Control of Mobile Robotics

CDA4621

Spring 2017

Lab 3

SLAM: Simultaneous Localization And Mapping Task

Total: 100 points

Due Date: 3-27-17 by 8am

A. **Lab Requirements**

The lab requires use of the course robotic hardware ("Robobulls-2017") provided to students at no charge for the duration of the course. Required software can be downloaded free of charge from the web. All labs are to be done by teams of no more than two students. Note that no diagrams or descriptions by hand will be accepted. Each student is required to submit his or her joint report through CANVAS. Penalties will apply to any individual student submitting a late assignment even if the partner has already submitted it. Accepted documentation file types are PDF, Word (doc, docx), and Powerpoint (ppt, pptx). You need to upload all your code together with the document as part of a "zip" or "rar" file. Any corrupt files uploaded to Canvas will not be considered and will not be graded. This includes incorrect or private links to videos. Other than external links to videos, no external links to other documents or images will be considered, such as links to google drives, etc.

- Hardware Requirements

The "Robobull-2017" (Figure 1) is the main robot hardware used for the course.

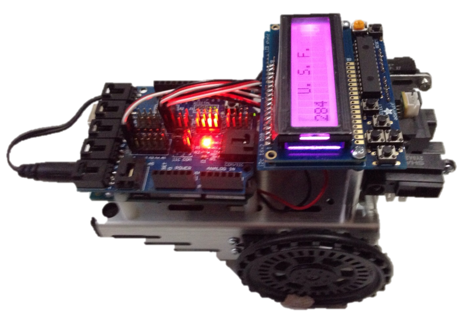

Figure 1: Robobulls-2017

- Software Requirements

Arduino Software (Version 1.8.1 or later)

https://www.arduino.cc/en/Main/Software

B. **Task Evaluation**

Each individual task is worth a specific number of points where these points are always split 50% between Task Execution and Task Report:

- Task Execution

The robot should execute the task correctly with a video clearly and completely showing the task execution (points will be taken for errors or missing aspects of task execution).

- Task Report

Each task report requires an accompanying document to be uploaded to Canvas together with ALL the files required to run the program in the robot. The task report needs to include ALL the following sections (points will be taken off if anything is missing):

1. Task description.
2. Flowchart describing the detailed solution used to solve the task. (Example of flowchart: https://en.wikipedia.org/wiki/Flowchart.
3. Code sections with detailed explanations showing how they relate to each block in the flowchart used to solve the task (this is in addition to "ino" code submitted as separate files).
4. Images taken from the actual robot task execution (at least one image per task).
5. Conclusions where you analyze any issues you encountered when running the task and how these could be improved.
6. Video link to different task executions (you should split each task execution as a different video link or as separate sections in a single video most preferably in YouTube). Provide at the beginning of each video your name and description of the task being performed.

C. **Task Description**

The goal of this assignment is to develop an algorithm to navigate the complete maze. The robot has to build a map of the maze and localize within the map. The exact same algorithm has to work for the three mazes shown in Figure 2. There will be tape on the carpet marking the border of each grid cell and you may use the encoders to count wheel rotations and corresponding wheel velocities.

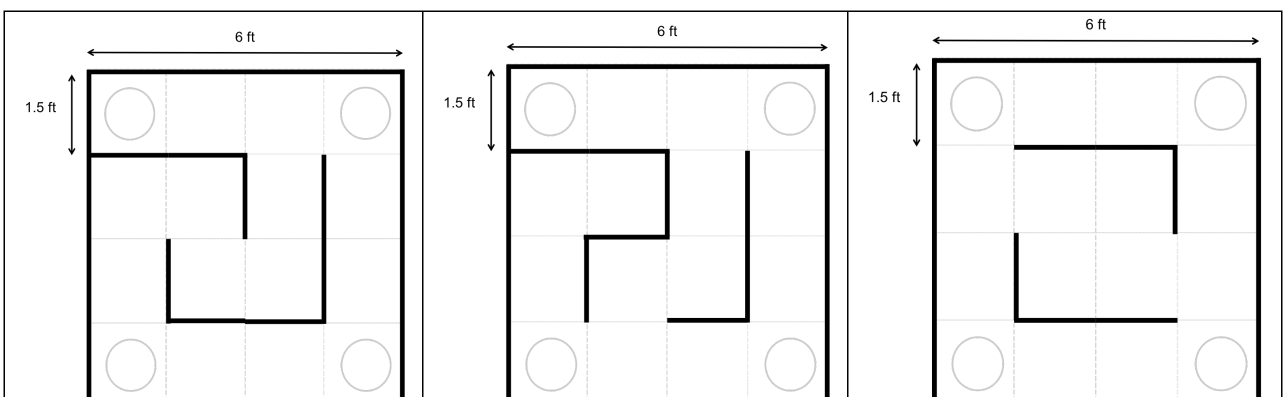

Figure 2. Three different maze configurations (left, center, right). Each maze is 6x6 sq. ft. consisting of a 4x4 marked grid cells of 1.5x1.5 sq. ft.

1. Maze Navigation Task (40 points)

The objective of Task 1 is to develop a program that allows for the robot to navigate the full maze. The robot is considered to have navigated the full maze when it has passed through each of the 16 grid cells at least once. The robot may start to navigate from any of the 4 corner cells pointing in any desired direction and may do so by wall following. The task requires the robot to navigate in a single uninterrupted sequence all grids in each maze at least once without human intervention. The robot may pass through any particular grid multiple times. Note that for this task you may manually stop the robot when finished.

2. Localization and Mapping Task (60 points)

The objective of Task 2 is to provide mapping and localization as the robot navigates the full maze. As an example, if the robot follows the "orange" path shown in Figure 3 (a), then the corresponding grid cell localization will be represented according to Figure 3 (b) based on the grid cell numbering from 1 to 16. Finally, Figure 3 (c) provides a global grid cell orientation.

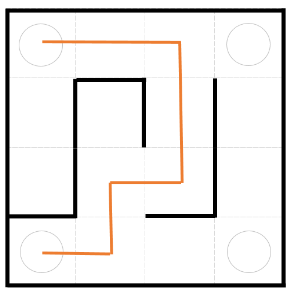

Figure 3. (a) Grid Cell Robot Path

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Figure 3. (b) Grid Cell Robot Path

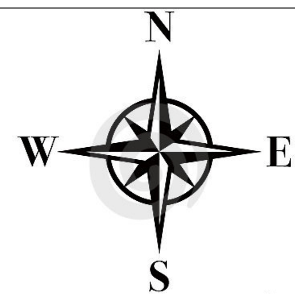

Figure 3. (c) Global Grid Cell Orientation

The task requires the robot to navigate in a single uninterrupted sequence all grids in each maze In Figure 2 at least once without human intervention, where similar to Task 1, the robot may pass through any particular grid multiple times. The main difference with Task 1 is that the robot needs to know exactly which grid it is traversing and provide information on where the walls for each grid are located.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	X	X	X	0	0	0	X	0	0	X	X	0	X	X	0	0
2	G	1		W	U		N	X		S	0		E	0		

Figure 4. Robot Path and Wall Information as shown in LCD

Figure 4 shows an example of such information shown in 2 rows and 16 columns in the LCD that must be updated as the robot traverses the maze. The first row in the LCD displays marks with an "X" already visited cells, and marks with a "0" a cell not yet visited with numbering as shown on top of the columns. The second row displays information on wall sensory information at each step. "G" and "1" enumerates the current grid cell where the robot is located, e.g. Grid Cell #1. The rest of the cells in the second row describe compass points based on Figure 3 (c), e.g. "W" –

West, “N” – North, “S” – South, “E” – East, and whether there are walls sensed by the robot distance sensors at that grid location, represented by “U” – Unknown, “X” – Wall, and “0” – No Wall. Note that some wall information may not be available, as the robot does not have a distance sensor in its back. LCD information should be updated every time the robot moves to a new grid cell. Additionally the LCD display must provide a flashing backlight color each time it changes grid according to the following: (a) Red when moving to a grid that is “E”, (b) Green when moving to a grid that is “W”, (c) Blue when moving to a grid that is “N”, and (d) Yellow when moving to a grid that is “S”. Single flashes should take place per individual grids. The rest of the time the LCD should show a white background.