COMP0127 Robotic Systems Engineering
Coursework 1: Linear Algebra and Forward Kinematics

Dr. Agostino Stilli
Department of Computer Science
University College London

October 18, 2021

To get full credit for an answer, you are *required* to provide a .pdf report, and a fully working coding solution by filling in the provided code templates. These templates provide additional information on how to implement each script. **Do not remove anything from the templates, and try to only fill-in the code in the specified fields.** For the coding questions, you also are expected to include a simple breakdown of your algorithms in the report. When ready, *upload* your 'cw1' package on Moodle along with your submitted coursework report, in .zip or .rar extension. The necessary ROS packages are available on the course's *GitHub repository*.

Linear Algebra

1. a. If a matrix H satisfies the property $H^T H = I$, show that the columns of H have length 1 and they are perpendicular to each other. [report - 3 pts]

   b. Given two vectors v_1 and v_2 in R_3, show that their scalar product is invariant to rotations when the same rotation R is applied on both v_1 and v_2, i.e. their scalar product does not depend on the reference frame where v_1 and v_2 are defined. [report - 3 pts]

   c. Show that the length of a vector v does not change when a rotation is applied to it: $||v|| = ||Rv||$ [report - 3 pts]

   d. Show that the distance between two points P_1 and P_2 does not depend on the reference frame in which they are defined: $||p_1 - p_2|| = ||Rp_1 - Rp_2||$ [report - 3 pts]

   [12 pts]

2. a. Provide a matricial example, i.e. a succession of 3 matrices along the 3 different axes, of gimbal lock for the YZY (Euler, extrinsic) and XYZ (Tait-Bryan, intrinsic) rotations. Why do we need to avoid gimbal lock when controlling robotic arms? How is this achieved? [report - 4 pts]

b. Show mathematically how to pass from Quaternion representation to rotation matrix representation. [report - 4 pts]

c. What rotation representation would you suggest to use in the following cases:

- Nano-robot with very limited memory storage
- Nano-robot with very limited computational power
- Iphone navigation system
- Robotic arm wth 6 DOF [report - 2 pts]

[10 pts]

3. a. Why do we work with decompositions? What are they useful for? [report - 4 pts]

   b. Describe Singular Value Decomposition (SVD) and highlight pros and cons w.r.t. eigendecomposition. [report - 4 pts]

[8 pts]

4. Complete the following tasks by filling in the python code templates in the packages ”cw1/cw1q4_srv” and ”cw1/cw1q4” to create services that perform representation transformations.

   a. Fill in the template in package ”cw1/cw1q4_srv” with the appropriate request and response message types for each service. [code - 5 pts]

   - The quaternion message is ”geometry_msgs/Quaternion”, namely ”q”.
   - The rodrigues representation message is three ”std_msgs/Float64” values, namely ”x” ,”y” and ”z”.
   - The Z-Y-X euler angle representation message is three ”std_msgs/Float64” values, namely ”z” ,”y” and ”x”.

   b. Fill in the template in package ”cw1/cw1q4” to create a service that converts a quaternion representation to an euler angle representation $\mathbf{R}_z\mathbf{R}_y\mathbf{R}_x$. Your request should contain the quaternion you need to convert, whereas your response should store the requested euler angles. [report - 3 pts, code - 7 pts]

   c. Fill in the template in package ”cw1/cw1q4” to create a service that converts a quaternion representation to a rodrigues representation. Your request should contain the quaternion you need to convert, whereas your response should store the requested rodriguez representation. [report - 3 pts, code - 7 pts]

[25 pts]

Forward Kinematics

5. Apply forward kinematics on the KUKA YouBot manipulator.

   a. Identify the standard Denavit-Hartenberg parameters for the simplified dimensions of the Youbot shown in Figure 1. Your report should include a picture with the frames on the robot joints, as well as a brief explanation of how the parameters were derived. [report - 10 pts]

   b. Complete this task by filling in the 'cw1q5b_node.py' code template, inside the the package "cw1/cw1q5". Write a ROS script to compute the forward kinematics using the standard Denavit-Hartenberg convention. To complete this assignment, you must do the following:

      - Fill the "youbot_dh_parameters" dictionary with the youbot DH parameters you found in question 5a.
      - Implement the 'standard_dh()' function that defines the generic homogeneous transformation based on the four standard DH parameters.
      - Implement the 'forward_kinematics()' function that solves the forward kinematics by multiplying frame transformations.
      - Implement the 'fkine_wrapper()' function that integrates your robotics code with ROS and listens to the topic where joint states are published.
      - Initialize the subscriber to the topic that publishes joint states and its callback function fkine_wrapper() [report - 5 pts, code - 15 pts]

   c. Identify the standard D-H parameters following the complete Youbot dimensions found in the 'robot_description/youbot_description/urdf/youbot_arm/arm.urdf.xacro' URDF file. Based on the URDF description of each joint, you should be able to come up with a new set of DH parameters, as well as the joint offsets that the xacro file incorporates. Your report should include a brief explanation of how the parameters were derived. [report - 10 pts]

   d. Complete this task by filling in the 'cw1q5d_node.py' code template, inside the the package "cw1/cw1q5". Write a ROS script to compute the forward kinematics based on the URDF description. To complete this assignment, you must do the following:

      - Fill the "youbot_dh_parameters" dictionary with the youbot DH parameters you found in question 5c.
      - Fill the "youbot_joint_offsets" dictionary to account for the joint offsets between the "youbot_dh_parameters" you found and the xarco representation.
      - Implement the 'fkine_wrapper()' function and initialize the subscriber [code - 5 pts]

[45 pts]

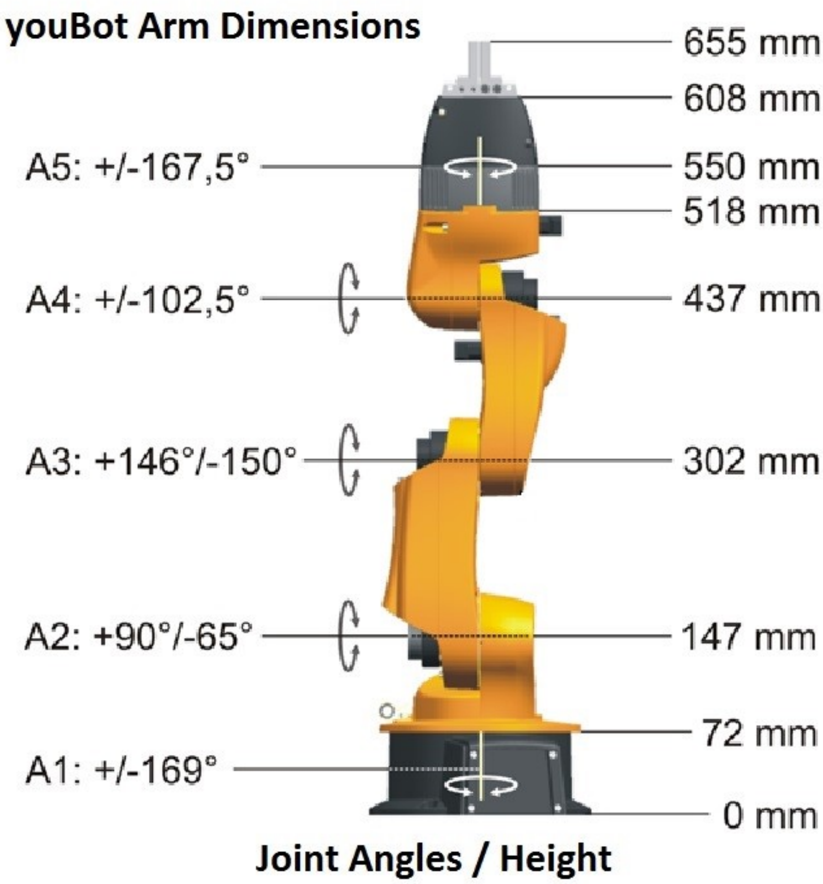

Figure 1: Kuka youBot Manipulator's simplified dimensions