RangeAero

ASSIGNMENT

Submitted By:-

SUBHAM

⇒ subham.1@iitj.ac.in

Run and Setup.

=> Unzip the file "rangeaero_assignment.zip". into the "src" folder of your catkin work space.

=> Build the Catkin workspace with Command.

=> Catkin_make in folder catkin_ws.

=> Run the following Commands

roslaunch rangeaero_assignment rrr_subham_xy.launch.

RangeAero → Assignments

Submitted by :— Subham

mail → Subham.1@iitj.ac.in

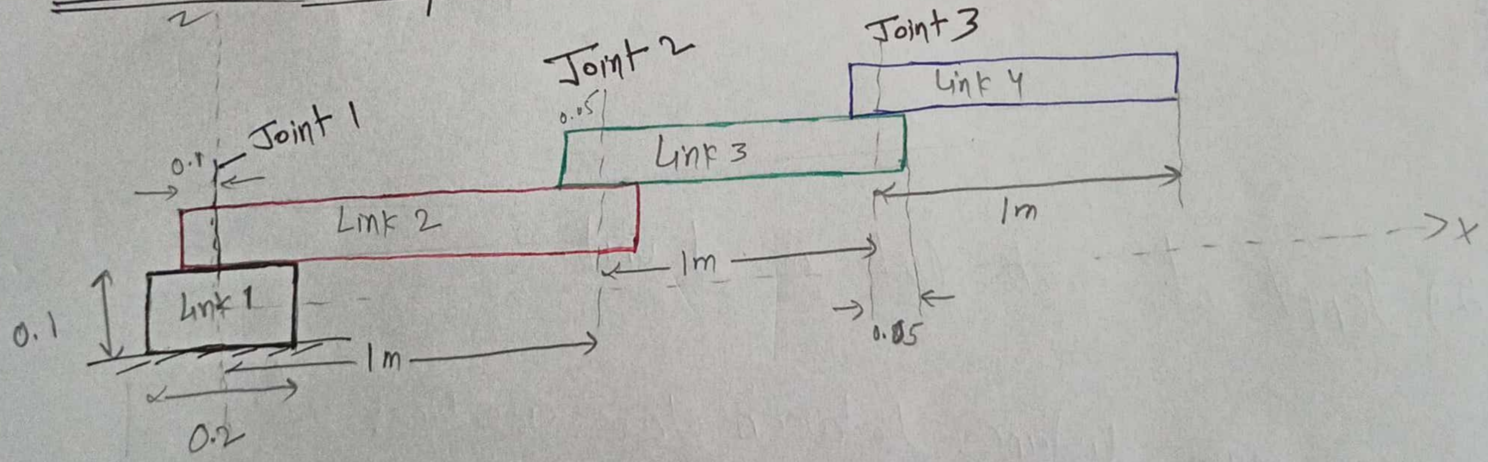
Problem Statement:-

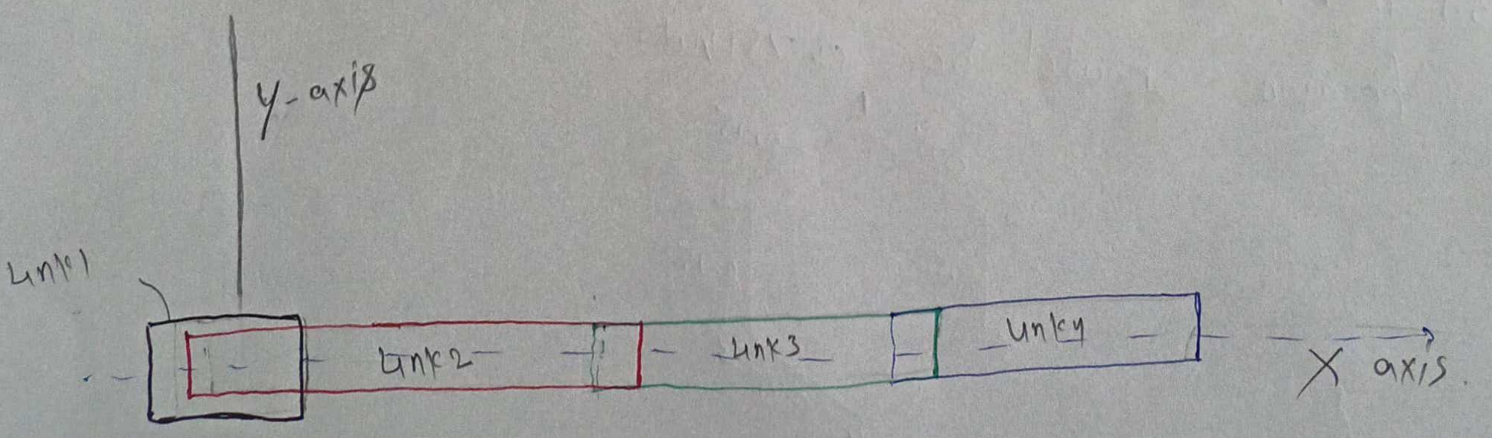
Make a simulation of a kinematical model for 3DoF planar robotic arm with 3 revolute. Then implement a control strategy to follow hexagonal way points in minimum time. Use Gazebo and Robot operating system to solve the above.

for Robot in xy-plane Run. launch file.

rrr_subham_xy.launch.

Robot Description:-





Constraints:

1.) Revolute joint can have a sweep range of -170 to +170 degree (-2.967 to 2.967 radians)

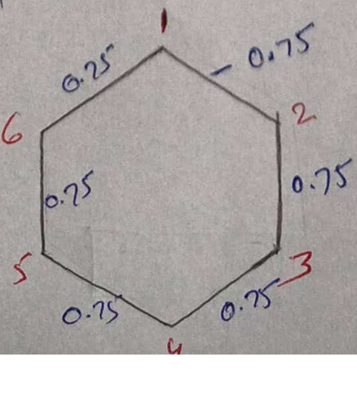
⇒ This constant is achieved by fixing the range of each revolute joint in urdf (xacro) file. as by adding a command line.

<limit effort = "10000" lower = "-2.967" upper = "2.967" velocity = "0.5" />

adds the constraints on each joint angle from -170 to +170 degree.

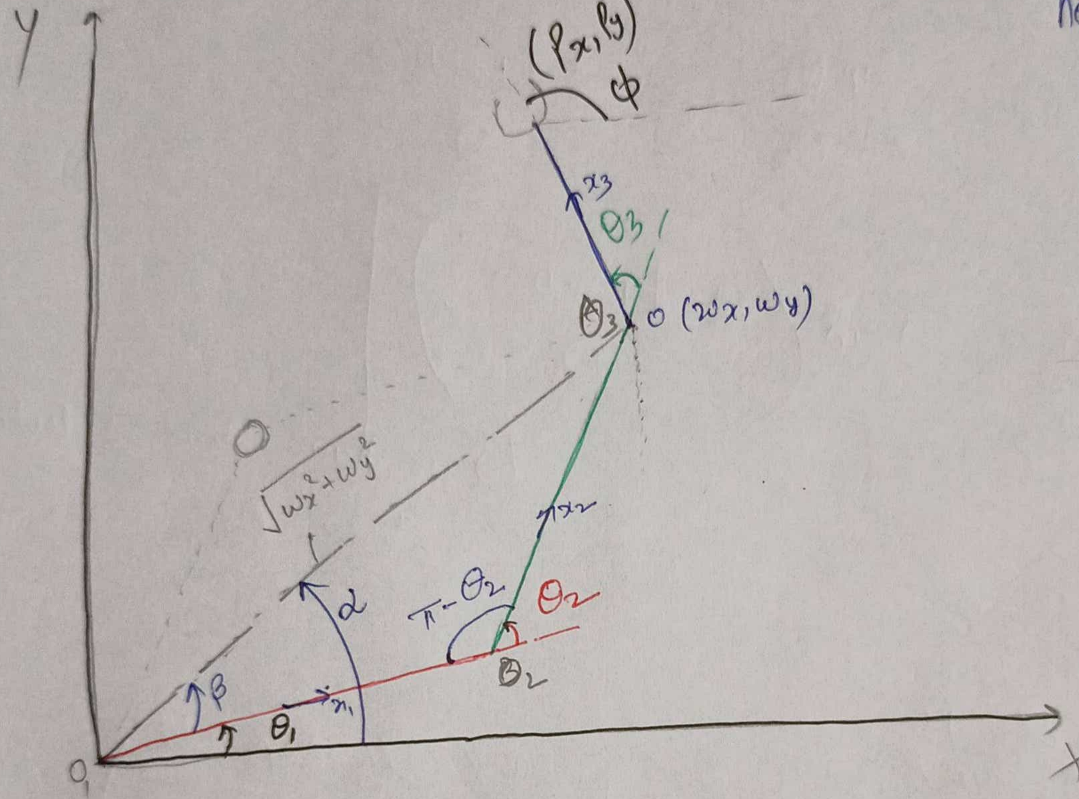
2.) length of each link is 1 unit.

3.) minimum distance between two vertices of hexagon should be 0.75 unit.



Inverse Kinematics:-

To map end effector position and orientation to the joint angles of manipulator.



here lengths of each link $l_1 = l_2 = l_3 = 1$

from geometry:

$$\left.\begin{aligned} P_x &= w_x + l_3 \cos\phi \\ P_y &= w_y + l_3 \sin\phi \end{aligned}\right\} \Rightarrow \begin{aligned} w_x &= P_x - l_3 \cos\phi \\ w_y &= P_y - l_3 \sin\phi \end{aligned}$$

in $\Delta O_1 O_2 O_3$

$$\cos(\pi-\theta_2) = \frac{l_1^2 + l_2^2 - (w_x^2 + w_y^2)}{2 l_1 l_2}$$

$$\cos\theta_2 = \frac{w_x^2 + w_y^2 - l_1^2 - l_2^2}{2 l_1 l_2} \equiv (1)$$

$$\sin\theta_2 = \pm\sqrt{1-\cos^2\theta_2}$$

$$\sin\theta_2 = \pm\sqrt{1-D^2}$$

Hence

$$\tan\theta_2 = \frac{\sin\theta_2}{\cos\theta_2} = \frac{\pm\sqrt{1-D^2}}{D}$$

$$\boxed{\theta_2 = \tan^{-1}\left(\frac{\pm\sqrt{1-D^2}}{D}\right)}$$

↳ we have two solutions for θ_2.

$$\theta_1 = \alpha - \beta$$

$$\theta_1 = \tan^{-1}\left(\frac{w_y}{w_x}\right) - \tan^{-1}\left(\frac{l_2\sin\theta_2}{l_1 + l_2\cos\theta_2}\right)$$

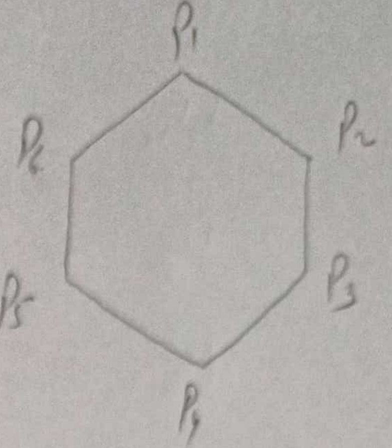
finally:

$$\theta_1 + \theta_2 + \theta_3 = \phi$$

$$\boxed{\theta_3 = \phi - \theta_1 - \theta_2}$$

Approach to follow Hexagonal way points:-

Given 6 points $[x, y]$ of hexagon. as $P_1, P_2, P_3 \ldots P_6$.



Let the model is.

$$\boxed{P_{i+1}(t) = P_i(t) + \Delta t \, u_i} \quad \text{——} \quad (1)$$

where $P_{i+1} \rightarrow$ Position at $i+1$

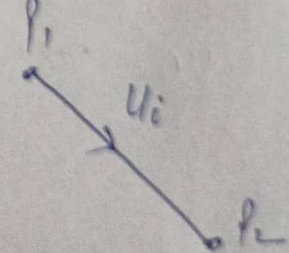
$P_i \rightarrow$ Position at i time

$\Delta t \rightarrow$ time step.

$u_i \rightarrow$ Control ~~to~~ input to the system.

Let u_i is given ~~the~~ as the velocity in direction of point P_i to P_{i+1}.

So $\quad u_i = P_{i+1} - P_i$



Start with P_1 and P_2

when robot reached to the Point P_2

update Control low with new way points.

So $\quad u_1 = P_2 - P_1 \longrightarrow$ for P_1 to P_2

$u_2 = P_3 - P_2$

$$u_3 = P_4 - P_3$$

$$u_4 = P_5 - P_4$$

$$u_5 = P_6 - P_5$$

and. $$u_6 = P_1 - P_6$$

Visualization and results:

⇒ Visualization through gazebo.

Run. launch file:

⇒ roslaunch rangeaero_assignment rrr_subham_xy.launch.

⇒ Visualization through python script

open VS code.

run rrr_hexagon_subham.py file.

To view direct results.

→ go to the folder

results in rangeaero_assignment package