Direct and Inverse Kinematics of Serial Manipulators (Nyrio One 6-axis Robotic Arm)

ALMEIDA, João*
joaotiago99@gmail.com
IST id: 90119

ROSA, Daniel†
dgd.rosa@gmail.com
IST id: 90041

VIEGAS, Guilherme‡
ggmviegas@gmail.com
IST id: 90090

May 29, 2021

***Abstract*—This paper presents the development of a 6-axis rigid body kinematics. A rigid body in three dimensions has six degrees of freedom composed by three translatory DOF and three rotational DOF. The multiples revolute joints of the rigid body form a serial chain, which allows successive rotations about the *x, y and z* axes accomplishing a specific 3D position, this method is the so called direct kinematics. In addition, it is also designed the reverse procedure, inverse kinematics, that places the end of the kinematic chain (e.g. gripper) in an exact place with an specific orientation.**

***Keywords*—** Rigid Body, 6DOF, Serial Chain, Niryo One, Direct Kinematics, Inverse Kinematics

II. Introduction‡

Robots can be studied from different perspectives. Perhaps one of the most important parts in the study of a robot is understanding its geometrically possible motion without considering the forces involved. This is called *the robot kinematics*, often referred as the *geometry of motion*, and aims to provide a description of the spatial position of systems (and also their velocity and acceleration).

Within this subject, two forms of kinematics can be distinguished:

- The direct kinematics that enables to determine the robots' configuration in the work space, given a set of values/angles for that the robots' joints can move;
- The inverse kinematics that solves the inverse problem, with the goal of, given a work space configuration, determine the corresponding joint variables to move in order to achieve the desired position and/or orientation.

The two problems are solved using distinctive mathematical tools which, although explained in more detail later on, it can be said that for solving the direct kinematics problem the need to specify positions and orientations of many referentials is very important. Thereafter a composition of transformations between those specified referentials is computed. On the other hand, for the inverse kinematics both geometric and algebraic methods are used to solve specific problems that, altogether, resolve the more complex problem of the inverse kinematics.

III. Background/Concepts*

Behind the shield of the arm, the rigid body can be described as a series of mathematical calculations. Each joint varies from the previous by one transformation that can be expressed as a rotation plus a translation. In 3D coordinates (*X-Y-Z*), the T-matrix is a linear transformation mapping $\mathcal{R}^4$ to $\mathcal{R}^4$, so a basic knowledge about this mapping is required (see more in [1]). Another method to describe the multiple transformations between joints is using Denavit-Hartenberg parameters, although it is not used in this application, it might help understanding the construction of the skeleton. In addition, throughout the report it is mentioned the orientation of the rigid body, which is computed with the Euler angles and describes the orientation of the current frame according the world frame. This application uses the Z-Y-Z convention, however other conventions can be studied in [2]. Moving to the inverse kinematics process, there are mainly two threads to bear in mind. Firstly the properties of the rotation matrix

such as the determinant equal to 1 and the inverse matrix means a rotation by the symmetric angle (transpose). Secondly, a general knowledge in trigonometry involving the law of cosines and planar geometric (e.g. 2D double pendulum equations) is necessary.

Throughout the report, the scale of the images showed are in centimetres, even though the calculus of the position were made in millimetres.

IV. State of the art*

This paper explains, step by step, the kinematics of the robot alongside examples. Moreover this paper does not go into much detail, however it is a plus for whom wants to be familiar with 6DOF Robotic Arm movements. For a more detailed approach see [3].

V. Ease of use*

The implementation consists on two contrary processes. Firstly it is computed the position of the end effector and the Euler angles with the call function: *direct_kinematics(A:1x6 matrix)*, from the vector of angles of each joint. This step is super fast, since it is a "simple" series of matrix operations. On the other hand, the *inverse_kinematics(O:1x6 matrix)* function behaves as the inverse, since it transforms *X-Y-Z* end effector's coordinates and *Z-Y-Z* Euler angles into multiple solutions for the angles of the joints. This approach is slower once it compares and validates each possible solution after every step. The metrics are millimetres (mm) for the position and radians (rad) for the joint angles.

VI. Development*†‡

A. Direct Kinematics‡

1) Choice of the Coordinate System‡: As with many robotic manipulators, the robot described in this report is considered a **serial** manipulator. These types of manipulators are constituted by a set of rigid bodies, called links, connected by joints that enable different kinds of movements and that have different types of restrictions, and contrasting to parallel manipulators, there are no closed loops.

In the direct kinematics problem, where the goal is, given a set of values for the joint variables, compute the robot configuration (position + orientation), the solution for the end effector position is found through a series of multiplications between transformation matrices. For this, a crucial initial step is to define a set of referentials capable of defining the entire motion of the manipulator. Although possible to define those referentials according to multiple conventions, some of them enable not only an easier understanding of the robot movement but are also easier to compute the needed chain of transformations. For this specific project, it was defined a direct coordinate system for each joint, with the particularity that all of the systems had the same orientation, as seen in Figure 1, where (for the reader's view) every *x axis* was pointing left, every *z axis* was pointing upwards and every *y axis* pointing towards the "viewer", orthogonal to the *x and y axes*, as in the referential in Figure 3.

With this the only need was to take care of the rotation that each joint made and the change in position for each and every frame.

Knowing the manipulator specifications it could then be built the matrix transformations for each consecutive pair of frames, noting the choice of using millimetres for translations. For the pair between the world frame and frame 1 (the so called link 0), because it only rotates through the *z axis*, it originates

$$ {}^{0}_{1}P = \begin{bmatrix} 0 \\ 0 \\ 103 \end{bmatrix} \quad {}^{0}_{1}R_z = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{1} $$

leading to the transformation matrix

$$ {}^{0}_{1}T = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 103 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2} $$

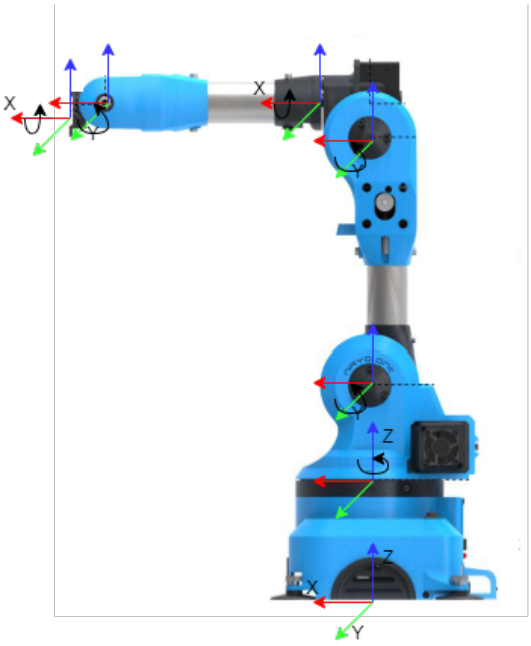


Fig. 1. Coordinate Frames for the 6 joints and World Frame

Then, for the pair between frame 1 and frame 2 (so link 1 that connects joint 1 and joint 2), because there's only a rotation through the *y axis*, it originates

$$ {}^{1}_{2}P = \begin{bmatrix} 0 \\ 0 \\ 80 \end{bmatrix} \quad {}^{1}_{2}R_y = \begin{bmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) \\ 0 & 1 & 0 \\ -\sin(\theta_2) & 0 & \cos(\theta_2) \end{bmatrix}, \tag{3} $$

leading to the transformation matrix

$$ {}^{1}_{2}T = \begin{bmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta_2) & 0 & \cos(\theta_2) & 80 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4} $$

For the next pair, between frame 2 and frame 3 (link 2 connecting joints 2 and 3) there's the same rotation around the *y axis* as previously described, originating

$$ {}^{2}_{3}P = \begin{bmatrix} 0 \\ 0 \\ 210 \end{bmatrix} \quad {}^{2}_{3}R_y = \begin{bmatrix} \cos(\theta_3) & 0 & \sin(\theta_3) \\ 0 & 1 & 0 \\ -\sin(\theta_3) & 0 & \cos(\theta_3) \end{bmatrix}, \tag{5} $$

leading to the transformation matrix

$$ {}^{2}_{3}T = \begin{bmatrix} \cos(\theta_3) & 0 & \sin(\theta_3) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta_3) & 0 & \cos(\theta_3) & 210 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6} $$

The next pair, where frame 4 is positioned in fourth joint, a rotation around *x axis* is formed, leading to

$$ {}^{3}_{4}P = \begin{bmatrix} 41.5 \\ 0 \\ 30 \end{bmatrix} \quad {}^{3}_{4}R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_4) & -\sin(\theta_4) \\ 0 & \sin(\theta_4) & \cos(\theta_4) \end{bmatrix}, \tag{7} $$

leading to the transformation matrix

$$ {}^{3}_{4}T = \begin{bmatrix} 1 & 0 & 0 & 41.5 \\ 0 & \cos(\theta_4) & -\sin(\theta_4) & 0 \\ 0 & \sin(\theta_4) & \cos(\theta_4) & 30 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{8} $$

Then another rotation around *y axis* for the transformation between frame 4 and 5 (link 4) and the rotation that happens in fifth joint, leading to

$$ {}^{4}_{5}P = \begin{bmatrix}180\\0\\0\end{bmatrix} \; {}^{4}_{5}R_y = \begin{bmatrix}\cos(\theta_5) & 0 & \sin(\theta_5)\\ 0 & 1 & 0\\ -\sin(\theta_5) & 0 & \cos(\theta_5)\end{bmatrix}, \tag{9} $$

leading to the transformation matrix

$$ {}^{4}_{5}T = \begin{bmatrix}\cos(\theta_5) & 0 & \sin(\theta_5) & 180\\ 0 & 1 & 0 & 0\\ -\sin(\theta_5) & 0 & \cos(\theta_5) & 0\\ 0 & 0 & 0 & 1\end{bmatrix}. \tag{10} $$

For the last transformation, there is the connection between frame 5 and 6 and the rotation around the *x axis* in joint 6 that leads to

$$ {}^{5}_{6}P = \begin{bmatrix}23.7\\0\\-5.5\end{bmatrix} \; {}^{5}_{6}R_X = \begin{bmatrix}1 & 0 & 0\\ 0 & \cos(\theta_6) & -\sin(\theta_6)\\ 0 & \sin(\theta_6) & \cos(\theta_6)\end{bmatrix}, \tag{11} $$

leading to the transformation matrix

$$ {}^{5}_{6}T = \begin{bmatrix}1 & 0 & 0 & 23.7\\ 0 & \cos(\theta_6) & -\sin(\theta_6) & 0\\ 0 & \sin(\theta_6) & \cos(\theta_6) & -5.5\\ 0 & 0 & 0 & 1\end{bmatrix}. \tag{12} $$

Now that all the frames and transformations are defined, the direct kinematics problem may be divided into two parts, one for finding the end effector final position using a serial composition of the previously defined matrices and the other for finding the end effector orientation, according to the *Z-Y-Z Euler Angles* convention, making use of some geometric properties.

2) Solving the Position[‡]*:* Starting by the first problem of finding the end effector final's position in space, according to the defined world frame, it can be said that this position, in homogeneous coordinates, is the last column of the matrix formed by the multiplication series of the previously defined matrices, as in

$$ {}^{0}_{6}T = {}^{0}_{1}T *^{1}_{2}T *^{2}_{3}T *^{3}_{4}T *^{4}_{5}T *^{5}_{6}T. \tag{13} $$

3) Solving the Orientation[‡]*:* For the second part of the direct kinematics problem, which is finding the end effector orientation, according to a *Z-Y-Z convention* using Euler angles, it's started by making a correlation between the previously defined rotation matrix inside (13) between frame 6 and the world frame, ${}^{0}_{6}R$, and the generic rotation matrix defined by the three Euler angles, $\alpha, \beta \; and \; \gamma$

$$ \begin{bmatrix} c_\alpha c_\beta c_\gamma - s_\alpha s_\gamma & -c_\alpha c_\beta s_\gamma - s_\alpha c_\gamma & c_\alpha s_\beta \\ s_\alpha c_\beta c_\gamma + c_\alpha s_\gamma & -s_\alpha c_\beta s_\gamma + c_\alpha c_\gamma & s_\alpha s_\beta \\ -s_\beta c_\gamma & s_\beta s_\gamma & c_\beta \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{12} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33}. \end{bmatrix} \tag{14} $$

Therefore, from inspection and adequate choice of the equations to solve it is obtained

$$ \begin{cases} \beta = \arctan2(\sqrt{r_{13}^2 + r_{23}^2}, r_{33}) \\ \gamma = \arctan2(\frac{r_{32}}{s_\beta}, \frac{-r_{31}}{s_\beta}), \quad if \; s_\beta \neq 0 \\ \alpha = \arctan2(\frac{r_{23}}{s_\beta}, \frac{r_{13}}{s_\beta}), \quad if \; s_\beta \neq 0. \end{cases} \tag{15} $$

First, the β angle is computed because both the other angles depend on its value. If $s_\beta \neq 0$ then solving the other two angles is easy by applying (15). If $s_\beta = 0$ then $c_\beta = \pm 1$ which leads to $\alpha \pm \gamma = atan2(r_{21}, r_{11})$ (other rotation matrix elements could be chosen) and so it is needed to disambiguate their values, solved by stating

$$ \begin{cases} \gamma = 0 \\ \alpha = c_\beta \cdot atan2(r_{21}, r_{11}). \end{cases} \tag{16} $$

For the cases where β was $\pm\pi/2$ there was the need in *Matlab* to define the values for α and γ by

$$ \begin{cases} \alpha = 0 \\ \gamma = \arctan2(r_{21}, r_{11}), \quad if \; \beta = \frac{\pi}{2} \\ \gamma = -\arctan2(r_{21}, r_{11}), \quad if \; \beta = -\frac{\pi}{2}. \end{cases} \tag{17} $$

With this, a final output is given in vector form with the first three elements being the final end effector position, with respect to the world frame, given in millimetre, and the last three elements being the end effector orientation, according to the *Z-Y-Z Euler angles* convention, in radians.

B. Inverse Kinematics*†

1) Translatory and Rotational DOF[*]*:* The inverse kinematics can be divided into two groups of links: translation and orientation of the end effector. Considering rotations about the *x, y, and z* axes, it is evident that there are three rotational degrees of freedom. The other 3 DOF correspond to the translation of the rigid body. The handicap of this division is that the group of orientation also has influence on the translation of the end effector.

Due to the fact that every rotation affects the orientation of the rigid body and since the range is proportional to the length of the joints therefore to the weight not only of the links but also the motors, it is defined the translation group of links closer to the base, composed by the three first joints. Hence, the orientation group involves the joints 4, 5 and 6, and make the bridge between the third orientation and the final joint (end effector).

Consequently, it is defined a point as closest to the end effector as possible without being affected by the last 3 rotations. Accordingly to the skeleton of the rigid body, the fourth joint rotates according *x* axis corresponding to the axis where the fourth link lies, so any rotation will only be evident after the fifth joint (because the rotation axis is different). Moreover, according to the rigid body in analysis, the second and third joints rotates according the *y* axis, thus, forming a plan $(xy, z) \in \mathcal{R}^2$ which rotates through the first joint (z axis), making a 3D translation from the first joint to the fifth.

Finally the 5.5mm offset on the sixth joint in the *z* direction, imply that the last three joints do not intersect at one point, thus creating infinity admissible solutions to define the end effector. Therefore, in order to solve the inverse kinematics easily it is not taken into account the 5.5mm length, so the algorithm can generally have up to eight different solutions. Furthermore, a six-degree-of-freedom robot arm with this particularity is referred to as wrist-partitioned or as having inline wrists. At the end, it is considered the 5.5mm length and the problem is iterative solved for multiples values of the angle of rotation at the sixth joint. To conclude, the real solutions are obtained from the two combinations of the trunk (joint 1) rotation, from the two combinations of the planar chain and finally for both translatory combinations.

2) Position of the Fifth Joint[*]*:* The input of the inverse kinematics are the *X-Y-Z* position and *Z-Y-Z* Euler angles of the end effector. Additionally, in order to compute a middle joint position, the transformation from the world frame to the end effector can be split in the selected joint. In other words, the view of the fifth joint in the reference frame is equivalent to compute the end effector transformation and then reverse one step. This is illustrated in (18).

$$ {}^{0}_{5}P = {}^{0}_{6}P - {}^{0}_{5}R \, {}^{5}_{6}P \tag{18} $$

The same reasoning is applied to ${}^{0}_{5}R$, rotating from frame 5 to 6 and then from frame 6 to 0, obtaining

$$ {}^{0}_{5}R = {}^{0}_{6}R \, {}^{6}_{5}R. \tag{19} $$

Substituting (19) in (18), it is obtained (20).

$$ {}^{0}_{5}P = {}^{0}_{6}P - {}^{0}_{6}R \, {}^{6}_{5}R \, {}^{5}_{6}P \tag{20} $$

The rotation matrix ${}^0_6R$ is computed from the input Euler angles (see (21)), and the notation ${}^0_6P$ is simply queuing the 3D end effector's position.

$$
\begin{aligned}
{}^0_6R &= R_z(\alpha)R_y(\beta)R_z(\gamma) \\
&= \begin{bmatrix} ca*cb*cg - sa*sg & -ca*cb*sg - sa*cg & ca*sb \\ sa*cb*cg + ca*sg & -sa*cb*sg + ca*cg & sa*sb \\ -sb*cg & sg*sb & cb \end{bmatrix} \\
w/ \quad & ca = \cos(\alpha), \quad cb = \cos(\beta), \quad cg = \cos(\gamma), \\
& sa = \sin(\alpha), \quad sb = \sin(\beta), \quad sg = \sin(\gamma).
\end{aligned} \tag{21}
$$

At this point, the only variable missing is ${}^6_5R$, which is written by

$${}^6_5R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix}. \tag{22}$$

At this point there are only two missing variables: ${}^0_5P$ and ${}^6_5R(\theta)$. Considering the home position conditions to facilitate the calculus and ${}^5_6P = L_5 = (L_{5x}, 0, 0)$, in other other words the link distance between the fifth and sixth joint in the home position just differs in the x coordinate. Consequently, for any value of θ, (20) is given by

$${}^0_5P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{end\ effector} - L_{5x} \cdot \begin{bmatrix} ca*cb*cg - sa*sg \\ sa*cb*cg + ca*sg \\ -sb*cg \end{bmatrix} \tag{23}$$

3) Planar Chain:* By definition, a planar chain has all the links constrained to move in parallel to the same plane. This means that this type of chains only includes prismatic joints where their axes must be parallel or lie in the plain of the chain, and revolute joints that are perpendicular to it.

For each wrist position there are two symmetric planar geometries $\theta_1 = \{-\gamma, \gamma\}$, although there is only one option of slope (forward or backwards) to reach the ${}^0_5P = (xy, z)$. On the other hand, for each slope, there are two symmetric possibilities as it is shown in the figure 2.

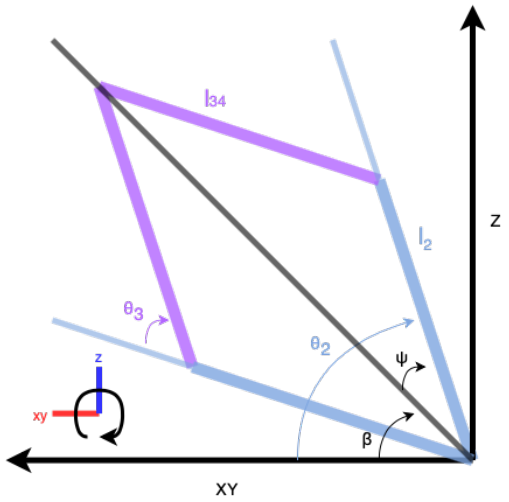


Fig. 2. Planar Chain in 2D view

According to the figure 2, the planar geometry is written as

$$\begin{cases} xy &= l_2\cos(\theta_2) + l_{34}\cos(\theta_2 + \theta_3) \\ z &= l_2\sin(\theta_2) + l_{34}\sin(\theta_2 + \theta_3). \end{cases} \tag{24}$$

Secondly it is computed the angle between the horizontal plan ($z = 0$) and the ${}^0_5P$. This angle, titled β, represents the symmetric axis between the two solutions, $\beta = \arctan2(z/xy)$. Therefore, the expressions are

$$\begin{cases} \beta = \theta_2 + \psi\,, & (\theta_3 < 0) \\ \beta = \theta_2 - \psi\,, & (\theta_3 \geq 0) \end{cases}. \tag{25}$$

With this being said, both θ_2 and θ_3 are easily computed using the cosine formula.

$$c^2 = a^2 + b^2 - 2ab\cos(\gamma) \quad , \quad \gamma = \angle\ ab. \tag{26}$$

Applying (26) in the inside upper triangle in 2, it is clear that $l_{34}^2 = l_2^2 + xyz^2 - 2 \cdot l_2 \cdot xyz \cdot \cos(\psi) \leftrightarrow \psi = \arccos\frac{l_2^2 + xyz^2 - l_{34}^2}{2 \cdot l_2 \cdot xyz}$, with $xyz = \sqrt{xy^2 + z^2}$. Analysing the same triangle but another angle, the problem is written by:
$\pi - \theta_3 = \arccos\frac{l_{34}^2 + l_2^2 - xyz^2}{2 \cdot l_2 \cdot l_{34}} \leftrightarrow \theta_3 = \arccos\frac{xy^2 + z^2 - l_{34}^2 - l_2^2}{2 \cdot l_2 \cdot l_{34}}$.
To conclude the reasoning, using (25) it is computed $\theta_2(\theta_3, \Psi)$ for the 2 solutions.

Lastly, it is mapped the angles from the 2D geometry for the 3D. Since it is used the direct configuration for the axes, the direction of the rotation between 2D and 3D is contrary, therefore the angles are symmetric. The final step is to subtract the angles of the home position (see figure 3) because in 2D the home position is lied in the horizontal plane. With this being said, the mapping is written by

$$\begin{cases} \theta_2^{3D} = -\theta_2^{2D} - (-\frac{\pi}{2}) \\ \theta_3^{3D} = -\theta_3^{2D} - (\alpha = \arctan\frac{L_{3x} + L_{4x}}{L_{3z}}). \end{cases} \tag{27}$$

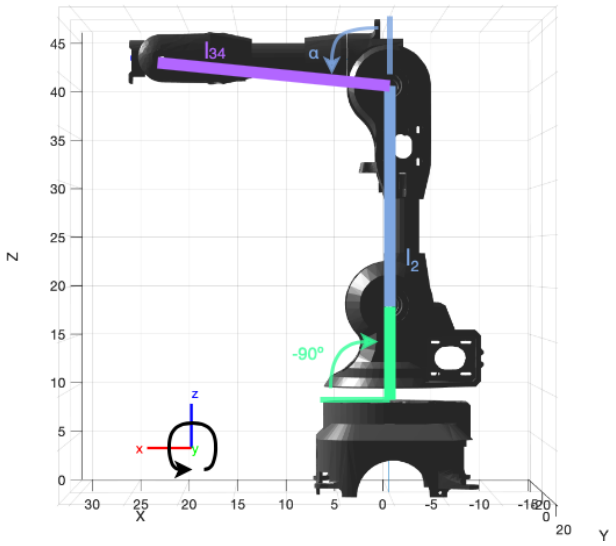


Fig. 3. Planar Chain in 3D view

4) Segmented Transformation†: At this point the orientation of the rigid body might be different from the home position's orientation. According to what is done in the direct kinematics, ${}^0_3T$ is computed, and with the same reasoning as (18), transforming from frame 6 to 0 to 3.

$${}^3_6T = {}^0_6T\ {}^3_0T. \tag{28}$$

Now, it is computed the transformation matrix ${}^3_6T$ analytically, with (12) (10) (8), in (29) and (30), where it is used the same notation as in (21) with θ_4, θ_5 and θ_6 represented only with the respective number.

$${}^3_6T = {}^5_6T\ {}^4_5T\ {}^3_4T. \tag{29}$$

$${}^3_6T = \begin{bmatrix} c_5 & s_5s_6 & s_5c_6 & c_5L_{6x} + L_{4x} + L_{5x} \\ s_4s_5 & c_4c_6 - s_4c_5s_6 & -c_4s_6 - s_4c_5c_6 & s_4s_5L_{6x} \\ -c_4s_5 & s_4c_6 + c_4c_5s_6 & -s_4s_6 + c_4c_5c_6 & -c_4s_5L_{6x} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{30}$$

From ${}^3_6T_{11} = \cos(\theta_5)$ it is computed the angle of the fifth joint, it is important to notice that the $\arccos(.)$ function has 2 solutions in $[-\pi, \pi]$. Knowing that, from each solution it is computed

$$\theta_4 = \arcsin({}^3_6T_{21}/\sin(\theta_5)), \tag{31}$$

that also gives 2 solutions for each θ_5, because $\arcsin(.)$ gives 2 solutions in $[-\pi, \pi]$. However, for each θ_5 there is only 1 value for θ_4, so those values need to be verified with

$$ {}^3_6T_{31} = -c_4 s_5, \tag{32}$$

where only 1 solution for each θ_5 pass.

With the same reasoning, θ_6 is computed by

$$\theta_6 = \arcsin({}^6_3T_{12}/sin(\theta_5)), \tag{33}$$

giving 2 solutions for each value of θ_5 that are verified with

$$ {}^3_6T_{13} = -s_5 c_6, \tag{34}$$

discarding 2 solutions. Finally, for each θ_5 there is only one value of θ_4 and θ_6, so there are only two combinations of the angles of the fourth, fifth and sixth joints that are feasible.

Now, if $\cos(\theta_5) = 1$, it means that θ_5 is 0, therefore the fourth joint's x axis is coincident with the sixth joint's x axis, which leads to a singularity, that is, there are infinite combinations of θ_4 and θ_6 that give the same end effector orientation. In this specif case, it is considered $\theta_6 = 0$, so it is possible to simplify ${}^3_6T_{22}$ and ${}^3_6T_{32}$ to what is represented in (35).

$$\begin{cases} {}^3_6T_{22} = c_4 \\ {}^3_6T_{32} = s_4 \end{cases} \tag{35}$$

Finally, θ_4 can be computed with $\theta_4 = \arctan2({}^3_6T_{32}, {}^3_6T_{22})$.

From now on, the 5.5mm offset in the z direction will be considered. The way to solve the inverse kinematics at this point is to iterate for different values of θ_6 in the interval $[-\pi, \pi[$, in this case the program starts with the value $-\pi$ and increments $\frac{\pi}{36}$ in each iteration until the value is $\pi - \frac{\pi}{36}$, and then proceed in the same way as without the offset. Firstly, it is computed ${}^0_5P$ with (20) and then the first 3 angles, θ_1, θ_2 and θ_3, as shown before, after that, ${}^0_3T$ is computed. Now, with (28), one gets ${}^3_6T$ and then can compute the last 3 joints angles showed before.

Lastly, to inspect if the computed angles of each of the eight solutions are correct, the difference between the two angles of the sixth joint, the one given by the iteration and the other computed with (33), is computed to see if it is within a threshold (e.g. $\frac{\pi}{36}$) in order to identify if it is a good solution or not. It is good to notice that with this method, it will have multiple solutions with a tiny error and, sometimes, with no error.

VII. Experimental Results*†‡

A. Solutions for the Inline Wrist Experiment (w/o 5.5mm)‡*

1) Direct Kinematic‡: To validate the work done, an important step is to validate the direct kinematics problem due to its importance for the correct testing of the inverse kinematics problem. Therefore, although tested several times and with several different sets of values for each of the six manipulator joints, in this section a simple simulation is described, with different than zero values for each joint variable and so that it does not coincide with any singularity. To note that it's possible, using a flag in the direct and inverse kinematics MATLAB functions, to solve both direct and inverse problems with the manipulator's joint constraints, altough being set as false by default. With this being said, the direct kinematics' input is defined to be the set of angles $[-\frac{\pi}{4}, \frac{\pi}{3}, -\frac{\pi}{6}, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{4}]$, where the simulated result of position plus orientation is seen in Figure 4, leading the position of the end effector to be, approximately, (295.64, -292.73, 19.97) [mm] and the orientation to be, approximately, (-1.8925, 2.7352, 0.8861) [rad]. From this, it can be seen that the joint 1 leads the manipulator to be in the fourth quadrant (according to the defined referentials, where Y is pointing towards the reader) due to the -π/4 rotation, the joint 2 rotates π/3 from its initial position and joint 3 -π/6, so considering a planar 2D geometry the arms look above a diagonal traced from joint 2 to joint 5. Then the last three joints lead to the manipulator's orientation, seen in the augmented circle of Figure 4. Being these Euler angles in a Z-Y-Z convention it means, from the starting frame defined in Figure 5(a), that there was a rotation of $\approx$ -108º along the Z axis, then a rotation of $\approx$ 156.7º along the Y axis and finally a rotation of $\approx$ 50.8º along the Z axis as showed in Figures 5 (b), (c) and (d), respective.

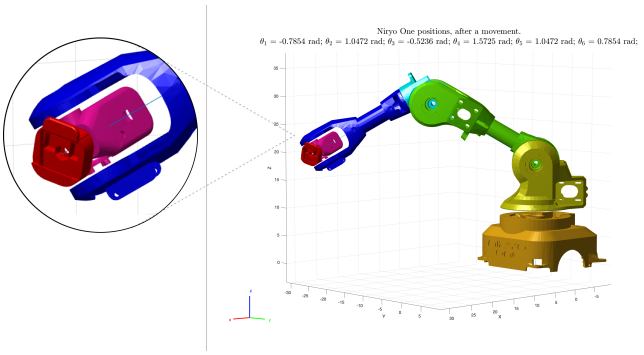


Fig. 4. Experimental Result - Direct kinematics

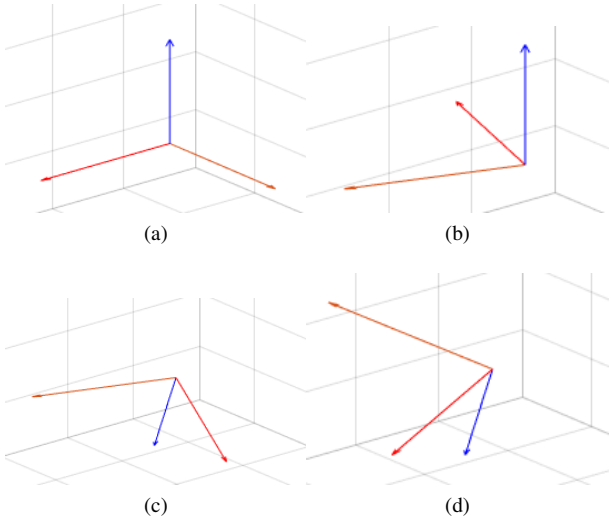
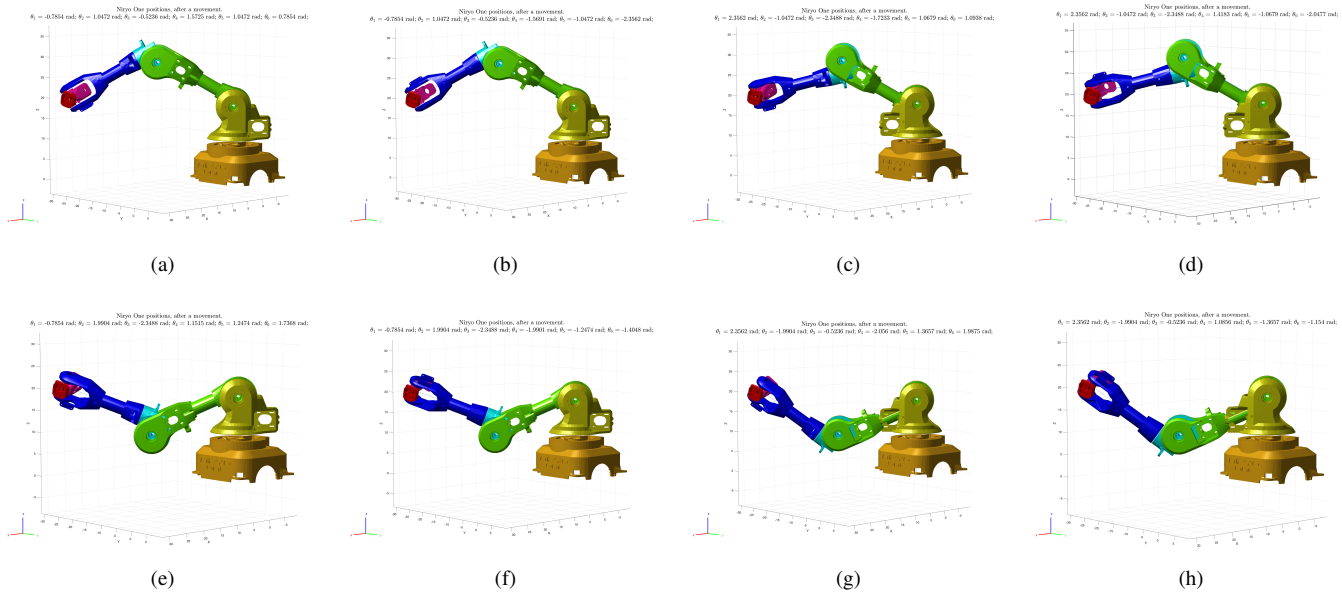


Fig. 5. Euler Angles rotation [-1.8925, 2.7352, 0.8861], with *Z-Y-Z*

Being the end effector starting position as in Figure 3, one can be certain that by moving these angles it gets to the orientation showed in the focused circle in Figure 4, because the end effector "plane" was perpendicular to the X axis and pointing contrary to the Z axis direction, and in Figure 5(d) it is seen that the end effector would be pointing upwards (contrary to the Z axis) and perpendicular to the X axis, which is precisely the orientation seen in Figure 4.

2) Inverse Kinematic:* Regarding the inverse kinematic solutions, for the purpose of exemplification, it is tested the software with the position and orientation of the previous section. As it was being said throughout the paper, an inline wrist robot arm may have up to eight solutions. Analysing in a top-down reasoning, the figure 6 can be studied in blocks. First things first, it is evident that each line contains a different direction for the elbow. This happens due to the combination of three different factors that will be discussed next. The first bifurcation is in the shoulder (yellow): the sub figures



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 6. Inline Wrist

{a, b, e, f} represents the arm with the θ_1 in the fourth quadrant, on the other hand the sub figures {c, d, g, h} show the robot with the opposite slope because the θ_1 is rotated by plus π. The second divergence is related with the planar geometry and it is evident in each column pair (sub figure {a-e, b-f, c-g, d-h}). This occurs once the elbow (light blue) has to adapt his angle (θ_3) to the angle of the arm (light green) (θ_2), in order to reach the same position for the fifth joint. On the other hand, for each line pair (sub figure {a-b, c-d, e-f, g-h}), is evident the change in the orientation of the links from the third joint to the end effector (hand). This is notorious in the head of the forearm (heavy blue) or in the body of the wrist (pink). Lastly a quick note regarding the real situation of this arm, due to the physical limitations on each joint just the combination of angles on the sub figures {a,b} would be possible to achieve.

TABLE I
SOLUTIONS FOR THE INVERSE KINEMATICS IN FIGURE 6 IN RAD.

Subfigure	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
(a)	-07854	1.0472	-0.5236	1.5725	1.0472	0.7854
(b)	-0.7854	1.0472	-0.5236	-1.5691	-1.0472	-2.3562
(c)	2.3562	-1.0472	-2.3488	-1.7233	1.0679	1.0938
(d)	2.3562	-1.0472	-2.3488	1.4183	-1.0679	-20477
(e)	-0.7854	1.9904	-2.3488	1.1515	1.2474	1.7368
(f)	-0.7854	1.9904	-2.3488	-1.9901	-1.2474	-1.4048
(g)	2.3562	-1.9904	-0.5236	-2.056	1.3657	1.9875
(h)	2.3562	-1.9904	-0.5236	1.0856	-1.3657	-1.154

B. Solutions for the not Inline Wrist Experiment (w/ 5.5mm)†‡

1) Inaccurate Solutions†*:* With the objective of exemplification, it is tested the inverse kinematics solutions, but with the hand not aligned with the wrist. For this example it is used the home position, where the given solutions are represented in the table II, with 8 different solutions. In figure 7, it is represented the best solution, that is, the solution with the less error in relation to the home position, with $\theta_1 = -3.1416$, $\theta_2 = -1.4905$, $\theta_3 = 0.002$, $\theta_4 = 3.1416$, $\theta_5 = 1.6513$ and $\theta_6 = 0$. Regarding this example, the best solution, theoretically, would be all the joints with value 0, but there are slightly differences due to approximated values and that the third combination of angles in the table II just zeros.

TABLE II
SOLUTIONS FOR THE INVERSE KINEMATICS FOR THE HOME POSITION

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
-3.1416	-1.4905	0.0002	3.1416	1.6513	0
0	1.4905	-2.8726	0	1.3821	0
0	0.0002	0.0002	0	0	0
-3.1416	-0.0002	-2.8726	3.1416	0.2688	0
3.1416	-1.5429	0.0555	0	-1.6542	-3.1416
3.1416	0.0054	-2.9279	0	-0.2191	3.1416
-0.0022	1.5429	-2.9279	3.1394	-1.3850	3.1412
-0.0022	-0.0055	0.0555	3.0983	0.0500	-3.0984

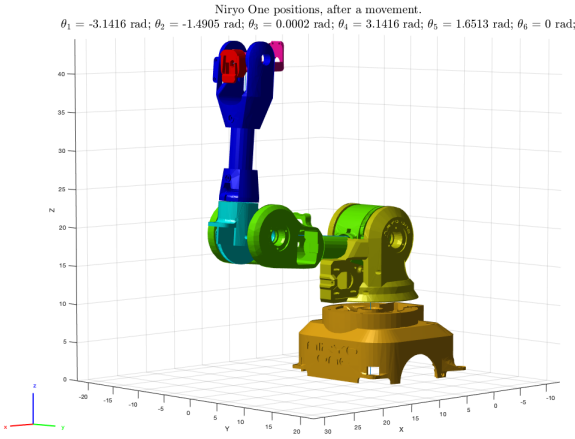


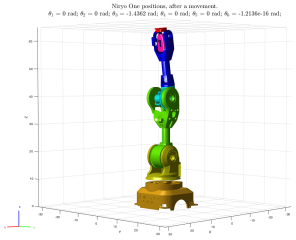
Fig. 7. Solution with less error in relation to the home position

In table III, there are roughly all the solutions to the fully stretched out robot. For this position, there are only 3 solutions, which are represented graphically in the figure 8.

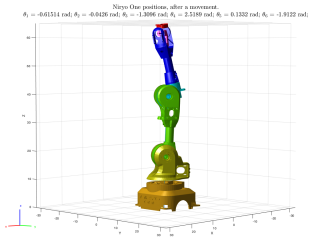
2) Accurate Solutions (Bonus Point)‡*:* Now, using *MATLAB's* function *fmincon*, a minimisation problem is solved, using as objective variable the Θ_6 angle, so that the best solutions are found, as showed in Table IV. To get these results, the regions for the *fmincon* function to search for the best solution need to be coded by hand to

TABLE III
SOLUTIONS FOR THE INVERSE KINEMATICS FOR THE STRETCHED OUT ROBOT

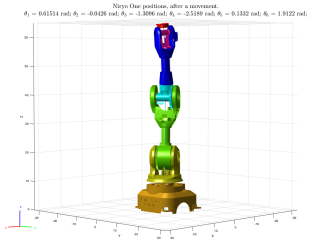
Subfigure	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
(a)	0	0	-1.4362	0	0	0
(b)	-0.6151	-00426	-1.3096	2.5189	0.1332	-1.9122
(c)	0.6151	-0.0426	-1.3096	-2.5189	0.1332	1.9122



(a)



(b)



(c)

Fig. 8. Inaccurate Solutions for the rigid body maximum stretched out

the function, so to get those lower and upper search regions, the plot showed in Figure 9 (generated in the inverse_kinematics.m file) is used, where it is shown the difference between the Θ_6 that is being iterated, or said otherwise, the proposed Θ_6, and the generated θ_6 from the inverse kinematics. This method generates the so called error that depends on the iterated Θ_6, where only the values lower than the $\frac{\pi}{36}$ threshold (and nearer than 2mm from the direct kinematics end effector position) are seen as acceptable values. With this, one can define certain bounded regions to apply in the *fmincon* region. As seen in Table IV, the accurate solutions are similar to the inaccurate ones seen in Table II, although, with some minor differences, which lead to a decrease in the order of magnitude of the error of around 10^{-3}.

TABLE IV
ACCURATE SOLUTIONS FOR THE HOME POSITION

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
-3.1416	-1.3858	-0.2606	3.1416	1.6513	0
0	1.4905	-2.8725	0	1.382	0
0.00003	0.002	0.002	0	0	0
-3.1416	-0.0002	-2.8726	3.1416	0.2688	0
3.1416	-0.0054	-2.928	0	-0.2191	-3.1416
3.1416	0.0054	-2.9279	0	-0.2191	3.1416
0.0022	1.5429	-2.928	3.1390	-1.3850	3.1416
-0.0022	-0.0055	0.0555	3.098	0.0500	-3.0982

3) Singularity at z axis:* One particularity about the movement of the rigid arm, that is clearly seen when there are more than eight cluster regions, happens when one or more degrees of freedom are lost. According to the author in [5], "A robot singularity is a configuration in which the robot end-effector becomes blocked in certain directions." This results can be seen in the Table V where for each solution display in each row, the joint angles θ_1 and θ_6 are symmetric since they cancel each other. Due to the sampling rate and

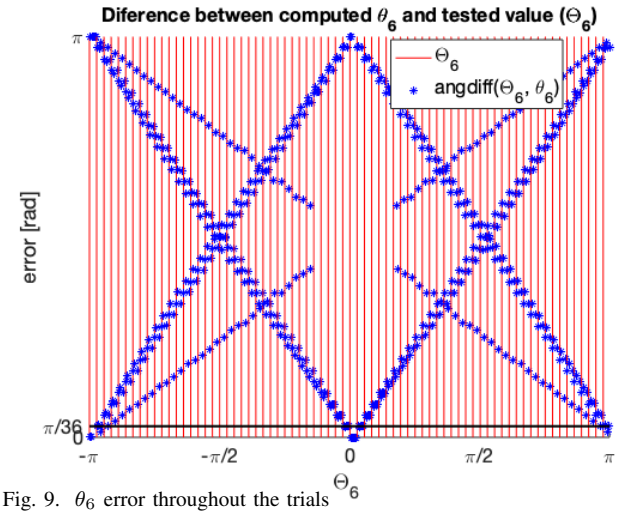
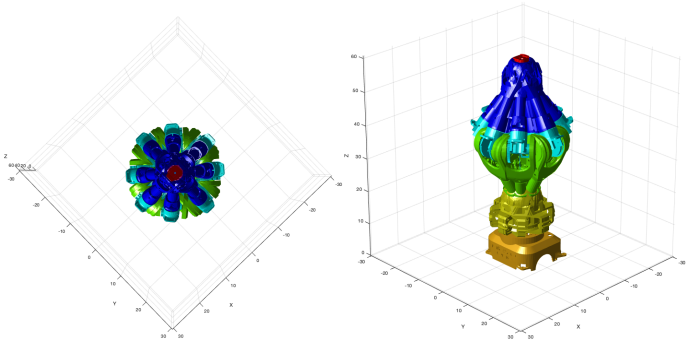


Fig. 9. θ_6 error throughout the trials

the clustering threshold it is only displayed 16 solutions, overlapped at Figure 10.

TABLE V
SOLUTIONS FOR THE INVERSE KINEMATICS WHEN A SINGULARITY HAPPENS

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
-3.0543	0.4627	-2.3046	-3.1416	-0.2711	-0.0873
1.6581	0.4627	-2.3046	3.1416	-0.2711	1.4836
-1.6581	0.4627	-2.3046	-3.1416	-0.2711	-1.4836
-0.0873	0.4627	-2.3046	-3.1416	-0.2711	-3.0543
0.8727	-0.4347	-0.5677	-3.1416	0.5684	2.2690
2.4435	-0.4347	-0.5677	-3.1416	0.5684	0.6981
-2.4435	-0.4347	-0.5677	3.1416	0.5684	-0.6981
-0.6981	-0.4347	-0.5677	-3.1416	0.5684	-2.4434
-1.5708	-0.4627	-0.5677	0	-0.5404	1.5708
0	-0.4627	-0.5677	0	-0.5404	0
1.5708	-0.4627	-0.5677	0	-0.5404	-1.5708
3.1416	-0.4627	-0.5677	0	-0.5404	-3.1416
-1.5708	0.4347	-2.3046	0	0.2991	1.5708
0	0.4347	2.3046	0	0.2991	0
1.5708	0.4347	-2.3046	0	0.2991	-1.5708
3.1416	0.4347	-2.3046	0	0.2991	-3.1416



(a) Top View (b) Side View

Fig. 10. Shoulder Singularity; X=Y=0, Z=600mm, $\alpha = \gamma = 0$, $\beta = -\arctan\frac{L_{3x}+L_{4x}}{L_{3z}}$ rad

For this family of rigid bodies, there are three well-known singularities: Wrist, Elbow and Shoulder. The last singularity is the one explained above when the axes of the last joint lie in the plane passing through the axes of joints 1 and 2, and a DOF is lost in the Z axis rotation. On the other hand, when the axes of the last joint lie on the

plane passing through the axes of joints 2 and 3 results on a DOF lost (Elbow singularity). Finally the Wrist rotation occurs when the axes of the fourth and sixth joints are coincident, and it happens in the wrist inline architecture (whitout the L_{5z}=5.5m).

VIII. Notes†

There were a few improvements from the first version (submitted):

- In the *inverse_kinematics.m* file, when the point and orientation are not possible and the program has not any solution, it crashes. The solution is to change the lines 78 to 89 after the *if* statement to the lines 96 to 100. This part of the code is a simple plot that could even be commented;
- In the file *bonusPoint.m*, the functions in *inverse_kinematics.m* need to be imported in order for it to work;
- The input vector of the clustering algorithm is flipped, in order to preserve the solution with less error, since when two solutions are recognised to be at the same group the second is privileged.

The Repository of this project including the code is available here.

IX. Conclusion†

The main focus of this work was to understand the calculations and the principles behind the direct and inverse kinematics for the Nyrio One robot. First, it is important to define the world frame, its position and orientation, as well as the auxiliary frames to ease this process. Note that one has to be very careful with the frames that he chose with respect to the world frame, because a small mistake may lead to a lot of hours wasted in order to solve it! With the frames set, the direct kinematics and inverse kinematics are easy to solve, because both just need to use the transformation matrixes between frames. For the direct kinematics, it gets the point of the end effector and orientation of the robot from given joint rotation angles, while for the inverse kinematics, it gets the joint rotation angles from a given point of the end effector and the orientation of the robot. However, for the inverse kinematics it is important to notice when the robot loses degrees of freedom, singularities, that gives an infinite number of solutions.

With that being said, it is possible to understand the importance of the kinematics for robotics manipulators, which do not consider the physics aspects of it, that is, its mass, the material of which it is made, etc., the kinematics not only allow one to know the actual position and orientation of the robot, but also they can compute the decisions to make for the robot to be in certain position with a certain orientation.

One way to improve the work developed in this laboratory could be to minimize the energy used in the movements for the inverse kinematics, which gives more than one solution, so the program chooses the solution that consumes the minimum energy. Other way to improve it is to consider not only physical constrains in the joints, but also constrains in the environment where the robot is. For example it could consider the floor or an obstacle, in order to restrict the robot movements.

Contents

List of Figures

List of Tables

References

[1] S. Ganapathy, *"Decomposition of transformation matrices for robot vision,"* Proceedings. 1984 IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 1984, pp. 130-139, doi: 10.1109/ROBOT.1984.1087163.

[2] F. Yang, Y. Zhu and P. B. Kingsley, *"A further investigation of the Euler angle calculation in diffusion tensor imaging,"* 2016 IEEE 13th International Conference on Signal Processing (ICSP), Chengdu, 2016, pp. 39-43, doi: 10.1109/ICSP.2016.7877792.

[3] Kumar. V. *"Introduction to Robot Geometry and Kinematics"*.

[4] *"Niryo One Mechanical Specifications"*

[5] B. Ilian *"What are Singularities in a Six-Axis Robot Arm?"*.