



UNIVERSITÀ DEGLI STUDI DI GENOVA

DIBRIS

DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY,  
BIOENGINEERING, ROBOTICS AND SYSTEM ENGINEERING

MODELLING AND CONTROL OF MANIPULATORS

---

## Second Assignment

Manipulator Geometry and Direct Kinematics

---

*Author:*

Balia Gian Marco  
Salterini Filippo  
Polese Carolina

*Student ID:*

s4398275  
s4516129  
s4862903

*Professors:*

Enrico Simetti  
Giorgio Cannata

*Tutors:*

Andrea Tiranti  
Luca Tarasi  
George Kurshakov

December 9, 2024

Contents

1 Assignment description 3

1.1 Exercise 1 . . . . . 3

2 Exercise 1 5

3 Appendix 8

3.1 Appendix A . . . . . 8

3.2 Appendix B . . . . . 8

Mathematical expression	Definition	MATLAB expression
$\langle w \rangle$	World Coordinate Frame	w
${}^a_b R$	Rotation matrix of frame $\langle b \rangle$ with respect to frame $\langle a \rangle$	aRb
${}^a_b T$	Transformation matrix of frame $\langle b \rangle$ with respect to frame $\langle a \rangle$	aTb

Table 1: Nomenclature Table

# 1 Assignment description

The second assignment of Modelling and Control of Manipulators focuses on manipulators' geometry and direct kinematics.

- Download the .zip file called *template\_MATLAB-assignment2* from the Aulaweb page of this course.
- Implement the code to solve the exercises on MATLAB by filling the template classes called *geometricModel* and *kinematicModel*
- Write a report motivating your answers, following the predefined format on this document.

## 1.1 Exercise 1

Given the following CAD model of an industrial 7 DoF manipulator:

**Q1.1** Define all the model matrices, by filling the structures in the *BuildTree()* function. Be careful to define the z-axis coinciding with the joint rotation axis, and such that the positive rotation is the same as showed in the CAD model you received. Draw on the CAD model the reference frames for each link and insert it into the report.

**Q1.2** Implement the method of *geometricModel* called *updateDirectGeometry()* which should compute the model matrices as a function of the joint position  $q$ . Explain the method used and comment on the results obtained.

**Q1.3** Implement the method of *geometricModel* called *getTransformWrtBase()* which should compute the transformation matrix from the base to a given frame. Calculate the following transformation matrices:  ${}^b_eT$ ,  ${}^5_3T$ . Explain the method used and comment on the results obtained.

**Q1.4** Implement the method of *kinematicModel* called *updateJacobian()* which should compute the jacobian of a given geometric model considering the possibility of having *rotational* or *prismatic* joints. Compute the Jacobian matrix of the manipulator for the end-effector. Explain the method used and comment on the results obtained.

*Remark:* The methods should be implemented for a generic serial manipulator. For instance, joint types, and the number of joints should be parameters.

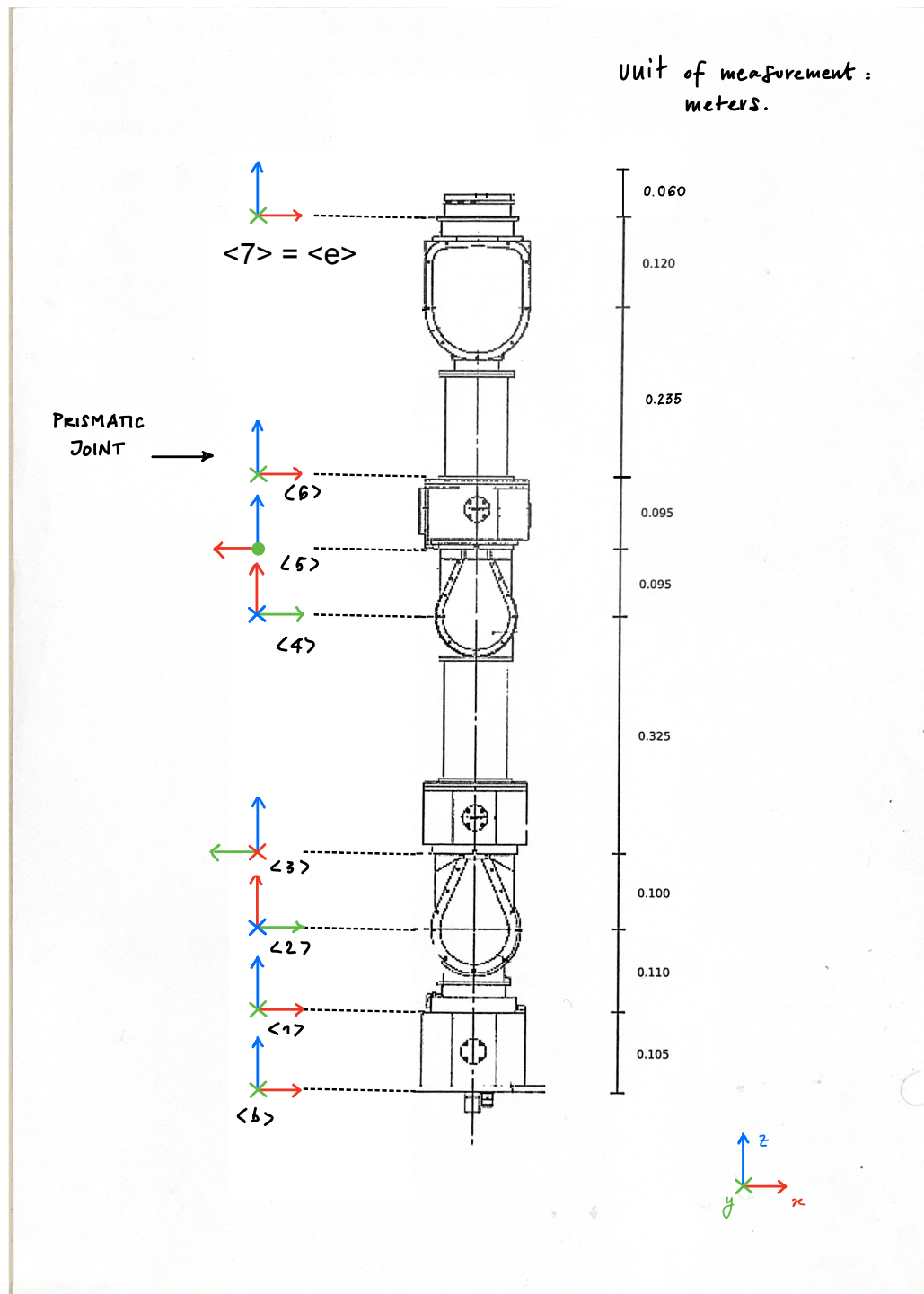


Figure 1: CAD model of the robot

## 2 Exercise 1

[Comment] For the last exercises include an image of the initial robot image of the final robot configuration

[Comment] For each exercise report the results obtained and provide an explanation of the result obtained (even though it might seem trivial). The matlab code is NOT an explanation of the algorithm.

### Q1.1

Given the ease of representation and comprehension of serial chain pose, it is crucial to compute the transformation matrices for each joint with respect to the preceding one. For instance, the transformation matrices  ${}^i_jT_0$  in the initial configuration ( $\mathbf{q}_0 = \mathbf{q} = \mathbf{0}_{1 \times 7}$ ) represent the transformation from frame  $\langle i \rangle$  with respect to the next consecutive frame  $\langle j \rangle$ . It takes into account both the rotational and translational contributions:

- The rotational contribution is computed with *Yaw-Pitch-Roll* convention for rotation, which defines the sequence of rotations around the  $z$ ,  $y$ , and  $x$  axes, respectively;
- The translational contribution is represented by the elements contained in the first three rows of the last column of the transformation matrix.

It's crucial to emphasize that these matrices vary based on the values of the vector  $\mathbf{q}$  that contain values for uniquely determining each joint's configuration. Given the general form of the transformation matrix:

$${}^i_jT = \begin{pmatrix} {}^i_jR & {}^i_jP \\ \mathbf{0}_{1 \times 3} & 1 \end{pmatrix} \quad (1)$$

where  $j$  is the considered chain's joint and  $i$  is the joint with respect is considered the transformation,  ${}^i_jR$  is the rotation matrix that define the rotation from the frame  $\langle i \rangle$  to  $\langle j \rangle$ , and  ${}^i_jP$  is the position of the frame  $\langle j \rangle$  with respect to frame  $\langle i \rangle$ . Taking into account  $\mathbf{q} = \mathbf{0}_{1 \times 7}$  the transformation matrixes for the initial configuration of the chain are:

$$\begin{aligned} {}^b_1T_0 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.105 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^1_2T_0 &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0.110 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^2_3T_0 &= \begin{pmatrix} 0 & 0 & 1 & 0.100 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^3_4T_0 &= \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.325 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ {}^4_5T_0 &= \begin{pmatrix} 0 & 0 & 1 & 0.095 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^5_6T_0 &= \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0.095 \\ 0 & 0 & 0 & 1 \end{pmatrix} & {}^e_6T_0 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.335 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

where:

- ${}^b_1T_0$  describes solely a translation along the  $x$  axis of  $0.105\text{ m}$ , then  ${}^b_1\mathbf{R}_0 = \mathbf{I}_{3 \times 3}$  and  ${}^b_1\mathbf{P} = (0, 0, 0.105)^T$ ;
- ${}^1_2T_0$  represents two  $\frac{\pi}{2}$  rotations, one around  $z$  and one other around  $y$ , both clockwise. There is also a translation component along the  $z$  axis of  $0.110\text{ m}$ , then  ${}^1_2\mathbf{R}_0 = \mathbf{R}_z(-\frac{\pi}{2}) \cdot \mathbf{R}_y(-\frac{\pi}{2})$  and  ${}^1_2\mathbf{P} = (0, 0, 0.110)^T$ ;
- ${}^2_3T_0$ , as  ${}^1_2T_0$ , has two clockwise rotations around  $z$  of  $\pi$  and  $y$  of  $\frac{\pi}{2}$  and a translation along  $x$  of  $0.100\text{ m}$ . Formally it means  ${}^2_3\mathbf{R}_0 = \mathbf{R}_z(-\pi) \cdot \mathbf{R}_y(-\frac{\pi}{2})$  and  ${}^2_3\mathbf{P} = (0.100, 0, 0)^T$ ;
- ${}^3_4T_0$  describes an counterclockwise rotations around  $z$  of  $\pi$  and a clockwise rotation  $y$  of  $\frac{\pi}{2}$  and a translation along  $z$  of  $0.325\text{ m}$ . Formally it means  ${}^3_4\mathbf{R}_0 = \mathbf{R}_z(\pi) \cdot \mathbf{R}_y(-\frac{\pi}{2})$  and  ${}^3_4\mathbf{P} = (0, 0, 0.325)^T$ ;
- ${}^4_5T_0$  represents two clockwise rotations around  $z$  and  $x$  of  $\frac{\pi}{2}$  and a translation along  $x$  of  $0.325\text{ m}$ . Formally it means  ${}^4_5\mathbf{R}_0 = \mathbf{R}_z(-\frac{\pi}{2}) \cdot \mathbf{R}_y(-\frac{\pi}{2})$  and  ${}^4_5\mathbf{P} = (0.095, 0, 0)^T$ ;
- ${}^5_6T_0$  express a counterclockwise rotation around  $z$  of  $\pi$  and a translation along  $z$  of  $0.095\text{ m}$ . Then  ${}^5_6\mathbf{R}_0 = \mathbf{R}_z(\pi)$  and  ${}^5_6\mathbf{P} = (0, 0, 0.095)^T$ ;
- ${}^e_6T_0$  highlight the presence of a translation along  $z$  of  $0.335\text{ m}$ . Then  ${}^e_6\mathbf{R}_0 = \mathbf{I}_{3 \times 3}$  and  ${}^e_6\mathbf{P} = (0, 0, 0.335)^T$ .

## Q1.2

When the chain's pose changes a good method to update the initial configuration is through the transformation matrix:

$${}^i_jT = {}^i_{j_0}T \cdot T_{\text{joint},j}$$

where  ${}^i_jT$  is defined as the transformation matrix for the new pose, and  $T_{\text{joint},j}$  is the transformation matrix that accounts for the joint's change of configuration, define as:

$$T_{\text{joint},j} = \begin{pmatrix} R_z(q_j) & P_z(q_j) \\ \mathbf{0}_{1 \times 3} & 1 \end{pmatrix}$$

where joint  $j$ , because the  $z$  axis is defined coinciding with the joint's rotation axis, can either rotate according to  $R_z(q_j)$  or translate by  $P_z(q_j)$ . If there is:

- a rotational joint: only the rotational part of the transformation matrix contribute to the new configuration. The rotation is along along the  $z$ -axis and the new  $T_{\text{joint},j}$  is:

$$T_{\text{joint},j} = \begin{pmatrix} \cos(q) & -\sin(q) & 0 & 0 \\ \sin(q) & \cos(q) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- a traslational joint: in this case, there is only a translation along the  $z$ -axis, then  $T_{\text{joint},j}$  becomes:

$$T_{\text{joint},j} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

For instance, if the vector is  $\mathbf{q} = (\pi/4, -\pi/4, 0, -\pi/4, 0, 0.15, \pi/4)$ , the updated trasformation matrix are:

$${}^b_1T(q_t) = \begin{pmatrix} 0.7071 & -0.7071 & 0 & 0 \\ 0.7071 & 0.7071 & 0 & 0 \\ 0 & 0 & 1.0000 & 0.1050 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix} \quad {}^1_2T(q_t) = \begin{pmatrix} -0.7071 & 0.7071 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 \\ 0.7071 & 0.7071 & 0 & 0.1100 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix}$$

$${}^2_3T(q_t) = \begin{pmatrix} 0 & 0 & 1.0000 & 0.1000 \\ 0 & -1.0000 & 0 & 0 \\ 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix} \quad {}^3_4T(q_t) = \begin{pmatrix} 0 & 0 & 1.0000 & 0 \\ 0.7071 & -0.7071 & 0 & 0 \\ 0.7071 & 0.7071 & 0 & 0.3250 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix}$$

$${}^4_5T(q_t) = \begin{pmatrix} 0 & 0 & 1.0000 & 0.0950 \\ -1.0000 & 0 & 0 & 0 \\ 0 & -1.0000 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix} \quad {}^5_6T(q_t) = \begin{pmatrix} -1.0000 & 0 & 0 & 0 \\ 0 & -1.0000 & 0 & 0 \\ 0 & -0 & 1.0000 & 0.2450 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix}$$

$${}^6_7T(q_t) = \begin{pmatrix} 0.7071 & -0.7071 & 0 & 0 \\ 0.7071 & 0.7071 & 0 & 0 \\ 0 & 0 & 1.0000 & 0.355 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix}$$

- ${}^b_1T(q_t)$ : in this case, the value of  $q$  is  $\frac{\pi}{4}$ , so there is only a rotation around the  $z$ -axis. The result can be verified by comparing the new matrix  ${}^b_1T(q_t)$  with  ${}^b_1T_0$  calculated in exercise 1. It is evident that there is a variation only in the  $R_z(q)$  component, while  $P_z(q)$  remains unchanged.
- ${}^1_2T(q_t)$ : for this configuration, the value of  $q$  is  $-\frac{\pi}{4}$ . As in the previous case, only  $R_z(q)$  contributes to the new configuration, while the traslational part remains unchanged. The value present in the last column derives from the initial configuration.
- ${}^2_3T(q_t)$ : in this case, since  $q$  is zero, the configuration remains unchanged. It is evident that  ${}^2_3T(q_t) = {}^2_3T_0$ .

- ${}^3_4T(q_t)$ : the value of  $q$  in this configuration is  $-\frac{\pi}{4}$ . As a result, there is only a rotational contribution along the  $z$ -axis and no translation.
- ${}^5_4T(q_t)$ : the value of  $q$ , as in the case of  ${}^2_3T(q_t)$ , is zero. Therefore, no rotations or variations occur. The resulting matrix is equivalent to the initial matrix,  ${}^4_5T_0$ .
- ${}^5_6T(q_t)$ : this configuration is characterized by a translational contribution, with  $q$  equal to 0.15. The matrix remains unchanged from the initial configuration in terms of the rotational component, while  $P_z(q)$  is affected by the translation along the  $z$ -axis.
- ${}^6_7T(q_t)$ : in the final configuration, the value of  $q$  is  $\frac{\pi}{4}$ , corresponding to a rotation around the  $z$ -axis. This means that the translational part,  $P_z(q)$ , remains unchanged. As a result, the new transformation matrix  ${}^6_7T(q_t)$  will differ from the initial configuration matrix  ${}^6_7T_0$  only in its rotational component, specifically in the elements of the  $R_z$  sub-matrix that encode the rotation about the  $z$ -axis.

### Q1.3

For this part of the assignment, we used the following equation to compute the transformation matrix from the base to a given frame:

$${}^i_eT = \prod_{i=1}^e {}^{i-1}_iT$$

The task required computing the transformation matrices for two different configurations:

$${}^b_eT = \begin{pmatrix} -0.5000 & -0.5000 & -0.7071 & -0.7039 \\ 0.5000 & 0.5000 & -0.7071 & -0.7039 \\ 0.7071 & -0.7071 & 0.0000 & 0.5155 \\ 0 & 0 & 0 & 1.0000 \end{pmatrix}$$

This matrix describes the final configuration of the manipulator after the simulation. The translational's result along  $z$  axes can be verified with Fig.3: it is evident that the total translation is 0.5155 metres; while the Fig.4 display the translation's result, along  $y$  and  $x$  with the same value of  $-0.7039$ .

$${}^5_3T =$$

### Q1.4

In the last exercise, it is required to compute the Jacobian matrix of a manipulator for the end-effector, based on a given geometric model.

The basic Jacobian matrix, which consists of two components: an angular contribution matrix  ${}^bJ_e^A$ , and a linear contribution matrix  ${}^bJ_{e/b}^L$ . The basic Jacobian is expressed as:

$${}^bJ_{e/b} = \begin{pmatrix} {}^bJ_e^A \\ {}^bJ_{e/b}^L \end{pmatrix}$$

These contributions are computed based on the type of joint (prismatic or rotational) considered, using the following relationships:

$$\begin{matrix} \text{Rotational joint} \\ \text{Prismatic joint} \end{matrix} \quad {}^bJ_e^A = \begin{cases} k_z \\ \mathbf{0} \end{cases} \quad {}^bJ_{e/b}^L = \begin{cases} k_z \times r_{e/b} \\ k_z \end{cases} \quad (2)$$

where  $k_z$  is the unit vector along the joint axis, and  $r_{e/b}$  is the position vector from the base to the end-effector. We obtain the Jacobian for the manipulator's end-effector, which reflects both the joint contributions and the rigid-body effects:

$${}^bJ_{e/b} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.0000 & 1.0000 & 1.0000 & 1.0000 & 1.0000 & 0 & 1.0000 \\ 0 & 0 & 0.0500 & 0.2125 & 0.2797 & 0 & 0.7039 \\ 0 & 0 & -0.0500 & -0.2125 & -0.2797 & 0 & -0.7039 \\ 0 & 0 & 0 & 0 & 0 & 1.0000 & 0 \end{pmatrix}$$



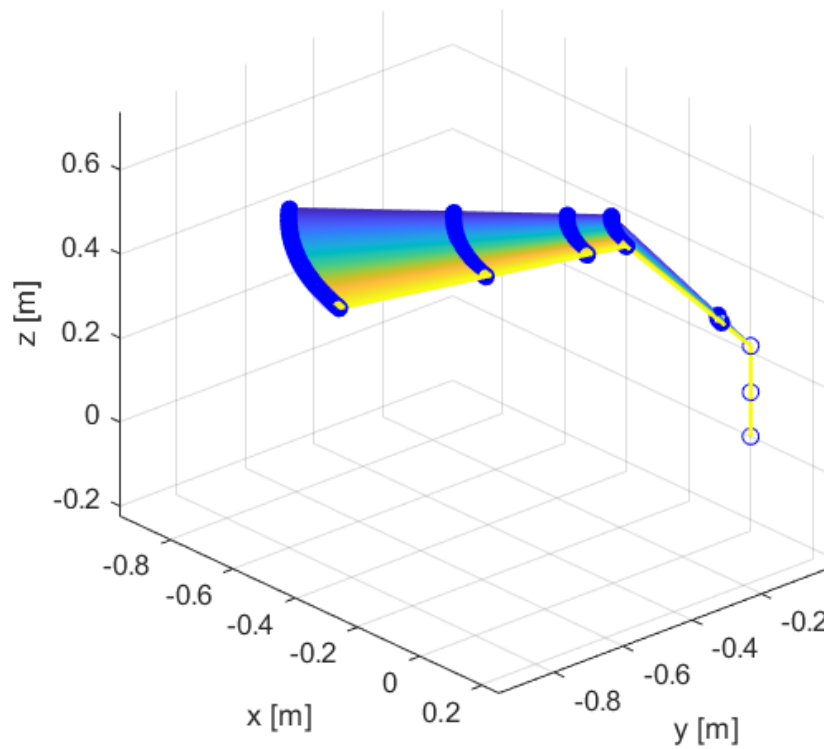


Figure 2: Description of the manipulator's motion with a 3 dimensional graph.  $x$  and  $y$  are horizontal axis, while  $z$  is vertical axes. Measurement's unit for the three axis is meters.

### 3 Appendix

*[Comment] Add here additional material (if needed)*

#### 3.1 Appendix A

#### 3.2 Appendix B

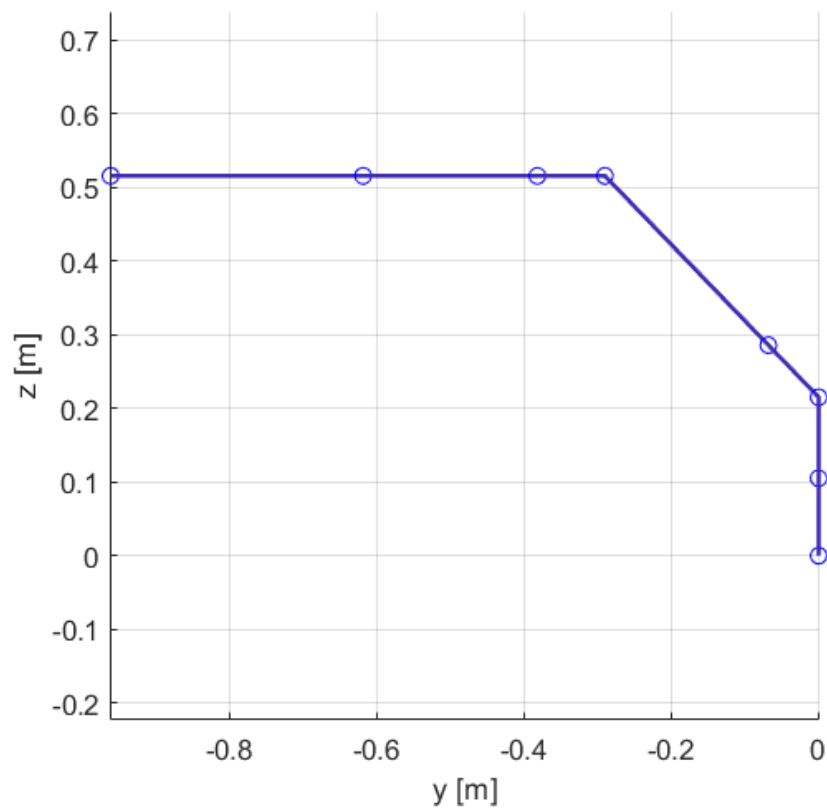


Figure 3: Configuration of the manipulator at the simulation's end. Measurement's unit for each axis is meters.

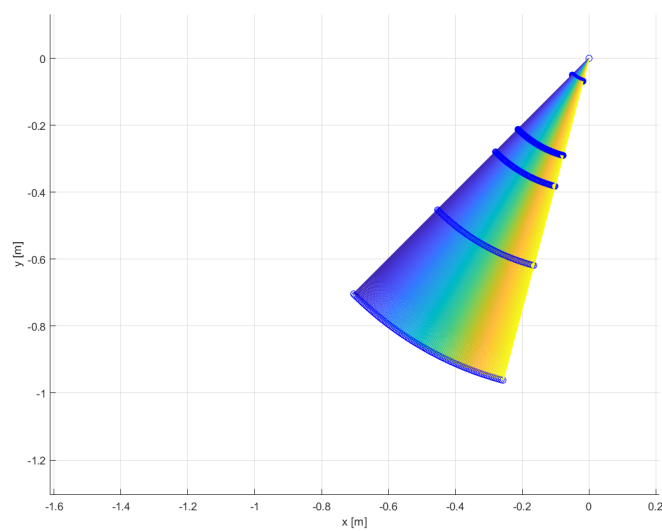


Figure 4: Fig.2 on plane  $xy$