|  |  |
| --- | --- |
|  | Rose-Hulman  Institute of Technology |

Memo

To: Dr. Carlotta Berry

From: Ander A Solorzano \_\_\_\_\_\_\_\_\_\_\_\_\_\_ and Ruffin White \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: ECE425 – Mobile Robotics

Date: 2/1/2013

Title: Lab07 – Map Making

**PURPOSE**

The purpose of this lab is to use the range sensors on the CEENBot platform to create an occupancy grid or a representation of the world. After the robot has generated the map of the world, the robot should then use the path planning algorithm developed in the previous lab to navigate the world from a start point to an end point. The robot is expected to use feedback to prevent wall collisions and stay on track.

**PROCEDURES AND STRATEGY**

Using our previous map structure, a binary method for indexing cells, and a bitwise representation of walls for individual cells, our map algorithm sequentially iterates cell by cell while reading its current sensor value and rotates its sensor reading appropriately. The navigation behavior is a simple maze algorithm where our robot seeks a prioritize movements.

Using the sensor reading, the robot makes a decision on whether the robot is able to make a left turn. If the robot is unable to make a left turn, it then proceeds to check if it can move forward. If the robot cannot turn left or right, the robot has reached a dead end and it must turn right by 90o. Using another FSM, we checked to see if our previous movement was a left spin in order to enforce a forward motion to complete the left turn. This completes the entire left turn action. We iterate the sequence of logic mentioned above for every cell until the robot detects that it has reached the starting cell and the starting orientation. Only then will the robot recognize that it has mapped the entire world.

In addition to mapping the world, we added a feature that displays the mapping of the world. Using the 32-bit LCD screen, we represent the occupancy cell and its four-wall structure within an 8x8 pixel representation. The boundary edges representing the existence of each wall and the center cell remaining empty until the 4x4 representation of the robot is drawn on the center of the cell. Along with the robot continuously representing its location in the map display, the robot also represent its orientation by a 5th pixel that is used a pointer in the 4x4 representation of the robot.

After the robot has completely mapped the world, it stores the world in a world map structure and proceeds to prompt the user for additional commands. The user them inputs all the necessary topological path planning commands and the robot executes the topological path planning algorithm to navigate from a start cell to the end cell. The rest of the algorithm is the as described by the previous memo.

|  |
| --- |
| **C:\Users\solorzaa\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Outlook\28316QXF\20130201_203852.jpg** |
| Figure 1: Map representation after the robot explored the world and mapped the cells. The dot on the upper left cell represents the robots actual position. |

**CONCLUSION**

The purpose of this lab was to map an unknown robot world and used the mapped wolrd for path planning. The greatest source of error observed was the odometry errors as the robot explored the unknown. It is important to note that the robot was not coded to explore using dead-reckoning but instead it is able to adapt itself to the environment. Due to our wall following and FSM exploration algorithm, the main feature of a world that can present a problem to the robot is an isolated island in the middle of an open space.

C:\Users\solorzaa\Desktop\Image.tif