Homogenous Coordinates

Homogenous coordinates defines a point in a plane using three coordinates instead of two.

This states that for a point P with coordinates (x, y), there exists a homogenous point i.e. (xt, yt, t) such that $X = x/t$ and $Y = y/t$

For eg:- $P(3, 4)$ has homogenous coordinates $(6, 8, 2)$ as, $3 = 6/2$ and $4 = 8/2$

But we cannot say that $(6, 8, 2)$ is unique homogenous coordinates of $(3, 4)$ because $(12, 16, 4)$, $(15, 20, 5)$, $(300, 400, 100)$ are also the possible homogenous cordinates of $(3, 4)$.

In 3×3 matrix for homogenous co-ordinates, the translation equations are :-

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Similarly equations of rotation and scaling may be modified as :-

Rotation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Scaling

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Similarly for reflection :-

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where a, b, c and d depend upon choice of coordinates axes or daigonal axis as mirrol line.

Ques A Triangle defined by $\begin{bmatrix} 2 & 4 & 4 \\ 2 & 2 & 4 \end{bmatrix}$

find the xformed coordinates after:

i) 90° rotation about origin

ii) reflection about line $y=-x$

Sol 90° rotation about origin, the xformed coordinates is :-

$$\begin{bmatrix} \cos 90^\circ & -\sin 90^\circ & 0 \\ \sin 90 & \cos 90^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 4 & 4 \\ 2 & 2 & 4 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 2 & 4 & 4 \\ 2 & 2 & 4 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -2 & -2 & -4 \\ 2 & 4 & 4 \\ 1 & 1 & 1 \end{bmatrix}$$

Finally after reflection about line $y=-x$,

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2 & -2 & -4 \\ 2 & 4 & 4 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -2 & -4 & -4 \\ 2 & 2 & -4 \\ 1 & 1 & 1 \end{bmatrix}$$

Ques Translate the square whose coordinates are ABCD :- A (0,0) B(3,0) C (3,3), D(0,3) by 2 units in both the direction and then scale it by 1.5 units in x-direction and 0.5 units in y-direction.

$T_x = T_y = 2$

Given :-

$$\text{Translation} = \begin{bmatrix} 0 & 0 & 1 \\ 3 & 0 & 1 \\ 3 & 3 & 1 \\ 0 & 3 & 1 \end{bmatrix}_{4\times3} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 2 & 1 \end{bmatrix}_{3\times3}$$

$$= \begin{bmatrix} 2 & 2 & 1 \\ 5 & 2 & 1 \\ 5 & 5 & 1 \\ 2 & 5 & 1 \end{bmatrix}$$

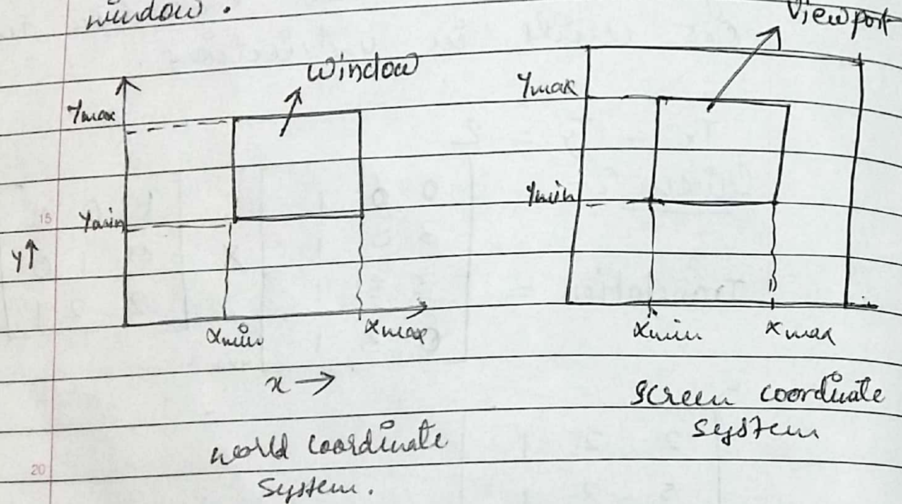
$$\text{Scaling} = \begin{bmatrix} 2 & 2 & 1 \\ 5 & 2 & 1 \\ 5 & 5 & 1 \\ 2 & 5 & 1 \end{bmatrix} \times \begin{bmatrix} 1.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{pmatrix} 3 & 1 & 1 \\ 7.5 & 1 & 1 \\ 7.5 & 2.5 & 1 \end{pmatrix}$$

Clipping.

Any procedure that identifies those portion of a picture that are either inside or outside of a specified region of space is referred to as clipping.

The region against which an object is to be clipped is called a clip window.



- If the object lies inside the window, then we can draw that object in the view port.
- If the object lies outside the window, then we dont draw the object in the viewport.
- While, if the object lies partially inside the window and partially outside the window then only those portion of the object is drawn which is inside the window. This procedure is called clipping.

Cohen Sutherland Point Clipping

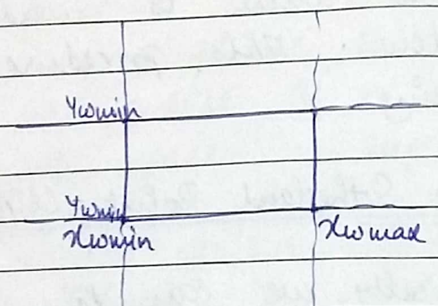
Basically we can take window of any shape, but let us assume the window to be in the shape of rectangle.

And, clipping as we know, that if object is inside window it is to be displayed and if it is outside window it is to be rejected.

Clipping - means discarding the portion that lies outside window.

i) Point clipping
ii) Line clipping
iii) Polygon clipping.

Point Clipping :- Here first we define the window coordinates then we find out whether the point coordinates lies inside window or outside window.



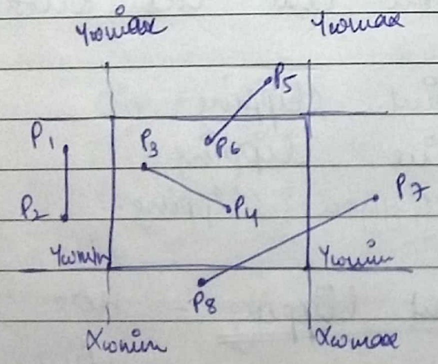
for a point (x, y)

$Xwmin \leq x \leq Xwmax$

$Ywmin \leq y \leq Ywmax$

for a point to be accepted or rejected, we have to check whether it is to be accepted or rejected.

Line Clipping



So, here we have to display the points that are inside the window and discard the points that lies outside the window.

① $P_1 P_2 \rightarrow$ discarded $\rightarrow$ (By visibility)

② $P_3 P_4 \rightarrow$ Accepted

③ $P_5 P_6 \rightarrow$ Clipping Required

④ $P_7 P_8 \rightarrow$ Clipping Required

- for condition ① and ② just apply point clipping identify both the coordinates and check if lies inside or outside.

- But for ③ and ④ we cannot apply conditions, as some portion of the line is to be clipped and some is to be displayed.

- for this, we have to apply line clipping algorithm called Cohen Sutherland line clipping Algorithm.

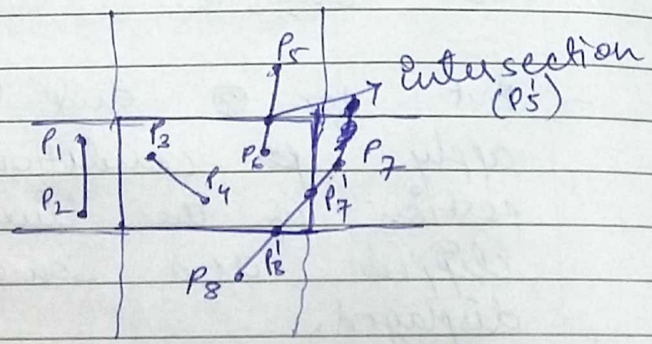
In this we apply region codes :-

Region code → 4 bit codes (ABRL).

1001	1000	1010
0001	0000	0010
0101	0100	0110

Identify the neighbours of window

Now consider the points of line again



Now consider P_1 & P_2
The region code here has non-zero values for bothe P_1 and P_2 i.e.

0001
0001 Perform AND operation
0001 (reject)

Again after AND we got a non zero value.
This non-zero value implies line is outside the window.

Consider Point P_3 & P_4
0000
0000 (AND)
0000 (Accept)

Consider Point P_5 & P_6
1000 (non-zero)
0000 AND
0000 (some portion line inside and some outside)

Clipping required (Partial acceptance)

here, we have to find the intersection point on the window.

Again consider P_5' and P_6
0000
0000 AND
0000 (inside window)

So, P_5 to P_5' is clipped.

Consider P_7 and P_8

0010 } non zero
0100 AND
0000 (zero)

clipping needed. (Partial acceptance)

~~P_7' 0010~~
~~P_7 0110 AND~~
~~0010 Reject~~

P_7' 0000
P_7 0010 AND
0000 (Partial)

P_7' 0000
P_8 0100 AND
0000 (Partial)

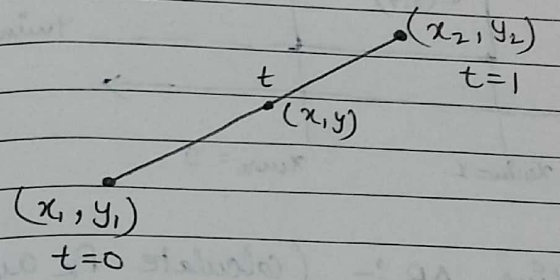
P_7' 0000
P_8' 0000 AND
0000 (accept)

P_8' 0000
P_8 0100 AND
0000 (partial)

Liang Barskey

Let us consider the parametric equation for line:-

$$x = x_1 + t\Delta x \quad \text{(i)}$$
$$y = y_1 + t\Delta y \quad \text{(ii)}$$
$$0 < t < 1$$



$$x = t \cdot x_2 + (1-t)x_1$$
$$x = t \cdot x_2 + x_1 - x_1 \cdot t$$
$$x = x_1 + t(x_2 - x_1)$$

Similarly for $y = y_1 + t(y_2 - y_1)$

here $\Delta x = x_2 - x_1$
$\Delta y = y_2 - y_1$

Here, in Liang Barskey this 't' intersection is only the intersection of line to the window. where window boundary are:-

X_{wmin} & Y_{wmin}
X_{wmax} & Y_{wmax}

Let us consider the point clipping algorithm. ~~where~~ where we use to check whether the points lies ~~between~~ inside the window ~~and~~ or outside the window.

So

$x_{wmin} \le x \le x_{wmax}$

$y_{wmin} \le y \le y_{wmax}$

Now write the above two conditions by taking values from equation (i) and (ii). we get :-

$x_{wmin} \le x_1 + t\Delta x \le x_{wmax}$

$y_{wmin} \le y_1 + t\Delta y \le y_{wmax}$

Simplyfying above conditions :-

$$\left.\begin{array}{l} x_1 + t\Delta x \ge x_{wmin} \\ x_1 + t\Delta x \le x_{wmax} \\ y_1 + t\Delta y \ge y_{wmin} \\ y_1 + t\Delta y \le y_{wmax} \end{array}\right\} \text{inequalities}$$

Deriving 4 inequalities

$t\Delta x \ge x_{wmin} - x_1$ — (1)

$t\Delta x \le x_{wmax} - x_1$ — (2)

$t\Delta y \ge y_{wmin} - y_1$ — (3)

$t\Delta y \le y_{max} - y_1$ — (4)

The above two condition should be of the form $\boxed{tP_K \le q_K}$ — (5)

$K = 1, 2, 3, 4$

For this change the eq (1) and (3) in the form of less than equalto ($\le$) by multiplying it with -1. we get :-

$-t\Delta x \le x_1 - x_{wmin}$

$t\Delta x \le x_{wmax} - x_1$

$-t\Delta y \le y_1 - y_{wmin}$

$t\Delta y \le y_{wmax} - y_1$

So we get :-

$P_1 = -\Delta x$	$q_1 = x_1 - x_{wmin}$
$P_2 = \Delta x$	$q_2 = x_{wmax} - x_1$
$P_3 = -\Delta y$	$q_3 = y_1 - y_{wmin}$
$P_4 = \Delta y$	$q_4 = y_{wmax} - y_1$

Based on these conditions only we need to find out whether the line is inside or outside the clipping window.

<u>Conditions :-</u>

1) If $P_K = 0$ (all P_1, P_2, P_3, P_4) then lines lies parallel to window.
(i.e. line may be inside, outside or within the boundary).

$p_k = 0$ &

Now, Corresponding q_k value :-

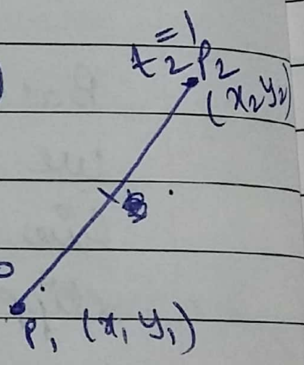
$q_k < 0$ line is outside
$q_k > 0$ line is inside / Partially inside.
$q_k = 0$ within the boundary partial / complete.

2) If $p_k < 0 \rightarrow t_1$ (find)
here the initial point lies outside the window.
we have to find the coordinates (x, y) of intersection.

$$t_1 = \max(0, q_k / p_k)$$

if

$$x = x_1 + t_1 \Delta x$$
$$y = y_1 + t_1 \Delta y$$



3) If $p_k > 0 \rightarrow t_2$ (find)
here the final point lies outside the window.
we have to find the intersecting coordinates (x, y)

$$t_2 = \min(1, q_k / p_k)$$
$$x = x_1 + t_2 \Delta x$$
$$y = y_1 + t_2 \Delta y$$

Liang Barskey Numerical

Ques Given :- $x_{wmin} = 5$ $\quad x_{wmax} = 9$

$y_{wmin} = 5$ $\quad y_{wmax} = 9$

$P_1 = (4, 12)$

$P_2 = (8, 8)$

find the values of P_K and q_K

$P_1 = -\Delta x = -4$	$q_1 = x_1 - x_{wmin} = -1$
$P_2 = \Delta x = 4$	$q_2 = x_{wmax} - x_1 = 5$
$P_3 = -\Delta y = 4$	$q_3 = y_1 - y_{wmin} = 7$
$P_4 = \Delta y = -4$	$q_4 = y_{wmax} - y_1 = -3$

All non-zeros in P_K
so find out intersection

$P_K < 0$ (P_1 & P_4)	$P_K > 0$ (P_2 & P_3)
$t_1 = \max(0, q_1/P_1, q_4/P_4)$	$t_2 = \min(1, q_2/P_2, q_3/P_3)$
$t_1 = \max(0, 1/4, 3/4)$	$t_2 = \min(1, 5/4, 7/4)$
$t_1 = 3/4$	$t_2 = 1$
t_1 is not zero it means the point P_1 lies outside the window, we need to clip it and find new coordinates	t_2 is 1 it means point P_2 lies inside the window.

new coordinates will be

$x = x_1 + t_1 \Delta x$

$y = y_1 + t_1 \Delta y$

$x = 4 + \frac{3}{4} \times 4 = 7$

$y = 12 + \frac{3}{4} \times -4 = 9$ $\quad (7, 9)$

Polygon Clipping

A polygon is a closed surface without any curvature.

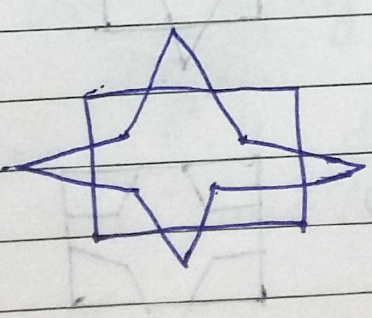
For polygon also, we will consider a window and then we will try to find out the portion of image that lies outside the window and the portion that lies inside the window and we will clip it accordingly.

Sutherland Hodgeman Algoritm

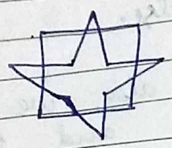
There are four steps for clipping a polygon using this algorithm they are :-

i) Left clip
ii) Right clip
iii) Top clip
iv) Bottom clip

Let us define a initial polygon.

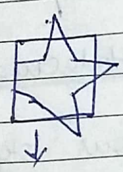


Here we will apply clipping :-

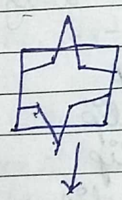


In this the output of the previous step will be the input of the next step.

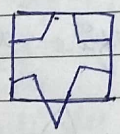
i) Left Clip



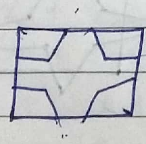
ii) Right Clip



iii) Top Clip

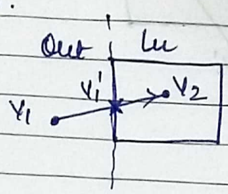


iv) Bottom Clip



We have to consider again 4 cases for all the above mentioned clip.

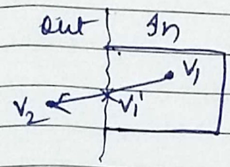
1)



Out - In

Output :- $V_1' V_2$

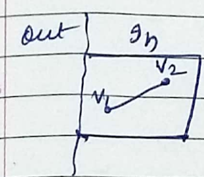
2)



In - Out

Output :- V_1'

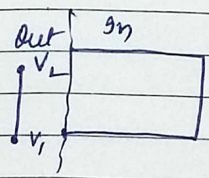
3)



In - In

Output :- V_2

4)



Out - Out

Output :- NIL

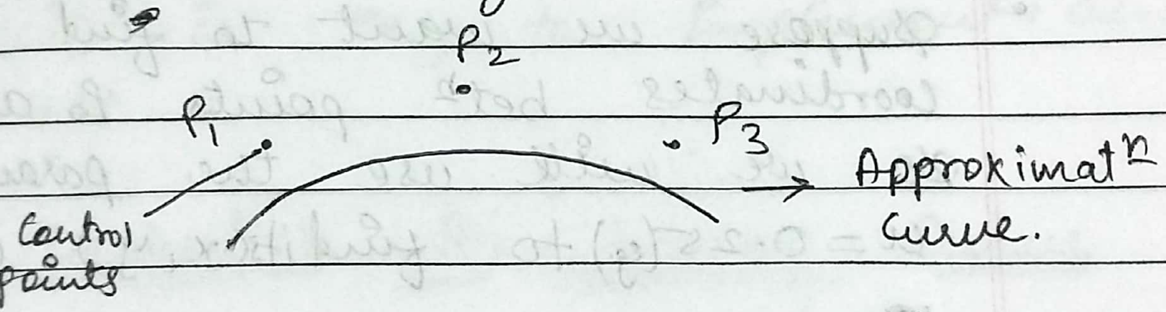
Now using these 4 conditions we can easily findout the clipped polygon by applying it for 4 different regions of the window.

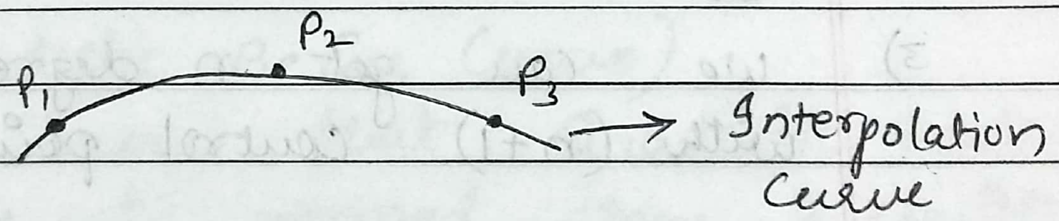
UNIT - 2

Bezier Curve

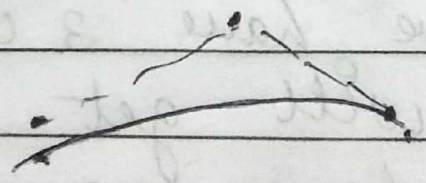
1) Bezier Curve is a spline curves.

- A spline curve is the curve form from the control points.
- Two types of curve could be form from these control points :
  - Curve By approximation
  - Curve By Interpolation

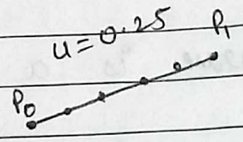




Bezier curve falls into the category of approximation curve



2) Bezier curve is a parametric curve.



$t =$ _ _

$u =$ _ _

$0 \leq u \leq 1$

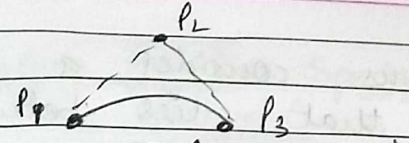
- Here this parametric means the curve depends on some parameter using the values of which we can find coordinates.

- Suppose we want to find out coordinates bet^n points P_0 and P_1, so we will use the parameter $u = 0.25$ (eg.) to find its (x, y) cordinates.

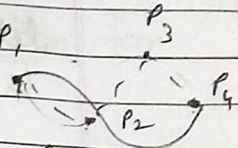
3) We can get n degree polynomial with $(n+1)$ control points.

(degree of polynomial is the highest power value in the polynomial equation).

eg: If we have 3 control points we will get polynomial of degree 2 also called quadratic polynomials.



x^2 (quadratic)



(Cubic) x^3

Similarly we can access other degrees of polynomial with the use of control points.

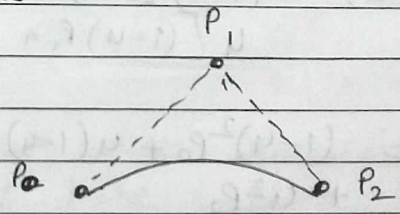
4) Used for CAD applications, Drawing use, typeface etc.

5) It is easy to implement and use.

<u>Derivation for Bezier Curve</u>

(Quadratic Bezier Curve)

We know that Quadratic Bezier Curve is formed from 3 control points.



- We know that Bezier curve is a parametric curve, so

Let us consider a parameter u that lies between 0 & 1

$$0 \leq u \leq 1$$

- Suppose Q_0 and Q_1 are the points that lies between the control points $P_0 \rightarrow P_1$ and $P_1 \rightarrow P_2$ respectively.

As we know that Q_0 lies between $P_0 \rightarrow P_1$ we can write the parametric equation as -

$$Q_0 = (1-u) P_0 + u(P_1) \quad — ①$$

Similarly, $Q_1 = (1-u) P_1 + u(P_2) \quad — ②$

Suppose $C(u)$ is a point between $Q_0 \rightarrow Q_1$

$$C(u) = (1-u) Q_0 + u Q_1$$

$$C(u) = (1-u) [(1-u)P_0 + uP_1] + u[(1-u)P_1 + u(P_2)]$$

$$C(u) = (1-u)^2 P_0 + u(1-u) P_1 + u(1-u)P_1 + u^2 P_2$$

$$\boxed{C(u) = (1-u)^2 P_0 + 2u(1-u) P_1 + u^2 P_2}$$

This is the Quadratic polynomial for the Bezier curve with 3 control point

Blending Function Specification for Bezier Curve

Blending function is a function used to specify Bezier curve.

Suppose we have $n+1$ control points, then for position $P_k(x_k, y_k, z_k)$, the positional vector $P(u)$ is given by :-

$$P(u) = \sum_{k=0}^{n} P_k \underbrace{BEZ_{k,n}(u)}_{\text{Blending function}}$$

where $0 \leq u \leq 1$

$$BEZ_{k,n} = C(n,k)\, u^k (1-u)^{n-k}$$

where $C(n,k)$ is a polynomial coefficient like we studied in permutation combination and is given by :-

$$C(n,k) = \frac{n!}{k!(n-k)!}$$

So, we can recursively calculate points here, that is why we say that implementation of Bezier curve is easy as we can recursively calculate the val

$$x(u) = \sum_{k=0}^{n} x_k \, BEZ_{k,n}(u)$$

$$y(u) = \sum_{k=0}^{n} y_k \, BEZ_{k,n}(u)$$

$$z(u) = \sum_{k=0}^{n} z_k \, BEZ_{k,n}(u)$$

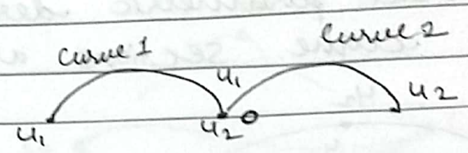
Parametric Continuity Conditions

Continuity Conditions defines the smoothness of the curve.

Continuity Condition:- We can impose various continuity condition at the connection point of piecewise parametric curve to achieve a smooth transition from one curve section to the next.

Zero Order Continuity

It is represented by C_0 continuity and it means simply the curve meets at a point "o" and the value of x, y, & z for the value of u_2 for the first curve section are equal to the ~~the~~ value of x, y, z at u_1 for the next curve section



for zero order continuity :-

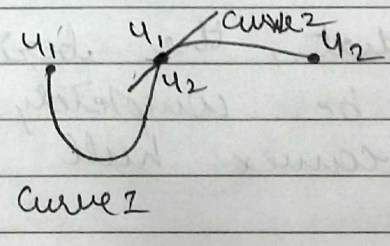
$$x = \underbrace{x(u_2)}_{\text{curve 1}} = \underbrace{x(u_1)}_{\text{curve 2}}$$

$$y = y(u_2) = y(u_1)$$

$$z = z(u_2) = z(u_1)$$

First Order Continuity :-

It is represented by C_1, it mean the first parametric derivative (tangent line) of the coordinate function $x = x(u)$, $y = y(u)$, $z = z(u)$ for the successive curve sections are equal at the joining points.



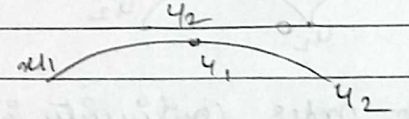
$$\frac{du_2}{dz} = \frac{du_1}{dz}$$

$$\frac{du_2}{dx} = \frac{du_1}{dx}$$

$$\frac{du_2}{dy} = \frac{du_1}{dy}$$

Second Order Continuity

It is represented by C_2 continuity. In this continuity the first and second order parametric derivative of two curve sections are equal.



$\frac{dy}{x}$, $\frac{d^2y}{dx}$ Both.

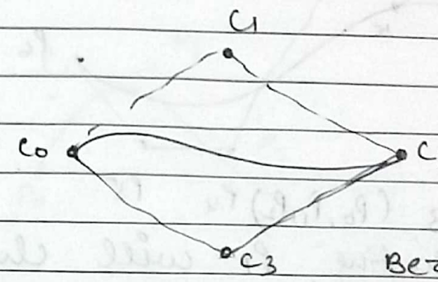
* Second Order ~~derivative~~ continuity is capable of drawing very smooth curve.
It is used in animations.

* First Order continuity is used in digital drawing.

ConvexHull Property Of Bezier Curve

It means that, the bezier curve will always be completely contained inside the convex hull of the control points.

Bezier curve will always lies entirely inside its planar or volumetric convex hull.



Bezier curve with convex hull.

B-Spline Curve

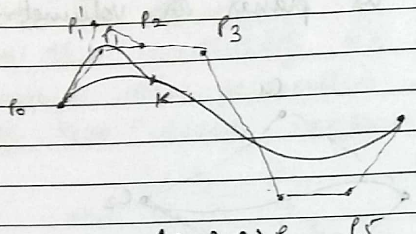
B-Spline curve is the curve where we can control the degree of the polynomial by imposing some conditions onto them.

In B-Spline curves there exists segments through which we control the degree of the polynomial.

Properties :-

1) B-Spline curve is made of $n+1$ control points and the order of the curve is K. [curve(K)

2) It ~~to~~ has local control over the curve.



If $K = 3$ (P_0, P_1, P_2) then shifting P_1 will change only one segment.

3) It is used to draw open & closed curve.

4) It gives us polynomial of degree $K-1$

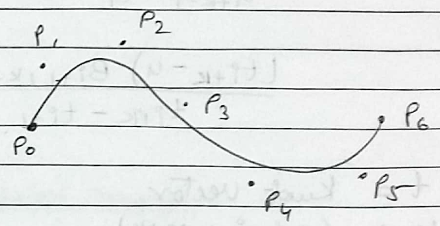
$K = 3$ x^2

$K = 4$ x^3

5) It has $n-K+2$ segments.

6) BSpline curve allow us to change the no. of control points without changing the degree of the polynomial.

eg Suppose we have $n+1$ control points i.e. $n+1 = 7$ & $K = 3$ then $n = 6$ and curve is defined as :-



Then segment will be given as

$n - K + 2 = 6 - 3 + 2 = 5$

0-1, 1-2, 2-3, 3-4, 4-5

K=order

Here $K = 3$ i.e degree $= x^2$

Mathematical Expression for B-Spline Curve.

$$P(u) = \sum_{i=0}^{n} B_{i,K}(u)\, P_K$$

$P_K\ (x_K\ y_K\ z_K)$

$$x(u) = \sum_{i=0}^{n} B_{iK}(u)\, x_K$$

$0 \le u \le n-K+2$

it is a recursive expression $2 \le K \le n+1$

Generalisation Of B-Spline Expression

$$B_{i,k}(u) = \frac{(u - t_i)\, B_{i,k-1}(u)}{t_{i+k-1} - t_i} +$$

$$\frac{(t_{i+k} - u)\, B_{i+1,k-1}(u)}{t_{i+k} - t_{i+1}}$$

t = Knot vector

$t_i \rightarrow (0 \leq i \leq n+k)$

Knot vector is basically a set of ~~sub~~ subinterval endpoints.

Ques Construct the Bezier Curve of order 3 and with 4 polygon vertices A(1, 1), B(2, 3), C(4, 3), D(6, 4)

Sol: The equation of Bezier curve is given by :-

$$P(u) = (1-u)^3 P_1 + 3u(1-u)^2 P_2 + 3u^2(1-u) P_3 + u^3 P_4$$

$$0 \le u \le 1$$

$P(u)$ is the point on the curve P_1, P_2, P_3, P_4

Let us take u as $0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$

$P(0) = P_1 = (1, 1)$

$$P(1/4) = (1-1/4)^3 P_1 + 3 \times 1/4 \, (1-1/4)^2 P_2 + 3(1/4)^2 \cdot (1-1/4) P_3 + (1/4)^3 P_4$$

$P(1/4) = (1.9218, 2.1718)$

Similarly for $P(1/2)$

$P(1/2) = (3.125, 2.875)$

$P(3/4) = (4.5156, 3.375)$

$P(1) = P_3 = (6, 4)$