Coverage Efficiency in Autonomous Robots With Emphasis on Simultaneous Localization and Mapping Algorithms

Mo Lu P.4

Abstract

Today, automated systems have supplemented humans in previously labor-intensive tasks. Automated lawnmowers are an example of these systems, but the currently available technology in automated lawnmowing is inefficient and primitive. This project will propose and implement an alternate method to automated lawnmowing, known as Simultaneous Localization and Mapping.

Background

Current approaches to commercial robotic lawnmowing operate under the idea that if a lawmower is constantly mowing the lawn, then the lawn stays constantly mowed. This is done by a series of random cuts and turns, which if given enough time, theoretically could cover an entire unmowed lawn. This method is horrifically inefficient in terms of time and energy, when backtracking is taken into consideration. This project proposes a different approach to this method: use of mapping techniques to recognize landmarks, avoid obstacles, and navigate an environment This method consists of three parts: 1) Use of a constantly updating laser scanner to recognize obstacles, 2) Creation of obstacle map using the laser data, and 3) Processing that obstacle map for runtime efficiency. Success is determined by how effectively the robot avoids the obstacles, how quickly it runs through the lawn, and how accurate the created obstacle map is.

Procedures, Methods, and Results

Before the robot can run in a live environment, it must be first tested on a simulation platform. The simulation I have programmed includes environment generation, obstacle generation, and robot movement. Obstacles include horizontal, vertical, diagonal, and circular blocks. As the robot runs through the environment, it looks through the environment with a 180 degree scan, and updates its obstacle map with appropriate values. Results look promising, the robot successfully maps most environments, with the exception of ones with circles. Diagonal mapping also turns up width problems, due to angle scanning methods.

```
000000000000000000
001000000000000000
001000000000111100
001000000000000000
001000000000000000
001000000000900000
000000111100000000
000100000000000000
000010000000000000
000001000000000000
000000100000000000
000000000000000000
000000000000000000
000000000000000000
```

Input 1

1=Obstacles
Horizontal, Vertical, and Diagonal Obstacles
9=Robot

```
111111111111111111
191000000000000001
101000000000111101
101000000000000001
101000000000000001
101000000000000001
100000111100000001
10●1●0000000000001
100●1●000000000001
1000●1●00000000001
10000●1●0000000001
100000000000000001
100000000000000001
111111111111111111
```

Output 1

Notice the border broundaies and the deadzone around the diagonal obstacle.