TPK4170 Robotics Project

Candidate numbers: 10061, 10060, 10027

2021-12-09

Contents

List of Figures

List of Tables

Task 1

Forward Kinematics Theory

1.1 The Denavit-Hartenberg convention

1.1.1 Introduction

A robot can be seen as a set of rigid bodies called links, which may be connected by joints. It is said to be *closed chain* if the links and joints form no loop, otherwise it is called *open chain*. In this task we only study open chain robots, in particular robot arms that can be represented by a succession of links and joints. The link that is fixed to the ground is the *base* and the link at the tip is the *end effector*.

A frame of reference is attached to each link and the position of one frame in relation to the previous one is represented by the homogenous transformation matrix T_i^{i-1}. Because the robot is articulated, the matrix T_i^{i-1} must somehow depend on the variable of joint i. The position of the end effector in relation to the base is represented by a homogeneous transformation matrix T_E^B, which can be computed by composing the transformations along the chain:

$$T_E^B = T_0^1 \cdot T_1^2 \cdot \ldots \cdot T_{i-1}^i \tag{1.1}$$

1.1.2 Describing a transformation with 4 numbers

The Denavit-Hartenberg (DH) is a convention broadly used in robotics. It was introduced in order to standardize the attachment of cordinate frames to robots. It provides rules on how to attach frames to each link in such a way that the homogeneous matrix T_i^{i-1} can be described with only 4 parameters instead of 16.

When using the DH convention, the transformation to go from one frame to the next is the composition of 4 operations shown on figure 1.1. Starting from frame $i-1$, one must:

1. Translate by a distance d_i along the z_i axis.
2. Rotate by an angle ϑ_i around the z_i axis. This new frame is called i'.
3. Translate by a distance a_i along the $x_{i'}$ axis.

4. Rotate by an angle α_i around the $x_{i'}$ axis. This new frame is now the frame i

The parameters a_i and α_i are constant and just depend on the geometry of the link. One of the parameter d_i or ϑi varies depending on wether joint i is translational or rotational. Having the four parameters a_i, d_i, α_i, ϑ_i, one can infer the corresponding homogeneous transformation:

$$T_i^{i-1} = \begin{bmatrix} \cos\vartheta_i & -\sin\vartheta_i \cos\alpha_i & \sin\vartheta_i \sin\alpha i & a_i \cos\vartheta_i \\ \sin\vartheta_i & \cos\vartheta_i \cos\alpha i & -\cos\vartheta_i \sin\alpha i & a_i \sin\vartheta_i \\ 0 & \sin\alpha i & \cos\alpha i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.2}$$

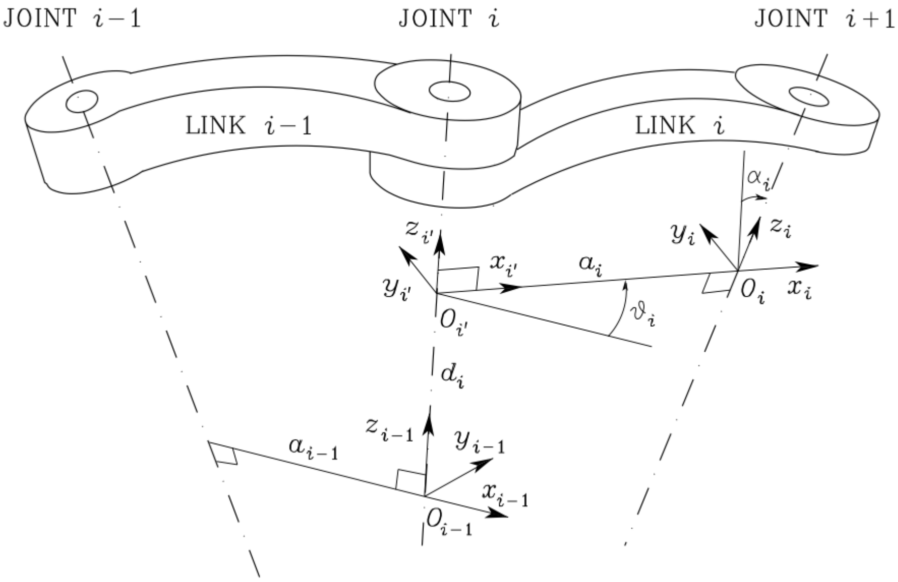

Figure 1.1: Denavit Hartenberg convention rules [1]

1.1.3 Attaching the frames

To be able to describe the transformations with only 4 numbers, the frames need to stick to the following rules:

- z_i aligns with the rotational axis of joint i+1
- O_i is placed at the intersection of z_i and the common normal of z_{i-1} and z_i
- $O_{i'}$ is placed at the intersection of z_{i-1} and the common normal of z_{i-1} and z_i
- x_i aligns with the common normal of z_{i-1} and z_i pointing away from O_i
- y_i is chosen according to the right hand rule for coordinate frames

Howerver these rules are not resctrive enough to make sure that there is just one unique way to attach the frames, because:

- There is no frame $i-1$ to constrain frame 0, so O_0 and x_0 can be fixed arbitrarily.
- There is no joint $n+1$ to constrain frame n so z_n is usually made parallel to z_{n-1}

There are some special cases which also do not allow a unique definition of frames:

- z_{i+1} and z_i are parallel: O_i and x_i can be selected arbitrarily.
- z_{i+1} and z_i intersect: O_i and x_i can be selected arbitrarily.
- joint i is prismatic: x_{i-1} can be selected arbitrarily.

1.2 A modified DH convention

The modified DH convention presented in [2] differs from the original DH convention presented in [1]. In figure 1.2 the frames are attached to the links differently than in figure 1.1 (see table 1.1). The 4 parameters are also defined differently, see table 1.2. Like mentioned before, the attachment of the frames does not change the forward dynamics. Anyway, sticking with one convention makes calculations more intuitive and easier to understand for others.

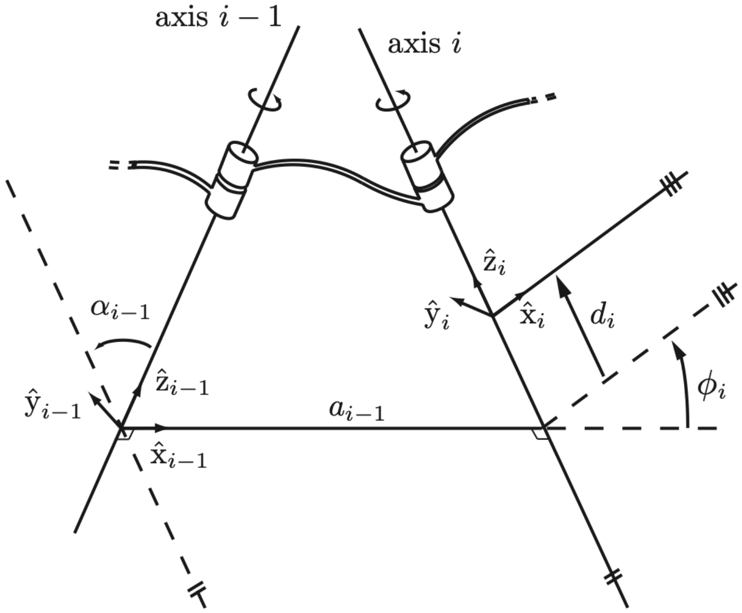

Figure 1.2: Modified Denavit Hartenberg convention rules [2]

frame parameters	original DH	modified DH
z_i	aligns with the rotational axis of joint i+1	aligns with the rotational axis of joint i
O_i	placed at the intersection of z_i and the common normal of z_{i-1} and z_i	is placed at the intersection of z_i and the common normal of z_i and z_{i+1}
x_i	aligns with the common normal of z_{i-1} and z_i pointing away from O_i	aligns with the common normal of z_{i+1} and z_i and pointing away from O_i

Table 1.1: frame attachment original versus modified DH

DH parameters	original DH	modified DH
a_i	length of common normal of z_{i-1} and z_i	length of common normal of z_i and z_{i+1}
d_i	distance between O_{i-1} and intersection of common normal of z_{i-1} and z_i with z_{i-1}	distance between O_i and intersection of common normal of z_{i-1} and z_i with z_i
α_i	angle between z_{i-1} and z_i about x_i	angle between z_{i-1} and z_i about x_{i-1}
ϑi	angle between x_{i-1} and x_i about z_{i-1}	angle between x_{i-1} and x_i about z_i

Table 1.2: DH parameter original versus modified DH

In the special case where two consecutive z_i and z_{i-1} intersect [2] recommends to pick a x_{i-1} which is perpendicular to the plane spanned by z_i andz_{i-1}. [1] says that an arbitrary x_i should be picked in such a case.

1.3 The Product-of-Exponentials convention

1.3.1 Twists

Rigid body motions can be expressed by consecutively applying a rotation and translation on a body. Actually any transformation that is a combination of rotations and translations can be seen as the combination of:

- One rotation by an angle θ around a certain axis in space, called the *screw axis*
- One translation by a distance $h\theta$ along this same *screw axis*

This operation is called a *twist*, and figure 1.3 illustrates this fact. The screw axis, as any axis, is the combination of a point q and a direction $\hat{s}$.

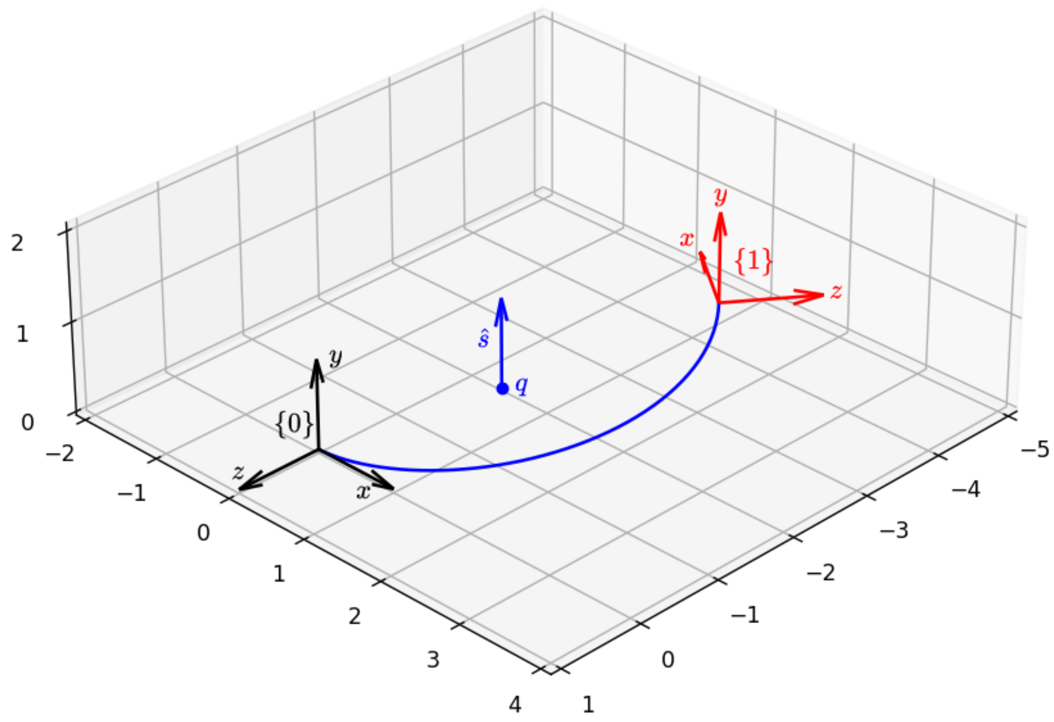

Figure 1.3: Visualization of a screw axis and a twist operation

Instead of representing the screw axis with q, $\hat{s}$, θ and h, we can represent it as a vector of 6 components, 3 of which describe the rotation and the 3 other describing the translation, which is more intuitive to think about:

$$\mathcal{V} = \begin{bmatrix} \omega \\ v \end{bmatrix} = \begin{bmatrix} \hat{s}\theta \\ -\hat{s}\theta \times q + h\hat{s}\theta \end{bmatrix}$$

From this representation, we notice that the twist $\mathcal{V}$ can be seen as the product of the angle θ and a "normalized twist" $\mathcal{S}$ called a *screw*:

$$\mathcal{V} = \mathcal{S}\theta$$

Here, θ determines "how far" around and along the screw axis do we rotate and translate. To summarize, twists are a generalization of the concept of *angle-axis* that is well known for rotations, this time applied to translations as well.

1.3.2 Relationship between twists and transformations

The axis-angle representation of a rotation is convertex to a rotation matrix by applying the exponential function. Equivalently, a twist is converted to a transformation matrix with the exponential function.

$$\exp : [S]\theta \in se(3) \rightarrow T \in SE(3)$$
$$\log : T \in SE(3) \rightarrow [S]\theta \in se(3)$$

In the next section, the PoE formula is presented for solving the forward kinematics for complex open chain robots. The homogeneous transformations between links is calculated with the matrix exponential of screw motions. In this sense, it is important for the concept of matrix logarithm and exponential of twists.

1.3.3 Relationship between twists and frames

The same twist can be represented in different reference frames. For instance, two twists need to be in the same frame to be composed or added together. To change the representation of a twist from frame a to frame b, we proceed as follows:

$$[S_b] = T_a^b [S_a] T_b^a \tag{1.3}$$

Conveniently, this double matrix multiplication can be collapsed into one, with a matrix called the *adjoint*:

$$[S_b] = Ad_{T_a^b}[S_a] \tag{1.4}$$

1.3.4 Product of Exponentials formulation

In comparison to DH convention in the Product of Exponential Formula (POE Formula) there is no convention for attaching a frame to each link. It is just necessary to attach a frame at a stationary point and the end effector. The rotation of each joint i is represented by a screw motion, which influences all links between joint i and end effector. When the robot is in its zero position and one does just move the last joint with θ_n, the end effector pose is represented by $T = e^{[S_n]\theta_n}M$. In contrast, when moving the last two joints θ_n and θ_{n-1}, the end effector pose is represented by $T = e^{[S_{n-1}]\theta_{n-1}}e^{[S_n]\theta_n}M$. This is what figure 1.4 shows. M is a homogeneous transformation representing the end effector frame relative to the base frame when the robot is in its home position. S_i is the screw axis of each joint i with the robot beeing in its home position. This screw axis can be represented in the fixed space frame or end effector frame. The first PoE formula is called spatial form of the Product-of-Exponentials formulations (see 1.5). The second PoE formula is called body form of the Product-of-Exponentials formulations (see 1.6). The homogeneous transformation representing the end effector relative to the base frame in the fixed space frame is shown in the following:

$$T = e^{[S_1]\theta_1} ... e^{[S_{n-1}]\theta_{n-1}} e^{[S_n]\theta_n} M \tag{1.5}$$

The screw motion of a joint i just impacts the pose of the joints i+1 to end effector frame but not any joint between the base and joint i-1. Therefore, it makes sense that in the formula 1.5 the M matrix is first transformed by the screw motion of the joint n. Using the matrix identity $e^{M^{-1}PM} = M^{-1}e^P M$ we can start to move the M matrix from the right side of the equation 1.5 to the left side:

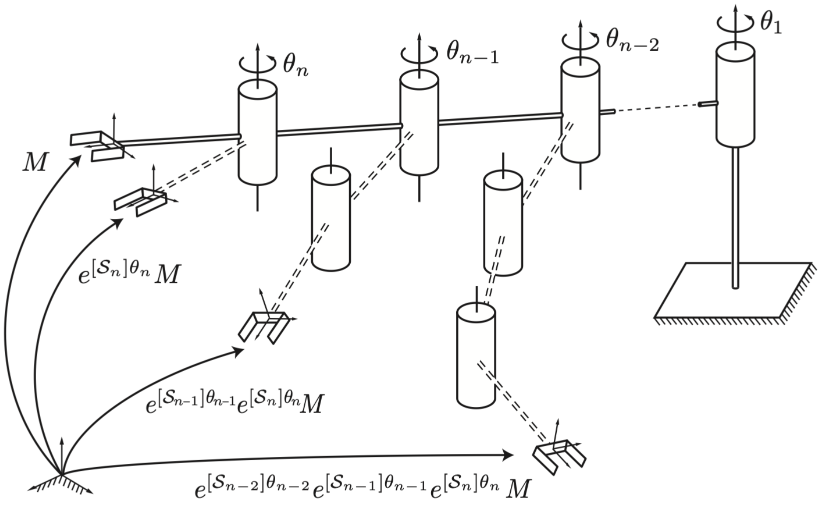

Figure 1.4: Visualization of PoE Formula

$$T = Me^{M^{-1}[S_1]M\theta_1} \ldots e^{[M^{-1}[S_{n-1}]M\theta_{n-1}} e^{M^{-1}[S_n]M\theta_n} \tag{1.6}$$

$M^{-1}[S_i]M$ is representing the screw axis of joint i in the end effector frame. Now the screw motion of joint i impacts all joints between base and joint i-1 but not any joint between joint i+1 and the end effector. Therefore, it makes sense, that in formula 1.6 M is first transformed by the screw motion of the joint 1.

1.4 Relationship between DH and PoE

1.4.1 Relation between DH-convention and PoE formula

When using the DH-convention, the relative motion between two links is represented by four parameters $a_i, d_i, \alpha_i, \vartheta i$. The homogeneous transformation between two links can be writen as following:

$$T_i^{i-1} = Rot(\hat{x}, \alpha_{i-1}) Trans(\hat{x}, a_{i-1}) Trans(\hat{z}, d_i) Rot(\hat{z}, \phi_i) \tag{1.7}$$

In the case of rotational joints the rotation around x and the translation along x and z is constant and can be written as following:

$$M_i = Rot(\hat{x}, \alpha_{i-1}) Trans(\hat{x}, a_{i-1}) Trans(\hat{z}, d_i) \tag{1.8}$$

The screw axis of the rotational joint is the following:

$$[A_i] = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{1.9}$$

Applying the matrix exponential of $[A_i]\theta_i$ returns the homogeneous transformation for the rotation around z:

$$Rot(\hat{z}, \phi_i) = e^{[A_i]\theta_i} \tag{1.10}$$

The equations from equation 1.8, 1.9 and 1.10 lead to the following homogeneous transformation:

$$T_i^{i-1} = M_i e^{[A_i]\theta_i} \tag{1.11}$$

Concatenating equation 1.11 for all links in the open chain solves the forward kinematics:

$$T_n^0 = M_1 e^{[A_1]\theta_1} M_2 e^{[A_2]\theta_2} ... M_{n-1} e^{[A_{n-1}]\theta_{n-1}} M_n e^{[A_n]\theta_n} \tag{1.12}$$

By using the following matrix identity $Me^P M^{-1} = e^{MPM^{-1}}$ to $Me^P = e^{MPM^{-1}} M$ we can rewrite equation 1.11:

$$T_i^{i-1} = e^{M_i[A_i]\theta_i M_i^{-1}} M_i \tag{1.13}$$

Applying equation 1.13 to 1.12:

$$\begin{aligned} T_0^n &= e^{M_1[A_1]M_1^{-1}\theta_1}(M_1 M_2) e^{[A_2]\theta_2} ... M_{n-1} e^{[A_{n-1}]\theta_{n-1}} M_n e^{[A_n]\theta_n} \\ &= e^{M_1[A_1]M_1^{-1}\theta_1} e^{(M_1 M_2)[A_2](M_1 M_2)^{-1}\theta_2}(M_1 M_2 M_3) e^{[A_3]\theta_3} ... M_n e^{[A_n]\theta_n} \\ &= e^{[S_1]\theta_1} e^{[S_2]\theta_2} ... e^{[S_n]\theta_n} M \end{aligned} \tag{1.14}$$

where

$$[S_i] = (M_1 ... M_{i-1})[A_n](M_1 ... M_{i-1})^{-1} \tag{1.15}$$

$$M = M_1 M_2 ... M_n \tag{1.16}$$

From equation 1.3 and 1.4 we know that the same Twist can be represented in different reference frames. A_i is the screw axis of joint i in the i frame. S_i is the same screw axis as A_i, but it is transformed into another frame. The screw axis of joint 1 is transformed by M_1, whereas the screw axis of joint i is transformed by $M_1 ... M_{i-1}$. S_i. This means that the screw axis of all joints are transformed from the local joint frames A_i into the base frame S_i.

After using the DH-convention to attach frames to each link, it is possible to extract the four DH-parameters. From these four parameters the corresponding homogenoeous transformation matrices can be established (rotation around x, translation along x, translation along z, rotation around z). Depending on the type of the link, three of the homogeneous transformation matrices are constant and can be summarized in a M matrix. Instead of using 1.2 the matrix exponential of the screw motion is used to find the forth homogeneous transformation. By concatenating the transformations between all links the transformation from base to end effector frame is found. Applying the matrix identity shown above we can pull the M_i matrices of all linkst to the right of the equation and combine them in the M matrix. During this process the reference frames of the screw axes in the the matrix exponential are changed to the base frame.

In summary, the DH was used to attach the frames to the links and to find the minimal set of parameters for the transformation between two consecutive links. Using some tricks the transformation from 1.7 can be rewritten in a way that it is identical to the PoE formula.

1.4.2 DH-convention vs PoE formula - advantages and disadvanteges

In this section we try to elaborate the differences between the DH convention and the PoE formula. Both methods differ in several points and have advantages and disadvantages. The DH convention on the one hand has fixed rules to attach frames to joints. This might sometimes seem to be tedious. If all frames are attached properly, one can easily find the homogeneous transformations between all consecutive frames T_1^0 to T_n^{n-1}. Using the DH convention, the homogeneous transformations is established by using the minimal number of parameters to describe joint movements. The minimal number of parameters in the DH convention are $a_i, d_i, \alpha_i, \vartheta i$. These DH parameters can be used to build the homogeneous transformations T_i. Multiplying all T_i returns the Transformation T from base to end effector. The forward kinematics is then solved. The easy establishment of the transformation matrix with the minimal number of parameters is the biggest advantage of the DH convention. Still there exist different DH conventions for attaching the frames to the links. This might sometimes be confusing. Another disadvantage is that the DH-parameters can become ill-conditioned. Especially in special cases where consecutive joint axes are parallel or do intersect, little changes in the robot geometry can make huge changes in the DH-parameters. Errors in manufacturing, CAD models or in other areas can therefore lead to bigger problems.

In contrast to that, the PoE does not expect to attach frames at each joint. After deciding for a base and end effector frame and establishing the M matrix, which defines the homogeneous transformation from base to end effector in the robot's home position, one can use the PoE formula to solve the forward kinematics. There is a smaller effort when attaching frames to the robot and not a such a problem of ill-conditioned parameters. The interpretation of the joints as screw motion might be more intuitive. Besides that the PoE formula does not differ when using it for prismatic and rotational joints. In the DH convention, the parameter d_i for prismatic joints and the parameter ϑi for rotational joints are variable. In these cases those DH-parameters are not solely determined by the geometry of the link. This problem does not appear in the PoE formula where prismatic and rotational joints are treated equally. Still, in the PoE-formula it might be more effort to establish the screw axis and applying the matrix exponential. When using a computer, this disadvantage might be irrelevant. Besides that the PoE formula does not work with a minimal number of parameters.

Task 2

Forward Kinematics Example

In this task we apply different forward kinematics methods on the multi-purpose robot arm KUKA KR6 R900 sixx "Agilus", which has 6 degrees of freedom. We will solve the position of the end effector of this robot using the Denavit-Hartenberg convention and the Product-of-Exponential convention.

2.1 Using the Denavit-Hartenberg convention

2.1.1 Introduction

First, let us introduce some naming conventions. We call A_1 to A_6 the angles at each joint. The lengths L_1, L_2, L_3, L_4, H_1 and H_3 are constants related to the robot's geometry. Figure 2.1 represents the robot at its resting position, i.e. when all joint angles are zero.

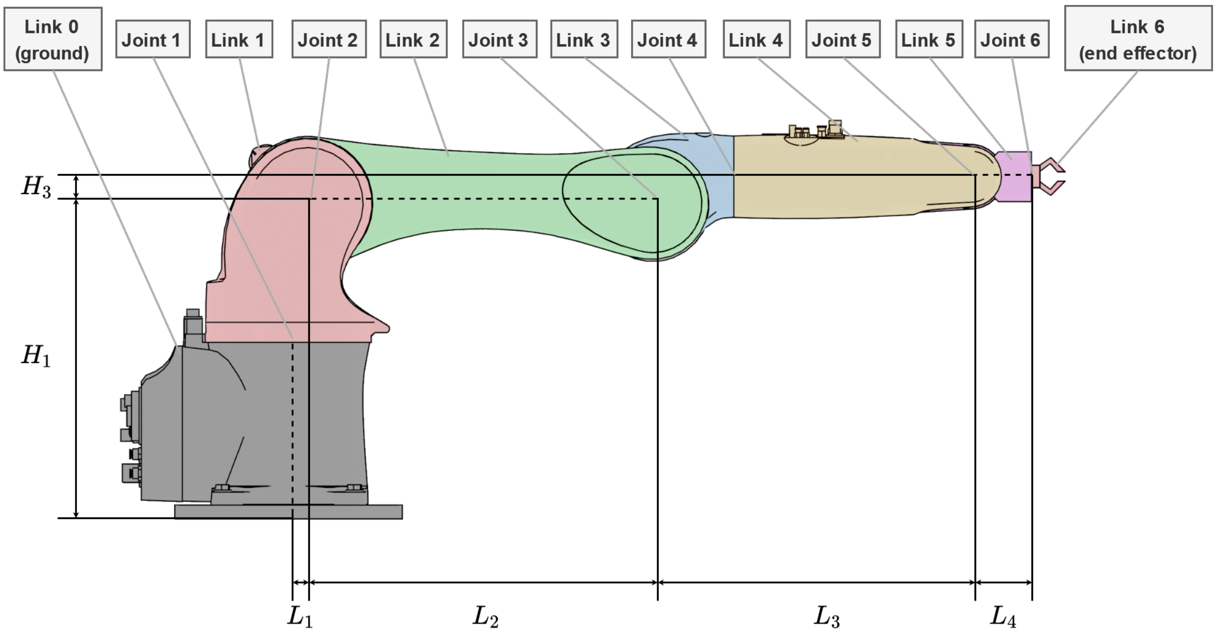

Figure 2.1: Robot dimensions and part names

The actual dimensions of the robot are $L_1 = 25\,\text{mm}$, $L_2 = 455\,\text{mm}$, $L_3 = 420\,\text{mm}$, $L_4 = 80\,\text{mm}$, $H_1 = 400\,\text{mm}$, $H_3 = 35\,\text{mm}$

2.1.2 Finding the parameters

Using the original DH method described in 1.1, we attach a frame of to each link as on figure 2.2 and we find the set of DH parameters, summarized in table 2.1. Each quadruplet of values d_i, θ_i, a_i, α_i describes a transformation that goes from frame $i-1$ to frame i.

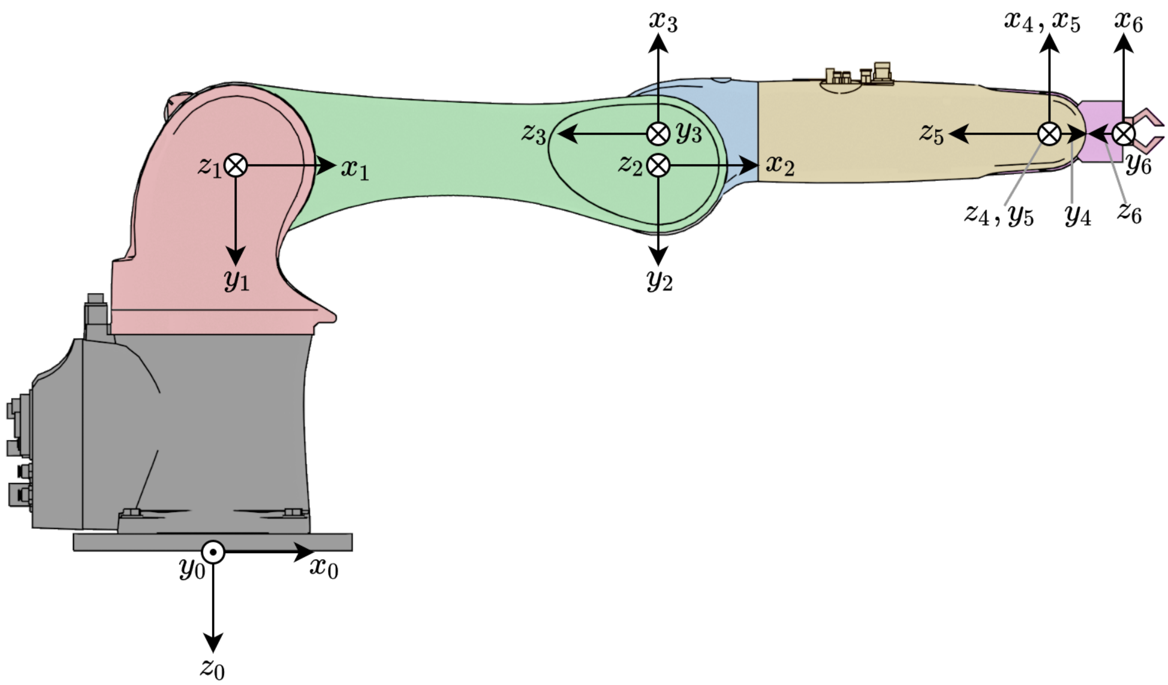

Figure 2.2: Placement of each frame according to the original DH convention

i	d_i	θ_i	a_i	α_i
1	$-H_1$	A_1	L_1	$\pi/2$
2	0	A_2	L_2	0
3	0	$A_3 - \pi/2$	H_3	$\pi/2$
4	$-L_3$	A_4	0	$-\pi/2$
5	0	A_5	0	$\pi/2$
6	$-L_4$	A_6	0	0

Table 2.1: Agilus DH parameters.

Note that the parameters θ_i, that correspond to the rotations around the z_{i-1} axes, are always the same as the joint angles A_i except for joint 3 where it is offset by $\pi/2$.

2.2 Using the Product-of-Exponential convention

2.2.1 Resting position of the end effector

First, we reuse the global frame 0 and the end-effector frame 6 defined on figure 2.2. The intermediate frames 1 to 5 are not used in the PoE convention. The resting position of the end effector M, which is the transformation matrix T_{06} when the joint angles are zero, is equal to:

$$M = \begin{bmatrix} 0 & 0 & -1 & L_1 + L_2 + L_3 + L_4 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & H_1 + H_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.2.2 Joint screws

A joint screw can be seen as a field of vectors that swirls around said joint, as illustrated on figure 2.3. $\mathcal{S}_i$ represents the value of this vector field at the base frame and $\mathcal{B}_i$ represents its value at the end effector frame, but they both represent the same screw of joint i.

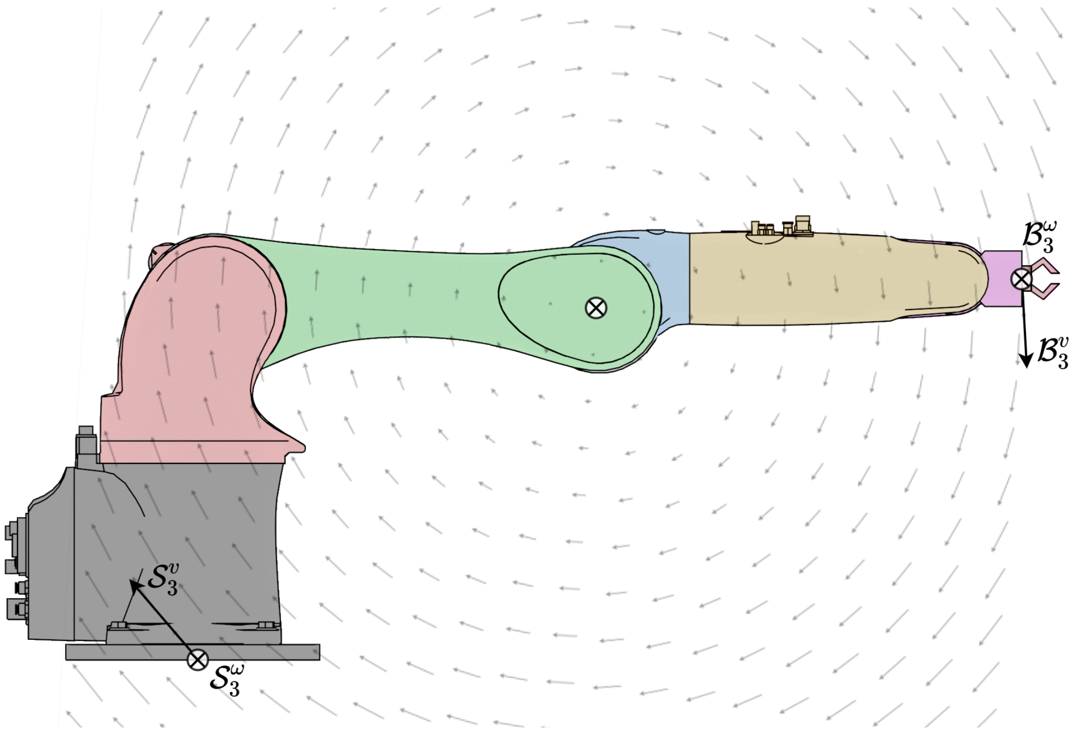

Figure 2.3: Representation of a screw as a field of vectors

i	$\mathcal{S}_i$	$\mathcal{B}_i$
1	$\begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}$	$\begin{bmatrix}-1\\0\\0\\0\\-L_1-L_2-L_3-L_4\\0\end{bmatrix}$
2	$\begin{bmatrix}0\\-1\\0\\-H_1\\0\\-L_1\end{bmatrix}$	$\begin{bmatrix}0\\1\\0\\-L_2-L_3-L_4\\0\\-H_3\end{bmatrix}$
3	$\begin{bmatrix}0\\-1\\0\\-H_1\\0\\-L_1-L_2\end{bmatrix}$	$\begin{bmatrix}0\\1\\0\\-L_3-L_4\\0\\-H_3\end{bmatrix}$
4	$\begin{bmatrix}-1\\0\\0\\0\\H_1+H_3\\0\end{bmatrix}$	$\begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}$
5	$\begin{bmatrix}0\\-1\\0\\-H_1-H_3\\0\\-L_1-L_2-L_3\end{bmatrix}$	$\begin{bmatrix}0\\1\\0\\-L_4\\0\\0\end{bmatrix}$
6	$\begin{bmatrix}-1\\0\\0\\0\\H_1+H_3\\0\end{bmatrix}$	$\begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}$

Table 2.2: Agilus PoE parameters.

2.3 Visualization

The figure 2.4 represents the frames attached to each link of the robot as described by the DH parameters that are summarized in table 2.1. The figure 2.5 shows the screw axes associated to each joint in table 2.2. By definition of the DH convention, the screw axes of joint i is the same the z_{i-1} axis.

The following figures have been generated with a python code that solves the forwards kinematics with both the DH and Poe methods. It does a bit more than calculating the end-effector position, as it must compute the local frames and transformations of each link for rendering purposes.

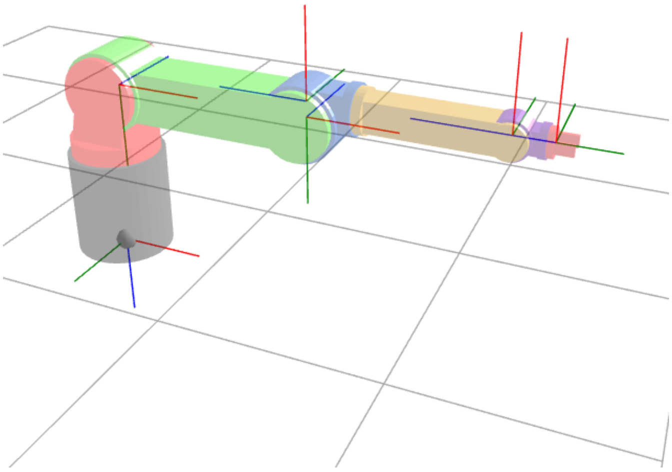

Figure 2.4: DH frames at resting position

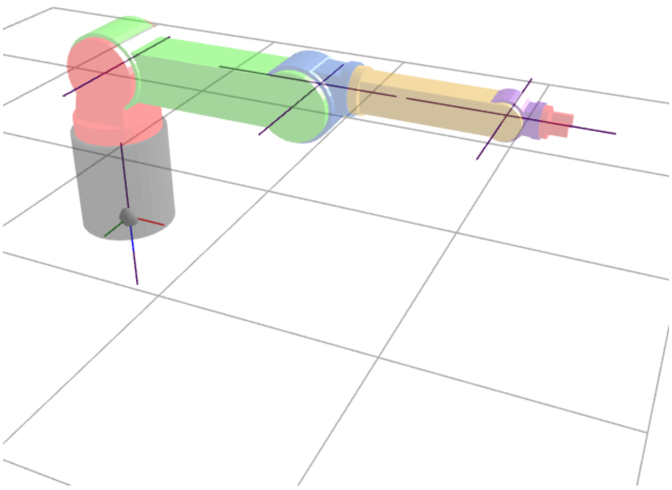

Figure 2.5: Screw axes at resting position

2.4 Comparing the DH and PoE conventions

The two methods of computing the end effector position, Product-of-Exponential and Denavit-Hartenberg, are strictly equivalent. They are visually the same on arbitrary configurations as shown on figure 2.6, and numerically equivalent to an absolute margin of error of 10^{-5}. Small variations in the numerical value of the end effector position are du to the different order of the floating point operations in each method.

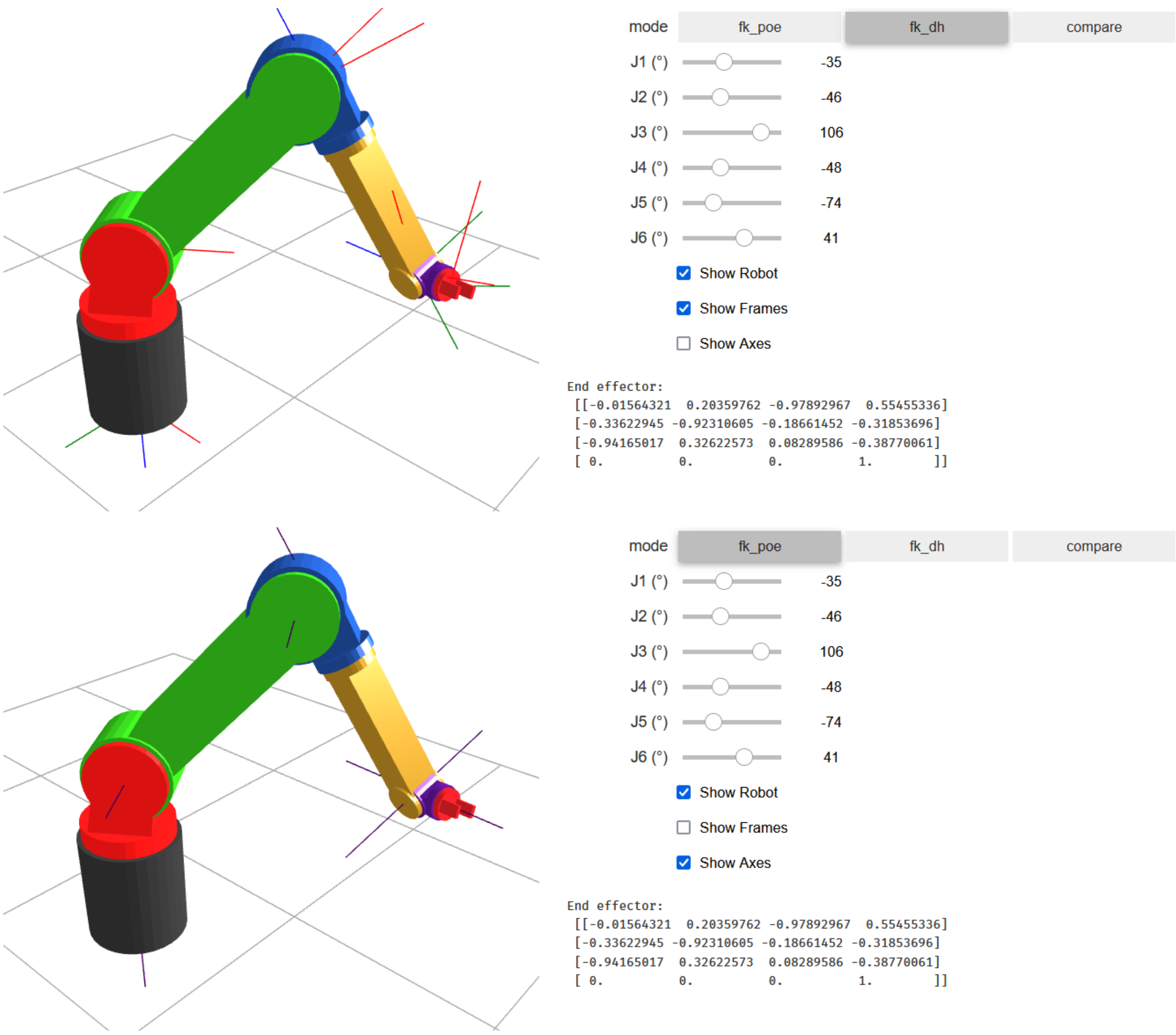

Figure 2.6: Robot's position computed numerically with DH (upper) and PoE (lower)

Task 3

Inverse Kinematics

3.1 Inverse Kinematics

The inverse kinematics problem is to determine one or more configurations (the set of joint positions) of an open-chain manipulator to achieve a desired end-effector pose [2]. This problem can be stated as, given a homogeneous transform X, find a set of θ such that $T(\theta) = X$. In contrast to forward kinematics, there may be zero or multiple solutions to the problem. In this section, we will present an overview of the two methods, analytical and numerical, for solving inverse kinematics problems, discuss the advantages and disadvantages of both approaches, and develop an analytical solution for the inverse kinematics of the Agilus robot.

3.1.1 Analytical Method

The analytical approach seeks to mathematically invert the equation $T(\theta) = X$ into a closed-form expression through symbolic manipulations [3]. $T(\theta)$ is the forward kinematics of the manipulator described by a homogeneous transformation matrix of each joint [4]. There are multiple advantages to this approach, one of which is the possibility to find all solutions to the problem, or determine whether no solution exists [3]. In addition, once the equations for all joint angles/parameters are derived, solutions for them are fast to compute. The inverse kinematics of an open-chain manipulator with n joints can be solved either algebraically to find the significant equations needed containing the unknowns, or by using geometry [1]. In this paper, we will only discuss how to find the equations geometrically. This can be achieved by decoupling the position and orientation inverse kinematics.

Position Inverse Kinematics

The position inverse kinematics can be broken down into several plane geometry problems, for example in the xy and rz-plane [5]. Considering the 2R planar arm in figure 3.1, θ_1 and θ_2 can easily be found using simple trigonometry [2]. By inspection, it follows that

$$\theta_1 = \gamma - \alpha \text{ and } \theta_2 = \pi - \beta \qquad \text{(Elbow down)},$$

or

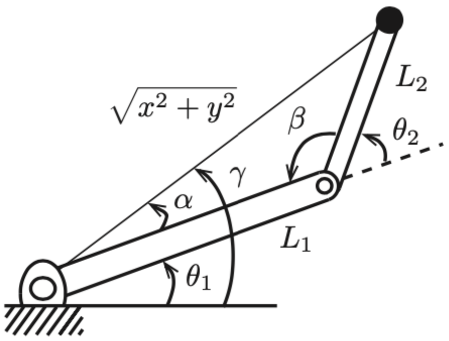

Figure 3.1: 2R Planar Arm

$$\theta_1 = \gamma + \alpha \text{ and } \theta_2 = \beta - \pi \quad \text{(Elbow up)}.$$

Also by inspection, $\gamma = \arctan(\frac{y}{x})$, however since arctan doesn't always return the angle in the correct quadrant and only in the range $(-\pi/2, \pi/2)$, we will use the function atan2(y,x) which returns angles in the range $(-\pi, \pi]$. Therefore,

$$\gamma = atan2(y, x).$$

The angles α and β can be found using the law of cosine,

$$c^2 = a^2 + b^2 - 2ab\cos C,$$

from which it follows that

$$C = \cos^{-1}(\frac{a^2 + b^2 - c^2}{2ab}) \tag{3.1}$$

If we put in β for C and a, b, and c for L_1, L_2, and $\sqrt{x^2+y^2}$ we have

$$\beta = cos^{-1}(\frac{L_1^2+L_2^2-x^2-y^2}{2L_1L_2}).$$

Doing the same for α results in

$$\alpha = cos^{-1}(\frac{x^2+y^2+L_1^2-L_2^2}{2L_1\sqrt{x^2+y^2}}).$$

Orientation Inverse Kinematics

Finding the orientation inverse kinematics is more straightforward. Lets turn the 2R planar arm in figure 3.1 into a 3D robot with four degrees of freedom: One in the shoulder (θ_1), one in the elbow (θ_2), and two in the wrist (θ_3 and θ_4). The PoE forward kinematics is then given by

$$X = e^{[S_1]\theta_1}e^{[S_2]\theta_2}e^{[S_3]\theta_3}e^{[S_4]\theta_4}M, \tag{3.2}$$

where X is the desired pose of the wrist center. Having already found θ_1 and θ_2, this can be written as

$$e^{-[S_2]\theta_2}e^{-[S_1]\theta_1}XM^{-1} = e^{[S_3]\theta_3}e^{[S_4]\theta_4}, \tag{3.3}$$

where the left side is known.

Although this example is quite simple, the principle is the same for more complex problems, as long as the manipulator isn't kinematically redundant, in which case there are infinite solutions. Additionally, with inverse kinematics in 3D, as seen in our solution for the Agilus robot, the equations may be challenging and time-consuming to find, while also being specific to a robot with the exact same kinematic structure [3].

3.1.2 Numerical Method

The numerical approach can be useful when there is no analytical solution [2]. This is done by iterating towards the solution from an initial guess. However, numerical methods do not always converge if this initial guess is too far from the solution. The iterations might oscillate or even diverge towards infinity [3]. If there are kinematical errors in the analytical solution for a robot, the configuration which is obtained analytically can be used as the initial guess to improve accuracy [2]. Also, in kinematically redundant robots, numerical methods are needed to find the closest optimal solution. One of these is the Newton-Raphson method for nonlinear root-finding, which is given by

$$\theta^{i+1} = \theta^i - \tfrac{g(\theta^i)}{g'(\theta^i)}.$$

In inverse kinematics g(θ) is defined by $g(\theta) = x_d - f(\theta_d)$ which results in

$$\theta^{i+1} = \theta^i + J^{+}(\theta^i)V$$

where $J^{+}(\theta^i)$ is the pseudoinverse of the Jacobian evaluated at θ^i, and V is the twist that takes $T(\theta^i)$ to T_{sd} in one second.

3.2 Analytical Solution: KUKA KR6 R900 sixx

The forward kinematics of the Agilus robot is given by

$$T(\theta) = e^{[S_1]\theta_1}e^{[S_2]\theta_2}e^{[S_3]\theta_3}e^{[S_4]\theta_4}e^{[S_5]\theta_5}e^{[S_6]\theta_6}M.$$

By decoupling the position and orientation as in 3.3, this turns into

$$e^{-[S_3]\theta_3}e^{-[S_2]\theta_2}e^{-[S_1]\theta_1}XM^{-1} = e^{[S_4]\theta_4}e^{[S_5]\theta_5}e^{[S_6]\theta_6}. \tag{3.4}$$

Firstly, we will find θ_1, θ_2, and θ_3.

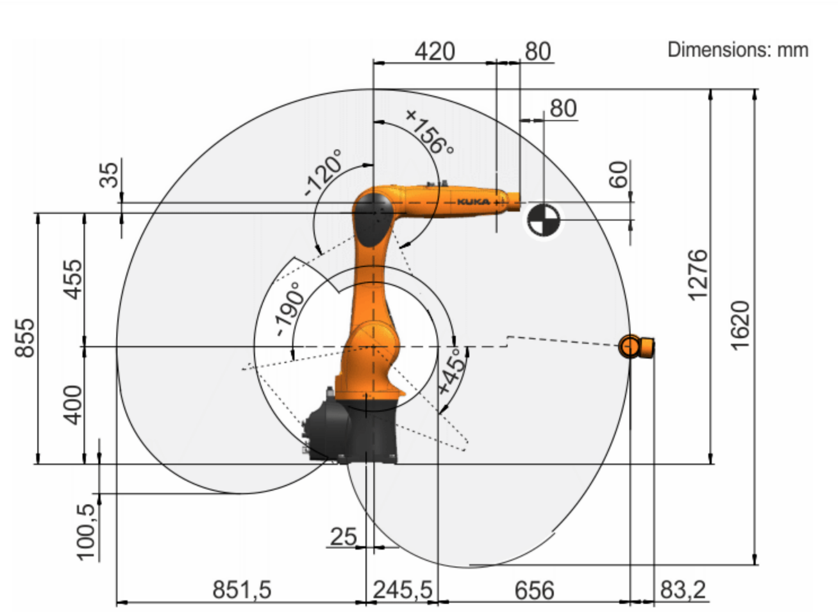

Figure 3.2: KUKA KR6 R900 sixx

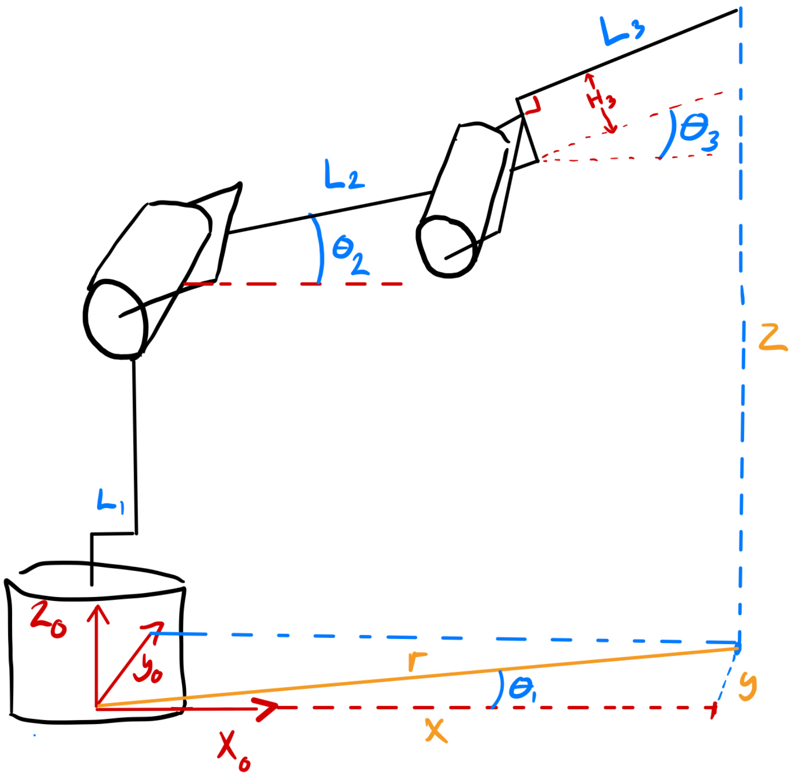

Figure 3.3: 3D representation of the first three joints

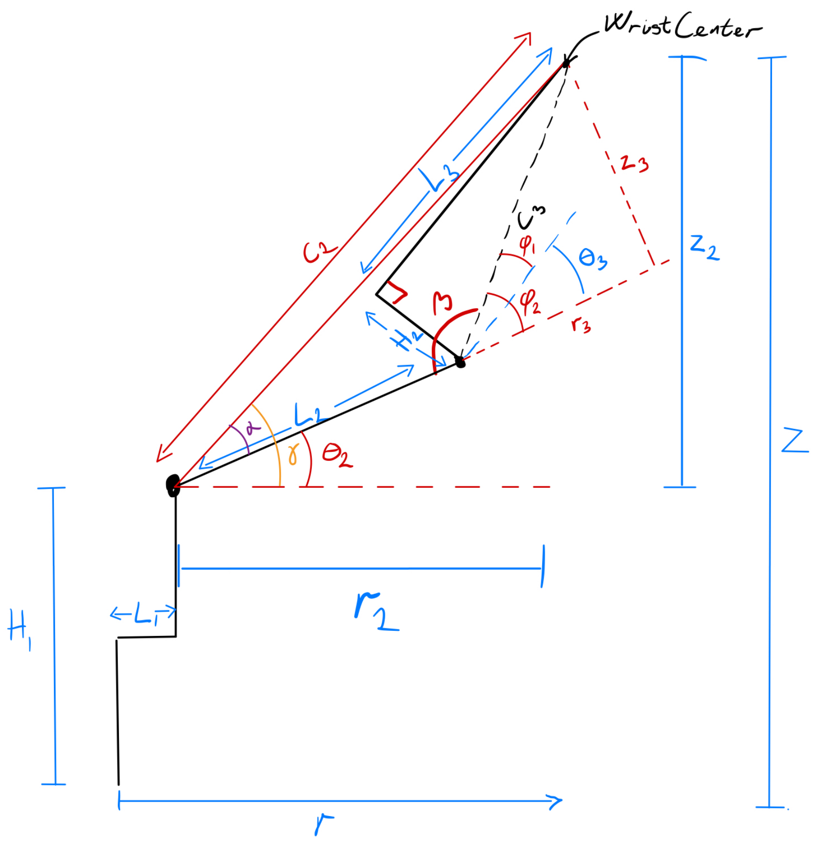

Figure 3.4: Agilus: Elbow Down

3.2.1 Position Inverse Kinematics

The inverse kinematics of the the Agilus robot is somewhat more complex than the 2R planar arm example because of the two offsets L_1 and H_3 as seen in figure 3.3. By inspection,

$$\theta_1 = atan2(y, x)$$

θ_2 and θ_3 have both more than one solution: Elbow up and elbow down. For the elbow down solution in figure 3.4, it can be seen by inspection that

$$\theta_2 = \gamma - \alpha \tag{3.5}$$

$$\theta_3 = \phi_2 - \phi_1 \tag{3.6}$$

where

$$\phi_1 = \tan^{-1}(\frac{H_3}{L_3}) \tag{3.7}$$

$$\phi_2 = \pi - \beta. \tag{3.8}$$

β can be found using equation 3.1:

$$\beta = \cos^{-1}\left(\frac{L_2^2 + C_3^2 - C_2^2}{2L_2C_3}\right) \tag{3.9}$$

If we put 3.7 and 3.8 into equation 3.6 we get

$$\theta_3 = \pi - \cos^{-1}\left(\frac{L_2^2 + C_3^2 - C_2^2}{2L_2C_3}\right) - \tan^{-1}\left(\frac{H_3}{L_3}\right), \tag{3.10}$$

where

$$\begin{aligned}
C_3 &= \sqrt{L_3^2 + H_3^2} \\
C_2 &= \sqrt{r_2^2 + z_2^2} \\
&= \sqrt{(r - L_1)^2 + (z - H_1)^2}. \\
&= \sqrt{(\sqrt{x^2 + y^2} - L_1)^2 + (z - H_1)^2}.
\end{aligned}$$

Note that x, y, and z corresponds to the wrist-center, which is not given. However, the end-effector frame is given as

$$X = \begin{bmatrix} r_{11} & r_{12} & r_{13} & Px \\ r_{21} & r_{22} & r_{23} & Py \\ r_{31} & r_{32} & r_{33} & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Also, the wrist-center is translated length L_4 along the z-axis of the end-effector frame, therefore

$$\begin{aligned}
x &= Px + L_4 r_{13} \\
y &= Py + L_4 r_{23} \\
z &= -Pz - L_4 r_{33}
\end{aligned}$$

Next we need to find γ and α in 3.5. It can easily be seen that

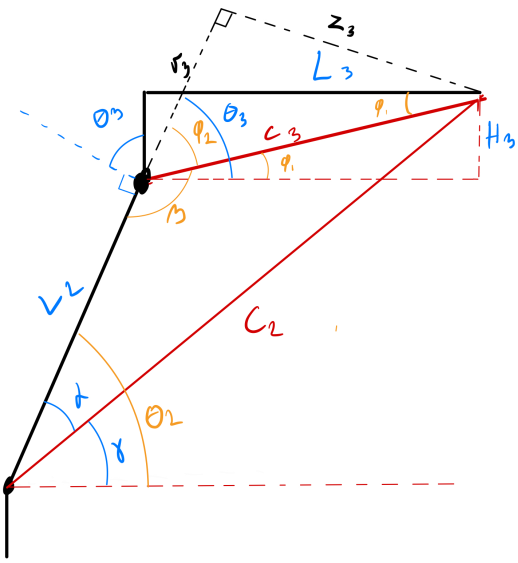

Figure 3.5: Agilus: Elbow Up

$$\gamma = atan2(z_2, r_2). \tag{3.11}$$

α can also be found using equation 3.1 or by using the properties of a right angle triangle, in which case

$$\alpha = atan2(z_3, L_2 + r_3) = atan2(C_3 \sin\phi_2, L_2 + C_3 \cos\phi_2) \tag{3.12}$$

If we put 3.11 and 3.12 into equation 3.5 we get

$$\theta_2 = atan2(z_2, r_2) - atan2(z_3, L_2 + r_3). \tag{3.13}$$

We can follow the same procedure to find θ_2 and θ_3 in the elbow up position.

By inspection of figure 3.5,

$$\theta_2 = \gamma + \alpha. \tag{3.14}$$

$$\theta_3 = -(\phi_1 + \phi_2). \tag{3.15}$$

By inserting 3.11 and 3.12 in equation 3.14, and 3.7 and 3.8 into equation 3.15:

$$\theta_2 = atan2(z_3, L_2 + r_3) + atan2(z_2, r_2), \tag{3.16}$$

$$\theta_3 = \cos^{-1}(\frac{L_2^2 + C_3^2 - C_2^2}{2L_2C_3}) - \tan^{-1}(\frac{H_3}{L_3}) - \pi. \tag{3.17}$$

3.2.2 Orientation Inverse Kinematics

With this the left side of equation 3.4 is known and the ω_i components of $\mathcal{S}_4$, $\mathcal{S}_5$, and $\mathcal{S}_6$ are

$$\begin{aligned}\omega_4 &= (-1, 0, 0),\\ \omega_5 &= (0, -1, 0),\\ \omega_6 &= (-1, 0, 0).\end{aligned}$$

Denoting the right side of equation 3.4 to SO(3) by R_w, the wrist joint angles can be determined as the solution to

$$\text{Rot}(-\hat{x}, \theta_4)Rot(-\hat{y}, \theta_5)Rot(-\hat{x}, \theta_6) = R_w,$$

which corresponds to the negative XYX Euler angles [2]. If we denote the left side of equation 3.4 to SO(3) by R_a, the equation can be simplified as

$$R_a = R_w$$

which is equivalent to

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = -\begin{bmatrix} c_5 & s_5s_6 & s_5c_6 \\ s_4s_5 & -s_4s_6c_5 + c_4c_6 & -s_4c_5c_6 - s_6c_4 \\ -s_5c_4 & s_4c_6 + s_6c_4c_5 & -s_4s_6 + c_4c_5c_6 \end{bmatrix}$$

The algorithm for solving this equation is given by [6]:

1. If $r_{11} \neq \pm 1$
   - $\theta_4 = -atan2(r_{21}, -r_{31})$
   - $\theta_5 = -cos^{-1}(r_{11})$
   - $\theta_6 = -atan2(r_{12}, r_{13})$
2. If $r_{11} = -1$
   - $\theta_4 = -atan2(-r_{23}, r_{22})$
   - $\theta_5 = -\pi$
   - $\theta_6 = 0$
3. If $r_{11} = 1$
   - $\theta_4 = atan2(-r_{23}, r_{22})$

- $\theta_5 = 0$
- $\theta_6 = 0$

It should be noted that the correct solution for the inverse kinematics in this algorithm is: $[\theta_1, -\theta_2, -\theta_3, \theta_4, \theta_5, \theta_6]$. The signs are flipped for θ_2 and θ_3, most likely because of an error in positive/negative rotation direction, but flipping them yields the correct solution.

3.3 Comparing Numerical and Analytical Solution

There seems to be some some discrepancy between the solutions of the analytical and numerical solver, and sometimes the numerical solution does not converge.

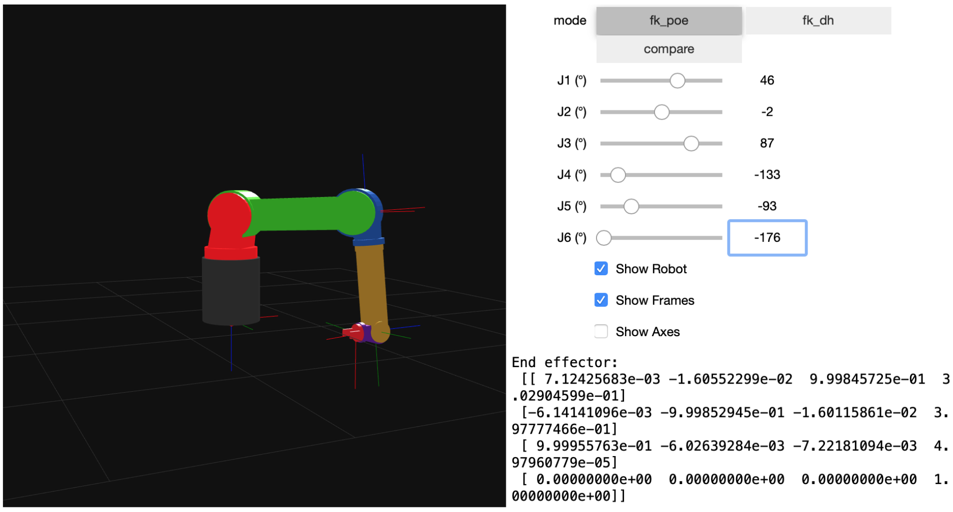

Figure 3.6: Analytical solution

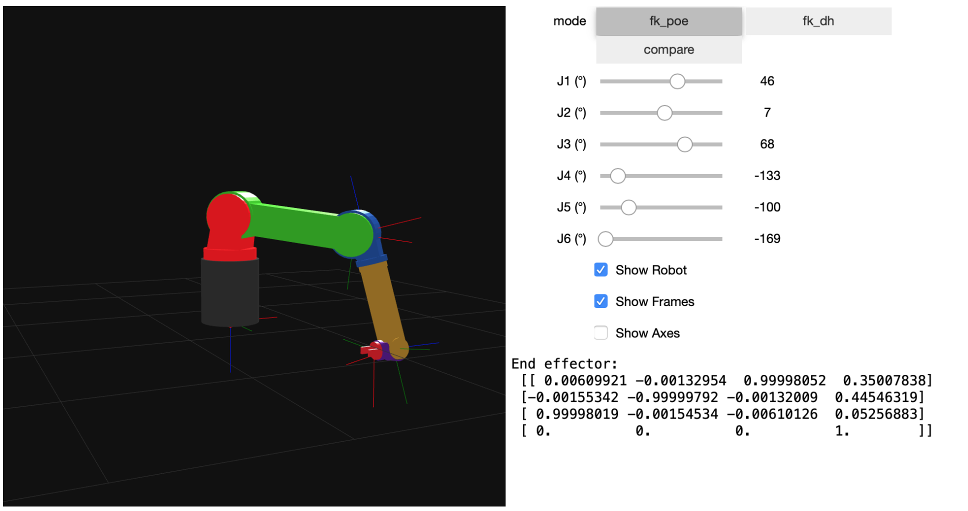

Figure 3.7: Numerical solution

Figure 3.6 and 3.7 show the difference in the configuration for the end-effector frame:

$$X = \begin{bmatrix} 0 & 0 & 1 & 0.3 \\ 0 & -1 & 0 & 0.4 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The other frames tested show both either completely different results or somewhat close, as in this case. These can be found and tested in the code attached to this report. However, testing the analytical solution with the forward kinematics gives the correct result. This may indicate that the numerical solver have been given the wrong input arguments.

3.4 Visualization

In the attached Jupyter Notebook file we have created a program which uses the forward and inverse kinematics algorithms to visualize the Agilus robot with an arbitrary end-effector frame. Figure 3.8 and 3.9 shows the elbow up and elbow down configuration for one of these frames.

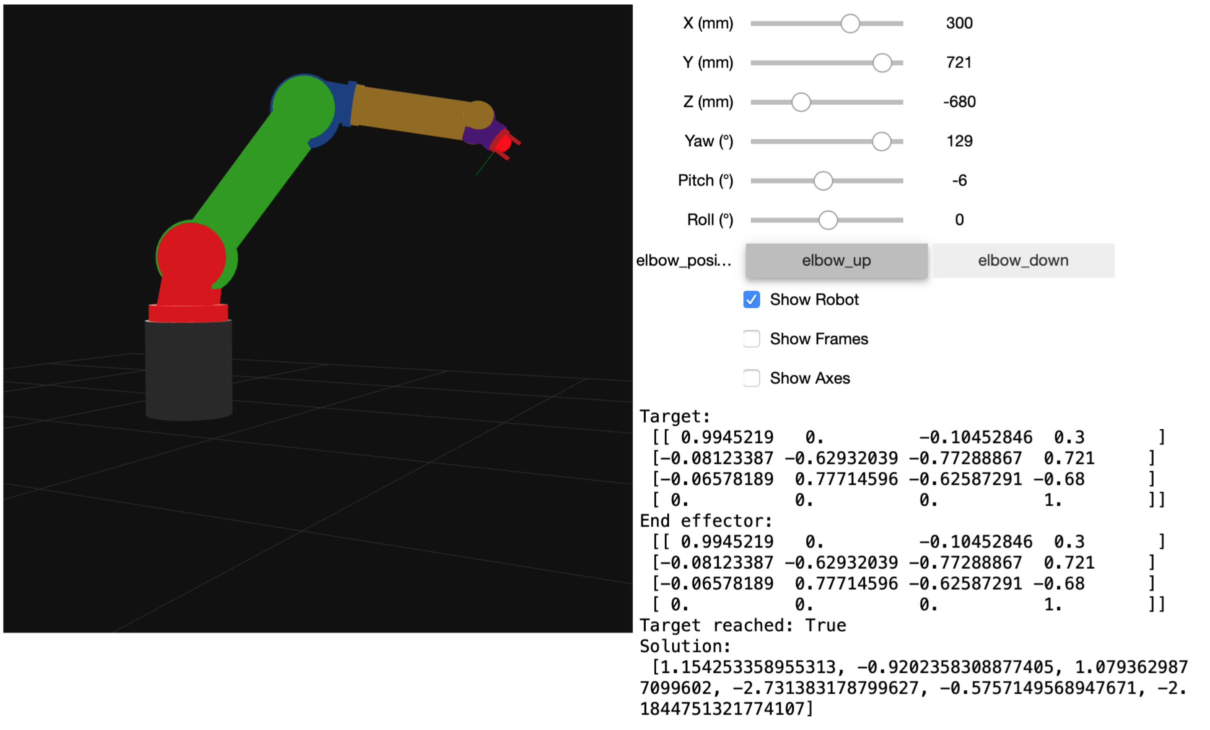

Figure 3.8: Agilus Visualization: Elbow Up

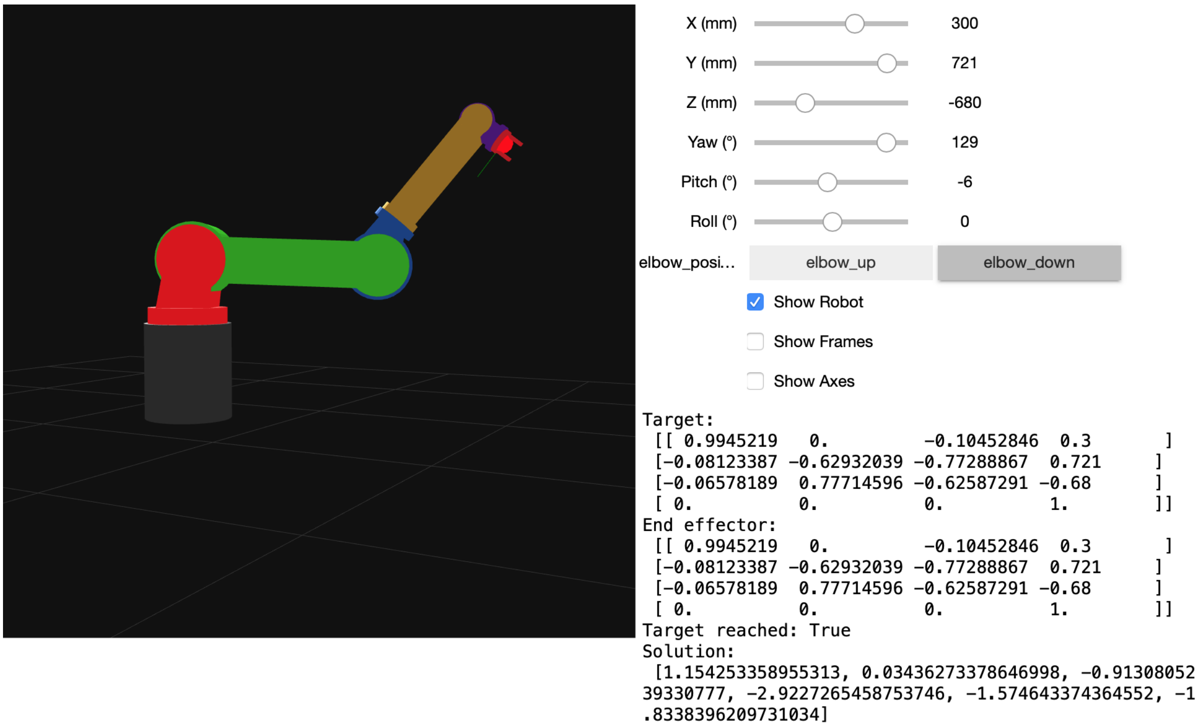

Figure 3.9: Agilus Visualization: Elbow Down

Task 4

Singularity Analysis

Two robot singularities were identified. In the following we will present the wrist and the shoulder singularity.

4.1 Wrist Singularity

A wrist singularity occurs when the axes of joints 4 and 6 are coincident. In this situation movement in these two joints have the same effect on the end effector pose. Therefore just five of the six joints of the robot influence the end effector pose. The effect of one joint is lost. The rank of the Jacobian matrix drops, which is an indicator for a robot singularity.

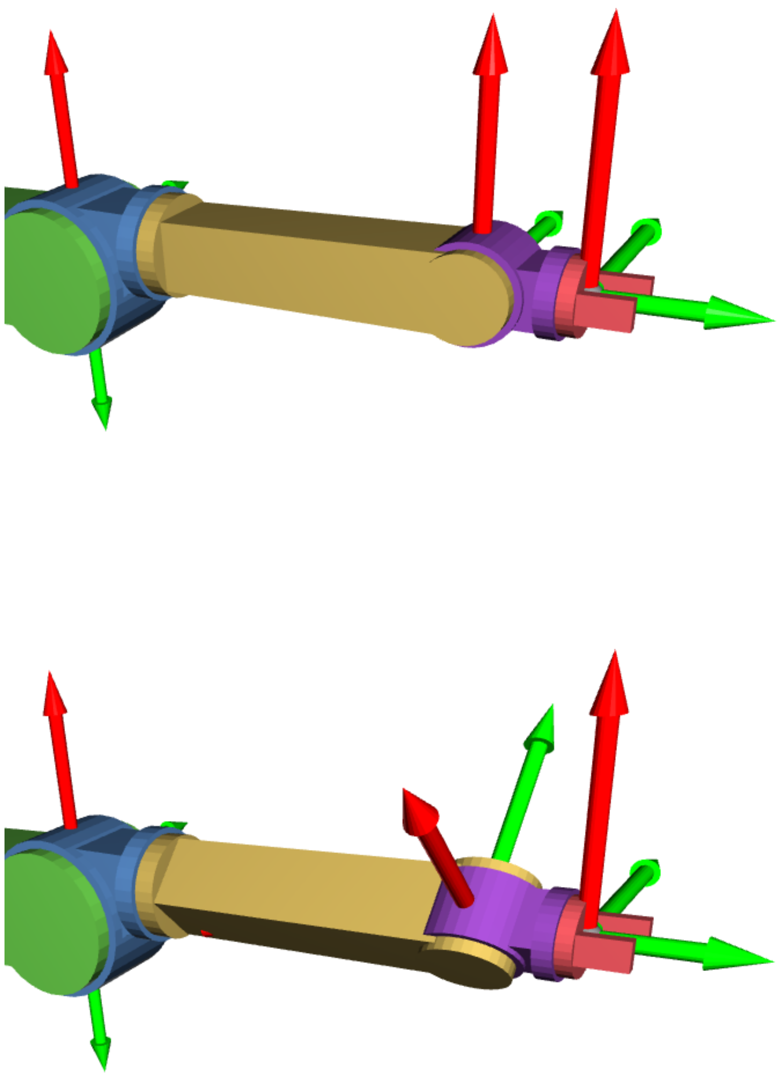

Figure 4.1: 6-axis robot wrist singularity

4.2 Shoulder Singulaitry

A shoulder singularity occurs when the wrist center lies in the plane passing through the axis of joint 1 and is parallel to axis of joint 2. In this case the robot is in an upright position. Movements around joint 1 can be counteracted by movements in joint 4, 5 and 6 such that the end effector pose is not affected by those. Therefore just five of the six joints of the robot influence the end effector pose. The effect of one joint is lost. The rank of the Jacobian matrix drops, which again indicates arobot singulrity.

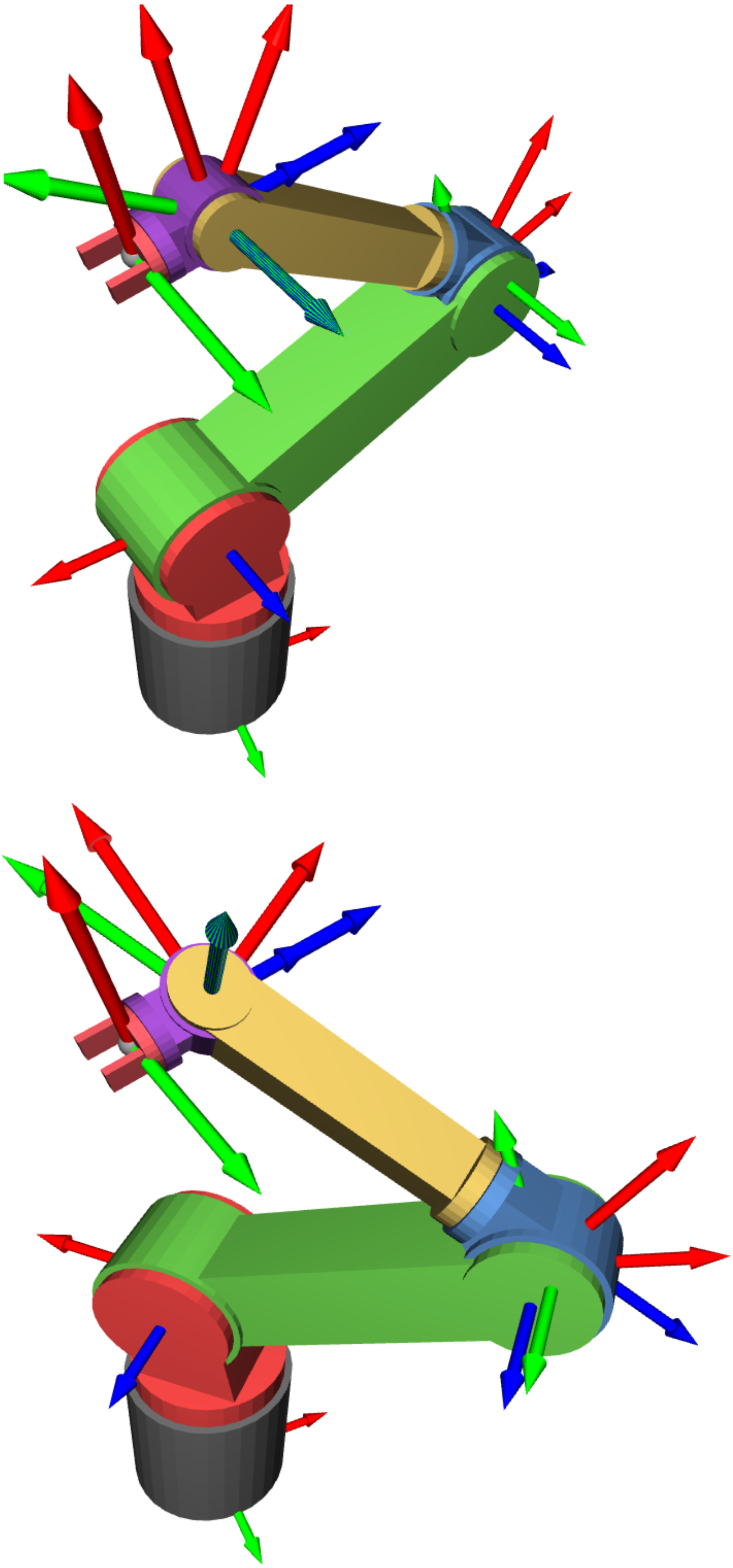

Figure 4.2: 6-axis robot shoulder singularity

References

[1] B. Siciliano, *Robotics: Modelling, Planning and Control*. Springer London, 2009, ISBN: 978-1-84628-642-1.

[2] F. C. P. Kevin M. Lynch, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017, ISBN: 978-1-107-15630-2.

[3] T. U. of Illinois, *Chapter 6. inverse kinematics*. [Online]. Available: http://motion.cs.illinois.edu/RoboticSystems/InverseKinematics.html.

[4] S. Kucuk and Z. Bingul, "Inverse kinematics solutions for industrial robot manipulators with offset wrists," *Applied Mathematical Modelling*, vol. 38, no. 7-8, pp. 1983–1999, 2014. DOI: 10.1016/j.apm.2013.10.014.

[5] S. Kucuk and Z. Bingul, "Robot kinematics: Forward and inverse kinematics," in *Industrial Robotics*, S. Cubero, Ed., IntechOpen, 2006, ch. 4. DOI: 10.5772/5015.

[6] D. Eberly, *Euler angle formulas*. [Online]. Available: https://www.geometrictools.com/Documentation/EulerAngles.pdf.