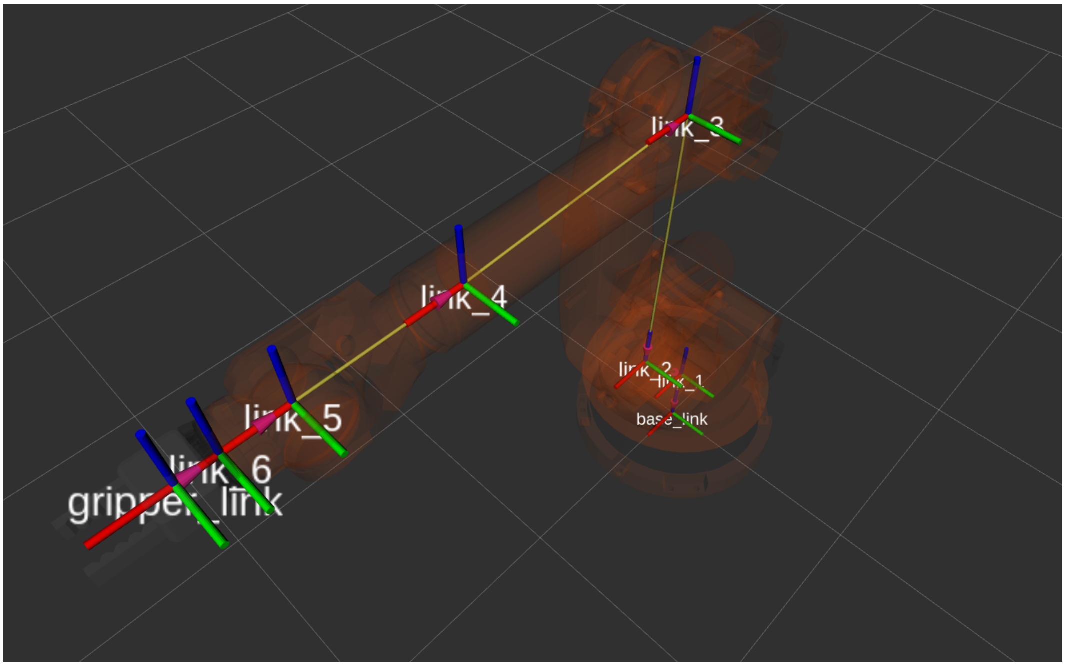
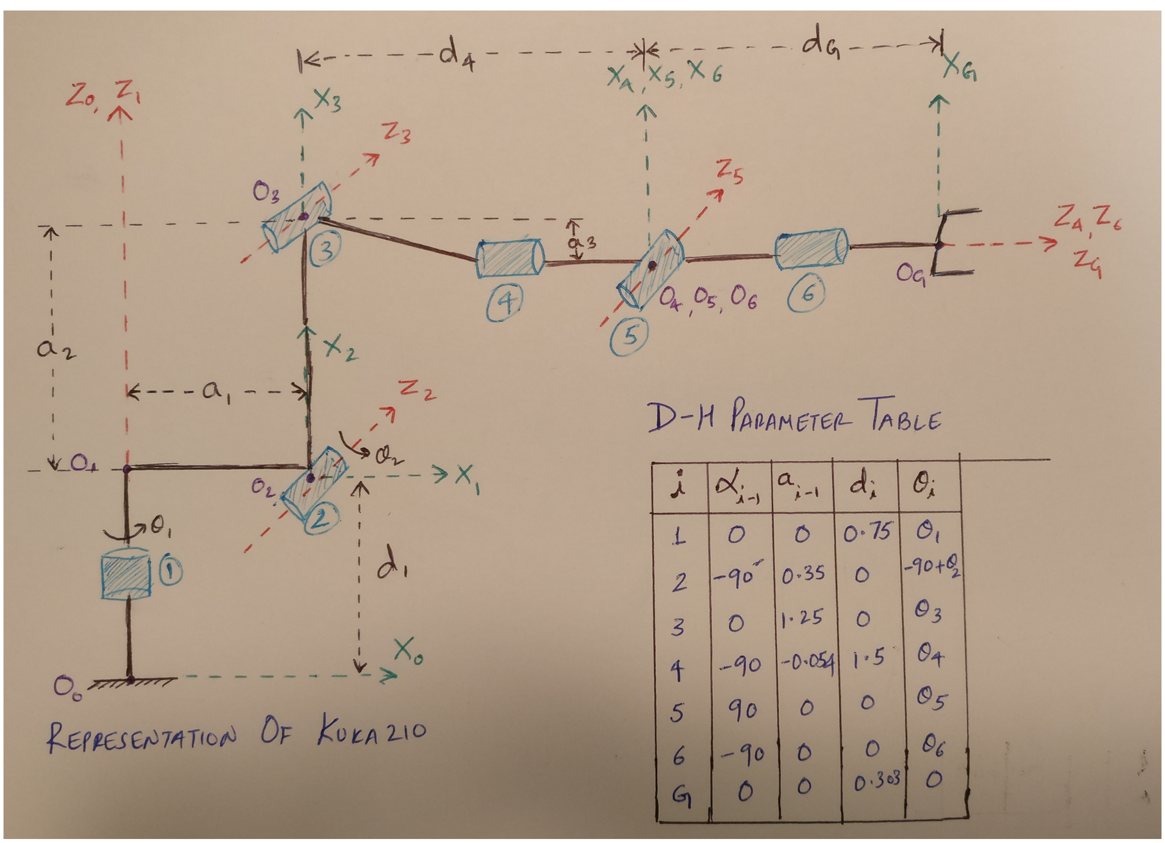
Robotic Arm: Pick & Place Project Report

- Mohit Chaturvedi

Forward Kinematics & D-H Parameters

Forward Kinematics demo and evaluating kr210.urdf.xacro file to derive DH parameters
From Rviz visualization, it's seen that the robot arm essentially lies in the X-Z plane.





i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0.75	θ_1
2	-90°	0.35	0	$-90+\theta_2$
3	0	1.25	0	θ_3
4	-90	-0.054	1.5	θ_4
5	90	0	0	θ_5
6	-90	0	0	θ_6
G	0	0	0.303	0

Steps towards sketching robot figure in zero configuration with link assignments and joint rotations:

1. Label all the joints from 1 to 6 and add an extra frame as the gripper frame rigidly attached to link 6.
2. Label all the links from 0 (fixed base_link) to 6 (link_6 joining the end-effector).
3. Draw dotted line defining all joint axes as Z axes.
   Z_0 and Z_1 are co-linear. Z_4 , Z_6 and Z_G are co-linear. Z_2, Z_3 and Z_5 are parallel.
4. Identify common normals between each frame Z_{i-1} and frame Z_i.
5. Assigning the direction of positive Z axis.
   As per intuition and following the convention of counter-clockwise rotation as positive, Z axis is defined positive in upward direction when viewed from top.
6. Define positive X axis for all the the intermediate links, excluding the base and end effector.
   For Z axes that are parallel there are infinitely many common normals between them.
   For Z_1 and Z_2 the positive X_1 axis points along the common normal from Z_1 to Z_2 from origin O_1. The position of origin O_1 is chosen so as to make (d2) parameter zero.
   Similarly, X_2 axis points along the common normal between Z_2 and Z_3 in the direction from Z_2 to Z_3.
   From the two directions available for X_3 (up and down), upward direction is chosen as positive so as to be consistent with the positive direction of X_2. Since these two axes are coincident, this makes (d3) zero.
   The same rule applies for the rest of the X axes. The positions of X_4, X_5, X_6 and X_G is so chosen as to make d5 and d6 zero and also to minimize the need for extra (d) parameters towards the end-effector.
7. Assign the X axis of base link, link_0.
   Here, positive direction of X_0 is chosen to be in the same direction as X_1 with origin O_0 chosen at base and not coincident with O_1 in order to make the frames distinct and the representation cleaner to understand. Keeping both origins at the same point would make both co-ordinate frames to be exact.
8. Assigning the X axis to the end effector (gripper).
   Following the DH rule, X_G is chosen parallel to the previous link i.e. X_6 axis with the origin O_G at the center of the gripper frame.

DH parameter table:

1. α_i is the twist angle between measured about X_{n-1} axis according to the right hand rule.
   Twist angle for parallel Z axes is zero. For non parallel Z axes, it is measured as per the right hand rule. Hence, α_0 is zero and α_1 is -90 deg and so on and so forth.
2. a_i (link length) is the distance from Z_{n-1} to Z_n measured along X_{n-1} axis.
   This parameter is obtained from the URDF file for kuka210. The values are calculated keeping in mind that URDF reference frame axes do not line up with DH reference frame axes. The origins and orientations of joints in URDF are different from those as defined for DH parameters. For example, position of joint 1 origin is different in URDF as compared to one that is defined in the sketch for DH parameters.
3. d_i (link_offset) is the sign distance from X_{n-1} to X_n measured along Z_n axis.

This parameter is also obtained from the URDF file compensating for the difference between URDF and DH reference frame axes.

4. θ_i is the angle between X_{n-1} and X_n about the Z_n axis in the right hand sense.

All the joints are revolute and their corresponding θ values are non-zero. In case of θ_2 since X_1 X_2 are not parallel when θ_2 is zero, using the right hand rule to compensate for the angle between the two X axes, -90 deg is added to θ_2.

Individual Transform Matrices between links

Following Craig's modified D-H convention, the individual homogeneous transformation about each joint is represented as a product of four basic transformations, 2 rotations and 2 translations in the following manner:

Transform for each link is made up of 2 rotations & 2 translations

$$ {}^{i-1}_{\;\;i}T = R_x(\alpha_{i-1})\, D_x(a_{i-1})\, R_z(\theta_i)\, D_z(d_i) $$

This gives us

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

Calculating individual transforms for all the links with the D-H parameters derived earlier:

$$
{}^{0}_{1}T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{1}_{2}T = \begin{bmatrix} \sin\theta_2 & \cos\theta_2 & 0 & 0.35 \\ 0 & 0 & 1 & 0 \\ \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{2}_{3}T = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & 1.25 \\ \sin\theta_3 & \cos\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{3}_{4}T = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & -0.054 \\ 0 & 0 & 1 & 1.50 \\ -\sin\theta_4 & -\cos\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{4}_{5}T = \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{5}_{6}T = \begin{bmatrix} \cos\theta_6 & -\sin\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin\theta_6 & -\cos\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{6}_{G}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.303 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Homogeneous Transform between base_link and gripper_link using only end-effector(gripper) pose

The total homogeneous transform between the base_link and the gripper_link is built incrementally by post multiplying individual homogeneous transform in the following fashion:

$$
{}^{0}_{G}T = {}^{0}_{1}T \times {}^{1}_{2}T \times {}^{2}_{3}T \times {}^{3}_{4}T \times {}^{4}_{5}T \times {}^{5}_{6}T \times {}^{6}_{G}T
$$

However, an orientation correction is applied to account for the difference in orientation of the gripper link frame as defined in the URDF vs the D-H convention. In order to get the two frames to align, a sequence of body-fixed (intrinsic) rotations are applied to the gripper frame. The frame is first rotated along the Z axis by 180degrees and then about the Y axis by -90degrees. The result is a rotation correction matrix which is a composition of these two rotations. The correction matrix is then applied to get the total homogeneous transform between the base link and the gripper link.

$$ROT\_z = \begin{bmatrix} \cos(180) & -\sin(180) & 0 & 0 \\ \sin(180) & \cos(180) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$ROT\_y = \begin{bmatrix} \cos(-90) & 0 & \sin(-90) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(-90) & 0 & \cos(-90) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot\_Error = ROT\_z \times ROT\_y$$

$$T\_total = {}^{0}_{G}T \times Rot\_Error$$

Inverse Kinematics

The last three joints(4, 5 and 6) being revolute with their respective joint axes intersecting at a single point (joint 5) makes it a spherical wrist and makes joint 5 as the wrist center. Thus we can decouple the IK problem into a separate Inverse position and a separate Inverse Orientation problem.

Inverse Position

The position of wrist center can be obtained using the total homogeneous transform between the base link and the gripper link based on the gripper pose. With given gripper positions as P_x , P_y and P_z and considering n as an orthonormal vector of gripper orientation along the Z axis, wrist positions can be represented as following:

$$w_x = p_x - (d_6 + l) \cdot n_x$$
$$w_y = p_y - (d_6 + l) \cdot n_y$$
$$w_z = p_z - (d_6 + l) \cdot n_z$$

Here, *l* is the gripper length and d_6 is the D-H parameter.

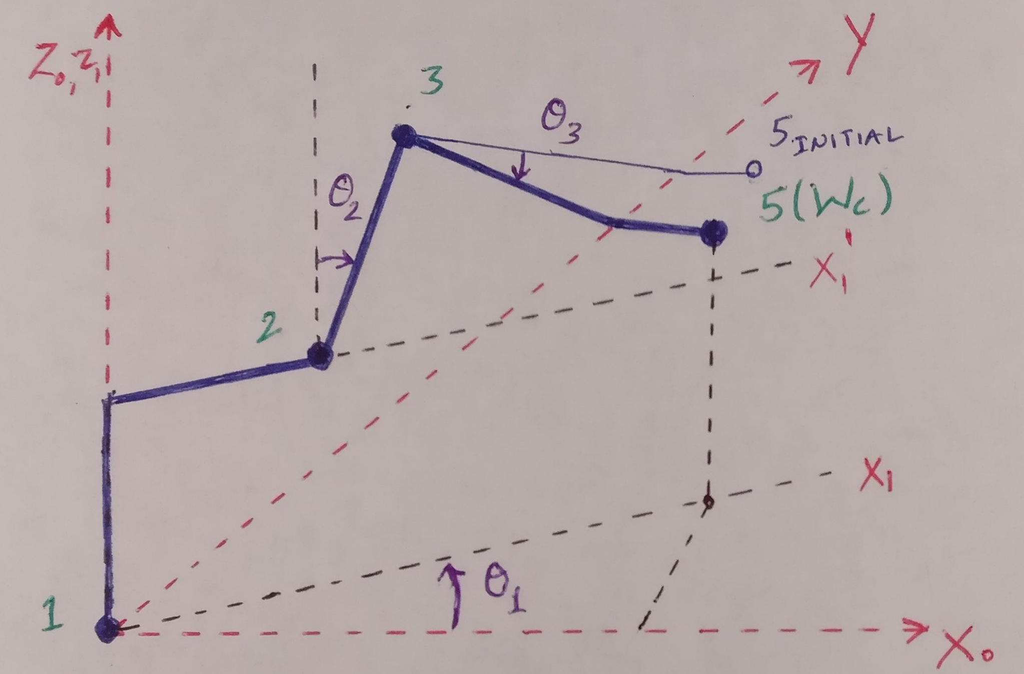
n_x , n_y and n_z are extracted from the rotation matrix determining the gripper pose with respect to the base link. Using the x-y-z extrinsic rotations convention to transform from one fixed frame to another, the rotation matrix is represented as

ROT_EE = ROT_z (yaw) * ROT_y (pitch) * ROT_x (roll) * Rot_Error

EE = Matrix([[px], [py], [pz]])
WC = EE - (0.303) * ROT_EE[:,2]

Once the wrist center position is calculated, the next step is to calculate joint angles for first 3 joints.

$\theta_1 \rightarrow$ Joint angle between X_0 & X_1 measured about Z_1 axis in right hand rule fashion.

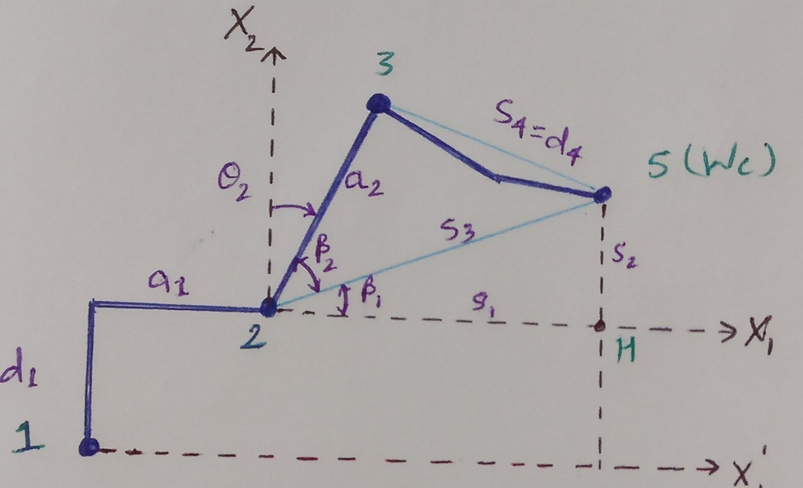


$\theta_2 \rightarrow$ Joint angle between X_1 & X_2 measured about Z_2

$\theta_3 \rightarrow$ Joint angle between X_2 & X_3 measured about Z_3

For calculating θ_1, we project W_c on the ground plane setting it's z coordinate zero. θ_1 can then be calculated as,

$$\theta_1 = \operatorname{atan2}(Wc_y, Wc_x)$$



To calculate θ_2, we visualize it from a different perspective so that the 90° angle between X_1 & X_2 is clearly seen. We draw a triangle between joints 2, 3 & 5 with sides a_2, S_4 and S_3. Now, θ_2 can be calculated as $\boxed{\theta_2 = \pi/2 - \beta_1 - \beta_2.}$

From the trigonometric function, $\boxed{\beta_1 = \operatorname{atan2}(s_2, s_1)}$

Here, $s_2 = Wc_z - d_1$ $\quad$ Wc_z is the z component of Wc position

$s_1 = \sqrt{(Wc_x)^2 + (Wc_y)^2} - a_1$ $\quad$ a_1, d_1 is obtained from D-H table

Now using cosine law β_2 can be obtained as

$$S_4^2 = S_3^2 + a_2^2 - 2 S_3 a_2 \cos\beta_2$$

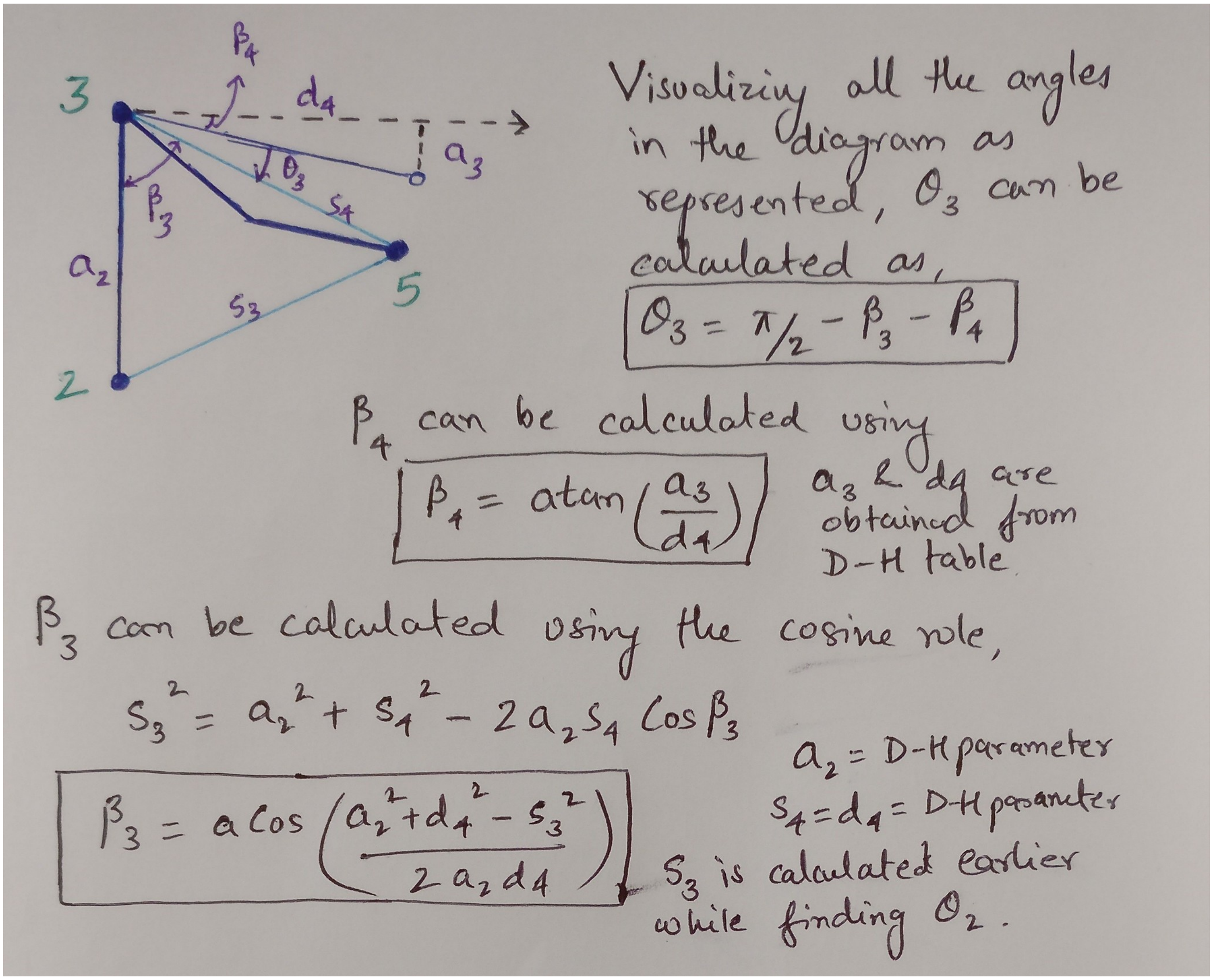
$$\boxed{\beta_2 = a\cos\left(\frac{S_3^2 + a_2^2 - d_4^2}{2 S_3 a_2}\right)}$$

where,

$S_3 = \sqrt{s_1^2 + s_2^2}$

a_2 = D-H parameter

$S_4 = d_4$ = D-H parameter



Inverse Orientation

For the inverse orientation part, we need to find out the values of final three joint variables.

From the D-H individual transforms, the resultant rotation can be obtained as

From the D-H individual transforms obtained earlier, a resultant rotation matrix can be calculated as,

$$ {}^{0}_{6}R = {}^{0}_{1}T \times {}^{1}_{2}T \times {}^{2}_{3}T \times {}^{3}_{4}T \times {}^{4}_{5}T \times {}^{5}_{6}T $$

Now, the rotation matrix for gripper obtained earlier by using x-y-z extrinsic rotations convention (ROT_EE), should equate to ${}^{0}_{6}R$.

ie $ {}^{0}_{6}R = ROT_EE $

We have calculated the first three joint angles θ_1, θ_2 and θ_3. These values along with others can be substituted to their respective individual transform matrices, giving us ${}^{0}_{3}R$. where,

$$ {}^{0}_{3}R = {}^{0}_{1}T \times {}^{1}_{2}T \times {}^{2}_{3}T $$

Pre-Multiplying the earlier rotation matrix equation by $inv({}^{0}_{3}R)$ gives us,

$$ {}^{3}_{6}R = inv({}^{0}_{3}R) \times ROT\_EE $$

After substituting θ_1, θ_2 & θ_3 values, the R.H.S of the equation has no unknown variables. Hence, equating the L.H.S to R.H.S will provide the values for θ_4, θ_5 & θ_6 through trigonometric equations contained in ${}^{3}_{6}R$.

A symbolic ${}^3_6R$ can be created as,

$${}^3_6R = {}^3_4T \times {}^4_5T \times {}^5_6T$$

Using python for matrix multiplication, ${}^3_6R$ looks like this:

$$\begin{bmatrix} -S\theta_4 S\theta_6 + C\theta_4 C\theta_5 C\theta_6 & -S\theta_4 C\theta_6 - S\theta_6 C\theta_4 C\theta_5 & -S\theta_5 C\theta_4 \\ S\theta_5 C\theta_6 & -S\theta_5 S\theta_6 & C\theta_5 \\ -S\theta_4 C\theta_5 C\theta_6 - S\theta_6 C\theta_4 & S\theta_4 S\theta_6 C\theta_5 - C\theta_4 C\theta_6 & S\theta_4 S\theta_5 \end{bmatrix}$$

Utilizing the atan function, θ_4, θ_5 & θ_6 can now be obtained as,

$$\theta_4 = \operatorname{atan}\left({}^3_6R[2,2],\ -{}^3_6R[0,2]\right)$$

$$\theta_5 = \operatorname{atan}\left(\sqrt{{}^3_6R[0,2]^2 + {}^3_6R[2,2]^2},\ {}^3_6R[1,2]\right)$$

$$\theta_6 = \operatorname{atan}\left(-{}^3_6R[1,1],\ {}^3_6R[1,0]\right)$$

Project

One of the factors that contributed to better and faster computations was to move the Forward kinematics code block outside of the for_loop provided in the default python script. Another change that improved results was replace using “simplify” with substitution. This reduced the time for “calculating inverse kinematics” step in the simulation considerably.

Replacing inverse LU decomposition with transpose. It works in this particular case since we are inverting a rotation matrix which is by definition symmetric. LU decomposition was computation heavy and probably introduced some inaccuracies which using atan2 made the results highly inaccurate and slower to compute.