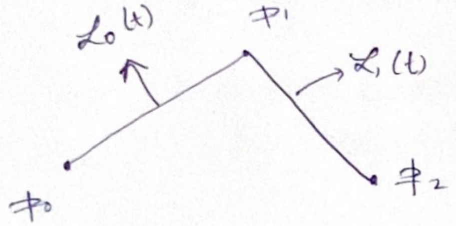
Bezier Curve

Linear Bezier Curve

$$L_0(t) = (1-t)p_0 + t p_1$$

Quadratic Bezier Curve :-



$$L_0(t) = (1-t)p_0 + t p_1$$

$$L_1(t) = (1-t)p_1 + t p_2$$

$$Q_0(t) = (1-t)L_0(t) + t L_1(t)$$

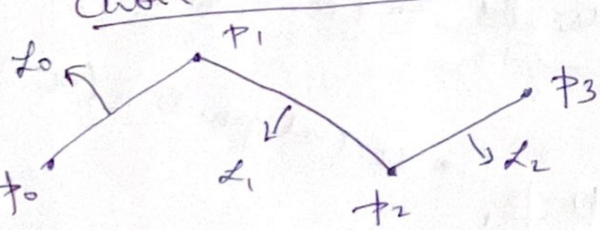
$$Q_0(t) = (1-t)^2 p_0 + 2(1-t)t p_1 + t^2 p_2$$

& by construction is Quadratic.

$Q_0(0) = p_0$ $\quad Q_0(1) = p_2$

NB :- 1st & Last Control points are end points of the curve.

Cubic Bezier Curve :-



$$L_0(t) = (1-t)p_0 + t p_1$$

$$L_1(t) = (1-t)p_1 + t p_2$$

$$L_2(t) = (1-t)p_2 + t p_3$$

$$Q_0(t) = (1-t)L_0(t) + t L_1(t)$$

& $Q_1(t) = (1-t)L_1(t) + t L_2(t)$

$$C_0(t) = (1-t)Q_0(t) + t Q_1(t)$$

$$= (1-t)^3 p_0 + 3(1-t)^2 t p_1 + 3(1-t)t p_2 + t^3 p_3$$

Bernstein Polynomials

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

$\boxed{i \le n}$

$B_{0,0} = 1$

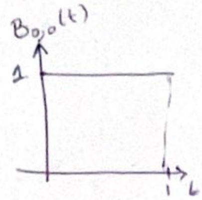
$B_{0,1} = 1-t$

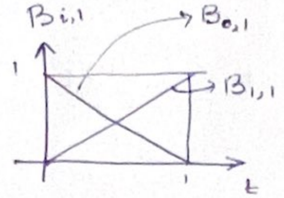
$B_{1,1} = t$

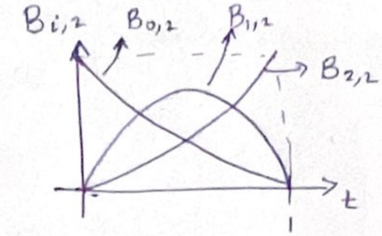
$B_{0,2} = (1-t)^2$

$B_{1,2} = 2t(1-t)$

$B_{2,2} = t^2$







Properties:-

→ $B_{i,n}(t) = B_{n-i,n}(1-t)$

→ $B_{i,n}(t) \geq 0$

→ $\sum_{0}^{n} B_{i,n}(t) = 0 \qquad 0 \leq t \leq 1$

→ $B_{i,n}(t)$ with $i \neq 0, n$ has a single max of $i^{i} n^{-n} (n-i)^{n-i} \binom{n}{i}$ at $t = i/n$

Using Bernstein Polynomials, the Bezier Curves Can be written as

$$B(t) = \sum_{i=1}^{n} B_{i,n}(t)\, P_i$$ → control points $= (x_i, y_i)$

→ this is a parametric curve $(x(t), y(t))$

$$x(t) = \sum_{i=1}^{n} x_i\, B_{i,n}(t)$$

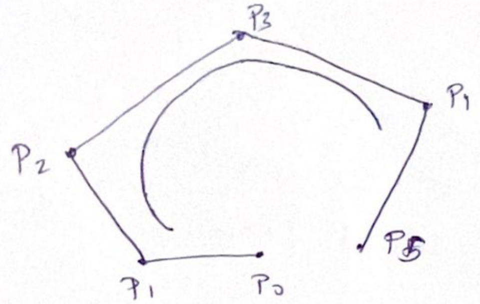
$$y(t) = \sum_{1}^{n} y_i\, B_{i,n}(t).$$

B-splines (Basis Splines)

- This uses Bézier Curves stuck end to end.

↳ A k degree B-spline defined by $n+1$ control points consists of $n-k+1$ Bezier curves

eg:- A cubic B-spline defined by 6 control points $p_0, \dots, p_5$ consists of $n-k+1 = 3$ Bezier curves.



↳ The final point on the 1st Bézier Curve has the same ~~point~~ coordinate as the 1st point on the 2nd Bezier Curve (This gives C^0 continuity).

↳ 1st derivative at the end of the 1st Bezier Curve is the same as the 1st derivative at the start of the 2nd (C^1)

↳ 2nd derivative is ||arly continuous (C^2).

The eqn of a B-spline of deg. k

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

where $\{P_i\}_0^n$ are control points

& $N_{i,k}(t)$ are Basis fns defined using Cox-De Boor recursion.

$$N_{i,0}(t) \begin{cases} 1 & \text{if } t_i \le t \le t_{i+1} \\ 0 & \text{otherwise.} \end{cases}$$

$$N_{i,j}(t) = \frac{t - t_i}{t_{i+j} - t_i} N_{i,j-1}(t) + \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}} N_{i+1,j-1}(t)$$

$$\begin{tikzcd}
N_{0,0} \arrow[r] & N_{0,1} \arrow[r] & \cdots & N_{0,k-1} \arrow[r] & N_{0,k} \\
N_{1,0} \arrow[ur] \arrow[r] & N_{1,1} \arrow[ur] \arrow[r] & \cdots & N_{1,k-1} \arrow[ur] & \\
\vdots & \vdots & & & \\
N_{n-1,0} \arrow[ur] \arrow[r] & N_{n-1,1} \arrow[ur] & & & \\
N_{n,0} \arrow[ur] & & & &
\end{tikzcd}$$

This shows us that in order to calculate $N_{0,k}$, we can lay out the ~~term~~ individual terms on a triangle.

→ **Knot Vector**

the $\{t_i\}_0^m$ are taken ~~ke~~ from the knot vector

$$T = (t_0, t_1, \dots, t_m) \quad \& \quad t \in [t_0, t_m]$$

~~⊕~~ The ~~#~~ knots that t lies between, ~~&~~ determine the basis function that affect the shape of the b-spline.

The # knots in T $(m+1)$ is related to the degree k ~~of~~ and the no. of control points $(n+1)$ by

$$m = k + n + 1$$

A cubic B-spline defined using control points $P_0, \dots, P_4$ requires $1 + m = 1 + 3 + 4 + 1 = 8 + 1$ knots.

$$T = (t_0, \dots, t_8)$$

→ Uniform quadratic B spline

If knots are equidistant, we have uniform B-splines

↳ If a uniform quad. B-spline is defined the control points (p_0, p_1, p_2) then

$m = k + n + 1 = 2 + 2 + 1 = 5$

$T = (t_0, t_1, \ldots, t_5) = (0, 1, 2, 3, 4, 5)$ (say).

B-spline is

$$S(t) = \sum_{0}^{2} N_{i,2}(t) P_i$$

quad

→ B-spline Surface

Defined by $(n+1) \times (m+1)$ array of control points given by extending the B-splines curves in 2 dimension.

$$P(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} P_{ij} N_{i,k}(u) N_{j,l}(v) \qquad 0 \le u \le u_{max}, \quad 0 \le v \le v_{max}$$

For illustration check "B-spline Curves with Knots.nb" in ~~the~~ Codes/Mathematica.

Let the control points be $\{P_i\}_0^n \quad P_i \in \mathbb{R}^k$

the knot vector is $(t_0, t_1, \dots, t_m)$

the degree is $d = m - n - 1 \Rightarrow m = d + n + 1$

Here the knot vector satisfies $0 \leq t_0 \leq t_1 \leq \dots \leq t_m \leq 1$

$$N_{i,0}(t) = \begin{cases} 1 & t \in [t_i, t_{i+1}) \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,d}(t) = \frac{t - t_i}{t_{i+d} - t_i} N_{i,d-1}(t) + \frac{t_{i+d+1} - t}{t_{i+d+1} - t_{i+1}} N_{i+1,d-1}(t)$$

the B-spline function is defined as

$$\underline{C}(t) = \sum_{i=0}^{n} P_i \, N_{i,d}(t)$$

for non-periodic B-splines, the 1st $\cancel{p+1}$ $d+1$ knots are equal to 0 & the last $d+1$ knots are equal to 1.

If k duplications happen at the other knots, the curve becomes $\cancel{d}$ $d-k$ times differentiable. ~~Th~~

Thus by overlapping the knots, ~~the curve becomes~~ you can generate curves with sharp turns or discontinuities.

The demonstration project above picks the rest of the knots uniformly.