ENPM662 - Fall 2023

Homework - 05

Due: 2nd December 2023 Points/Weightage: 5 points

Manipulator Dynamics

In Homework - 04, you were tasked with moving the pen mounted on the robot end-effector along a given circular trajectory using position and velocity kinematics. In industrial applications involving grasping and force control, it is important to take into consideration the workspace forces/moments at the tool of the robot and the produced torques at the robot's joints.

As set up previously, the robot's end-effector has a pen (length of 10 cm, see Fig. 1) rigidly mounted on it such that it points along the 'axis' of the end-effector frame (Frame {n}). For a 6 degree of freedom robot, the Jacobian matrix is a square matrix of size 6x6.

In this assignment, your task is to calculate the joint torques that are required to compensate for the robot's weight and ensure the pen is pushed against the wall with a force of 5 N while drawing the circle of **radius 10 cm within 200 seconds**. Assume the robot motion is quasi-static, such that,

```
q_dot ≅ 0, q_ddot ≅ 0
```

Here is a CAD model of the robot which you may use to measure specific lengths and mass information. You may also use information from other sources such as the robot's datasheet, technical documentation, user manual, or research papers involving the UR10 robot. Cite the source(s) used if any.

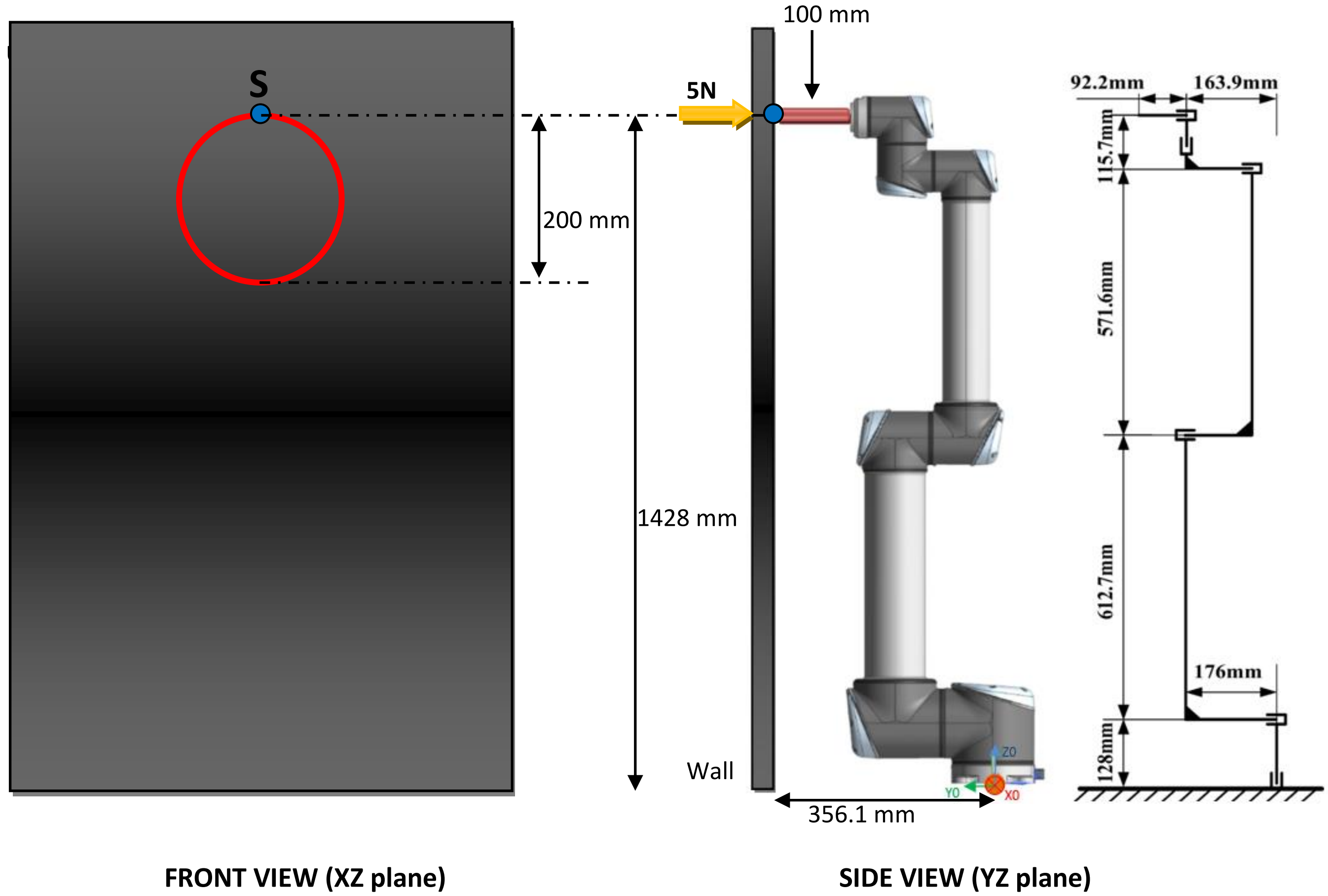

Fig. 1: UR10 Robot with the pen (Side and Front Views)

NOTE: Reuse code from the previous assignments to set up the Jacobian and Transformation matrices. You may use inbuilt functions that assist you in setting up the Gravity Matrix, such as taking derivatives, computing cross-products, matrix inversion, etc.

Deliverables

- A **PDF report** (preferably typed) containing the following. For results that are large or tedious to write manually, you may simply include a screenshot of that matrix as displayed in the terminal output after running your code.
  - Updated circular trajectory equations for a time duration of 200 seconds.
  - A brief explanation of the steps and equations involved in the computation of the joint torques using the Jacobian matrix and Lagrange equation. Clearly state assumptions, if any.
  - Mention the used masses and/or lengths of the robot links and cite all sources.
  - The picture of the plot output of the robot's joint torques between t=0 and t=200 seconds. Plot 6 graphs, one for each joint: 1, 2, 3, 4, 5 and 6.
  - Name your report: <your-directoryID>_hw5_report.pdf [Note: Directory ID ≠ UID!]

- All **codes** used (Make sure the code(s) submitted run without any errors!)
  - The code(s) **must** print the parametric Gravity matrix, g(q) to the terminal.
  - The code(s) **must** plot the computed joint torques (all 6).
    - Plotting the circle is optional.
  - Include an optional readme file that briefly describes how to run your code(s) and if any dependencies need to be installed.

Submit the above in 2/3 files (do not zip them up, drag the files and submit them on ELMS as a single submission)

Structure:

<your-directoryID>_hw3_report.pdf

<your-directoryID>_hw3_code.py —> should contain all code (.py)

<your-directoryID>_hw3_readme —> optional md or txt file with code execution instructions

Supplementary Material

Robot Dynamics Equation:

$$\mathrm{M}(\mathrm{q})\ddot{\mathrm{q}} + \mathrm{C}(\mathrm{q}, \dot{\mathrm{q}})\dot{\mathrm{q}} + \mathrm{g}(\mathrm{q}) = \tau + \mathrm{J}^{\mathrm{T}}(\mathrm{q})\ \mathrm{F}$$

Where,

$$\mathrm{F} = [\mathrm{F}_x, F_y, F_z, T_x, T_y, T_z]^T$$

(end-effector torque components are 0 in our case)

Lagrangian Dynamics:

$$\tau = \frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q}$$

Where,

L = K – P

K → Kinetic Energy of the system, P → Potential Energy of the system