The DOBOT CR3 (pictured in Fig. 1) is a collaborative robot (co-bot) with 6 axes and a maximum payload of 3 kg. The schematic in Fig. 1(b) shows the robot in its home configuration.

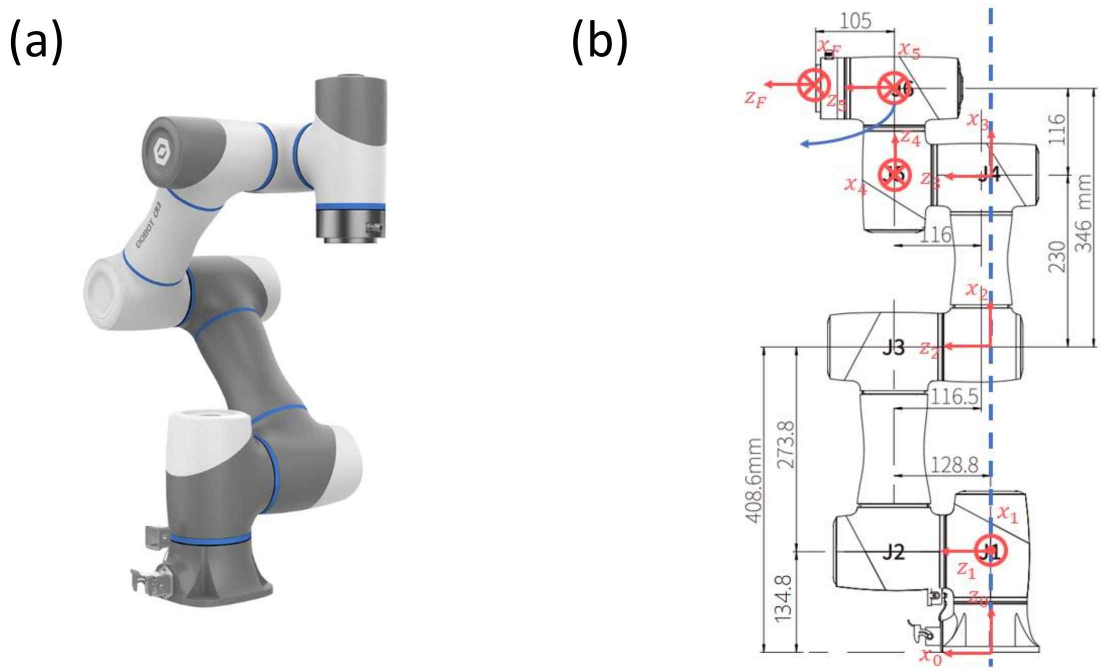

*Figure 1: (a) The CR3 robotic arm; (b) Robot schematic with dimensions. All dimensions are in meters. Frame {0} at the base of the robot is the space frame. Frame {F} is the end-effector frame. **Note**: while the schematic shows joint frames, we are not going to use them in this assignment. We will calculate the robot kinematics using the product of exponentials formula, which only requires knowledge of the joint axes ($z_1, \dots, z_6$).*

Questionnaire:

1. **Draw the kinematic chain of the robot**[1]
   Clearly note relevant dimensions, joint axes, and the world and end-effector frames.

2. **Calculate each of the screw axes $\xi_i = (\omega_i, v_i)$ with respect to the space frame**
   Show all the vector calculations needed to derive the screw axes.

3. **Calculate the manipulator kinematics using the product of exponentials formula**
   Open the starting code that was provided to you. Find the script called `poe.m`, open it, and start with your code. You can now run the script to create the kinematic chain of the robot. Complete this script to perform the calculation of the forward kinematics using the product of exponentials. *Note #1*: *For this and the following questions, you are welcome to reuse code that you developed for homework 2. Note #2: This script is expected to create a variable S containing the screw axes and a variable M containing the home*

[1] i.e., a diagram like the one shown in Fig. 4.3 in Modern Robotics.

configuration. If these variables are not created, the MATLAB test script used to grade exams will be unable to grade your submission.

4. **Calculate the space Jacobian of the manipulator**
   Open the starting code that was provided to you. Find the script called `jacobian.m`, open it, and start with your code. Complete this script to perform the calculation of the robot Jacobian matrix when the robot is in the home configuration. ***Note***: *This script is expected to create a variable J containing the Jacobian matrix expressed in the space frame*. *If this variable is not created, the MATLAB test script used to grade exams will be unable to grade your submission.*

5. **Calculate the inverse kinematics for a heart-shaped path**
   Open the starting code that was provided to you. Find the script called `ik.m`, and run it. A new window will appear (see Fig. 2), showing (1) the robot in its home configuration, and (2) a heart-shaped path composed of 36 points. Your goal is to implement the inverse kinematics of the robot and to make the robot trace the path with the end effector. Use an inverse kinematics method of your choice. The code is expected to create a $36 \times n$ `qList` matrix where each row is a set of joint variables necessary to trace the path. To demonstrate that your code is working, create an animation in MATLAB using the following line of code:

```
panda.plot(qList, 'trail', {'r', 'LineWidth', 5});
```

It will display an animation like the one shown here.

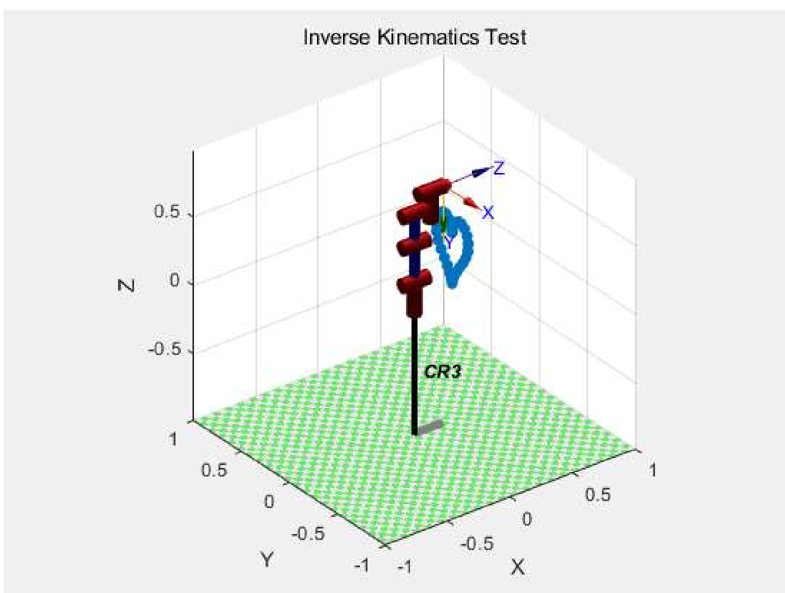

Figure 2: CR3 robot and heart-shaped path that we wish the robot to trace.