Design of a Pick and Place Controller for the Kinova Gen3 Arm with an Educational Tutorial

Group 23: Abdullah Barat, Daljot Gill, Cassandra Gut, Cyril Justin, Loida Perez, Usama Rasheed
Schulich School of Engineering, University of Calgary

SCHULICH School of Engineering

UNIVERSITY OF CALGARY

Introduction and Methodology

The focus of our capstone was to design a pick and place controller along with a GitHub tutorial to help educate robotic enthusiasts on the robotic and mathematical concepts behind our project.

The design of the pick and place controller required us to use an existing arm (Kinova Gen 3 Robotic Arm with 7 DOF) and implement code to allow the arm to detect, pick up, and place a cube. It's based off an existing project involving a 6 DOF robot arm [1].

The project is broken into two parts, one part focusing on the pick and place controller, and the other focusing on the GitHub Tutorial.

The tutorial is designed for 1st and 2nd year mechanical engineering students to learn about all the robotics behind the design of our pick and place controller. Because it is designed for 1st and 2nd years, the content will be designed for them in mind and thus it will not go too far into the derivations and origins of certain equations.

Forward Kinematics

Forward kinematics is a method used in robotics to determine the position and orientation of the end-effector (such as a robot arm or a character's hand) based on the given joint angles or configurations of the system. It essentially calculates how the parts of a mechanism move relative to each other, tracing the path from the base to the end-effector in a sequential manner.

Developing the relationship between the robot's joint positions and the end effector's pose includes:

1. Define Elementary Transformation Matrices
2. Define Transformation Matrices from Joint to Joint
3. Input Desired Joint Angles
4. Output end-effector orientation

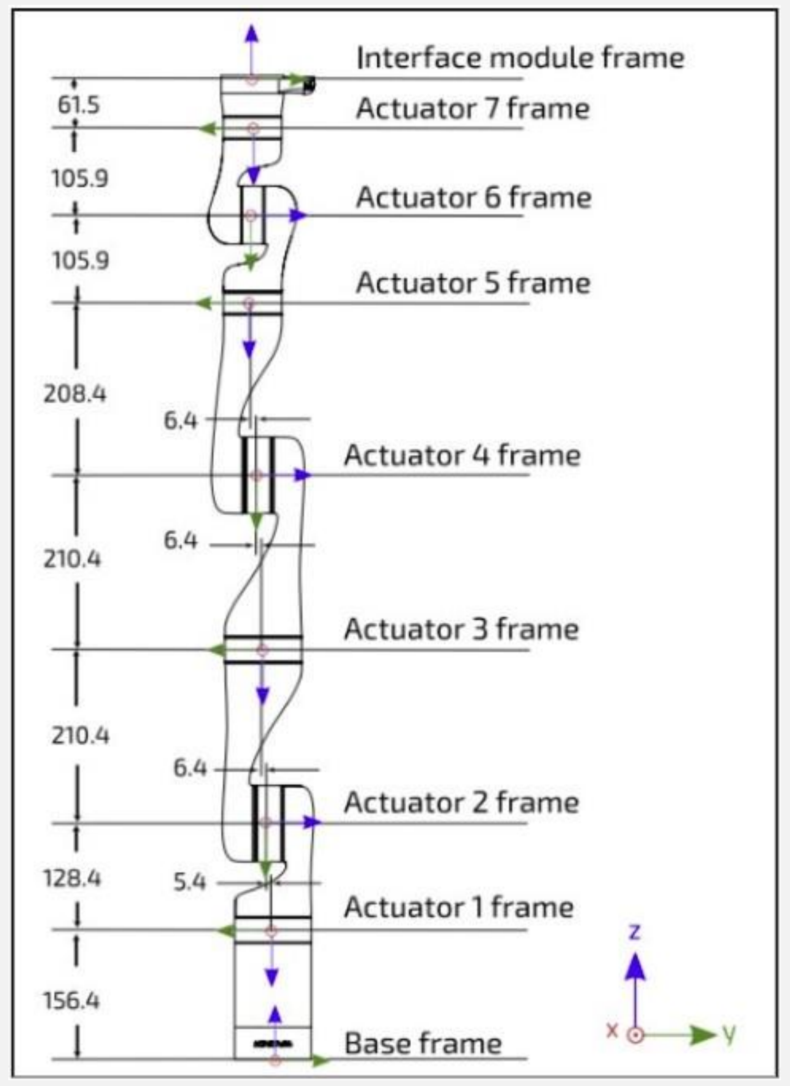

Figure 1: Schematic of the 7DOF Kinova Robot Arm and the Joint Offsets [2]

Inverse Kinematics

Inverse kinematics is a computational technique used in robotics to determine the joint angles required to achieve a desired position and orientation of the end-effector. It essentially works backwards from the desired end position, calculating the joint configurations needed to reach that position, considering constraints and limitations of the system.

Numerical inverse kinematics as opposed to analytical inverse kinematics uses an iterative technique. Quadratic error term $E = \frac{1}{2} e^T W_e e$ describes the error between the current end-effector pose and desired end-effector pose.

The three numerical Inverse Kinematics (IK) techniques we are using to minimize the error function are:

1. Newton-Raphson
2. Gauss-Newton
3. Levenberg-Marquardt methods (Wampler, Sugihara and Chan)

Jupyter Notebook Tutorial

Our comprehensive Jupyter Notebook tutorial provides a step-by-step guide to learning the mathematical theory and pick and place controller execution of our interactive Python programming.

The Jupyter Notebooks act as an interactive gateway for users to gain proficiency in editing and executing Python code while learning fundamental concepts in robotics,

The tutorial is divided into three main parts, each of which is further subdivided into subsections to provide a more detailed breakdown of information. The three topics of each part include:

1. Forward Kinematics
2. Inverse Kinematics
3. Design and Implementation of the Pick and Place Controller

The project scope and Jupyter Notebooks can be found on our project's GitHub repository. Scan the QR code in figure 2 for access.

Figure 2: Project GitHub QR Code

Usability Survey

To measure the success and usability of the Jupyter Notebooks tutorial, we conducted a survey with a small survey size to give an appropriate measure of the level of complexity in the engineering and mathematical concepts. From this feedback, the tutorial was modified to fit user's needs.

The figures below quantify survey results based on a user response scale rating from 1 – 5 with the label descriptions varying for each question.

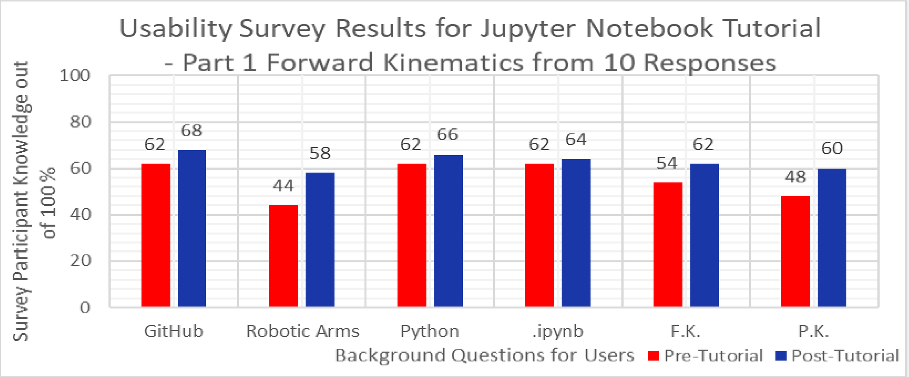

Figure 3: Averaged Usability Survey Responses For Jupyter Notebook Tutorial on Forward Kinematics

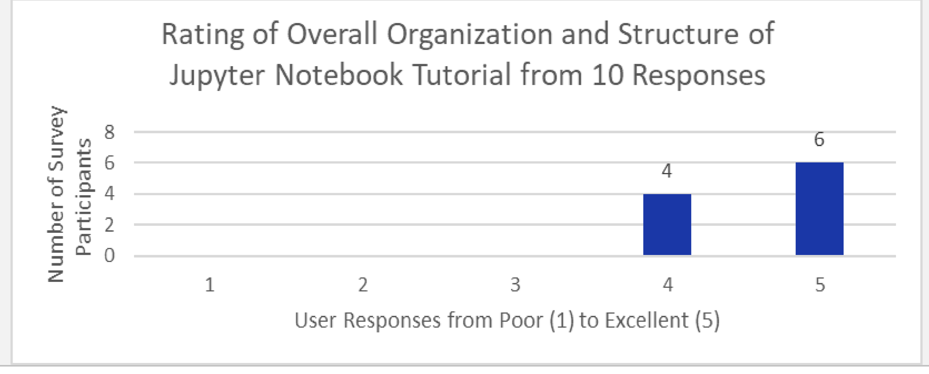

Figure 4: Final Survey Results for Organization and Structure for Jupyter Notebook Tutorial on Forward Kinematics

Conclusion

Our Capstone journey immersed us in the complexities of robotic control and programming, culminating in the successful creation of a Pick-and-Place controller for the Kinova Gen3 Robotic arm, alongside an interactive educational tutorial on Jupyter notebooks.

This comprehensive resource empowers robotics enthusiasts with a systematic understanding of fundamental concepts and practical implementation techniques, including serial link manipulators. By conducting multiple waves of public surveys, we iteratively refined and enhanced the tutorial based on valuable feedback, ensuring an optimized learning experience for all learners.

As technology advances, robotics promises to revolutionize our lives, seamlessly integrating into daily tasks and enhancing efficiency across industries, from autonomous vehicles to smart manufacturing and beyond. This innovation will shape a future where intelligent machines coexist harmoniously with humanity, leading to unprecedented possibilities and progress.

CONTACT
Cassandra Gut
Email: cassandra.gut@ucalgary.ca
Phone: (403)-796-8911
Website: https://www.linkedin.com/in/cassandra-gut/

References

1. Haviland, J. (2023, December 5). A Tutorial on Manipulator Differential Kinematics. GitHub. https://github.com/jhavl/dkt#a-tutorial-on-manipulator-differential-kinematics
2. Kinova Robotics. (2022). KINOVA Gen3 Ultra lightweight robot User Guide [Manual] Kinova Robotics. https://www.kinovarobotics.com/en/resources/gen3-user-guide