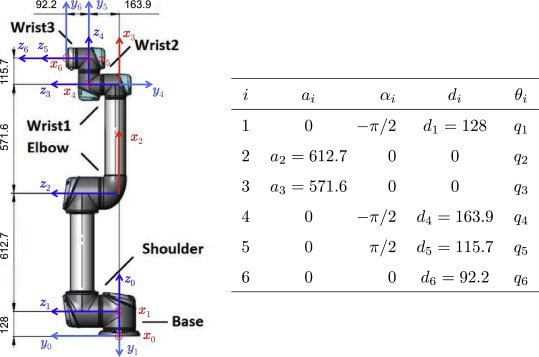
Jose Corona

Final Exam

1.Implement IK function for theUR10robot which takes only Cartesian position (x, y, z) as the input based on deferent approaches:

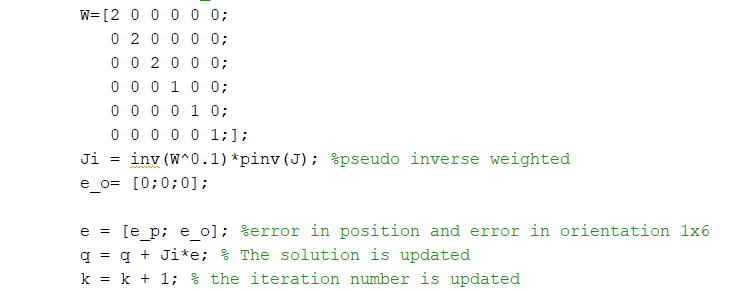


FK

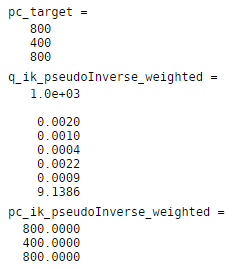
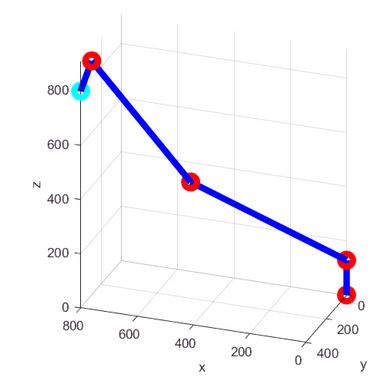


**a. Weighted pseudoinverse:**

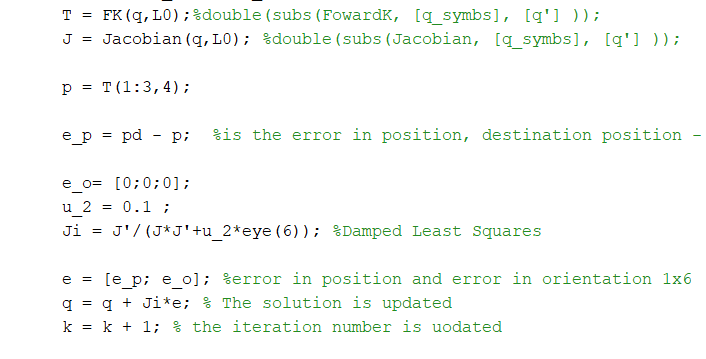
I choose in the Weighted matrix, a higher weight for the first two motors.



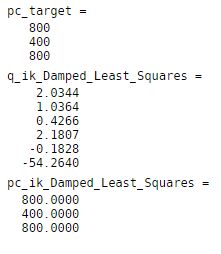
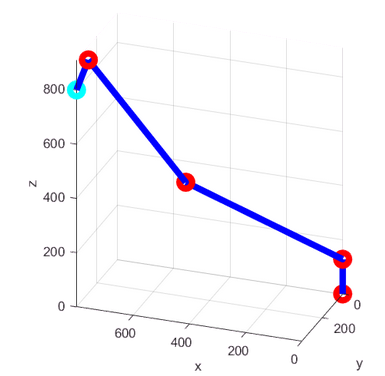
The target point was (800,400,800)

**b. Damped Least Squares**

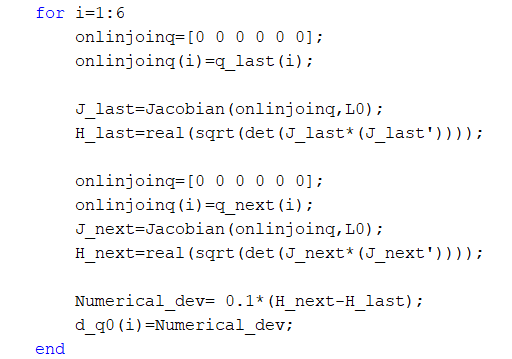


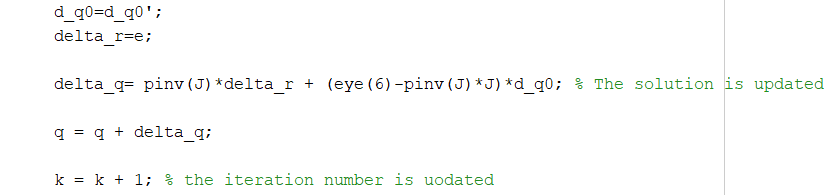
The target point was (800,400,800)

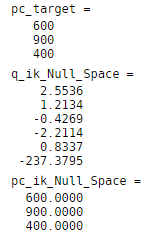
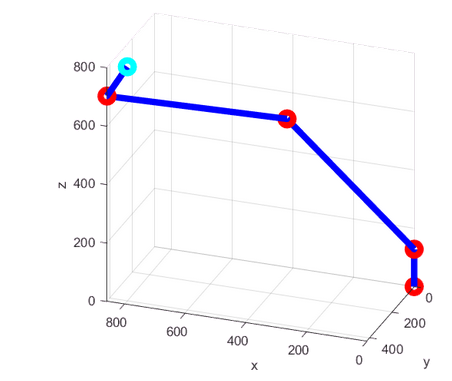
c. Null-space method with objective functions H(q) which maximize the distance from joint limits

I implemented Null space in other file. I calculated the jacobian for each joint, not all the joint at the same time to get a differentiation for each join for dq0 . Since the determinat of everything is just one value, was always cero if I use the jacaobian of all joints.





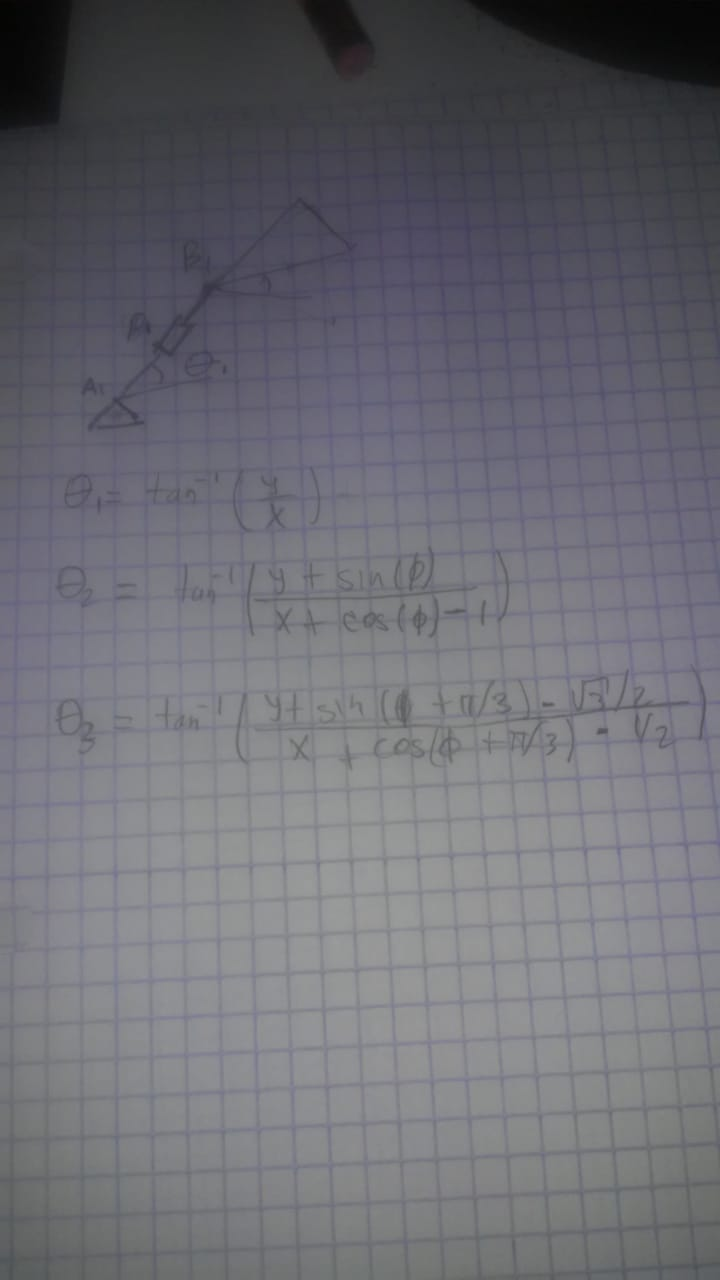
The target point was (600,900,400)

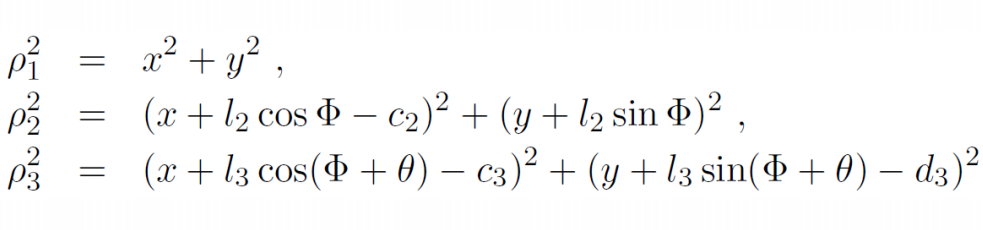
2.Implement functions for the3-RPR parallel

**Inverse Kinematics**

Knowing the orientation for each leg



we could find the displacement in the joint. As in presentation of lecture 9



Asumming:

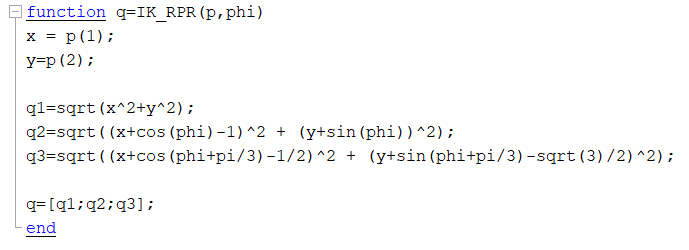
L1=L2=L3=1. Equilateral triangle . Then =pi/3 ,

And the positon of the base (ground) for each leg will be:

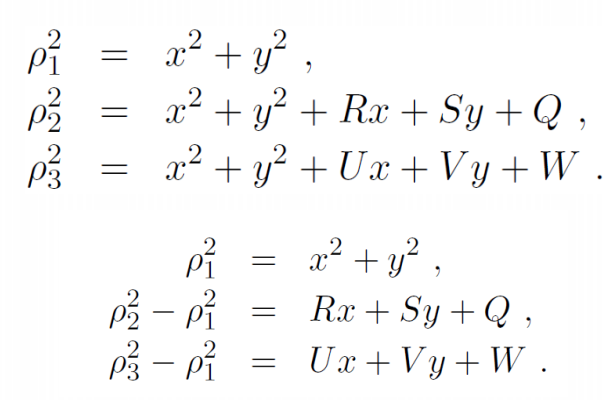
A1=[0,0] A2=[1,0] A3=[0.5,]

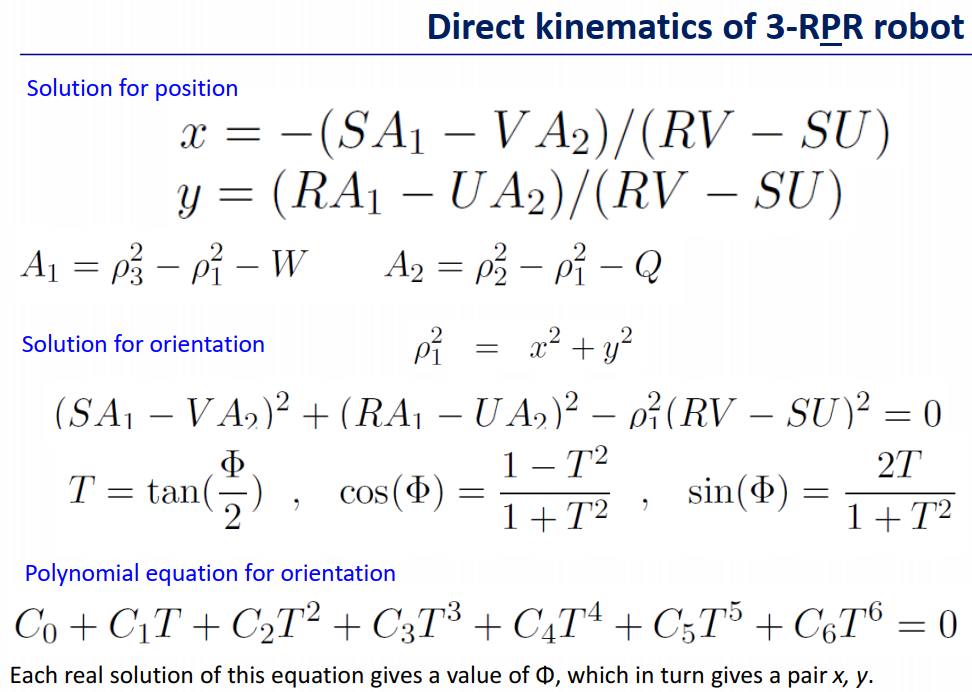
Then c2=1 , c3 = 0.5 and d3=

We could get

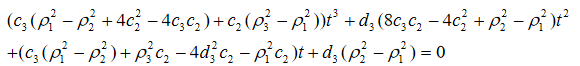


**For the Forward Kinematics**:

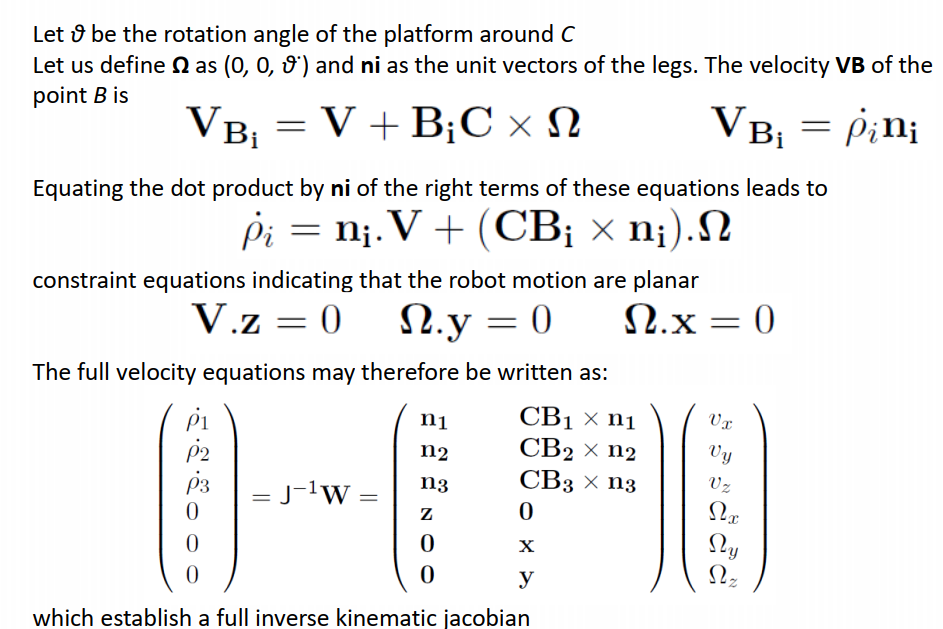




From the polynomial equation we could obtain



**Jacobian**



Git Hub

<https://github.com/Jose-R-Corona/FinalExam-AR>