1) A multi rotor UAV is a Holonomic System i.e.

-> every axis of motion ~~can~~ can be controlled independently

-> Given a set of waypoints we can fit 3 bernstein polynomials to the 3 axis $(x, y, z)$.

-> The trajectory can be parameterized as a bernstein polynomial as shown below

$$\begin{bmatrix} x(t_0) \\ x(t_1) \\ \vdots \\ x(t_f) \end{bmatrix} = P c_x \quad ; \quad \begin{bmatrix} \dot{x}(t_0) \\ \dot{x}(t_1) \\ \vdots \\ \dot{x}(t_f) \end{bmatrix} = \dot{P} c_x \quad ; \quad \begin{bmatrix} \ddot{x}(t_0) \\ \ddot{x}(t_1) \\ \vdots \\ \ddot{x}(t_f) \end{bmatrix} = \ddot{P} c_x$$

Where $P, \dot{P}, \ddot{P}$ are matrices formed by time dependent basis function i.e a Bernstein polynomial & $c_x$ are the coefficients.

We can ~~&~~ derive similar expressions for y & z axis in terms of $c_y$, $c_z$.

2) To accomodate more waypoints into the non-Holonomic Berstein we have to add the constrains as shown below.

To find $w_{x0}, w_{x1} \cdots w_{x5}$

$w_{x0} = x(t_0) = x_{t0}$

$w_{x5} = x(t_f) = x_{tf}$

$$
\begin{bmatrix}
x_{tc1} - w_{x0} B_0(M(t_{c1})) - w_{x5} B_5(M(t_{c1})) \\
x_{tc2} - w_{x0} B_0(M(t_{c2})) - w_{x5} B_5(M(t_{c2})) \\
\vdots \\
x_{tcn} - w_{x0} B_0(M(t_{cn})) - w_{x5} B_5(M(t_{cn})) \\
\hline
\dot{x}_{t0} - w_{x0} \dot{B}_0(M(t_{c1})) - w_{x5} \dot{B}_5(M(t_0)) \\
\dot{x}_{tf} - w_{x0} \dot{B}_0(M(t_{cf})) - w_{x5} \dot{B}_5(M(t_f)) \\
\hline
\dot{x}_{tc1} - w_{x0} \dot{B}_0(M(t_{c1})) - w_{x5} \dot{B}_5(M(t_{c1})) \\
\dot{x}_{tc2} - w_{x0} \dot{B}_0(M(t_{c2})) - w_{x5} \dot{B}_5(M(t_{c2})) \\
\vdots \\
\dot{x}_{tcn} - w_{x0} \dot{B}_0(M(t_{cn})) - w_{x5} \dot{B}_5(M(t_{cn}))
\end{bmatrix}
\begin{array}{l}
\left.\begin{array}{l} \\ \\ \\ \\ \end{array}\right\} n \text{ way point constrains} \\
\\ \\
\left.\begin{array}{l} \\ \\ \\ \end{array}\right\} n \text{ velocity constrains}
\end{array}
$$

$$
= \begin{bmatrix}
B_1(M(t_{c1})) & \cdots & B_4(M(t_{c1})) \\
B_1(M(t_{c2})) & \cdots & B_4(M(t_{c2})) \\
\vdots & \ddots & \vdots \\
\dot{B}_1(M(t_{cn})) & \cdots & \dot{B}_4(M(t_{cn}))
\end{bmatrix}
\begin{bmatrix}
w_{x1} \\ w_{x2} \\ w_{x3} \\ w_{x4}
\end{bmatrix}
$$

Similarly for $\theta$ and $y$ axis we get.

$$\left[\begin{array}{cccc} K_{c_1} & -w_{k_0}\, B(n|t_{c_1})) & -w_{k_5}\, B(n|t_{c_1}) \\ K_{c_2} & -w_{k_0}\, B(n|t_{c_2})) & -w_{k_5}\, B(n|t_{c_2})) \\ K_{c_3} & -w_{k_0}\, B(n|t_{c_3})) & -w_{k_5}\, B(n|t_{c_3})) \\ \vdots & \vdots & \vdots & \vdots \\ K_{c_n} & -w_{k_0}\, B(n|t_{c_n})) & -w_{k_5}\, B(n|t_{c_n})) \\ \hline K_{t_0} & -w_{k_0}\, \dot{B}(n|t_0)) & -w_{k_5}\, \dot{B}(n|t_f)) \\ \dot{K}_{t_f} & -w_{k_0}\, \dot{B}(n|t_f)) & -w_{k_5}\, \dot{B}(n|t_f)) \\ \hline y_0 & -w_{k_{00}}\, F_{00} & -w_{k_5}\, F_{05} \\ y_1 & -w_{k_0}\, F_{10} & -w_{k_0}\, F_{15} \\ y_2 & -w_{k_0}\, F_{20} & -w_{k_0}\, F_{25} \\ \vdots & \vdots \\ y_n & -w_{k_0}\, F_{n0} & -w_{k_0}\, F_{n5} \end{array}\right]$$

n orientation Constrains

n position (y) Constrains

$$= \begin{bmatrix} B_0(n|t_{c_1})) & B_2(n|t_{c_1})) & \cdots & B_n(n|t_{c_1}) \\ B_1(n|t_{c_1}) & B_2(n|t_{c_1}) & \cdots & B_n(n|t_{c_1}) \\ \vdots \\ F_1(t_0) & F_2(t_0) & \cdots & F_n(t_0) \\ F_1(t_1) & F_2(t_1) & \cdots & F_n(t_1) \\ \vdots \\ F_1(t_n) & F_2(t_n) & \cdots & F_n(t_n) \end{bmatrix} \begin{bmatrix} w_{k_1} \\ w_{k_2} \\ w_{k_3} \\ w_{k_4} \end{bmatrix}$$

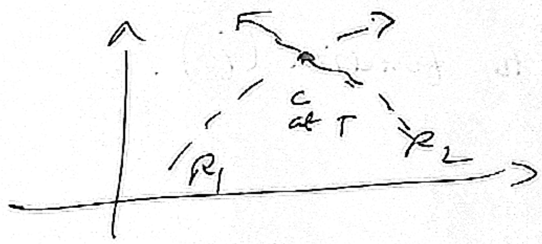
$$x(t) = \sum_{i=0}^{5} w_{xi} B_i(\kappa(t))$$

$$y(t) = y_0 + \sum_{i=0}^{5} w_{ki} F_i(t)$$

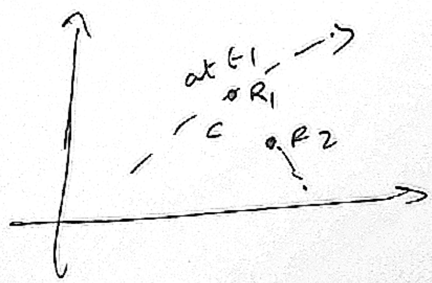
$$\theta(t) = \tan^{-1}\left( \sum_{i=0}^{5} w_{ki} B_i(\kappa(t)) \right)$$

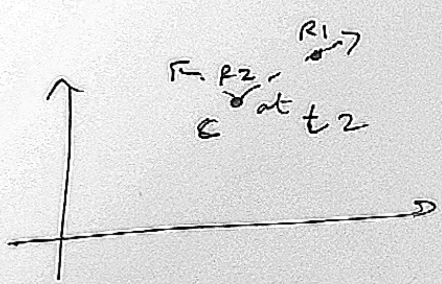
4) Time Scaling can be used to ensure that multiple non-Holonomic robots donot collide

-> Consider an Example shown below where the trajectory of two robots are set to collide at at time "T", at point C.




-> using time Scaling Robot $R_1$ can pass through C at time $t_1$ and $R_2$ can pass through C at time $t_2$. Thus avoiding each other.



5) ~~App~~ Advantages of Bernstein polynomial over other approximation techniques.

-> They are ~~numerically~~ numerically stable basis

-> Bernstein polynomials & its derivatives uniformly approximate ~~to the~~ the function (f) and ~~its~~ the derivative of the function (ḟ).