IN3140/IN4140 - Mandatory Assignment 2

Due: 6 March 2020, 23:59 (24h)

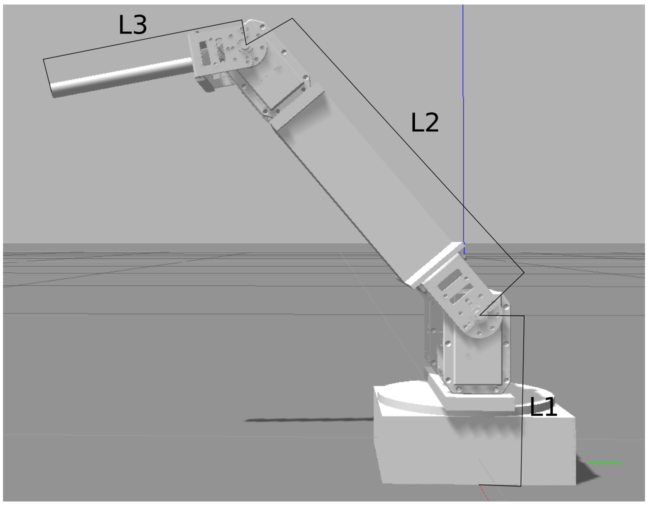

Figure 1: Simplified robot model in gazebo

Introduction

In this assignment we will continue working with the simplified CrustCrawler robot. The robot is displayed in figure 1. The dimensions in **millimeters** are:

- L1 = 100.9
- L2 = 222.1
- L3 = 136.2

Task 1 - Forward and inverse kinematics - (30%)

Implement the forward and inverse kinematics as function in Python.

a) 5% The forward kinematics function takes a list of joint angles as input and returns the corresponding cartesian coordinates for the tip of the arm as output:

```
def forward(joint_angles):
"""returns cart_cord"""
```

b) 5% The inverse kinematics function takes the cartesian position of the tip of the arm as input, and gives the corresponding joint configurations as output:

```
def inverse(cart_cord):
"""returns joint_angles"""
```

c) 5% Use the functions to show how you can verify that the inverse and forward kinematics are correctly derived. Hint: Use a function like round() in Python and at least 4 decimals accuracy.

d) 15% The TCP (Tool Center Point) is located at [x,y,z]=[0;-323.9033;176.6988] in the Base coordinate frame. Calculate all possible sets of solutions of the joint variables according to your DH setup. The answers must be in degrees with maximum four decimals.

Task 2 - Jacobian I - (40%)

a) 15% Derive the Jacobian matrix for the simplified CrustCrawler robot.

b) 2.5% What do we call configurations, for which rank J(q) is less than the maximum value?

c) 7.5% Use the Jacobian matrix to find such configurations for the robot.

d) 5% Give an evaluation of the results from the previous subtask and draw at least one of the obtained configurations. The drawing(s) shall have a simple 3D layout like the ones in the lecture slides, see referred standard below.

e) 5% A natural extension of the simplified robot would be a spherical wrist on the end. How can a spherical wrist alone (only picturing the wrist) be in singularity? Draw and explain. The drawing shall be a simple 3D illustration.

Use the standard for symbolic representation of 3D robot joints, found in Chapter 1.1.1 in the course book. You can find an example of a correct 3D representation of robot joints on page 85 (and 75) in the course book, figure 3.7.

f) 5% What are the practical consequences of not handling singularities?

Task 3 - Jacobian II - (30%)

Assume that axes Z_1, Z_2, and Z_3 all point out of the page in figure 2.

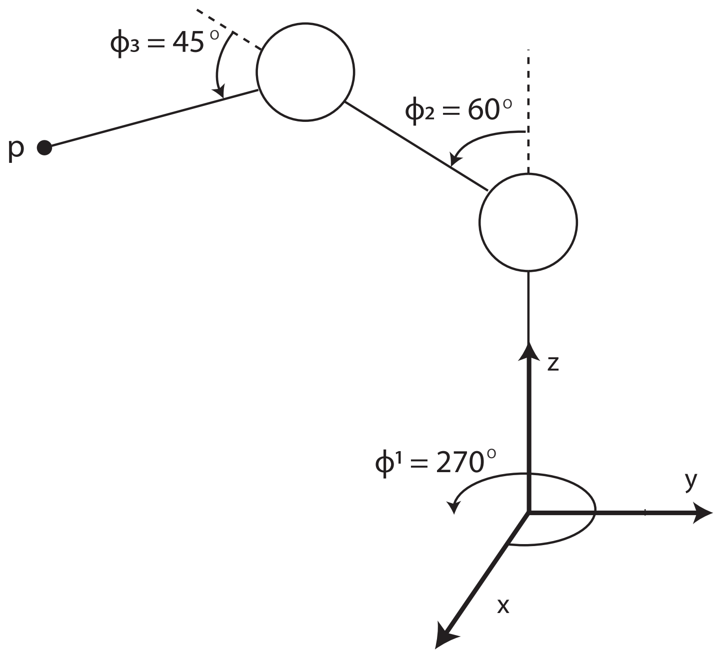

Figure 2: Robot configuration

a) 20% Implement the Jacobian matrix from Task 1 as a function. It takes the instant joint angles and joint velocities as input, returning a 3-dimensional vector of cartesian velocities of the end-effector as output.
The function shall look like this:

function cart_velocities = jacobian(joint_angles, joint_velocities)

where both *joint_velocities* and *cart_velocities* are vectors of size 3.

b) 10% Point p is located at the end-effector of the robot. We adjust the robot as displayed in figure 2, where $\phi_1 = 270°$, $\phi_2 = 60°$ and $\phi_3 = 45°$. *These (ϕ) angles are not to be used directly; you have to figure out the correct θ-angles corresponding to the placement of the joint coordinate frames.*

Given the configuration in figure 2 and the joint velocity vector $\dot{q} = [\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3]$, where $\dot{\theta}_1 = 0.1 rad/s$, $\dot{\theta}_2 = \dot{\theta}_3 = 0.05\ rad/s$, use your function to calculate the cartesian velocity of point "p" relative to the base coordinate frame.

REQUIREMENTS:
Each student must hand in their own assignment, and you are required to have read the following declaration on student submissions at the Department of Informatics: https://www.uio.no/studier/eksamen/obligatoriske-aktiviteter/mn-ifi-obliger-retningslinjer.html If any part of the declaration is unclear, contact one of the teachers for clarification.

IMPORTANT! Name the pdf file:

"in3140_oblig2_*your_username.pdf*"

Submit your assignment at https://devilry.ifi.uio.no.

Your submission must include:

1.) A pdf-document with answers to the questions.

2.) The two illustrations asked for in tasks 1d and 1e

3.) **A README.txt containing a short reflection on the assignment**; what was dificult, what was easy, was there anything you could have done better?

If you have used MATLAB, Python or other tools for computing an answer, your solution and approach must be illustrated and explained thoroughly in the pdf file. The files containing the code must also be delivered and named:

"in3140_oblig2_taskXX_*your_username.py .m .cpp etc*"

You are free to use whatever programming languages and tools you are familiar with, yet we strongly recommend solving task 1 by hand. **You are required to obtain a total score of at least 40% in order to pass this assignment.**

Deadline: 6 March 2020, 23:59 (24h)

You can use the slack channel *assignment 2* for general questions about the assignment, and the channels *forward_kinematics*, *inverse_kinematics*, *jacobian* and *matlab_and_python* for discussion. Slack team domain is; https://inf34804380robotics.slack.com
Do not hesitate to contact us if you have any further questions.

Artem Chernyshov - *artemch@uio.no*
Daniel Sander Isaksen - *daniesis@uio.no*