RoboND Pick & Place Project

July 14, 2018

1 Introduction

In this document the KR210 industrial robot joint angles are computed analytically from the robot end effector position and orientation. This is called the *inverse kinematics* of the robot.

A schematic of the robot with the coordinate systems (COS) placed according to Denavit-Hartenberg convention can be seen in Figure 1. The corresponding DH-Table is shown next. It is already filled with the values from the joints section of the "kr210.urdf.xacro" file.

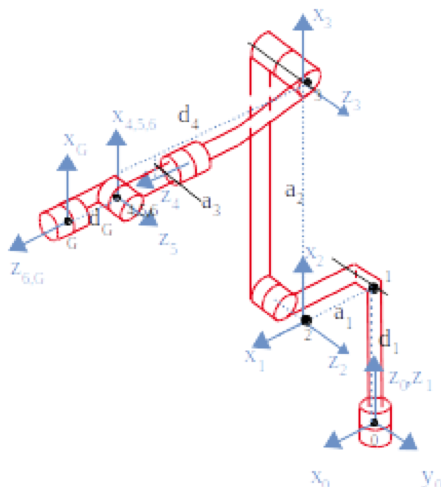

Figure 1: KR210 with coordinate systems according to DH-convention

$$\begin{bmatrix} \alpha_0 : 0 & a_0 : 0 & d_1 : 0.33 + 0.42 = 0.75 & \\ \alpha_1 : -\frac{\pi}{2} & a_1 : 0.35 & d_2 : 0 & q_2 : q_2 - \pi/2 \\ \alpha_2 : 0 & a_2 : 1.25 & d_3 : 0 & \\ \alpha_3 : -\frac{\pi}{2} & a_3 : -0.054 & d_4 : 0.96 + 0.54 = 1.5 & \\ \alpha_4 : \frac{\pi}{2} & a_4 : 0 & d_5 : 0 & \\ \alpha_5 : -\frac{\pi}{2} & a_5 : 0 & d_6 : 0 & \\ \alpha_6 : 0 & a_6 : 0 & d_G : 0.11 + 0.193 = 0.303 & q_G : 0 \end{bmatrix} \tag{1}$$

As can be seen from the table, the DH coordinate systems are not always coincident with the joint locations from the urdf file. Therefore d_1, d_4 and d_G have to be calculated from the joint locations of the urdf file. The homogeneous transformation from i to i+1 COS is defined as follows:

$$T_{i,i+1} = \begin{pmatrix} \cos q_{i+1} & -\sin q_{i+1} & 0 & a_i \\ \sin q_{i+1}\cos\alpha_i & \cos q_{i+1}\cos\alpha_i & -\sin\alpha_i & -d_{i+1}\sin\alpha_i \\ \sin q_{i+1}\sin\alpha_i & \cos q_{i+1}\sin\alpha_i & \cos\alpha_i & d_{i+1}\cos\alpha_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2}$$

With eq. 2 and the values from the DH-table it is then possible to calcuate the relationship between joint positions and position and orientation of the robot links w.r.t the inertial reference frame. This is called the robot forward kinematics and it shall be used in the following to analytically derive the inverse kinematics of the robot. The individual transformation matrices can be seen in the appendix.

As the orientation of the robot gripper in the urdf file and therefore in the gazebo simulator differs from the gripper position and orientation as computed with eqs. 2 a correction has to be introduced. This is achieved by multiplication with the homogeneous transformation matrix $T_{U,G}$ between gripper "G" and urdf "U"-COS. The transformation consists of a rotation around the y-axis by $-\pi/2$ followed by a rotation around the z'-axis by π. Thus follows:

$$T_{U,G} = \begin{pmatrix} \cos -\frac{\pi}{2} & 0 & \sin\frac{\pi}{2} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\frac{\pi}{2} & 0 & \cos\frac{\pi}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\pi & -\sin\pi & 0 & 0 \\ \sin\pi & \cos\pi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3}$$

2 Inverse Kinematics

The idea behind the calculation of the inverse kinematics of the robot is that the first three joints are used to control the position of the robot wrist and the remaining three joints control the wrist orientation.

The joints $q_{1,2,3}$ which are then responsible for the wrist position shall be calculated first. From the - externally provided - position ${}_0p_G$ and orientation $(\varphi, \vartheta, \psi)$[1] of the robot gripper COS ("G" in Figure 1) the robot wrist position ${}_0p_{O_6}$ can be calculated as follows[2]:

$${}_0p_{O_6} = {}_0p_{O_G} - R_{0,G}\begin{pmatrix} 0 \\ 0 \\ d_G \end{pmatrix} \tag{4}$$

$R_{0,G}$ here is the rotation from gripper to inertial COS. It can be calculated as follows:

$$R_{0,G} = R_{0,U}R_{U,G} \tag{5}$$

[1] φ: rotation around 0-COS x-axis, ϑ: rotation around 0-COS y-axis, ψ: rotation around 0-COS z-axis

[2] the subscript prior to the vectors gives the reference frame in which the respective vector is calculated

$R_{U,G}$ is the rotation component of the homogeneous transformation $T_{U,G}$ from eq. 3. $R_{0,U}$ is the rotation matrix between urdf and inertial COS and can be calculated from the external gripper orientation angles:

$$R_{0,U} = R_z(\psi)\, R_y(\vartheta)\, R_x(\varphi) \tag{6}$$

Please note the order of the multplication (from "yaw" to "roll"). This is due to the extrinsic nature of the transformation.

With ${}_0p_{O_6}$ now q_1 to q_3 can be calculated. For q_1 the calculation is straightforward:

$$q_1 = atan2\left({}_0p_{O_6,y},\ {}_0p_{O_6,x}\right) \tag{7}$$

For the calculation of q_2 and q_3 Figure 2 is used: First the length of the side B

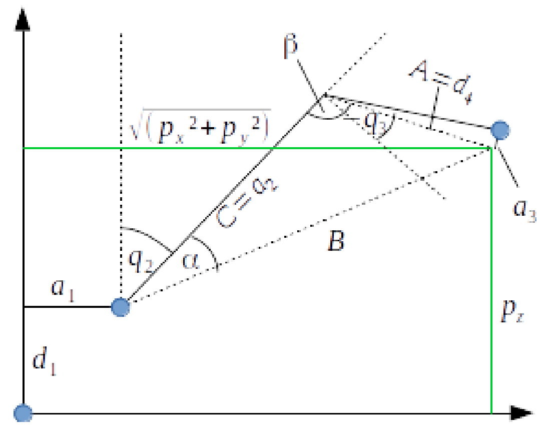

Figure 2: calculation of q_2 and q_3

from the triangle "ABC" is calculated:

$$B = \sqrt{\left(\sqrt{{}_0p^2_{O_6,x} + {}_0\, p^2_{O_6,y}} - a_1\right)^2 + \left({}_0p_{O_6,z} - d_1\right)^2} \tag{8}$$

By using the cosine theorem now α and β from Figure 2 can be calculated:

$$\alpha = acos\frac{B^2 + C^2 - A^2}{2BC}; \quad \beta = acos\frac{A^2 + C^2 - B^2}{2AC}; \tag{9}$$

q_2 and q_3 then are:

$$q_2 = \frac{\pi}{2} - \alpha - atan2\left(\left({}_0p_{O_6,z} - d_1\right), \left(\sqrt{{}_0p^2_{O_6,x} + {}_0\, p^2_{O_6,y}} - a_1\right)\right) \tag{10}$$

$$q_3 = \frac{\pi}{2} - \left(\beta + atan\left(a_3/d_4\right)\right) \tag{11}$$

With these values known now the rotation matrix from COS 3 to G can be computed numerically:

$$R_{3,G} = R_{3,0} R_{0,G} \tag{12}$$

As this rotation is achieved by a composition of elementary rotations around the joint angles q_4 to q_6, the joint angles shall be resolved from $R_{3,G}$ in the following. To do this, $T_{3,G} = T_{3,4}T_{4,5}T_{5,6}T_{6,G}$ is derived analytically by applying eq. 2 (with "s" for *sine* and "c" for *cosine*). The rotation component of $T_{3,G}$ is shown in the following.

$$R_{3,G} = \begin{pmatrix} -sq_4sq_6 + cq_4cq_5cq_6 & -sq_4cq_6 - cq_4cq_5sq_6 & -cq_4sq_5 \\ sq_5cq_6 & -sq_5sq_6 & cq_5 \\ -sq_4cq_5cq_6 - cq_4sq_6 & sq_4cq_5sq_6 - cq_4cq_6 & sq_4sq_5 \end{pmatrix} \tag{13}$$

q_5 is now computed directly from eq. 13 ($R_{3,G}(x,y)$ stands for the component in the x-th row and y-th column of $R_{3,G}$):

$$q_5 = atan2\left(\sqrt{R_{3,G}(1,3)^2 + R_{3,G}(3,3)^2}, R_{3,G}(2,3)\right) \tag{14}$$

For the computation of q_4 now the atan2 function is applied to the fraction of row 3, column 3 with row one, column one. As sq_5 can be reduced from the equation it is possible to compute q_4 in this fashion. However care has to be taken whether q_5 is greater or smaller than zero[3]. This gives the two solutions for q_4:

$$q_4 = atan2\left(R_{3,G}(3,3), -R_{3,G}(1,3)\right); \quad sq_5 > 0 \tag{15}$$
$$= atan2\left(-R_{3,G}(3,3), R_{3,G}(1,3)\right); \quad sq_5 < 0 \tag{16}$$

The same argumentation can be applied for q_6 and finally gives:

$$q_6 = atan2\left(-R_{3,G}(2,2), R_{3,G}(2,1)\right); \quad sq_5 > 0 \tag{17}$$
$$= atan2\left(R_{3,G}(2,2), -R_{3,G}(2,1)\right); \quad sq_5 < 0 \tag{18}$$

Now the inverse kinematics of the KR210 robot is finally complete and can be implemented. This is done in the script "IK_server". A screenshot of the robot when controlled with the script can be seen in Figure 3.

[3]I found the solution in a slack posting

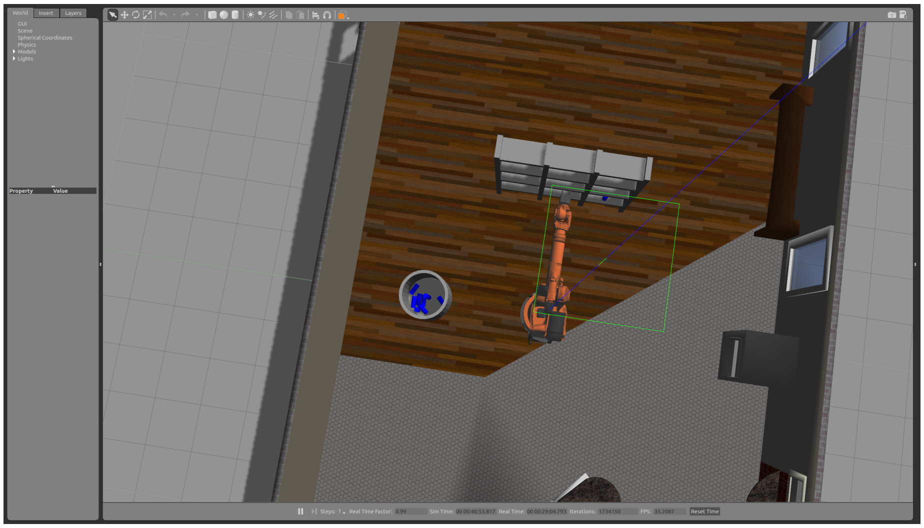

Figure 3: Robot during pick & place operation

3 Appendix

$$T_{0,1} = \begin{bmatrix} cq_1 & -sq_1 & 0 & 0 \\ sq_1 & cq_1 & 0 & 0 \\ 0 & 0 & 1 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{19}$$

$$T_{1,2} = \begin{bmatrix} sq_2 & 0 & cq_2 & 0.35 \\ 0 & 0 & 1 & 0 \\ cq_2 & 0 & -sq_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{20}$$

$$T_{2,3} = \begin{bmatrix} cq_3 & -sq_3 & 0 & 1.25 \\ sq_3 & cq_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}$$

$$T_{3,4} = \begin{bmatrix} cq_4 & -sq_4 & 0 & -0.054 \\ 0 & 0 & 1 & 1.5 \\ -sq_4 & -cq_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{22}$$

$$T_{4,5} = \begin{bmatrix} cq_5 & -sq_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ sq_5 & cq_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{23}$$

$$T_{5,6} = \begin{bmatrix} cq_6 & -sq_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -sq_6 & -cq_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

$$T_{6,G} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.303 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{25}$$

$T_{0,G}$ can be obtained by multiplying $T_{0,1}T_{1,2}T_{2,3}T_{3,4}T_{4,5}T_{5,6}T_{6,G}$. As the expression is quite complex to type in latex it will be omitted here.