Introduction to Robotics

LAB 2 – FORWARD AND INVERSE KINEMATICS OF A 3-DOF ROBOTIC ARM

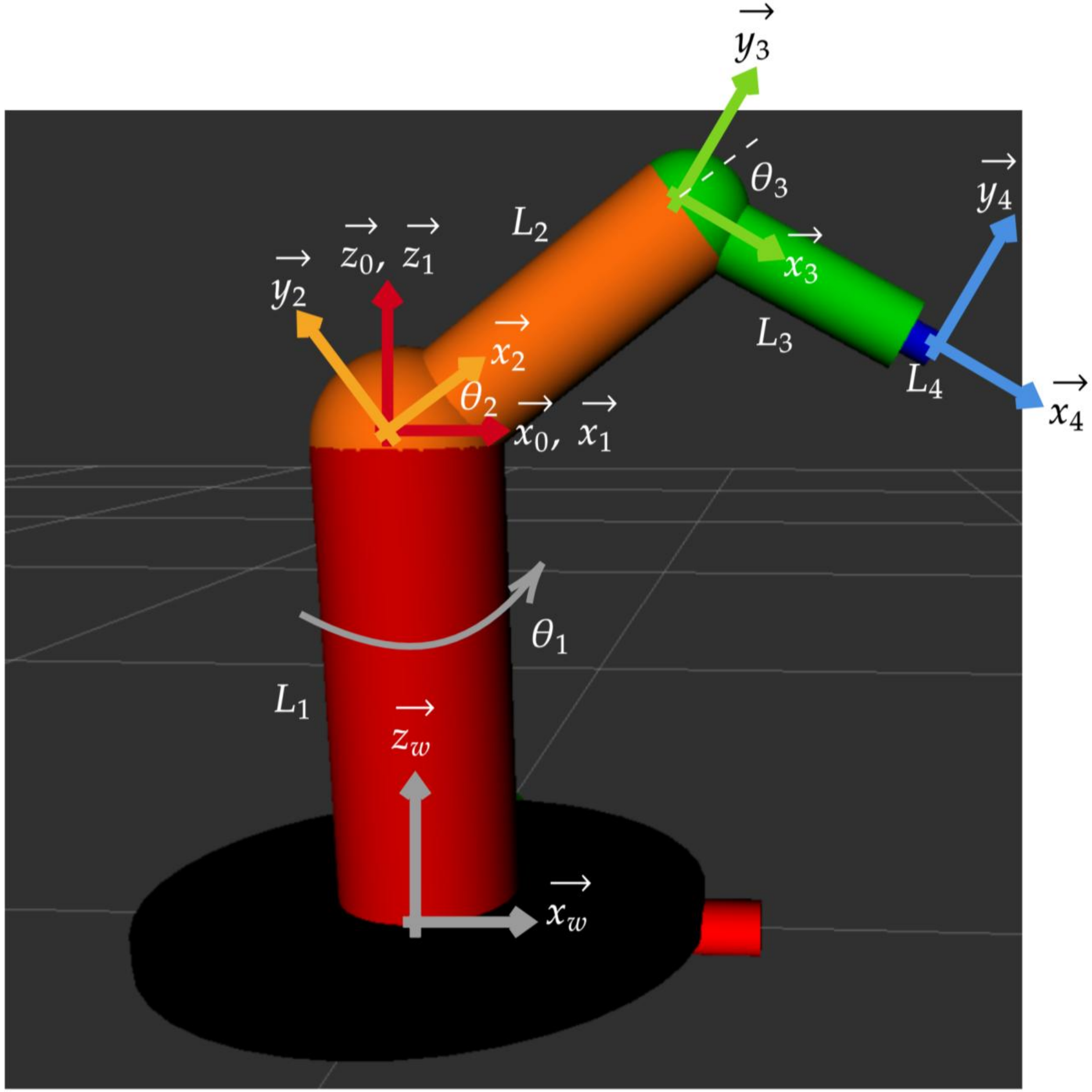
Objectives

Objectives

Apply the notions discussed during the lectures to position the end effector of the robot you designed during Lab 1.

- Determine the forward kinematics of the robot (FK)
- Determine its inverse kinematics (IK)
- Compute the IK on ROS
- Validate the IK with Gazebo and RViz

Part 1
Forward Kinematics



Robot’s Frames

Exercises

Referring to the previous scheme, determine the modified Denavit-Hartenberg parameters

$\boldsymbol{S_{i\to i+1}}$	$\boldsymbol{a_i}$	$\boldsymbol{\alpha_i}$	$\boldsymbol{d_{i+1}}$	$\boldsymbol{\theta_{i+1}}$
$S_{w\to 0}$				
$S_{0\to 1}$				
$S_{1\to 2}$				
$S_{2\to 3}$				
$S_{3\to 4}$				

$\boldsymbol{a_i}$: distance between $\vec{z}_i$ and $\vec{z}_{i+1}$ along the axis $\vec{x}_i$

$\boldsymbol{\alpha_i}$: angle between $\vec{z}_i$ and $\vec{z}_{i+1}$ around $\vec{x}_i$

$\boldsymbol{d_{i+1}}$: distance between $\vec{x}_i$ and $\vec{x}_{i+1}$ along the axis $\vec{z}_{i+1}$

$\boldsymbol{\theta_{i+1}}$: angle between $\vec{x}_i$ and $\vec{x}_{i+1}$ around $\vec{z}_{i+1}$

Exercises

Based on the previous table, determine the homogeneous transformation matrix for each joint:

$$T_{i \to i+1} = T_{i+1}^{i} = \begin{bmatrix} c_{\theta_{i+1}} & -s_{\theta_{i+1}} & 0 & a_i \\ s_{\theta_{i+1}} c_{\alpha_i} & c_{\theta_{i+1}} c_{\alpha_i} & -s_{\alpha_i} & -s_{\alpha_i} d_{i+1} \\ s_{\theta_{i+1}} s_{\alpha_i} & c_{\theta_{i+1}} s_{\alpha_i} & c_{\alpha_i} & c_{\alpha_i} d_{i+1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Exercises

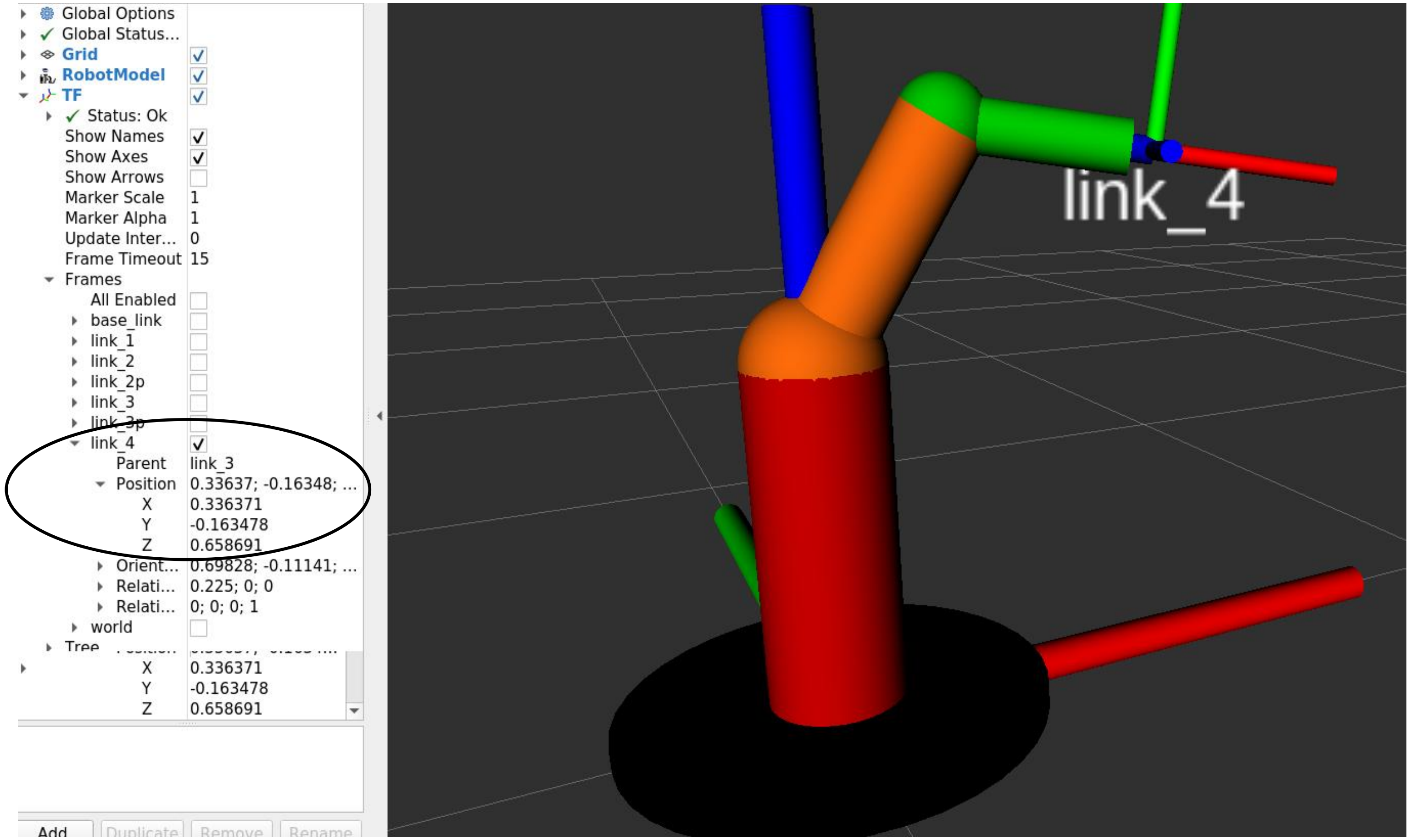
1. Deduce the forward kinematics
2. Determine the coordinates (x, y, z) of the end effector for each configuration $(\theta_1, \theta_2, \theta_3)$:

Parameters	L_1	L_2	L_3	L_4
Lengths (meters)	0.43	0.3	0.2	0.025

Angles (radians)	x	y	z
$(0, 0, 0)$			
$(0.45, 1.44, -1.23)$			
$(-0.58, 0.4, -1.5)$			

3. Compare the results with the coordinates displayed on RViz (see next slide)

End effector position on RViz



Part 2
Inverse Kinematics

Exercises

- Starting from the forward kinematics, determine the inverse kinematics of the robot.
- To solve the equations more easily, it is recommended to use the following notation:

$$z_2 = z - L_1$$

- Then, complete the following table.

Coordinates (x, y, z)	θ_1	θ_2	θ_3
(0.22, -0.47, 0.44)			
(0.44, 0.23, 0.56)			
(0.26, -0.26, 0.77)			
(0.01, -0.42, 0.16)			
(0.12, -0.5, 0.53)			

Part 3
IK implementation

```
+---in426_motion
|    |    package.xml
|    |    setup.cfg
|    |    setup.py
|    |
|    +---in426_motion
|    |         ik_1.py
|    |         motion_robot_2.py
|    |         send_goal_test.py
|    |         __init__.py
|    |
|    +---launch
|    |         ik_1_launch.py
|    |         motion_robot_2_launch.py
```

Implementation of the IK

READ CAREFULLY THE CONTENT OF THE FILE IK_1.PY AND EDIT ITS METHOD IK().

Part 4
Validation of your IK

Validation process

You will define desired positions the end effector of the robot should reach. They will be sent to the topic */goal* using a terminal.

The positions (x, y, z) given in the table of part 2 will be used for this purpose.

Once your IK node will compute the adequate joints the robot should have, it will start moving on Gazebo.

You will then compare the position reached by the end effector to the desired one (Refer to the next slide).

IK validation with Gazebo

1. Launch the simulation (terminal 1):

   ros2 launch in426_simu simu_1_launch.py

3. Launch your IK node (terminal 2):

   ros2 launch in426_motion ik_1_launch.py

4. Publish the desired coordinates (x, y, z) to the topic **/goal** (terminal 3):

   ros2 topic pub /goal geometry_msgs/msg/Vector3 "{x: 0.22, y: -0.47, z: 0.44}" --once

5. Check on RViz the position of the end effector and compare it to the desired one.