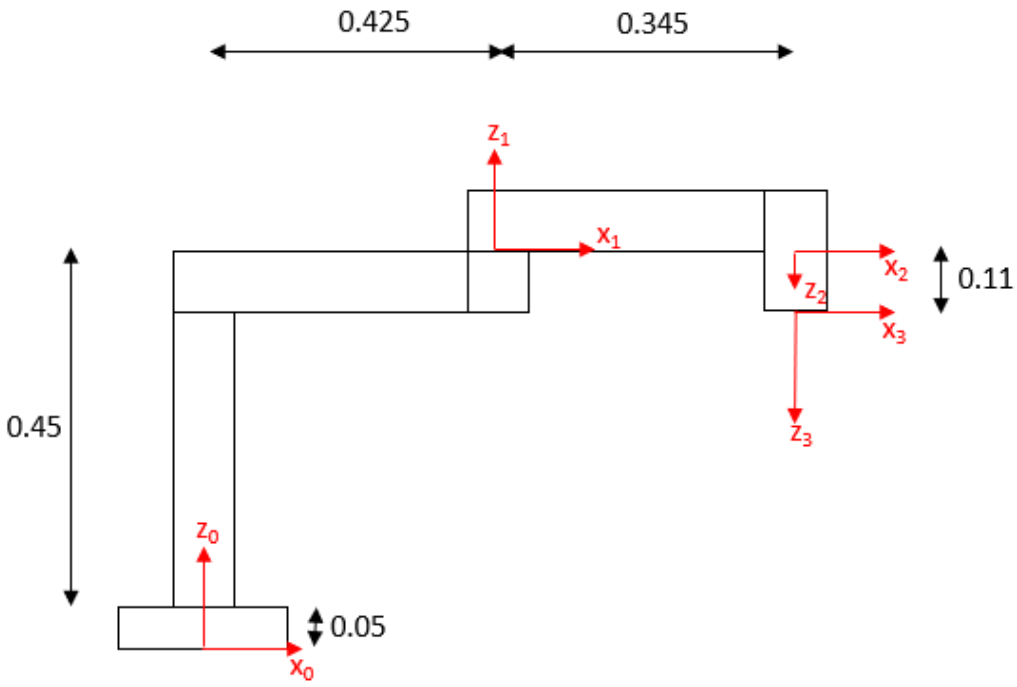
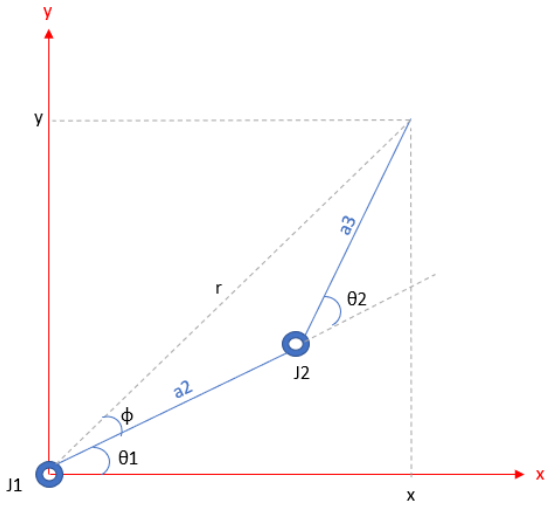
FORWARD KINEMATICS



DH parameter table

	θ	d	a	α
1	**θ1**	0.5	0.425	0
2	**θ2**	0	0.345	**π**
3	0	D3 + 0.11	0	0

INVERSE KINEMATICS



$$r = \sqrt{x^2 + y^2}$$

$$\theta_2 = \cos^{-1}\left(-\frac{a_2^2+a_3^2-r^2}{2a_2a_3}\right)$$

$$\phi = \cos^{-1}\left(\frac{a_2^2+r^2-a_3^2}{2a_2r}\right)$$

$$\theta_1 + \phi = \tan^{-1}\left(\frac{y}{x}\right)$$

$$\theta_1 = \tan^{-1}\left(\frac{y}{x}\right) - \phi$$

$$D_3 = 0.45 + 0.05 - 0.11 - z = 0.39 - z$$

CODE

The ‘rrp_ik_server.py’ program contains a service ‘inver_server’ which takes the end effector position as input, calculates inverse kinematics using geometric approach (equations as shown above) and gives the joint variables as the output.

The ‘rrp_pid.py’ program is the design of PID controllers for each of the joint of the robot. A subscriber with JointStates is called which continuously updates the current joint variables. There are 3 publishers which publish the joint forces to the robot. Four end effector positions are given, one of each is given to the ‘inver_server’ and the corresponding joint variables are obtained. These variables are given to a separate PID controllers and obtain the forces needed to be applied for those joint variables. These forces are given to the topic ‘rrp/jointN_effort_controller/command’ for each joint.

The gains for each joint’s controller is as shown below:

Joint 1: P = 5, I = 0.5, D = 10

Joint 2: P = 5, I = 0.5, D = 10

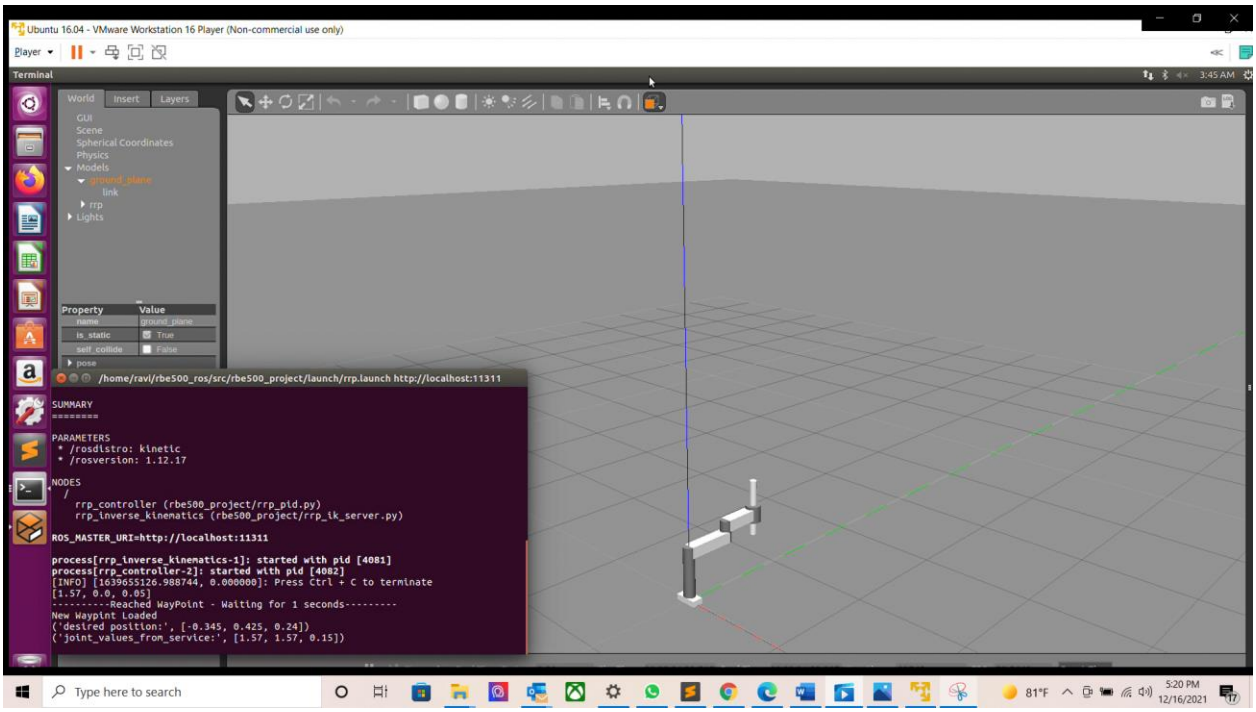
Joint 3: P = 10, I = 0.01, D = 5

The maximum acceptable errors were maintained to be 0.15 radians for each of the revolute joints, whereas 0.002 for the prismatic joint.

RESULTS

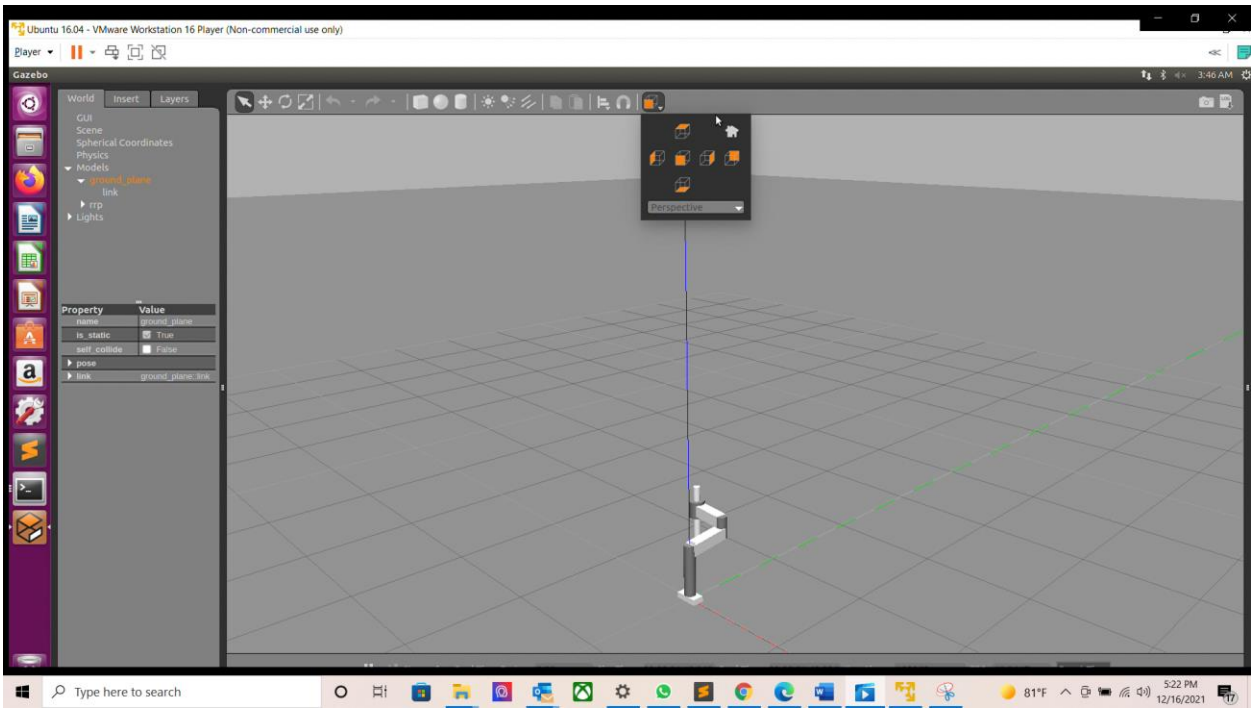
P1 = [0, 0.77 , 0.34] ------ Joint variables : θ1=1.57, θ2=0, D3=0.05

```
ravi@ubuntu: ~
ravi@ubuntu:~$ rosservice call /inver_server "x: 0.0
y: 0.77
z: 0.34"
q1: 1.57
q2: 0.0
q3: 0.05
ravi@ubuntu:~$
```



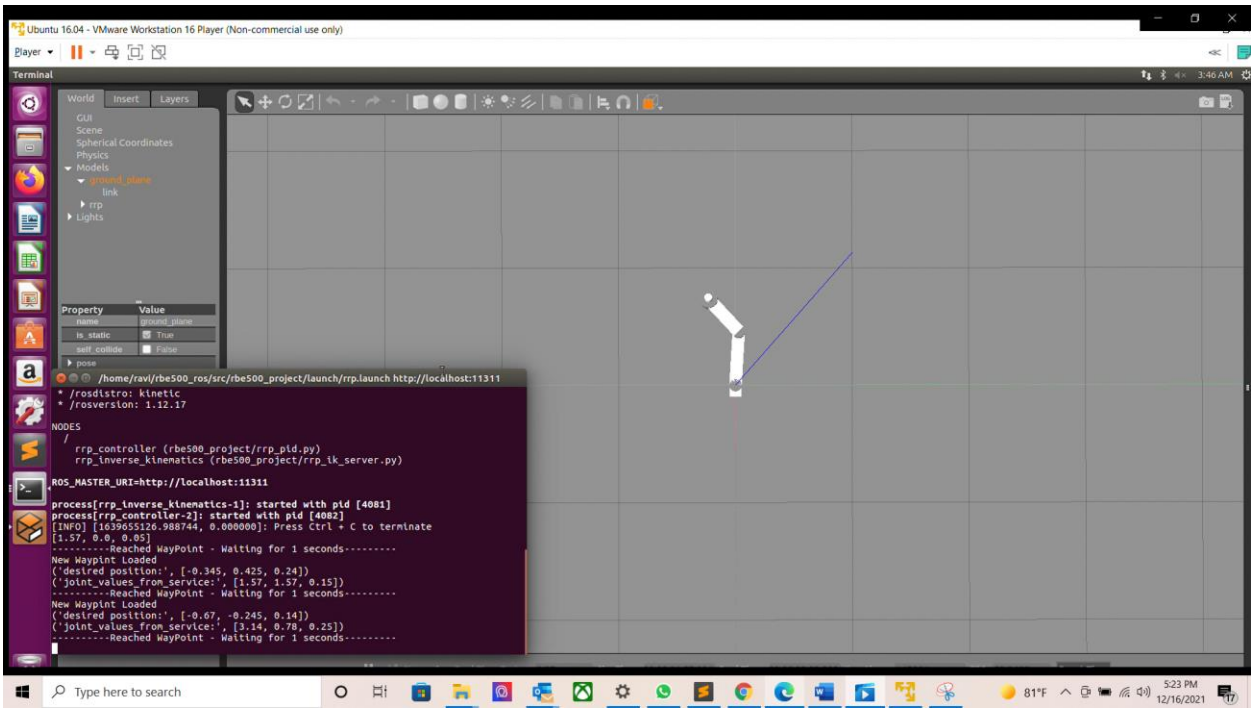
P2 = [-0.345, 0.425 , 0.24] ------ Joint variables : θ1=1.57, θ2=1.57, D3=0.15

```
ravi@ubuntu: ~
ravi@ubuntu:~$ rosservice call /inver_server "x: -0.345
y: 0.425
z: 0.24"
q1: 1.57
q2: 1.57
q3: 0.15
ravi@ubuntu:~$
```



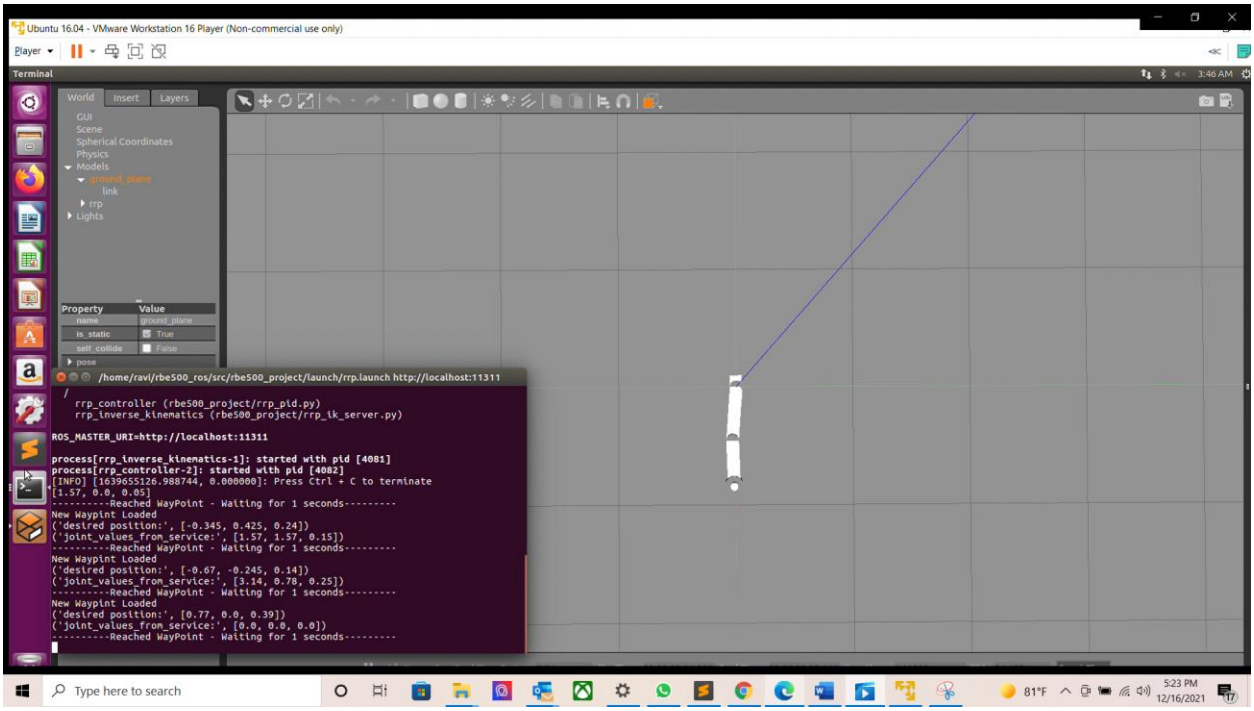
P3 = [-0.67, -0.245 , 0.14] ------ Joint variables : θ1=3.14, θ2=0.78, D3=0.25

```
ravi@ubuntu: ~
ravi@ubuntu:~$ rosservice call /inver_server "x: -0.67
y: -0.245
z: 0.14"
q1: 3.14
q2: 0.78
q3: 0.25
ravi@ubuntu:~$
```



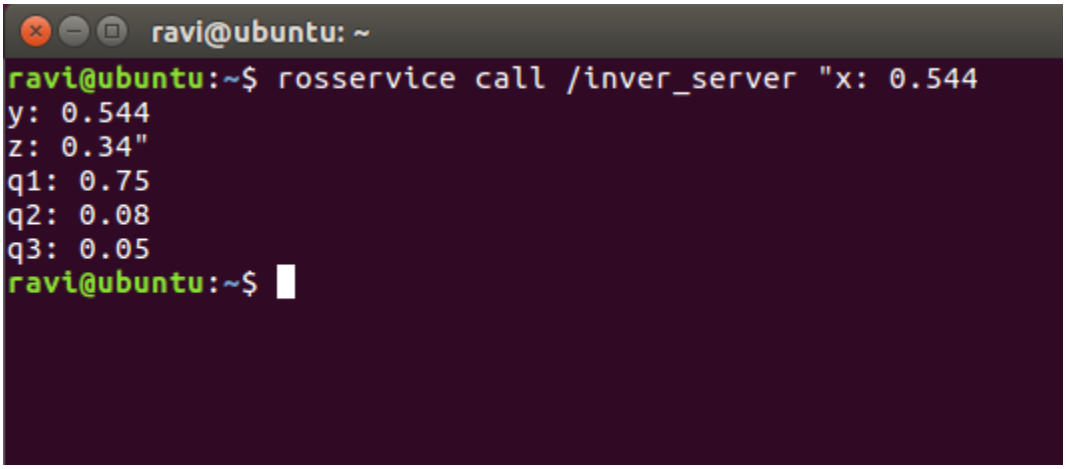
P4 = [0.77, 0, 0.39] ------ Joint variables : θ1=0, θ2=0, D3=0

```
ravi@ubuntu: ~
ravi@ubuntu:~$ rosservice call /inver_server "x: 0.77
y: 0.0
z: 0.39"
q1: 0.0
q2: 0.0
q3: 0.0
ravi@ubuntu:~$
```

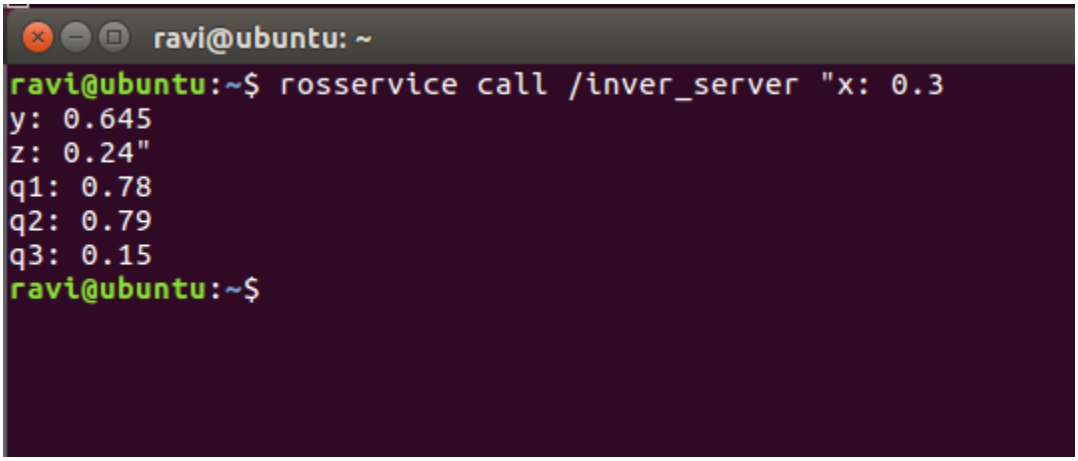


TEST CASES

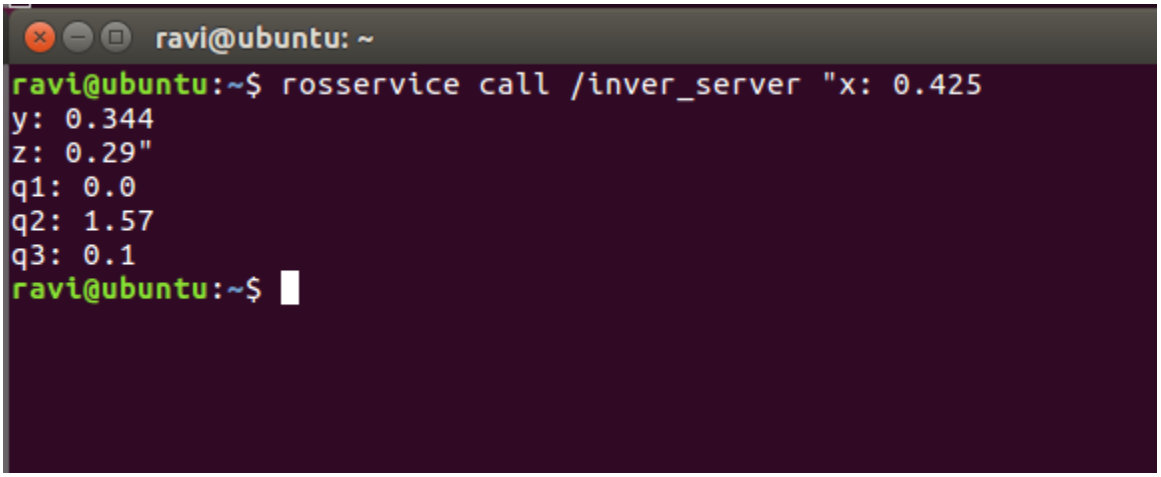
T1 = [0.544, 0.544, 0.34] ------ Joint variables : θ1=0.75, θ2=0.08, D3=0.05



T2 = [0.3, 0.645, 0.24] ------ Joint variables : θ1=0.78, θ2=0.79, D3=0.15



T3 = [0.425, 0.344, 0.29] ------ Joint variables : θ1=0, θ2=1.57, D3=0.1



T4 = [-0.426, -0.344, 0.24] ------ Joint variables : θ1=3.14, θ2=1.57, D3=0.15