

CS26020 Assignment
Developing a Maze Exploring Robot Controller
Project Report

Rhys Evans (rhe24@aber.ac.uk)

25th April 2018

Contents

1	Introduction	3
2	Design Analysis	3
2.1	Modelling The Maze	3
2.2	Modelling The Robot	3
2.3	Navigating The Maze	3
2.4	Drawing The Maze	4
3	Implementation	4
3.1	The Maze Model	4
4	Testing	4
5	Evaluation	4

1 Introduction

The objective of this project was to create a robot controller capable of exploring a perfect maze. This was accomplished by combining reactive and deliberative robotic behaviours, thus creating a 'hybrid' system. In this report I will discuss and evaluate the process of designing, implementing and testing the controller, along with the challenges I faced. In addition, I will discuss any changes I would make to the system if done again.

2 Design Analysis

In this section I will analyse various aspects of the controller's design and discuss some of the decisions I made when designing the system.

2.1 Modelling The Maze

Arguably the most important aspect of this controller is the maze model, without a reliable maze model there is no basis for the deliberative behaviour of the controller. I decided to use a fairly simple 2D Array of 'Cell' structs to represent the maze. As there was no need for the robot to start in an arbitrary position I felt this was a sufficient method and would make indexing the maze (a common occurrence) very easy.

2.2 Modelling The Robot

Having already decided on a method to store the maze I also needed to put some thought into representing the robot within the maze in a way that supported the maze model. I came to the conclusion that in order to fully encapsulate the robot I needed to store the following information:

- The robot's position within the maze (X and Y)
- The robot's orientation (North, West, South East)
- The number of cells the robot has visited

Most of these seemed pretty straight forward to implement, however in reflection I feel that I could have put more thought into the robot's orientation and how it would be decided.

2.3 Navigating The Maze

Now that I had a model in mind for both the robot and the maze it was easier to think about how the robot would use its hybrid behaviour to navigate the maze.

Firstly I decided on the method that the robot would use to fully explore the maze, I settled on the 'left hand rule'. This would mean that at each given cell the robot's preference of direction would be as follows:

1. Turn Left
2. Go Forward
3. Turn Right
4. Turn Around

If the first option is unavailable, attempt the second and so on.

2.4 Drawing The Maze

3 Implementation

3.1 The Maze Model

Within the 'Cell' structures of the maze array are two attributes 'visited' and 'walls'. The 'visited' attribute is simply a boolean to store whether or not the cell has already been explored. The 'walls' attribute is a boolean array of 4 elements to represent the four walls of the maze (0 if no wall is present, 1 if wall is present). Lastly there is a single global variable of type 'Cell' to store the 'next cell'.

4 Testing

5 Evaluation