Robotics Lab

Assignment 2

Maxim Fenko 410534

Abdelkarim Ben Salah 376393

Bryan Oppong-Boateng 390549

November 24, 2024

Contents

1 A Forward Kinematics

1.1 A1 DH Parameters for 3-DOF RRR Puma

In order to determine the DH Parameter for the 3-DOF Puma we have to follow the steps presented in the tutorial.

1. Identify joint axes; consider i-1 and i
2. Identify common perpendicular
3. Label frame origin at perpendicular (or intersection)
4. Assign $\hat{Z}_i$ along joint axis
5. Assign $\hat{X}_i$ along perpendicular; if joint axes intersect, orthogonal to the axes plane
6. Complete frame by adding Y axis (right-hand-rule)
7. Assign 0 to match 1
8. Choose end-effector frame n

Most of the steps were already performed in the Puma Robot sketch below. The x-axis and y-axis were already assigned for all joints and the $\hat{Z}_i$ can be determined through the right hand rule to be pointing into the schematic. Therefore positive change in angle is clockwise.

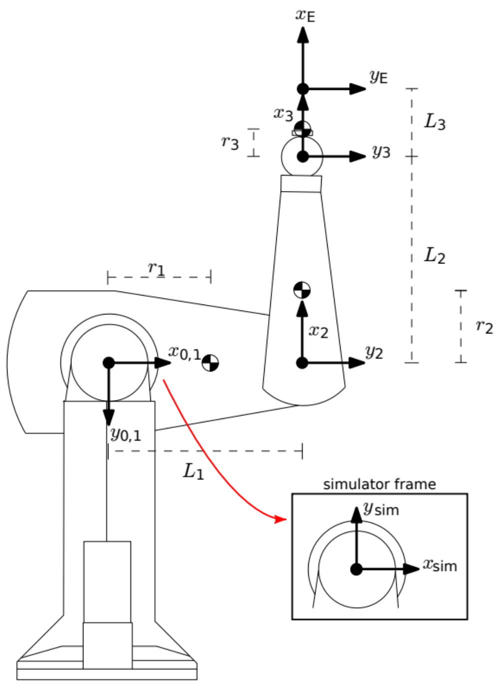


Figure 1: 3-DOF RRR Puma depiction from the task sheet.

To fill the DH parameter in the table, the following in the tutorial presented steps must be followed.

- α_i = the angle between $\hat{Z}_i$ and $\hat{Z}_{i+1}$ measured about $\hat{X}_i$
- a_i = the distance from $\hat{Z}_i$ and $\hat{Z}_{i+1}$ measured along $\hat{X}_i$
- d_i = the distance from $\hat{X}_{i-1}$ and $\hat{X}_i$ measured along $\hat{Z}_i$
- θ_i = the angle between $\hat{X}_{i-1}$ and $\hat{X}_i$ measured about $\hat{Z}_i$

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	q_1
2	0	l_1	0	$q_2 - \frac{\pi}{2}$
3	0	l_2	0	q_3
4(E)	0	l_3	0	0

Table 1: DH Parameters.

As one can clearly see in Figure 1, when choosing x0=x1 and y0=y1 and following the right-hand rule, all $\hat{Z}$ axes are aligned. Therefore the column of the α_i's that represent the angle of the $\hat{Z}_i$'s to each other is zero. The distance between the $\hat{Z}_i$'s is obviously the different lengths of the parts connecting the joints. As $\hat{Z}_0 = \hat{Z}_1$; a_0=0 and the others l_1, l_2 and l_3 respectively. As all joints have the same oriented $\hat{Z}$ it is also clear that the distance along $\hat{Z}_i$ of the $\hat{X}_i$ is always zero. The angle between $\hat{X}_i$ and $\hat{X}_{i-1}$ represents the angle between the joints. That's why joint 3 compared to the end effector is 0, $\theta_1 = q_1$ and $\theta_3 = q_3$. As in the neutral configuration the second joint is rotated by $-\frac{\pi}{2}$ about $\hat{Z}_2$ (negative because the $\hat{Z}_i$'s are pointing into the image according to the right-hand rule), $\theta_2 = q_2 - \frac{\pi}{2}$.

1.2 A2 Transformation between frames

$$
{}^{i-1}_{\ \ \ i}T = \begin{pmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i \cdot c\alpha_{i-1} & c\theta_i \cdot c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} \cdot d_i \\ s\theta_i \cdot s\alpha_{i-1} & c\theta_i \cdot s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} \cdot d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.1}
$$

Above is depicted the general formula for the individual transformation matrices. When we now insert the θ_i's according to the definition in Table 1, we get a -$\frac{\pi}{2}$ shift in all terms that include θ_2. So the sin terms including θ_2 become $-\cos$ and cos terms including θ_2 become sin. So inserting the values from Table 1 in Equation 1.2.1, brings the following four transformation matrices.

$$ {}^{0}_{1}T = \begin{pmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.2} $$

$$ {}^{1}_{2}T = \begin{pmatrix} s_2 & c_2 & 0 & l_1 \\ -c_2 & s_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.3} $$

$$ {}^{2}_{3}T = \begin{pmatrix} c_3 & -s_3 & 0 & l_2 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.4} $$

$$ {}^{3}_{E}T = \begin{pmatrix} 1 & 0 & 0 & l_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.5} $$

$$ {}^{0}_{E}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T \cdot {}^{3}_{E}T \tag{1.2.6} $$

$$ {}^{0}_{2}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \tag{1.2.7} $$

$$ {}^{0}_{2}T = \begin{pmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} s_2 & c_2 & 0 & l_1 \\ -c_2 & s_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.8} $$

$$ {}^{0}_{2}T = \begin{pmatrix} c_1 \cdot s_2 + c_2 \cdot s_1 & c_1 \cdot c_2 - s_1 \cdot s_2 & 0 & c_1 \cdot l_1 \\ s_1 \cdot s_2 - c_1 \cdot c_2 & s_1 \cdot c_2 + s_2 \cdot c_1 & 0 & s_1 \cdot l_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.9} $$

According to the following sin and cos sum formulas:

$$ sin(\alpha \pm \beta) = sin(\alpha)cos(\beta) \pm cos(\alpha)sin(\beta) \tag{1.2.10} $$

$$ cos(\alpha \pm \beta) = cos(\alpha)cos(\beta) \mp sin(\alpha)sin(\beta) \tag{1.2.11} $$

the equation can be simplified, where c_{12} refers to $cos(q_1+q_2)$ and s_{12} refers to $sin(q_1+q_2)$.

$$ {}^0_2T = \begin{pmatrix} s_{12} & c_{12} & 0 & c_1 \cdot l_1 \\ -c_{12} & s_{12} & 0 & s_1 \cdot l_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.12} $$

Analogously ${}^0_3T$ can be calculated.

$$ {}^0_3T = {}^0_2T \cdot {}^2_3T \tag{1.2.13} $$

$$ {}^0_3T = \begin{pmatrix} s_{12} & c_{12} & 0 & c_1 \cdot l_1 \\ -c_{12} & s_{12} & 0 & s_1 \cdot l_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} c_3 & -s_3 & 0 & l_2 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.14} $$

$$ {}^0_3T = \begin{pmatrix} s_{123} & c_{123} & 0 & c_1 \cdot l_1 + s_{12} \cdot l_2 \\ -c_{123} & s_{123} & 0 & s_1 \cdot l_1 - c_{12} \cdot l_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.15} $$

$$ {}^0_ET = {}^0_3T \cdot {}^3_ET \tag{1.2.16} $$

$$ {}^0_ET = \begin{pmatrix} s_{123} & c_{123} & 0 & c_1 \cdot l_1 + s_{12} \cdot l_2 \\ -c_{123} & s_{123} & 0 & s_1 \cdot l_1 - c_{12} \cdot l_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & l_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.17} $$

$$ {}^0_ET = \begin{pmatrix} s_{123} & c_{123} & 0 & c_1 \cdot l_1 + s_{12} \cdot l_2 + s_{123} \cdot l_3 \\ -c_{123} & s_{123} & 0 & s_1 \cdot l_1 - c_{12} \cdot l_2 - c_{123} \cdot l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.18} $$

This matrix can be easily tested with the simulator software. For the configuration q_{test1}=(q_1, q_2, q_3)=(0, $\frac{\pi}{2}$, 0) we expect the Puma to be fully extended in x direction. So the resulting x translation should be the length of the puma and the y translation 0.

$$ {}^0_ET(q_{test1}) = \begin{pmatrix} 1 & 0 & 0 & 1 \cdot l_1 + 1 \cdot l_2 + 1 \cdot l_3 \\ 0 & 1 & 0 & 0 \cdot l_1 - 0 \cdot l_2 - 0 \cdot l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.19} $$

The translation can be looked up from the last column of the matrix. Therefore it would be $l_1 + l_2 + l_3$ on the x-axis and 0 on the y-axis. Which fulfills the expectation.
Another test configuration q_{test2}=(q_1, q_2, q_3)=($-\frac{\pi}{4}$, $\frac{\pi}{2}$,0). In this configuration the puma is extended at an angle of 45°. The x coordinate should be sin/cos(45°)=$\frac{\sqrt{2}}{2} \cdot (l_1 + l_2 + l_3)$ and y=-$\frac{\sqrt{2}}{2} \cdot (l_1 + l_2 + l_3)$, as the simulator frame has a mirrored y axis.

$$
{}^0_ET(q_{test2}) = \begin{pmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \cdot l_1 + \frac{\sqrt{2}}{2} \cdot l_2 + \frac{\sqrt{2}}{2} \cdot l_3 \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \cdot l_1 - \frac{\sqrt{2}}{2} \cdot l_2 - \frac{\sqrt{2}}{2} \cdot l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.20}
$$

In this case the transformation matrix delivers the expected results as well.
If we wanted to align our base frame (x0,1/y0,1) with the simulator frame, we would choose x0,1 and y0,1 in its orientation and would only have a rotation along the $\hat{X}_1$. So that the adjusted DH table and the transformation matrix ${}^0_ET$ would be as follows to mirror the base frame y-axis.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	q_1
2	$-\pi$	l_1	0	$q_2 - \frac{\pi}{2}$
3	0	l_2	0	q_3
4(E)	0	l_3	0	0

Table 2: DH parameter if orientated according to the simulator frame.

$$
{}^0_ET = \begin{pmatrix} s_{-123} & c_{-123} & 0 & c_1 \cdot l_1 + s_{12} \cdot l_2 + s_{123} \cdot l_3 \\ c_{-123} & -s_{-123} & 0 & -s_1 \cdot l_1 + c_{12} \cdot l_2 + c_{123} \cdot l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.2.21}
$$

As not otherwise stated in the task we gonna proceed using the original ${}^0_ET$ (see Equation 1.2.18) and DH (see Table 1) parameters in the following.

1.3 A3 - End effector position in operational space

As seen in the previous subtask, the translational x and y components of the F vector can be taken from the first two rows of the last column of ${}^0_ET$. For the angle alpha we only need to sum up the angle θ_i as those describe the angular translation between the starting joint and the endeffector. The resulting F is as follows.

$$
F(q) = \begin{pmatrix} c_1 \cdot l_1 + s_{12} \cdot l_2 + s_{123} \cdot l_3 \\ s_1 \cdot l_1 - c_{12} \cdot l_2 - c_{123} \cdot l_3 \\ q_1 + q_2 - \frac{\pi}{2} + q_3 \end{pmatrix} \tag{1.3.1}
$$

We can test this once again with the Stanford Puma simulator, by calculating F according to Equation 1.3.1 and comparing it to the simulator results.

$$F(0, \frac{\pi}{2}, 0) = \begin{pmatrix} l_1 + l_2 + l_3 \\ 0 \\ 0 \end{pmatrix} \tag{1.3.2}$$

$$F(-\frac{\pi}{4}, \frac{\pi}{2}, 0) = \begin{pmatrix} \frac{\sqrt{2}}{2} \cdot (l_1 + l_2 + l_3) \\ -\frac{\sqrt{2}}{2} \cdot (l_1 + l_2 + l_3) \\ -\frac{\pi}{4} \end{pmatrix} \tag{1.3.3}$$

$$F(-\frac{3\pi}{4}, \frac{\pi}{2}, 0) = \begin{pmatrix} -\frac{\sqrt{2}}{2} \cdot (l_1 + l_2 + l_3) \\ -\frac{\sqrt{2}}{2} \cdot (l_1 + l_2 + l_3) \\ -\frac{3\pi}{4} \end{pmatrix} \tag{1.3.4}$$

Those results are all equivalent in the puma simulator (with respect to the mirrored y-axis frame) which concludes the testing of the calculated F matrix.

1.4 A4 - Compute the endeffector jacobian

The Jacobian of F can be computed by deriving each of its rows by all of the states (q_1, q_2, q_3). So the Jacobian is defined in our case as follows. (F_i refers to the i-th row of $F(q)$).

$$J(q) = \begin{pmatrix} \frac{\delta F_1}{\delta q_1} & \frac{\delta F_1}{\delta q_2} & \frac{\delta F_1}{\delta q_3} \\ \frac{\delta F_2}{\delta q_1} & \frac{\delta F_2}{\delta q_2} & \frac{\delta F_2}{\delta q_3} \\ \frac{\delta F_3}{\delta q_1} & \frac{\delta F_3}{\delta q_2} & \frac{\delta F_3}{\delta q_3} \end{pmatrix} \tag{1.4.1}$$

As the angles have no factor in front of the the derivatives can be calculated trivially

$$J(q) = \begin{pmatrix} -s_1 \cdot l_1 + c_{12} \cdot l_2 + c_{123} \cdot l_3 & c_{12} \cdot l_2 + c_{123} \cdot l_3 & l_3 \cdot c_{123} \\ c_1 \cdot l_1 + s_{12} \cdot l_2 + s_{123} \cdot l_3 & s_{12} \cdot l_2 + s_{123} \cdot l_3 & s_{123} \cdot l_3 \\ 1 & 1 & 1 \end{pmatrix} \tag{1.4.2}$$

1.5 A5 - Understanding the jacobian matrix and pose singularities

When calculating the following tasks it is important to keep in mind that the base frame y-axis (y0,1) is pointing downwards. So negative y change is an upward movement and positive y a downward movement (as it is counterintuitive).

1.5.1 i

$q_1 = (0, 0, -\frac{\pi}{2})$
Inserting q_1 in Equation 1.3.1 to get $F(q_1)$.

$$x(q_1) = F(q_1) = \begin{pmatrix} l_1 + 0 - l_3 \\ 0 - l_2 + 0 \\ -\frac{\pi}{2} - \frac{\pi}{2} \end{pmatrix} = \begin{pmatrix} l_1 - l_3 \\ -l_2 \\ -\pi \end{pmatrix} \tag{1.5.1}$$

Inserting q_1 in Equation 1.4.2 to get $J(q_1)$.

$$J(q_1) = \begin{pmatrix} l_2 & l_2 & 0 \\ l_1 - l_3 & -l_3 & -l_3 \\ 1 & 1 & 1 \end{pmatrix} \tag{1.5.2}$$

To get the velocities in the x coordinate frame we need to apply the following formula.

$$\dot{x}(q_1) = J(q_1) \cdot \dot{q} \tag{1.5.3}$$

Für $\dot{q} = (1\ 0\ 0)^T$:

$$\dot{x}_1(q_1) = J(q_1) \cdot (1\ 0\ 0)^T = \begin{pmatrix} l_2 \\ l_1 - l_3 \\ 1 \end{pmatrix} \tag{1.5.4}$$

Für $\dot{q} = (0\ 1\ 0)^T$:

$$\dot{x}_2(q_1) = J(q_1) \cdot (0\ 1\ 0)^T = \begin{pmatrix} l_2 \\ -l_3 \\ 1 \end{pmatrix} \tag{1.5.5}$$

Für $\dot{q} = (0\ 0\ 1)^T$:

$$\dot{x}_3(q_1) = J(q_1) \cdot (0\ 1\ 0)^T = \begin{pmatrix} 0 \\ -l_3 \\ 1 \end{pmatrix} \tag{1.5.6}$$

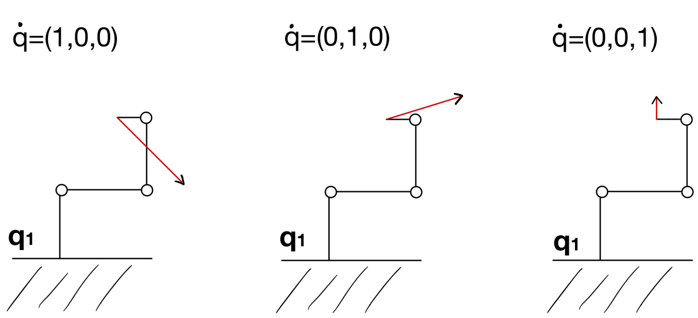


Figure 2: Effect of each joint on translational velocities for initial configuration q_1.

As we can see from the calculation of the impact of each joint on the x frame velocities $\dot{x}_1$, $\dot{x}_2$, $\dot{x}_3$ and the sketch representing it, the puma in this configuration is controllable in each direction. This can be seen directly in the Jacobian J(q_1), as none of its rows is close to or equal to zero.

1.5.2 ii

$q_2 = (-\frac{\pi}{2}, \frac{\pi}{2}\ , \mathbf{0.1})$
Inserting q_2 in Equation 1.3.1 to get $F(q_2)$.

$$F(q_2) = \begin{pmatrix} sin(0.1) \cdot l_3 \\ -l_1 - l_2 - cos(0.1) \cdot l_3 \\ 0.1 - \frac{\pi}{2} \end{pmatrix} \tag{1.5.7}$$

Inserting q_2 in Equation 1.4.2 to get $J(q_2)$.

$$J(q_2) = \begin{pmatrix} l_1 + l_2 + cos(0.1) \cdot l_3 & l_2 + cos(0.1) \cdot l_3 & cos(0.1) \cdot l_3 \\ sin(0.1) \cdot l_3 & sin(0.1) \cdot l_3 & sin(0.1) \cdot l_3 \\ 1 & 1 & 1 \end{pmatrix} \tag{1.5.8}$$

$$J(q_2) \approx \begin{pmatrix} l_1 + l_2 + 0.99 \cdot l_3 & l_2 + 0.99 \cdot l_3 & 0.99 \cdot l_3 \\ 0.1 \cdot l_3 & 0.1 \cdot l_3 & 0.1 \cdot l_3 \\ 1 & 1 & 1 \end{pmatrix} \tag{1.5.9}$$

Für $\dot{q} = (1\ 0\ 0)^T$:

$$\dot{x}(q_2) = J(q_2) \cdot (1\ 0\ 0)^T = \begin{pmatrix} l_1 + l_2 + 0.99 \cdot l_3 \\ 0.1 \cdot l_3 \\ 1 \end{pmatrix} \tag{1.5.10}$$

Für $\dot{q} = (0\ 1\ 0)^T$:

$$\dot{x}(q_2) = J(q_2) \cdot (0\ 1\ 0)^T = \begin{pmatrix} l_2 + 0.99 \cdot l_3 \\ 0.1 \cdot l_3 \\ 1 \end{pmatrix} \tag{1.5.11}$$

Für $\dot{q} = (0\ 0\ 1)^T$:

$$\dot{x}(q_2) = J(q_2) \cdot (0\ 1\ 0)^T = \begin{pmatrix} 0.99 \cdot l_3 \\ 0.1 \cdot l_3 \\ 1 \end{pmatrix} \tag{1.5.12}$$

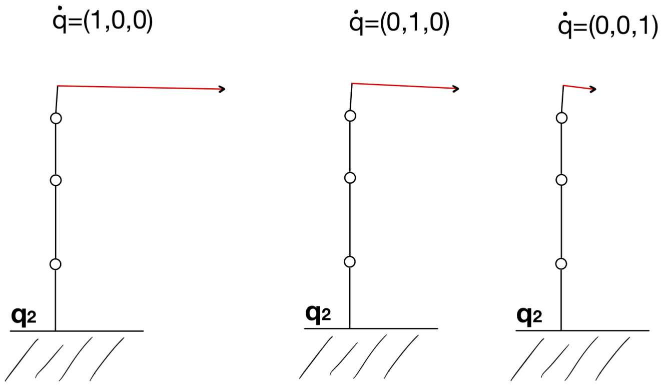


Figure 3: Effect of each joint on translational velocities for initial configuration q_2.

For configuration q_2 we can see that we are close to a singularity. The second row of the Jacobian J(q_2) is getting close to zero and the y-components of the x frame velocities is therefore also getting smaller. However we are not at a singularity as we can still impact the y velocity component a little bit by the change in joint angles.

1.5.3 iii

$q_2 = (-\frac{\pi}{2}, \frac{\pi}{2}$ **, 0**)
Inserting q_3 in Equation 1.3.1 to get $F(q_3)$.

$$F(q_3) = \begin{pmatrix} 0+0-0 \\ -l_1 - l_2 - l_3 \\ -\frac{\pi}{2} + \frac{\pi}{2} - \frac{\pi}{2} + 0 \end{pmatrix} = \begin{pmatrix} 0 \\ -l_1 - l_2 - l_3 \\ -\frac{\pi}{2} \end{pmatrix} \tag{1.5.13}$$

Inserting q_3 in Equation 1.4.2 to get $J(q_3)$.

$$J(q_3) = \begin{pmatrix} l_1 + l_2 + l_3 & l_2 + l_3 & l_3 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \tag{1.5.14}$$

Für $\dot{q} = (1\ 0\ 0)^T$:

$$\dot{x}(q_3) = J(q_3) \cdot (1\ 0\ 0)^T = \begin{pmatrix} l_1 + l_2 \\ 0 \\ 1 \end{pmatrix} \tag{1.5.15}$$

Für $\dot{q} = (0\ 1\ 0)^T$:

$$\dot{x}(q_3) = J(q_3) \cdot (0\ 1\ 0)^T = \begin{pmatrix} l_2 + l_3 \\ 0 \\ 1 \end{pmatrix} \tag{1.5.16}$$

Für $\dot{q} = (0\ 0\ 1)^T$:

$$\dot{x}(q_3) = J(q_3) \cdot (0\ 1\ 0)^T = \begin{pmatrix} l_3 \\ 0 \\ 1 \end{pmatrix} \tag{1.5.17}$$

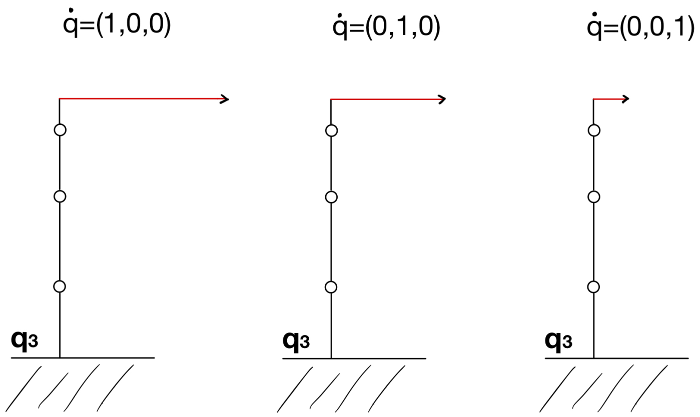


Figure 4: Effect of each joint on translational velocities for initial configuration q_2.

Configuration q_3 is clearly a singular case as the second row of the Jacobian J(q_3) is zero and none of the changes in the joint angles have an impact on the y-axis velocity.

2 Task B

2.1 B1 - Generation of smooth trajectories with polynomial splines

In this task, we should generate a trajectory that starts at a point in time t_{start} in configuration q_a passes after 2.5s the via configuration q_b(referred to as t_{via} in the following) on its way to the final configuration q_c which it reaches at 5s (referred to as t_{end} in the following). We therefore need to compute for each joint two trajectory splines. One that connects the start to the via (in 2.5s) and one which connects the via to the desired end point. The basic formula was shown in the tutorial of the Robotics course and while the structure of Equation 2.1.1 and 2.1.2 seems to be similar the a_{ij} and the time offset is different.

$$q(t) = a_{1,1} + a_{1,2} \cdot (t - t_{start}) + a_{1,3} \cdot (t - t_{start})^2 + a_{1,4} \cdot (t - t_{start})^3 \tag{2.1.1}$$

for $t_{start} \leq t \leq t_{via}$

$$q(t) = a_{2,1} + a_{2,2} \cdot (t - t_{via}) + a_{2,3} \cdot (t - t_{via})^2 + a_{2,4} \cdot (t - t_{via})^3 \tag{2.1.2}$$

for $t_{via} \leq t \leq t_f$.
The general formula for the $a_{i,j}$'s was derived in the tutorial of this Robotics course and can be specified for our case as follows.

$$a_{1,1} = q_a \tag{2.1.3}$$

$$a_{1,2} = \dot{q_a} \tag{2.1.4}$$

$$a_{1,3} = \frac{3}{(t_{via} - t_{start})^2} \cdot (q_b - q_a) - \frac{2}{t_{via} - t_{start}} \cdot \dot{q_a} - \frac{1}{t_{via} - t_{start}} \cdot \dot{q_b} \tag{2.1.5}$$

$$a_{1,4} = -\frac{2}{(t_{via} - t_{start})^3} \cdot (q_b - q_a) + \frac{1}{(t_{via} - t_{start})^2} \cdot (\dot{q_a} + \dot{q_b}) \tag{2.1.6}$$

$$a_{2,1} = q_b \tag{2.1.7}$$

$$a_{2,2} = \dot{q_b} \tag{2.1.8}$$

$$a_{2,3} = \frac{3}{(t_{end} - t_{via})^2} \cdot (q_c - q_b) - \frac{2}{t_{end} - t_{via}} \cdot \dot{q_b} - \frac{1}{t_{end} - t_{via}} \cdot \dot{q_c} \tag{2.1.9}$$

$$a_{2,4} = -\frac{2}{(t_{end} - t_{via})^3} \cdot (q_c - q_b) + \frac{1}{(t_{end} - t_{via})^2} \cdot (\dot{q_b} + \dot{q_c}) \tag{2.1.10}$$

As the different joints have different trajectories it is advisable to deduce the spline polynomials for them separately. Starting with q_1, we have to think about conditions that allow us to solve the equation of all a_{ij}. First of all, to implement a smooth spline trajectory we demand the start and end velocity $\dot{q_a}$ and $\dot{q_c}$ to be zero. In order to, know the via velocity we have to follow the heuristic approach presented to us in the tutorial of this robotics course. When looking at q_{a1}= 0, q_{b1}= $-\frac{\pi}{4}$ and q_{c1}=$-\frac{\pi}{2}$, we can see that that there is no sign change. Therefore we know that the velocity of the via should be the mean velocity that is taken to the via from the start and from the via to the end. As we have a change in angle of -$\frac{\pi}{4}$ in 2.5s in both segments $\dot{q_b}$ can be calculated as:

$$\dot{q_b} = \frac{1}{2} \cdot (-\frac{\pi}{4 \cdot 2.5s} - \frac{\pi}{4 \cdot 2.5s}) = -\frac{\pi}{10} \tag{2.1.11}$$

Now that enough conditions are established the a_{ij}'s can be calculated for the trajectory of angle q_1.

$$a_{1,1} = q_a = 0 \tag{2.1.12}$$

$$a_{1,2} = \dot{q_a} = 0 \tag{2.1.13}$$

$$a_{1,3} = \frac{3 \cdot (q_b - q_a)}{(t_{via} - t_{start})^2} - \frac{\dot{q_b}}{t_{via} - t_{start}} = -\frac{2\pi}{25} \tag{2.1.14}$$

$$a_{1,4} = -\frac{2 \cdot (q_b - q_a)}{(t_{via} - t_{start})^3} + \frac{\dot{q_b}}{(t_{via} - t_{start})^2} \approx 0.05 \tag{2.1.15}$$

$$a_{2,1} = q_b = -\frac{\pi}{4} \tag{2.1.16}$$

$$a_{2,2} = \dot{q_b} = -\frac{\pi}{10} \tag{2.1.17}$$

$$a_{2,3} = \frac{3 \cdot (q_c - q_b)}{(t_{end} - t_{via})^2} - \frac{2 \cdot \dot{q_b}}{t_{end} - t_{via}} = -\frac{\pi}{25} \tag{2.1.18}$$

$$a_{2,4} = -\frac{2 \cdot (q_c - q_b)}{(t_{end} - t_{via})^3} + \frac{\dot{q_b}}{(t_{end} - t_{via})^2} \approx 0.05 \tag{2.1.19}$$

When inserting the calculated a_{ij}'s in the Equation 2.1.1 and 2.1.2 the following trajectory spline was determined for angle q_1.

$$q_1(t) \approx -\frac{2\pi}{25}(t - t_{start})^2 + 0.05(t - t_{start})^3 \tag{2.1.20}$$

for the time interval $t_{start} \leq t \leq t_{via}$ and
for $t_{start} \leq t \leq t_{via}$

$$q_1(t) \approx -\frac{\pi}{4} - \frac{\pi}{10}(t - t_{via}) - \frac{\pi}{25}(t - t_{via})^2 + 0.05(t - t_{via})^3 \tag{2.1.21}$$

for the time interval $t_{via} \leq t \leq t_{end}$.
In contrast to q_1 the trajectory of q_2 starts at q_a=0, passes the via point at $q_b = \frac{\pi}{2}$ and goes back down to $q_c = \frac{\pi}{4}$. As q_2 encounters a sign change of the velocity the via velocity $\dot{q_{b2}}$ is zero. The starting and end velocity $\dot{q_{a2}}$ and $\dot{q_{c2}}$ is again assumed to be zero for a smooth trajectory. With this conditions a_{ij} can be calculated for trajectory q_2

$$a_{1,1} = q_a = 0 \tag{2.1.22}$$

$$a_{1,2} = \dot{q_a} = 0 \tag{2.1.23}$$

$$a_{1,3} = \frac{3 \cdot (q_b - q_a)}{(t_{via} - t_{start})^2} = \frac{6\pi}{25} \tag{2.1.24}$$

$$a_{1,4} = -\frac{2 \cdot (q_b - q_a)}{(t_{via} - t_{start})^3} \approx -0.20 \tag{2.1.25}$$

$$a_{2,1} = q_b = \frac{\pi}{2} \tag{2.1.26}$$

$$a_{2,2} = \dot{q_b} = 0 \tag{2.1.27}$$

$$a_{2,3} = \frac{3 \cdot (q_c - q_b)}{(t_{end} - t_{via})^2} = -\frac{3\pi}{25} \tag{2.1.28}$$

$$a_{2,4} = -\frac{2 \cdot (q_c - q_b)}{(t_{end} - t_{via})^3} \approx 0.10 \tag{2.1.29}$$

When inserting the calculated a_{ij}'s in the Equation 2.1.1 and 2.1.2 the following trajectory spline was determined for angle q_2.

$$q_2(t) \approx \frac{6\pi}{25}(t - t_{start})^2 - 0.2(t - t_{start})^3 \tag{2.1.30}$$

for the time interval $t_{start} \leq t \leq t_{via}$ and
for $t_{start} \leq t \leq t_{via}$

$$q_2(t) \approx -\frac{\pi}{4} - \frac{3\pi}{25}(t - t_{via})^2 + 0.1(t - t_{via})^3 \tag{2.1.31}$$

The plots were created by using the formulas directly and not working with the rounded values as those create not entirely smooth trajectories. This can be avoided to an extent by rounding a_{14} and a_{24} to the 5th decimal after the comma instead of the second (for q_1: $a_{14} = a_{24} \approx 0.05026$ and for q_2: $a_{14} \approx 0.20106,\ a_{24} \approx 0.10053$). Besides that it must be noted that even though q_3 is included in the angle vector it starts and ends at zero, that's why no trajectory needed to be calculated, as all the a_{ij}'s would become zero.

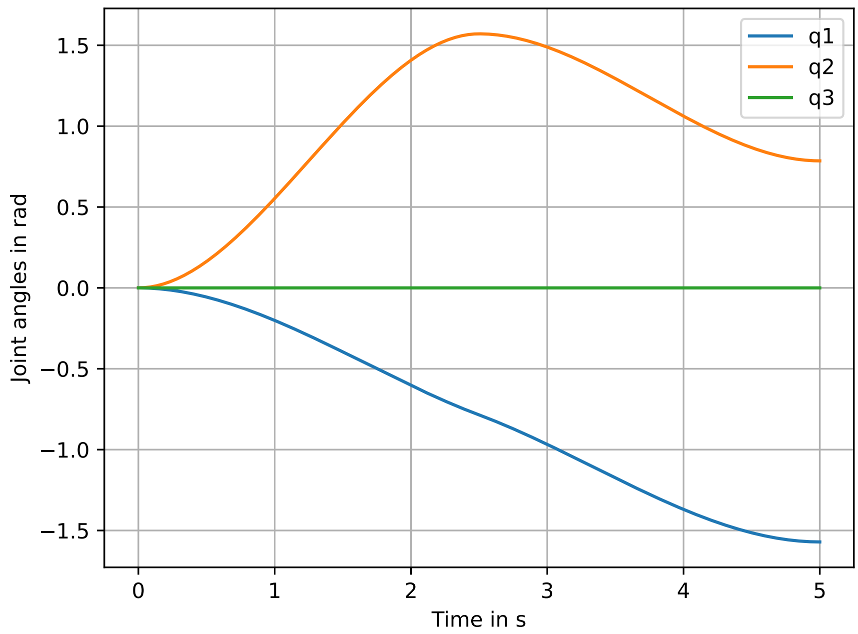


Figure 5: Cubic Spline trajectory joint angles with respect to time.

2.2 B2 - Trajcetory Generation

2.2.1 a

In this task we need to write a function that calculates the time Tf that it takes for the trajectory from start to the end to reach a defined angle configuration while fulfilling the constrains defined in the task.

$$|\dot{q}_i| \leq \dot{q}_{max_i}(\text{gv.qdmax}) \tag{2.2.1}$$

$$|\ddot{q}_i| \leq \ddot{q}_{max_i}(\text{gv.qddmax}) \tag{2.2.2}$$

As the task demands no as fast as a possible trajectory or a minimal Tf time, it could be solved trivially. One could calculate what the maximum constraints are and simply calculate the time Tf by the equation.

$$t_f = \frac{|q_{end} - q_{start}|}{\gamma \cdot \dot{q}_{max_i}} \tag{2.2.3}$$

Where $0 \leq \gamma \leq 1$ is a factor that we choose arbitrarily low, to definitely fulfill the constraints but not have the fastest trajectory implementation.
However, it was decided that it was more interesting to try to approach this task from a demand of minimum rise time while still fulfilling constraints sufficiently. For this several calculations had to be made in order to find out how to calculate the minimum Tf depending on the change in angle $q_{end} - q_{start}$ and the velocity/acceleration constraint. For this purpose, the first two equations must be found to fulfill the max velocity and max acceleration criteria and then a rule must be made to choose the stricter criteria. Because if for example, the user sets the max acceleration to a small value and the max velocity to a relatively high value, the Tf should be chosen that way that it fulfills the stricter criteria of acceleration.
In the following the start angle configuration and starting time will be called q_s and t_s respectively and the final configuration time when its reached are called q_f and t_f respectively. As there is no via in the trajectory generation and the initial velocity should be zero, the equation for the trajectory generation can be simplified to the following.

$$q(t) = a_1 + a_3 \cdot (t - t_s)^2 + a_4 \cdot (t - t_s)^3 \tag{2.2.4}$$

With the first and second derivatives being Equation 2.2.5 and 2.2.6.

$$\dot{q}(t) = 2 \cdot a_3 \cdot (t - t_s) + 3 \cdot a_4 \cdot (t - t_s)^2 \tag{2.2.5}$$

$$\ddot{q}(t) = 2 \cdot a_3 + 6 \cdot a_4 \cdot (t - t_s) \tag{2.2.6}$$

As we assume that the starting and end velocity is zero we can plug in the formulas for a_i in Equations 2.2.5 and 2.2.6

$$\dot{q}(t) = 2 \cdot \frac{3(q_f - q_s)}{(t_f - t_s)^2} \cdot (t - t_s) + 3 \cdot \frac{-2(q_f - q_s)}{(t_f - t_s)^3} \cdot (t - t_s)^2 \tag{2.2.7}$$

$$\ddot{q}(t) = 2 \cdot \frac{3(q_f - q_s)}{(t_f - t_s)^2} + 6 \cdot \frac{-2(q_f - q_s)}{(t_f - t_s)^3} \cdot (t - t_s) \tag{2.2.8}$$

First lets investigate the maximum velocity criteria. The Equation 2.2.7 is a parabola and from it can be deduced that when the change in angle q_f-q_s is positive, it has a maximum and if the change is negative it has only a minimum. As the velocity starts and ends at a value of zero this indicates that the max absolute velocity (positive or negative sign) is reached at the maximum/minimum of the function.
The next consequential step is to find the point in time for the maxima and input this time point in the Equation 2.2.4 to find out its value at this point in time.

$$\ddot{q}(t) = 0 = \frac{6(q_f - q_s)}{(t_f - t_s)^2} - \frac{12(q_f - q_s)(t - t_s)}{(t_f - t_s)^3} \tag{2.2.9}$$

$$t_{|\dot{q}_{max}|} = \frac{t_f - t_s}{2} \tag{2.2.10}$$

Reinsert this into the velocity Equation 2.2.7 to get the maximum velocity and insert it in the initial velocity constraint.

$$\dot{q}_{max_i} \geq |\dot{q}(t_{|\dot{q}_{max}|})| \tag{2.2.11}$$

$$\dot{q}_{max_i} \geq |\frac{6(q_f - q_s)}{(t_f - t_s)^2} \cdot (\frac{t_f - t_s}{2} - t_s) - \frac{6(q_f - q_s)}{(t_f - t_s)^3} \cdot (\frac{t_f - t_s}{2} - t_s)^2| \tag{2.2.12}$$

$$\frac{\dot{q}_{max_i} \cdot (t_f - t_s)^3}{|6(q_f - q_s)|} \geq |(t_f - t_s) \cdot (\frac{t_f - t_s}{2} - t_s) - (\frac{t_f - t_s}{2} - t_s)^2| \tag{2.2.13}$$

In our case we can simply define t_s to be zero, as we can program a simple time variable that is set to the current time during the init function of the trajectory generation. This simplifies the formula and provides us with the following equation.

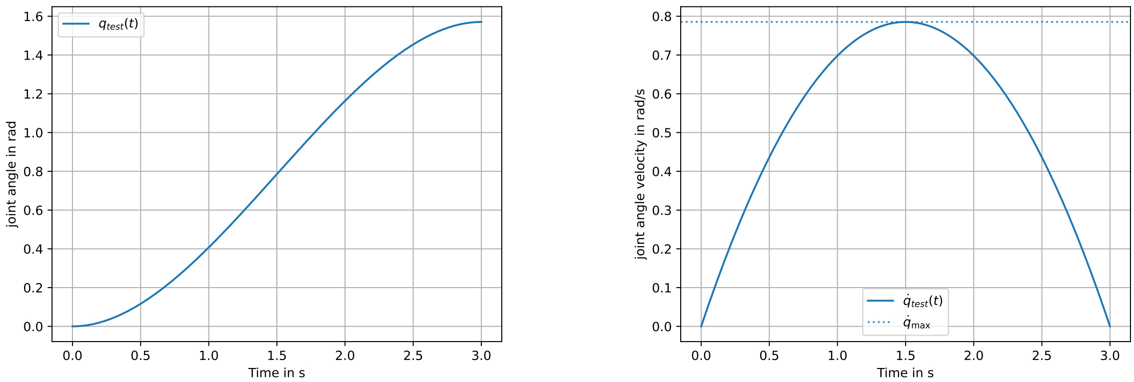
$$t_f \geq \frac{|1.5(q_f - q_s)|}{\dot{q}_{max}} \tag{2.2.14}$$

This equation shows that for a specific change in angle and a predefined max angular velocity the t_f must be at least equal to the formula to reach the maximum velocity. To test the formula assume the case that the maximum absolute allowed angular velocity is $\frac{\pi}{4}\frac{rad}{s}$, q_s=0, $q_f = \frac{\pi}{2}$ and $t_s = 0$. Inserting this in Equation 2.2.14, provides us with a $t_f = 3$. Inserting this in Equation 2.2.4 and calculate a_3 and a_4 accordingly provides us with the trajectory spline polynomial.

$$q_{test}(t) = \frac{3(q_f - q_s)}{t_f^2} \cdot t^2 + \frac{-2(q_f - q_s)}{t_f^3} \cdot t^3 \tag{2.2.15}$$

$$q_{test}(t) = \frac{\pi}{6} \cdot t^2 - \frac{\pi}{27} \cdot t^3 \tag{2.2.16}$$

$$\dot{q}_{test}(t) = \frac{\pi}{3} \cdot t - \frac{\pi}{9} \cdot t^2 \tag{2.2.17}$$



(a) q_{test} plotted over time. (b) $\dot{q}_{test}$ plotted with specified $\dot{q}_{max}$.

Figure 6: Cubic spline trajectory of q_{test} and $\dot{q}_{test}$.

As we can see the derived Equation 2.2.14 for the shortest execution time t_f possible with the maximum angular velocity as possible works exactly as intended. However, it might be later beneficial to adjust it by adding a safety margin to the maximum allowed velocity, so that in the real tests we do not overshoot the max velocity.

Now that the minimum criteria according to the velocity is clear its time to look at the acceleration. The acceleration of the cubic spline polynomial is a linear term which absolute maximum at timepoint t_s and t_f. This seems plausible when looking at Figure 6a as in the beginning and in the end the absolute acceleration is maximum and in the middle, a nearly constant velocity is reached. Therefore we know that $\ddot{q}(t)$ will be max for $t = t_s$. To get the t_f condition to fulfill the $|\ddot{q}(t)| \leq \ddot{q}_{max}$, we need to insert the timepoint $t = t_s$ in Equation 2.2.8 and assume that we can choose t_s to be our starting timepoint so that $t_s = 0$ holds.

$$\ddot{q}_{max} \geq |\ddot{q}(t)| \tag{2.2.18}$$

$$\ddot{q}_{max} \geq |\frac{6(q_f - q_s)}{t_f^2}| \tag{2.2.19}$$

$$t_f \geq \sqrt{\frac{|6(q_f - q_s)|}{\ddot{q}_{max}}} \tag{2.2.20}$$

Assuming the same example as previous but we say that the maximum allowed acceleration $\ddot{q}_{max} = \frac{\pi}{3}\frac{rad}{s}$. According to the Equation 2.2.20 the compute minimum t_f= would also be 3s. So we generate the same polynomial spline as previously but plot it alongside the maximum allowed acceleration.

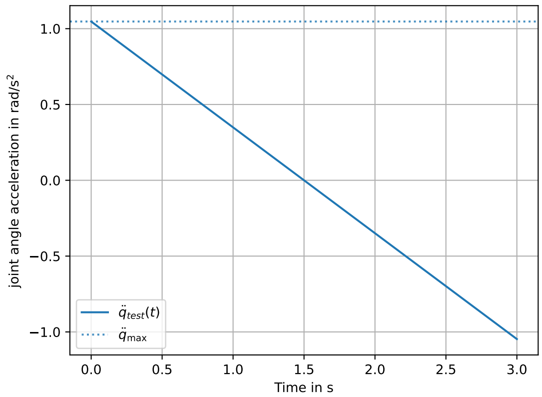


Figure 7: Cubic spline trajectory of $\ddot{q}_{test}$ and $\ddot{q}_{max}$.

Figure 7 shows us that the formula for the minimum t_f with acceleration contains is working perfectly. To select the stricter of the two criteria so that t_f fulfills both constraints, the Equation can be written as follows.

$$t_f = max(\frac{|1.5(q_f - q_s)|}{\dot{q}_{max}}, \sqrt{\frac{|6(q_f - q_s)|}{\ddot{q}_{max}}}) \tag{2.2.21}$$

However, it is important to add that all the joint angles have to reach their desired trajectory at the same time, so the t_f must be chosen according to the largest $q_i - q_{d_i}$ (current joint angle - desired joint angle). Therefore the final formula which computes the common minimum t_f for all joint trajectories which fulfills both constraints while still being as fast as possible can be defined as.

$$t_{f1} = max(\frac{|1.5(q_{f1} - q_{s1})|}{\dot{q}_{max}}, \sqrt{\frac{|6(q_{f1} - q_{s1})|}{\ddot{q}_{max}}}) \tag{2.2.22}$$

$$t_{f2} = max(\frac{|1.5(q_{f2} - q_{s2})|}{\dot{q}_{max}}, \sqrt{\frac{|6(q_{f2} - q_{s2})|}{\ddot{q}_{max}}}) \tag{2.2.23}$$

$$t_{f3} = max(\frac{|1.5(q_{f3} - q_{s3})|}{\dot{q}_{max}}, \sqrt{\frac{|6(q_{f3} - q_{s3})|}{\ddot{q}_{max}}}) \tag{2.2.24}$$

$$t_{fmin} = max(t_{f1}, t_{f2}, t_{f3}) \tag{2.2.25}$$

In the code we added an additional safety margin to the calculated t_{fmin}, by in creasing it by 5% of its length.

2.2.2 c) Trajectory to a specific point

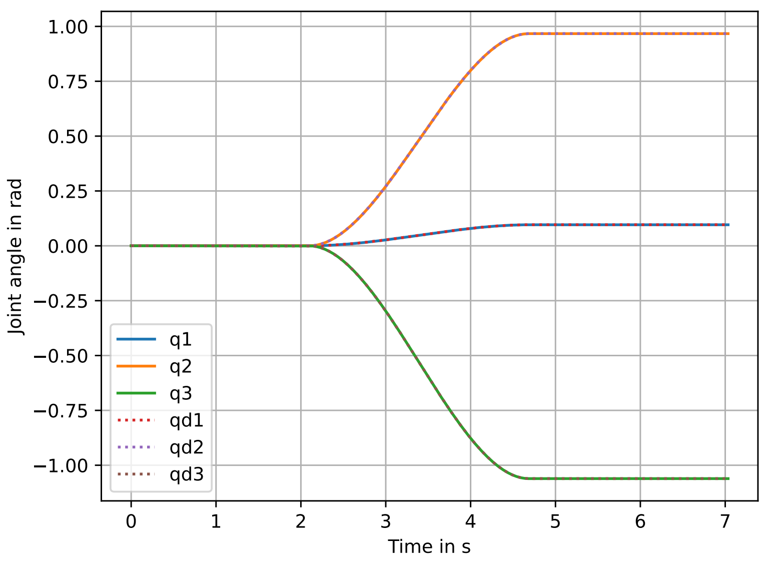


Figure 8: q and qd of the cubic spline trajectory of proj1 with gains k_{p1}=54400, k_{p2}=44400, k_{p3}=10400, k_{v1}=240, k_{v2}=140, k_{v3}=140.

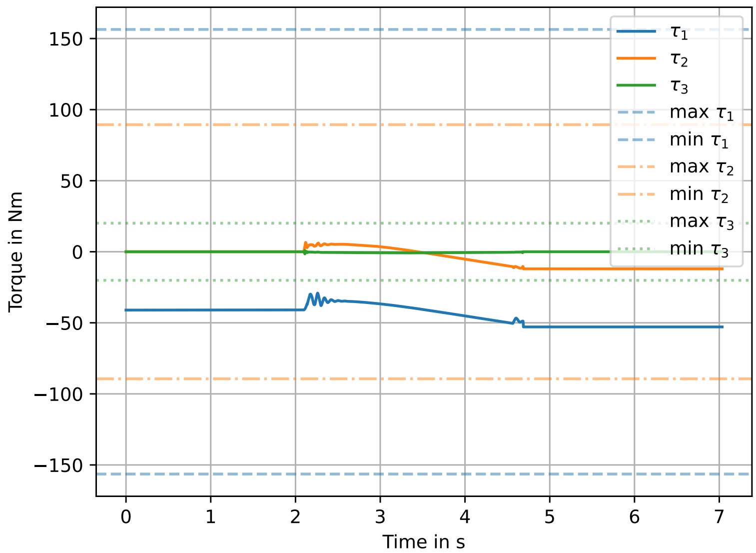


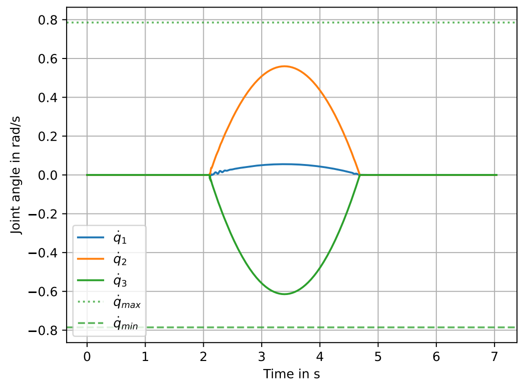
Figure 9: Torque of the cubic spline trajectory of proj1 with gains k_{p1}=54400, k_{p2}=44400, k_{p3}=10400, k_{v1}=240, k_{v2}=140, k_{v3}=140.

As one can see in Figures 8 and 9 the cubic spline trajectory is well followed by proj1 control. The gains can be chosen significantly higher than in the assignment 1 as the differences in angle change only very slowly with the trajectory per timestep. As one can see this also shows itself in the relatively small torque that is needed to initiate the motion.

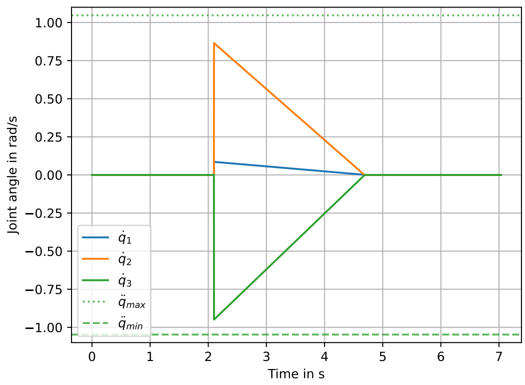
In the following Figure 10 the angular velocity and acceleration is shown. From the figures we can presume that the stricter constraint was the max acceleration as the acceleration plot of q_3 is closest to the max allowed value. From the GUI of the Pumasim we can extract that the maximum allowed velocity is per default set to $45\frac{\circ}{s}$ and max acceleration to $60\frac{\circ}{s^2}$. When writing those in rad/s and rad/s^2 we can insert it in Equation 2.2.24 as q_3 has the largest change in angle and will therefore require the longest t_f.

$$t_{f_{min}} = max(\frac{|1.5 \cdot (-1.061\ rad)|}{\frac{\pi}{4} \frac{rad}{s}}, \sqrt{\frac{|6 \cdot (-1.061\ rad)|}{\frac{\pi}{3} \frac{rad}{s^2}}}) \approx max(2.03s, 2.47s) = 2.47s \tag{2.2.26}$$

Additionally the previously mentioned 5% safety margin were added by the code to the execution time of the spline so that t_f=$t_{f_{min}} \cdot 1.05 \approx 2.59$s. This execution time is shown in the Figures and especially in Figure 10b one can see that the acceleration of q_3 is close to the max acceleration with a margin of around 5%.



(a) $\dot{q}$ plotted with specified $\dot{q}_{max}$.



(b) $\ddot{q}$ plotted with specified $\ddot{q}_{max}$.

Figure 10: $\dot{q}$ and $\ddot{q}$ of the cubic spline trajectory of proj1 with gains k_{p1}=54400, k_{p2}=44400, k_{p3}=10400, k_{v1}=240, k_{v2}=140, k_{v3}=140.and_{test}.

3 Task C

3.1 Project2-Circle

The endeffector needs to move in a circular anticlockwise motion as the z-axis of the simulator frame is pointing out of the scheme in Figure 1. Therefore the position trajectory of the end effector can be calculated by the following formula:

$$x_d(t) = x_c + rcos(\dot{\beta}t) \tag{3.1.1}$$

$$y_d(t) = y_c + rsin(\dot{\beta}t) \tag{3.1.2}$$

With the center point, the velocity and the radius defined as followed:

$$x_{center} = (x_c, y_c) = (0.8m, 0.35m) \tag{3.1.3}$$

$$r = 0.2m \tag{3.1.4}$$

$$\dot{\beta} = \frac{2\pi}{5s} \tag{3.1.5}$$

The desired velocity trajectory of the end effector can be consequently defined as:

$$\dot{x}_d(t) = \frac{x(t)}{dt} = -\dot{\beta}rsin(\dot{\beta}t) \tag{3.1.6}$$

$$\dot{y}_d(t) = \frac{y(t)}{dt} = \dot{\beta}rcos(\dot{\beta}t) \tag{3.1.7}$$

The controller must keep the position of the end effector as close to the desired trajectory as possible while keeping the angle of the end effector upright. The control law to accomplish this can therefore be defined as:

$$F = \begin{pmatrix} F_x \\ F_y \\ F_\alpha \end{pmatrix} \tag{3.1.8}$$

$$F_x(t) = -k_{p1} \cdot (x(t) - x_d(t)) - k_{v1} \cdot (\dot{x}(t) - \dot{x}_d(t)) \tag{3.1.9}$$

$$F_y(t) = -k_{p1} \cdot (y(t) - y_d(t)) - k_{v1} \cdot (\dot{y}(t) - \dot{y}_d(t)) \tag{3.1.10}$$

$$F_\alpha(t) = -k_{p1} \cdot (\alpha(t) - 0) - k_{v1} \cdot (\dot{\alpha}(t) - 0) \tag{3.1.11}$$

The torque that needs to be applied to the joints to control the trajectory was given in the lecture and is defined as.

$$\tau = J^T F(t) \tag{3.1.12}$$

In the following τ, x, x_d and e are plotted for the tuned gains first (see Figures 11, 12, 13, 14, 15). Later on for comparison lower and higher gains results can be seen (see Figures 17, 16, 19).

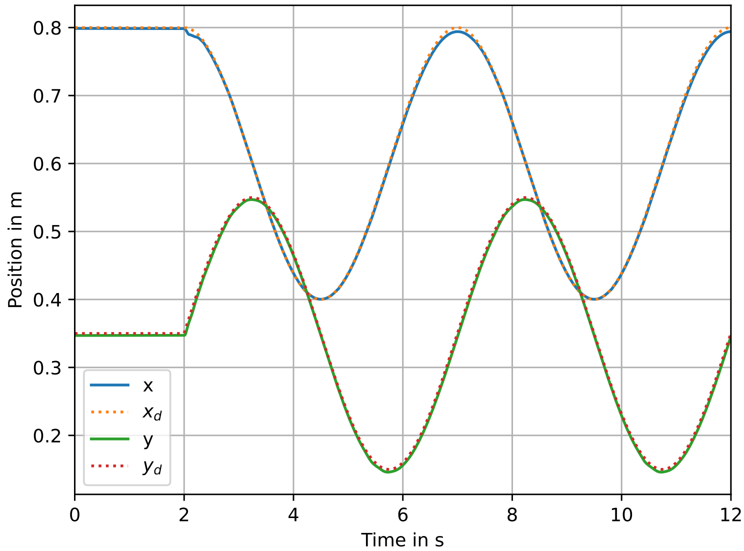


Figure 11: Desired and reached trajectory of proj2 control with tuned gains k_{p1}=26400, k_{p2}=22400, k_{p2}=1700, k_{v1}=140, k_{v2}=80, k_{v3}=80.

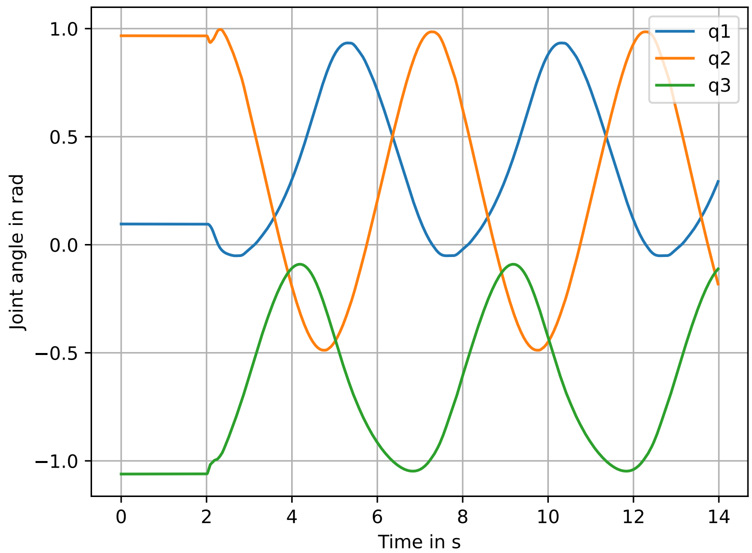


Figure 12: Joint angles during proj2 control with tuned gains k_{p1}=26400, k_{p2}=22400, k_{p2}=1700, k_{v1}=140, k_{v2}=80, k_{v3}=80.

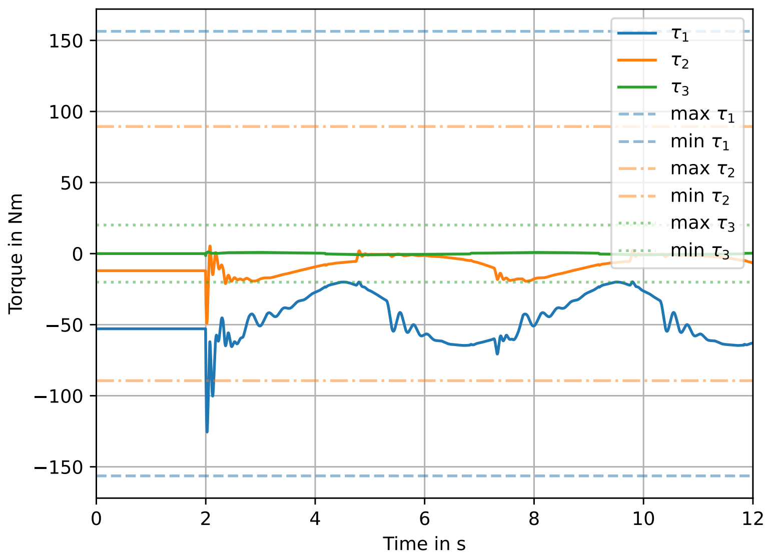


Figure 13: Joint torques during proj2 control with tuned gains k_{p1}=26400, k_{p2}=22400, k_{p2}=1700, k_{v1}=140, k_{v2}=80, k_{v3}=80.

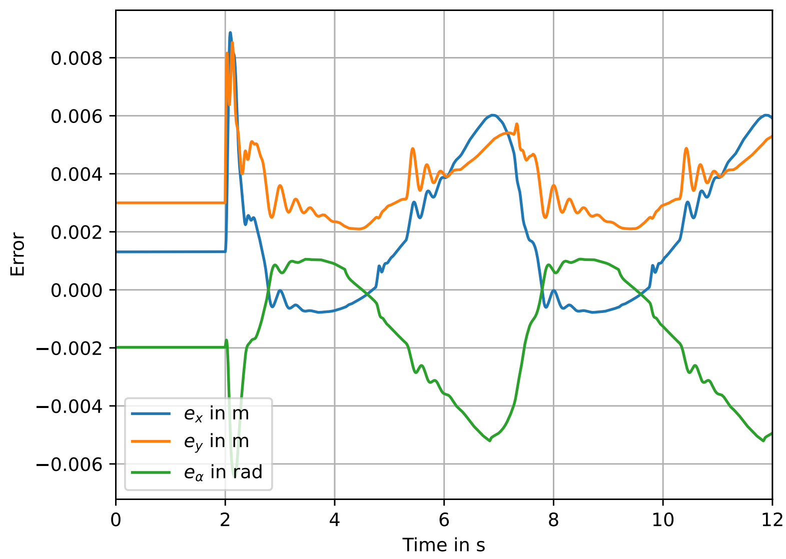


Figure 14: Position error during proj2 control with tuned gains k_{p1}=26400, k_{p2}=22400, k_{p2}=1700, k_{v1}=140, k_{v2}=80, k_{v3}=80.

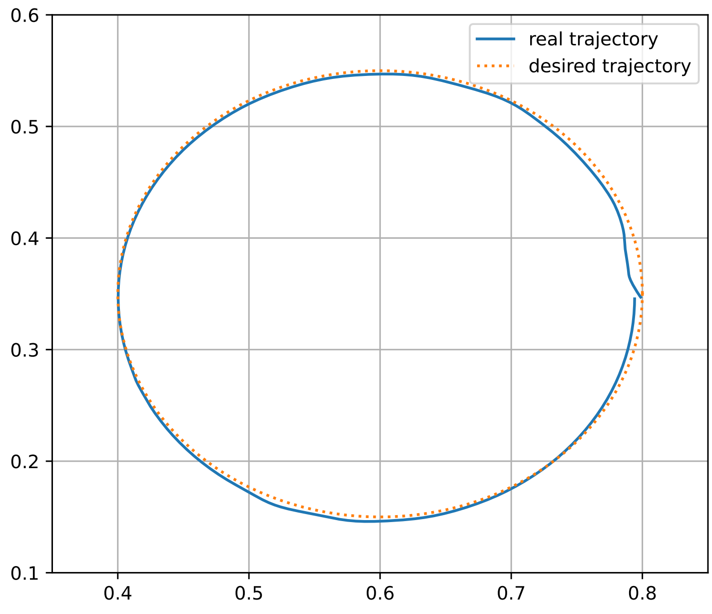
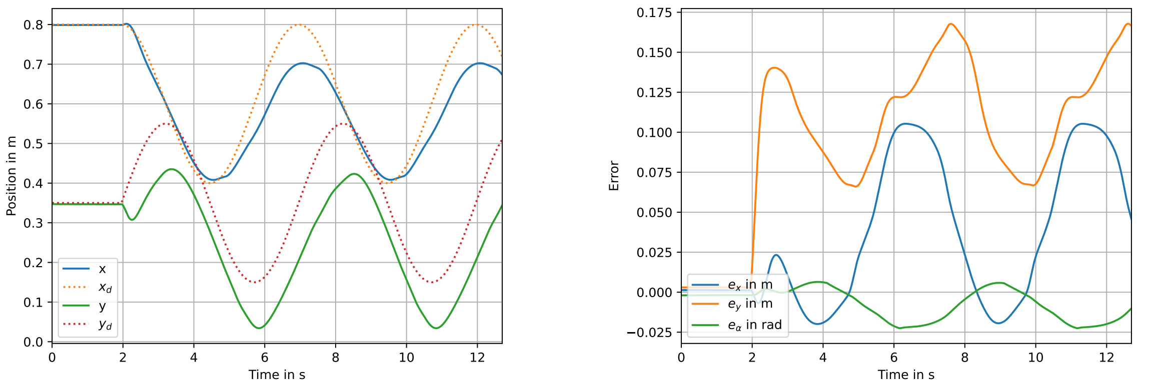


Figure 15: X-Y Plane trajectory during proj2 control with tuned gains k_{p1}=26400, k_{p2}=22400, k_{p2}=1700, k_{v1}=140, k_{v2}=80, k_{v3}=80.

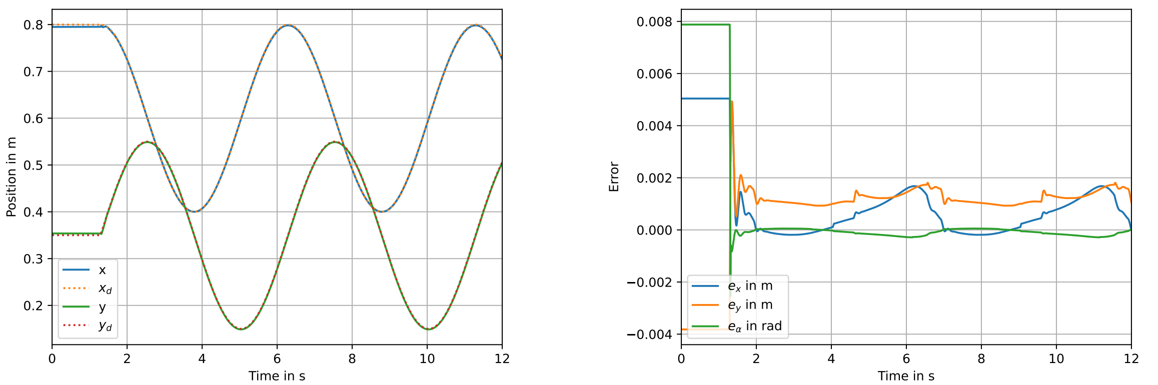
The gains k_{pi} were limited by the allowed max torque. At the beginning, where the end effector is not moving at the starting position and then should start to move is the position where maximum torques need to be applied. Therefore the gains for the tuned proj2 control were chosen so that such max torque is not exceeded. As one can see in the following figures, lower gain sets increase the error between x and x_d (y-y_d and α and α_d). Besides that, the controller is not fast enough to follow the trajectory, so it even has a phase delay (see Figure 16). Therefore in the error graph we can see that the lower gain controller has higher amplitude errors with a phase shift. The error graph for the higher gains in Figure 17b has a smaller amplitude and is not as strongly phase-dependent as the lower gain controller. However, we can see that when looking at Figure 19, the small k_{pi}'s have a way smaller spike when initiating the movement. While the large gain k_pi's cause the τ to overshoot the maximum allowed boundary.



(a) x and x_d plotted over two revolutions.

(b) e_x, e_y and e_α plotted over two revolutions.

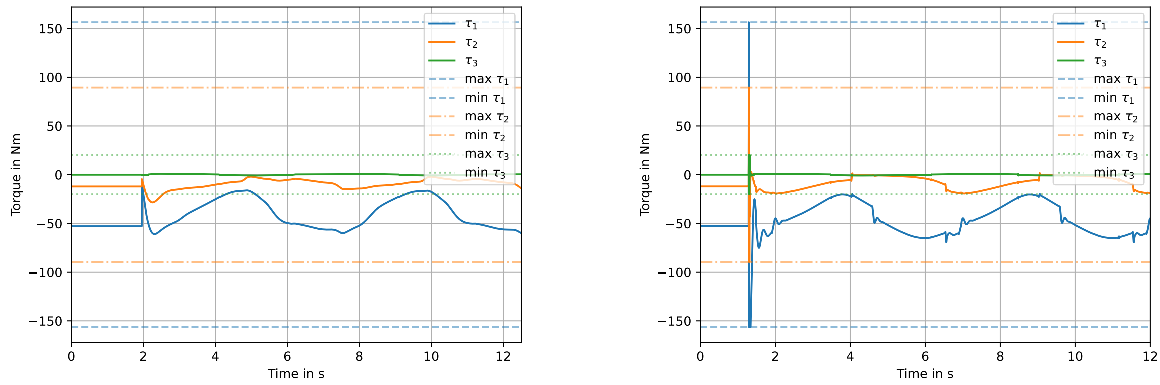
Figure 16: Proj2 control with smaller gains: k_{p1}=800, k_{p2}=600, k_{p2}=200, k_{v1}=40, k_{v2}=40, k_{v3}=40.



(a) x and x_d plotted over two revolutions.

(b) e_x, e_y and e_α plotted over two revolutions.

Figure 17: Proj2 control with higher gains: k_{p1}=99400, k_{p2}=65400, k_{p2}=33400, k_{v1}=1240, k_{v2}=1240, k_{v3}=140.



(a) τ for gains: k_{p1}=800, k_{p2}=600, k_{p2}=200, k_{v1}=40, k_{v2}=40, k_{v3}=40.

(b) τ for gains: k_{p1}=99400, k_{p2}=65400, k_{p2}=33400, k_{v1}=1240, k_{v2}=1240, k_{v3}=140.

Figure 18: Proj2 control τ for lower and higher gainset.

3.2 Project3-Parabolic Blend

First, we have to compute $t_{f,min}$. We need this later for calculating the blend time t_b.
The minimum parabolic blend trajectory consists of two blend phases. Then, $t_{f,min} = 2 \cdot t_b$.
The end effector has to make 3 full circles, so $\beta(t_{f,min}) = 6\pi$.
After one blend phase, we have $\beta(t_b) = 0.5 \cdot \ddot{\beta}_{max} \cdot t_b^2$.
After two blend phases (complete trajectory), we have $\beta(t_{f,min}) = \ddot{\beta}_{max} \cdot t_b^2$.
Now we can write:

$$\ddot{\beta}_{max} \cdot t_b^2 = 6\pi$$

So, solving for t_b^2, we get:

$$t_b^2 = \frac{6\pi}{\ddot{\beta}_{max}}.$$

Now we can substitute this into the expression for $t_{f,min}^2$:

$$t_{f,min}^2 = 4 \cdot t_b^2 = 4 \cdot \frac{6\pi}{\ddot{\beta}_{max}} = \frac{24\pi}{\ddot{\beta}_{max}}.$$

Substituting $\ddot{\beta}_{max} = \frac{2\pi}{25}$, we get:

$$t_{f,min}^2 = \frac{24\pi}{\frac{2\pi}{25}} = 24\pi \cdot \frac{25}{2\pi} = 300.$$

Thus, we find:

$$t_{f,min} = \sqrt{300} \approx 17.32s.$$

We chose $t_f = 20s$

2-Trajectory equations

1. **Acceleration phase:** for $t \in [0, t_b]$

$$\beta(t) = \frac{1}{2} \cdot \ddot{\beta}_{max} \cdot t^2$$

2. **Constant velocity phase:** for $t \in [t_b, t_b + t_c]$

$$\beta(t) = \frac{1}{2}\ddot{\beta}_{max}t_b^2 + \dot{\beta}_{\max}(t - t_b) \quad with \quad \dot{\beta}_{\max} = t_b \cdot \ddot{\beta}_{max}$$

3. **Deceleration phase:** for $t \in [t_b + t_c, t_f]$

$$\beta(t) = \beta(t_f) - \frac{1}{2}\ddot{\beta}_{max}(t - t_f)^2$$

As we see to get the trajectory we have to find t_b:

$$\beta(t_b) = 0.5\ddot{\beta}_{max}t_b^2$$

$$\beta(t_c) = 0.5\ddot{\beta}_{max}t_b^2 + \ddot{\beta}_{max}t_b(t_c - t_b)$$

$$\beta(t_c) = 0.5\ddot{\beta}_{max}t_b^2 + \ddot{\beta}_{max}t_b(t_f - 2t_b)$$

$$\beta(t_c) = -1.5\ddot{\beta}_{max}t_b^2 + t_b \cdot t_f$$

$$\beta(t_f) = \beta(t_c) + \beta(t_b) = 0.5\ddot{\beta}_{max}t_b^2 - 1.5\ddot{\beta}_{max}t_b^2 + \ddot{\beta}_{max}t_b \cdot t_f = 6\pi$$

$$\beta(t_f) = -\ddot{\beta}_{max}t_b^2 + \ddot{\beta}_{max}t_b \cdot t_f = 6\pi$$

$$\ddot{\beta}_{max}t_b^2 - \ddot{\beta}_{max}t_b \cdot t_f + 6\pi = 0$$

$$t_b = \frac{t_f \pm \sqrt{t_f^2 - \frac{6\pi}{\ddot{\beta}_{max}}}}{2}$$

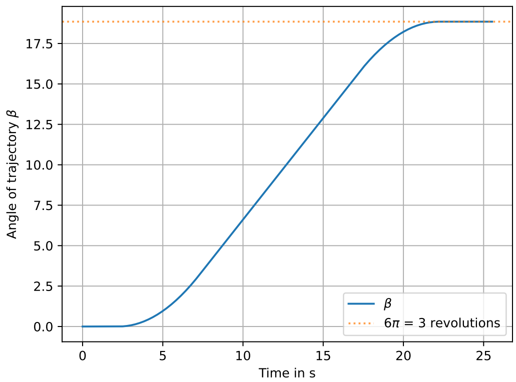
We chose $\ddot{\beta}_{max} = \frac{2\pi}{25s^2} \quad and \quad t_f = 20s$
So we get $t_{b1} = 5s \quad and \quad t_{b2} = 15s$
we have $t_f - 2t_{b2} = -10s$ this must be within the time interval of the trajectory. Also we reject this solution. and the solution is $t_{b1} = t_b = 5s$.
Now we have equation of the angular position of the end effort but we need for the simulation a coordinate.

$$x = rcos(\beta(t)) + x_0$$

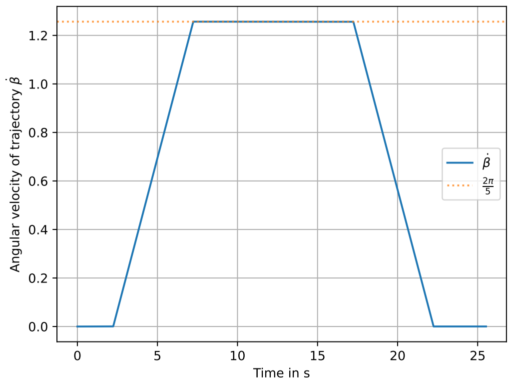
$$y = rsin(\beta(t)) + y_0$$

$$\dot{x} = -r\dot{\beta}(t)sin(\beta(t))$$

$$\dot{y} = r\dot{\beta}(t)cos(\beta(t))$$



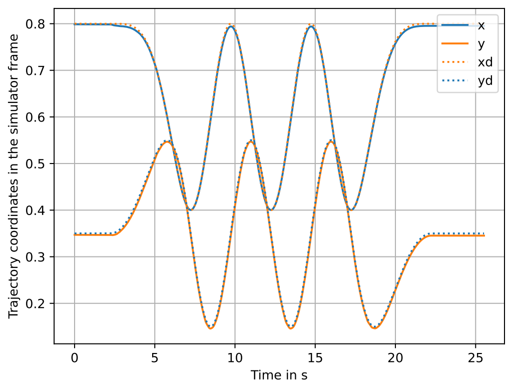
(a) Position of angle β(t).



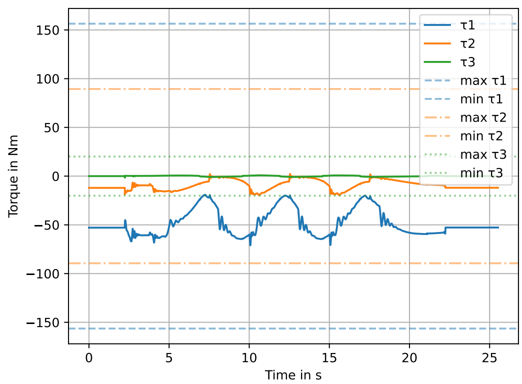
(b) Angular velocity $\dot{\beta}(t)$.

Figure 19: Position and angular velocity in proj3.

While choosing the same gains as in the proj2 control we can see that through the parabolic blends we do not required rapid movements in the beginning of the motion and therefore experience no spike in torque at the start of the motion (see Figure 20b). As we can additionally see in Figure 20a the desired trajectory is followed well. So in theory this enables us to use higher gains to have less error while still fulfilling the constraints on maximum torque expense.



(a) Desired trajectory xd/yd and controlled trajectory x/y.



(b) Torque required for trajectory tracking.

Figure 20: Further graphs displaying trajectory and torque.

Student Name	A1	A2	A3	A4	A5	B1	B2	C1	C2	C3	C4	C5	Documentation
Maxim Fenko	X	X	X	X	X	X	X	X	X	X	X	X	X
Abdelkarim Ben Salah	X	X	X	X	X	X	X	X	X	X	X	X	X
Bryan Oppong-Boateng	X	X	X	X	X	X	X						X