ROBOTIC ARM: PICK AND PLACE

Project 2

AUGUST 19, 2017

YOUSSEF KHAKY

Youssef Khaky
Project 2: Kinematics Pick & Place

Criteria I Forward Kinematics

The forward kinematics script was helpful in generating the DH table and making sense of the DH parameters.

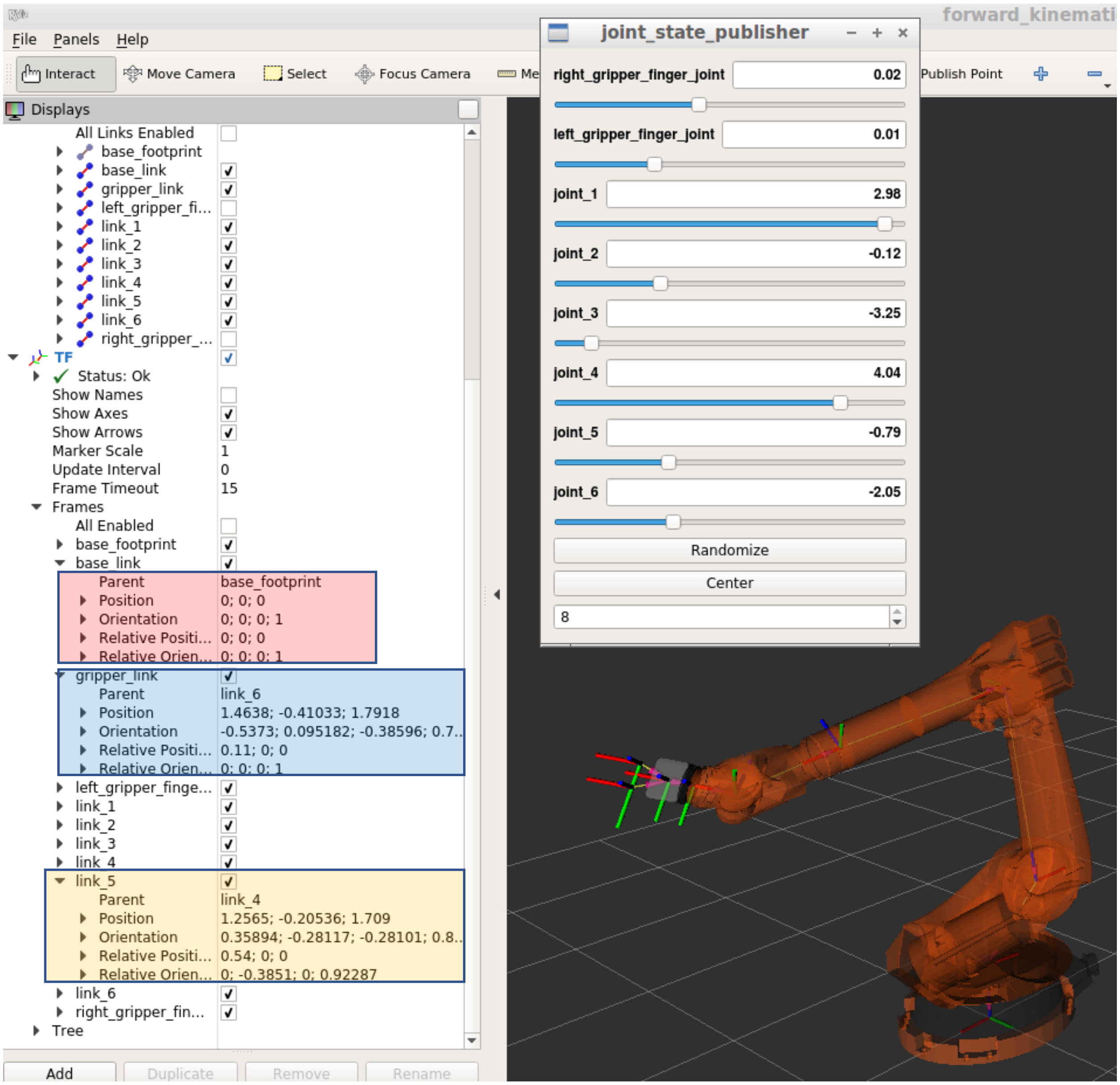


Figure 1Running the Forward Kinematics using roslaunch kuka_arm forward_kinematics.launch, base link, gripper link and wrist center are shaded red, blue and yellow/ orange respectively.

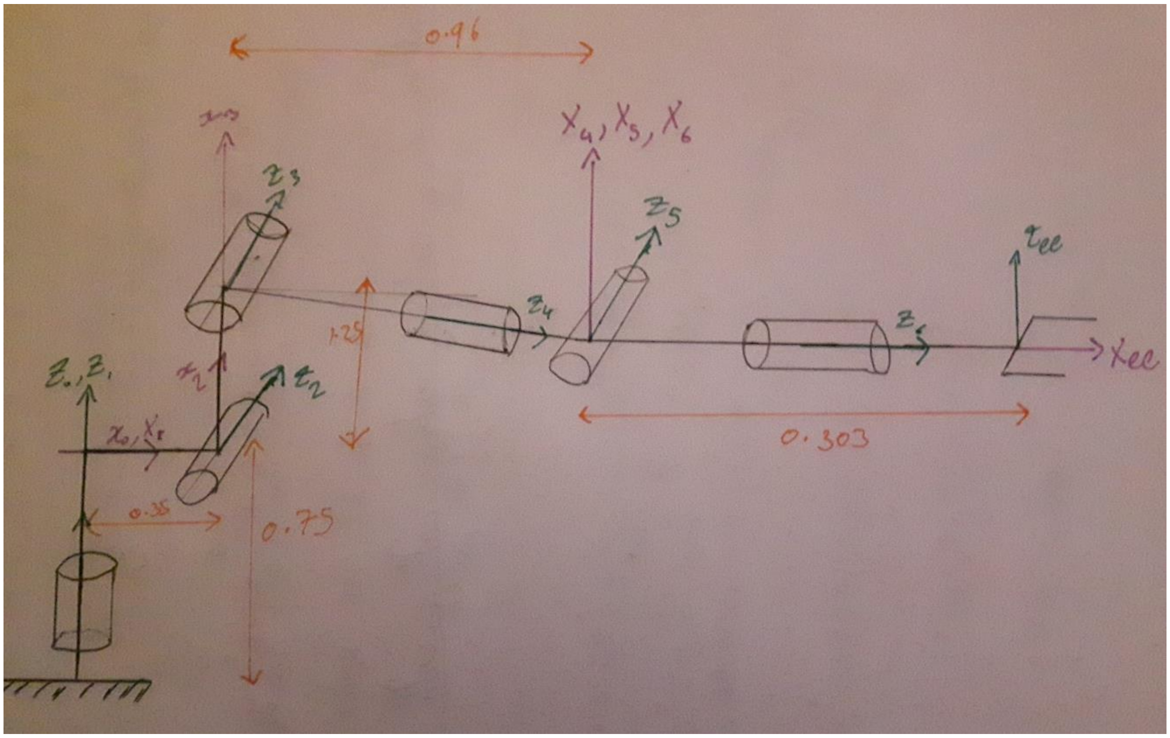


Figure 2 Overview of the robot joints and the DH frames used

DH parameter table

from → to	i	a_{i-1}	α_{i-1}	d_i	θ_i
0 → 1	1	0	0	0.75	θ_1
1 → 2	2	0.35	$-\pi/2$	0	$\theta_2 - \pi/2$
2 → 3	3	1.25	0	0	θ_3
3 → 4	4	-0.054	$-\pi/2$	1.5	θ_4
4 → 5	5	0	$\pi/2$	0	θ_5
5 → 6	6	0	$-\pi/2$	0	θ_6
6 → 7	7	0	0	0.303	0

How this table was obtained

The values for this table were obtained from the URDF (Unified Robot Description Format) file. The file name is kr210.urdf.xacro. The specific section at which these values were found is the **joints** section which starts from line 316.

A segment of the script is pasted below, the line at which these values were obtained is highlighted green. These values represent the origins of the frames corresponding to each joint.

```
<!-- joints -->
<joint name="fixed_base_joint" type="fixed">
  <parent link="base_footprint"/>
  <child link="base_link"/>
  <origin xyz="0 0 0" rpy="0 0 0"/>
</joint>
<joint name="joint_1" type="revolute">
  <origin xyz="0 0 0.33" rpy="0 0 0"/>
  <parent link="base_link"/>
  <child link="link_1"/>
```

```
    <axis xyz="0 0 1"/>
    <limit lower="${-185*deg}" upper="${185*deg}" effort="300" velocity="${123*deg}"/>
  </joint>
  <joint name="joint_2" type="revolute">
    <origin xyz="0.35 0 0.42" rpy="0 0 0"/>
    <parent link="link_1"/>
    <child link="link_2"/>
    <axis xyz="0 1 0"/>
    <limit lower="${-45*deg}" upper="${85*deg}" effort="300" velocity="${115*deg}"/>
  </joint>
  <joint name="joint_3" type="revolute">
    <origin xyz="0 0 1.25" rpy="0 0 0"/>
    <parent link="link_2"/>
    <child link="link_3"/>
    <axis xyz="0 1 0"/>
    <limit lower="${-210*deg}" upper="${(155-90)*deg}" effort="300" velocity="${112*deg}"/>
  </joint>
  <joint name="joint_4" type="revolute">
    <origin xyz="0.96 0 -0.054" rpy="0 0 0"/>
    <parent link="link_3"/>
    <child link="link_4"/>
    <axis xyz="1 0 0"/>
    <limit lower="${-350*deg}" upper="${350*deg}" effort="300" velocity="${179*deg}"/>
  </joint>
  <joint name="joint_5" type="revolute">
    <origin xyz="0.54 0 0" rpy="0 0 0"/>
    <parent link="link_4"/>
    <child link="link_5"/>
    <axis xyz="0 1 0"/>
    <limit lower="${-125*deg}" upper="${125*deg}" effort="300" velocity="${172*deg}"/>
  </joint>
  <joint name="joint_6" type="revolute">
    <origin xyz="0.193 0 0" rpy="0 0 0"/>
    <parent link="link_5"/>
    <child link="link_6"/>
    <axis xyz="1 0 0"/>
    <limit lower="${-350*deg}" upper="${350*deg}" effort="300" velocity="${219*deg}"/>
  </joint>
```

Youssef Khaky
Project 2: Kinematics Pick & Place

Criteria II Transform Matrixes

Using the DH parameter table and the generic homogeneous transform below, we obtain the individual transform matrixes about each joint

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Transform from frame 0 → 1

$$ \begin{bmatrix} \cos(q1), -\sin(q1), 0, 0 \\ \sin(q1), \cos(q1), 0, 0 \\ 0, 0, 1, 0.75 \\ 0, 0, 0, 1 \end{bmatrix} $$

Transform from frame 1 → 2

$$ \begin{bmatrix} \sin(q2), \cos(q2), 0, 0.35 \\ 0, 0, 1, 0 \\ \cos(q2), -\sin(q2), 0, 0 \\ 0, 0, 0, 1 \end{bmatrix} $$

Transform from frame 2 → 3

$$ \begin{bmatrix} \cos(q3), -\sin(q3), 0, 1.25 \\ \sin(q3), \cos(q3), 0, 0 \\ 0, 0, 1, 0 \\ 0, 0, 0, 1 \end{bmatrix} $$

Transform from frame 3 → 4

$$ \begin{bmatrix} \cos(q4), -\sin(q4), 0, -0.054 \\ 0, 0, 1, 1.5 \\ -\sin(q4), -\cos(q4), 0, 0 \\ 0, 0, 0, 1 \end{bmatrix} $$

Transform from frame 4 → 5

$$ \begin{bmatrix} \cos(q5), -\sin(q5), 0, 0 \\ 0, 0, -1, 0 \\ \sin(q5), \cos(q5), 0, 0 \\ 0, 0, 0, 1 \end{bmatrix} $$

Transform from frame 5 → 6

$$ \begin{bmatrix} \cos(q6), -\sin(q6), 0, 0 \\ 0, 0, 1, 0 \\ -\sin(q6), -\cos(q6), 0, 0 \\ 0, 0, 0, 1 \end{bmatrix} $$

Transform from frame 6 → EE

$$ \begin{bmatrix} 1, 0, 0, 0 \\ 0, 1, 0, 0 \\ 0, 0, 1, 0.303 \\ 0, 0, 0, 1 \end{bmatrix} $$

Youssef Khaky
Project 2: Kinematics Pick & Place

A generalized homogeneous transform between base_link and gripper_link using only end-effector(gripper) pose is shown below. The way the rotational part of the matrix was obtained is also shown below.

$R_{End\ Effector}^{Base\ Link} = R_z(\alpha)R_y(\beta)R_x(\gamma)$; α = roll, β = pitch, γ = yaw [1]

$$T_{End\ Effector}^{Base\ link} = \begin{bmatrix} \cos(\alpha)\cos(\beta) & \cos(\alpha)\sin(\beta)\sin(\gamma) - \sin(\alpha)\cos(\gamma) & \cos(\alpha)\sin(\beta)\cos(\gamma) + \cos(\alpha)\cos(\gamma) & p_x \\ \sin(\alpha)\cos(\beta) & \sin(\alpha)\sin(\beta)\sin(\gamma) + \cos(\alpha)\cos(\gamma) & \sin(\alpha)\sin(\beta)\cos(\gamma) - \cos(\alpha)\sin(\gamma) & p_y \\ -\sin(\beta) & \cos(\beta)\sin(\gamma) & \cos(\beta)\cos(\gamma) & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

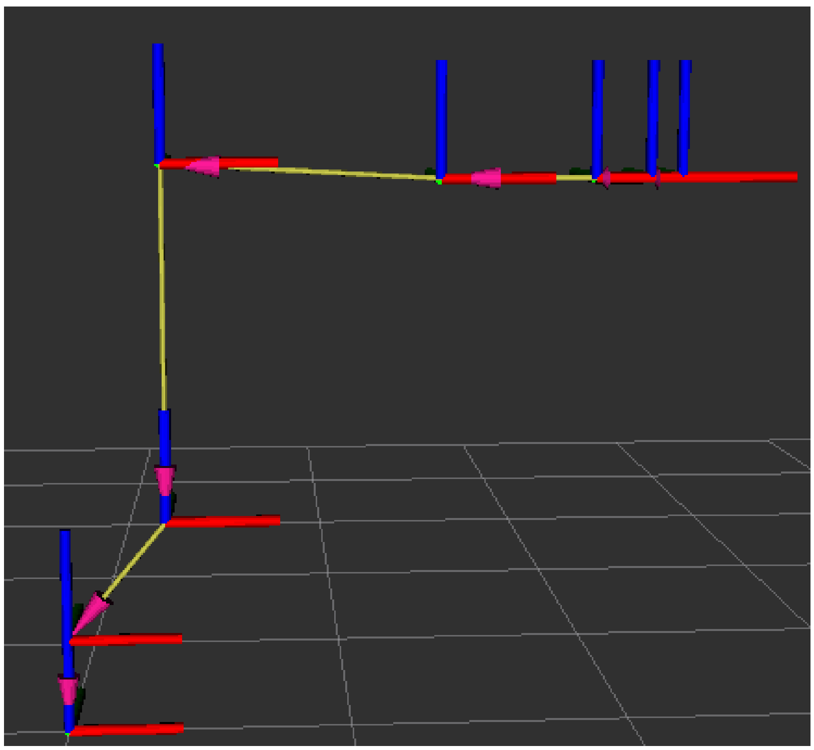


Figure 3 showing the frames for each link including base and gripper

CRITERIA III Inverse Kinematics

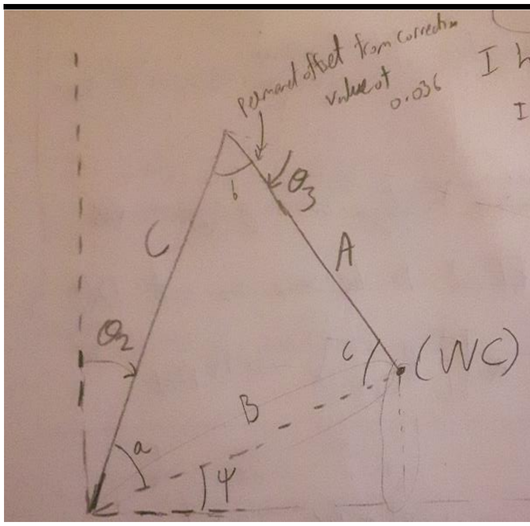


Figure 4 Top down representation of joints 2 and 3 and wrist centre that will aid in explaining the next part

Finding wrist center

Wrist center for this robot is joint 5, it is shaded orange in figure1. Before we decouple the inverse kinematics we first need to find the wrist center (WC) as this will help us dividing the kinematics of the robot into positional and orientational motions.
The WC with respect to the gripper center is simple to locate.

$$[WC] = [EE] - l * \left[R_{rpy}[:,2]\right]$$

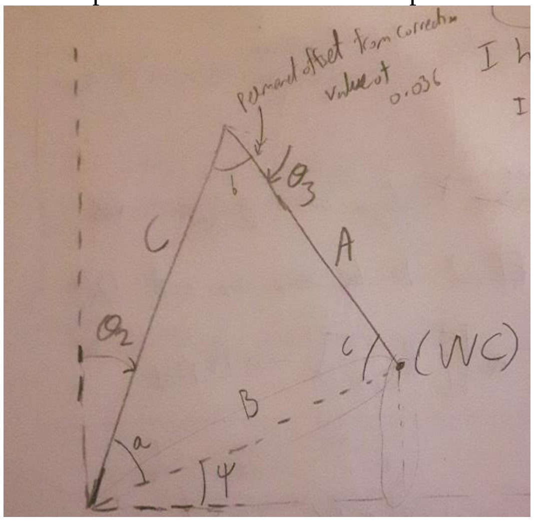
```
EE = Matrix([[px], [py], [pz]])
l = 0.303
nx = Rrpy[0,2]
ny = Rrpy[1,2]
nz = Rrpy[2,2]

wx = px - l*nx #should be the centre of joint#5 / wrist centre
wy = py - l*ny
wz = pz - l*nz
```

Youssef Khaky
Project 2: Kinematics Pick & Place

Decoupling inverse kinematics

Inverse Position Kinematics

Inverse position Kinematics will help us obtain the first 3 joint angles, theta 1, theta 2, theta 3.



Finding Theta1, (not in the above diagram) is just a matter of using atan2.

```
theta1 = atan2(wy,wx)
```

For the next two joints, we correct for the offset to be able to use joint2 as the origin (to make the math easier)

```
#correcting origin to start from joint 2
wx_corr = wx - 0.35*cos(theta1)
wy_corr = wy - 0.35*sin(theta1)
wz_corr = wz - 0.75
```

Next we need to find all sides of the trinagle followed by finding all the angles of the triangle

```
side_a = (1.5**2+0.054**2)**0.5
side_b = (((wx**2+wy**2)**0.5-0.35)**2+(wz-0.75)**2)**0.5
side_c = 1.25

angle_a = acos((side_b**2 + side_c**2 - side_a**2)/(2*side_b*side_c))
angle_b = acos((side_a**2 + side_c**2 - side_b**2)/(2*side_a*side_c))
angle_c = acos((side_a**2 + side_b**2 - side_c**2)/(2*side_a*side_b))
```

After finding all the angles, theta2 and theta3 can be found by using the two equations below.

```
theta2 = pi/2 - angle_a - atan2(wz-0.75,(wx**2+wy**2)**0.5-0.35)
theta3 = pi/2 - angle_b - 0.036
```

Inverse Orientation Kinematics

After we have found the first three joints, we are able to use them and find the remaining three joints.

R03 is the rotation matrix from base link to link3. We can now calculate it using the jont angles theta1, theta2, theta3 that we just previously obtained.

```
R03 = T0_1[0:3, 0:3] * T1_2[0:3, 0:3] * T2_3[0:3, 0:3]
#R03 = R03[0:3, 0:3]
```

Youssef Khaky
Project 2: Kinematics Pick & Place

```
R03 = R03.evalf(subs={q1:theta1,q2:theta2,q3:theta3})
```

Now we need to solve for the last three joints. So, we arrive at a matrix that contains these three terms and isolate for it.

$$R_{EE}^{0} = R_{rpy}$$
$$R_{EE}^{3} = {R_3^0}^T R_{rpy}$$

The right side of the equation is known, so we solve for the matrix on the left

```
R3_EE = (R03.T)*Rrpy
```

After inspecting the symbolic R_{EE}^{3} matrix, we can derive the following equations by matching the sin and the cosine of each angle together and isolating for them and putting them in the atan2 function.

```
theta4= atan2(R3_EE[2,2],-R3_EE[0,2])
theta5= atan2(sqrt(R3_EE[0,2]**2+R3_EE[2,2]**2),R3_EE[1,2])
theta6= atan2(-R3_EE[1,1],R3_EE[1,0])
```

Implantation

The implementation of the project is in the accompanied .py file.

Youssef Khaky
Project 2: Kinematics Pick & Place

References:

[1] http://planning.cs.uiuc.edu/node102.html

Some parts of the code were obtained from the walkthrough video