Final Project:

Localization and Map Making

Lectures: Week 1 - 8 Lectures

Reading: *Mobile Robotics for Multidisciplinary Study* Textbook

(Demonstrations due in class during **Week 9 and Week 10**)

(Project Report and Code due in Moodle drop box by midnight on **Sunday of Finals Week**)

****

Purpose: The purpose of the final project is to demonstrate the integration of some of the concepts learned this quarter by creating localization and mapping algorithms for the Arduino Robot. The localization task involves using sensor feedback with a navigation routine to determine the location of a lost or kidnapped robot in the world and then rescue it by moving it to its home location. The mapping task involves the creation of a hybrid topological and metric map of the robot's world by using sensor feedback and an exploration algorithm. The SLAM technique involves integrating mapping and localization to correct for odometry error and use path planning to move the robot to a location in the world.

Objectives: At the conclusion of this project, the student should be able to:

- Use sensor data and an exploration algorithm to localize a robot in a world given an a priori map
- Use sensor data and a coverage algorithm to create an occupancy grid or topological map of the robot's world
- Create a world model that the robot can use for navigation
- Use a SLAM algorithm to correct for sensor and odometry error to create a map and use it to plan a path for the robot to move to a location in the world

Equipment: Arduino Robot

Range sensors (3 or 4)

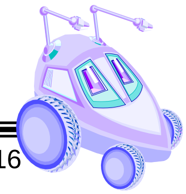
Software: Arduino IDE

PROCEDURE

****

Part I - Mapping

1. In this exercise, you will build a map of the 6' x 6' test arena by using a cover algorithm and sensor feedback. The robot should be able to use the created map to plan a path from a start position to a goal location. Try to reuse as much code as possible from prior labs once the map has been created to simplify the path planning and execution.
2. Note that mapping will be tricky because of odometry error, sensor error, and other sources of error. A completely accurate map would require localization to reduce the uncertainty of position. However, this would require SLAM which is difficult to implement. Therefore, using only the robot's sensor data and odometry try to create the best world model possible.
3. You can use wander, cover, wall following or any other motion algorithms to move the robot through the environment. One possible technique is to create a generalized Voronoi diagram or follow center behavior to efficiently explore the environment. Use the robot's pushbuttons to start the motion algorithm.
4. Then, use Histogrammic in motion mapping (HIMM) to identify objects in the environment. Although your textbook states that this algorithm was created for a sonar sensor, it is reasonable to use it for the primary axis of the IR sensor. For this method when the cell value exceeds some threshold it is coded as occupied or used to create a topological map. You do not have to use this technique you have the flexibility of selecting your own technique to create the model. Devise a mechanism to display the map created on the LCD after the motion completes.
5. Alternately, you can create an occupancy grid with 0's and 1's based upon the free and occupied space. Figure 1 provides an example of using a Generalized Voronoi Graph (GVG) and HIMM to create a world map.



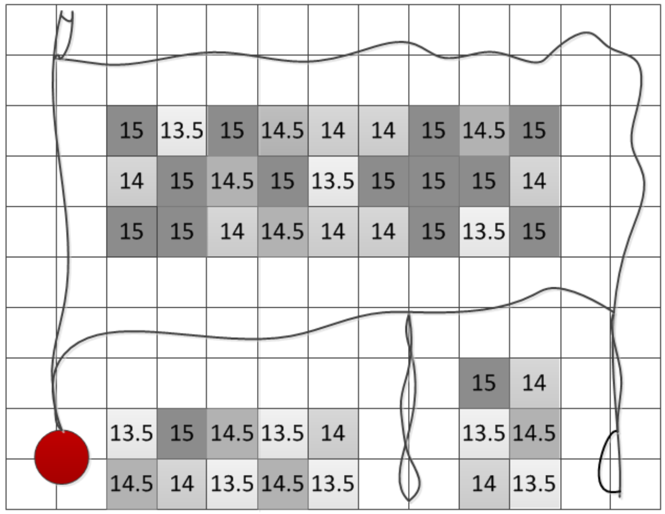


Figure 1: GVG with HIMM

6. You should show the robot's state on the LCD as it creates the world map. In addition, upon completion of the motion algorithm, the robot should display the world map on the LCD.

7. Once the robot has created the map, it should be able to use this map to create a path from a start position to a goal location. Similar to Lab 7, you will be given a start position and goal location and the robot then plans a path to the goal.

8. You will be graded on the ability to generate the map, the accuracy and detail in the map and how well the robot is able to use your world map to plan a path to a goal location while avoiding obstacles.

<u>Part II - Localization</u>

1. The user will provide the robot with an a priori map of the world as a text file. The map will be a 4 x 4 array with topological map or occupancy grid encoding.
2. The localization algorithm should process this map to identify key features such as the gateways or distinctive places [corners (C), hallways (H), dead-ends (D), t-junctions (T)]. Although it is a not a gateway, a series of 10's or 5's or neighboring 1's and 4's or 8's and 2's indicate a hallway (H) in the world. The numbers (1, 2, 3) indicate a gateway where the robot can make a navigation decision. The distinctive features or gateways are the nodes in the world and the hallways can be used with a local control strategy such as follow center or follow wall to move between nodes. An example of coding a map is shown in Figure 2.

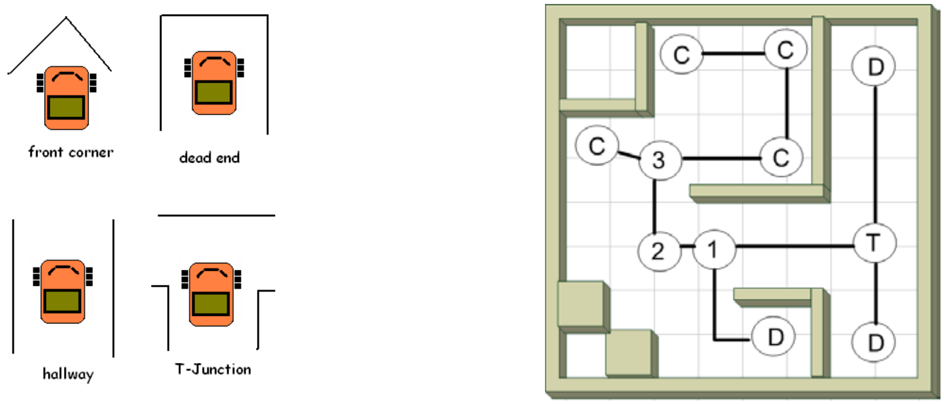


Figure 2: Feature Extraction (Map Coding)

3. Once the robot is placed in the world, the robot should then use some motion algorithm such as random wander, follow center, or wall following to explore the world and identify gateways. It should keep track of the gateways passed and the order in which they were passed. Although there will be some odometry error, it would also aid the localization algorithm to keep track of odometry such as distance traveled and turns. Within three to four iterations of this process, the robot should be able to use a probabilistic method such as the Partially Observable Markov Decision Process (PMDP) to localize itself. Figure 3 provides an example of this localization process with the proposed robot locations after each step marked on the map.

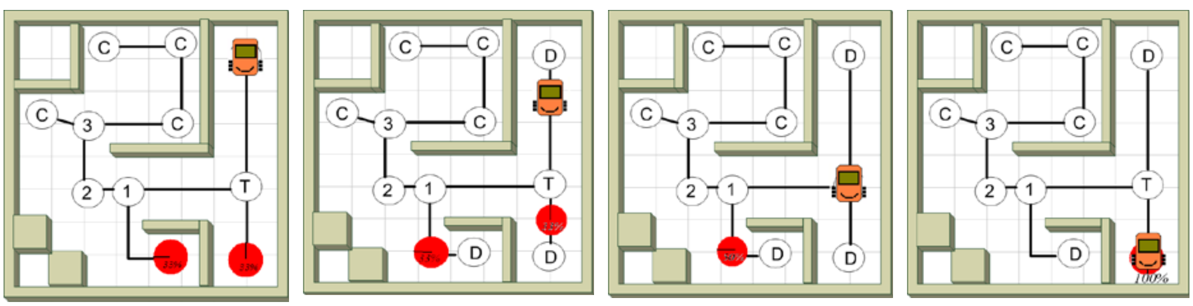
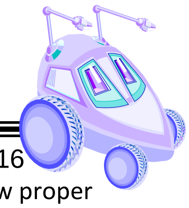


Figure 3: PMDP Localization

4. Lastly, after determining its location using registration, the robot should use wavefront propagation to plan a path from its current location to the goal location (or home). The robot's home position and goal location will be specified at run time. After localization, you can use your lab 7 code to plan a path for the robot.
5. During the demonstration, the robot will be placed in the world at an arbitrary location and move between nodes until it is able to determine its location in the world. The robot should then plan a path and move home.

<u>Part III – Simultaneous Localization and Mapping (SLAM)</u>

There is no explicit procedure for this portion of the final project. Since this is the culmination of the mobile robotics course, the team is expected to use what they have learned this quarter to complete this portion of the final project. The team should use the textbook, readings, lectures, and perform research in order to determine how to implement SLAM on the Arduino Robot. There are many online resources on this topic and a great resource



to use is the library and Google Scholar. In the final report, the team must list all sources used and follow proper APA format for references and citations. Please review Purdue OWL at the following link for help and the APA style guide (https://owl.english.purdue.edu/owl/resource/560/01/).

During the demonstration, the team must clearly explain how SLAM was used to correct for odometry and sensor error in map making. In addition, the team must completely demonstrate the working algorithm on the robot in a 6’ x 6’ world with 18” x 18” cells. This should include using the LCD or buzzer to indicate state as well as how the robot can build a map, localize in the map, and plan a path through the map. Finally, the team must upload properly commented code to explain the algorithm.

Evaluation

Due to the complexity of this project and the fact that it requires an integration of several concepts, there will be **three** demonstrations required and a graduated grading scale. Table 1 shows the point distribution for the final project.

Table 1: Project Grade Point Distribution

Task	Percent	Due by
Demo 1 – Mapping and Path Planning	18	Thursday of Week 9
Demo 2 –Localization and Path Planning	18	Monday of Week 10
Demo 3 - SLAM	14	Thursday of Week 10
Code	25	Sunday of Finals Week
Report	25	Sunday of Finals Week

Submission Requirements

You must demonstrate mapping and path planning by **Thursday of Week 9**.

You must demonstrate localization and path planning by **Monday of Week 10**.

You must demonstrate an integration of mapping, localization and path planning by **Thursday of Week 10**.

You must submit your properly commented code and final project report by midnight on **Sunday of Finals week**. Recall that properly commented code has a header with the solution name, team members’ names, description of the functionality and key functions, revision dates. In addition, all of the key variables and functions in the code are commented. Please use the following checklist to insure that your final project report meets the minimum guidelines.

Project Report Guidelines

1. The document should have default Word settings with respect to font and margins
2. All pages should be numbered
3. All headings must be numbered, left-justified, bolded, and capitalized at the beginning of the section.
4. All figures must have a number and caption underneath (i.e. GUI screenshots)
5. All tables must have a number and title above it (i.e. results error analysis)
6. The cover page should have title, partner names, course number and title, date

7. The report should order should be:

    Cover page
    Abstract
    Table of Contents
    I. Objective
    II. Theory
    III. Methods
    IV. Results
    V. Conclusions and Recommendations
    Appendix/Supplementary Materials
    References (if any)

8. The ***abstract*** should be a brief statement of the experiment purpose, verification and relevant results
9. The ***objective*** should state the purpose of the project and associated tasks in your own words
10. The ***theory*** should state relevant theory that will be used to achieve the purpose
11. The ***methods*** section should summarize the algorithms implemented to achieve the purpose and the procedures used to test the robot algorithms
12. The ***results*** section should summarize the results of the tests and verify that the robot was able to achieve the purpose and meet the project objectives
13. The ***conclusions and recommendations*** should address whether the purpose was achieved, possible sources of error, recommendations to improve the robot algorithm and answer any relevant questions related to the project.
14. Remember this is only a guide for the minimum requirements of your report. You are required to answer any questions or provide any details that you feel aid the reader in understanding the objective, theory, procedure, implementation and results of your project.