Unit 5
Curves & Fractals

- A curve is an infinitely large set of points. Each point has two neighbours excepts endpoints.

Types of curves -

1. Implicit curve.
- Define the set of points on a curve by employing a procedure.
- It can test to see if a point is on the curve.
- Defined as $\Rightarrow f(x, y) = 0$
- A common example is circle
$$x^2 + y^2 - r^2 = 0.$$

2. Explicit Curves.
- Mathematical Function $y = f(x)$.
- For each value of x, only single value of y is normally computed by the function.

3. Parametric curve -
$$p(t) = f(t), g(t)$$
OR
$$P(t) = x(t), y(t)$$

All the curves are specified by parametric function
Why curves ? - Irregular surfaces

- The function f & g become the (x, y) coordinates of any point on the curve.
- Points are obtained when the parameter t is varied over a certain interval [a, b]. Normally [0, 1]

* Interpolation -
- "Smoothing things in between"
- Creation of new values that lie between known values.
- When objects are rasterized into two dimensional images from their corner points, all the pixels between those points are filled by interpolation.
- It determines color and other attributes.
- Another example is when video image of low resolution, missing lines.

Interpolate the existing values at fixed grid location to compute values anywhere else on the grid.

2D => bilinear interpolation
3D => trilinear interpolation.

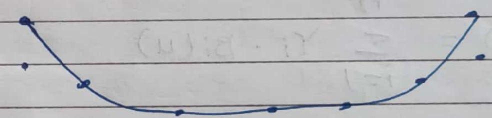
Linear interpolation is an equation of kind => $a(1-t)+bt$ with $0 \leq t \leq 1$
It requires only two values (a & b) & few simple arithmetic opern. t is in range 0 to 1.

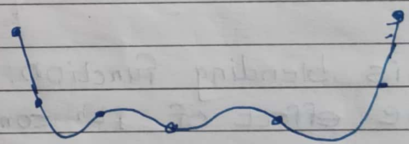
More points / line segments that are used, the smoother the curve.

Interpolant -
A function used to interpolate the values on the regular grid.

- Used in image processing.
- Fluid simulation
- Volume Rendering
- Texture mapping

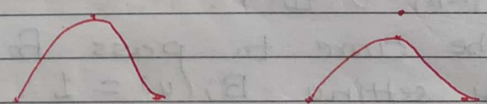
* Approximation
- Bringing something close enough.
- If the curve does not pass through the given control points and we approximate the shape.





Animation & Specifying camera motion

Interpolation



Interpolation Approximation

Blending Function -

- Assume that curve passes through the n control points (x_i, y_i, z_i), $i = 1, 2, \ldots n$.
- Can draw a curve passes through sample points.
- To draw a curve need to fill the points, to do this, need to find the points that lie in beth sample points.

If the point satisfies the line or circle equation then point is found. Similarly for curve also.

$$f_x(u) = \sum_{i=1}^{n} x_i \cdot B_i(u)$$

$$f_y(u) = \sum_{i=1}^{n} y_i \cdot B_i(u)$$

$$f_z(u) = \sum_{i=1}^{n} z_i \cdot B_i(u)$$

- Function $B_i(u)$ is blending function.
- It determines the effect of i^{th} control point for given 'u' on the rest of the curve points.
- u varies from 0 to 1.
- Can force the curve to pass from given point i by setting $B_i(u) = 1$.

when $u = -1 \Rightarrow B_1(u) = 1$ and 0 for $u = 0, 1, 2, \ldots n-2$

$u = 0 \Rightarrow B_2(u) = 1$ and 0 for $u = -1, 1, \ldots n-2$

$u = (n-2) \Rightarrow B_n(u) = 1$ and 0 for $u = -1, 0, \ldots n-1$

Lagrange Interpolation

$$B_1(u) = \frac{(u-1)(u-2)}{(-1)(-2)(-3)}$$

$$B_2(u) = \frac{(u+1)(u-1)(u-2)}{(1)(-1)(-2)}$$

$$B_3(u) = \frac{(u+1)\,u\,(u-2)}{(2)(1)(-1)}$$

$$B_4(u) = \frac{(u+1)\,u\,(u-1)}{(3)(2)(1)}$$

For $u = -1$, $B_1(u) = +1$

$B_2(u) = B_3(u) = B_4(u) = 0$

For $u = 0$, $B_2(u) = 1$

$B_1(u) = B_3(u) = B_4(u) = 0$

For $u = 1$, $B_3(u) = +1$

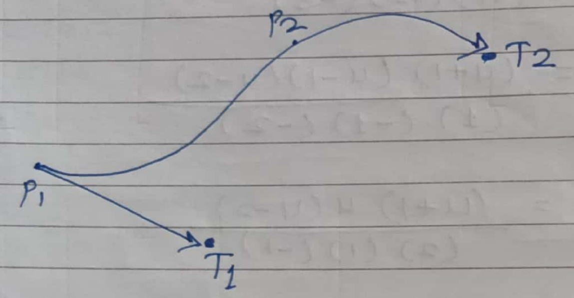
$B_1(u) = B_2(u) = B_4(u) = 0$

For $u = 2$, $B_4(u) = 1$

$B_1(u) = B_2(u) = B_3(u) = 0$.

$$x = x_1 B_1(u) + x_2 B_2(u) + x_3 B_3(u) + x_4 B_4(u)$$

Spline Interpolation Methods :-



$P_1 \Rightarrow$ start point of the hermite curve
$T_1 \Rightarrow$ Tanget of start point
$P_2 \Rightarrow$ End point
$T_2 \Rightarrow$ Tanget of end point.
$p(u) \Rightarrow$ parametric cube point function.

$p(0) = p_k$
$p(1) = p_{k+1}$
$p'(0) = dp_k$
$p'(1) = dp_{k+1}$.

dp_k & $dp_{k+1} \Rightarrow$ Parametric derivatives of p_k & p_{k+1}.

Vector generation:

$P(u) = au^3 + bu^2 + cu + d$.

where x component of p is

$x(u) = a_x u^3 + b_x u^2 + c_x u + d_x$.

$y(u) = a_y u^3 + b_y u^2 + c_y u + d_y$

$z(u) = a_z u^3 + b_z u^2 + b_z u + d_z$.

Matrix form is $\Rightarrow$

$$p(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

derivates of $p(u)$ is $p'(u)$

$p'(u) = 3au^2 + 2bu + c + 0$

Matrix form of $p'(u)$ is

$$p'(u) = \begin{bmatrix} 3u^2 & 2u & 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

$u = 0$ & 1

$$\begin{bmatrix} p_k \\ p_{k+1} \\ dp_k \\ dp_{k+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_k \\ P_{k+1} \\ dP_k \\ dP_{k+1} \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = M_H \begin{bmatrix} P_k \\ P_{k+1} \\ dP_k \\ dP_{k+1} \end{bmatrix}$$

Putting it in above equation

$$P[u] = [u^3 \; u^2 \; u \; 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_k \\ P_{k+1} \\ dP_k \\ dP_{k+1} \end{bmatrix}$$

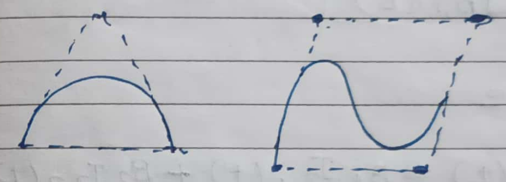
$$p[u] = P_k(2u^3 - 3u^2 + 1) + P_{k+1}(-2u^3 + 3u^2) + dP_k(u^3 - 2u^2 + u) + dP_{k+1}(u^3 - u^2)$$

$$p(u) = P_k H_0(u) + P_{k+1} H_1(u) + dP_k H_2(u) + dP_{k+1} H_3(u).$$

Bezier Curve -

- Pierre Bezier => french engineer.

Degree of polynomial is always 1 less than a number of control points.



With $(n+1)$ control points, parametric equation is

$$p(t) = \sum_{i=0}^{n} P_i \cdot BEZ_{i,n}(t), \; 0 \le t \le 1$$

$$BEZ_{i,n}(t) = C(n,i)\, t^i (1-t)^{n-i}$$

$C(n,i)$ => Bionomial coefficient

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

For each direction

$$x(t) = \sum_{i=0}^{n} x_i \cdot BEZ_{i,n}(t), \; 0 \le t \le 1$$

Same for y and z.

It interpolate the first and last point known as anchor points.

* Cubic bezier curve -

Equation ⇒

$$P(t) = \sum_{i=0}^{n} B_i J_{n,i}(t)$$

Expanding above

$$p(t) = B_0 J_{3,0}(t) + B_1 J_{3,1}(t) + B_2 J_{3,2}(t) + B_3 J_{3,3}(t).$$

i) $J_{3,0}(t) = \frac{3!}{0!(3-0)!} t^0 (1-t)^{3-0}$

$J_{3,0}(t) = (1-t)^3$

ii) $J_{3,1}(t) = \frac{3!}{1!(3-1)!} t^1 (1-t)^{3-1}$

$J_{3,1}(t) = 3t(1-t)^2.$

$J_{3,2}(t) = \frac{3!}{2!(3-2)!} t^2 (1-t)^{3-2}$

$= 3t^2(1-t)$

$J_{3,3}(t) = \frac{3!}{3!(3-3)!} t^3 (1-t)^{3-3}$

$J_{3,3}(t) = t^3$

$$p(t) = B_0(1-t)^3 + B_1 3t(1-t)^2 + B_2 3t^2(1-t) + B_3 t^3.$$

Matrix Representation =

$$p(t) = \sum_{i=0}^{n} P_i \, BEZ_{i,n}(t), \quad 0 \le t \le 1$$

$= P_0 \cdot BEZ_{0,3}(t) + P_1 \cdot BEZ_{1,3}(t) + P_2 \cdot BEZ_{2,3}(t) + P_3 \cdot BEZ_{3,3}(t)$

$= P_0 (1-t)^3 + P_1 3t(1-t)^2 + P_2 3t^2(1-t) + P_3 t^3$

$= (1-3t+3t^2-t^3) P_0 + (3t-6t^2+3t^3) P_1 + (3t^2-3t^3) P_2 + t^3 P_3$

$$p(t) = [t^3 \; t^2 \; t \; 1] \cdot \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

$$p(t) = T \cdot M_{BEZ} \cdot G_{BEZ}$$

* Example:
Given a bezier curve with 4 control points
$B_0[1,0]$, $B_1[3,3]$, $B_2[6,3]$, $B_4[8,1]$
Determine any 5 points lying on the curve.
Also draw rough sketch of cure.

⇒ Parametric equation is

$$P(t) = B_0(1-t)^3 + B_1 3t(1-t)^2 + B_2 3t^2(1-t) + B_3 t^3$$

Substituting values

$$P(t) = [1\ 0](1-t)^3 + [3\ 3](1-t)^2 + [6\ 3]3t^2(1-t) + [8\ 1]t^3.$$

Assume any 5 values for t as
$0 \le t \le 1$
0, 0.2, 0.5, 0.7, 1.

For $t = 0$

$$p[0] = [1\ 0](1-0)^3 + [3\ 3]\ 3(0)(1-t)^2 + [6\ 3]\ 3(0)^2(1-0) + [8\ 1](0)^3$$

$$= [1\ 0] + 0 + 0 + 0$$

$$p(0) = \cancel{[0\ 0]}\ [1\ 0]$$

$t = 0.2$

$$P[0.2] = [2.304\ \ 1.448]$$

$t = 0.5$

$$P(0.5) = [4.5\ \ 2.375]$$

$t = 0.7$

$$P[0.7] = [5.984,\ 2.233]$$

$t = 1$
$$P[1] = [8, 1]$$

⇒ Given $B_0[1,1]$, $B_1[2,3]$, $B_2[4,3]$, $B_3[3,1]$, the vertices of a Bezier polygon determine the points on the Bezier curve.

$$Q(t) = T \cdot M_B \cdot G_B$$

$$= [t^3\ t^2\ t\ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 2 & 3 \\ 4 & 3 \\ 3 & 1 \end{bmatrix}$$

$$= [t^3 \; t^2 \; t \; 1] \begin{bmatrix} -4 & 0 \\ 3 & -6 \\ 3 & 6 \\ 1 & 1 \end{bmatrix}$$

$$= [-4t^3 + 3t^2 + 3t + 1 \quad -6t^2 + 6t + 1]$$

$t = 0 \Rightarrow Q_0(0)$

$$= [-4(0)^3 + 3(0)^2 + 3(0) + 1 \quad -6(0)^2 + 6(0) + 1]$$

$$= [1 \; 1]$$

$t = 0.2 \Rightarrow Q(0.2)$

$$= [-4(0.2)^3 + 3(0.2)^2 + 3(0.2) + 1 \quad -6(0.2)^2 + 6(0.2) + 1]$$

$$= [-0.032 + 0.12 + 0.6 + 1 \quad -0.24 + 1.2 + 1]$$

$$= [1.688 \quad 1.96]$$

$t = 0.5 \Rightarrow Q(0.5)$

$$= [2.75 \quad 2.5]$$

$t = 1.0 \Rightarrow Q(1.0)$

$$= [3 \; 1]$$

$t = 0.8 \Rightarrow Q(0.8)$

$$= [3.272 \quad 1.96]$$

Find the equation for the bezier curve, which passes through control points (0,0) & (-3,2) and controlled by (6,4) and (3,1). Also find points on curve for t = 0, 0.4, 0.8, 1.

$\Rightarrow$ Assume curve is defined by P_0, P_1, P_2, P_3 passes through endpoints (0,0) & (-3,2)

$P_0 = (0,0)$, $P_3 = (-3,2)$

$P_1 = (6,4)$, $P_2 = (3,1)$

$$Q[t] = T \cdot M_B \cdot G_B$$

$$= [t^3 \; t^2 \; t \; 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 6 & 4 \\ 3 & 1 \\ -3 & 2 \end{bmatrix}$$

$$= [t^3 \; t^2 \; t \; 1] \begin{bmatrix} 6 & 11 \\ -27 & -21 \\ 18 & 12 \\ 0 & 0 \end{bmatrix}$$

$$Q(t) = [6t^3 - 27t^2 + 18t \quad 11t^3 - 21t^2 + 12t]$$

Let us find point on curve $\Rightarrow$

$t(0) \Rightarrow Q(0)$.

$$= [6(0)^3 - 27(0)^2 + 18(0) \quad 11(0)^3 - 21(0)^2 + 12(0)]$$

$$= [0 \quad 0]$$

$t = (0.4) \Rightarrow Q(0.4) = [3.264 \quad 2.144]$

$t = 0.8 \Rightarrow Q(0.8) = [2.752 \quad 1.792]$

$t = 1 \Rightarrow Q(1) = [-3 \quad 2]$

* Given four control points (10,10), (15,15), (20,15) and (30,10). Find the points to plot bezier curve by using step 0.2.

$= [6(0.8)^3 - 27(0.8)^2 + 18(0.8) \quad 11(0.8)^3 - 21(0.8)^2 + 12(0.8)]$

$= [6(0.512) - 27(0.64) + 14.4$

$= [3.072 - 17.28 + 14.4$

$= [0.192$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 2 & 4 & 5 & 1 \end{bmatrix} \begin{bmatrix} 2 & 2 & 2 \\ 3 & 4 & 7 \\ 8 & 9 & 12 \\ 1 & 1 & 1 \end{bmatrix}$$

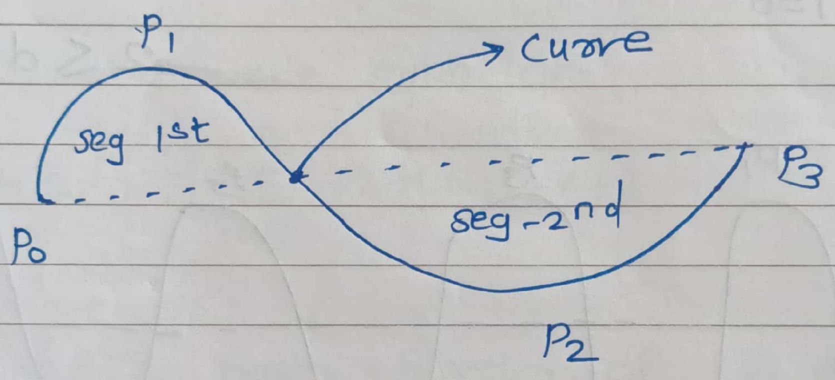
$$= \begin{bmatrix} 2 & 2 \\ & \\ & \end{bmatrix}$$

36,:

1, 11, 12, 13, 18
36, 41, 44, 45, 49, 50, 52, 58, 59, 64,
66, 69, 77, 78

B-spline curve :

- Both curves are parametric in nature.
- Disadvantage of Bezier Curve is if there is any change in one of the control point, the whole shape of bezier curve changes.
- To overcome it B-spline curve is there. It changes only that shape where control point is changed.



Control points $\{P_0, P_1, P_2, P_3\}$

Fig : Before changing the position of P1.

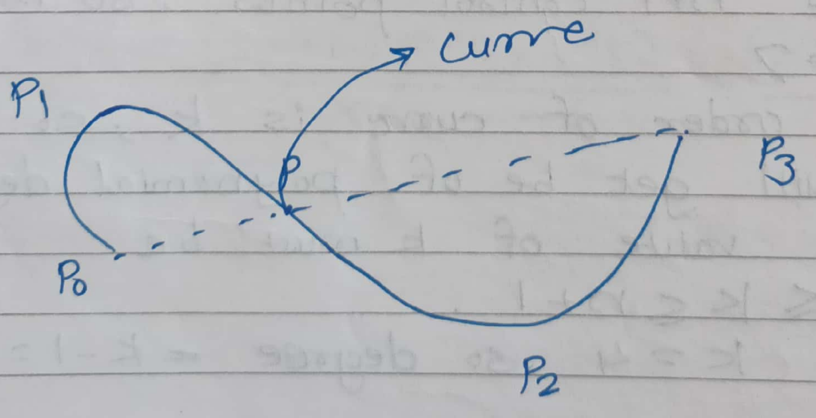


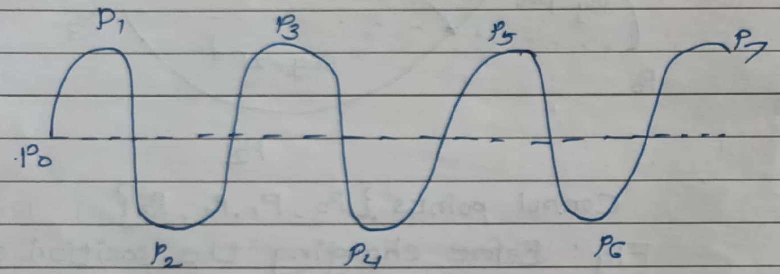
Fig : After changing the position of P1.

- Curves are independent of the number of control points and made up of joining number of segments smoothly.
- Each segment shape is decided by specific control points that come in the region of segment.

B spline curve with n+1 control points

$$P(t) = \sum_{i=0}^{n} P_k \cdot B_{i,d}(t), \quad t_{min} \leq t \leq t_{max}$$

$$2 \leq d \leq n+1$$



We have n+1 control points, so n+1 = 8
so n = 7
Assume order of curve is k, so the curve will get be of polynomial degree k-1.
so the value of k must be
$2 \leq k \leq n+1$.
Assume k = 4, so degree = k-1 = 4-1 = 3

Total number of segments calculated using
n-k+2 = 7-4+2 = 5

Segment	Control points	Parameter
S0	P_0, P_1, P_2, P_3	$0 \leq t \leq 2$
S1	P_1, P_2, P_3, P_4	$2 \leq t \leq 3$
S2	P_2, P_3, P_4, P_5	$3 \leq t \leq 4$
S3	P_3, P_4, P_5, P_6	$4 \leq t \leq 5$
S4	P_4, P_5, P_6, P_7	$5 \leq t \leq 6$.

The point between two segments of a curve that joins each other is known as knots.

B-spline curve equation

$$Q(t) = \sum_{i=0}^{n} P_i * N_{i,k}(t)$$

$N_{i,k}$ ⇒ Basis function of B-spline
P_i, k, t ⇒ Control points, degree, parameter of curve.

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

* Fractals :
- French / American mathematician ⇒ Dr. Benoit Mandelbrot.
- Derived from latin word fracts which means broken.

- Fractals are infinitely complex patterns that are self-similar across different scales.
- Complex pictures generated by computer from a single formula.
- Created using iterations.
- One formula is repeated with slightly different values, over and over again, taking into account results from previous iteration.

* Classification

1 Exact self similarity
- Strongest type.
- Appears identical at diff scales.
- Iterated functions systems.

2 Quasi self similarity
- Loose form.
- Approximately identical @ diff. scales.

3. Statistical
- Weakest type.

Applications -

Space science
- Analyzing galaxies, positioning of starts, behavior, position of planet.

Medical discipline
- Bacteria cultures, chemical reactions, human anatomy, molecules, plants.

Others :
- Representing clouds, coastline, border area, data compression, diffusion, landscapes etc.

* Topological Dimension :
- Assume object is made up of soft materia clay.
- If the object is broken down in terms of line or segment, Dimension $D_t = 1$
- In the form of plane then $D_t = 2$.
- In terms of 3D object like cube, sphere, then $D_t = 3$.

* Fractal dimension :

- Imagine 'a' line segment of length L and is divided into N identical pieces each length $l = L/N$.

Scanned with OKEN Scanner

$$\frac{1}{S} = \frac{1}{L}$$

$N = S^1$

For square $N = S^2$

For cube $N = S^3$

$N = S^D$

$\log N = D \log S$.

$$D = \frac{\log N}{\log S}.$$

Amount of variation in object detail is described by a number called fractal dimension.

Jaggy boundary have larger D value.

Fractal lines :

- Computer procedure can easily generate self-similar process by using recursive procedure.
- It needs to end it at a certain instant.

(x_1, y_1, z_1) & (x_2, y_2, z_2)

Halfway point

$$\left(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}, \frac{z_1+z_2}{2}\right)$$

But for fractal line we have to add some offset to each co-ordinate

$$\left(\frac{x_1+x_2}{2}+dx, \frac{y_1+y_2}{2}+dy, \frac{z_1+z_2}{2}+dz\right)$$

Random effect is calculated as :

$dx = L * W * \text{Gauss}$.

L ⇒ Length of segment

W ⇒ Weight factor the curve roughness.

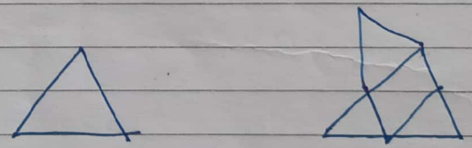
Gauss ⇒ Random value from -1 to 1.

* Fractal surfaces :
- To draw 3D objects like mountain ranges.
- Ex : Triangle in space.

i) Consider each edge of triangle. Compute halfway point.

ii) Connect it using line segments. Four smaller triangles.

iii) Recursively apply the procedure.

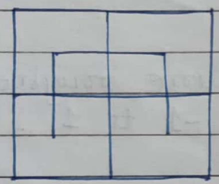


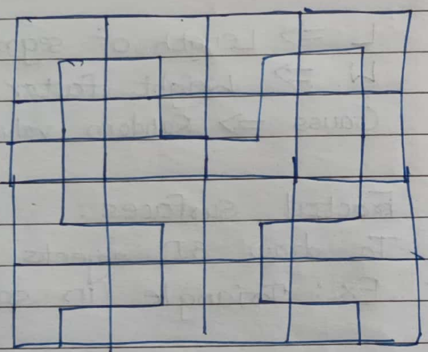
1) Koch curve : ⇒ Helga von Koch ⇒ 1904

2) Hilbert curve :

David Hilbert

Square is divided into 4 quadrants and draw the curve that connects the center points





No limit to subdivision.
Length of the curve is infinite.
Length is increased by 4.

$N = s^{Df}$

$4 = 2^{Df}$

$Df = 2$.

Koch curve -

- Mathematical curve.

We need to draw the objects onto the screen. Objects are not flat all the time. We need to draw the curves many times to draw an object.

Curve - ~~set of~~.
Infinitely large set of points.

Implicit curves -

$$f_{x,y} = 0$$

$$x^2 + y^2 - R^2 = 0$$

Explicit curve -

$$y = f_x.$$

Parametric curve -

$$P_t = f_t , g_t \quad \text{or} \quad p_t = x_t , y_t$$