Mini- Project 3: Spatial Robot Arm Motion: Rigid Body Motion in *SE(3),* Forward Kinematics, Inverse Kinematics

Robotics 1
Varun Dhir
MANE 4560
10/19/21
https://github.com/Varun-ABC/Robotics1/tree/main/mini_project_3

I, Varun Dhir, certify that the following work is my own and completed in accordance with the academic integrity policy as described in the Robotics I course syllabus.

Summary:

The ABB 1200/ 0.9 is a 6 joint arm that has 6 degrees of freedom. For any desired position (Xt, Yt, Zt) it can have a desired orientation as well, allowing a lot of control over position as may be needed in applications such as manufacturing or robotic surgery. Within this framework, kinematics of the arm can be derived in many ways.

The forward kinematics are useful when we have a set of joint angles and want to know the position of the arm.

The inverse kinematics are useful when we have a target pose and want a set of joint angles.

The approach taken was based on Product of Exponentials parameters. However, Standard Denavit–Hartenberg and Modified Denavit–Hartenberg parameters were also explored for this application.

The velocity of the robot and the path can also be constrained using the path velocity and/ or joint velocity constraints.

Technical Content:

Problem 1: Poses and Representation

Derive and represent the target robot end effector pose, in the following representations, along the curve using the Cartesian positions and orientation of N equally spaced points along the curve (N=101).

(a) Unit quaternion, $\mathsf{q} = \begin{bmatrix} \cos\frac{\theta}{2} \\ \left(\sin\frac{\theta}{2}\right) k \end{bmatrix}$

(b) yaw-pitch-roll ZYX (or roll-pitch-yaw xyz) Euler angles, $(\beta_1, \beta_2, \beta_3)$:

$$R = \mathsf{R}_z(\beta_1)\mathsf{R}_y(\beta_2)\mathsf{R}_x(\beta_3).$$

(c) Angle-axis product, $\beta = k\theta$.

Derivations (Approach)

Euler angles from R:

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

$C(B_1) = \cos(B_1)$
$S(B_1) = \sin(B_1)$

$$R_x(B_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C(B_1) & -S(B_1) \\ 0 & S(B_1) & C(B_1) \end{bmatrix}$$

$$R_{\cancel{y}}(B_2) = \begin{bmatrix} C(B_2) & 0 & S(B_2) \\ 0 & 1 & 0 \\ -S(B_2) & 0 & \cos(B_2) \end{bmatrix}$$

$$R_z(B_3) = \begin{bmatrix} C(B_3) & -S(B_3) & 0 \\ S(B_3) & C(B_3) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R = R_z R_y R_x \Rightarrow \begin{bmatrix} C(B_2)C(B_3) & SB_2CB_3SB_1 + CB_1SB_3 & SB_2CB_3CB_1 + SB_1SB_3 \\ CB_2SB_3 & SB_2CB_3SB_1 + CB_1CB_3 & SB_2CB_3CB_1 + SB_1CB_3 \\ -SB_2 & CB_2SB_3 & CB_2CB_3 \end{bmatrix}$$

Thus B_1, B_2, B_3 can be found:

$B_2 = -a\sin(R_{31})$

$B_1 = \operatorname{atan2}(R_{32}, R_{33})$

$B_3 = \operatorname{atan2}(R_{21}, R_{11})$

deriving Quaterions From Rotation matrix

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$\varepsilon_1 = \cos\frac{\theta}{2}, \quad \varepsilon_2 = n_x \sin\frac{\theta}{2}, \quad \varepsilon_3 = n_y \sin\frac{\theta}{2}, \quad \varepsilon_4 = n_z \sin\frac{\theta}{2}$$

$$\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 + \varepsilon_4^2 = 1$$

Thus R can be written as

$$R = \begin{bmatrix} \varepsilon_1^2 + \varepsilon_2^2 - \varepsilon_3^2 - \varepsilon_4^2 & 2(\varepsilon_2\varepsilon_3 - \varepsilon_1\varepsilon_4) & 2(\varepsilon_2\varepsilon_4 + \varepsilon_1\varepsilon_3) \\ 2(\varepsilon_2\varepsilon_3 + \varepsilon_1\varepsilon_4) & \varepsilon_1^2 - \varepsilon_2^2 + \varepsilon_3^2 - \varepsilon_4^2 & 2(\varepsilon_2\varepsilon_4 - \varepsilon_1\varepsilon_2) \\ 2(\varepsilon_2\varepsilon_4 - \varepsilon_1\varepsilon_3) & 2(\varepsilon_3\varepsilon_4 + \varepsilon_1\varepsilon_4) & \varepsilon_1^2 - \varepsilon_2^2 - \varepsilon_3^2 + \varepsilon_4^2 \end{bmatrix}.$$

thus:

$$\varepsilon_1 = \frac{1}{2}\sqrt{1 + r_{11} + r_{22} + r_{33}}$$

$$\varepsilon_2 = \frac{1}{2}\sqrt{\frac{(r_{32} - r_{23})^2 + (r_{12} + r_{21})^2 + (r_{23} + r_{32})^2}{3 + r_{11} + r_{22} + r_{33}}}$$

$$\varepsilon_3 = \frac{1}{2}\sqrt{\frac{(r_{13} - r_{31})^2 + (r_{12} + r_{21})^2 + (r_{23} + r_{32})^2}{3 + r_{11} - r_{22} + r_{33}}}$$

$$\varepsilon_4 = \frac{1}{2}\sqrt{\frac{(r_{21} - r_{12})^2 + (r_{31} + r_{13})^2 + (r_{32} + r_{23})^2}{3 + r_{11} + r_{22} - r_{33}}}$$

Angle Axis

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

The axis of rotation is a line that remains unchanged by rotation & the $\measuredangle$ is always $\perp$ to the axis.

The angle can be found by

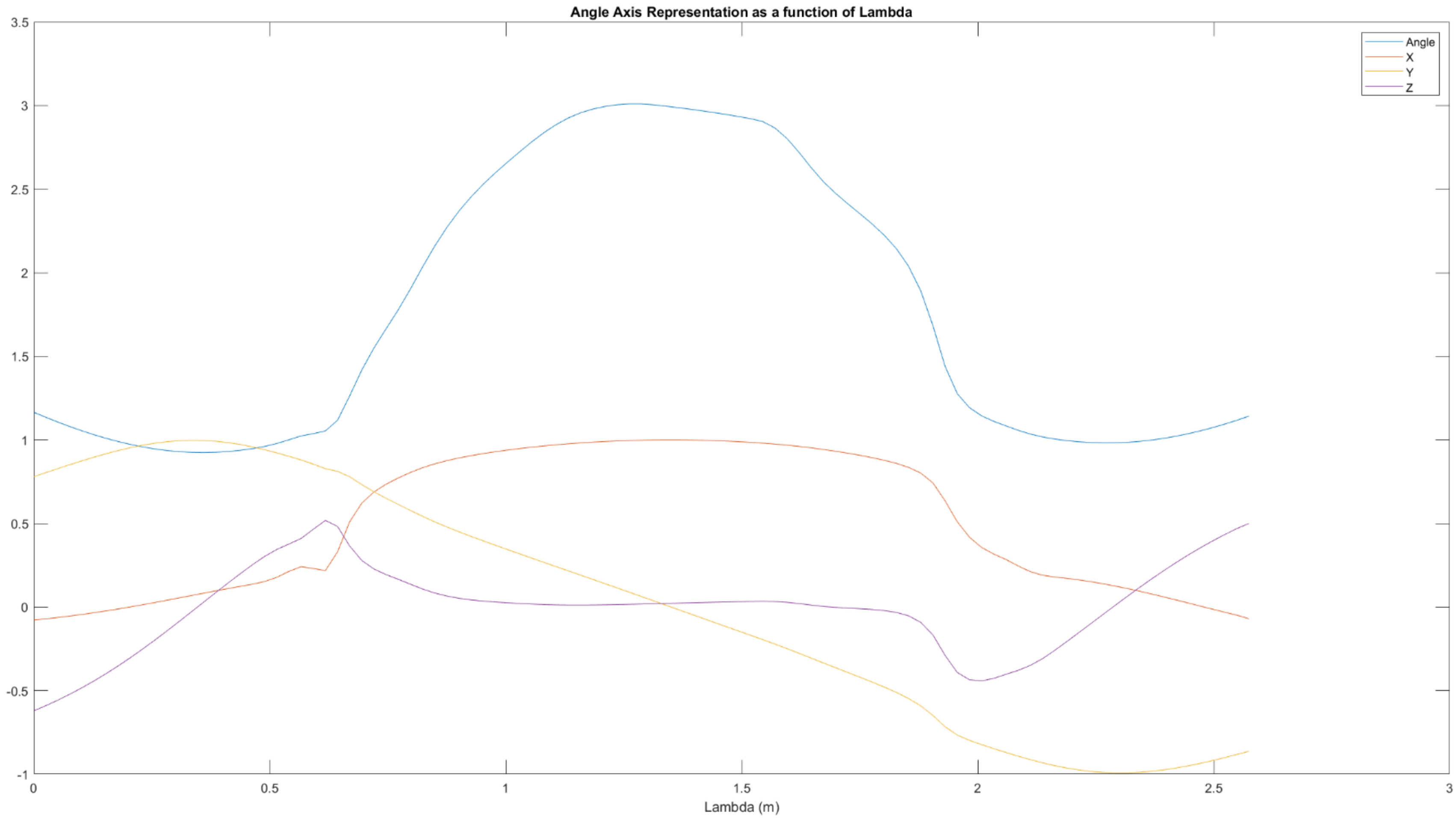
$$1 + 2\cos\theta = \text{Trace}(R)$$

$$\boxed{\theta = a\cos\left(\frac{\text{Tr}(R) - 1}{2}\right)}$$

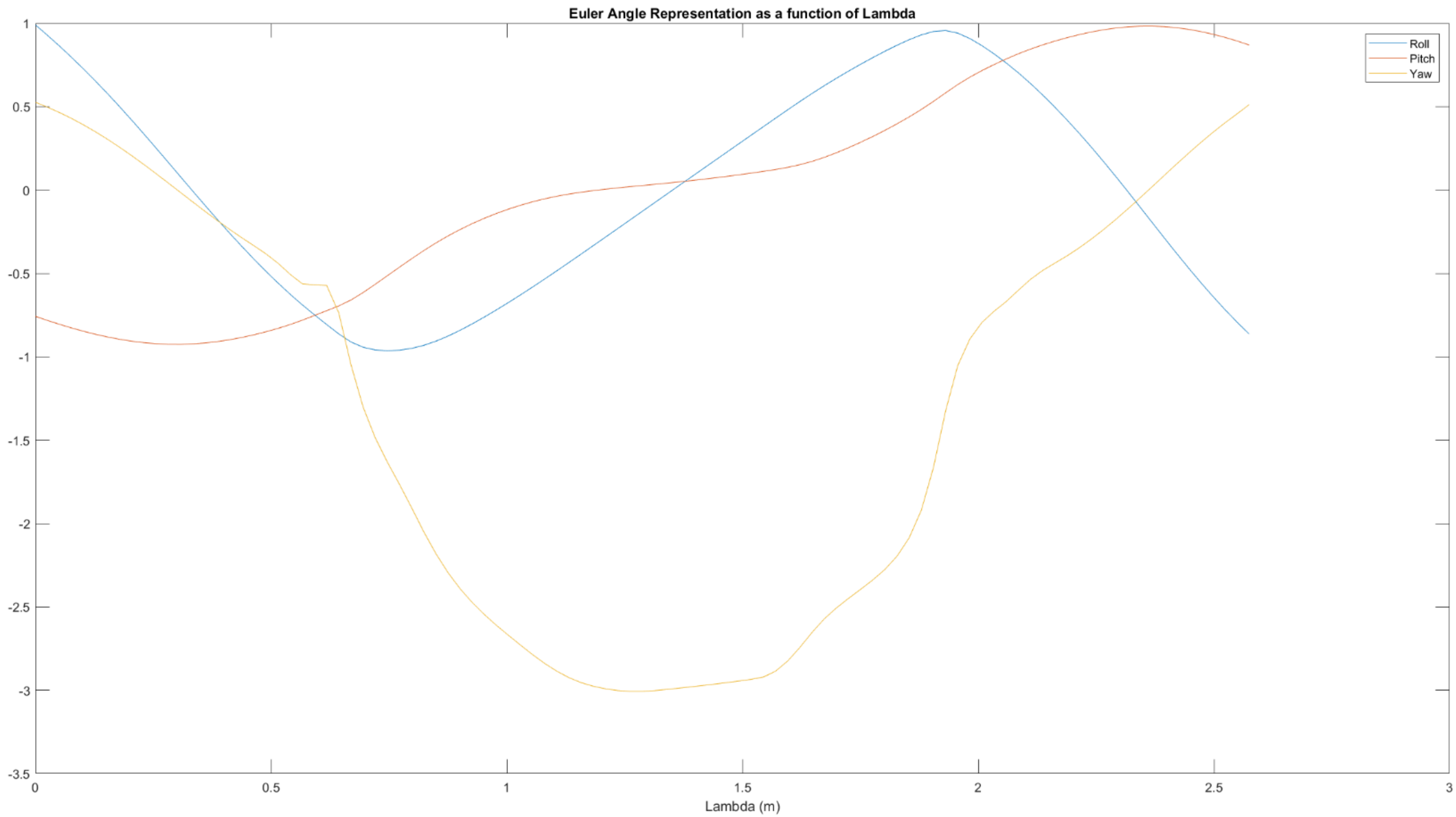
The axis can then be found by

$$\boxed{\text{axis} = \frac{1}{2\sin\theta}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}}$$

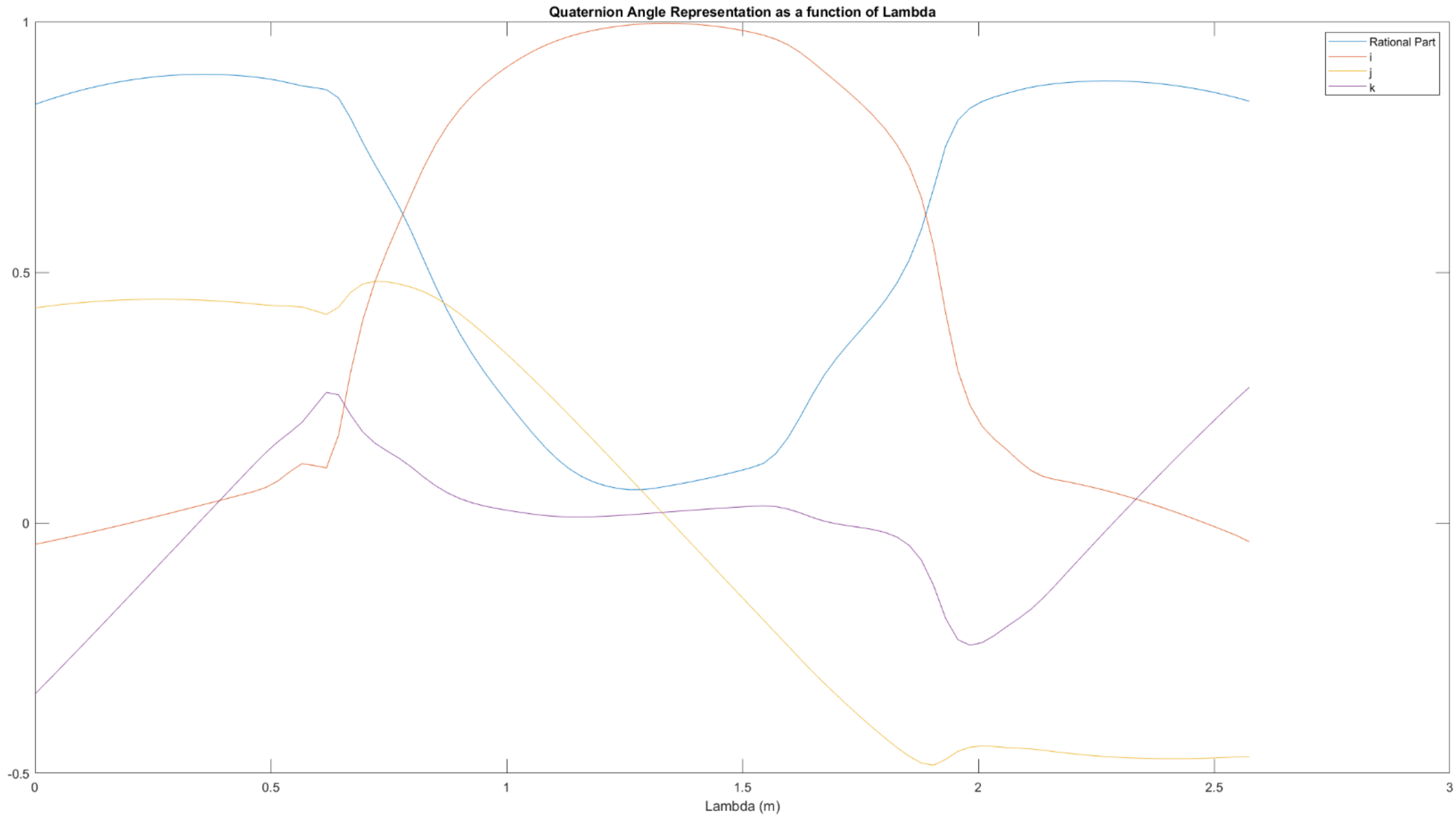
Representations (Results)



Angle Axis Representation

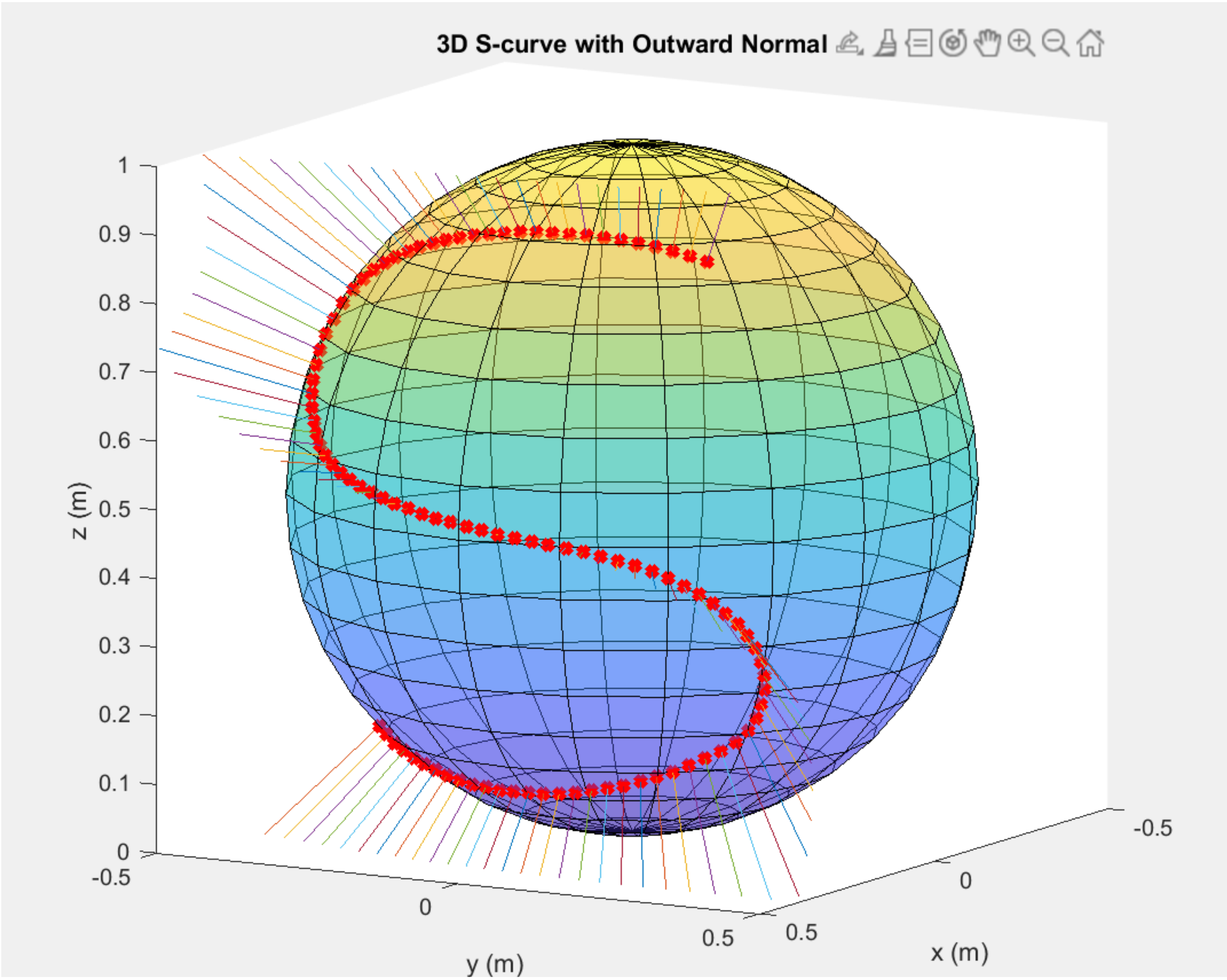


Euler Angle Representation



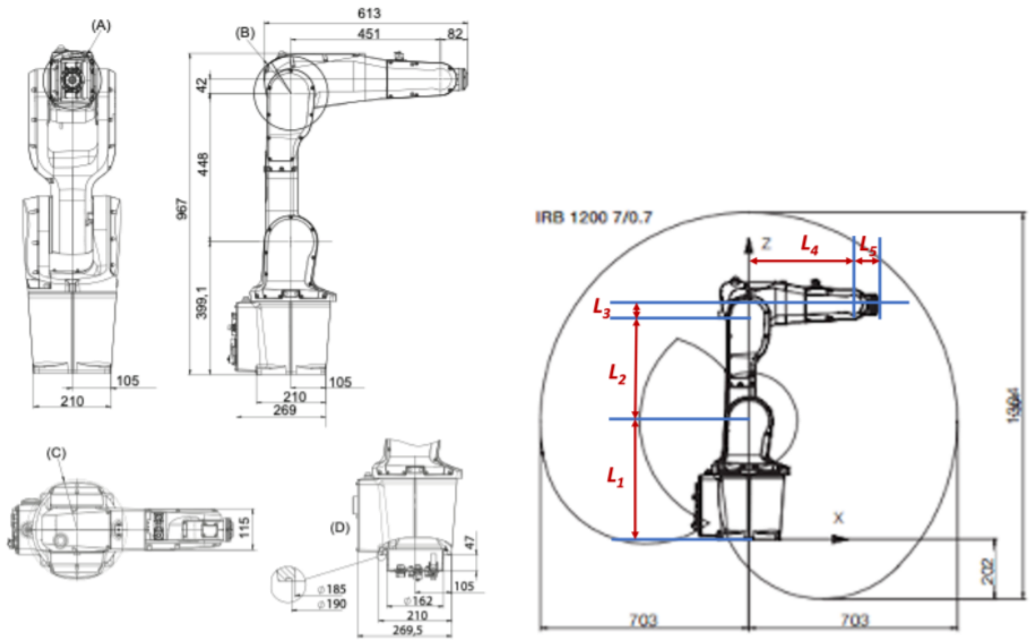
Quaternion Representation

An S-curve on a Sphere with its outward normal is also shown here:



Problem 2: Forward and Inverse Kinematics

The ABB 1200-5/0.9 is shown below in its zero configuration. In order to find the inverse kinematics Product of Exponentials (POE), Standard Denavit-Hartenberg (SDH), and/ or Modified Denavit-Hartenberg (MDH) parameters need to be found. The inverse kinematics can be found using subproblem decomposition from the POE parameters. Forward kinematics can be done iteratively if the joint angles are known.



Product of Exponentials (POE) (Approach)

Figure showing POE Derivation:

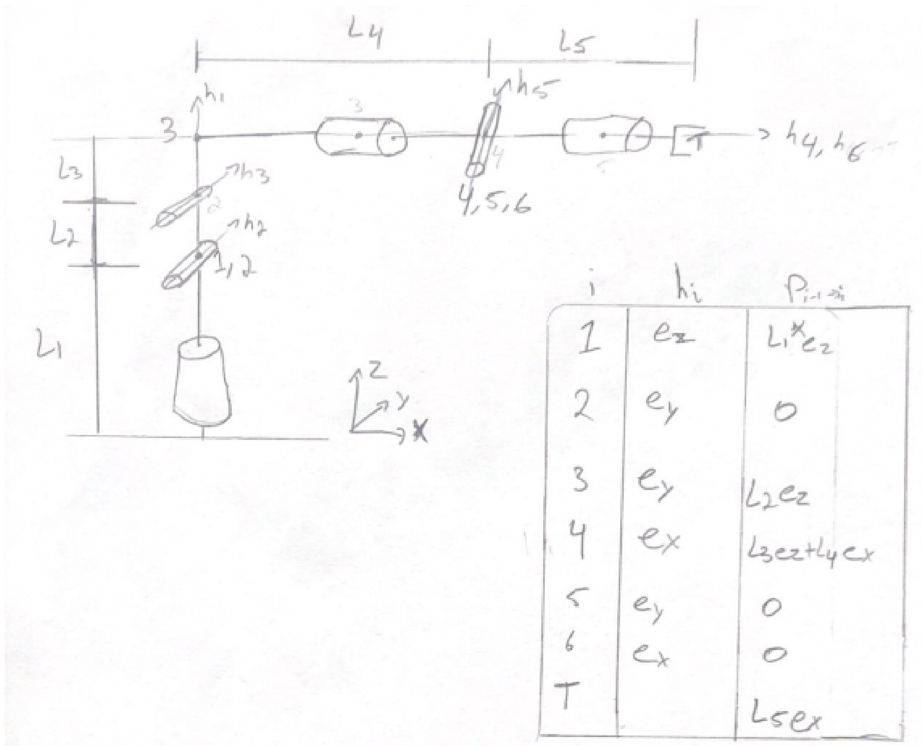
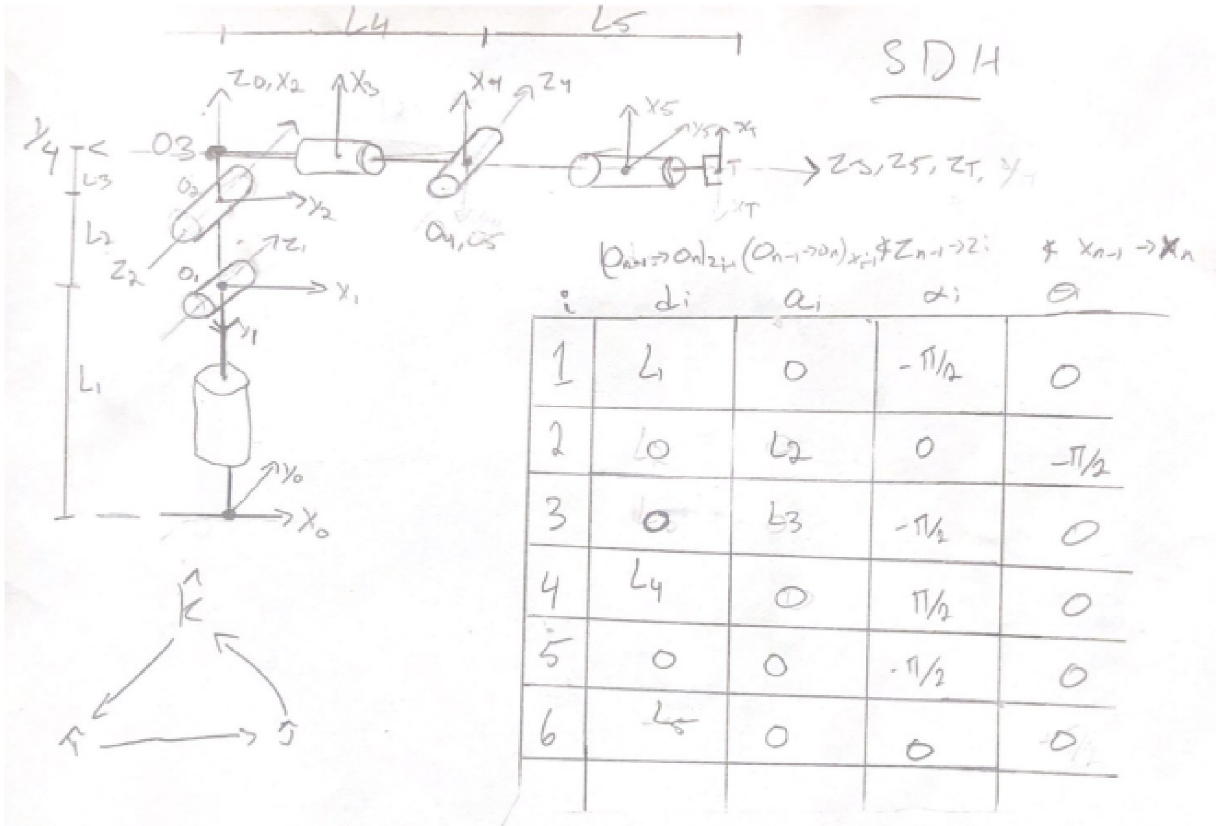


Table of POE Parameters:

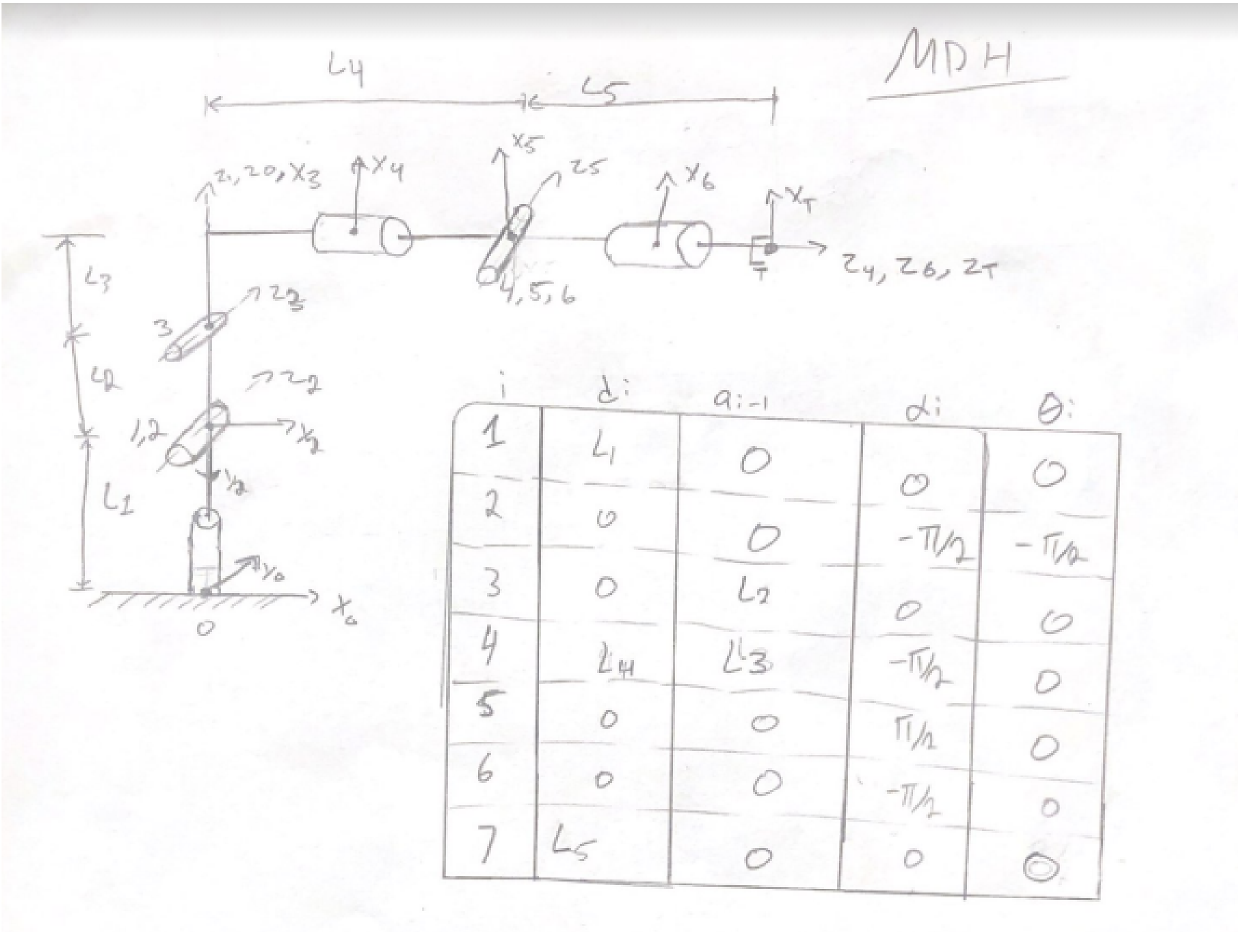
i	**h_i**	**$P_{i-1,i}$**
1	ez	L1*ex
2	ey	0
3	ey	L2*ez
4	ex	L3*ez +L4*ex
5	ey	0
6	ex	0
Tool		L5*ex

Standard Denavit-Hartenberg (SDH) (Approach)



i	d_i	a_i	$alpha_i$	$theta_i$
1	L1	0	-pi/2	0
2	0	L2	0	-pi/2
3	0	L3	-pi/2	0
4	L4	0	pi/2	0
5	0	0	-pi/2	0
6	L5	0	0	0

Modified Denavit-Hartenberg (MDH) (Approach)



i	d_i	a_{i-1}	$alpha_i$	$theta_i$
1	L1	0	0	0
2	0	0	-pi/2	-pi/2
3	0	L2	0	0
4	L4	L3	-pi/2	0
5	0	0	pi/2	0
6	0	0	-pi/2	0
7	L5	0	0	0

Forward Kinematics (Approach)

Using POE parameters, the forward kinematics is quite simple. The rotational matrix of the tool in the base (O) frame is the rotation matrix of each joint with respect to its axis of rotation. And the position of the end effector is the position of each of the POE origins summed. The forms are shown below:

$$R_{OT} = Rot(h1,\ q1) * Rot(h2,\ q2) * Rot(h3,\ q3) * Rot(h4,\ q4) * Rot(h5,\ q5) * Rot(h6,\ q6)$$

$$P_{OT} = P_{01} + Rot(h1,\ q1) * P_{12} + Rot(h1,\ q1) * Rot(h2,\ q2) * P_{23} ... + \prod_{i=1}^{T} Rot(hi,\ qi) * P_{i-1,1}$$

Inverse Kinematics (Subproblem Decomposition) (Approach)

This is handwritten for ease. :

Inverse Kinematics → Sub Problem decomposition:

POE Parameters:

$h_1 = e_z$	$P_{01} = L_1 e_z$
$h_2 = e_y$	$P_{12} = 0$
$h_3 = e_y$	$P_{23} = L_2 e_z$
$h_4 = e_x$	$P_{34} = L_3 e_z + L_4 e_x$
$h_5 = e_y$	$P_{45} = 0$
$h_6 = e_y$	$P_{56} = 0$
	$P_{6T} = L_5 e_z$

$$R_{0T} = R_z(q_1) R_y(q_2) R_y(q_3) R_x(q_4) R_y(q_5) R_x(q_6)$$

$$P_{0T} = P_{01} + \cancel{R_{01} P_{12}}^{\to 0} + R_{01} R_{12} P_{23}$$
$$+ R_{01} R_{12} R_{23} P_{34}$$
$$+ R_{01} R_{12} \cancel{R_{23} R_{34} P_{45}} \to 0$$
$$+ R_{01} R_{12} R_{23} \cancel{R_{34} R_{45} P_{56}} \to 0$$
$$+ \underbrace{R_{01} R_{12} R_{23} R_{34} R_{45} R_{56}}_{R_{0T}} P_{6T}$$

$$P_{0T} = P_{01} + R_{01} R_{12} P_{23} + R_{01} R_{12} R_{23} P_{34} + R_{0T} P_{6T}$$

$$\underset{\text{Known}}{P_{0T}} = \underset{\text{Known}}{P_{01}} + R_{01} R_{12} P_{23} + R_{01} R_{12} R_{23} P_{34} + \underset{\text{Known}}{\underline{R_{0T} P_{6T}}}$$

$$\underbrace{P_{0T} - P_{01} - R_{0T} P_{6T}}_{(P_{16})_0} = \underset{R_z(q_1)R_y(q_2)}{R_{01} R_{12} P_{23}} + \underset{R_z(q_1)R_y(q_2)R_y(q_3)}{\underline{R_{01} R_{12}} R_{23} P_{34}}$$

$$\| P_{0T} - P_{01} - R_{0T} P_{6T} \| = \| \cancel{R_{01} R_{12}} (P_{23} + R_{23} P_{34}) \|$$ ← since rotation does not effect the norm; R_{01} & R_{12} go away

Sub Problem 3:

$$\| P_2 - R(k, q) P_1 \| = d$$

$d = \| P_{0T} - P_{01} - R_{0T} P_{6T} \|$

$P_2 = P_{23}$

$P_1 = -P_{34}$

$k = e_y$

} q_3 will have 2 solutions.

$$R_z(-q_1)(P_{16})_0 = R_y(q_2)(P_{23} + R_y(q_3) P_{34})$$

Sub Problem 2: Parameters:

$k_1 = e_z$; $P_1 = (P_{16})_0$ $k_2 = e_y$; $P_2 = P_{23} + R_y(q_3) P_{34}$

2 solutions of q_1 & q_2 for each solution of q_3

so 4 solutions of $[q_1, q_2, q_3]$

$$Rot = R_z(q_1)\, R_y(q_2+q_3)\, R_z(q_4) R_y(q_5)\, R_x(q_6)$$

$$\left(Rot\, R_y(-q_2-q_3)\, R_z(-q_1) = R_x(q_4) R_y(q_5)\, R_x(q_6)\right) e_x$$

$$Rot\, R_y(-q_2-q_3)\, R_z(-q_1)\, e_x = R_x(q_4)\, R_y(q_5)\, e_x \cdot \cancel{R_x(q_6) e_x}$$

$$R_x(-q_4)\, R_y(-q_2-q_3)\, R_z(-q_1)\, Rot\, e_x = R_y(q_5)\, e_x$$

1 a rotation about the X axis multiplied by the x-axis cancels out

Sub Problem 2:

$$R(k_1, q_1)\, p_1 = R(k_2, q_2)\, p_2$$

$k_1 = -e_x$

$q_1 = q_4$

$p_1 = R_y(-(q_2+q_3))\, R_z(-q_1)\, Rot\, e_x$

$k_2 = e_y$

$q_2 = q_5$

$p_2 = e_x$

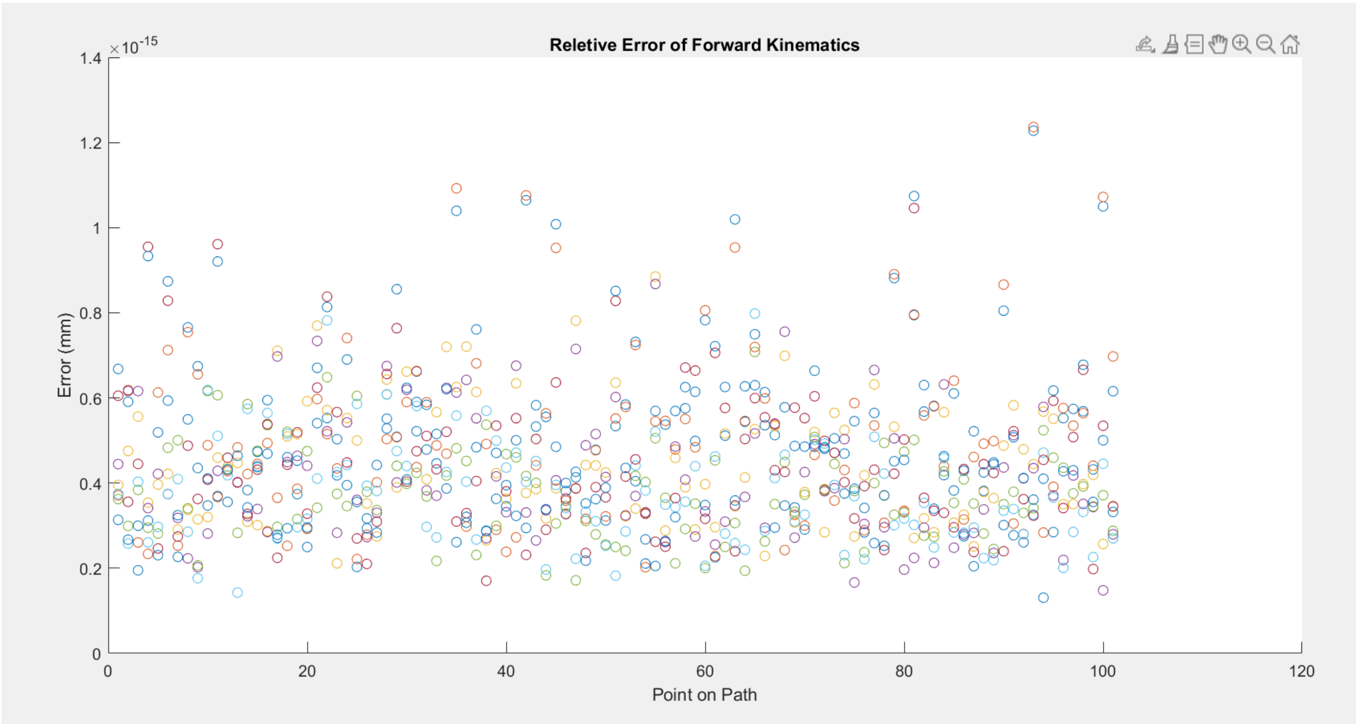
2 solutions for q_4 & q_5 so 8 sols for $[q_1, q_2, q_3, q_4, q_5]$

$$\underbrace{R_y(-q_5) R_x(-q_4)\, R_y(-(q_2+q_3)) R_z(-q_1)\, Rot\, e_y = R_x(q_6)\, e_y}_{\text{Sub problem 1}}$$

1 solution for q_6 ∴ 8 total solutions for $[q_1, q_2, q_3, q_4, q_5, q_6]$

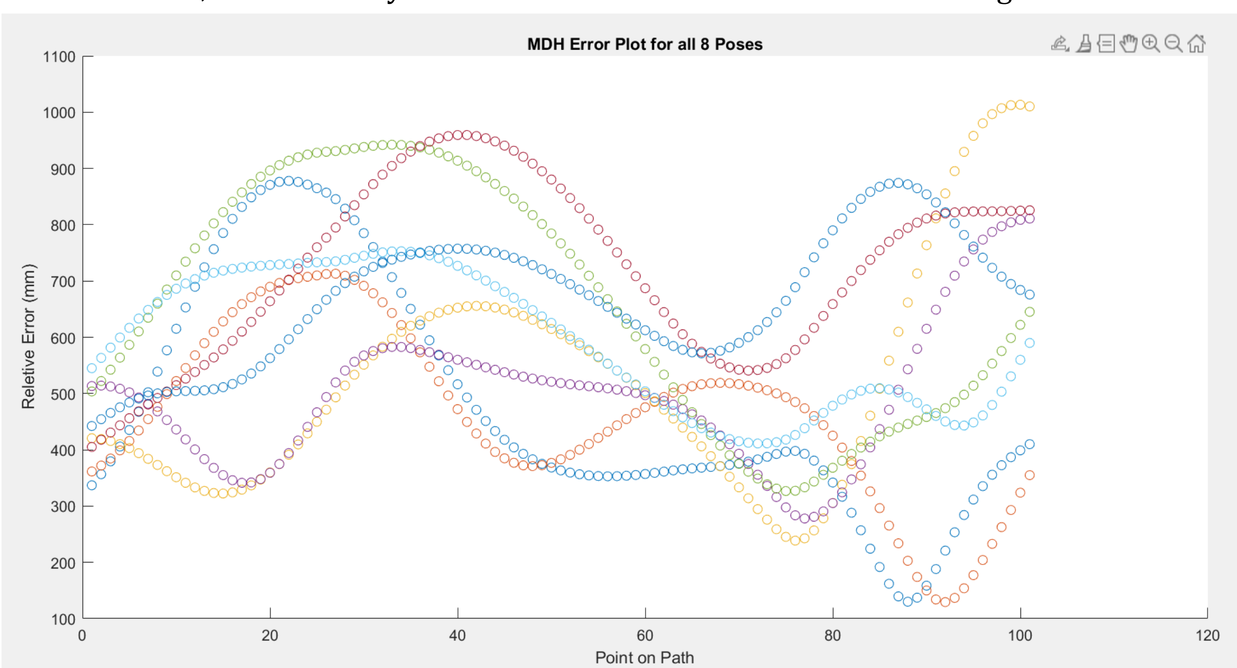
Inverse Kinematics (Results)

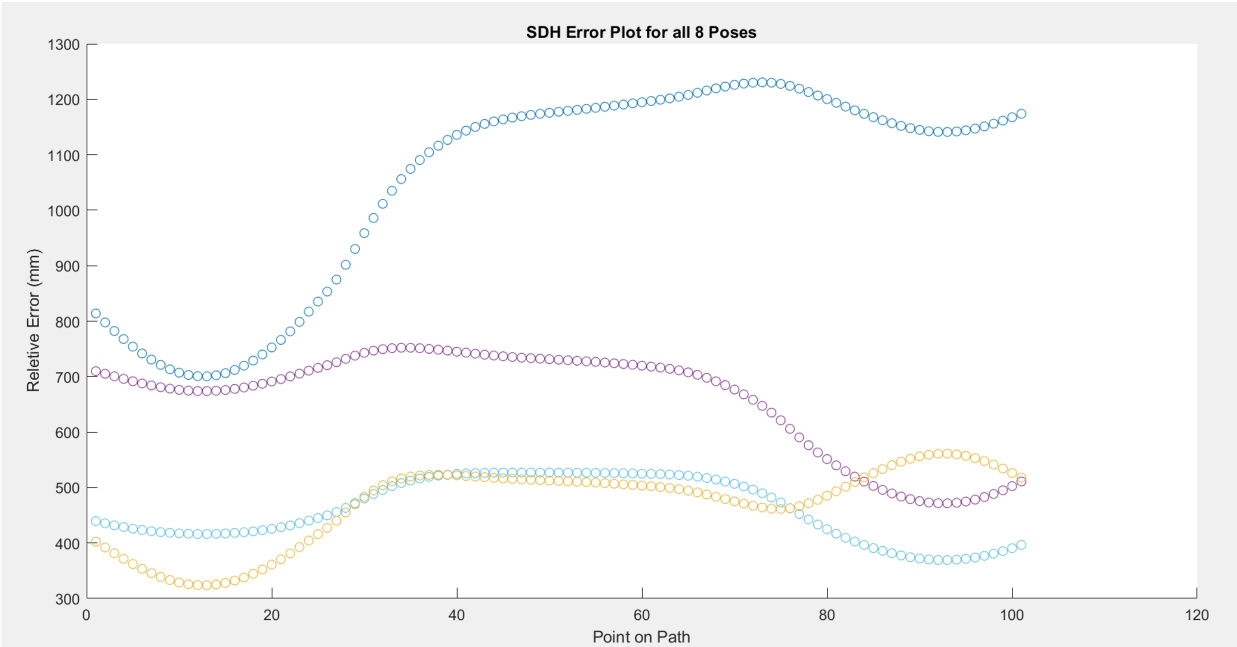
Forward Kinematics Error (Results)



As can be seen in the figure above, error is exceedingly low for forward kinematics using the POE method, on the order of 10^-16th.

The MDH and SDH errors were quite large, this is due to human error, likely in parameters or in execution; unfortunately the root cause of this was not able to be diagnosed in time.

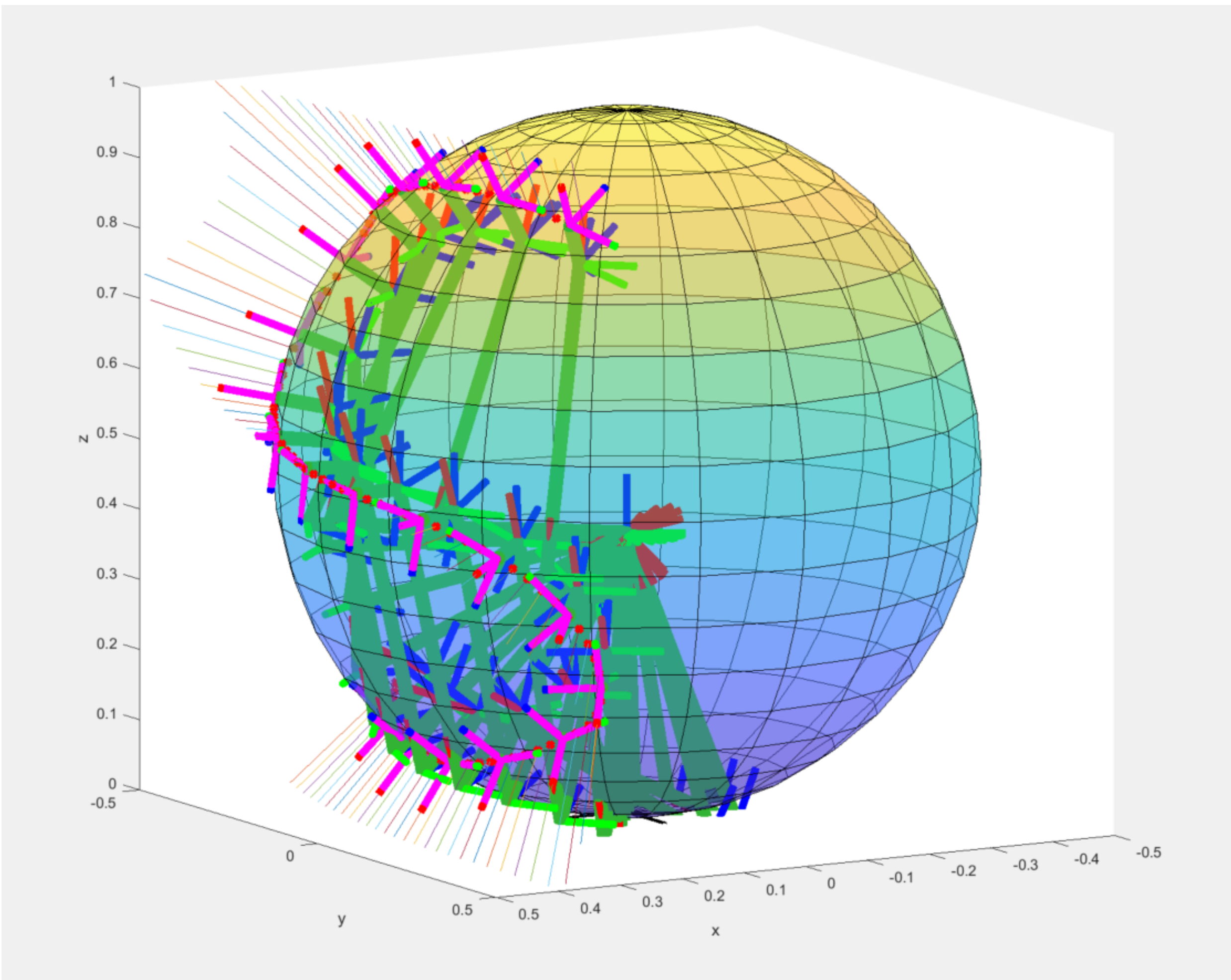



SDH Error Plot for all 8 Poses
Reletive Error (mm)
Point on Path
1300
1200
1100
1000
900
800
700
600
500
400
300
0
20
40
60
80
100
120

Problem 3: Path Speed

From inverse kinematics there are 8 possible poses. They will each have different variations in joint angles, which will result in different maximum path speeds for each pose, assuming path speed is constant and given a constraint on joint velocity.

One of Eight Poses (Results)



Video of the arm is shown at the Youtube link below:
https://youtu.be/4XmoP8-bzgQ (recommended to put on .5- .25 speed)

Path Speed (Approach)

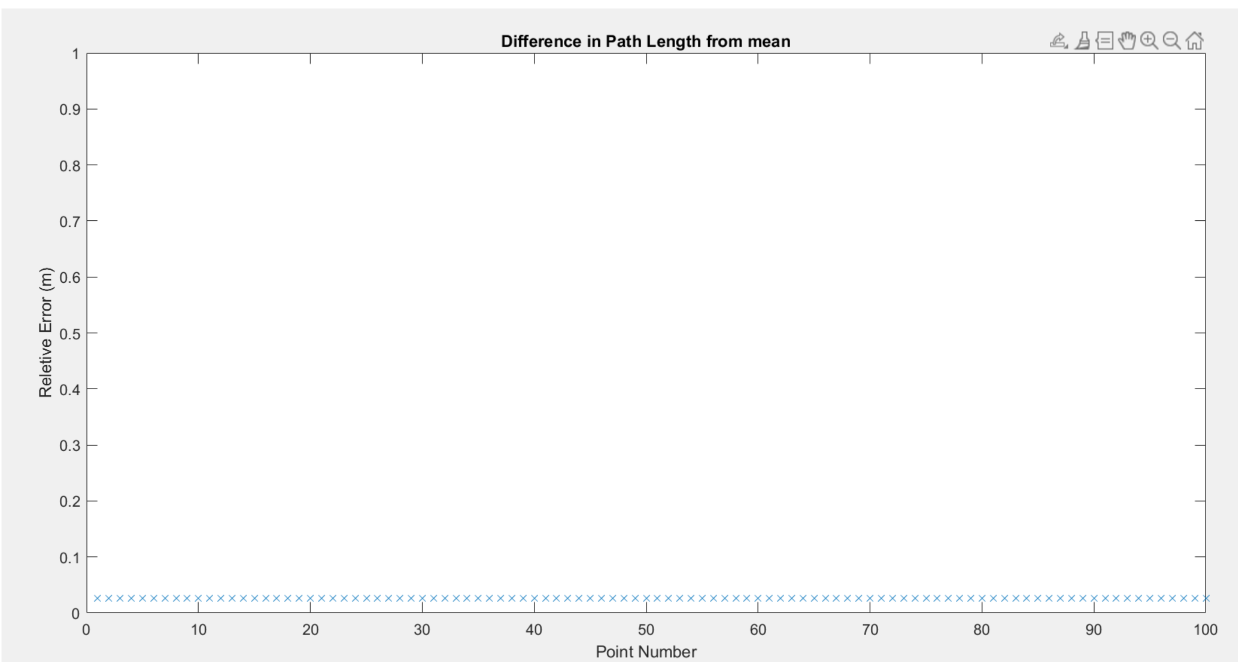
To find the maximum possible constant path speed for each of the 8 poses, we first need to find the maximum angle any joint has to traverse. This is done by taking the forward difference for each joint at each point in the path and saving that in an array. This array is 6 joints X 100 points long and called “dq”. From there the maximum of these 600 points can be found, this is the maximum angle any joint has to travel while tracing this curve.

Since we have a joint velocity constraint of ±2 rad/sec and we know the maximum angle any joint to travel we can divide the angle by the velocity constraint to find the time it takes to get from one point on the path to the next, “dt”.
The rate of change of the joint angles, qdot, is just dq/dt. Linear path speed is just the norm of the difference of the points; in the x, y and z directions, over the change in time, “dt”. This assumption only holds if path length between points is held constant.

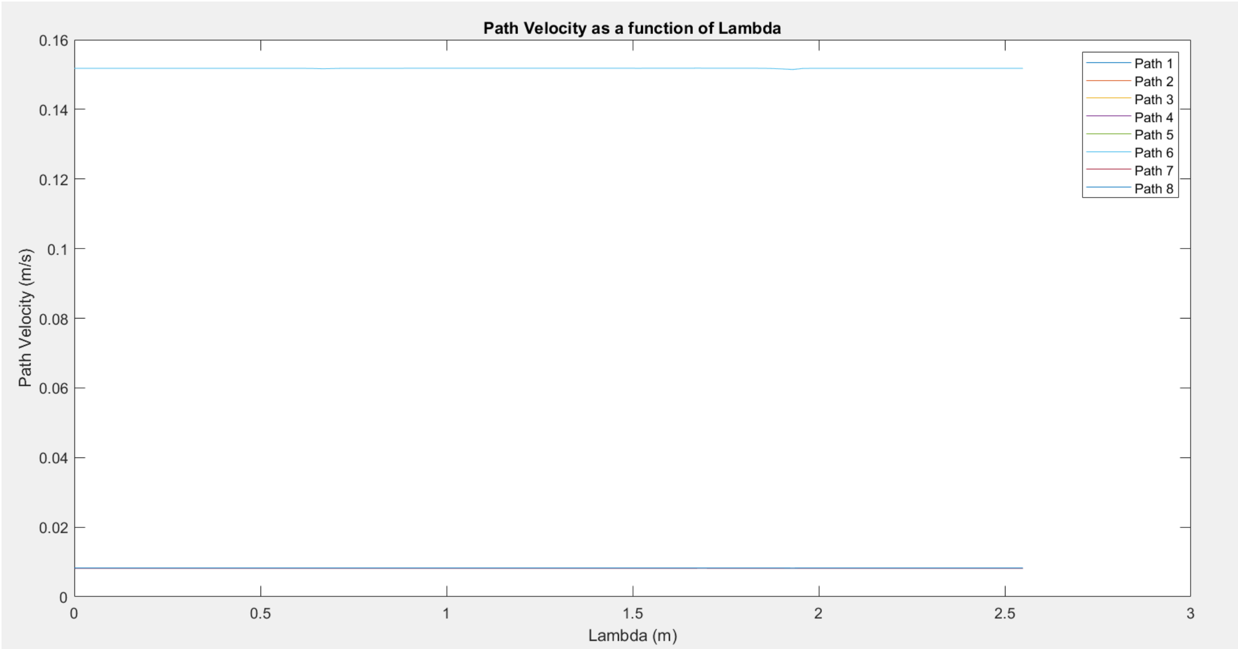
Joint Variation (Approach)

To find out which q3 varied the least, the standard deviation was found for all of the q3s for all of the 8 poses.

Path Speed (Results)



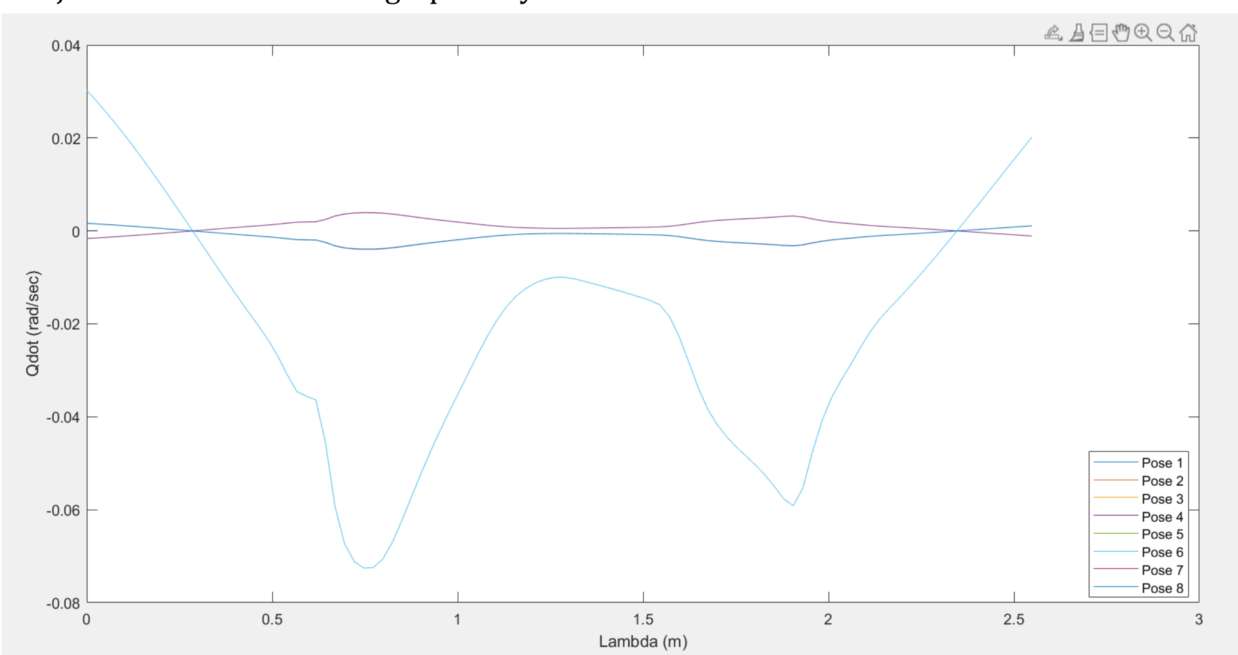
To prove that path length is sufficiently equivalent between points, the difference in path length for each point from the mean is shown above. This is deemed sufficiently small for the analysis to continue. Next the linear path speed is found as a function of lambda, this analysis is shown below.



Each is sufficiently constant and the highest path velocity is Pose 6,with a path velocity of 0.1518 m/s.

Joint Variation (Results)

The joint variation is shown graphically below:



It is clear that all the poses have the same standard deviation, found to be 0.0014 (rad/sec), except for pose 6, which has a much higher standard deviation, found to be .0255(rad/sec).

Conclusion:

The ABB 1200/ 0.9 is a 6 joint arm that has 6 degrees of freedom. We have seen that for any desired position (Xt, Yt, Zt) it can enforce a desired orientation constraint as well, allowing a lot of control over position as may be needed in applications such as manufacturing or robotic surgery. Within this framework, kinematics of the arm can be derived in many ways. In this report we explored the POE, SHD and MDH methods for forward kinematics. POE method is the easiest to execute successfully as shown, the SDH and MDH methods may be powerful however leave a lot to human error.

The inverse kinematics are useful when we have a target pose and want a set of joint angles. The approach taken was based on Product of Exponentials parameters. Subproblem decomposition of the arm results in the quicker computation time than an iterative method (not explored in depth in this paper but good future work).

The velocity of the robot and the path can also be constrained using the path velocity and/ or joint velocity constraints. It is good to have low variations in joints and minimize outliers in change of total joint angle if the desired outcome is to have a high constant path velocity.