Toby William Towler

Registration number 100395626

2025

Robot Mower Mapping and Pathing

Supervised by Edwin Ren

University of East Anglia
Faculty of Science
School of Computing Sciences

Abstract

The abstract of your report summarises your entire work () in no more than half a page. It should include the context of your work including its main objective, what methods you employed, how you implemented these, what the outcomes were and a final statement as a conclusion. It should not contain acronyms, abbreviations, elements of a literature review (though a statement of related work is permissible if it is crucial to your work) or future work. The abstract should be written when everything else has been written up and the project is finished! is this workng

Acknowledgements

Edwin Ren, Eden Attlebourgh

Contents

1. Introduction **5**

2. Background **5**

3. Methodology **5**

3.1. Map Generation 5

3.2. Complete Coverage Path Planning 7

3.3. Aerial Map Generation 7

4. Implementation and Evaluation **7**

5. Conclusion and Future Work **7**

References **9**

A. Producing tables, figures, etc. **10**

List of Figures

1. Algorithm for Generating Shapes with k Corners 6

1. Introduction

The robot mower is an already existing project developed by previous masters students from the University of East Anglia. However, it still has areas that can be improved. One of these areas is the way the Robot covers an area in the real world in order to complete its task. More specifically, the map generation and path planning. These aspects have been the key focuses of my project, I have broken these down into 3 key goals:

1. Basic map generation
2. complete coverage path planning
3. map generation from an aerial image

2. Background

Another section that is essential and should keep its title as is! Although you could perhaps call it "Literature Review" instead, this is not advisable as at this stage of your project we do not expect an extensive literature review since this was already done in the second formative assignment. The rationale is simply because you will lose valuable pages that could be used better in the next two sections that will cover the preparation and implementation of actual work done. So just provide the context in which your project operates here, and then provide a brief overview of similar work that is directly relevant to yours. Try to avoid blatant copying and pasting from the formative literature review as it is bound to read awkwardly.

3. Methodology

3.1. Map Generation

For the basic map generation, a simple shape generation algorithm which takes k points should suffice. As the outputs of this section will mostly be used for testing the path planning algorithm on regions with differing area, number of corners and complexity, no excessive algorithmic complexity is needed. This program should also be able to create n obstacles within the main field, such areas would represent obstructions in the mowers desired path, for example trees or telephone poles in the real world. This means we need a function with 2 parameters:

- K, number of angles in the outer field
- N, number of obstacles within the field, since it would not be sensible to take a parameter for the number of corners for every hole, we can generate them randomly assuming 3-8 corners staying inline with the complexity of the rest of the field without being unreasonably over engineered.

This leads us to an algorithm similar to the one below:

Figure 1: Algorithm for Generating Shapes with k Corners

```
GenerateShape(k, n):
    points = []
    angleIncrement = 2/k
    radius = 1
    center = (0, 0)

    for i = 0 to k-1:
        angle = i * angleIncrement
        x = center.x + radius * cos(angle)
        y = center.y + radius * sin(angle)
        points.append((x, y))

    shape = CreatePolygon(points)

    if n > 1:
        GenerateShape(k, n-1)

    return shape
```

3.2. Complete Coverage Path Planning

Complete coverage path planning(CCPP) is "the task of determining a path that passes over all points of an area or volume of interest while avoiding obstacles" Zhao and Hwang (2023)

3.3. Aerial Map Generation

4. Implementation and Evaluation

Could be a section each for implementation and evaluation if this suits you better or you could use subsections instead. The difference between this section and the previous "Methodology" section is that this one covers "action" or in other words your active contributions to the project. These may include:

- Implementation of programming code: Describe your final code architecture using for example (UML) diagrams and code snippets. Make sure that code snippet (figure) captions are self-explanatory which means that you should not have to consult the text body to understand what is shown in the figure. Many code snippets of the same kind should end up in an appendix instead.
- Results from experiments run, including testing (user and software). Use figures and tables with self-explanatory captions (see earlier statement). Multiple figures and tables that cover several pages should be put in an appendix.
- Analysis of results: Discuss your experimental and/or test findings in depth. Compare them against other studies and/or benchmarks, and point out limitations of your work (that could be due to limited time) and elaborate on scope for improvement.

5. Conclusion and Future Work

Another essential section that should keep its title as suggested. Briefly discuss your main findings, outcomes, results; what worked and what could have been done differently. Then summarise your work in a concluding statement by comparing your outcomes against the main and sub-objectives and/or MoSCoW requirements (if used) and suggest potential future work that could be done if more time would be available.

References

Chardaire, P. (2013). Tutorial 5: Producing professional looking tables.

Zhao, S. and Hwang, S.-H. (2023). Complete coverage path planning scheme for autonomous navigation ros-based robots. *ICT Express*, 9(3):361–366.

A. Producing tables, figures, etc.

Please refer to the original template for different ways of formatting tables Chardaire (2013), figures, code snippets and pseudo code for algorithms. Make sure the caption of each of these is self-explanatory which means that you should be able to understand the figure, what's in the table, what the code is about, etc. without having to consult the text where they are referenced from.