EXPERIMENT 1: MANIPULATOR KINEMATICS

The purpose of this experiment is to study the forward Kinematics of 4 degree-of-freedom (DOF) serial link robots. The following topics will be studied in this experiment.

Topics Covered

- The concept of kinematics
- Coordinate frame assignment
- DH parameters and DH table
- Manipulator kinematics derivation

Prerequisites

- The robot has been setup and tested. See the Quick Start Guide for details.
- You have access to the User Manual.
- You are familiar with the basics of MATLAB® and SIMULINK®.

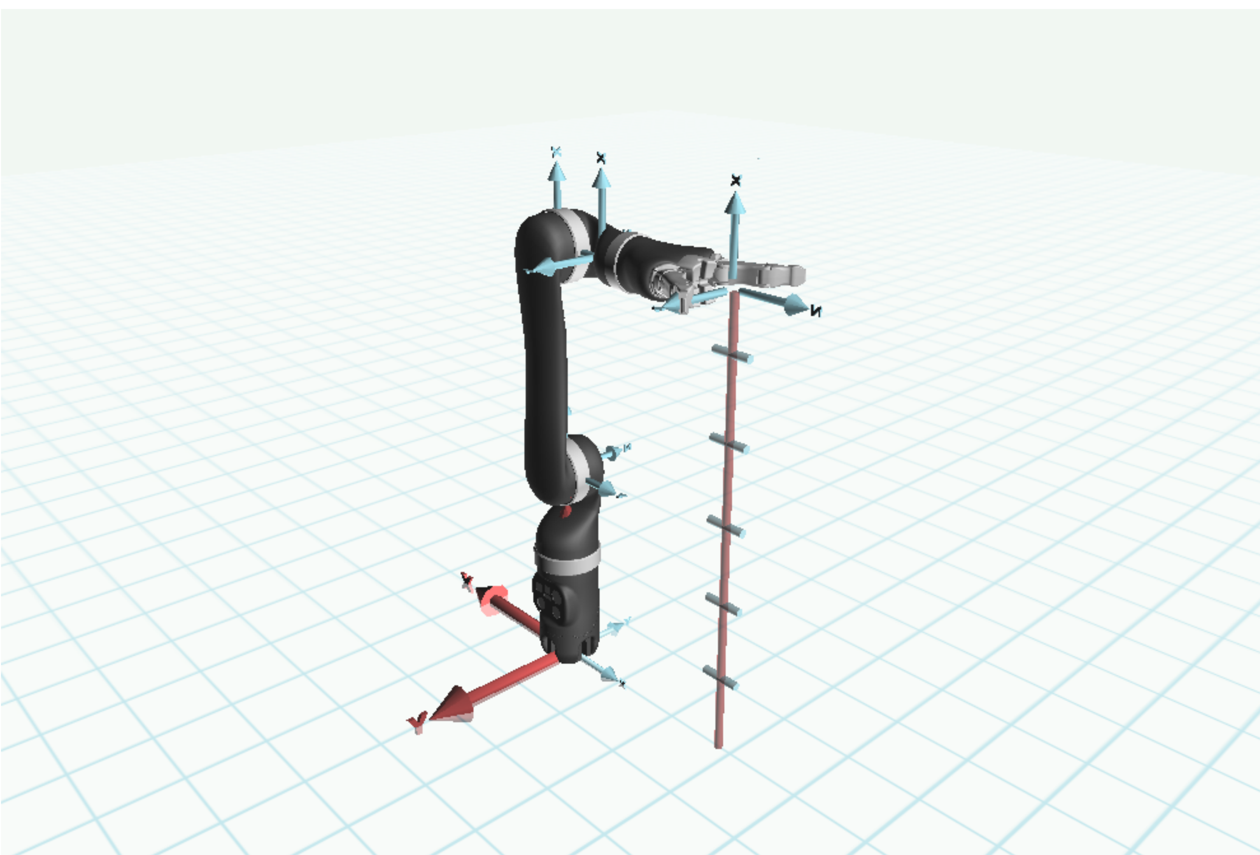

Figure 0.1: Kinematics is used to determine the position of the robot end-effector given the joint angles.

1 Pre-Lab Preparations

1.1 Background

Kinematics refers to the geometric and time-based properties of the motion of an object without considering the forces and moments that cause the motion. In this lab, we will study this relationship in the context of position and orientation of manipulator linkages and end-effector with respect to joint angles in static situations. This particular aspect of kinematics is called forward position kinematics.

In order to analyze geometrically complex manipulators, coordinate frames are attached to various parts of the manipulator including the base frame of the robot which is a fixed coordinate frame, and the end-effector frame of the robotic arm which is attached to the robot end-effector. The study of manipulator kinematics describes how the location and orientation of these frames vary in different configurations, based on the joints angles of the robot arm.

1.1.1 Coordinate Frame Assignment and DH parameters

Robotic manipulators are generally constructed from joints and links. Joints can be *revolute*, meaning they rotate, or *prismatic*, meaning they linearly slide. Each joint is considered to be a single degree-of-freedom (DOF). In this lab, we consider a 4DOF manipulator with four revolute joints. The links of the robot are numbered from the fixed base, L_0, all the way to the end-effector, L_4. Note that for the 4-DOF MICO arm, we have $L_0 + L_1 = 0.2755$ m, $L_2 = 0.29$ m, $L_3 = 0.1233$ m, and $L_4 = 0.16$ m.

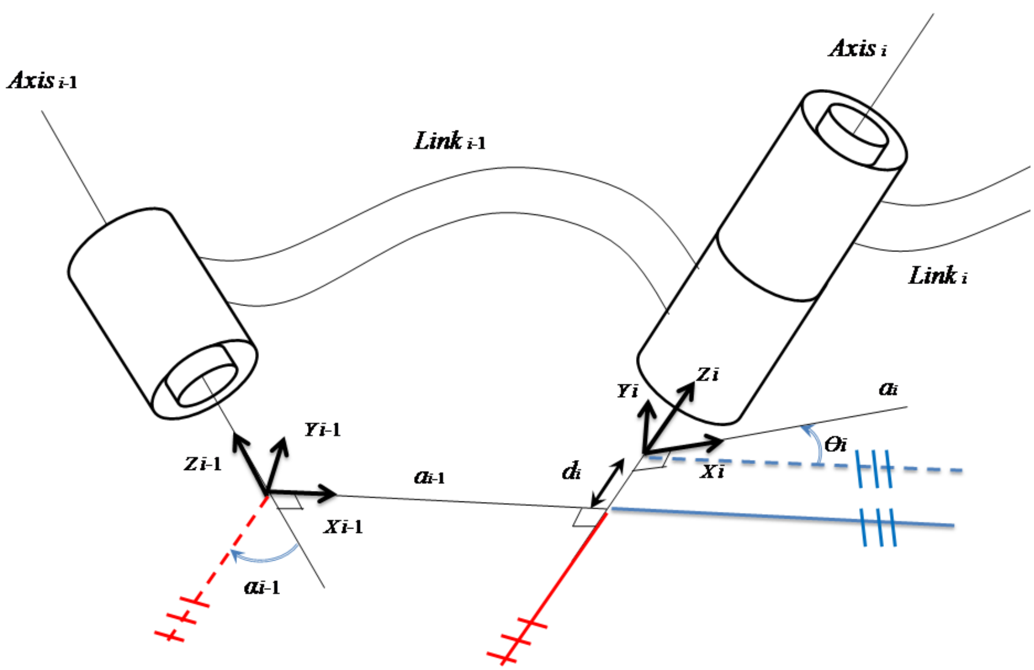

Figure 1.1: Link frame transformation.

Each joint of the manipulator has an axis of rotation (indicated by z_i where i is the joint number) called the *joint axis*. Each link of the manipulator is considered a rigid body that defines the relationship between the two neighbouring joint axis.

Consider the two neighbouring joint axes in Figure 1.1. To determine the relative position and orientation of two fixed axes (that do not move), only two parameters are required: *link distance* or *link length* (also called common normal), and the *link twist*. Link distance, a_{i-1}, is measured along a line that is mutually perpendicular to the joint axes of neighbouring joints, $i-1$ and i. Link twist, α_{i-1}, is defined as the angle between the two joint axis, $i-1$ and i about the common normal (if the axes are parallel, then $\alpha_{i-1} = 0$).

If additional axes are considered (three or more fixed axes), in general case the common normals do not intersect the common axes at the same time. Therefore, a new parameter is needed to describe the relationship between the neighbouring axes, which is called *link offset*. Link offset, d_i is the distance between the common normals a_{i-1} and a_i along the axis i.

If the axes are not fixed, a final parameter is required to describe the relationship between the joint axes, called *joint angle*. The joint angle, θ_i is the angle between the common normals a_{i-1} and a_i about axis i. These four parameters are the *Denavit-Hartenberg* or *DH* parameters of the robot.

As was mentioned earlier, a coordinate frame is attached to each link in order to describe the location and orientation of each link relative to its neighbours. Here, we assume that when all joints are zero, the x_0 axis is towards the front of the robot, y_0 is towards left, and z_0 is upward (to be aligned with the first joint axis of rotation).

Follow the steps below to attach frames to the robot links.

1. To each link i assign a frame i.
2. z_i axis is the joint axis (of rotation).
3. The origin of the frame i is chosen at the intersection of joint axis i and the common normal a_i (or at the intersection of joint axes i and $i+1$ if a_i = 0).
4. x_i is placed along the common normal a_i pointing to the joint $i+1$. If a_i = 0 the axis x_i is perpendicular to both z_i and z_{i+1} and direction is arbitrary; preferably select it along x_{i-1} when $\theta_i = 0$.
5. y_i can be achieved with the right hand rule.

With the above frame assignment, a_{i-1} is the distance between z_{i-1} and z_i along x_{i-1}, α_{i-1} is the angle between z_{i-1} and z_i about x_{i-1}, d_i is the distance between x_{i-1} and x_i along z_i, and θ_i is the angle between x_{i-1} and x_i about z_i.

Note: There are multiple standards in assigning frames and DH parameter derivation. This lab follows the standard proposed by J.Craig[1].

1.1.2 Forward Kinematics Using DH Parameters

After multiplying the matrices of rotations and translations related to the coordinate frames attached to robot links, the transformation matrix for two consecutive link frames (frame i to frame $i-1$), with the defined DH parameters, is defined as follows:

$$ {}^{i-1}_{i}T = \begin{pmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & a_{i-1} \\ s_{\theta_i}c_{\alpha_{i-1}} & c_{\theta_i}c_{\alpha_{i-1}} & -s_{\alpha_{i-1}} & -s_{\alpha_{i-1}}d_i \\ s_{\theta_i}s_{\alpha_{i-1}} & c_{\theta_i}s_{\alpha_{i-1}} & c_{\alpha_{i-1}} & s_{\alpha_{i-1}}d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.1} $$

1.2 Pre-Lab Exercise

1. Using the procedure outlined in the background section for attaching coordinate frames to robotic manipulators, assign appropriate coordinate frames to the schematic of the MICO robotic manipulator in Figure 1.2. The base and end-effector of the arm are shown with red dots, and some axes are shown for simplicity.

[1] J. Craig: Introduction to Robotics, Addison-Wesley, 1986.

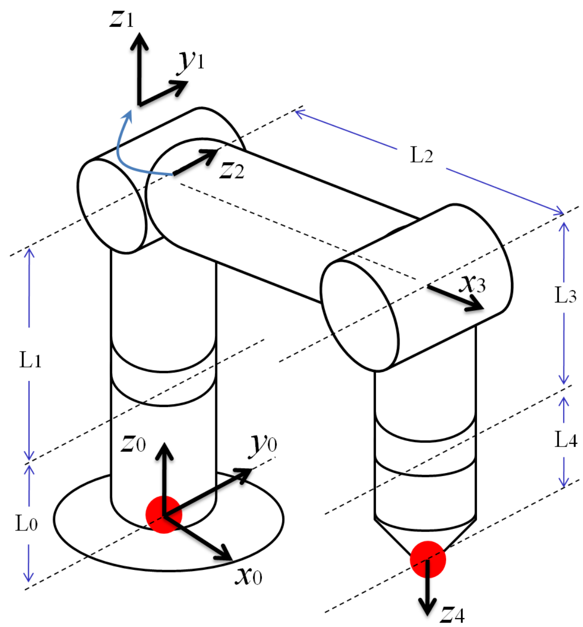

Figure 1.2: Schematic of the 4-DOF MICO robotic manipulator.

2. Fill in the DH parameters of the robot in Table 1.1 according to the frames you assigned in Question 1.

Table 1.1: DH table for the 4-DOF MICO robotic manipulator

i	α_{i-1}	a_{i-1}	d_i	θ_i
1				
2				
3				
4				

3. Derive individual link transformation matrices using Equation 1.1. Determine the portion of the matrices that represents the rotation matrices and translation vectors.
4. How would you derive the forward position kinematics of the MICO 4-DOF robotic manipulator?

2 In-Lab Exercise

2.1 Simulation

The QUARC model for this exercise is called ”MICO_Kinematics_Simulation.mdl” a snapshot of which shown in Figure 2.1.

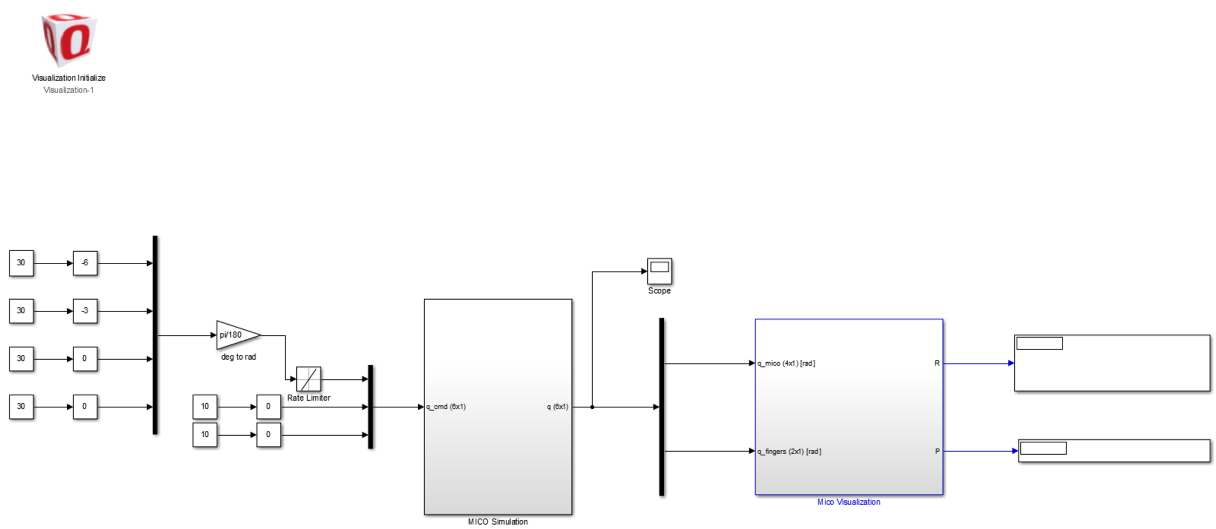

Figure 2.1: Snapshot of the controller model ”MICO_Kinematics_Simulation.mdl”

The ”Mico 4DOF FPK - Extended” block, inside the ”Mico Visualization” subsystem shown in blue, receives the joint angles as input and computes the position of the end-effector. Compile and run the model, then follow the procedure outlined below. Observe the motion of the robot, and answer the associated questions.

1. Describe the configuration of the robot when all joint angles are zero. You can use keyboard to view the robot from different angles. (W for zooming in, S for zooming out, A for moving to the left, D for moving to the right. Hold down the middle mouse button, and use the mouse to rotate. ESC will reset the view)
2. Press ESC key to reset the view. Using the slider gains 1, 2 and 3, move the first joint to +90 °, the second joint to -30 ° and joint 3 to zero. Looking at the virtual robot, determine the position of the robot end-effector (the distance between the grid lines is 10 cm).
3. Apply Equation 1.1 to all links using the angles specified in Question 2, and robot’s DH parameters, to verify the values for (x, y, z) (use MATLAB for matrix multiplications).
4. Modify joints 2 and 3 to raise the z value to 30 cm. What are the resultant joint angles? Can you achieve the same x, y and z position using a different joint configuration?
5. Modify the command to the last joint (θ_4) and observe its effect on the virtual robot. Does the last joint angle affect the position or orientation of the end-effector of the robot? Observe the motion of the robot as you change the angle of each joint, and outline how each joint angle changes the position and orientation of the end-effector?

2.2 Experiment

In this section, you will experimentally evaluate the forward kinematics of the 4DOF MICO. While working with the robot, we recommend that you always keep a safe distance from the robot, and test your position commands on the virtual robot first before applying them to the actual robot.

The QUARC model for this experiment is called "MICO_Kinematics_Experiment.mdl" shown in Figure 2.2. Before running the model, manually move the robot arm into a pose where the joints have a reasonable range of motion and the robot end-effector is away from the table (preferably similar to Figure 0.1). You can move the arm easily with the power off. Turn on the robot power to hold the initial pose. Before running the model, make sure that the robot is disabled (the yellow manual switch, is set to "Disabled").

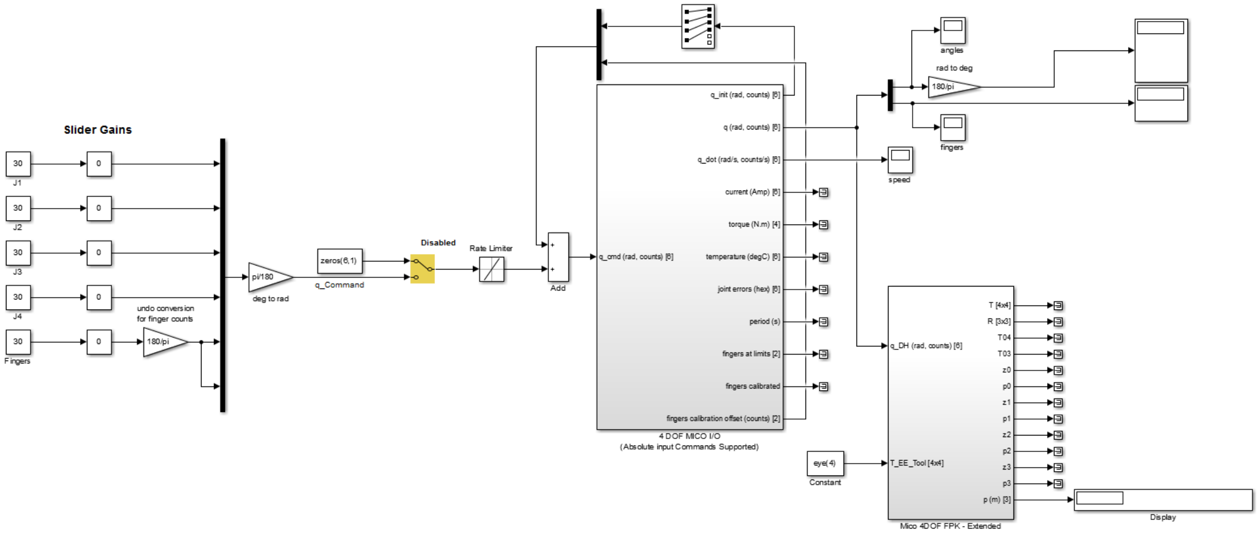

Figure 2.2: Snapshot of the controller model "MICO_Kinematics_Experiment.mdl"

Compile and run the model. Follow the steps below and answer the questions.

Caution **Be sure to set the Ports in the 4-DOF MICO I/O block to the correct ports for your serial card. For more information, refer to the User Manual.**

1. Move the first joint to +90 °, the second joint to -30 ° and joint 3 to zero by setting the command inputs. Looking at the virtual robot, determine the position of the robot end-effector (the distance between the grid lines is 10 cm).

2. Compare the position and orientation of the robot with the virtual robot from the simulation. Do they match?

3. With the last joint angle is set to zero, gradually change the first three joints of the robot so that the last joint axis of the robot (z_4 coming out of the hand parallel to the last joint axis) becomes parallel to the global x axis (If you forgot the frame axes directions, run the simulation model first). Derive the transformation matrix (${}_4^0T$) in this configuration using the joint angles, and discuss the rotation matrix (${}_4^0R = {}_4^0T(1:3, 1:3)$). Verify that the rotation matrix you derived with the output of the"MICO 4DOF FPK - Extended" block.

Quanser Inc.
119 Spy Court
Markham, Ontario
L3R 5H6
Canada
info@quanser.com
Phone: 1-905-940-3575
Fax: 1-905-940-3576

Printed in Markham, Ontario.

For more information on the solutions Quanser Inc. offers, please visit the web site at:
http://www.quanser.com