CDA 4621 / CDA 6626

Grad Total: 110 points + Extra Credit

Undergrad Total: 100 points + Extra Credit

Spring 2024

Lab 1

Kinematics

Due Date: 2-1-2024 by 11:59pm

The assignment is organized according to the following sections: (A) Objective, (B) Requirements, (C) Task Description, (D) Task Evaluation, and (E) Lab Submission.

A. Objective

This lab will teach you how to compute the robot kinematics and translate them to motion control to navigate through a set of waypoints. It will also teach you about (1) encoders and (2) IMU - Inertial Measurement Unit.

A.1 Encoders

Encoders provide feedback on wheel rotational speed in read per second.

A.2 IMUs

Inertial Measurement Units (IMUs) measure robot rotation angles around the *z*-axis (yaw). IMUs return a value in range of [0, 360] with respect to the global frame of reference (not the robot local frame). For example, Figure 1 shows coordinates with 0^0 corresponding to East.

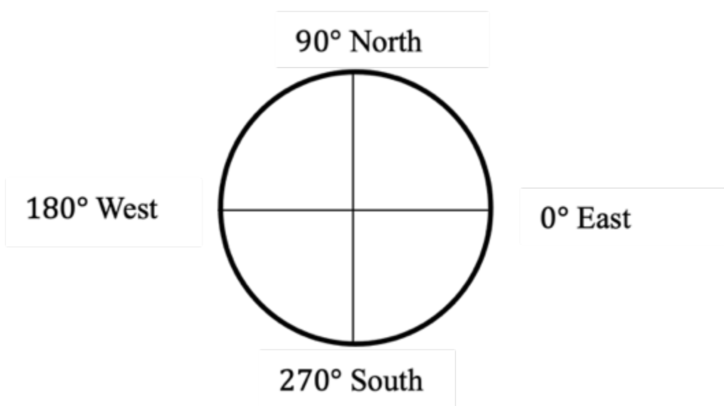

Figure 1. IMU coordinate readings.

B. Requirements

Programming: Python

B.1 Physical Robot

See Appendix B.1.1 Robobulls-2023 or Appendix B.1.2 Yahboom 2024.

B.2 Robot Simulator

See Appendix B.2.1 Webots "RosBot".
Github Server: https://github.com/biorobaw/FAIRIS-Lite (see "README.md")

C. Task Description

The lab consists of the following navigation tasks:

- Task 1: Waypoint Navigation (Required all students)
- Statistic Task (Required only Grads, Extra Credit Undergrads)
- Extra Credit Task: Ellipse (Optional all students)

C.1 Task 1 – Waypoint Navigation

Calculate and follow the precise trajectory shown by the dashed lines in Figure 2. Trajectory is composed exclusively of straight lines and curve arcs. All measurements are in meters and radians.

- Robot starts at pose $P_0 = (1.5,-1.5,\pi)$ and finishes at $P_7 = (0,0,\pi/2)$
- Robot navigates through waypoints $P=\{P_1,\ldots,P_7\}$
  $P = \{ P_1 = (-1,-1.5,\pi), P_2 = (-1.5,-1,\pi/2), P_3 = (-1.5,1,\pi/2), P_4 = (-1,1.5,0),$
  $P_5 = (0,1.5,0), P_6 = (1.5,0,3\pi/2), P_7 = (0,0,\pi/2) \}$
- Compute and follow the trajectory at maximum possible motor speeds.
- Compute and apply navigate velocities V_{li} corresponding to the left wheel velocities, and V_{ri} corresponding to the right wheel velocities. Note that you can choose V_{li} and V_{ri} as long it follows the trajectory shown by the dashed line.
- Compute D_i corresponding to the distance between P_{i-1} to P_i, and T_i corresponding to the time it takes to travel D_i.
- Before your robot navigates, <u>your code should print V_{li}, V_{ri}, D_i, T_i between pairs of waypoints</u>.

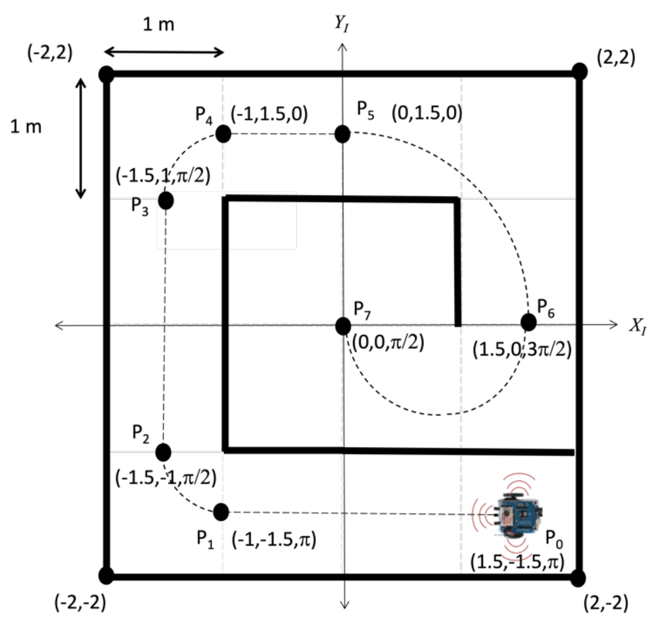

Figure 2. Path to be followed by the robot.

It is highly recommended that you write solutions using functions that generalize to any set of waypoint navigation. Specifically, any path can be broken down into three motions: (a) straight line, (b) circular motions, and (c) rotations in place. It will also be useful to create functions that return preprocessed sensor readings such as compass readings and encoder readings.

C.2 Stat Task – Statistics on Waypoint Navigation (Required for grad students)

Perform statistics on the following:

- Run the full path at 3 different speed combinations while trying to minimize overall travel time.

- Analyze and graph min, max and variance of velocities and travel times.
- Discuss your results including any navigation errors.

C.3 Extra Credit Task - Ellipse

- Specify an ellipse trajectory between P_1 and P_4 with at least 5 waypoints on the ellipse.
- Navigate the specified the ellipse trajectory between P_1 and P_4.

D. Task Evaluation

Task evaluation: (1) program code, (2) report, including a link to a video showing each different task, and (3) task presentation videos of robot navigation. Note that tasks needed in future labs.

D.1 Programs (90 points)

The following section shows the rubric for the different tasks:

- Task 1 (90 points) - Required for all students.
  - Velocities, distances, and travel times are correctly computed (35 points)
  - Robot travels the correct path with encoder control (35 points)
  - Program prints prior to navigation a summary of all velocities, distances, and times, between all waypoints (10 points)
  - Program prints during navigation all changes in velocities (10 points)
- Statistics Task (10 points) - Required for grad students. Extra credit for undergrad students.
  - Statistics are correctly computed, analyzed, and graphed (10 points)
- Extra Credit Task (20 points) – Extra credit for all students.
  - Compute correct ellipse (5 points)
  - Compute correct additional endpoints (5 points)
  - Navigate correct path (10 points)

D.2 Report (10 Points)

The report should include the following (points will be taken off if anything is missing):

- Mathematical computations for all kinematics. Show how you calculated the speeds of the left and right servos given the input parameters for each task. Also, show how you decide whether the movement is possible or not.
- Conclusions where you analyze any issues you encountered when running the tasks and how these could be improved. Conclusions need to show an insight of what was learnt (if anything) during the project. Statements such as “everything worked as expected” or “I enjoyed the project” will not count as conclusions.
- Video uploaded to Canvas showing the robot executing the different tasks. You should include in the video a description, written or voice, of each task. You can have a single or multiple videos. Note that videos will be critical in task evaluations.
- Deductions:
  - Handwritten text or images (-5 points)
  - Missing video link requires in person presentation (-20 points)
  - The submission format is not correct (see assignment details) (-10 points)
  - Failure to answer TA's email within 48 hours of the initial email (-50 points)
  - Additional deductions may apply depending on submission

<u>Videos</u>

Video needs to be clear and audible and must show the robot performing the correct path and showing the program outputs printed to the console, as outlined in the assignment. TA may ask you additional questions to gauge your understanding via Canvas Message or MS Teams. Failure to reply may result in point deduction.

- Task 1 Video
  - Record a single video of the robot performing the waypoint navigation.

- Extra Credit Video
  - Record a single video of the robot performing the ellipse waypoint navigation.

E. Lab Submission

Each student needs to submit the programs and report through Canvas under the correct assignment. Submissions should have multiple file uploads containing the following:

- The "zip" file should be named "yourname_studentidnumber_labnumber.zip" and should contain the Python file containing your controller. Controllers should be named as "Lab1_Task*X*.py", where *X* is the task number. The zip file should contain any supporting python files needed to run your code. For example, if you created functions to perform actions and these are kept in a file, this file needs to be included. At the top of each python file include a comment with the relative path from FAIRIS_Lite that this file needs to be placed in. Example #WebotsSim/libraries/my_functions.py.
- Videos should be contained in a separate zip folder with the name "yourname_studentidnumber_labnumber_videos.zip" and added as a separate file upload to Canvas.
- The report should be in PDF and should be uploaded to Canvas as a separate file.
- All zip files must be in .zip format. We will not accept RAR, 7z, tar, or other.