Robotic Manipulators Project: Step #2

ECE 9053A "Robot Manipulators"

Fall 2024

Objective

The purpose of this project is to learn how to implement the kinematics scripts in MATLAB. You are required to continue using the manipulator and application selected in Step #1. Please note that you will continue to use these scripts in the next two steps, so it is important to test that they are working properly.

NOTE: If you chose a manipulator with more than 3 degrees of freedom, you do not need to model all of the joints. Select which portion of the manipulator you will be modelling and justify your decision. Ideally, you are modelling the three degrees of mobility that relate to positioning of the end-effector.

Instructions

1. **Cover page.** List project title and student name.

   Recommended length: 1 page

2. **Forward Kinematics.** Compute the forward kinematics of the manipulator. Create a MATLAB function for the forward kinematics (e.g., $T = \text{forwardkin}(q)$ - feel free to use any name you wish for these functions, these are just examples). Test that your forward kinematics script works by giving it 3 different vectors of joint variables as input and confirming that the position and orientation of the resulting transformation matrix match the assigned frames.

   Recommended reporting instructions: 1 - 2 pages in the main document demonstrating that it works. As many pages as needed in the appendix with the MATLAB code.

3. **Inverse Kinematics.** Compute the inverse kinematics of the manipulator. Create a MATLAB function for the inverse kinematics (e.g., $q = \text{invkin}(T)$). Test that your inverse kinematics script works by going back and forth between the forward and inverse kinematics functions with at least 3 different poses to ensure that the

results match. Consider using different quadrants for the values to really test the inverse kinematics within the entire workspace.

Recommended reporting instructions: 1 - 2 pages in the main document demonstrating that it works. As many pages as needed in the appendix with the MATLAB code.

4. **Differential Kinematics.** Compute the Jacobian matrix of the manipulator. Create a MATLAB function for the Jacobian matrix (e.g., $J = \mathrm{Jacob}(q)$). Test that your Jacobian matrix script works by giving it 3 different vectors of joint variables as input and determining by observation if the results make sense.

   Recommended reporting instructions: 1–2 pages in the main document demonstrating that it works. As many pages as needed in the appendix with the MATLAB code.

Evaluation

- Marks will be assigned on the basis of originality, correctness of solution, thoroughness and clarity of presentation.
- Present all MATLAB scripts developed as an appendix. Add comments to the scripts to explain what each command does. Include all figures necessary to assess the effectiveness of the methods implemented.
- The evaluation of this report will follow the weights in the rubric below and the marking scheme posted separately.

Rubric .	**Weight**
Forward Kinematics	30%
Inverse Kinematics	30%
Differential Kinematics	30%
Presentation (including appendices)	10%
Total	100%

Warnings

Scholastic offences are taken seriously, and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site: http://www.uwo.ca/univsec/handbook/appeals/scholastic_discipline_grad.pdf. This report will be subject to submission for textual similarity review to the commercial plagiarism detection software under license to the University for the detection of plagiarism. All papers submitted for such checking will be included as source documents in the reference database for the purpose of detecting plagiarism of papers subsequently submitted to the system. Use of the service is subject to the licensing agreement, currently between The University of Western Ontario and Turnitin.com (http://www.turnitin.com).

You must write your report in your own words. Whenever students take an idea, image, or a passage from another author, they must acknowledge their debt both by using quotation marks where appropriate and by proper referencing such as footnotes or citations. Please note that this includes software code. University policy states that cheating, including plagiarism, is a scholastic offence. The commission of a scholastic offence is attended by academic penalties, which might include expulsion from the program. If you are caught cheating, there will be no second warning.

Reports

Project reports are due on **November 11th, 2024, 11:59pm**, and should be submitted electronically through OWL (section "Assignments"). Reports are to be completed individually. No group work is allowed.