

Agri Robot Mapping and Pathing Progress Report

Toby Towler

registration: 100395626

1. Introduction

This "Agri Robot" was created by several masters students, I am improving the mapping and pathing side of this project. The robot will be able to take in a map and path. Both will be made up of lists of 2D points. Using other modules, the robot will be able to accurately

The final aim of this project is to have a finished, publishable and sellable product that could be used in the real world.

I have a few smaller goals to complete my part:

1. Random map generation
2. Pathing
3. Map generation from an aerial image

My motivation for this project is to make it run as fast and efficiently as possible, allowing to be run on the lowest specification hardware possible. In a real world business context this would mean the products cost would be lowered.

2. Progress

2.1. Mapping

It is important to be able to test the robot's coverage of an area with many shapes of field to ensure it always works on any given area. So there must be a way to generate maps to test all steps of the process on Firstly, I had to build an algorithm to generate maps to represent fields. The maps are made up of the outline of the main field, and a series of "holes" these would be trees or telegraph poles in real life, areas in the field that must be avoided by the robot.

My solution was the following:

1. Generate n random (x, y) points
2. Select origin point with lowest y value
3. Sort remaining points by polar angle to the origin point

This always produces a full polygon that never overlaps and can have any number of vertices where $n > 2$. For simplicity, I used a fixed origin of (0, 0) and generate points with positive x and y coordinates. Although the algorithm could be adapted to work with all random points. The polar angle method comes from the Graham Scan for making a convex hull. My algorithm is based on the first part of this idea. This algorithm is repeated for the outline and the holes in the field, due to the varying number of vertices and random point generation, all polygons in the field can be different.

2.2. Fields2Cover Mapping

Fields2Cover (F2C) is a Complete Coverage Path Planning (CCPP) open source library. I used this library primarily for CCPP, but it also has methods to generate fields of its own, Which I used this for testing the F2C pathing solutions. This method works by taking parameters for the area of the field and the number of vertices required. There are no essential differences to using the F2C maps or my own algorithm as they both return a collection of 2-dimensional points meaning that the CCPP solution should be able to work with either map as input.

2.3. Complete Coverage Path Planning

Complete coverage path planning (CCPP) is the process of making a path for an object of width x to traverse an area while covering all points of the given area. This is important for the agri robot as it needs to mow an entire area of grass leaving no area uncut to be a successful product. At the beginning of trying to use the library, I had several compilation errors when trying to build from source. I was able to fix the issues myself and then submit a GitHub pull request that was later approved, making now a contributor to this open source project.

After fixing these issues, I was able to use the library to generate CCPP solutions to inputs of any map. This library also allows easy configuration of the sizing, turning circle and velocities of the robot input. Altering any of these will alter the path to give another usable solution to the problem

A. Appendix: Additional Figures

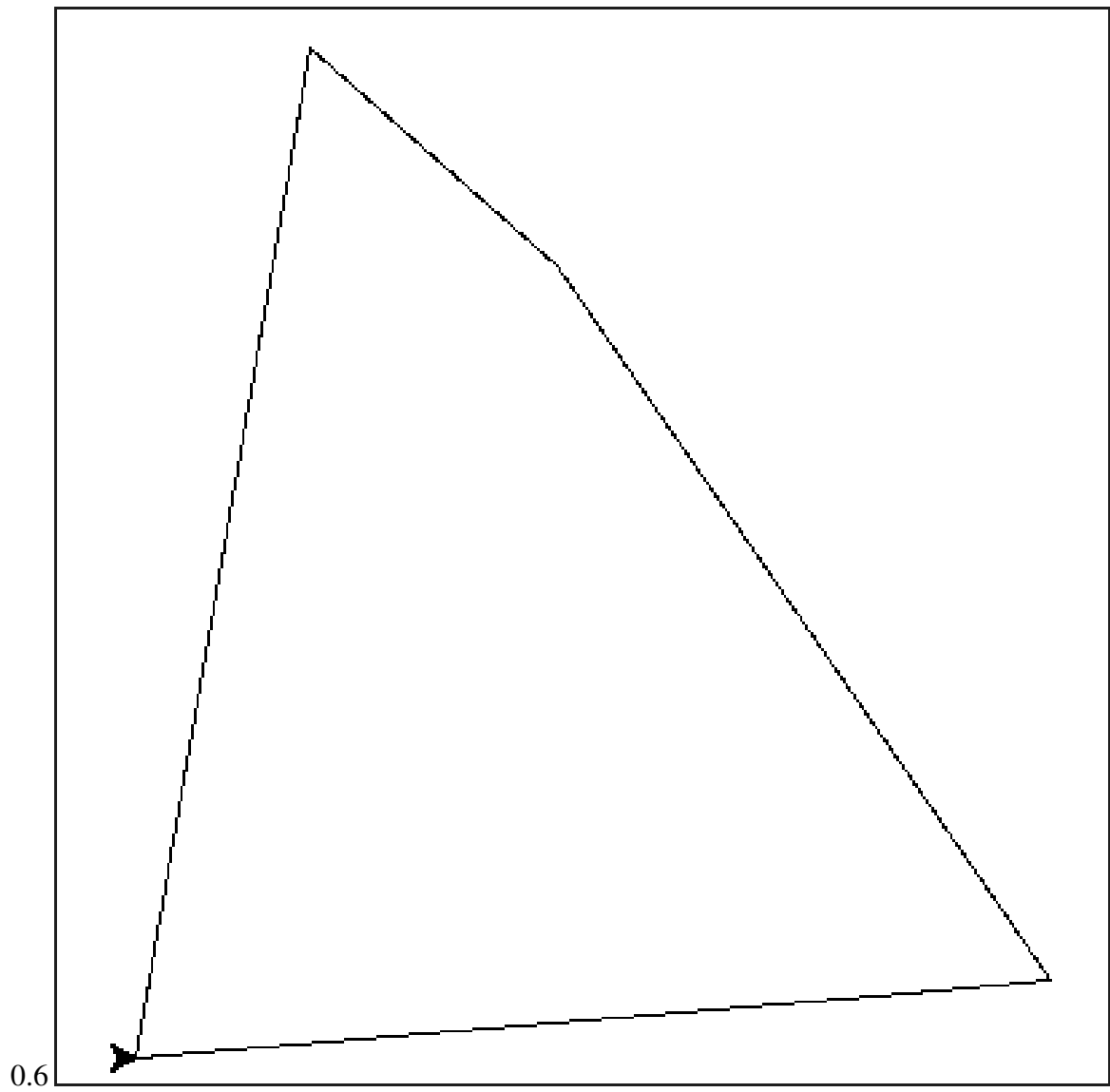


Figure 1: Random map generation with a small number of vertices

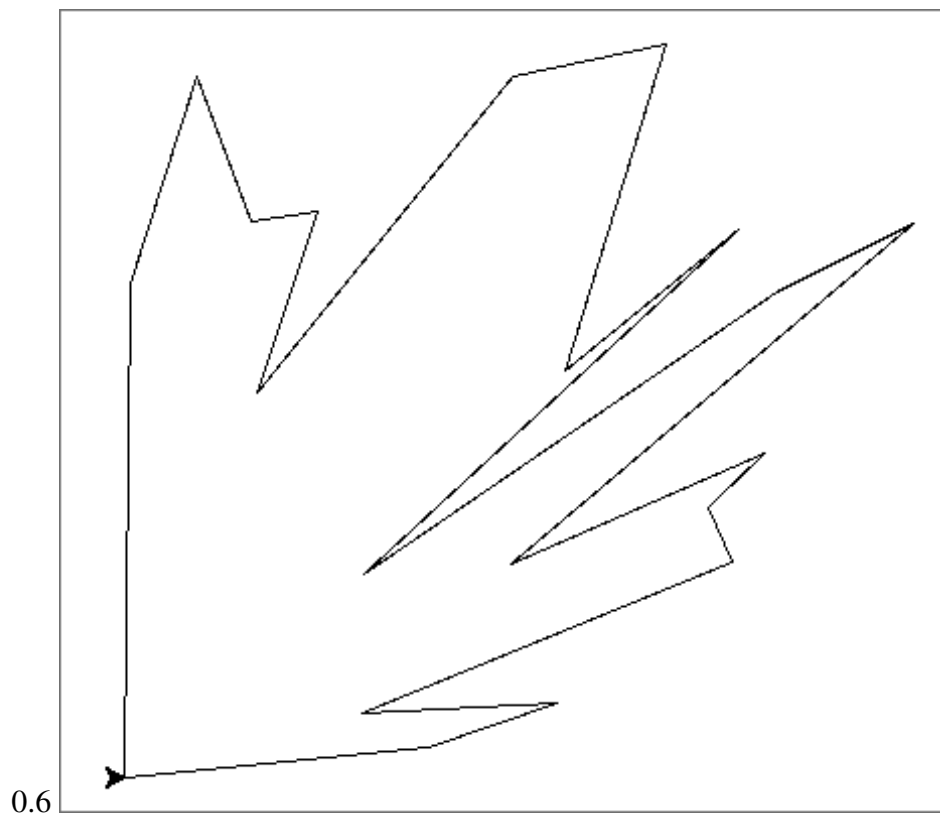


Figure 2: Random map generation with many vertices

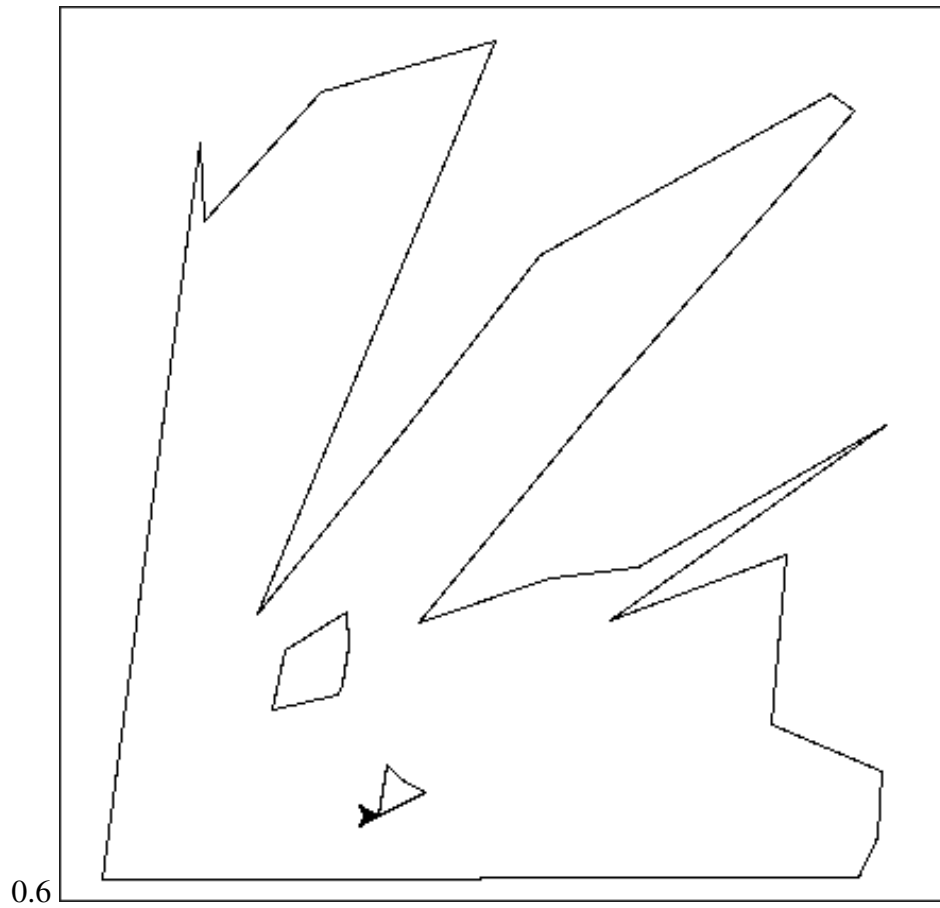


Figure 3: Random map generation with many vertices