QArm Lab Procedure

Workspace Identification

Setup

1. Launch Quanser Interactive Labs and load the QArm Workspace.
2. Launch MATLAB and browse to the working directory for Lab 2 – Workspace Identification.

Forward Kinematics

1. Open the ForwardKinematics.slx model via MATLAB. This will be used to implement and validate your forward kinematics formulation. The 'Task Space' section will display the position and the orientation matrix of the end-effector frame with respect to the base frame corresponding to the static joint configurations described in the 'Joint Space' section, as shown in Figure 1.

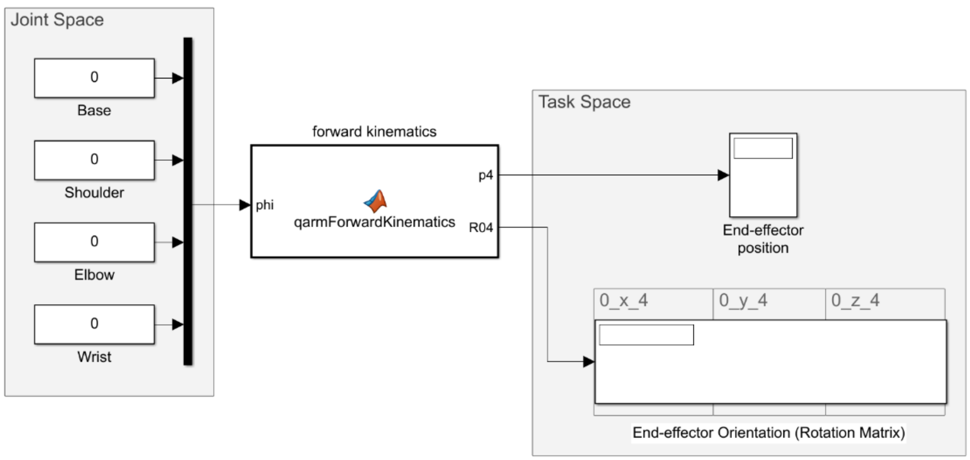

Figure 1: Forward Kinematics Simulink model.

2. Double-click the function labelled *qarmForwadKinematics*. This MATLAB function is incomplete and must be completed before use.
3. Using the transformations from Table 2 in the Workspace Identification Concept Review, complete lines 20-24 and 27-30 in the *qarmForwardKinematics* function. Note that $l1$, $l2$ and $l3$ correspond to λ_1, λ_2 and λ_3. Also, change lines 33-37 to correctly convert joint angles from **phi** to **theta**.
4. Complete the DH Table below for the QArm manipulator based on the schematic in Figure 1 from the Workspace Identification Concept Review document.

i	a_i	α_i	d_i	θ_i
1				
2				
3				
4				

5. Use the DH table from step 4 to complete lines 42 to 45 in the *qarmForwardKinematics* function. Note that the *quanser_arm_DH* function call is described in line 41. Lastly, leave the theta parameter in lines 42 to 45 as $theta(i)$ where i corresponds to the correct index 1 to 4.

6. Before you proceed, you must complete the ***quanser_arm_DH*** function. You can find the function definition within ***qarmForwardKinematics*** at line 61. This function calculates and returns the homogenous transform ${}^{i-1}T_i$ given the DH parameters a_i, α_i, d_i and θ_i as input parameters. Complete the correct definitions for $\mathrm{T\_R\_z}$, $\mathrm{T\_T\_z}$, $\mathrm{T\_T\_x}$, and $\mathrm{T\_R\_x}$ in terms of the inputs a, $alpha$, d and $theta$. Lastly, multiply the four individual transformations in the correct order to find the return transform T.

7. Now that you have completed ***quanser_arm_DH***, lines 42 to 45 will be functional. Complete lines 47, 48 and 49 to correctly calculate $\mathrm{T02}$, $\mathrm{T03}$ and $\mathrm{T04}$ in terms of $\mathrm{T01}$, $\mathrm{T12}$, $\mathrm{T23}$ and $\mathrm{T34}$. Note that TAB in MATLAB code corresponds to ${}^{A}T_B$.

8. Lastly, complete lines 54 and 57 to extract the correct elements of the $\mathrm{T04}$ transform.

9. Close ***qarmForwardKinematics***. Run the model using the green Play button under the Simulation Tab of your model With the Joint Space set to $[0 \quad 0 \quad 0 \quad 0]^T$, the end-effector position and orientation in the Task Space section should display the following

$$p_4 = \begin{bmatrix} 0.45 \\ 0 \\ 0.49 \end{bmatrix}, \qquad {}^{0}R_4 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{1}$$

Verify the position parameters from the schematic 1.1 in the Workspace Identification Concept Review.

Workspace Identification

1. Copy the ***qarmForwardKinematics*** function that you have completed. Ensure that you keep a copy of your files for future use. Close ForwardKinematics.slx and open WorkspaceIdentification.slx. Replace the ***qarmForwardKinematics*** function with the competed version.

2. Open the subsystem ***Keyboard I/O***. This subsystem shown in Figure 3 contains an incomplete section called Map keys to joint/gripper speeds. You will need to resolve the mapping logic and fill out the values in gain blocks and the bias block. Can you describe what this section is doing and why?

   HINT: Double-click the Host Keyboard block and take a note of which arrow keys are being read from the keyboard and in what order. Now close the block and double click on the demux block (black bar with 1 input and 2 outputs). This tells you how the 3 signals are being broken down into a set of 2 in the first output and a single signal in the second output. You should be able to track the second output to the Gripper (%) command and the first output to the Phi Cmd (rad) command.

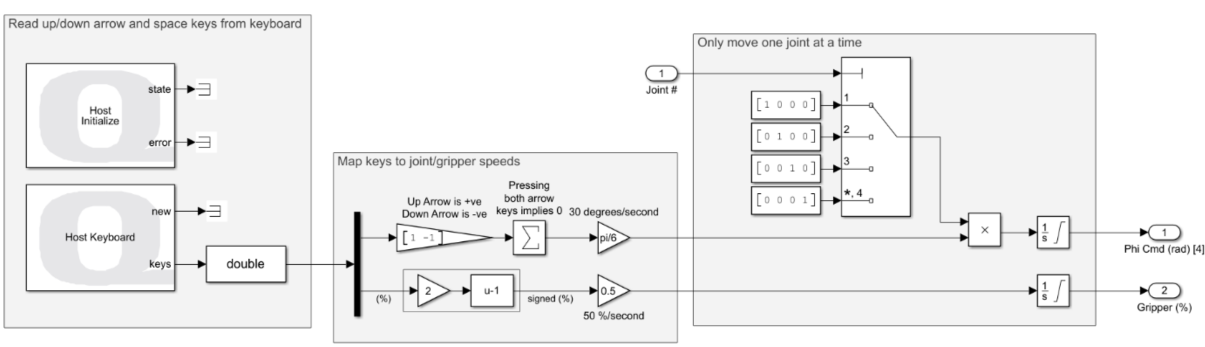

Figure 1.3: Correct Keyboard I/O subsystem.

3. Prior to running the model, open the model's Configuration Parameters and verify that they are configured as follows:

   a. Solver type: Fixed-step

   b. Solver: ode4 (Runge-Kutta)

   c. Fixed-step size (fundamental sample time): 500 Hz.

4. Run the model using the green Play button under the Simulation Tab of your model. Once started, the model will command 0 rad angles to all four of the manipulator's joints.

5. Change the number in the Joint constant parameter to select which joint you want to move. Use the up/down arrow key on your keyboard to move that joint and validate the positive direction convention. Use space to close or open the gripper. To select a different joint, change the Joint constant and repeat, as necessary.

6. Explore the minimum and maximum ranges of all joints. For example, have the elbow completely open (joint 3) and move the shoulder (joint 2) to have the arm completely horizontal. In this configuration, move the base (joint 1) between its minimum and maximum limits.

7. Stop the model.

8. In MATLAB, right-click visualize_data.m and click run, which will spawn two plots.

   The first plot (Figure 1: QArm Workspace) provides a 3D plot of the end-effector trajectory. It draws the end-effector trajectory of the manipulator in 3D to help you visualize the end-effector motion and its extremes. Rotate the 3D plot using the tool at the top right of the plot. Take note of the workspace limits, and take screenshots of the X vs Y, Y vs Z, and X vs Z planes.

   The second plot (Figure 2: End-effector Trajectory) displays the Task Space trajectories that the end-effector went through. Find the maximum x, y, and z ranges of the manipulator's end-effector. Take note of these extremes.

9. If you are not satisfied with the plots you visualize, re-run steps 5-6.

10. In the MATLAB Workspace, select the variables t, x, y, and z. Right-click and then select 'Save As…'. Save the data as **myData.mat** for future use.

11. Close all models and Quanser Interactive Labs.