Rigid Transformations & Forward Kinematics

Last update: 2021-04-21

Gen3 lite Cursus

KINOVA

Table of contents

Objectives

During this lab, we will build a model for the forward kinematics of Gen3 lite using rigid transformations, a formalism less systematic than the Denavit-Hartenberg method, but which gives more flexibility on the definition of the reference frames used. Then, we will validate our model experimentally by sending commands to the robot.

About Gen3 lite

All the information required to build the kinematic model of Gen3 lite is available in the User Guide in the section *Guidance for advanced users*, on pages 122-123. For convenience, the actuator frames used to generate the transformation matrices given by the User Guide are transcribed in Figure 1.

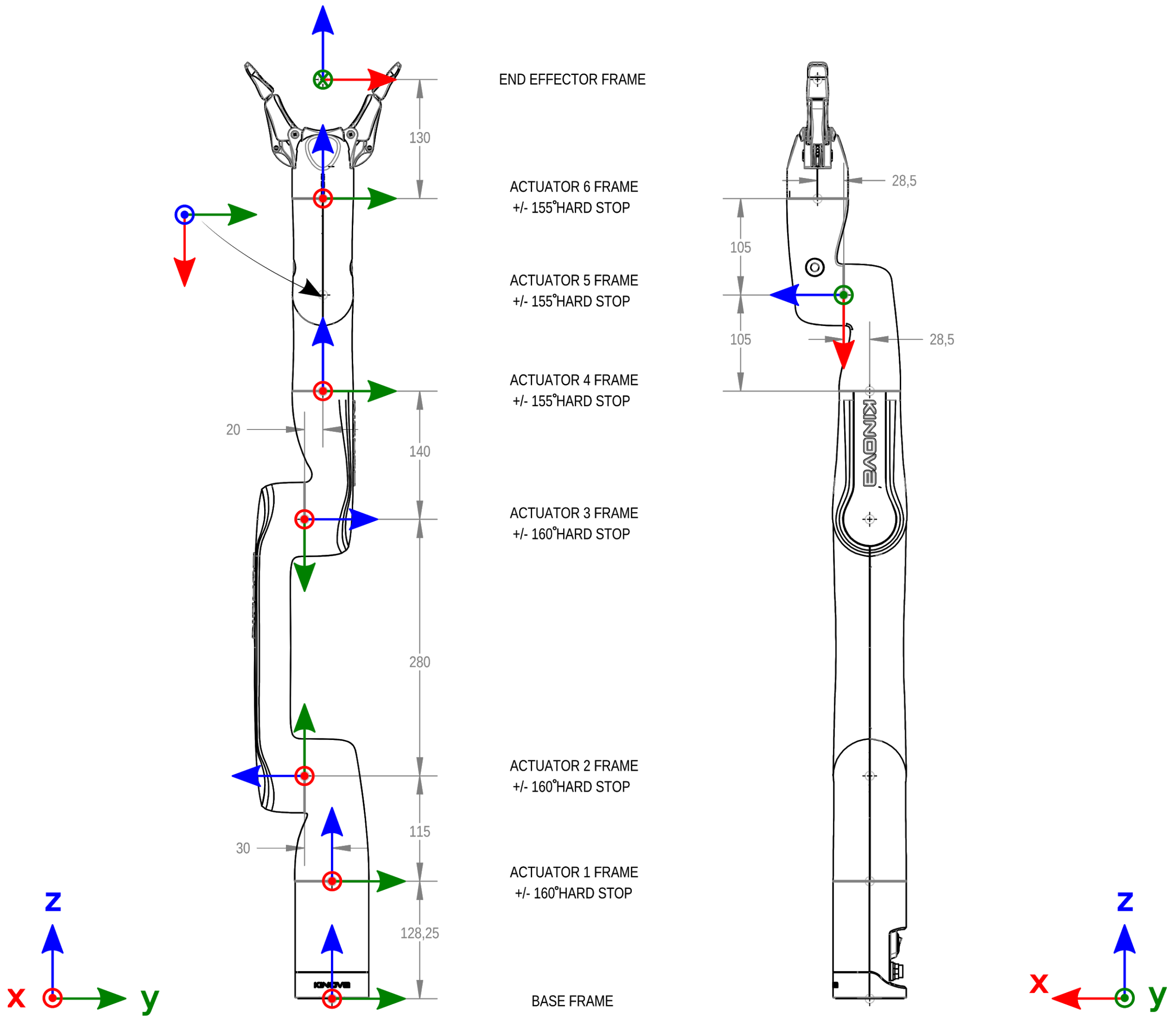

FIGURE 1: Actuator frames and dimensions

Theory

To obtain the full kinematic model of the robot using the rigid transformation formalism, one must first place a global fixed reference frame (the Base frame in the case of Gen3 lite), then one additional frame for each joint. These don't need to be directly on the joints, although it does help a lot when the time comes to compute the transformations. Additional frames, fixed in one joint frame, can be added for convenience, such as the Tool frame fixed in the frame of the last joint.

Then, the homogeneous transformation matrices from each consecutive frame must be assembled on a case by case basis. Finally, these matrices can be chained to obtain the full model.

Manipulations

Part 1: Implementation of the forward kinematics

In this section, we will use the data from *Guidance for advanced users* section of the User Guide to implement the calculations necessary to compute the forward kinematics in MATLAB.

1.1 Explain how the matrices ${}^{i-1}\mathbf{T}^*_i$ are obtained from ${}^{i-1}\mathbf{T}_i$ in the Transformation matrices table.

1.2 Write a function called **transf_rt**(i, $\boldsymbol{\theta}_i$), which takes as inputs the index of a reference frame and the position of the corresponding joint to output the homogeneous transformation matrix between this frame and the previous one. You can reuse the information from the Transformation matrices table.

1.3 Use the previous function to create a new function called **fk_gen3_lite_rt**($\boldsymbol{\theta}$), which takes as an input a vector containing the position of each joint.

1.4 Write a new function called **cart2pose**(x,y,z, $\boldsymbol{\theta}_x$, $\boldsymbol{\theta}_y$, $\boldsymbol{\theta}_z$) which takes as inputs the elements of the pose returned by Gen3 lite and returns the corresponding homogeneous transformation matrix.

Part 2: Validation with the robot

In this section, we will write a script to send angular commands to the robot and compare the cartesian pose calculated by the robot to the one obtained from our own calculation. You should reuse the code that we developed during the *Introduction to Gen3 lite* lab. As a reminder, connection to the robot is established via the **CreateRobotApisWrapper**() function, angular commands are sent using **ReachJointAngles**() and the cartesian pose readings are obtained via **RefreshFeedback**(). For more detail on the MATLAB API, consult the documentation.

Reminder: All angle values sent to or received from Gen3 lite should be expressed in degrees.

2.1 Adapt your script to go to each of the angular configurations defined in Table 2 and record the cartesian pose returned by the robot. Convert these poses to homogeneous transformation matrices.

Test #	$\boldsymbol{\theta}$ (deg)
1	[0, 0, 0, 0, 0, 0,]
2	[90, 0, 0, 45, 45, 45]
3	[0, 344, 75, 0, 300, 0]
4	[7, 21, 150, 285, 340, 270]

TABLE 2: Angular positions to test

2.2 Use your **fk_gen3_lite_rt**($\boldsymbol{\theta}$) function developed in the previous section to compute the pose of the robot yourself and compare it to the results obtained experimentally. **Hint:** A good way to compare two rotation matrices is to take the dot product between corresponding columns. Similar orientations should yield dot products close to 1.

2.3 How do you explain the difference between the experimental results and what you obtained with your own calculations?

2.4 What would be an advantage of using the rigid transformation method over the Denavit-Hartenberg method?

Acknowledgements

This document was produced in collaboration with the following establishments

POLYTECHNIQUE MONTRÉAL

UNIVERSITÉ D'INGÉNIERIE

Including direct implication from the following people:

- Prof. David Saussié, Eng., M.A.Sc., Ph.D., Department of Electrical Engineering.
- Alexis Tang, M.Eng., Department of Electrical Engineering
- Prof. Jérôme Le Ny, Eng., M.A.Sc., Ph.D., Department of Electrical Engineering.
- Prof. Richard Gourdeau , B.A.Sc., M.A.Sc., Ph.D., Department of Electrical Engineering.