Introduction to Robotics 035001

Spring-Summer 2024

Course Project

Made by:

Alon Ben-David

And help from Salih B and Jonathan J, and Roni AL.

Submitted In:

10/11/2024

Contents

Problem, Solutions, Difficulties, Assumptions and Results

Summary of Problem

Serial 6R Robot (FANUC model M-10iA) is fixed on a rail. The arm moves in a nuclear waste storage corridor in $V = 1\left[\frac{m}{s}\right]$.

A camera is installed on the robot gripper to scan the barcodes.

The distance between each barrel is 50 cm and their diameter is 60 cm.

The bar code of the barrels is 80 cm above the floor.

For the robot to scan the barcode with the camera, the camera must be perpendicular to the barrel (assume the same direction for the camera and the last link).

For an adequate scan, a zero relative speed is required for at least 0.5 second between the camera and the barcode, at 20-30 cm from the bucket.

The weight of the camera is $m = 0.7[kg]$.

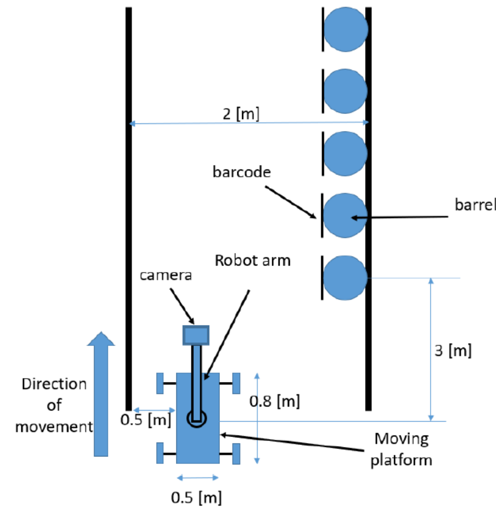


illustration1: A system in which the robot moves, including the moving platform on which the robotic arm is assembled and barrels with barcodes

Assumptions:

1. Weight of each link is $M_i = M = 4[kg]$ with a center of gravity in the center of the line between two joints.
2. The dimensions of the camera and the gripper are negligible compared to the dimensions of the arm and the geometry of the system.
3. Dynamic forces are negligible.
4. Same direction for camera and last link.
5. **Let's assume that the robot arm is in the zero-position described in the robot specification, in the direction described in Figure 1 and at joint angles corresponding to the start of the trajectory, as will be described in the trajectory planning chapter.**

purpose:

The robot must scan the barcode on the first bucket only along the wall.

Tasks:

1. Propose and sketch an optimal location for the robot with the workspace, in addition determine the height of the robot's base.
2. Select points that make up the barcode reading path for only one bucket (check flats error).
3. Calculate the direct kinematics of the robot.
4. Calculate the inverse kinematics of the robot. If there are multiple solutions, determine an appropriate selection criterion.
5. Calculate the full Jacobian of the robot.

6. Find singular points for the linear velocity Jacobian. Draw these points and explain their meaning.
7. Design the joint-space based trajectory of the robot, when the robot starts in the state in Figure 1, and it is going to read only one barcode. Instructions:
   - Choose a polynomial-articulated trajectory that starts and ends at velocity 0.
   - Between 2 points a time interval of 0.01 seconds.
   - The track must meet requirements on the speed of the catch and the joints.
   - As mentioned before, the robot must read the barcode for at least 0.5 second.

In addition, there are tasks related to data and graphs that must be presented in the project submission.

The chosen way to solve the problem

We chose to solve the problem by dividing the project into three main parts:

1. Performing the analytic analysis of the industrial 6R Robot of the project: Direct and inverse kinematics, finding the full Jacobian and the places in space where it has a singularity (linear, angular or mixed).
2. Planning the route considering the task, conversion to joint values and familiarization with the limitations: Building the desired track and orientation in time according to the requirements and the main situations in which the gripper will pass, connecting the tracks continuously with speed and acceleration and converting these requirements to the functions of the robot's joints in time.
3. Analysis of the results: Drawing the movement of the gripper, drawing graphs to check the execution of the task and compliance with the limitations. Checking the speeds, accelerations and torques in the robot.

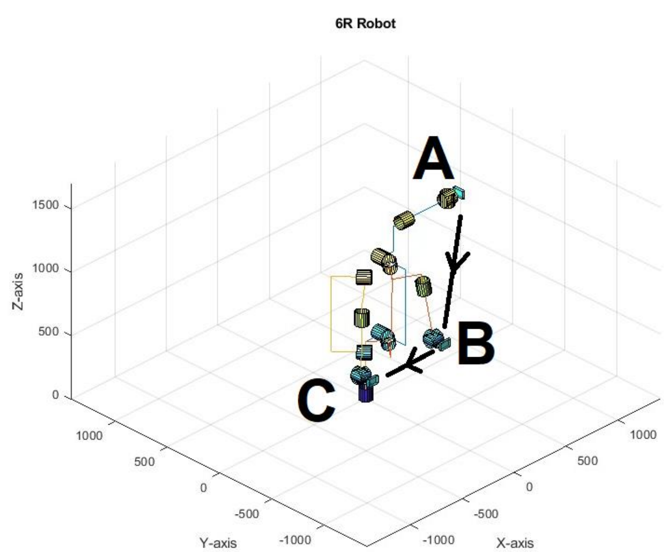
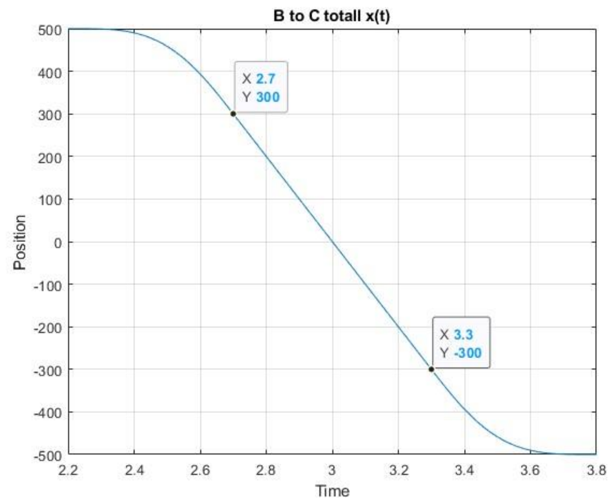


illustration2: A planned route for the gripper in 3D space

First, we will note that relative to the base of the robot, the barcode is a point moving at the opposite speed of the robot itself, this is how we will refer to it throughout the project.

In the trajectory planning part, the gripper will start in the zero state of the robot - state A (with joint angles 4,6 necessary to perform the route continuously), will move in a straight line within a certain time to point B with zero speed and acceleration and change the orientation continuously to that needed to scan the barcode. From state B to C, we will make a path for tracking the barcode consisting of three parts:

1. Acceleration within 0.5 second from rest to the speed of the barcode relative to the base of the robot $V = 1\left[\frac{m}{s}\right] = 1000\left[\frac{mm}{s}\right]$and zero acceleration.
2. Reach a constant speed and continue at this speed for 0.6 seconds. That is, the scan is performed at this current interval.
3. Deceleration within 0.5 second from constant speed and zero acceleration to rest.



Graph 1: position of the gripper along an axis x0 during the scanning movement from point B to C

This completes the robot's path.

The difficulties in solving the project and choosing the route solution

In the inverse kinematics:

The solution for the robot6R was challenging and required attention to the methods taught in the lecture and the geometric features of its movement. In the solution for an industrial robot, it is important to decompose the problem into a position and orientation problem, where the first three joints, $(\theta_1, \theta_2, \theta_3)$, are sufficient for the position solution. After solving the position, we move on to solve the rotation using the second triple joints $(\theta_4, \theta_5, \theta_6)$.

From a kinematic point of view, the gripper behaves as a Rigid-Body with 6 degrees of freedom, leading to 8 possible discrete solutions (without singularities). By checking the states in the code, the appropriate branch of the solution is chosen for planning the trajectory. This choice is critical for the continuity of the trajectory and the values of the joints in time.

In code and Jacobian:

In the results of the direct kinematics and the Jacobian you get great results because of the geometric complexity of this robot. We were not able to get from MATLAB the determinant needed to determine what the singular states of the linear Jacobian are.

To simplify the analysis and obtain solutions, we moved on to solve the overall singularity of the robot. We found an article that shows that this analysis can be simplified by assuming the determinant of the large Jacobian and then from there get the 3 cases in which the robot is in a general singularity.

In planning the route:

We had to choose between two main methods: planning Trajectory in joint space or Cartesian space. In any case, the robot receives at the end functions of joint values in time, which are translated into currents in the motors, which limits the accelerations to be at least continuous in time (i.e. continuous speed and shear in time).

We realized that tracking required a precise route in a straight line at a constant speed, so we focused on planning in Cartesian space. Bringing the catcher to the preparation point was also planned in the space Cartesian. We planned the times parametrically in MATLAB for maximum flexibility. The main challenge in planning was to bring the gripper to Zero relative speed in relation to the barcode while moving in front of oriented perpendicular to it.

Beyond building the desired route, there is the need to stay away from singular points on the route and therefore the importance of finding them in advance if possible. Even after the desired trajectory is ready, due to passing through singular points in the trajectory there may be multiple solutions or a requirement for the robot to use huge joint speeds. Knowing them in advance helps to understand the source of the problem and with the help of this it is possible to repair or change the route to bypass these areas.

the result

Finally, we got that the gripper was moving out of position A, which started in position B, accelerates to follow the barcode at a relative speed of zero for 0.6 seconds at 25 cm from it in a suitable orientation and slows down to a stop in position C. After that, we printed the movement of the robot relative to the floor (in place relative to the moving base) and checked that the robot's task is indeed carried out In the system described in the project, this while maintaining the limits on the values and speeds of the joints, speed and acceleration of the gripper and checking the torques in its arm required to perform this task.

Numerical and graphical results are described later in the report.

Manufacturer's Specifications

The robot:

FANUC M-10iA is a series of robots that are built for high speeds and precision. The company FANUC is an international Japanese corporation specializing in robotics, computer numerical control systems and factory automation.

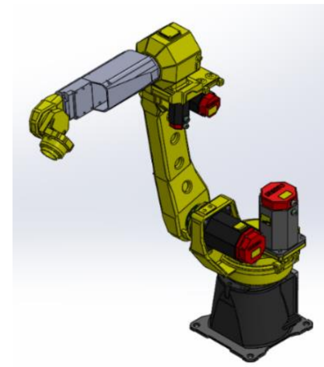


Figure 3: The CAD model of the robot

We have chosen this robot from the specification: M-10iA/12.

We chose this robot considering that the target we want to reach is 650 mm away from the robot and for such a distance it is sufficient to use the smallest model among the three in the specification. Main features:

- Cargo capacity (maximum weight a robot can carry):$12[kg]$
- A workspace that reaches:$1420[mm]$
- Duke, the robot has the apartment error-Repeatability (returning consistently to a certain position):$\pm 0.03[mm]$
- A relatively lightweight robot that is easy to assemble in existing production lines.

$$\begin{aligned} \theta_1 &= \pm 170° \\ \theta_2 &= \pm 125° \\ \theta_3 &= \pm 222.5° \\ \theta_4 &= \pm 190° \\ \theta_5 &= \pm 190° \\ \theta_6 &= \pm 360° \end{aligned}$$

In addition, we must refer to the limitations of the joints.

When it is possible, with an expansion option, to expand the rotation of θ_1 to $360°$.

We see that for this project, these limitations are not of big concern (for only one scan, there is no need for rotations beyond $360°$) We will define the zero angles by DH, with 90 degrees added to the angle of the second joint to match the zero configuration of the robot in the specification.

In addition, the specification also shows limitations on the speed of the joints (maximum joint speed).

To calculate the direct and inverse kinematics of the robot, we draw an abstract sketch for the robot that includes lines, cylinders and a floor. This is the skeleton of the robot:

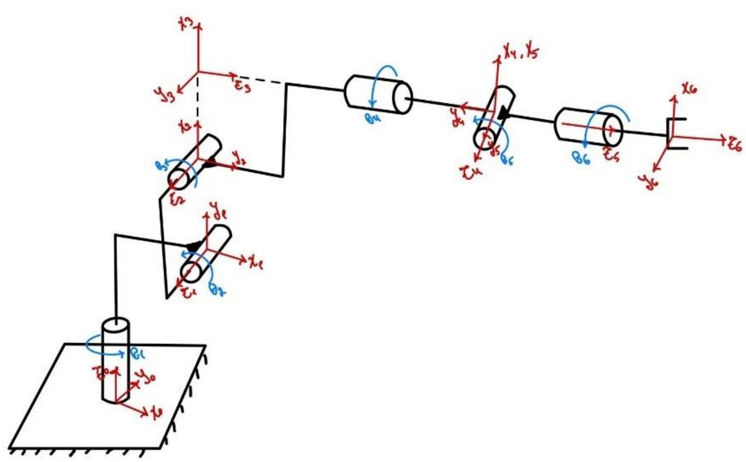


illustration4: The robot skeleton

The Forward Kinematics Solution

We will sketch the robot in the form of a skeleton with axle systems according to DH:

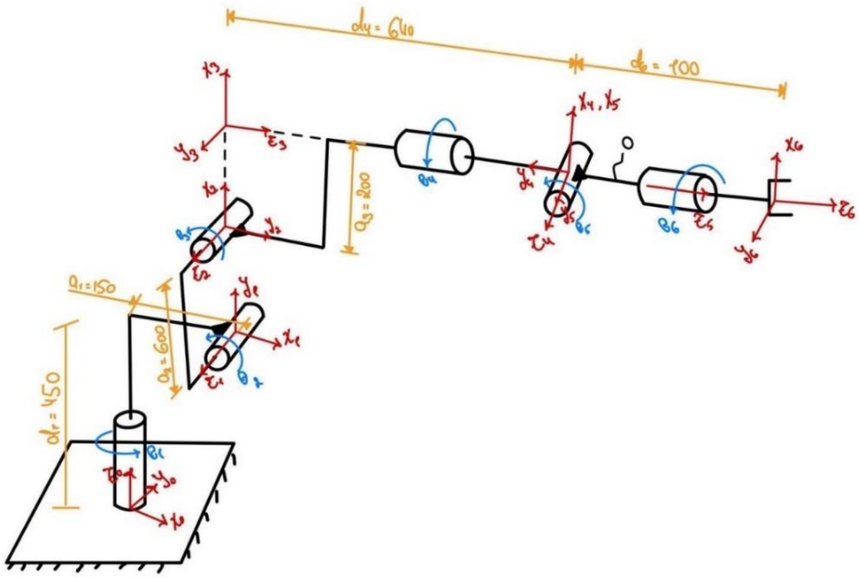


illustration5: The robot skeleton with DH parameters

- The distance between joints 5 and 6 is 0 (spherical joint)

Now we will fill a table of DH parameters:

parameter\ Joint number	d_i	a_i	θ_i	α_i
1	$d_1 = 450$	$a_1 = 150$	θ_1	$+90°$
2	$d_2 = 0$	$a_2 = 600$	$\boldsymbol{\theta_2 + 90°}$	$0°$
3	0	$a_3 = 200$	θ_3	$+90°$
4	$d_4 = 640$	0	θ_4	$-90°$
5	0	0	θ_5	$+90°$
6	$d_6 = 100$	0	θ_6	$0°$

The transformation between two consecutive systems in a serial robot–**translation matrix**:

$$ {}^{i-1}A_i = \begin{pmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

and the FK of the robot (direct kinematics mapping):

$$ \boxed{ {}^{0}A_6 = {}^{0}A_1 \dots {}^{4}A_5 \, {}^{5}A_6 \rightarrow \begin{pmatrix} \vec{x}_0 \\ 1 \end{pmatrix} = {}^{0}A_6 \begin{pmatrix} \vec{x}_6 \\ 1 \end{pmatrix} } $$

Calculate the explicit matrices in MATLAB.

In the end we get:

$$
{}^{0}A_6 = \begin{pmatrix} | & | & | & | \\ \hat{x}_6^{(0)} & \hat{y}_6^{(0)} & \hat{z}_6^{(0)} & \vec{d}_6 \\ | & | & | & | \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

Where:

$$
\vec{d}_6 = \begin{pmatrix} C_1(a_1 + a_2C_{\tilde{2}} + a_3C_{\tilde{2}3} + d_4S_{\tilde{2}3}) + d_6(S_5(S_1S_4 + C_4C_1C_{\tilde{2}3}) + C_1S_{\tilde{2}3}C_5) \\ S_1(a_1 + a_2C_{\tilde{2}} + a_3C_{\tilde{2}3} + d_4S_{\tilde{2}3}) - d_6\ (S_5(C_1S_4 - C_4S_1C_{\tilde{2}3}) - C_5S_1S_{\tilde{2}3}) \\ d_1 - d_4C_{\tilde{2}3} + a_2S_{\tilde{2}} - d_6(C_5C_{\tilde{2}3} - C_4S_5S_{\tilde{2}3}) + a_3S_{\tilde{2}3} \end{pmatrix}
$$

$$
\hat{x}_6^{(0)} = \begin{pmatrix} C_6(C_5(S_1S_4 + C_4C_1C_{\tilde{2}3}) - S_5C_1S_{\tilde{2}3}) + S_6(S_1C_4 - S_4C_1C_{\tilde{2}3}) \\ -C_6(C_5(C_1S_4 - C_4S_1C_{\tilde{2}3}) + S_5S_1S_{\tilde{2}3}) - S_6(C_1C_4 + S_4S_1C_{\tilde{2}3}) \\ C_6(S_5C_{\tilde{2}3} + C_4C_5S_{\tilde{2}3}) - S_4S_6S_{\tilde{2}3} \end{pmatrix}
$$

$$
\hat{y}_6^{(0)} = \begin{pmatrix} C_6(S_1C_4 - S_4C_1C_{\tilde{2}3}) - S_6(C_5(S_1S_4 + C_4C_1C_{\tilde{2}3}) - S_5C_1S_{\tilde{2}3}) \\ -C_6(C_1C_4 + S_4S_1C_{\tilde{2}3}) + S_6(C_5(C_1S_4 - C_4S_1C_{\tilde{2}3}) + S_5S_1S_{\tilde{2}3}) \\ S_6(S_5C_{\tilde{2}3} + C_4C_5S_{\tilde{2}3}) - C_6S_4S_{\tilde{2}3} \end{pmatrix}
$$

$$
\hat{z}_6^{(0)} = \begin{pmatrix} C_5C_1S_{\tilde{2}3} + S_5(S_1S_4 + C_1C_4C_{\tilde{2}3}) \\ C_5S_1S_{\tilde{2}3} - S_5(C_1S_4 - S_1C_4C_{\tilde{2}3}) \\ C_4S_5S_{\tilde{2}3} - C_5C_{\tilde{2}3} \end{pmatrix}
$$

Recall that here: $\tilde{\theta}_2 = \theta_2 + 90°$

The Inverse Kinematics Solution

given: Robot specification and the FK of the robot: ${}^{0}A_6\left(\underline{q}\right),\ \underline{q}=\begin{pmatrix} q_1 \\ \vdots \\ q_6 \end{pmatrix}$

And will get a desired position and orientation for the robot in relation to the base: $\vec{d}, R$

as $T=\begin{pmatrix} R & \vec{d} \\ 000 & 1 \end{pmatrix}$–The Task/Target Matrix

Need to Find: Values of the robot parameters, joint values $\underline{q}$, which will give:

$$ {}^{0}A_6\left(\underline{q}\right)=T \leftrightarrow \begin{cases} {}^{0}R_6\left(\underline{q}\right)=R \\ {}^{0}\vec{d}_6\left(\underline{q}\right)=\vec{d} \end{cases} $$

Solution Scheme for the IK of this robot

Note that the axes of the 3 joints $\theta_4, \theta_5, \theta_6$ intersect at a common point.

1. **Location problem:** First we solve for the point $\vec{P}_c=\vec{P}_c(\theta_1,\theta_2,\theta_3)$:

$$ \vec{P}_c=\vec{d}-d_6\hat{z}_6=\vec{d}-d_6R\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} $$

From here you get solutions for $\theta_1, \theta_2, \theta_3$ that give the desired target location.

2. **Orientation problem:** Now we will use the requirement on the rotation of the gripper and the known angles to determine the other angles:

$$ {}^{0}R_6\left(\underline{q}\right)={}^{0}R_3(\theta_1,\theta_2,\theta_3)\,{}^{3}R_6(\theta_4,\theta_5,\theta_6) \overset{\text{נדרוש}}{\triangleq} R $$

$$ \rightarrow {}^{3}R_6(\theta_4,\theta_5,\theta_6)={}^{0}R_3^{-1}R={}^{0}R_3^{T}R $$

From here you solve algebraically for $\theta_4, \theta_5, \theta_6$.

1 - Location problem

$\vec{P}_c$known. Now we will solve this problem geometrically.

Let's start by finding the position of the point according to the first 3 joints:

To check the results of the code, we will also find this location manually.

$$\vec{d}_{0\to4} = d_1\hat{z}_0 + a_1\hat{x}_1 + a_2\hat{x}_2 + a_3\hat{x}_3 + d_4\hat{z}_3$$

Where (notice that for $\theta_2 = 0$ here $\hat{x}_1 = \hat{x}_2$ without the addition of 90° in the FK):

$$\hat{x}_1 = C_1\hat{x}_0 + S_1\hat{y}_0 = \begin{pmatrix} C_1 \\ S_1 \\ 0 \end{pmatrix}, \hat{x}_2 = C_2\hat{x}_1 + S_2\hat{y}_1 = C_2\begin{pmatrix} C_1 \\ S_1 \\ 0 \end{pmatrix} + S_2\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} C_1C_2 \\ S_1C_2 \\ S_2 \end{pmatrix}$$

$$\hat{z}_1 = \begin{pmatrix} S_1 \\ -C_1 \\ 0 \end{pmatrix}$$

$$\hat{x}_3 = C_3\hat{x}_2 + S_3\hat{y}_2 = \begin{pmatrix} C_1C_2C_3 \\ S_1C_2C_3 \\ S_2C_3 \end{pmatrix} + S_3(-S_2\hat{x}_1 + C_2\hat{y}_1) = \begin{pmatrix} C_1C_2C_3 \\ S_1C_2C_3 \\ S_2C_3 \end{pmatrix} + S_3\left(-S_2\begin{pmatrix} C_1 \\ S_1 \\ 0 \end{pmatrix} + C_2\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}\right)$$

$$\hat{x}_3 = \begin{pmatrix} C_1C_2C_3 - C_1S_2S_3 \\ S_1C_2C_3 - S_1S_2S_3 \\ S_2C_3 + C_2S_3 \end{pmatrix}$$

$$\hat{z}_3 = S_3\hat{x}_2 - C_3\hat{y}_2 = S_3\begin{pmatrix} C_1C_2 \\ S_1C_2 \\ S_2 \end{pmatrix} - C_3\left(-S_2\begin{pmatrix} C_1 \\ S_1 \\ 0 \end{pmatrix} + C_2\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}\right) = \begin{pmatrix} C_1C_2S_3 + C_1S_2C_3 \\ S_1C_2S_3 + S_1S_2C_3 \\ S_2S_3 - C_2C_3 \end{pmatrix}$$

So: $\vec{d}_{0\to4} = d1\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + a_1\begin{pmatrix} C_1 \\ S_1 \\ 0 \end{pmatrix} + a_2\begin{pmatrix} C_1C_2 \\ S_1C_2 \\ S_2 \end{pmatrix} + a_3\begin{pmatrix} C_1C_2C_3 - C_1S_2S_3 \\ S_1C_2C_3 - S_1S_2S_3 \\ S_2C_3 + C_2S_3 \end{pmatrix} + d_4\begin{pmatrix} C_1C_2S_3 + C_1S_2C_3 \\ S_1C_2S_3 + S_1S_2C_3 \\ S_2S_3 - C_2C_3 \end{pmatrix}$

We will get: $\vec{d}_{0\to4} = \begin{pmatrix} a_1C_1 + a_2C_1C_2 + a_3C_1(C_2C_3 - S_2S_3) + d_4C_1(C_2S_3 + S_2C_3) \\ a_1S_1 + a_2S_1C_2 + a_3S_1(C_2C_3 - S_2S_3) + d_4S_1(C_2S_3 + S_2C_3) \\ d_1 + a_2S_2 + a_3(S_2C_3 + C_2S_3) + d_4(S_2S_3 - C_2C_3) \end{pmatrix}$

We note that: $C_2C_3 - S_2S_3 = C_{23},\ C_2S_3 + S_2C_3 = S_{23},\ S_2S_3 - C_2C_3 = -C_{23}$

Hence: $\vec{d}_{0\to4} = \begin{pmatrix} a_1C_1 + a_2C_1C_2 + a_3C_1C_{23} + d_4C_1S_{23} \\ a_1S_1 + a_2S_1C_2 + a_3S_1C_{23} + d_4S_1S_{23} \\ d_1 + a_2S_2 + a_3S_{23} - d_4C_{23} \end{pmatrix} \overset{Require}{\triangleq} \vec{P}_c = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}$

In conclusion, the equations for the location:

$$\begin{cases} C_1(a_1 + a_2C_2 + a_3C_{23} + d_4S_{23}) = x_0 \\ S_1(a_1 + a_2C_2 + a_3C_{23} + d_4S_{23}) = y_0 \\ d_1 + (a_2S_2 + a_3S_{23} - d_4C_{23}) = z_0 \end{cases}$$

<u>Now find the angle θ_1:</u>

Find the horizontal distance between the axis $\hat{z}_0$ and the desired point:

We will mark:

$$a_1 + a_2C_2 + a_3C_{23} + d_4S_{23} \equiv K$$

$$x_0^2 + y_0^2 = (C_1^2 + S_1^2)K^2 = K^2 = f(\theta_2, \theta_3)$$

so:

$$\boxed{K = \pm\sqrt{x_0^2 + y_0^2}}$$

Now the equations:

$$\begin{cases} C_1K = x_0 \\ S_1K = y_0 \\ d_1 + (a_2S_2 + a_3S_{23} - d_4C_{23}) = z_0 \end{cases}$$

so:

<u>If $K \neq 0$:</u>
From equations 1,2:

$$C_1 = \frac{x_0}{K}, S_1 = \frac{y_0}{K} \rightarrow \theta_1 = Atan2(S_1, C_1)$$

<u>If $K = 0 (x_0 = y_0 = 0)$:</u>
So θ_1 is free.

Here the solutions are a right or left arm (from the base of the robot, the arm faces straight to the target point, or the robot faces the other way around and the arm turns towards the point).

For a simple and logical solution for our robot, we chose $K = +\sqrt{x_0^2 + y_0^2}$ in the project.

<u>Now we will solve for joint angle 3:</u>

When it is known, it is enough to focus on the part of the robot from the first joint to the fourth joint, because now the position of joint 1 is known and the MZ 1 is known.θ_1

We will additionally use the fact that the distance between joint 1 and the point $\vec{P}_c$ is now depending only on θ_3.

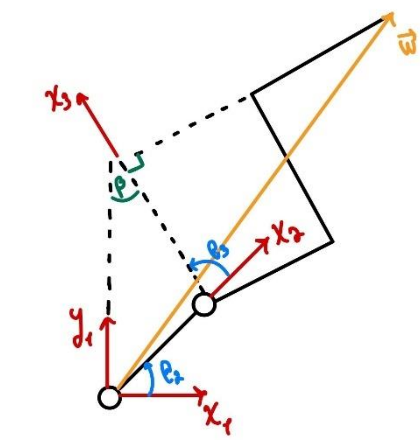


illustration6: Side view of the joints and links of the robot from the second joint to the point $\vec{P}_c$

$\vec{P}_c$ known. We will express $\vec{w}$ in the 1st axis-system:

$$\vec{w} = \vec{r}_{1\to4} = -d_1\hat{z}_0 - a_1\hat{x}_1 + \vec{P}_c = known$$

$$\vec{w}^{(1)} = -d_1\hat{y}_1 - a_1\hat{x}_1 + {}^{1}R_0\vec{P}_c = -d_1\hat{z}_1 - a_1\hat{x}_1 + {}^{0}R_1^T\vec{P}_c$$

Now we will focus on the subproblem from which we will get θ_3:

$$c^2 = a_2^2 + a_3^2 - 2a_2a_3C(\pi-\theta_3) \to c = +\sqrt{a_2^2 + a_3^2 - 2a_2a_3C(\pi-\theta_3)}$$

$$\frac{a_2}{S(\beta)} = \frac{c}{S(\pi-\theta_3)} \to S(\beta) = \frac{a_2}{c}S(\pi-\theta_3)$$

$$w^2 = c^2 + d_4^2 - 2cd_4C\left(\frac{\pi}{2}+\beta\right)$$

$$w^2 = a_2^2 + a_3^2 - 2a_2a_3C(\pi-\theta_3) + d_4^2 - 2cd_4\big(-S(\beta)\big)$$

$$w^2 - a_2^2 - a_3^2 - d_4^2 = 2a_2a_3C_3 + 2cd_4\frac{a_2}{c}S(\pi-\theta_3)$$

$$w^2 - a_2^2 - a_3^2 - d_4^2 = 2a_2a_3C_3 + 2a_2d_4S_3$$

This is an equation with a known solution:

$$AC_3 + BS_3 = C \to \theta_3 = Atan2(B,A) + Atan2\left(\pm\sqrt{A^2+B^2-C^2}, C\right)$$

where:

$$A = 2a_2a_3, B = 2a_2d_4, C = w^2 - a_2^2 - a_3^2 - d_4^2$$

There are 2 solutions for this angle (upper and lower joint).

<u>Now we will solve for joint angle 2:</u>

$$C_1(a_1 + a_2C_2 + a_3C_{23} + d_4S_{23}) = x_0 + (3) \to \begin{cases} a_1 + a_2C_2 + a_3C_{23} + d_4S_{23} = \dfrac{x_0}{C_1} \\ d_1 + (a_2S_2 + a_3S_{23} - d_4C_{23}) = z_0 \end{cases}$$

We note that:

$$C_2C_3 - S_2S_3 = C_{23}, \qquad C_2S_3 + S_2C_3 = S_{23}, \qquad S_2S_3 - C_2C_3 = -C_{23}$$

And from here we get linear equations for the solution of θ_2:

$$\begin{cases} a_1 + a_2C_2 + a_3(C_2C_3 - S_2S_3) + d_4(C_2S_3 + S_2C_3) = \frac{x_0}{C_1} \\ d_1 + a_2S_2 + a_3(C_2S_3 + S_2C_3) - d_4(C_2C_3 - S_2S_3) = z_0 \end{cases}, \begin{cases} C_2(a_2 + a_3C_3 + d_4S_3) + S_2(d_4C_3 - a_3S_3) = \frac{x_0}{C_1} - a_1 \\ C_2(a_3S_3 - d_4C_3) + S_2(a_2 + a_3C_3 + d_4S_3) = z_0 - d_1 \end{cases}$$

For the case (shown to be singular later) where $K = 0$:

$$\begin{cases} a_1 + a_2C_2 + a_3C_{23} + d_4S_{23} = 0 \\ d_1 + (a_2S_2 + a_3S_{23} - d_4C_{23}) = z_0 \end{cases} \rightarrow \begin{cases} C_2(a_2 + a_3C_3 + d_4S_3) + S_2(d_4C_3 - a_3S_3) = -a_1 \\ C_2(a_3S_3 - d_4C_3) + S_2(a_2 + a_3C_3 + d_4S_3) = z_0 - d_1 \end{cases}$$

We will solve this system of equations. But, in case $C_1 = 0$:

$$\begin{cases} S_1(a_1 + a_2C_2 + a_3C_{23} + d_4S_{23}) = y_0 \\ d_1 + (a_2S_2 + a_3S_{23} - d_4C_{23}) = z_0 \end{cases} \rightarrow \begin{cases} a_2C_2 + a_3(C_2C_3 - S_2S_3) + d_4(C_2S_3 + S_2C_3) = \frac{y_0}{S_1} - a_1 \\ C_2(a_3S_3 - d_4C_3) + S_2(a_2 + a_3C_3 + d_4S_3) = z_0 - d_1 \end{cases}$$

$$\rightarrow \begin{cases} C_2(a_2 + a_3C_3 + d_4S_3) + S_2(-a_3S_3 + d_4C_3) = \frac{y_0}{S_1} - a_1 \\ C_2(a_3S_3 - d_4C_3) + S_2(a_2 + a_3C_3 + d_4S_3) = z_0 - d_1 \end{cases}$$

We will solve this system of equations.

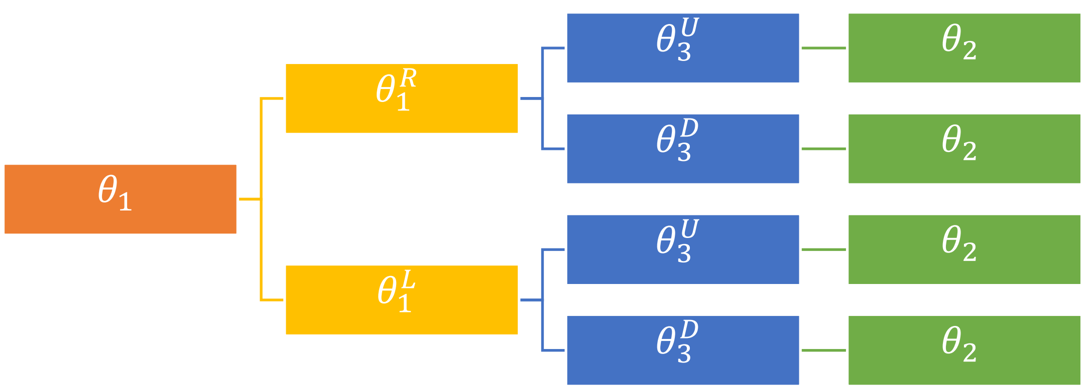
- It was possible to solve for K in general (using $a_1 + a_2C_2 + a_3C_{23} + d_4S_{23} = K$) and get a single solution for θ_2 for every situation.

From here you can solve a system of linear equations and get:

$$\rightarrow \theta_2 = Atan2(S_2, C_2)$$

This is a single solution for this angle.

<u>**Hence, the solution tree for the location problem:**</u>



- For the case where the required point on a vertical axis of the base joint ($K = 0$) there are identical solutions for the angles except for infinite solutions for the angle of the base joint.

2 - the orientation problem

$${}^{0}R_6\left(\underline{q}\right) = {}^{0}R_3(\theta_1,\theta_2,\theta_3)\,{}^{3}R_6(\theta_4,\theta_5,\theta_6) \overset{\text{נדרוש}}{\triangleq} R$$

$$\rightarrow {}^{3}R_6(\theta_4,\theta_5,\theta_6) = {}^{0}R_3^{-1}R = {}^{0}R_3^{T}R$$

From here you must solve for $\theta_4,\theta_5,\theta_6$ algebraically.

<u>We will find ${}^{3}R_6$:</u>

We will use the properties of the homogeneous matrix:

$$\vec{x}_{i-1} = {}^{i-1}R_i\vec{x}_i + {}^{i-1}\vec{d}_i \rightarrow \begin{pmatrix}\vec{x}_{i-1}\\1\end{pmatrix} = {}^{i-1}A_i\begin{pmatrix}\vec{x}_i\\1\end{pmatrix}$$

$$\vec{x}_{i-2} = {}^{i-2}R_{i-1}\vec{x}_{i-1} + {}^{i-2}\vec{d}_{i-1} \rightarrow \begin{pmatrix}\vec{x}_{i-2}\\1\end{pmatrix} = {}^{i-2}A_{i-1}\begin{pmatrix}\vec{x}_{i-1}\\1\end{pmatrix} = {}^{i-2}A_{i-1}\,{}^{i-1}A_i\begin{pmatrix}\vec{x}_i\\1\end{pmatrix}$$

$$so{:}\begin{pmatrix}\vec{x}_3\\1\end{pmatrix} = {}^{3}A_4\,{}^{4}A_5\,{}^{5}A_6\begin{pmatrix}\vec{x}_6\\1\end{pmatrix} \rightarrow \boxed{{}^{3}A_6 = {}^{3}A_4\,{}^{4}A_5\,{}^{5}A_6 \rightarrow {}^{3}R_6 = {}^{3}A_6(1{:}3,1{:}3)}$$

According to MATLA: $${}^{3}R_6(\theta_4,\theta_5,\theta_6) = \begin{pmatrix} C_4C_5C_6 - S_4S_6 & -C_6S_4 - C_4C_5S_6 & C_4S_5 \\ C_4S_6 + C_5C_6S_4 & C_4C_6 - C_5S_4S_6 & S_4S_5 \\ -C_6S_5 & S_5S_6 & C_5 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & E & f \\ g & h & I \end{pmatrix}$$

<u>From here we get a solution for θ_5:</u>

$$C_5 = I \rightarrow S_5 = \pm\sqrt{1-I^2} \rightarrow \theta_5 = Atan2(S_5,C_5)$$

This angle has 2 options.

Note that if: $I = C_5 = \pm 1 \rightarrow S_5 = 0$, in this case:

$${}^{3}R_6(\theta_4,\theta_5,\theta_6) = \begin{pmatrix} C_4C_5C_6 - S_4S_6 & -C_6S_4 - C_4C_5S_6 & 0 \\ C_4S_6 + C_5C_6S_4 & C_4C_6 - C_5S_4S_6 & 0 \\ 0 & 0 & C_5 \end{pmatrix}$$

<u>Solution for θ_4,θ_6:</u>

Assuming that $I = C_5 \neq \pm 1$:

$$\boxed{C_4 = \frac{c}{S_5}, S_4 = \frac{f}{S_5}, C_6 = -\frac{g}{S_5}, S_6 = \frac{h}{S_5} \rightarrow \theta_4 = Atan2(S_4,C_4), \theta_6 = Atan2(S_6,C_6)}$$

If $I = C_5 = \pm 1$:

$$\begin{pmatrix} C_4C_5C_6 - S_4S_6 & -C_6S_4 - C_4C_5S_6 & 0 \\ C_4S_6 + C_5C_6S_4 & C_4C_6 - C_5S_4S_6 & 0 \\ 0 & 0 & C_5 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & E & f \\ g & h & I \end{pmatrix}$$

For a case in which $\theta_5 = 0$, we get $I = C_5 = +1$ and hence:

$$\begin{pmatrix} C_4C_6 - S_4S_6 & -C_6S_4 - C_4S_6 & 0 \\ C_4S_6 + C_6S_4 & C_4C_6 - S_4S_6 & 0 \\ 0 & 0 & C_5 \end{pmatrix} = \begin{pmatrix} C_{46} & -S_{46} & 0 \\ S_{46} & C_{46} & 0 \\ 0 & 0 & C_5 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & E & f \\ g & h & I \end{pmatrix}$$

and we get:

$$\theta_4 + \theta_6 = Atan2(S_{46},C_{46})$$

And in fact, infinite solutions are obtained!

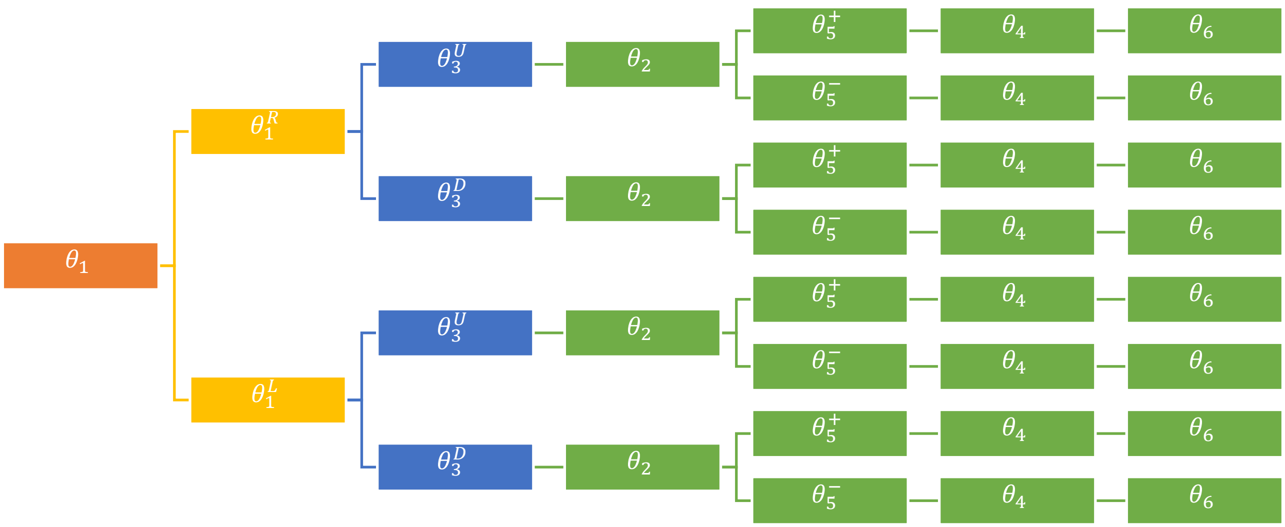
We will check that the second case is possible:

$$I = \hat{z}_3 \cdot \hat{z}_6 = C_5$$

This case even happens when the robot is in zero mode in the specification.

Hence, the solution tree for the total solution:

Therefore, there are a total of 8 solutions for the inverse kinematics of this robot.



- Not including singular cases in which have infinite solutions

For Trajectory Planning, we need a solution-selection criteria

In the choice of $K = \pm\sqrt{x_0^2 + y_0^2}$, we choose $K = +\sqrt{x_0^2 + y_0^2}$ because we want the robot to reach the target point in front of it and not turn around and in the opposite direction from the point and then turn in its direction. So, choose $\theta_1 = \theta_1^R$.

In the selection $\theta_3 = Atan2(B, A) + Atan2\left(\pm\sqrt{A^2 + B^2 - C^2}, C\right)$, a solution of an upper and a lower joint is obtained for the position of axis-system 4. We would like a solution of an upper joint, and according to MATLAB this happens for $-\sqrt{A^2 + B^2 - C^2}$.

In the choice $S_5 = \pm\sqrt{1 - I^2}$, $S_5 = -\sqrt{1 - I^2}$ is chosen. There is no difference between the solutions except for the values of joints 4,5,6.

Finally, a unique solution for the project is obtained. It is important to focus on planning the trajectory in a single solution branch, in choosing a solution from the collection of discrete solutions of the robot for the inverse kinematics, because between them there are discontinuous jumps in position, and we want a "smooth" position in time of the gripper for the purpose of planning the trajectory.

Obtaining the Full Jacobian of the Robot

The Full Jacobian: $J = \begin{pmatrix} J_{L_1} & J_{L_2} & J_{L_3} & J_{L_4} & J_{L_5} & J_{L_6} \\ J_{A_1} & J_{A_2} & J_{A_3} & J_{A_4} & J_{A_5} & J_{A_6} \end{pmatrix}$

Where, for a rotation joint $q_i = \theta_i$: $\begin{pmatrix} J_{L_i} \\ J_{A_i} \end{pmatrix} = \begin{pmatrix} \hat{b}_{i-1} \times \underline{r}_{i-1} \\ \hat{b}_{i-1} \end{pmatrix}$

And the vectors are:

$\hat{b}_i = \hat{b}_i^{(0)}$ –axis $\hat{z}_i$ of i-th system (in the base system)

$\underline{r}_{i-1} = \underline{r}_{i-1,e}^{(0)} = \underline{r}_{i-1}^{(0)}$ –A vector from the beginning of joint i (System $i-1$) to the beginning of the gripper system (in the base system)

The Jacobian from the code:

$$\begin{pmatrix} L_1 \\ J_{A_1} \end{pmatrix} = \begin{pmatrix} -S_1(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23}) + d_6(S_5(C_1S_4 + S_1C_4S_{23}) - S_1C_5C_{23}) \\ C_1(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23}) + d_6(S_5(S_1S_4 - C_1C_4S_{23}) - C_1C_5C_{23}) \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} J_{L_2} \\ J_{A_2} \end{pmatrix} = \begin{pmatrix} -C_1\left(a_2C_2 + a_3C_{23} + d_4S_{23} + d_6(C_4S_5C_{23} + C_5S_{23})\right) \\ -S_1\left(a_2C_2 + a_3C_{23} + d_4S_{23} + d_6(C_4S_5C_{23} + C_5S_{23})\right) \\ -a_2S_2 - a_3S_{23} + d_4C_{23} + d_6(C_5C_{23} - C_4S_5S_{23}) \\ S_1 \\ -C_1 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} J_{L_3} \\ J_{A_3} \end{pmatrix} = \begin{pmatrix} -C_1\left(a_3C_{23} + d_4S_{23} + d_6(C_4S_5C_{23} + C_5S_{23})\right) \\ -S_1\left(a_3C_{23} + d_4S_{23} + d_6(C_4S_5C_{23} + C_5S_{23})\right) \\ -a_3S_{23} + d_4C_{23} + d_6(-C_4S_5S_{23} + C_5C_{23}) \\ S_1 \\ -C_1 \\ 0 \end{pmatrix}, \begin{pmatrix} J_{L_4} \\ J_{A_4} \end{pmatrix} = \begin{pmatrix} d_6S_5(S_1C_4 + C_1S_4S_{23}) \\ d_6S_5(-C_1C_4 + S_1S_4S_{23}) \\ -d_6C_{23}S_4S_5 \\ C_1C_{23} \\ S_1C_{23} \\ S_{23} \end{pmatrix}$$

$$\begin{pmatrix} J_{L_5} \\ J_{A_5} \end{pmatrix} = \begin{pmatrix} d_6(C_5(S_1S_4 - C_1C_4S_{23}) - C_1S_5C_{23}) \\ d_6(-C_5(C_1S_4 + S_1C_4S_{23}) - S_1S_5C_{23}) \\ d_6(C_4C_5C_{23} - S_5S_{23}) \\ C_4S_1 + C_1S_4S_{23} \\ S_1S_{23}S_4 - C_1C_4 \\ -C_{23}S_4 \end{pmatrix}, \begin{pmatrix} J_{L_6} \\ J_{A_6} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -(C_1S_{23}C_4 - S_1S_4)S_5 + C_1C_{23}C_5 \\ -(S_1S_{23}C_4 + C_1S_4)S_5 + S_1C_{23}C_5 \\ C_{23}C_4S_5 + S_{23}C_5 \end{pmatrix}$$

Here θ_2 is the angle of the second joint without the 90° ($\tilde{\theta}_2 = \theta_2 + 90°$).

Finding Singularities of the Linear Jacobian

Theory and solution method

We need find joint values for which the robot is in a singularity of its linear speed and angular speed.

We are focusing in finding situations in which the linear speed that the robot can have at the gripper point is singular:

$$\underline{V} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = J_{L_{3x6}}\underline{\dot{q}} = J_{L_1}\dot{\theta}_1 + J_{L_2}\dot{\theta}_2 + J_{L_3}\dot{\theta}_3 + J_{L_4}\dot{\theta}_4 + J_{L_5}\dot{\theta}_5 + J_{L_6}\dot{\theta}_6$$

Singularity will occur when the columns will be linearly dependent, which is equivalent to demanding a drop in degree from the full rank of the linear Jacobian matrix.

The rank of the matrix satisfies: $r = r(J_L) \leq \min(3,6) = 3$. We will check for which situations $r < 3$. In the case J_L is not square: $r \downarrow \Leftrightarrow det(J_L J_L^T) = 0$

Equivalently, when there is a singularity in the linear velocity or the angular velocity, the rank of the full Jacobian matrix will decrease and for a square matrix it is equivalent to the situation where the determinant is zero:

$$J = \begin{pmatrix} JL_{1,1} & JL_{2,1} & JL_{3,1} & JL_{4,1} & JL_{5,1} & 0 \\ JL_{1,2} & JL_{2,2} & JL_{3,2} & JL_{4,2} & JL_{5,2} & 0 \\ 0 & JL_{2,3} & JL_{3,3} & JL_{4,3} & JL_{5,3} & 0 \\ 0 & S_1 & S_1 & JA_{4,1} & JA_{5,1} & JA_{6,1} \\ 0 & -C_1 & -C_1 & JA_{4,2} & JA_{5,2} & JA_{6,2} \\ 1 & 0 & 0 & JA_{4,3} & JA_{5,3} & JA_{6,3} \end{pmatrix}$$

According to an article about finding a singularity of an industrial 6R robot (link in the appendix), this is true: $|J| = |J_w|$

where, the singularity problem is equivalent to the singularity solution for the spherical joint:

$$J_w = J_{d_6=0} = \begin{pmatrix} JL_{1,1} & JL_{2,1} & JL_{3,1} & 0 & 0 & 0 \\ JL_{1,2} & JL_{2,2} & JL_{3,2} & 0 & 0 & 0 \\ 0 & JL_{2,3} & JL_{3,3} & 0 & 0 & 0 \\ 0 & S_1 & S_1 & JA_{4,1} & JA_{5,1} & JA_{6,1} \\ 0 & -C_1 & -C_1 & JA_{4,2} & JA_{5,2} & JA_{6,2} \\ 1 & 0 & 0 & JA_{4,3} & JA_{5,3} & JA_{6,3} \end{pmatrix} = \begin{pmatrix} A & 0_{3x3} \\ B & C \end{pmatrix}$$

then:

$$|J| = |A||C| \overset{require}{\cong} 0$$

Therefore, we will get a singularity when:

$$|A| = \begin{vmatrix} -S_1(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23}) & -C_1(a_2C_2 + a_3C_{23} + d_4S_{23}) & -C_1(a_3C_{23} + d_4S_{23}) \\ C_1(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23}) & -S_1(a_2C_2 + a_3C_{23} + d_4S_{23}) & -S_1(a_3C_{23} + d_4S_{23}) \\ 0 & -a_2S_2 - a_3S_{23} + d_4C_{23} & -a_3S_{23} + d_4C_{23} \end{vmatrix} = 0$$

$$\begin{aligned} = -S_1(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big(&-S_1(a_2C_2 + a_3C_{23} + d_4S_{23})(-a_3S_{23} + d_4C_{23}) \\ &+ S_1(a_3C_{23} + d_4S_{23})(-a_2S_2 - a_3S_{23} + d_4C_{23})\big) \\ &- C_1(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big(-C_1(a_2C_2 + a_3C_{23} + d_4S_{23})(-a_3S_{23} + d_4C_{23}) \\ &+ C_1(a_3C_{23} + d_4S_{23})(-a_2S_2 - a_3S_{23} + d_4C_{23})\big) \end{aligned}$$

$$\begin{aligned} = S_1^2(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big(&(a_2C_2 + a_3C_{23} + d_4S_{23})(-a_3S_{23} + d_4C_{23}) \\ &- (a_3C_{23} + d_4S_{23})(-a_2S_2 - a_3S_{23} + d_4C_{23})\big) \\ &+ C_1^2(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big((a_2C_2 + a_3C_{23} + d_4S_{23})(-a_3S_{23} + d_4C_{23}) \\ &- (a_3C_{23} + d_4S_{23})(-a_2S_2 - a_3S_{23} + d_4C_{23})\big) \end{aligned}$$

$$\begin{aligned} = (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big(&(a_2C_2 + a_3C_{23} + d_4S_{23})(-a_3S_{23} + d_4C_{23}) \\ &- (a_3C_{23} + d_4S_{23})(-a_2S_2 - a_3S_{23} + d_4C_{23})\big) \end{aligned}$$

$$\begin{aligned} = (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big(&a_2C_2(-a_3S_{23} + d_4C_{23}) \\ &+ (a_3C_{23} + d_4S_{23})(-a_3S_{23} + d_4C_{23} + a_2S_2 + a_3S_{23} - d_4C_{23})\big) \end{aligned}$$

$$= (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})\big(a_2C_2(-a_3S_{23} + d_4C_{23}) + (a_3C_{23} + d_4S_{23})(a_2S_2)\big)$$

$$= (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})a_2(-a_3C_2S_{23} + d_4C_2C_{23} + a_3S_2C_{23} + d_4S_2S_{23})$$

$$= (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})a_2(a_3C_2S_{-2-3} + d_4C_2C_{23} + a_3S_2C_{23} + d_4S_2S_{23})$$

$$= (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})a_2(a_3S_{-3} + d_4C_3)$$

so:

$$|A| = (a_1 - a_2S_2 - a_3S_{23} + d_4C_{23})a_2(d_4C_3 - a_3S_3)$$

The second matrix:

$$|C| = -S_5$$

And a singularity will be obtained for:

$$(a_1 - a_2S_2 - a_3S_{23} + d_4C_{23}) = 0$$
$$(d_4C_3 - a_3S_3) = 0$$
$$\theta_5 = 0/\pi$$

We will check with MATLAB the type of each singularity

We would like to find a singularity specifically of J_L and not the general of J , although all singularities of the large Jacobian include singularities in particular only of the linear velocity.

It seems that these modes are singular in the linear and angular velocity together for the most part and only in specific cases will there be a linear singularity only. An immediate case for linear singularity only is when the gripper reaches the outer edges of the robot's workspace (for example, when the arm is "fully stretched" in the second case we will talk about).

We will go over these three cases and their geometric meaning.

In finding the singularity, it is important to get the points in the space that we do not want the gripper's trajectory to pass through, due to the appearance of discontinuities in the solution of the joints or the requirement of high speeds from the robot's joints.

Recall that the position of the gripper is (according to direct kinematics):

$$\vec{d}_6 = \begin{pmatrix} C_1(a_1 + a_2C_{\tilde{2}} + a_3C_{\widetilde{23}} + d_4S_{\widetilde{23}}) + d_6(S_5(S_1S_4 + C_4C_1C_{\widetilde{23}}) + C_1S_{\widetilde{23}}C_5) \\ S_1(a_1 + a_2C_{\tilde{2}} + a_3C_{\widetilde{23}} + d_4S_{\widetilde{23}}) - d_6\,(S_5(C_1S_4 - C_4S_1C_{\widetilde{23}}) - C_5S_1S_{\widetilde{23}}) \\ d_1 - d_4C_{\widetilde{23}} + a_2S_{\tilde{2}} - d_6(C_5C_{\widetilde{23}} - C_4S_5S_{\widetilde{23}}) + a_3S_{\widetilde{23}} \end{pmatrix}$$

The Comparison between the Robotic Arm and a Human Arm

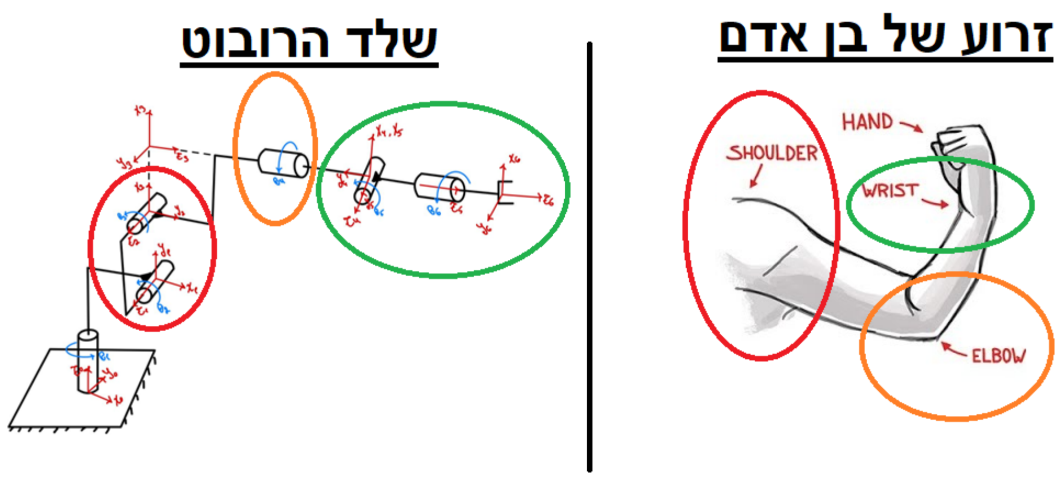


illustration7: The comparison between the joints in a human arm and those in the 6R robot

Anatomy of the robot:

The robotic arm is divided into a lower part (between joints 1 and 4), an upper part (between 4 and 5) and a hand (between 5 and the gripper). When the role of each joint:

- Chapter 1–The rotation of the base: forms the base of the robot and allows rotation of the entire arm around the vertical axisz0.
- Chapters 2 and 3–the shoulder (shoulder): constitute the shoulder of the robot and connect the elbow to the base. Colors move vertically. As we will see in the torque calculations and according to the skeleton, they are the main supports in applying torques against the weights in the arm and grip because their axis is always in the horizontal plane (they apply a "bending" torque).
- Chapter 4–elbow (elbow): Connects the upper and lower part of the arm, twists the arm.
- Chapters 5 and 6–Bending and twisting the wrist (wrist): connects the gripper itself to the upper part of the arm.

For a 6R Industrial Robot, there are 3 cases of singularity

First case - shoulder singularities

In this case (the angle from before with the deviation $\tilde{\theta}_2 = \theta_2 + 90°$):

$$(\mathrm{a}_1 - \mathrm{a}_2\mathrm{S}_2 - \mathrm{a}_3\mathrm{S}_{23} + \mathrm{d}_4\mathrm{C}_{23}) = (\mathrm{a}_1 + \mathrm{a}_2\mathrm{C}_{\tilde{2}} + \mathrm{a}_3\mathrm{C}_{\tilde{2}3} + \mathrm{d}_4\mathrm{S}_{\tilde{2}3}) = 0$$

Let's look at the location in this situation:

$$\vec{\mathrm{d}}_6 = \begin{pmatrix} \mathrm{d}_6(\mathrm{S}_5(\mathrm{S}_1\mathrm{S}_4 + \mathrm{C}_4\mathrm{C}_1\mathrm{C}_{\tilde{2}3}) + \mathrm{C}_1\mathrm{S}_{\tilde{2}3}\mathrm{C}_5) \\ -\mathrm{d}_6\ (\mathrm{S}_5(\mathrm{C}_1\mathrm{S}_4 - \mathrm{C}_4\mathrm{S}_1\mathrm{C}_{\tilde{2}3}) - \mathrm{C}_5\mathrm{S}_1\mathrm{S}_{\tilde{2}3}) \\ \mathrm{d}_1 - \mathrm{d}_4\mathrm{C}_{\tilde{2}3} + \mathrm{a}_2\mathrm{S}_{\tilde{2}} - \mathrm{d}_6(\mathrm{C}_5\mathrm{C}_{\tilde{2}3} - \mathrm{C}_4\mathrm{S}_5\mathrm{S}_{\tilde{2}3}) + \mathrm{a}_3\mathrm{S}_{\tilde{2}3} \end{pmatrix}$$

In this case, the center of the ball joint is above the vertical axis of the base joint.

If a trajectory passes through this singularity, joints 1 and 4 may require high speeds to rotate the robot off the z0-axis. Because the whole shoulder must rotate.

In the workspace, the singularity is in a cylinder with a radius d_6 around the z0-axis, up to the edges of the workspace. But only in solutions where the position of the gripper in this volume and its orientation maintains that the center of the ball joint is in the z0-axis.

Practically, the singularity is near the z0-axis, and in planning a trajectory we would like to stay away from there.

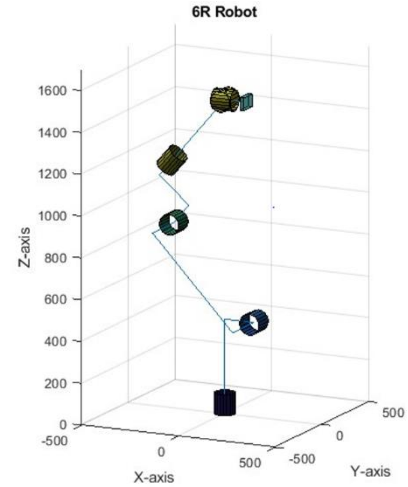


illustration8: Shoulder singularity configuration of the robot, when the gripper faces in the x1 direction

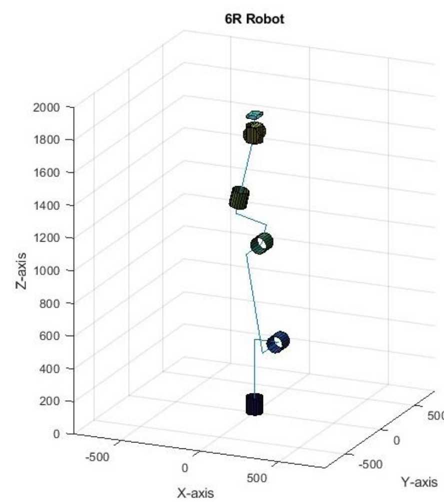


illustration9: Singularity configuration of the robot's shoulder, with the gripper facing up

In the case in Figure 8, a rank drop is obtained in J but not in J_L (or in J_A). according to the eigen vectors and values of J^T we see that in this configuration the robot losses directions of linear and angular velocity together. From this we will get that in the first case we will not necessarily get a singularity in the linear speed only. It is challenging to find all subcases where the robot will have a singularity that is entirely linear velocity only.

An example of a configuration in which linear velocity is singular is in Figure 9. This is a case where the gripper is directed upwards and rises as much as the robot can raise (approximately 1810 mm from the base of the robot, according to the specification 1720 mm), the gripper approaches a linear singularity.

Second case - elbow singularity (elbow singularities)

In this case:$(d_4 C_3 - a_3 S_3) = 0$

When that happens, we get $\vec{r}_{3\to5} \parallel \hat{x}_2$. That is, the axis between joints 2,3 merges with the vector between joint 3 and the spherical joint. This can happen when the arm is stretched or folded on itself, but only the first case is physically possible.

Meaning, in this case the robot looks as if it is "fully stretched" and the solution of upper and lower elbow are intersecting. In the workspace, this is all the farthest places the robot can reach, that is, the edges of the workspace.

For the case in Figure 10 in which the gripper is also on the same axis with the joints 2,3 and the spherical joint, a singularity of the overall Jacobian is obtained in the code and the determinant of the linear Jacobian is very low, which is dangerous and a singularity for the linear velocity in particular.

- By stretching the clamp farthest in any direction from the base, a completely linear singularity can be obtained.

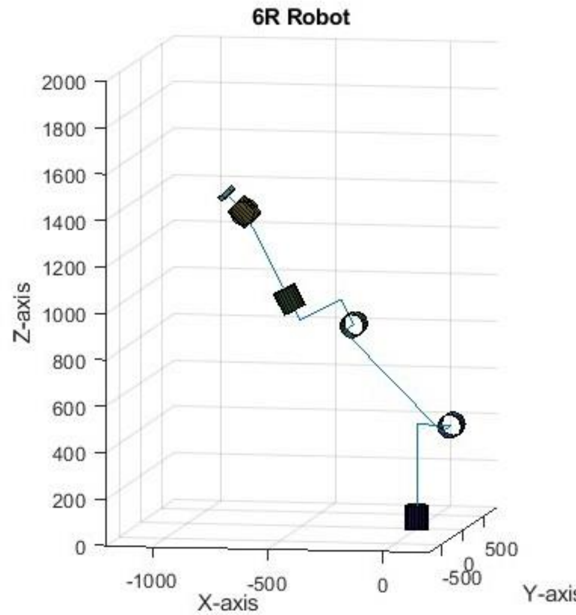


illustration10: Elbow singularity configuration of the robot, when the gripper faces in a direction parallel to joints 2,3

Third case - wrist singularity (wrist singularities)

In this case $\theta_5 = 0, \pi$: That is, the joint axes 4,6 intersect.

This happens already in the zero state of the robot in the specification. In this situation there is degeneration between joints 4,6 and they perform the same action in the solution to the orientation of the gripper and there are infinite solutions to IK.

In the case in Figure 11, there is a singularity in the total Jacobian but not in the linear one.

This singularity is the reason that at the beginning of the movement there are infinite solutions for joint angles 4,6 so that there is a particular initial orientation at the zero state, and it is necessary to choose the angles so that the functions of the joints are continuous when creating the movement path.

This case can happen within the robot's workspace.

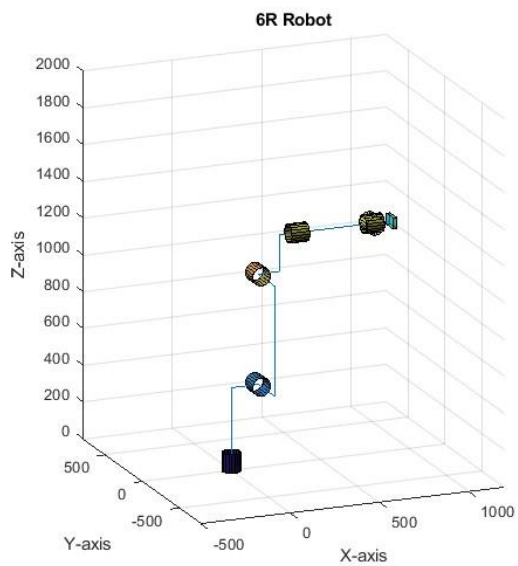


illustration11: Singularity configuration of the robot's wrist. It is in the zero-position described in the manufacturer's specification

Route planning

Optimum position for the robot on the rail and determining the height of the robot base

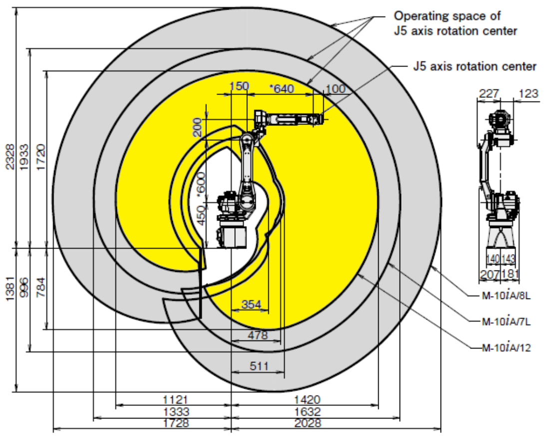


illustration12: The workspace from the robot specification, with a yellow emphasis on the workspace of the version chosen for the project

The workspace of the robot we chose is marked in yellow:

A base height is chosen, so that the height at which the barcode passes in front of the robot will be at the height where the workspace is the "widest".

According to the specifications, at a height of 450 mm from the base of the robot, the widest working space in the horizontal plane is therefore we would like the barcode to be at this height.

The height of the barcode from the floor is 800 mm (80 cm).

Therefore, a base with a height of 350 mm was chosen, so that the barcode would be at the desired height in relation to her arm.

Due to the proximity of the barcode to her arm, other base heights would also work.

Meaning, in relation to the base of the robot, the barcode is a point that moves along a height of 450 mm, at a fixed distance along the $\hat{y}_0$ axis and at a fixed speed in the $-\hat{x}_0$ direction.

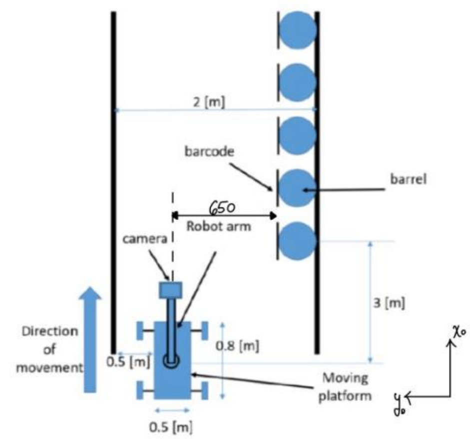


illustration13: The system of the problem with the distance between the barcode and the base of the robot and the system of axes we will work with

Firstly, we will use the known data to see for how long and in what positions the camera should be to scan the barcode:

According to the figure, for half a second the camera should scan the barcode at 20-30 cm from it, when the speed of the barcode relative to the base of the robot is 1 meter per second. Therefore, the distance the point will travel during this time:

$$\Delta x = \frac{\Delta t}{V} = 0.5\ [m] = 500[mm]$$

That is, at least for 500 mm, the robot should be in front of the barcode, perpendicular to it and at a certain distance from it.

It is chosen that the gripper will be at 25 cm from the barcode and will pass by it at a total distance of 1000 mm, where part of the distance will be because the gripper will accelerate to the speed of the barcode relative to the base of the robot, will reach its speed and will stop at the end.

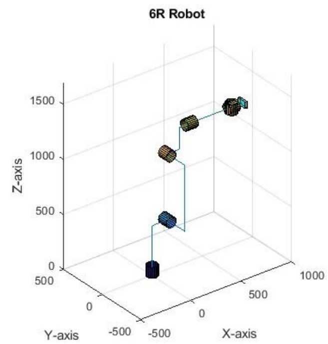
Now that we understand how much time and how much space is needed for tracking, we will plan a route according to the following steps:

We determine states we want the robot to pass through, and a route is built between them. There are two methods for constructing a trajectory from this:

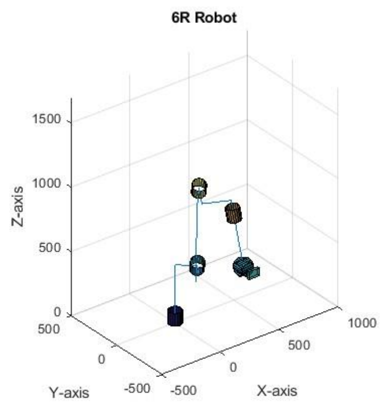
1. If we are not interested in the way in which the robot moves between the states, we will simply take the values of the joints in the two states between which we will want a continuous transition and build a polynomial-continuous transition in the joint-space.
2. In the part of the tracking, we will be interested in the form of the tracking in 3D space, therefore a specific geometric path is needed, we will make sure that the values of the first three joints give a desired position and that the values of the last three joints give a desired orientation. If we use this method, we will get a specific trajectory and path-based joint values.

It is chosen that the robot will be in the zero-position described in the robot specification, in the direction described in Figure 13 with joint angles 4,6 which are necessary at the beginning for the continuity of the robot's trajectory.

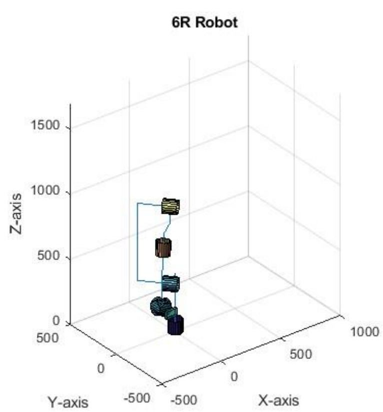
The main situations the robot will go through



$$T = \begin{pmatrix} 0 & 0 & 1 & 890 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 1250 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$ –situation A, the zero state of the robot



$$T = \begin{pmatrix} 0 & -1 & 0 & 500 \\ 0 & 0 & -1 & -400 \\ 1 & 0 & 0 & 450 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$ –situation B, in which the gripper is in wanted (y, z), and orientation



$$T = \begin{pmatrix} 0 & -1 & 0 & -500 \\ 0 & 0 & -1 & -400 \\ 1 & 0 & 0 & 450 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$ –situation C, same orientation and (y, z), after finishing the scanning

In total, the locations where the robot will move visually:

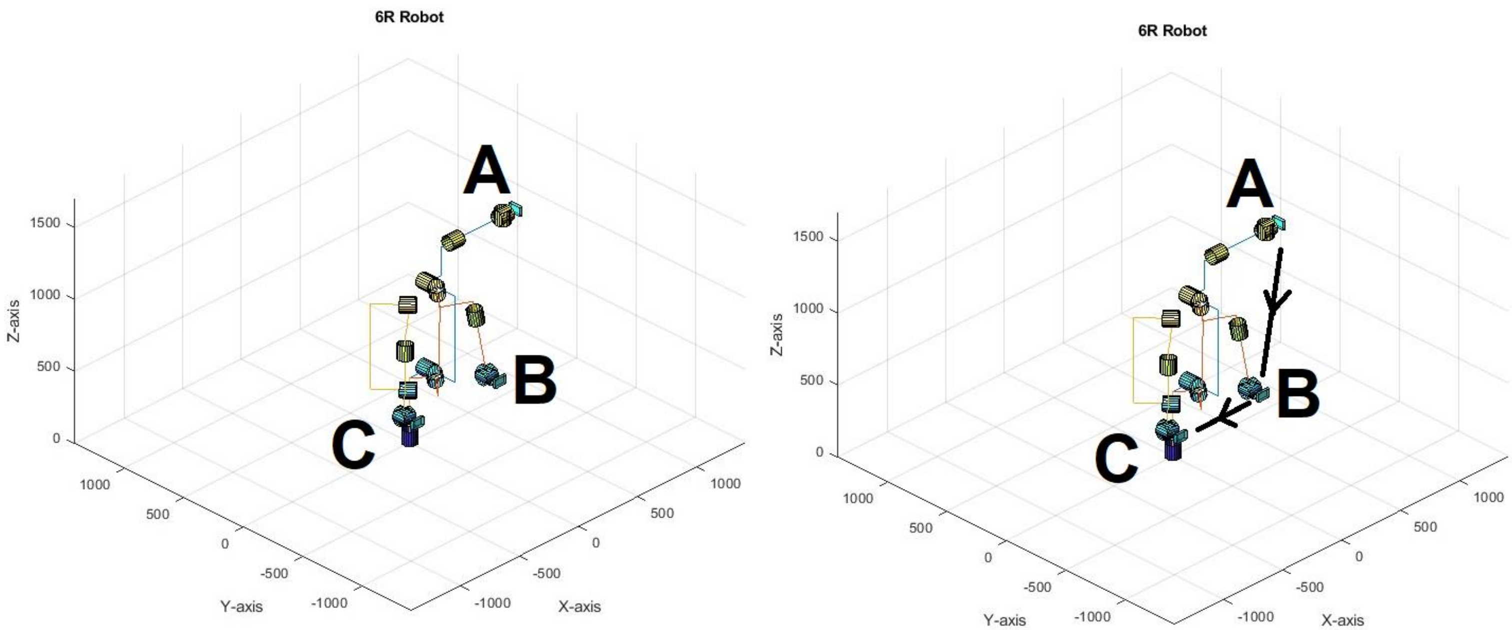


illustration14: A total points-based route is required for the gripper

Regarding the repeatability error

As we mentioned in the introduction, the robot has the Repeatability error (returning consistently to a certain position): $\pm 0.03[mm]$

The accuracy required in this project is the distance from the barcode so that the error in the resulting location is in magnitude of $1[mm]$(for the purpose of the discussion), therefore this error does not interfere with the planning of the route. As we mentioned, this robot is designed to perform operations in space at high speed and with good positional accuracy.

the chosen route

A route was chosen for barcode scanning in a single bucket, where the parts of the route:

1. **between the first state and the second (A to-B)** –It is chosen that the three primary joints will take care of a straight-line path between the two locations, and others to change the orientation from the zero-state orientation to the desired one (with zero velocity and acceleration).
2. **between the second and third state (B to-C)** –The orientation is fixed, the position in y, z fixed and the position in the x-axis changes in time, so that the scanner will accelerate to follow the barcode at relative zero speed and will eventually slow down to a stop after the scan is finished.

Description of the track and its construction

1. **From state A to the state B:**

A transition between two points in space with zero BC:

A path will be built for which the catcher passes between two points in the three-dimensional space, so that at the ends the speed and acceleration are zero:

$$\underline{P}(t) = \underline{P}_0 + \lambda(t)\left(\underline{P}_f - \underline{P}_0\right), \qquad 0 \le t \le T_1$$

where:

$\underline{P}_0$ – the starting point

$\underline{P}_f$ – End point

$\lambda(t)$ –A scalar polynomial that takes care of moving between the points, with the necessary BC

The necessary BC:

$$\lambda(0) = 0, \lambda(T) = 1, \dot{\lambda}(0) = \dot{\lambda}(T) = 0, \ddot{\lambda}(0) = \ddot{\lambda}(T) = 0$$

A polynomial is chosen for which free parameters are the number of requirements:

$$\lambda(t) = a_0 + a_1\left(\frac{t}{T}\right) + a_2\left(\frac{t}{T}\right)^2 + a_3\left(\frac{t}{T}\right)^3 + a_4\left(\frac{t}{T}\right)^4 + a_5\left(\frac{t}{T}\right)^5$$

we will get:$\lambda(t) = 10\left(\frac{t}{T}\right)^3 - 15\left(\frac{t}{T}\right)^4 + 6\left(\frac{t}{T}\right)^5$

Regarding the orientation: We will build a function that receives two rotation matrices R_i, R_f and an initial and final time for their rotation t_i, t_f. In this function, with the help of the relative rotation between the matrices ${}^{i}R_f = R_i^T R_f$, it is possible to build a matrix $R(t)$ that starts with the initial orientation matrix and rotates to the final one by rotating around an axis for the duration of the inserted time up to the rotation angle needed to move between the two matrices . The rotation angle $\theta(t)$ is a polynomial so that the necessary Ts are met.

2. **From stat B to the state C:**

The tracking route for scanning the barcode:

The y, z values of this trajectory are known: $(x(t), y(t) = -400, z(t) = 450),\ T_1 \le t \le T_2$

Now we will divide the tracking into three parts: the gripper starts from rest and accelerates to the speed of the barcode relative to the robot's base ($V = 1\left[\frac{m}{s}\right]$), continues to follow the robot at this speed for the necessary time, then slows down to rest at the end.

First part:

We will require:

$$x(T_1) = 500, x(T_2) = 300,$$

$$\dot{x}(T_1) = 0, \dot{x}(T_2) = -V, \ddot{x}(T_1) = \ddot{x}(T_2) = 0$$

From here we will solve in MATLAB for a polynomial that satisfies these BC.

Part two:

We will require:

$$\mathrm{x}(\mathrm{T}_2) = 300, \mathrm{x}(\mathrm{T}_3) = -300,$$

$$\dot{\mathrm{x}}(\mathrm{T}_2) = \dot{\mathrm{x}}(\mathrm{T}_3) = -\mathrm{V}, \ddot{\mathrm{x}}(\mathrm{T}_2) = \ddot{\mathrm{x}}(\mathrm{T}_3) = 0$$

When the solution:$\mathrm{x}(\mathrm{t}) = -\mathrm{Vt} + \mathrm{a}$

Part three:

We will require:

$$\mathrm{x}(\mathrm{T}_3) = -300, \mathrm{x}(\mathrm{T}_4) = -500,$$

$$\dot{\mathrm{x}}(\mathrm{T}_3) = -\mathrm{V}, \dot{\mathrm{x}}(\mathrm{T}_4) = 0, \ddot{\mathrm{x}}(\mathrm{T}_3) = \ddot{\mathrm{x}}(\mathrm{T}_4) = 0$$

From here we will solve in MATLAB for a polynomial that satisfies these ts.

Regarding the orientation: In this portion of the trajectory, the orientation is kept constant.

The total required route

It is symbolically acquired $\begin{cases} \underline{\boldsymbol{P}}(\boldsymbol{t}), \boldsymbol{0} \le \boldsymbol{t} \le \boldsymbol{T_4} \\ \boldsymbol{R}(\boldsymbol{t}), \boldsymbol{0} \le \boldsymbol{t} \le \boldsymbol{T_4} \end{cases}$, this are the position and orientation needed as a function of time for the gripper to fulfill the task requirements. With the help of discrete steps in time $dt = 0.01[s]$, at each time step, position and orientation requirements are translated into joint values according to the inverse kinematics, the joint values are obtained in time and the robot is printed to verify the correctness of the trajectory.

With the help of the joint values, we can check the speed of the gripper, the loads on the robot's motors (torques on the motors) and draw the robot in 3D space.

Determining the times and distances in each part of the route

It is determined under the requirement that the speed of the joints be within the allowed range according to the manufacturer.

We found that for an industrial robot that weighs less than a kilogram in a clamp, if you want a balance between high speed, stability and tracking, it is recommended to use:

- Maximum linear acceleration of $a_{max} = 2000 \left[\frac{mm}{s^2}\right]$, for aggressive work 4000 is possible.
- Maximum wire speed of $V_{max} = 1000 \left[\frac{mm}{s}\right]$

We will check the fulfillment of the constraints after building the final route in the section verifying the result and performance of the task.

For the first part:

The distance between the points is: $d = |P_0 - P_1| = 975.7561 \cong 976[mm]$.

Therefore: .$\Delta t_{1,min} \approx \frac{976[mm]}{1000\left[\frac{mm}{s}\right]} \approx 1[s]$

In the end we will take:$\mathbf{\Delta t_1 = 2.2[s]}$The reason why the time is much greater than our requirement is related to the time it takes for the barcode to reach the area where the tracking is done.

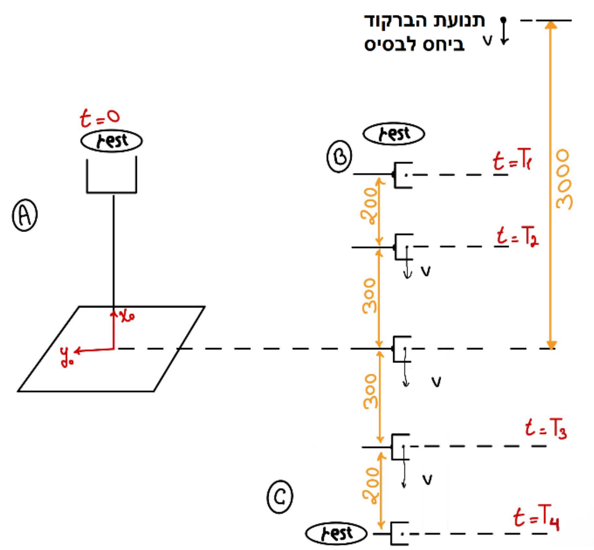


illustration15: Situations in which the gripper is planned to pass to scan the barcode

For the second part:

To accelerate to the speed needed for tracking, we will take an approximation: a $= \frac{1000\left[\frac{\text{mm}}{\text{s}}\right]}{\Delta\text{t}} = 2000 \left[\frac{\text{mm}}{\text{s}^2}\right]$, and from here we will get :$\Delta\text{t} = 0.5[s]$. That is, it is a reasonable acceleration time. We will take an acceleration time of: $\mathbf{\Delta t_2 = 0.5[s]}$

<u>Regarding the choice of distances:</u>

At these times, the location of the barcode is axisx0In relation to the robot will be:

$$x(T_2) = 3000[mm] - 1000\left[\frac{mm}{s}\right] \cdot 2.7[s] = 300[mm]$$

We chose that at this time the gripper will reach a point 500 mm from the axisy0, so that it has 200 mm to accelerate to the speed of the barcode and then 600 mm to follow it (meaning 0.6 seconds duration of scanning the barcode) and then 200 mm to decelerate symmetrically so that it stops at the point C at 500 mm upside down.

In this case we got a maximum linear acceleration of 3555 $\left[\frac{\text{mm}}{\text{s}^2}\right]$ in the part of the acceleration and it is in a acceptable.

Now the path is fully configured and performs the task of scanning a single barcode for 0.6 seconds, at 25 cm from it and so that the camera is perpendicular to it.

In addition- Regarding the idea of scanning more barcodes

We will check if scanning the next barcode, under the speed constraints and minimum time is possible.

We will require that the camera scans the barcode for a minimum time of 0.5 seconds, meaning it scans in the length 500 mm. Hence a path is needed that within half a second will take the gripper 500 mm back, so that it is moving at the same speed and zero acceleration at the boundaries.

We will get that if we want this kind of path done with a polynomial, we are requiring the manipulator to accelerate at unreachable accelerations.

A graph of the position of the gripper based on a polynomial with BC such that it returns 500 mm back to continue the scanning cycle, the velocities correspond to continuing scanning and zero acceleration between the two ends.

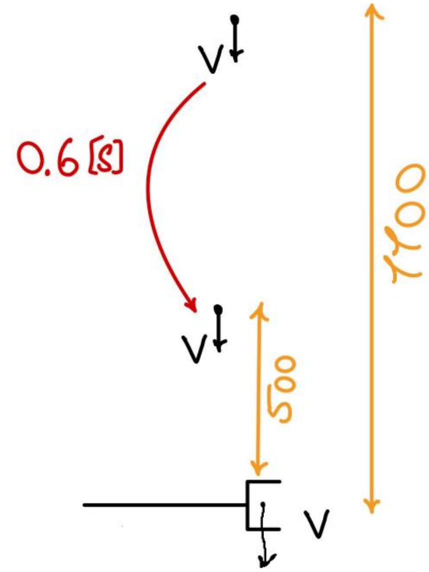
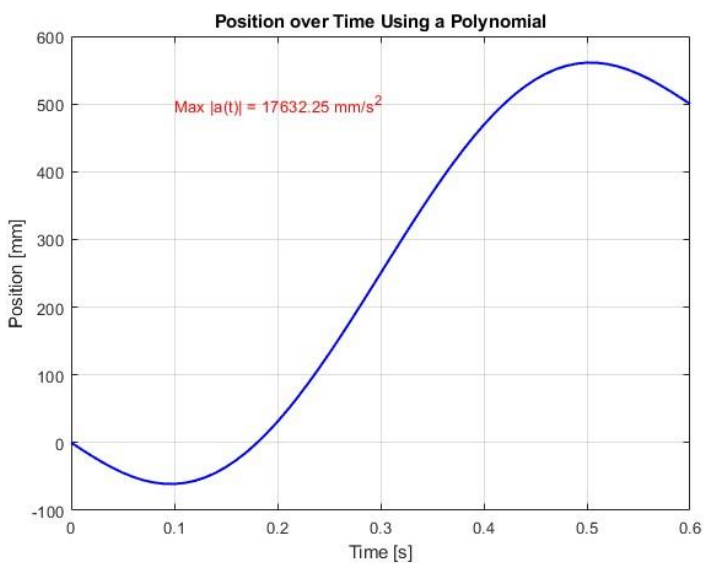


illustration 16: Transition between 2 states if required to scan the barcodes in a cycle



You get a requirement for a maximum acceleration of $17{,}632[\mathrm{mm/s^2}]$. So, even though the robot is designed to be fast, this acceleration is 4 times greater than the maximum allowed.

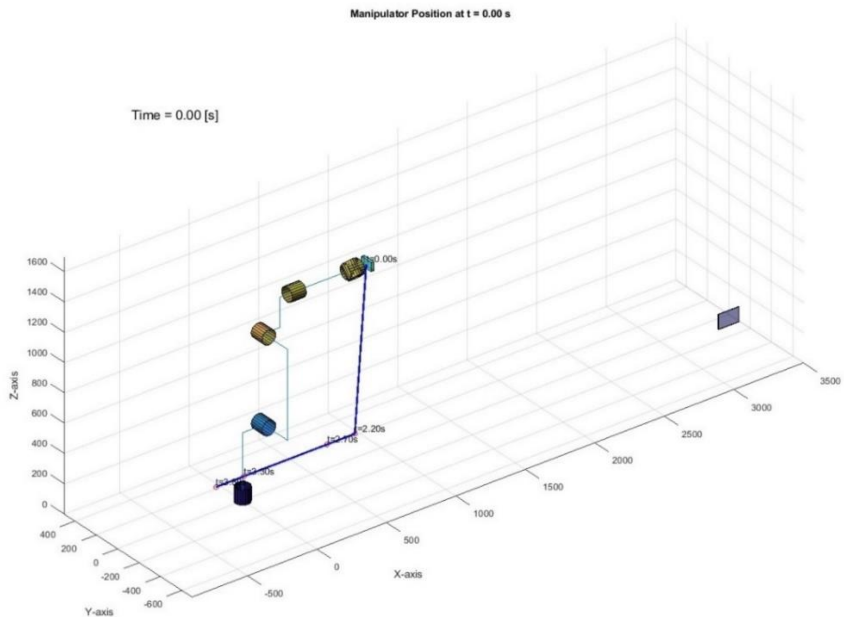
Therefore, it seems that with polynomial trajectories, achieving this goal is not feasible.

Proof of task performance

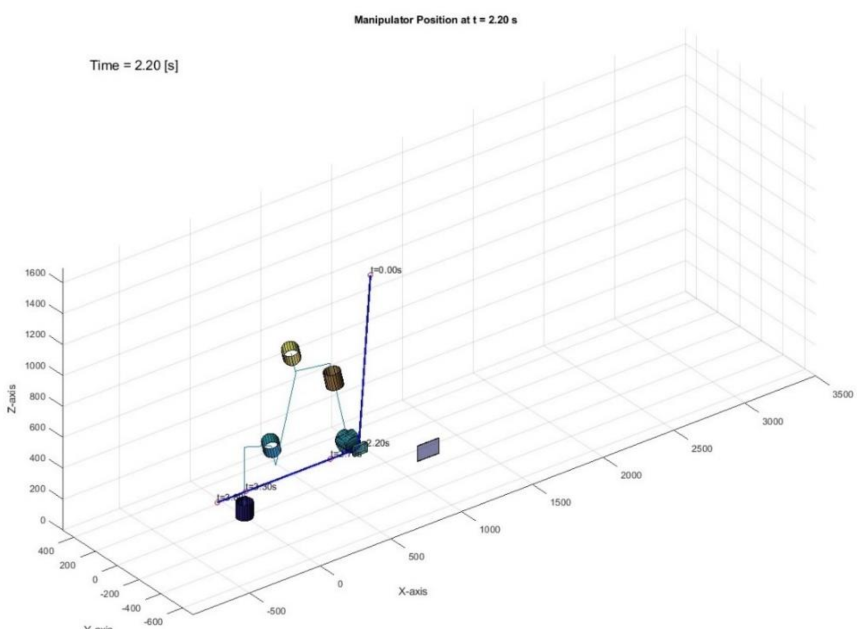
Gripper's track, relative to the base and the floor (the base at the beginning)

We will present the path of movement of the gripper in relation to the base of the robot:

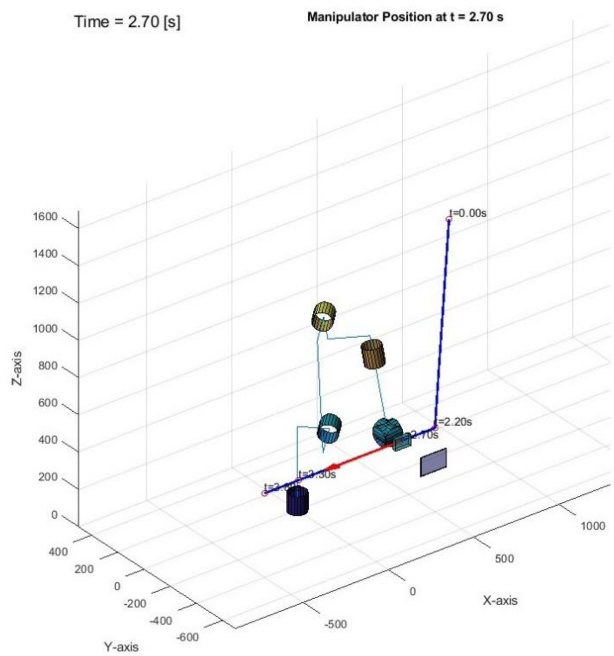
State A, where the gripper begins:



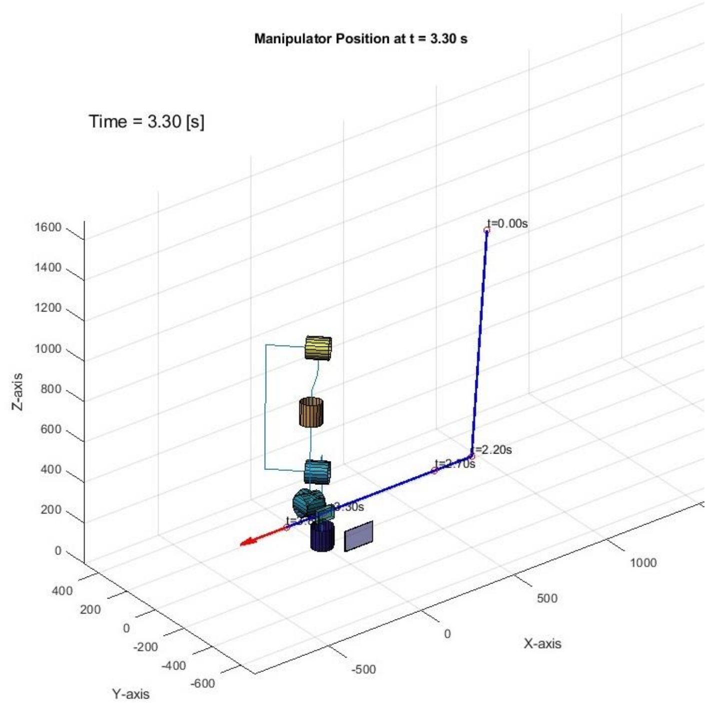
State B, the gripper has rested and is going to accelerate towards tracking the barcode:



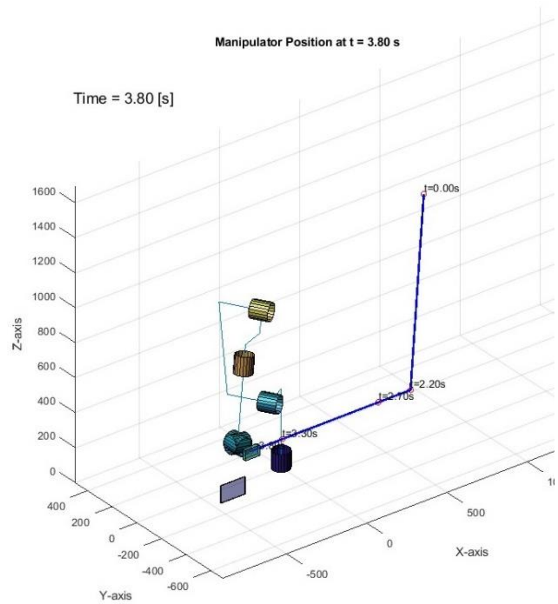
Between states B and C, where the gripper was able to accelerate (relative to the base) to the speed of the barcode for the scan:



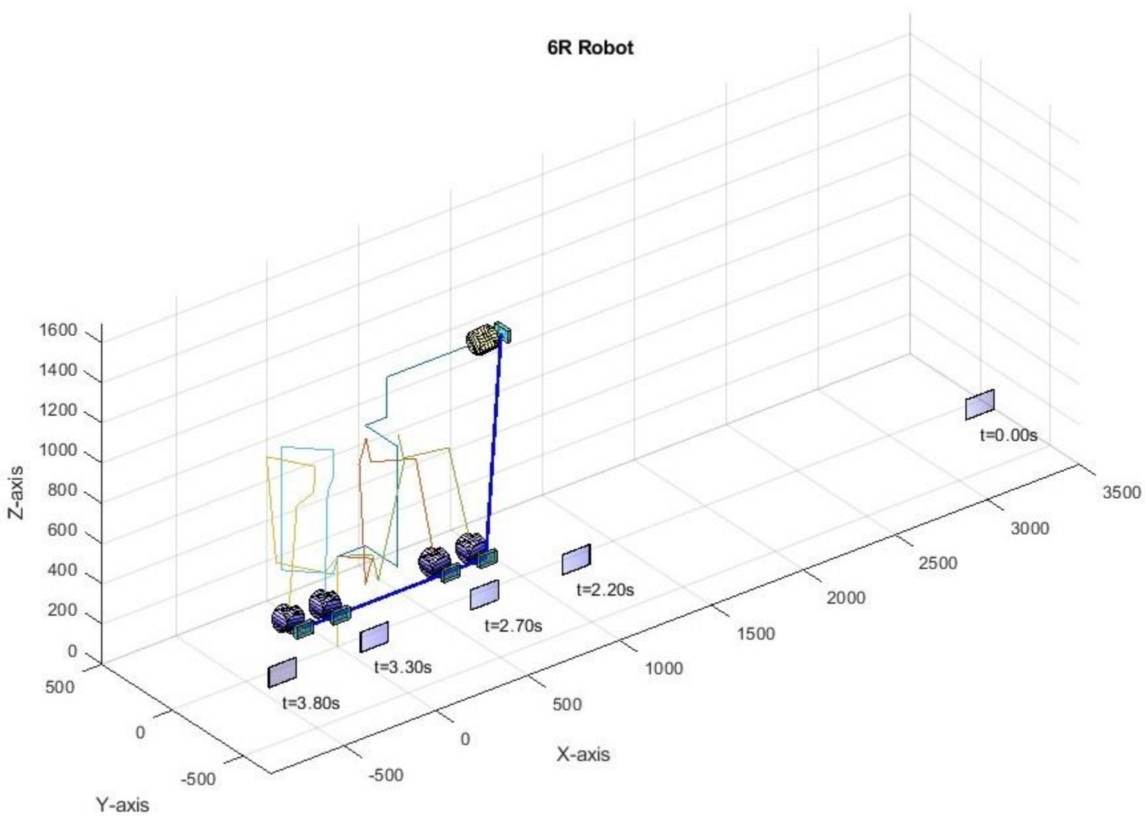
It starts decelerating (relative to the base) after the scan is finished:



State C, ending the scan and stopping the gripper to rest (relative to the base):

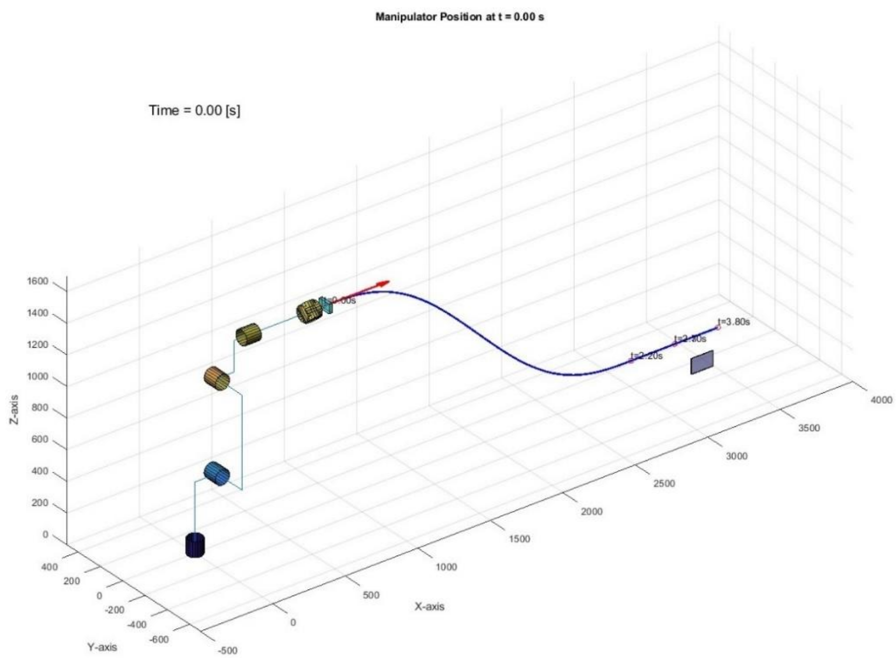


Combined image:

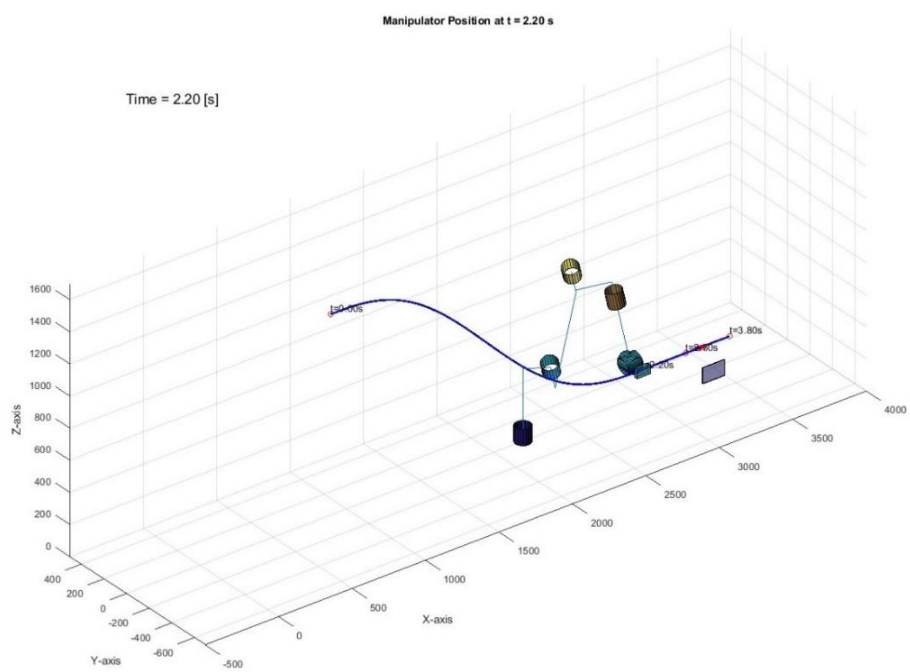


<u>We will present the path of movement of the gripper in relation to the floor (in relation to the base at the beginning, height in relation to the height of the base):</u>

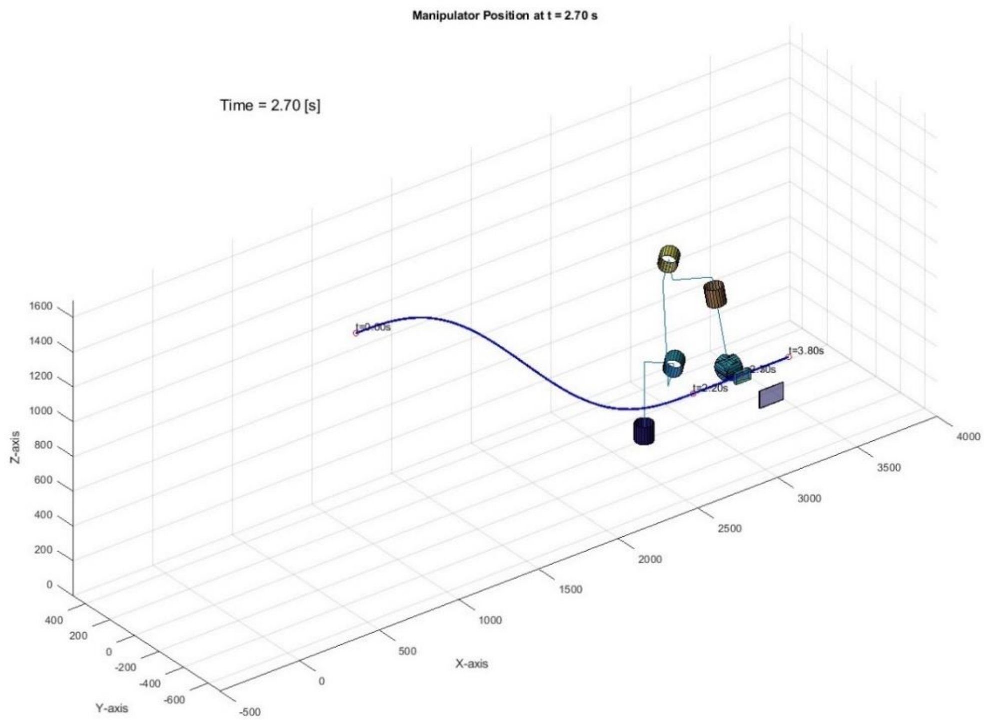
State A, where the gripper begins:



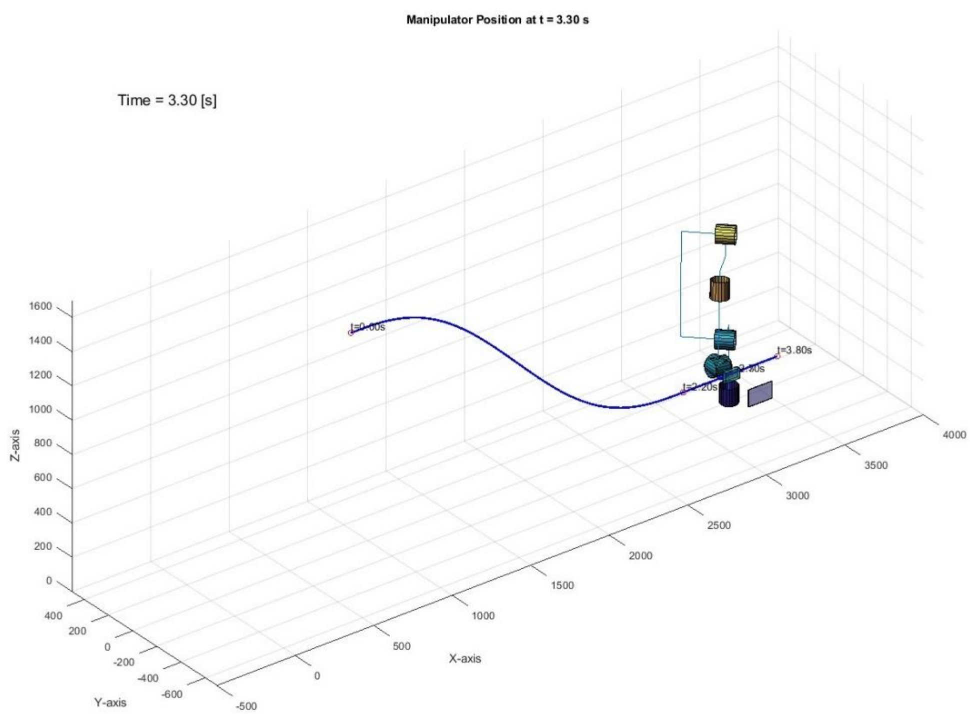
State B, it rested and is going to accelerate (relative to the base) towards tracking the barcode:



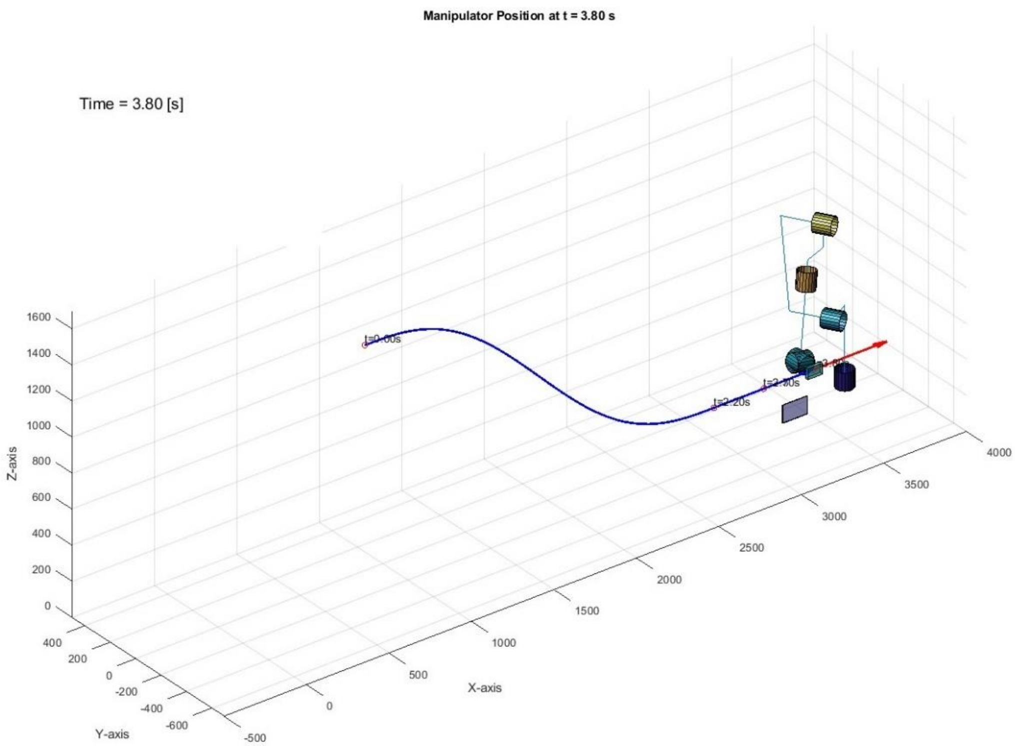
Between states B and C, where the catcher was able to accelerate (relative to the base) to the speed of the track for the scan:



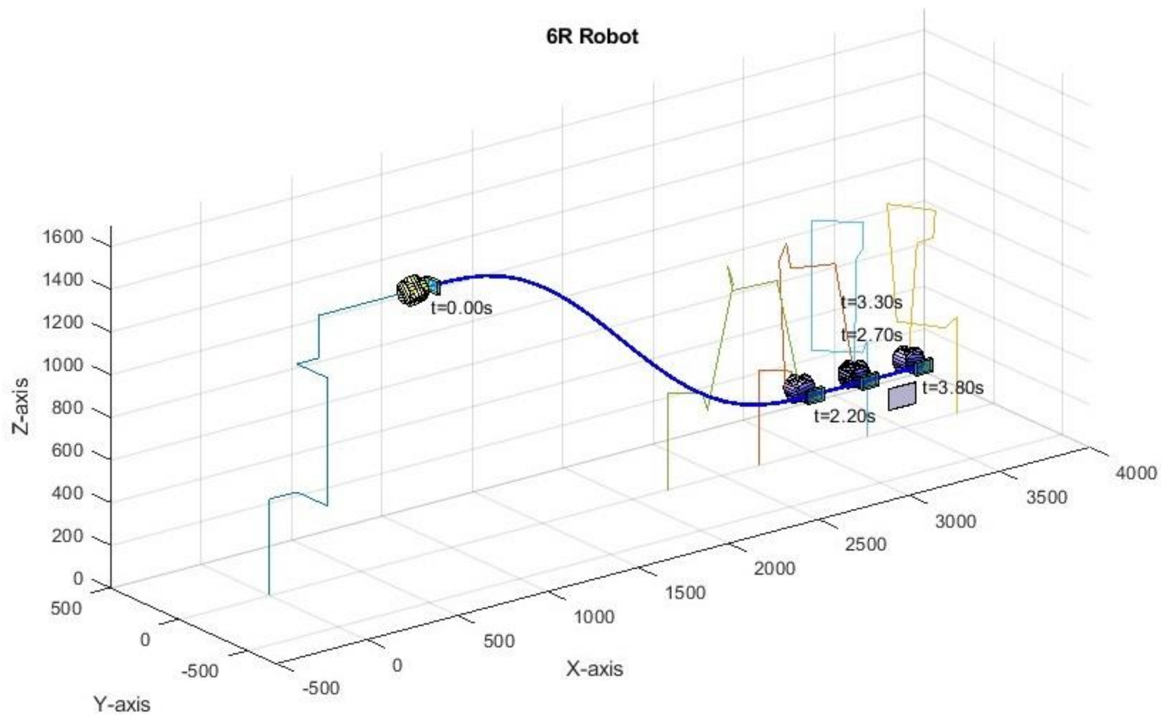
It starts decelerating (relative to the base) after the scan is finished:



State C, ending the scan and stopping the gripper to rest (relative to the base):



Combined image:



Graphs of joint values, positions, speeds, accelerations and tracking check

Using the desired dt.

We would like to show graphs of the joint values over time, their speed and acceleration. In addition, to verify the fulfillment of the requirements we will also want to look at the position, speed and acceleration of the gripper in time (relative to the base of the robot and in relation to the base of the robot at the beginning).

We calculated the joint values from the trajectory and orientation requirement of the gripper and using inverse kinematics with the selection of the inverse kinematics joint branch. We chose to work with discrete time jumps. Therefore, the joint values are discrete in time (with an interval of 0.01 seconds). From here, their speed and accelerations are calculated with the help of numerical derivatives (middle derivative between two time points and front and rear derivatives at the ends).

The speed and acceleration of the catcher is calculated using the kinematic relations of the Jacobian:

$$\underline{V} = J_{L_{3x6}}\underline{\dot{q}} = J_{L_1}\dot{q}_1 + \cdots + J_{L_6}\dot{q}_6$$

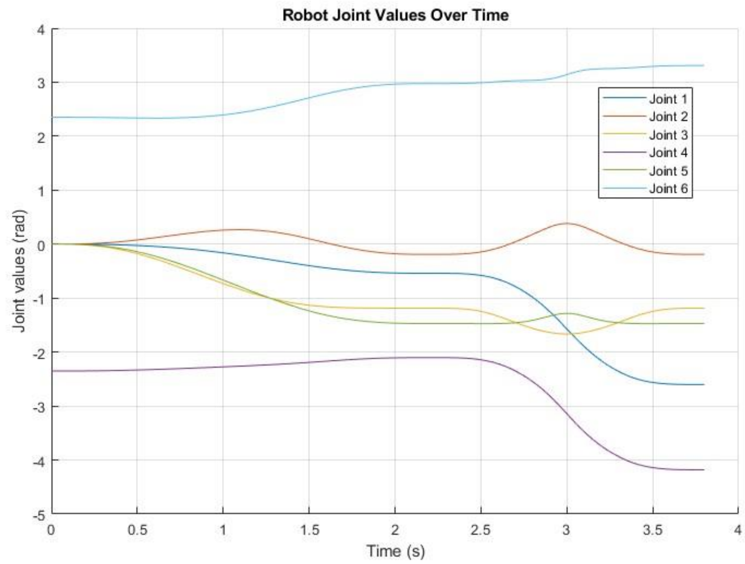
The relative acceleration between the gripper and the robot base is the same as that with the floor because the robot base moves at a constant speed. Therefore, the acceleration of the catch:

$$\underline{a} = \frac{d\underline{V}}{dt} = \frac{d}{dt}\left(J_L\left(\underline{q}\right)\underline{\dot{q}}\right) = \frac{d}{dt}\left(J_L\left(\underline{q}\right)\right)\underline{\dot{q}} + J_L\left(\underline{q}\right)\underline{\ddot{q}}$$

For simplicity it is considered numerically:

$$\underline{a} = \frac{d\underline{V}}{dt} = \dot{V}_x\hat{x} + \dot{V}_y\hat{y} + \dot{V}_z\hat{z}$$

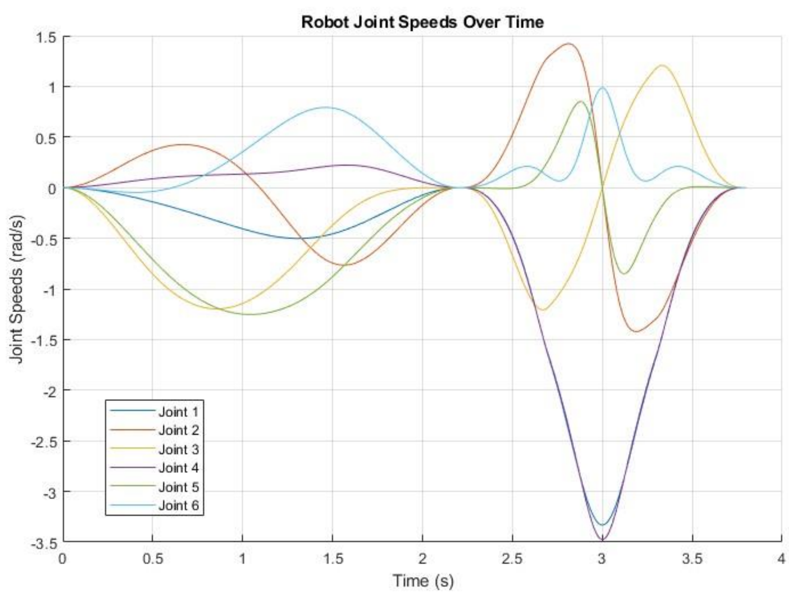
<u>The values of the joints in time:</u>



As we mentioned in the inverse kinematics, the arm needs to start from joint angles 4,6 that correspond to the required trajectory, because in the initial state the robot is in the wrist singularity where there is an intersection in these angles and a suitable solution must be chosen for the continuity of the trajectory.

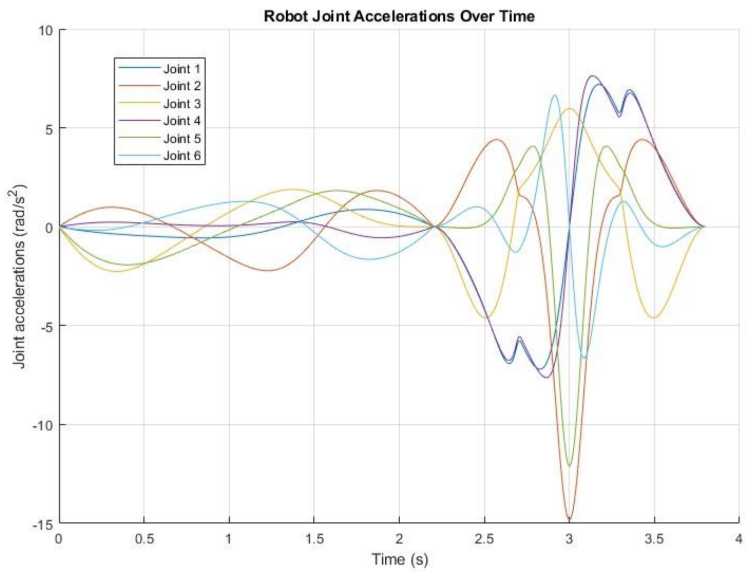
The robot's trajectory fulfills constraints on joint values.

<u>Joint speeds in time:</u>

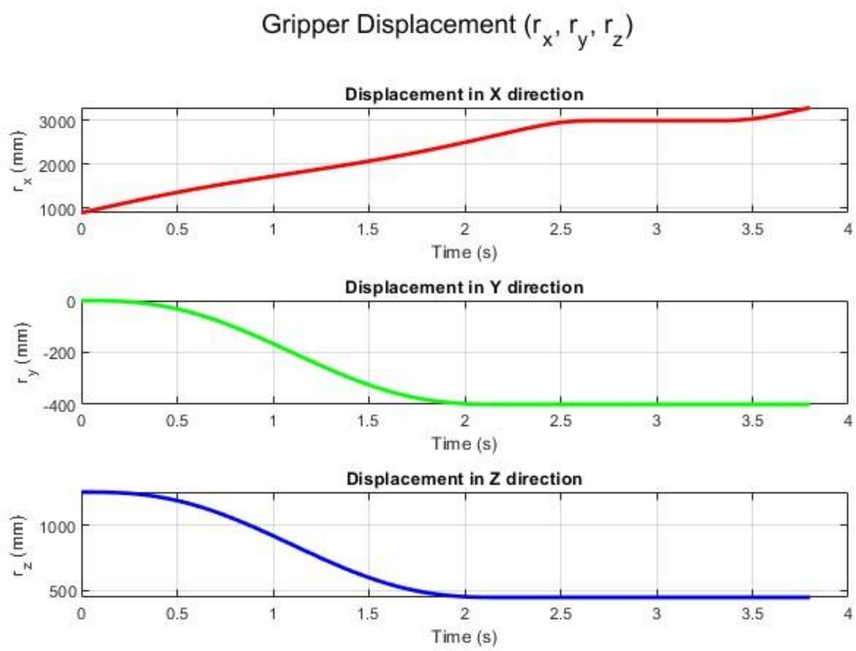


The robot's trajectory fulfills limits on the speed of the joints.

The acceleration of the joints in time:

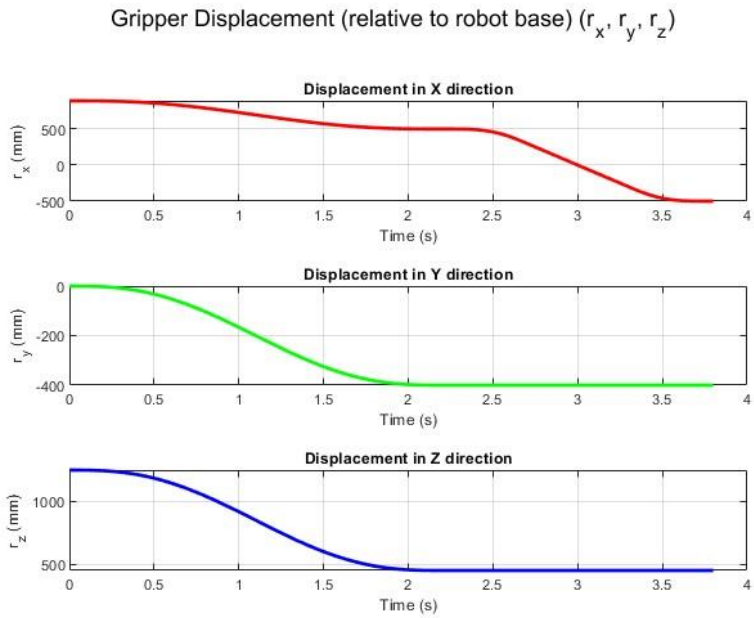


Position of the gripper (in relation to the position of the robot base at the beginning) in time:



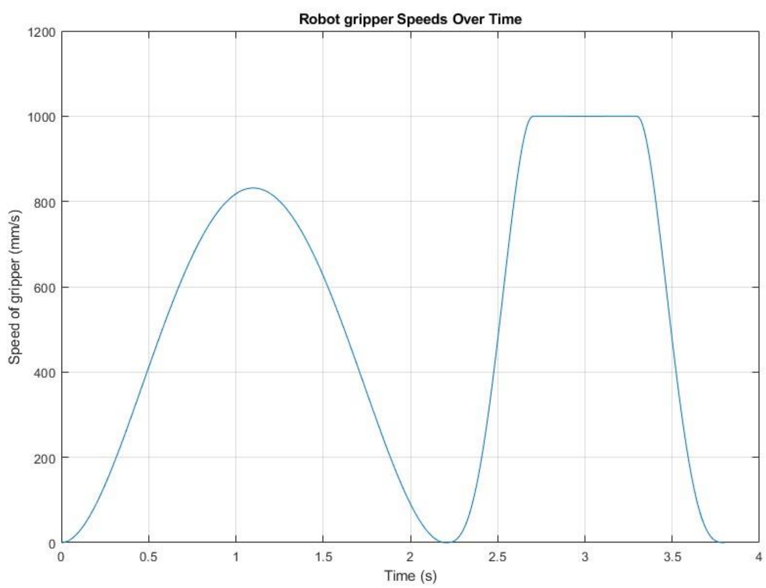
It holds that the robot arrives at the x0-axis position in 2.7 seconds in front of the barcode, when the components of y, z are such that the height of the gripper and the barcode is the same and the distance of the camera from the barcode is 25 cm. This is for 0.6 seconds, until the time is 3.3 seconds and than the scan is finished and the gripper stops being in a fixed position in front of the barcode.

<u>Position of the gripper (in relation to the position of the robot base–relative position) in time:</u>



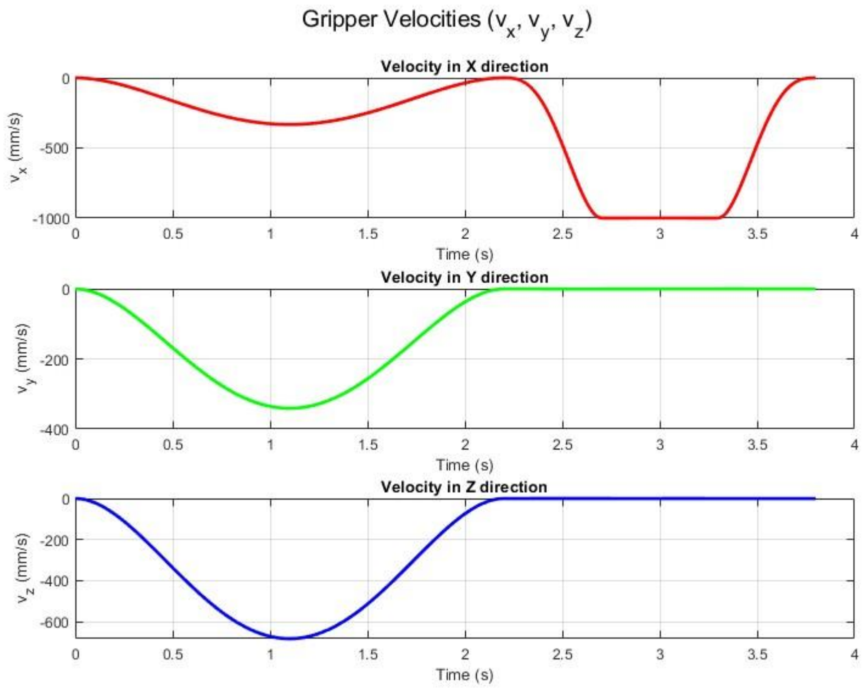
relative to the robot base, while moving from B to-c, the gripper moves in an x-axis only and at the speed of the barcode relative to the robot base as required.

<u>The magnitude of the gripper's speed (relative to the robot's base) in time:</u>



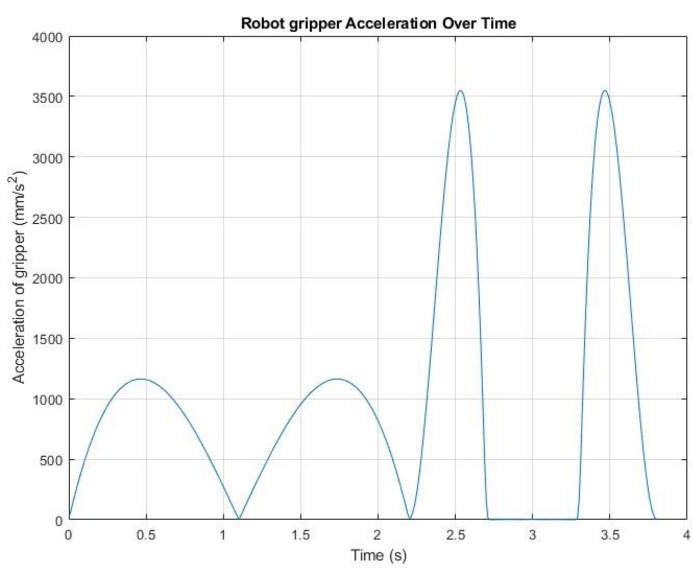
It holds that the speed of the robot (relative to the base) maintains tracking and the magnitudes of the speeds are reasonable (do not pass 1 $\left[\frac{m}{s}\right]$).

<u>Gripper velocity components (relative to the robot base) in time:</u>

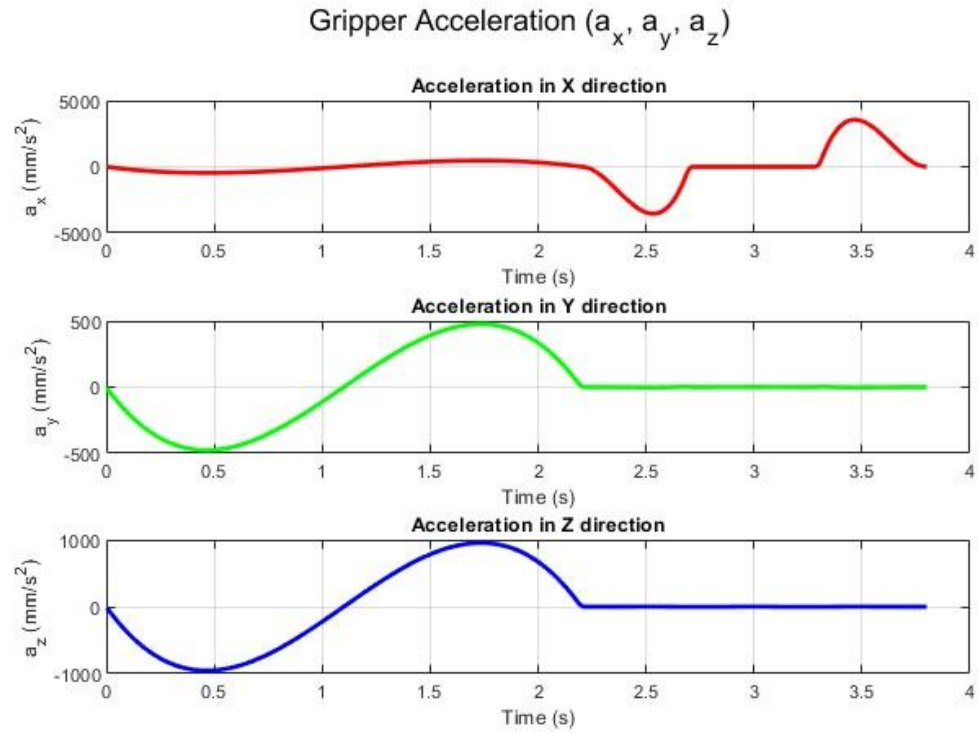


During the tracking between 2.2 and 3.8 seconds the gripper moves, relative to the base, only in the direction of the x0-axis to track the barcode. The speed obtained in the scanning between 2.7 and 3.3 seconds (a scan of 0.6 seconds) corresponds to the speed of the rail in relation to the floor, so that in relation to the floor the catcher was at rest (absolute stop) at this time.

<u>Magnitude of acceleration of the catch in time:</u>



Acceleration components of the trap in time:



We got a continuous acceleration profile in time. This is necessary because if there was discontinuous acceleration it would require discontinuous or infinite torques from the engines, which is not acceptable for most. You need continuous acceleration in time because when operating the robot, continuous torques are required from the motors in time, which is more convenient to realize in contrast to non-continuous torques.

Here, too, you can see that there is movement only in the x-axis during the entire tracking path from B to C.

Calculation of the torques on the Motors

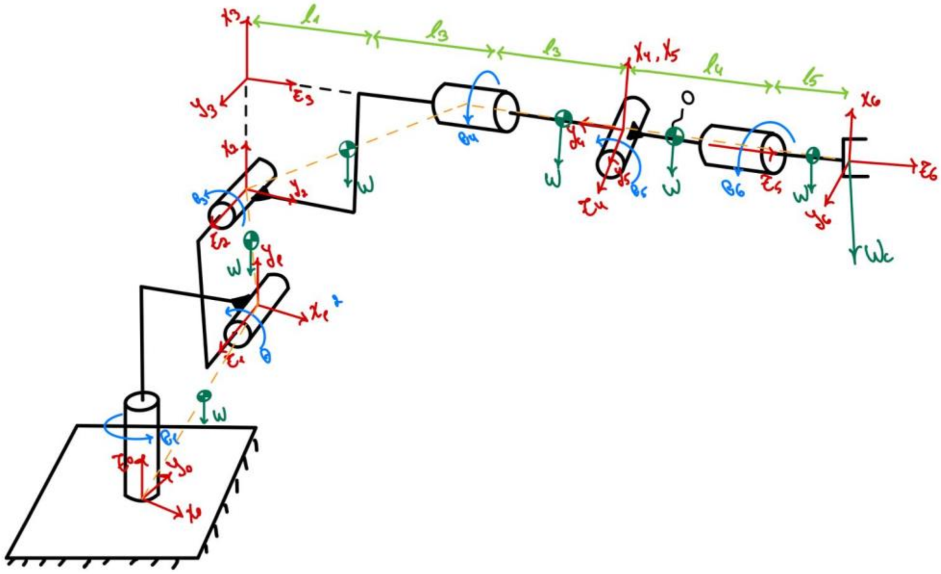


illustration17: The robot skeleton with dimensions for analyzing response to load neglecting movement dynamics (static weightlifting)

- **Link 5 has no mass (does not exist).**

Now we calculate the torques activated by the motors to lift the weight of the camera and the links in time, neglecting the dynamics of the movement (lifting the weight statically).

For systems in statics, the response of the system to several loads is the sum of the responses to each load separately (superposition). Therefore, we calculate the torques that act on the joints of the robot to hold the arm in Nm against each load separately and add up.

Because the distances between some of the joints do not exist in the manufacturer's specifications, for the purpose of the solution we assume:

$$\boxed{l_1 = l_2 = l_3 = \frac{d_4}{3}, \qquad l_4 = 0, l_5 = \frac{d_6}{2}}$$

For camera weight:

Its dimensions are negligible and therefore exerts a straight weight on the gripper:

$$\rightarrow \underline{\tau}_{cam} = \begin{pmatrix} \tau_1 \\ \vdots \\ \tau_6 \end{pmatrix} = J^T \begin{pmatrix} 0 \\ 0 \\ -W_{cam} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

For the vertebrae:

We will note that thanks to the principle of superposition and that the robot is serial, such a solution becomes identical to the case where the load is in the clamp only.

Vertebra from joint 6 to gripper:

We can refer to this as if the gripper is at a distance $\frac{l_5}{2}$ from joint 6 and the weight on it is W.

$$\rightarrow \underline{\tau}_{link\ 6} = \begin{pmatrix} \tau_1 \\ \vdots \\ \tau_6 \end{pmatrix} = J_1^T \begin{pmatrix} 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

when it is the large Jacobian with the substitution$\mathrm{J}_1 \mathrm{d}_6 \rightarrow \frac{\mathrm{l}_5}{2}$

Link from joint 5 to joint 6:nothing

Link from joint 4 to joint 5:

Now $\tau_5 = \tau_6 = 0$.

We can refer to this as if the gripper is at a distance $\frac{\mathrm{l}_3}{2}$ from joint 4 and the weight on it is W.

$$\rightarrow \underline{\tau}_{link\ 4} = \begin{pmatrix} \tau_1 \\ \vdots \\ \tau_4 \end{pmatrix} = J_3^T \begin{pmatrix} 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

When J_3 is the large Jacobian with the replacement $\mathrm{d}_4 \rightarrow \mathrm{l}_1 + \mathrm{l}_2 + \frac{\mathrm{l}_3}{2}$, and deletion of the 6,5th columns.

Link from joint 3 to joint 4:

Now $\tau_4 = \tau_5 = \tau_6 = 0$.

We can refer to this as if the gripper is half the distance between joints 3,4 and the weight on it isW

$$\rightarrow \underline{\tau}_{link\ 3} = \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} = J_4^T \begin{pmatrix} 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

When J_4 the Jacobian is large, when: $d_4 \rightarrow \frac{l_1+l_2}{2}, a_3 \rightarrow \frac{a_3}{2}$. And columns 4,5,6 go down.

<u>Link from joint 2 to joint 3:</u>

Now $\tau_3 = \tau_4 = \tau_5 = \tau_6 = 0$.

We can refer to this as if the gripper is at a distance $\frac{a_2}{2}$ from joint 2 and the weight on it is W.

$$\rightarrow \underline{\tau}_{\text{link }2} = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} = J_5^T \begin{pmatrix} 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Where J_5 is the large Jacobian with the replacement $a_2 \rightarrow \frac{a_2}{2}$, and deleting the 3,4,5,6 columns

<u>Last link from chapter 1 to chapter 2:</u>

Now $\tau_2 = \tau_3 = \tau_4 = \tau_5 = \tau_6 = 0$.

We can refer to this as if the gripper is half the distance between joints 1,2 and the weight on it W.

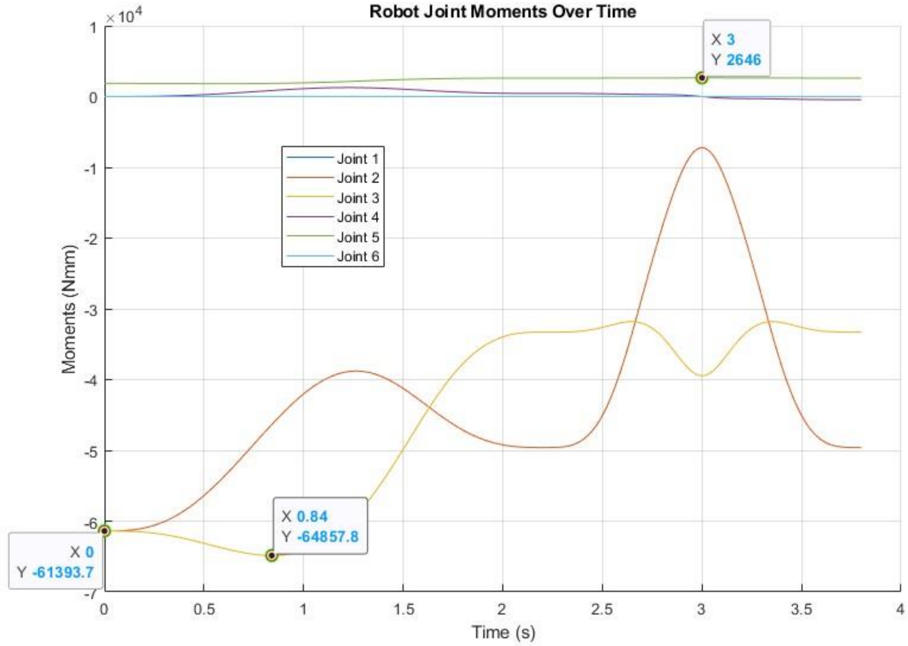
$$\rightarrow \tau_{link\ 1} = J_6^T \begin{pmatrix} 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

when J_6 is the great Jacobian that is considered for this case.

But, in this case, from the link between joints 1 and 2, we get that its weight does not require a torque in the vertical axis operated by the engine, therefore $\tau_{link\ 1} = 0$.

In conclusion, we will accept that the invested torques: $\underline{\tau} = \sum \underline{\tau}_i$

<u>We will get the following torques as a function of time in MATLAB:</u>



It makes sense that joints 2 and 3 invest most of the torque in lifting because lifting weight requires torques whose axes are in the x, y plane and not joints with sloped of vertical axes. Joint 5 also invests torque in lifting, but it is much lower than in joints 2,3.

<u>We will check if the torque invested by joint 3 makes sense (of 2 should be in the same size because they are geometrically close):</u>

Joints 3 lifts a total weight of three links and a camera that is $W_3 = 12.7[kgf]$. Therefore, the estimated torque it needs to lift is roughly half the distance to the clamp, that is $r = \sqrt{\left(\frac{a_3}{2}\right)^2 + \left(\frac{d_4+d_6}{2}\right)^2} \approx 380[mm]$, such that the torque it supports $\boldsymbol{\tau_3 \sim W_3 r = 4.7 \cdot 10^4 [Nmm]}$. It's as obtained in the graph.

<u>For Joint 5:</u>Approximately one can get $\boldsymbol{\tau_5 \approx \frac{d_6}{2}[mm] \cdot (W + W_{cam})[kgf] = 2303[Nmm]}$ which matches the results in the code (received $1861[Nmm] < \tau_5 < 2646[Nmm]$).

<u>Testing the robot's ability to handle these torques:</u>

According to the specification:$\tau_{4,5} < 22[Nm] = 22{,}000[Nmm], \tau_6 < 9.8[Nm] = 9{,}800[Nmm]$

Joints 5, as we have seen, is in the desired range of torques (8 times smaller than the maximum). There are no torques on joint 6 and lower torques on joint 4 than on joint 5.

For joint 3, a maximum of $\boldsymbol{\tau_{3,max} = 65[Nm]}$ is obtained, which is a greater torque than is recommended for these robots to lift (according to the Internet, between 40 and 50 Newton-meters is recommended). For joint 2 a maximum of $\boldsymbol{\tau_{2,max} = 61.4[Nm]}$.

Appendix - reference article and the code

Reference article for calculating singularity of the robot:

This article presents a method for an efficient identification of Singularities in and industrial 6R Robot. According to this article: $|J| = |J_w|$

Vukobratović, M., Kircanski, N., & Stokić, D. (2014). Singularity Analysis for a 6 DOF Family of Robots. In Proceedings of the Conference on Advances in Robot Kinematics (ARK). Springer, Cham.https://doi.org/10.1007/978-3-319-02054-9_34

6R robot singularities.pdf

The code used in the project:

Includes several key parts:

1. Solving the direct kinematics (4-90)
2. Solving the inverse kinematics (91-140)
3. Jacobian achievement (141-208)
4. route planning (209-829)
    - Route planning and parametric orientation (209-400)
    - Converting the track into joint values and drawing it in relation to the robot base and making a video (401-686)
    - The same but for the track relative to the robot base at the beginning "relative to the floor" (687-829)
5. Drawing the results (830-1042)
6. Defined Functions (1782-1043)