Advanced Robotics 2021

CMP9764M

Workshop week 1

16 February 2021

Robot Kinematic and Dynamic

University of Lincoln

Instructor: Amir Ghalamzan

TA: Marta Crivellari

Attendance monitoring access Code: 602344

In this workshop you will exercise and learn forward and inverse kinematics.

You need to follow the steps below:

- Go to https://colab.research.google.com/github/
- On the tab GitHub copy and paste the GitHub link of the workshop https://github.com/imanlab/ar21_CMP9764M
- Press search
- Select w1 under Branch drop-down
- Open Workshop2DRobot.ipynb
- Save a copy of the original file on your google drive or GitHub so that you can edit the workshop file.
- Here you go!

The libraries below provide you with the functionalities you will need to complete the tasks in this workshop.

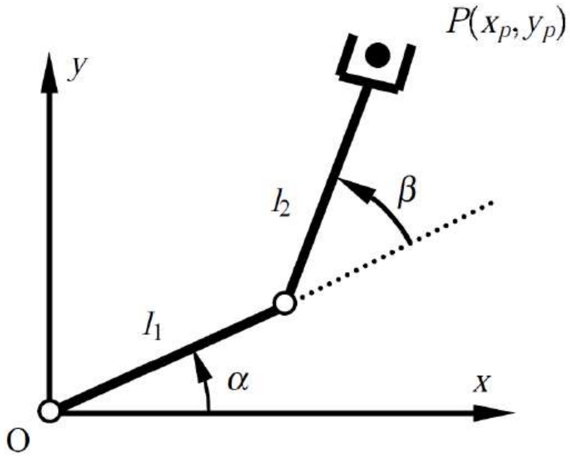
For the workshop, import the following library:

```
import math

from numpy.matlib import matrix,rand,zeros,ones,empty,eye
import numpy.matlib as M

import numpy as np
import numpy.matlib as ml
from numpy import linalg as LA
import matplotlib.pyplot as plt
import numpy.random as rnd
from matplotlib.patches import Ellipse, Circle
```

Scara Robot Kinematics

EX1: Write a function for the inverse (input = [S,length_arms], return Q) and forward kinematic (input = [Q,length_arms], return S).
Verify your function with S = [x,y] = [-70,-100] and length_arms=[100,70].

Forward kinematic = determine the End-Effector (S = [x,y]) position knowing the actuator positions (Q = [alpha,beta]).



$$\begin{cases} x_p = l_1 \cos\alpha + l_2 \cos(\alpha+\beta) \\ y_p = l_1 \sin\alpha + l_2 \sin(\alpha+\beta) \end{cases} \qquad (1)$$

Inverse kinematics = determine the actuator positions (Q = [alpha,beta]) position knowing the End-Effector (S = [x,y]).

The IK has two solution except for the singularity configuration (beta equal to 0 or 180°).
NB. In the formula (3) *s* = sin() and *c* = cos().

$$\beta = \pm\arccos\left(\frac{x_p^2 + y_p^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right) \qquad (2)$$

$$\alpha = \arctan_2(y_p, x_p) - \arctan_2(l_2 s\beta, l_1 + l_2 c\beta) \qquad (3)$$

EX2: Write a function that, given as input Q and the lengths of the arms, returns the Jacobian. Then write another function that, given as input S and the length of the arms, returns the Jacobian.
Verify with S=[-70,-100] and length_arms=[100,70]).

The relation between the velocity in joint space (Configuration space -- C-space) and in operational space in (cartesian space) is define as:

$$S = F(Q) \rightarrow \dot{S} = \frac{\partial F}{\partial Q}\dot{Q} = J\dot{Q} \qquad (4)$$

$$\begin{cases} x_p = l_1 \cos\alpha + l_2 \cos(\alpha+\beta) \\ y_p = l_1 \sin\alpha + l_2 \sin(\alpha+\beta) \end{cases}$$

We call J(Q) **Jacobian Matrix** of the system and for a 2-link robot it is equal to:

$$\begin{bmatrix} -l_1 \sin\alpha - l_2 \sin(\alpha+\beta) & -l_2 \sin(\alpha+\beta) \\ l_1 \cos\alpha + l_2 \cos(\alpha+\beta) & l_2 \cos(\alpha+\beta) \end{bmatrix} \tag{5}$$

To write the governing equitation of motion for the 2-link robot, we need to compute the time derivative of the equitation (4) where you have the relation between acceleration in Operational space and Configuration space.

$$\ddot{S} = \dot{J}\,\dot{Q} + J\,\ddot{Q} \tag{6}$$

Where $\dot{J}$ is equal to:

$$\dot{J} = \begin{bmatrix} -\dot{\alpha} l_1 \cos\alpha - (\dot{\alpha}+\dot{\beta}) l_2 \cos(\alpha+\beta) & -(\dot{\alpha}+\dot{\beta}) l_2 \cos(\alpha+\beta) \\ -\dot{\alpha} l_1 \sin\alpha - (\dot{\alpha}+\dot{\beta}) l_2 \sin(\alpha+\beta) & -(\dot{\alpha}+\dot{\beta}) l_2 \sin(\alpha+\beta) \end{bmatrix} \tag{7}$$

The inverse kinematic for the velocity and acceleration are obtain by inverting the relation (6) and (4).

EX3: Considering the following 5 trajectories from point1 to point2, calculate the manipulability of each trajectories and chose the best one. The pose [x,y] of the End-Effector along all the 5 trajectories is given.

The concept of **manipulability** is based on the ability to position and re-orientate the end-effector of the robotic arm in different directions. Optimizing the manipulability leads to increased performance for a robotic structure which can be used to design better robotic arms.

The purpose of manipulability indices is to give a quantitative measure of the ability to move and apply forces in arbitrary directions.

The manipulability measure based on the Jacobian matrix, J, is defined as:

$$\mu(\boldsymbol{\theta}) = \sqrt{\det(\mathbf{J}\mathbf{J}^T)} \tag{8}$$

Larger values of μ represents greater freedom for the specific configuration.

EX4: Plot the Velocity Ellipses considering that the end-effector follows the line y=0. (length_arms=[70,70]).

The **velocity manipulability ellipsoid** is a representation of the ability for the manipulator to change its end-effector position. The directions of the principal axes of the ellipsoid are given by the direction of the *eigenvectors* of JJ^T and the lengths of the axes by the *square root of the eigenvalues*. The manipulability ellipsoid provides an intuitive and easily illustrated perception of the ability for the configuration to move in a certain direction. A large singular value indicates high manipulability in the corresponding direction. The velocity manipulability ellipsoid also indicates how near we are a singular configuration since one of the axes of the ellipsoid will converge towards zero.

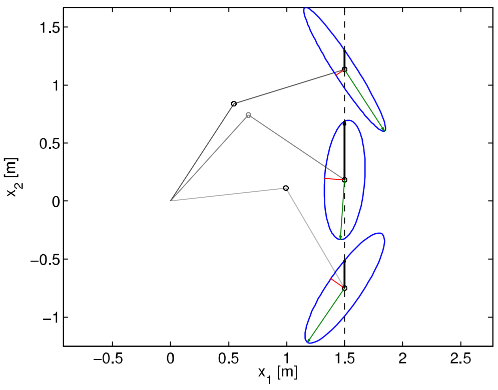


Figure 1 A sample 2 link robot arm with corresponding manipulability ellipsoids at different configurations.

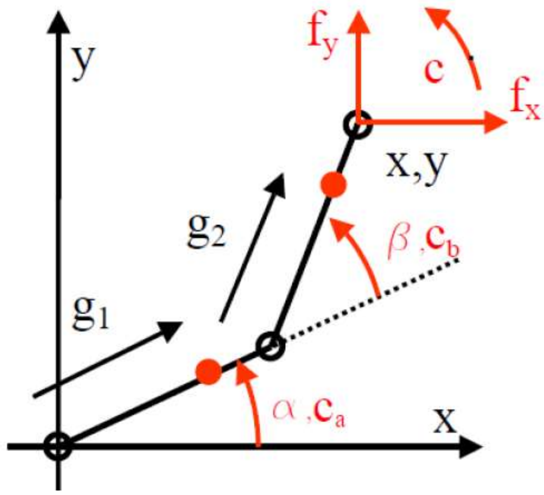
Part 2 (Challenge -- Optional)

Scara Robot Dynamics

EX1: Solve the inverse dynamic problem to obtain the torques during the movement of the gripper along a line from Si=[0.2,0.5] to Sf=[−0.5,0.2]. Consider a cycloidal motion curve that takes 5s to complete the task under the following external forces, Fse = [1;1;0;0;0;0;0;0].

Characteristics:

- Planar movement (x-y)
- 2 links (*l1, l2*)
- 2 Revolute joints (α, β –relative rotation between two consecutive links)
- 8 external forces (i.e. F_x, F_y, C_s + gravitational terms) some of which may be equal to 0.
- 2 actuation forces (i.e. C_α, C_β), as the number of Dofs.



Inverse dynamic problem: computation of the torque required by the motors to obtain a defined movement.
From the dynamic equation of the system, we can derive the torque applied as:

$$F_q = -J_e^T(-M\ddot{S}_e + F_{se}) \tag{8}$$

Where F_{se} is the external force define as: $F_{se} = [f_x, f_y, 0, 0, c, 0, 0, 0]^T$

$-M\ddot{S}$ is the inertia force of the system where M and S are define as:

$$S = [x,\, y,\, x_{g2},\, y_{g2},\, \theta,\, x_{g1},\, y_{g1},\, \alpha]^T \qquad M = \mathrm{diag}(m, m, m_2, m_2, (J + J_{g2}), m_1, m_1, J_{g1})$$

We know the acceleration of the EE ($\ddot{x}$ and $\ddot{y}$) from the cycloidal motion, we need to calculate the acceleration of the other centre of mass (g1,g2).

In the system (9), the kinematic of the system is defined.

$$\begin{cases} x_p = l_1 \cos\alpha + l_2 \cos(\alpha+\beta) \\ y_p = l_1 \sin\alpha + l_2 \sin(\alpha+\beta) \\ x_{g2} = l_1 \cos\alpha + g_2 \cos(\alpha+\beta) \\ y_{g2} = l_1 \sin\alpha + g_2 \sin(\alpha+\beta) \\ \vartheta = \alpha+\beta \\ x_{g1} = g_1 \cos\alpha \\ y_{g1} = g_1 \sin\alpha \\ \alpha = \alpha \end{cases} \tag{9}$$

The Jacobian matrix (J_e and $\dot{J}_e$) for this case are:

$$J_e = \begin{bmatrix} -l_1 \sin\alpha - l_2 \sin(\alpha+\beta) & -l_2 \sin(\alpha+\beta) \\ l_1 \cos\alpha + l_2 \cos(\alpha+\beta) & l_2 \cos(\alpha+\beta) \\ -l_1 \sin\alpha - g_2 \sin(\alpha+\beta) & -g_2 \sin(\alpha+\beta) \\ l_1 \cos\alpha + g_2 \cos(\alpha+\beta) & g_2 \cos(\alpha+\beta) \\ 1 & 1 \\ -g_1 \sin\alpha & 0 \\ g_1 \cos\alpha & 0 \\ 1 & 0 \end{bmatrix} \qquad \dot{J}_e = \begin{bmatrix} -\dot{\alpha} l_1 \cos\alpha - (\dot{\alpha}+\dot{\beta}) l_2 \cos(\alpha+\beta) & -(\dot{\alpha}+\dot{\beta}) l_2 \cos(\alpha+\beta) \\ -\dot{\alpha} l_1 \sin\alpha - (\dot{\alpha}+\dot{\beta}) l_2 \sin(\alpha+\beta) & -(\dot{\alpha}+\dot{\beta}) l_2 \sin(\alpha+\beta) \\ -\dot{\alpha} l_1 \cos\alpha - (\dot{\alpha}+\dot{\beta}) g_2 \cos(\alpha+\beta) & -(\dot{\alpha}+\dot{\beta}) g_2 \cos(\alpha+\beta) \\ -\dot{\alpha} l_1 \sin\alpha - (\dot{\alpha}+\dot{\beta}) g_2 \sin(\alpha+\beta) & -(\dot{\alpha}+\dot{\beta}) g_2 \sin(\alpha+\beta) \\ 0 & 0 \\ -\dot{\alpha} g_1 \cos\alpha & 0 \\ -\dot{\alpha} g_1 \sin\alpha & 0 \\ 0 & 0 \end{bmatrix} \tag{10}$$

We can calculate the joint position and velocity ($\alpha, \beta, \dot{\alpha}, \dot{\beta}$) using the inverse kinematic considering only the first two equation of the system (9) and of the Jacobian (J_e).

Then, having $\alpha, \beta, \dot{\alpha}\ and\ \dot{\beta}$, we can compute the Extended Jacobian and derive the complete vector $\ddot{S}$ using the forward kinematics.

Now we have all the data for eq(8) and we can easily calculate the torque applied to the system during the motion.