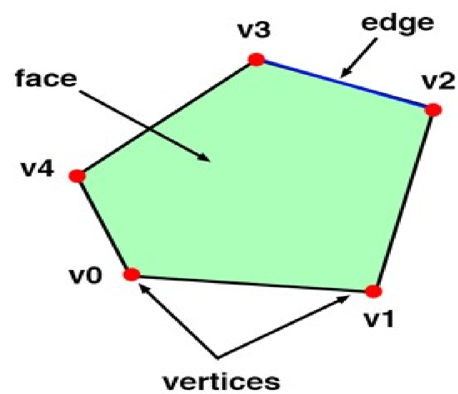
3D surfaces and solids can be approximated by a set of polygonal and line elements. Such surfaces are called **polygonal meshes**. In polygon mesh, each edge is shared by at most two polygons. The set of polygons or faces, together form the "skin" of the object.

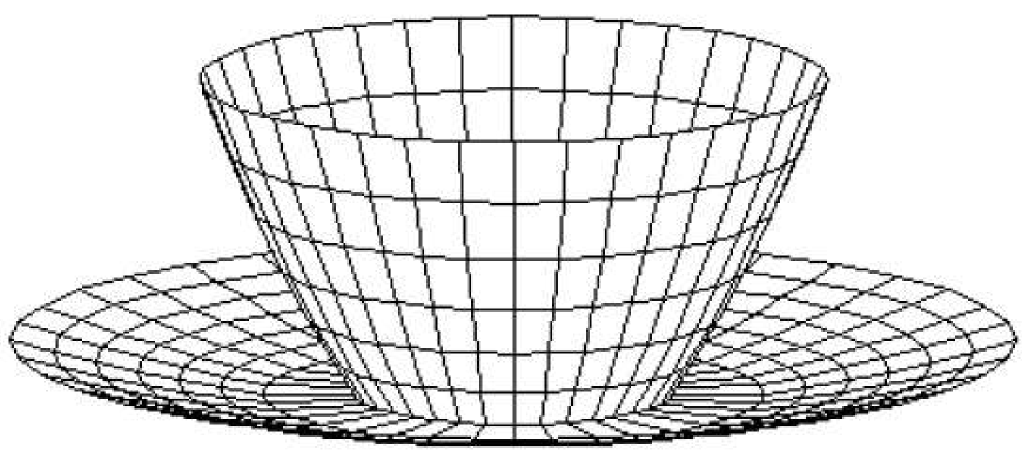
This method can be used to represent a broad class of solids/surfaces in graphics. A polygonal mesh can be rendered using hidden surface removal algorithms. The polygon mesh can be represented by three ways −

- Explicit representation
- Pointers to a vertex list
- Pointers to an edge list

"A Polygon mesh is a surface that is constructed out of a set of polygons that are joined together by common edges".

"A polygon mesh is a collection of vertices, edges and faces that defines the shape of a polyhedral object in 3D computer graphics and solid modeling".





Advantages

- It can be used to model almost any object.
- They are easy to represent as a collection of vertices.
- They are easy to transform.
- They are easy to draw on computer screen.

Disadvantages

- Curved surfaces can only be approximately described.
- It is difficult to simulate some type of objects like hair or liquid.

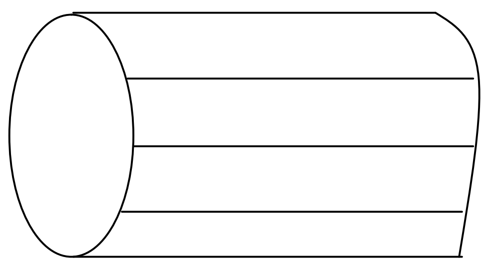
Polygon Surfaces

Objects are represented as a collection of surfaces. 3D object representation is divided into two categories.

- **Boundary Representations** B−repsB−reps − It describes a 3D object as a set of surfaces that separates the object interior from the environment.
- **Space–partitioning representations** − It is used to describe interior properties, by partitioning the spatial region containing an object into a set of small, non-overlapping, contiguous solids usually cubes.

The most commonly used boundary representation for a 3D graphics object is a set of surface polygons that enclose the object interior. Many graphics system use this method. Set of polygons are stored for object description. This simplifies and speeds up the surface rendering and display of object since all surfaces can be described with linear equations.

The polygon surfaces are common in design and solid-modeling applications, since their **wireframe display** can be done quickly to give general indication of surface structure. Then realistic scenes are produced by interpolating shading patterns across polygon surface to illuminate.

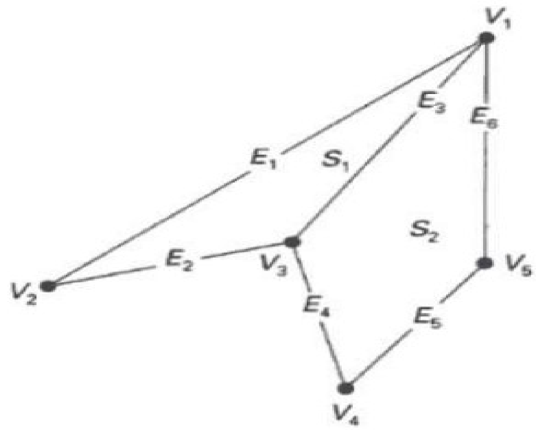


A 3D object represented by polygons

Polygon Tables

In this method, the surface is specified by the set of vertex coordinates and associated attributes. As shown in the following figure, there are five vertices, from v_1 to v_5.

- Each vertex stores x, y, and z coordinate information which is represented in the table as v_1: x_1, y_1, z_1.
- The Edge table is used to store the edge information of polygon. In the following figure, edge E_1 lies between vertex v_1 and v_2 which is represented in the table as E_1: v_1, v_2.
- Polygon surface table stores the number of surfaces present in the polygon. From the following figure, surface S_1 is covered by edges E_1, E_2 and E_3 which can be represented in the polygon surface table as S_1: E_1, E_2, and E_3.



VERTEX TABLE	
V_1:	x_1, y_1, z_1
V_2:	x_2, y_2, z_2
V_3:	x_3, y_3, z_3
V_4:	x_4, y_4, z_4
V_5:	x_5, y_5, z_5

EDGE TABLE	
E_1:	V_1, V_2
E_2:	V_2, V_3
E_3:	V_3, V_1
E_4:	V_3, V_4
E_5:	V_4, V_5
E_6:	V_5, V_1

POLYGON-SURFACE TABLE	
S_1:	E_1, E_2, E_3
S_2:	E_3, E_4, E_5, E_6

Plane Equations

The equation for plane surface can be expressed as −

$$Ax + By + Cz + D = 0$$

Where x, y, z is any point on the plane, and the coefficients A, B, C, and D are constants describing the spatial properties of the plane. We can obtain the values of A, B, C, and D by solving a set of three plane equations using the coordinate values for three non collinear points in the plane. Let us assume that three vertices of the plane are (x_1, y_1, z_1), (x_2, y_2, z_2) and (x_3, y_3, z_3).

Let us solve the following simultaneous equations for ratios A/D, B/D, and C/D. You get the values of A, B, C, and D.

$$A/D \; x_1 + B/D \; y_1 + C/D \; z_1 = -1$$

$$A/D \; x_2 + B/D \; y_2 + C/D \; z_2 = -1$$

$$A/D \; x_3 + B/D \; y_3 + C/D \; z_3 = -1$$

To obtain the above equations in determinant form, apply Cramer's rule to the above equations.

$$A = \begin{bmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{bmatrix} B = \begin{bmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{bmatrix} C = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix} D = -\begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}$$

For any point x, y, z with parameters A, B, C, and D, we can say that −

- $Ax + By + Cz + D \neq 0$ means the point is not on the plane.
- $Ax + By + Cz + D < 0$ means the point is inside the surface.
- $Ax + By + Cz + D > 0$ means the point is outside the surface.

Curve and Representation

WHAT IS CURVE :- There are lots of definitions for Curve but we will focus on 2 main definitions for our understanding.

DEF I :- When sets of points infinite or finite are Joined Continous then what we get is Called Curve.

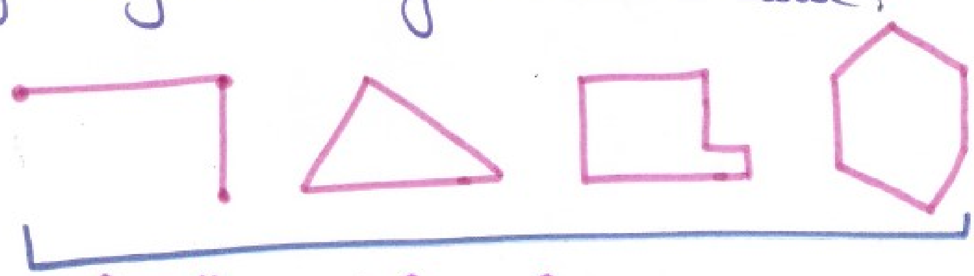
DEF 2- When we start from a point for drawing a geometrical figure and end at some other point without any GAP, so what we get is called CURVE.

One Question comes in mind that as per definition is LINE ALSO A CURVE?

Curve ?

YES, Mathematically a line is also Curve.

IF LINE is a Curve then all the geometrical figures generated by Line also a Curve?



Are they all Curve ?

As the Mathematics Says all the above figures are Curve.

But we focus here on other definition as well which Says

In Mathematics, a Curve is generally speaking, an Object Similar to a line but that need not to be Straight, Thus, a Curve is generalization of a line, in that it may be Curved (bend, Smoothness).

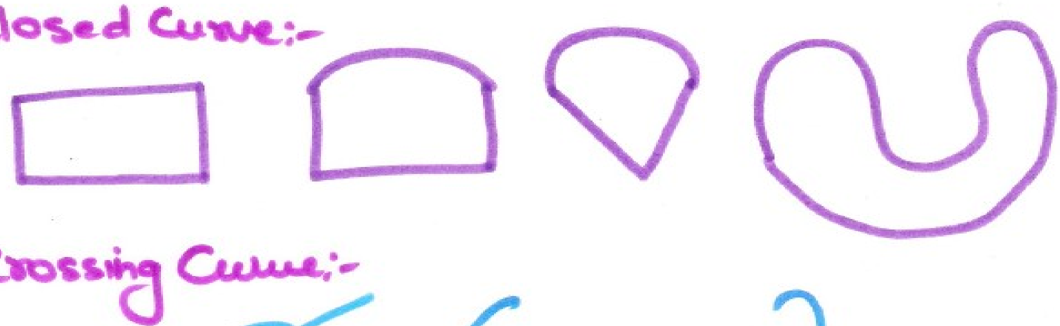
This Circle is Curve

but this is Not as there is GAP.

There are many types of Curves like:-

Open Curve:-

Closed Curve:-



Crossing Curve:-

Representing Curves :-

In Computer Graphics we daily need to draw or design different types of objects which are not flat but have bends and deviations and most importantly Smoothness.

Like Human face, Automobiles designs and many more.

So Computer need to Calculate or Compute all Curves so they can provide the Smoothness in Curve.

We Can represent basically Curves by 3 mathematical function

Curve Representation

- Explicit Fn for Curve
- Implicit fn
- Parametric Curve fn.

Explicit Representation of Curves :-

→ In this the dependent Variable has been given "Explicitly" in terms of the independent Variable. denoted as

$y = F(x)$ **Example :-** $y = ax^n + bx \ldots$

$y = 5x^3 + 2x + 1$

or $5x^2 + x$

Like a line $y = mx + c$

→ Explicit representation is Single valued for each Value of x only a Single value of y is Computed.

Implicit Representation of Curves:-

In this dependent variable is not expressed in terms of Some independent variables.

$$f(x,y) = 0 \qquad x^2+y^2-1=0$$
$$y^4+x^3+18=0$$

It can represent multivalued Curves (Multiple y values for an x value) $x^2+y^2-R^2=0$ Circle.

Although you can convert a implicit f^n into Explicit f^n but generally it should not be done. b/c.

The new explicit function becomes very complex and Some times also gives two different function branch.

for example :- If we convert implicit Curve $x^2+y^2+1=0$ to explicit Curve it will give us

$$y = \pm\sqrt{1-x^2}$$

Now new explicit f^n become very Complex and Some times it gives us 2 branches.

here y has 2 branch One is +ve & Second is -ve.

Parametric Curves :-

→ Most of the Curve representation's follow the parametric form.

→ Curves having parametric form are called parametric Curves.

→ There are many Curves which we Cannot write down as a Single equation in terms of only x and y.

→ Instead of defining y in terms of x ($y = f(x)$) or x in terms of y ($x = h(y)$) we defining both x and y in terms of a third variable called a Parameter

$$x = f_x(u) \qquad u \text{ is parameter.}$$
$$y = f_y(u)$$

Like line Parametric equation is

$$x = (1-u)x_0 + ux_1$$
$$y = (1-u)\cancel{y_0} + uy_1$$

Parametric Curve

The Parametric representation for Curves is :-

$$x = x(t)$$
$$y = y(t)$$
$$z = z(t)$$

A Curve is approximated by a piecewise polynomial Curve instead of piece linear Curve

Piecewise Linear Curve

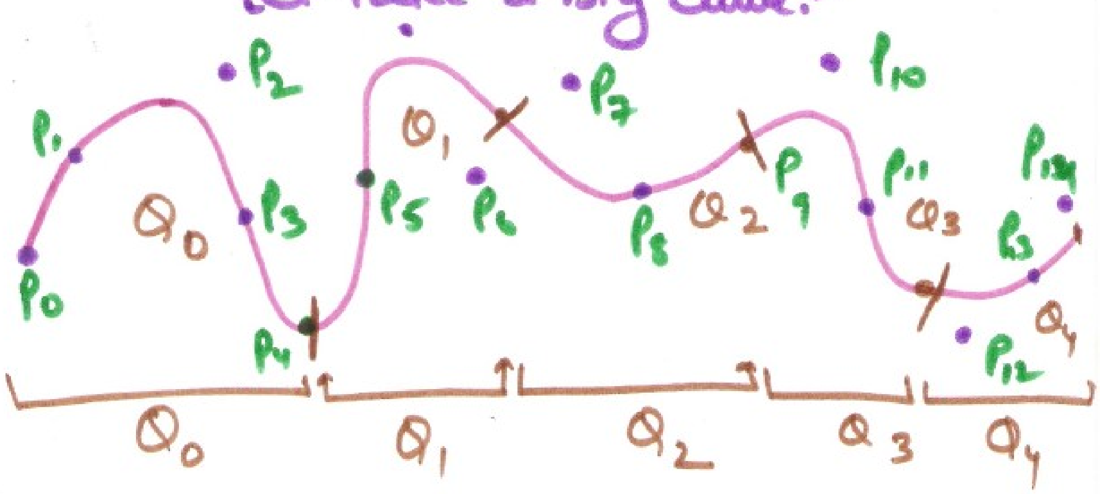
by Polylines & using Linear Equation

Piecewise polynomial Curve.

Represented by Polynomial Equation.

For drawing Curve we need to specify some points through which it may or maynot completely Follow:-

let take a big Curve:-



Q_{01}, Q_1, Q_2 Q_3 & Q_4 are the sections or Segment of big curve. & P's are Sample or Control Points

Each segment Q of the overall curves is given by three 3 Functions x, y, z which are Cubic polynomials in the Parameter t or u.

***

Cubic means here is that the polynomial Eq. Which is used to represent the curve is has degree of 3

The Cubic polynomials that define a curve Segment

$$Q(t) = [x(t) \quad y(t) \quad z(t)]$$

$$\left.\begin{aligned} x(t) &= a_x t^3 + b_x t^2 + c_x t + d_x \\ y(t) &= a_y t^3 + b_y t^2 + c_y t + d_y \\ z(t) &= a_z t^3 + b_z t^2 + c_z t \cdot d_z \end{aligned}\right\} \quad (1)$$

$$0 \leq t \leq 1$$

where $T = [t^3 \; t^2 \; t \; 1]$

The Coefficient matrix is defined as

$$C = \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{bmatrix} - (2)$$

So we can Rewrite equ. 1 as

$$Q(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix} = T \cdot C \quad \text{--- (3)}$$

In General:-

In this C can be further be divided

$$C = M \cdot G$$

where $M = \begin{bmatrix} m_g \end{bmatrix}_{4 \times 4}$ & $G = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 \end{bmatrix}^t$

M is a 4x4 basis matrix and G is a four element Column Vector of geometric Constants. Called the geometric Vector.

So $Q(t) = T \cdot M \cdot G$.

The Curve is a weighted Sum of the elements of the geometry matrix.

The weights are each Cubic polynomials of t, and are called the blending functions:-

$$B = T \cdot M$$

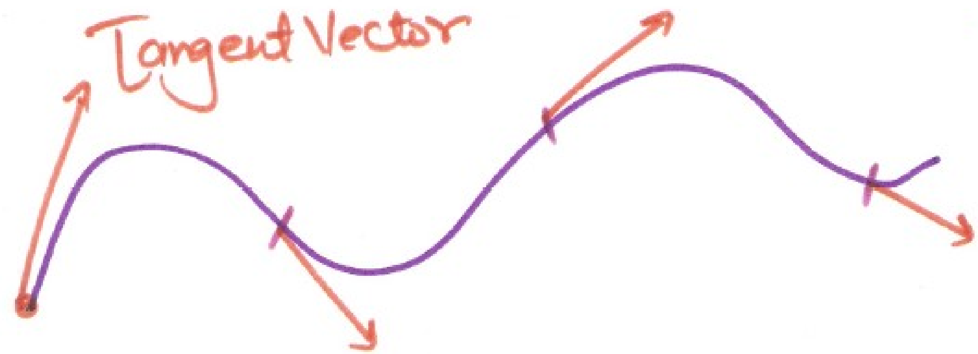
The parametric tangent-Vector to the Curve is

$$\frac{d\,Q(t)}{d(t)} = Q'(t) = \begin{bmatrix} \frac{d\,x(t)}{dt} & \frac{d\,y(t)}{dt} & \frac{d\,z(t)}{dt} \end{bmatrix}$$

$$= \frac{d\,T \cdot C}{dt}$$

$$= \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix} \cdot C$$

$$= \begin{bmatrix} 3a_x t^2 + 2b_x t + c_x & 3a_y t^2 + 2b_y t + c_y & 3a_z t^3 + 2b_z t + c_z \end{bmatrix}$$



Parametric and Geometric Continuity

A big Question Comes in mind when we join 2 piecewise polynomial parametric Curve that how to specify the smoothness of the Curve.

There are two approaches which determine the Smoothness of Curve i.e Curve Continuity:-

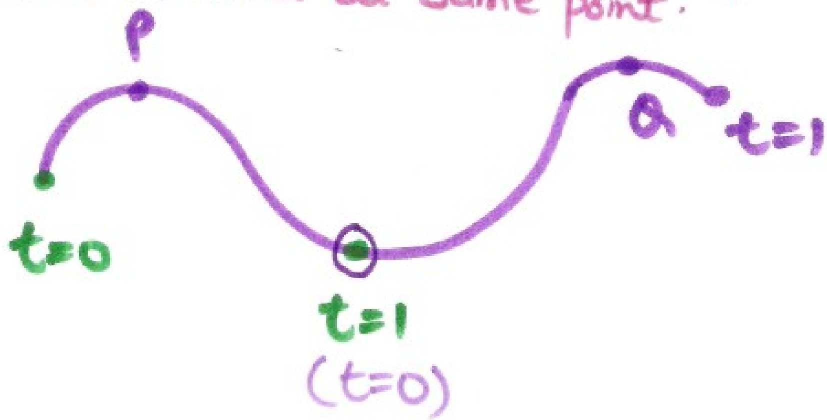
- Parametric Continuity Conditions — C
- Geometrical Continuity Conditions — G

PARAMETRIC CONTINUITY

- Parametric Continuity deals in parametric equations associated to piecewise parametric polynomial Curve. not the Shape or appearance of the Curve.

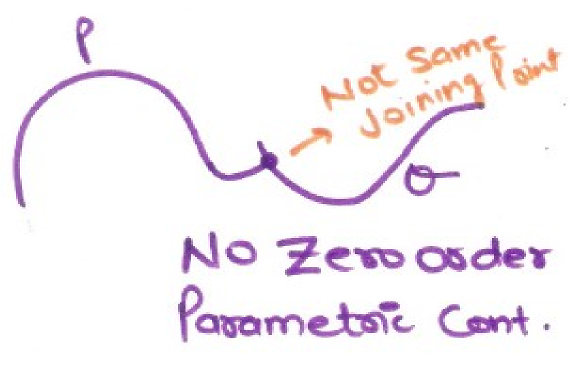
→ **Zero Order Parametric Continuity :- C^0**

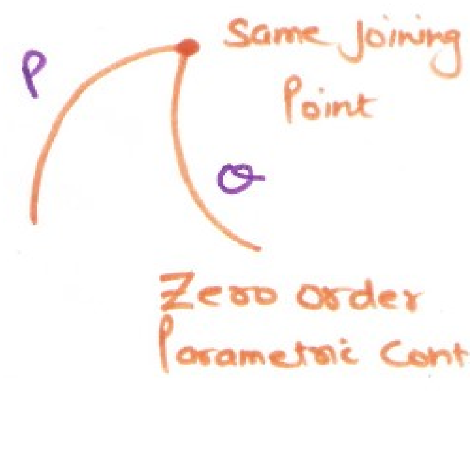
C^0 Continuity means that 2 piece of Curves are joined or meet at Same point.



There are two piece of Curve P & Q

IF $P(t=1) = Q(t=0)$ ⇒ Zero order Parametric Continuity.



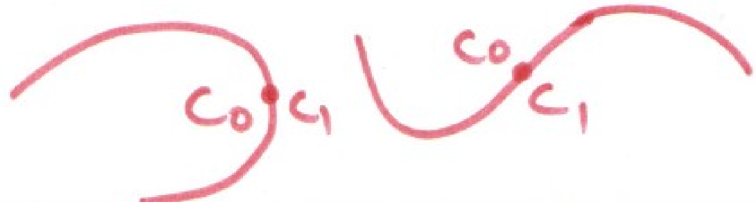


First Order Parametric Continuity (C^1) :-

In first Order Parametric Continuity C' means that first Parametric derivatives of the coordinate F^n for 2 Successive Curve Sections are equal at the joining point. C1 → First Derivatives are equal.

$$P'(t=1) = Q'(t=0)$$

P' & Q' are first order Derivative.



Second Order Parametric Continuity C^2 :-

It means both first and Second derivative of 2 Curve Section are Same at the intersection point

$$P''(t=1) = Q''(t=0)$$

GEOMETRIC CONTINUITY CONDITION :-

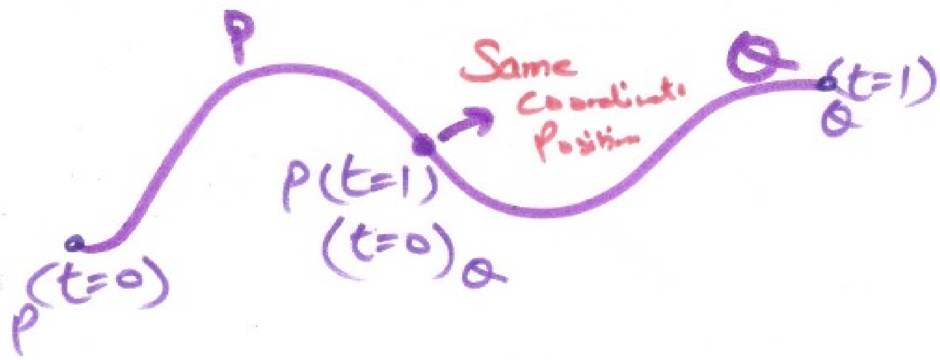
Geometric Continuity refers to the way that a Curve or Surface looks (unit tangent or Curvature vector Continuity.)

Parametric Continuity implies geometric Continuity and Vice Versa. however exception do exist.

Zero order geometric continuity G^0

It is same as C^0 → Zero order parametric Continuity

It means two curves sections must have the Same Coordinate position at the boundary point.



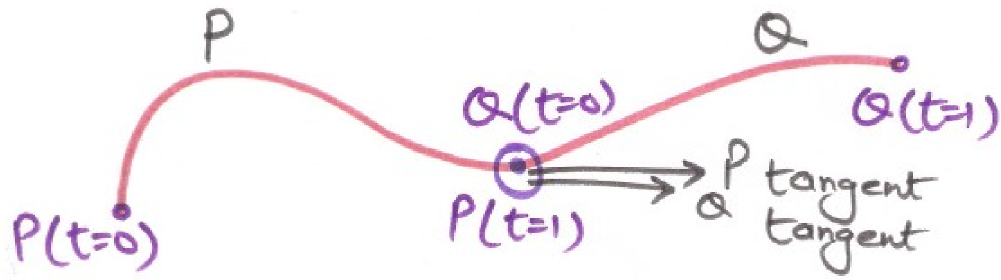
P & Q are two segments of Curves.

$$\boxed{P(t=1) = Q(t=0)}$$

First Order Geometric Continuity :- G^1

Geometric first order Continuity means that the Parametric first derivative are proportional at the intersection of 2 Successive sections.

If P and Q are two piece of Curves, then $P'(t=1)$ & $Q'(t=0)$ must have Same direction of tangent Vector but not necessary the Same magnitude



Here tangent vector has same direction but their magnitude are May or may not same (length)

$$\underbrace{c1'(1) = (a,b,c) \;\&\; c2'(0) = (k*a,\, k*b,\, k*c)}_{\text{Proportional.}}$$

$P'(t) \neq Q'(t)$ $\qquad$ $G^1 \not\Rightarrow C^1$ May or May not

$C^1 \Rightarrow G^1$

Second Order Geometrical Continuity :-

Both first & Second derivative are propotional at their boundary point & tangent vector direction is Same & Magnitude may or may not same.

$$G^2 = C^2 \quad , \quad C^2 \not\Rightarrow G^2$$

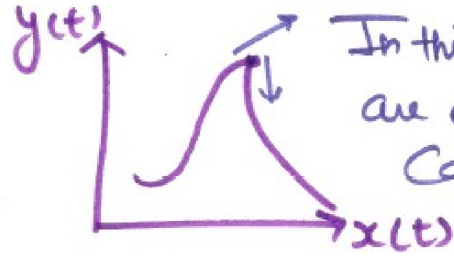
The tangent vector $Q'(t)$ is the velocity of a point on the curve with respect to parameter t.

Similarly $Q''(t)$ is the acceleration

In general C^1 continuity $\Rightarrow G^1$ but converse is not true generally

Join point with G^1 continuity will appear just as Smooth as those with C^1 continuity.

Special Case :- C^1 Continuity does not imply G^1 Continuity when Segments tangent vector are $[0 \quad 0 \quad 0]$ at the Join point.



In this Case the tangent vectors are equal but there directions Can be different

Spline curve

Spline :- Spline is a flexible strip which was long ago used for designing the Ships.

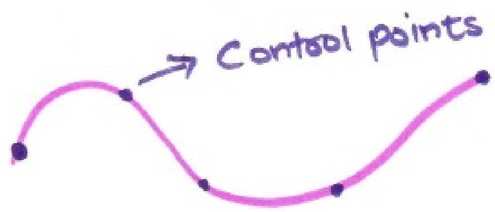
Spline Curve :- A Spline Curve is mathematical representation for which it is easy to build an interface that will allow a user to design and Control the Shape of Complex Curves & Surfaces.

Spline Curve mathematically described with a piecewise Cubic polynomial function whose first & Second derivatives are continuous across the Various Curve Section. C^1 & C^2 Continuity.

Control points:- We specify a spline Curve by giving a set of **Coordinate positions,** called Control points. which indicates the general shape of the Curve. These Control points are then fitted with piecewise Continuous parametric polynomial functions in on the 2 ways:-

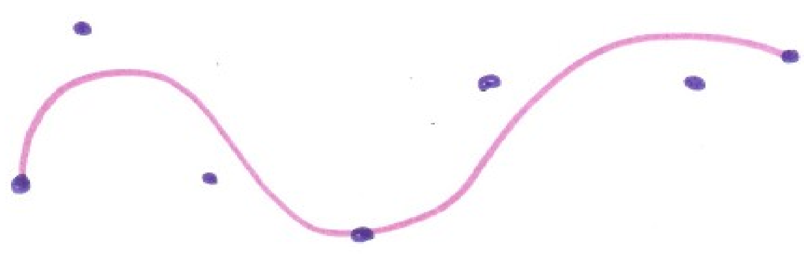
Interpolate or Interpolation Spline:-

When polynomial Sections are fitted so that the Curve passes through all control points, then the resulting Curve is said to be Interpolate the set of Control points.



Approximate or Approximation Spline:-

When the polynomials are fitted to the path which is not necessarily passing through all Control points, the resulting Curve is Said to approximate the set of Control points.



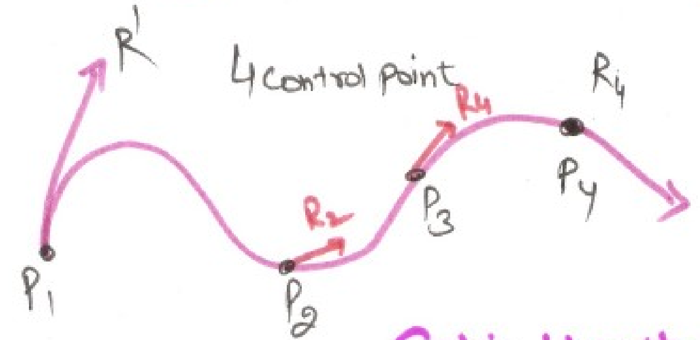
Approximation Spline

Approximation Curves are Commonly used as design tools to Structure object Surface.

Hermite Spline Curve

Hermite spline Curve is our Interpolation spline Curve.

The Hermite form of the Cubic polynomial Curve Segment is determined by Constraints on the end points P_1 & P_4 and the Tangent Vectors at the end points R_1 & R_4



Cubic Hermite Spline Curve

It has Local Control over the Curve.

Let $Q(t)$ is the Curve where $t \in [0,1]$

$Q(t) = (x(t) \ \ y(t) \ \ z(t))$, $t \in [0,1)$ where all points Satisfy Cubic parametricity.

As we know the general Curve Equation

$P(t) = at^3 + bt^2 + ct + d \quad 0 \leq t \leq 1$ t is parameter

So $x(t) = a_x t^3 + b_x t^2 + c_x t + d_x \cdot 1$

$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y \cdot 1$

$z(t) = a_z t^3 + b_z t^2 \ \ c_z t + d_z \cdot 1$

$$\begin{bmatrix} x(t) \\ y(t) \\ Z(t) \end{bmatrix} = \underbrace{\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}}_{T} \cdot \underbrace{\begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{bmatrix}}_{C}$$

$$Q(t) = T \cdot C$$

but As per Rule for Specifying a spline curve we need a basis f^n matrix So

$$Q(t) = T \cdot \underset{\text{basis matrix}}{M} \cdot \underset{\text{Geometry Vector}}{G} \quad \text{where } [C = M \cdot G]$$

Let for Hermite we may write it as

$$Q(t) = T \cdot M_H \cdot G_H.$$

M_H = Hermite basis matrix

G_H = Hermite Geometry Matrix vector

$$= [t^3\ t^2\ t\ 1] \cdot M_H \cdot G_H$$

$$Q_x(t)\Big|_{t=0} = P_1(t) = [0\ 0\ 0\ 1] \cdot M_H \cdot G_{H_x} \quad — (A)$$

$$Q_x(t)\Big|_{t=1} = P_4(t) = [1\ 1\ 1\ 1] \cdot M_H \cdot G_{H_x} \quad — (B)$$

$$Q'_{(x)}t\Big|_{t=0} = R_1(t)\ [0\ 0\ 1\ 0] \cdot M_H \cdot G_{H_x} \quad — (C)$$

$$Q'_{(x)}t\Big|_{t=1} = R_4(t)\ [3\ 2\ 1\ 0]\ M_H \cdot G_{H_x} \quad — (D)$$

From A, B, C, D Equations

$$\begin{bmatrix} P_1(x) \\ P_4(x) \\ R_1(x) \\ R_4(x) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \underbrace{M_H \cdot G_{Hx}}_{C}$$

$$M_H \cdot G_{Hx} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} P_{1_x} \\ P_{4_x} \\ R_{1_x} \\ R_{4_x} \end{bmatrix}$$

So

$$M_H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \text{ \&}$$

$$G_{Hx} = \begin{bmatrix} P_{x_1} \\ P_{x_2} \\ R_{1x} \\ R_{4x} \end{bmatrix}$$

$$Q(t) = T \cdot M_H \cdot G_H$$

$$\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_1 \\ P_4 \\ R_1 \\ R_4 \end{pmatrix}$$

$$Q(t) = (2t^3 - 3t^2 + 1)P_1 + (-2t^3 + 3t^2)P_4 + (t^3 - 2t^2 + t)R_1 + (t^3 - t^2)R_4$$

$$= P_1 H_0(t) + P_4 H_1(t) + R_1 H_2(t) + R_4 H_3(t)$$

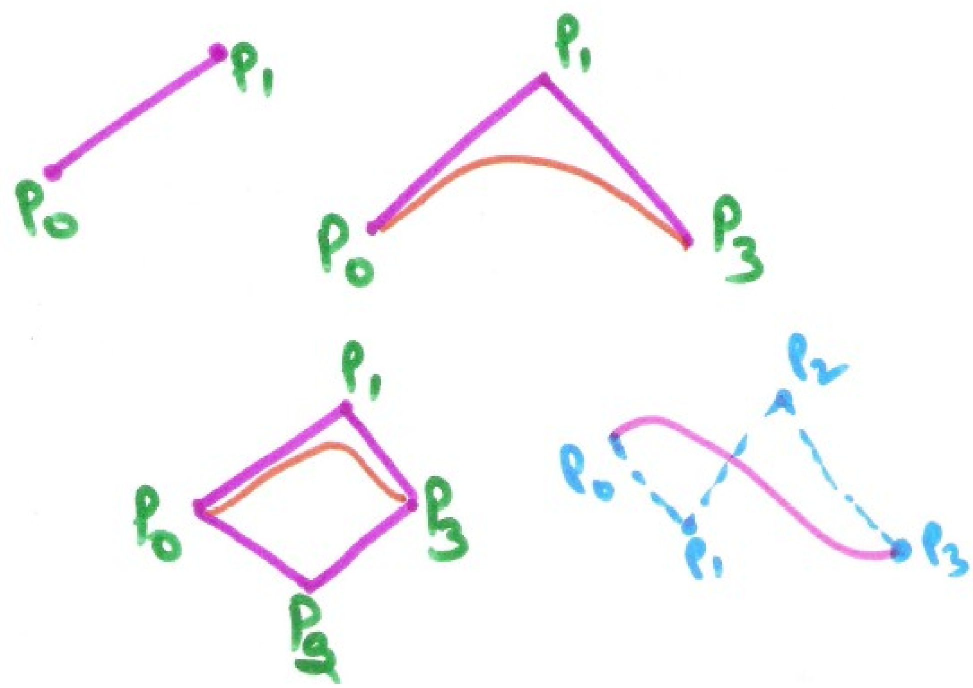
$H_0(t)$, $H_1(t)$, $H_2(t)$, $H_3(t)$ Hermite blending fn.

Bezier Curve

→ Bezier Curve is another approach for the construction of the Curve.

→ It is approximate spline Curve.

→ Instead of endspoints and tangents, we have four Control points in the Case of Cubic Bezier Curve



→ Bezier splines are widely used in various CAD System, COREL DRAW Packages and many more Graphic packages.

→ As with Splines, a bezier Curve can be specified with boundary Conditions with a characterizing matrix or with blending f^n. For general bezier Curves, the blending function specification is most Convenient.

Let Suppose we are given (n+1) control points positions. then $P_i = (x_i, y_i, z_i)$ with i varying from 0 to n.

These coordinate points can be blended to produce the following position vector $P(u)$, which Describes the path of an approximation So Bezier polynomial fn b/w P_0 to P_n is

$$\boxed{P(u) = \sum_{i=0}^{n} P_i \, B_{i,n}(u)} \quad 0 \leq u \leq 1$$

P_i Control Points

$B_{i,n}$ or $BEZ_{i,n}$ is Bezier fn or Bernstein Polynomials.

→ The Bernstein polynomial or the Bezier fn is very important fn which will dictate the smoothness of this Curve & the weight will be dictated by boundary Conditions.

$$BEZ_{i,n}(u) = {}^{n}C_i \cdot u^i (1-u)^{n-i}$$

where

$${}^{n}C_i = \frac{n!}{i!(n-i)!}$$ [Binomial Coefficient]

For Individual Coordinates

$$x(u) = \sum_{i=0}^{n} x_i \, BEZ_{i,n}(u)$$

$$y(u) = \sum_{i=0}^{n} y_i \, BEZ_{i,n}(u)$$

$$z(u) = \sum_{i=0}^{n} z_i \, BEZ_{i,n}(u)$$

Bezier curve for

3 Points

$$Q(u) = P_0 B_{0,2}(u) + P_1 B_{1,2}(u) + P_2 B_{2,2}(u)$$

Now Calculate $B_{0,2}$

$$B_{0,2}(u) = {}^{2}C_0 u^0 (1-u)^{2-0}$$

$$= \frac{2!}{0!\,2-0!} (1-u)^2 \cdot 1$$

$$= \frac{2 \times 1}{2!} (1-u)^2$$

$$= 1 \cdot (1-u)^2 \Rightarrow (1-u)^2$$

Now $B_{1,2}(u)$ in Same way

$$= 2(1-u)u$$

$$B_{2,2}(u) = u^2$$

Now using in main Equation

$$Q(u) = (1-u)^2 P_0 + 2 \cdot (1-u) \cdot u \, P_1 + u^2 P_2$$

$$x(u) = (1-u)^2 x_0 + 2 \cdot (1-u) u \, x_1 + u^2 x_2$$

4 Points

$$Q(u) = P_0 B_{0,3}(u) + P_1 B_{1,3}(u) + P_2 B_{2,3}(u) + P_3 B_{3,3}(u)$$

Now we will Calculate $B_{0,3}(u)$, $B_{1,3}(u)$ - - as we have Calculated & get

$$B_{0,3}(u) = (1-u)^3$$

$$B_{1,3}(u) = 3u(1-u)^2$$

$$B_{2,3}(u) = 3u^2(1-u)$$

$$B_{3,3}(u) = u^3$$

Now Putting them in main Equation (1)

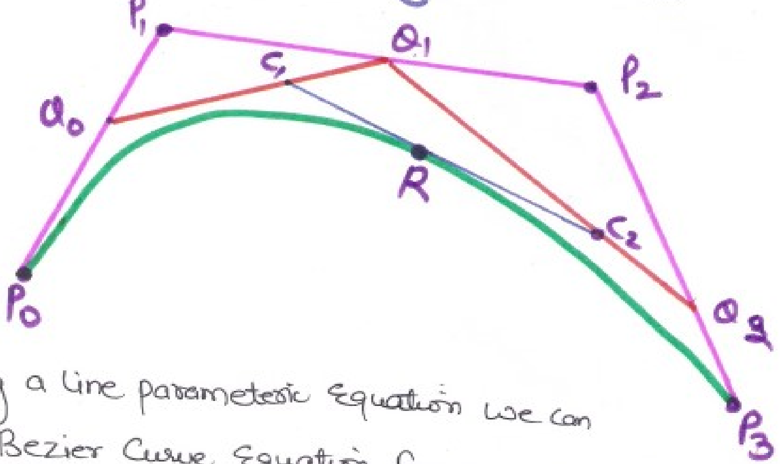
$$Q(u) = P_0 (1-u)^3 + P_1 u (1-u)^2 + P_2 \cdot 3u^2 (1-u) + u^3 \cdot P_4$$

$$x(u) = (1-u)^3 x_0 + u \cdot (1-u)^2 \cdot x_1 + 3u^2(1-u) x_2 + x_3 \cdot u^3$$

$y(u) =$ In Same way

$z(u) =$

Let See the main way of Calculating the Bezier Curve or where from we get Bezier fn:-



By using a line parameteric Equation we can drive Bezier Curve Equation for any no. of Control points:-

$$Q_0 = (1-u)P_0 + uP_1 \qquad [Q_0 \text{ Point on } P_0 \rightarrow P_1$$
$$Q_1 = (1-u)P_1 + uP_2 \qquad Q_1 \text{ Point on } P_1 \rightarrow P_2$$
$$Q_2 = (1-u)P_2 + uP_3 \qquad Q_2 \text{ Point on } P_2 \rightarrow P_3]$$

$$C_1 = (1-u)Q_0 + u\cdot Q_1 \quad [C_1 \text{ Point on } Q_0 \rightarrow Q_1$$
$$C_2 = (1-u)Q_1 + u\cdot Q_2 \quad C_2 \text{ Point on } Q_1 \rightarrow Q_2]$$

$$R = (1-u)C_1 + u\cdot C_2 \quad [R \text{ point on } C_1 \rightarrow C_2]$$

Now we will use C_1, C_2, Q_0, Q_1, Q_2 Values in R:-

$$R(u) = (1-u)C_1 + u\cdot C_2$$
$$= (1-u)\left[(1-u)Q_0 + u\cdot Q_1\right] + u\cdot\left[(1-u)Q_0 + u\cdot Q_2\right]$$
$$= (1-u)^2 Q_0 + \underbrace{(1-u)\cdot u\cdot Q_1 + (1-u)\cdot u\cdot Q_1} + u^2\cdot Q_2$$
$$= (1-u)^2\left[(1-u)P_0 + uP_1\right] + 2(1-u)\cdot u\cdot Q_1 + u^2\left((1-u)P_2 + uP_3\right)$$
$$= (1-u)^3 P_0 + (1-u)^2\cdot u\cdot P_1 + 2(1-u)\cdot u\cdot\left[(1-u)P_1 + u\cdot P_2\right] + u^2\left((1-u)P_2 + u\cdot P_3\right)]$$

$$= (1-u)^3 P_0 + \underline{(1-u)^2\cdot u}\cdot P_1 + \underline{2(1-u)^2\cdot u}\cdot P_1 + \underline{2(u-1)\cdot u^2} P_2$$
$$+ \underline{(1-u)\cdot u^2\cdot P_2} + u^3\cdot P_3$$

$= (1-u)^3 P_0 + \underline{(1-u)^2 \cdot u \cdot P_1} + \underline{2(1-u)^2 \cdot u \cdot P_1} + \underline{2(u-1) \cdot u^2 P_2}$
$+ \underline{(1-u) \cdot u^2 \cdot P_2} + u^3 \cdot P_3$

$= (1-u)^3 P_0 + 3(1-u)^2 \cdot u \cdot P_1 + 3(1-u) \cdot u^2 \cdot P_2 + u^3 \cdot P_3$

For x, y, z coordinate

$(1-u)^3 x_0 + 3(1-u)^2 \cdot u \cdot x_1 + 3(1-u) \cdot u^2 \cdot x_2$
$+ u^3 \cdot x_3$

Same equation which we get from Bernstein Polynomial form:—

Properties of Bezier Curves:-

(i) A very useful property of Bezier Curve is that it always Passes through the first and last Control points.

$$P(0) = P_0$$
$$P(1) = P_n$$

(ii) They generally follow the shape of the Control polygon Which Consists of the Segments joining the Control points.

(iii) The Curve is contained within Convex hull of defining Polygon.

(iv) The degree of the polynomial defining the Curve Segment is one less than the number of defining Control polygon points. For 4 Control points the degree of polynomial is 3. i.e Cubic Bezier Curve.

(v) It is quite easy to implement.

Drawback:-

→ The degree of Bezier Curve depends on number of Control Points

2→ Bezier Curve exhibit global Control property means moving a Control point alters the shape of the whole Curve.

B-Spline Curve

Properties of B-Spline Curve :-

1) B-Spline basis is non-global (LOCAL) effect.
In this each Control point affects the shape of the Curve Only over range of parameter Values where its associated basis fn is non zero.

We have some limitations in Bezier Curve like →

1) The Bezier Curve produced by Bernstein basis fn has limited flexibility.
Numbers of Control points decides the degree of the Polynomial Curve. Ex:- 4 Control points results a Cubic polynomial Curve.
So only one way to reduce the degree of the Curve is to reduce the no. of Controlpoints and vice versa.

2) The Second limitation is that the Value of the blending fn is non-zero for all parameter Values over the entire Curve.
Due to this Change in one Vertex, Changes the entire Curve and this eliminates the ability to produce a local Change with in a Curve.

So B-Spline Curve — Basis-Spline Curve is Solution of this limitations of Bezier Curve.

Properties of B-Spline Curve :-

1) B-Spline basis is non-global (LOCAL) effect.
In this each Control point affects the shape of the Curve Only over range of parameter Values where its associated basis fn is non zero.

2) B-Spline Curve made up of n+1 Control point

3) B-Spline Curve let us specify the order of basis (k) fn and the degree of the resulting Curve is independent on the no. of vertices.

4) It is possible to change the degree of the resulting Curve with out changing the no. of control points.

5) B-Spline Can be used to define both open & close Curves.

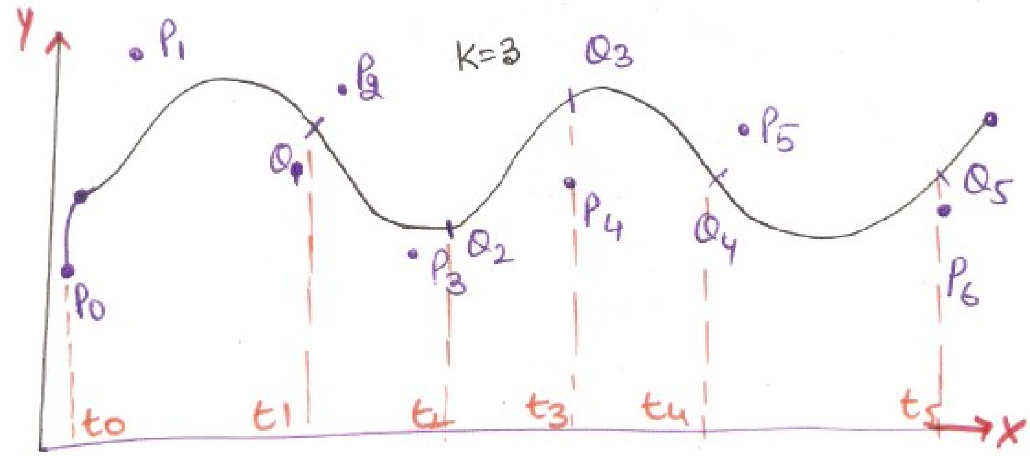
6) Curve generally follows the shape of defining polygon
If we have order k = 4 then degree will be 3 $P(k) = x^3$

7) The Curve line with in the Convex hull of its defining Polygon.

In B-Spline we Segment out the whole Curve which is decided by the order (K). **by Formula 'n-k+2'**

for Example :-

If we have 7 control points and order of Curve k = 3 then n = 6
And this B-Spline Curve has Segments
$6-3+2 = 5$



Five segments Q_1, Q_2, Q_3, Q_4, Q_5

Segment	Control points	Parameter
Q_1	$P_0\ P_1\ P_2$	$t_0 = 0,\ t_1 = 1$
Q_2	$P_1\ P_2\ P_3$	$t_1 = 1,\ t_2 = 2$
Q_3	$P_2\ P_3\ P_4$	$t_2 = 2,\ t_3 = 3$
Q_4	$P_3\ P_4\ P_5$	$t_3 = 3 \quad t_4 = 4$
Q_5	$P_4\ P_5\ P_6$	$t_4 = 4 \quad t_5 = 5$

There will be a join point or knot between Q_{i-1} & Q_i for $i \geq 3$ at the parameter value t_i know as KNOT VALUE [X].

If $P(u)$ be the position vectors along the curve as a fn of the parameter u, a B-spline curve is given by

$$P(u) = \sum_{i=0}^{n} P_i\, N_{i,K}(u) \qquad 0 \leq u \leq n-k+2$$

$N_{i,k}(u)$ is B-spline basis fn

$$N_{i,k}(u) = \frac{(u - x_i)\, N_{i,k-1}(u)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - u)\, N_{i+1,k-1}(u)}{x_{i+k} - x_{i+1}}$$

The values of X_i are the elements of a knot vector Satisfying the relation $X_i \leq X_{i+1}$.

The parameter u varies from 0 to n-k+2 along the P(u)

So there are some conditions for finding the KNOT VALUES [X]

$X_i \ (0 \leq i \leq n+k) \rightarrow$ Knot Values

$X_i = 0$ if $i < k$

$X_i = i - k + 1$ if $k \leq i \leq n$

$X_i = n - k + 2$ if $i > n$

So as B-Spline Curve has Recursive Equ. so we stop at

$N_{i,k}(u) = 1$ if $x_i \leq u \, x_{i+1}$

$= 0$ otherwise

Example :-

$n=5, \ k=3$

then $X_i \ (0 \leq i \leq 8)$ knot values

$$X_i \{ \underset{X_0}{0}, \underset{X_1}{0}, \underset{X_2}{0}, \underset{X_3}{1}, 2, 3, 4, 4, \underset{X_8}{4} \}$$

$$\underbrace{N_{0,3}(u) = (1-u)^2 \cdot N_{2,1}(u)}_{\text{After calculation.}}$$

→ when i=0, k=3 so i<k is true
$X_0 = 0$

→ i=1, k=3 $X_1 = 0$

i=2, k=3 $X_2 = 0$

i=3, k=3 $X_3 = i-k+1 = 3-3+1$
$X_3 = 1$

i=4, k=3 $X_4 = i-k+1 \Rightarrow 4-3+1$
$X_4 = 2$

i=8 k=3 $X_8 = i > n \quad n-k+2 = 5-3+2 = 4$
$X_8 = 4$

In this way we will calculate