Contents

1 Introduction

In robotics, SLAM is defined as simultaneous localisation and mapping, in this project different SLAM methods will be explored, used together and evaluated in an outdoor environment, this is in order to achieve robust localisation and mapping for the Husqvarna Automower 430X[2].

Figure 1: Husqvarna Automower 430X[2]

SLAM can be summed as the following: “robots will have to be able to generate a map of the environment using sensor information. Mapping an unknown space with a robot that experiences positioning error is an especially challenging “chicken and egg” problem – without a map the robot cannot determine its own position, and without knowledge about its own position the robot cannot compute the map. This problem is often called simultaneous localization and mapping or simply SLAM.”[6]. In the initial phase of this project, the following methods/sensors will be used for SLAM: Odometry, GPS Receiver, Automower boundary sensors (magnetic field sensors). And in the second phase: Visual based SLAM method (using an on-board camera). Each of the methods will be evaluated and compared against each other. Allowing for the most optimal solution to be found.

Resultant maps generated will be displayed inside a mobile application for viewing. The maps will be able to be interacted with. Highlighting areas of the map in which to restrict or forbid the Automower. The quality of the localisation and mapping will directly contribute to the effectiveness of this feature. The maps will also be used to increase the complexity of the Automower’s navigation algorithm. This is possible due to the extra information the Automower will have about its environment (from the maps and the localisation), allowing it to make more educated decisions for navigation. The default Automower implementation moves forward in a straight line until a boundary loop is met, then chooses a new random direction (away from the boundary loop) and repeats.

An additional stretch goal of the project, if the best SLAM method is sufficiently accurate, is to be able to upload an image to the mobile device to be placed inside the generated map, brighter areas of the image would be mowed more and darker areas mowed less. Leaving an imprint of the image in the grass, viewable from above.

1.1 Background Information

In a normal setting, the Husqvarna Automower operates inside a lawn that is restrained with a boundary loop, the Automower stays within the loop using its boundary sensors to detect when it has crossed the wire (the polarity of the magnetic field switches). Additionally the boundary loop will extend into the garden to cut off “islands” in the lawn, the connecting wire between the “islands” and the boundary is able to be ignored by the Automower. Finally in the lawn is the guidewire, this wire is used specifically so the Automower can find the charging station when low on battery. Below is an example lawn from the Husqvarna Automower manual.

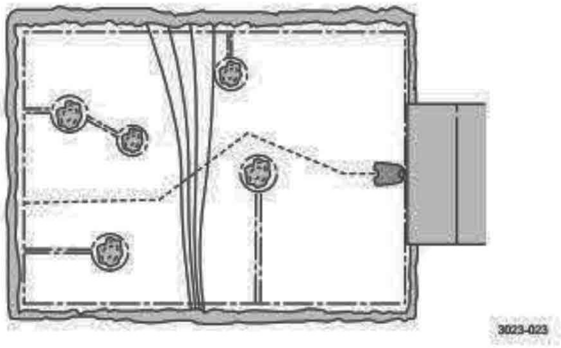

Figure 2: An example lawn, showcasing the boundary wire and the islands in the centre, along with the finer dotted line, which represents the guide wire [7].

By default the Husqvarna Automower comes equipped with wheel encoders on its wheels and boundary sensors inside of it, however the current model does not have GPS, so a receiver will be fitted to it. To make use of all the sensors, a Raspberry Pi was installed on the Automower (previous project [12]), the Raspberry Pi receives information from the GPS receiver and from the Automower’s sensors via USB. Using ROS Kinetic [4] (Robot operating system) and HRP[5] (Husqvarna research platform), access to the Automower’s sensor readings can be achieved. ROS will be used for all processes in this project. ROS’s main strength is its inter-process communication. Each process is run as a ROS node, where each node represents a specific component of the overall system, e.g. calculating odometry, reading sensors, etc. Nodes must first register with the master (for the purpose of locating other nodes), then they can communicate with other nodes via ROS topics, a topic can have data published to it (of a specific data type, e.g., Int, Float, arrays, or a custom message), and any nodes that subscribe to that topic will receive the published data, for example this is how data is read from the Automower for use in this project (One node reads and publishes the data, other can subscribe to gain access to it). Each element of the project will have specific ROS nodes associated with it (including mobile device).

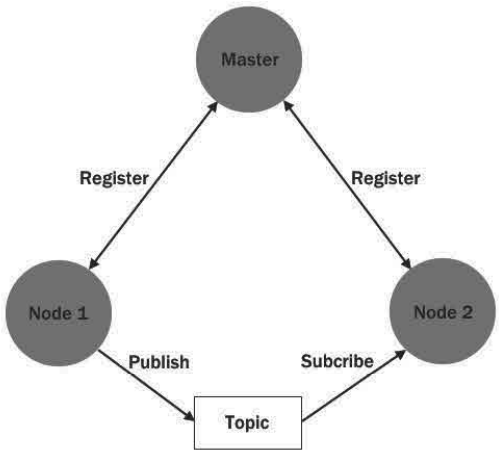

Figure 3: Highlighting the fundamentals behind ROS node communication

In HRP (Husqvarna research platform) is a gazebo simulation of the Automower, this simulation will be used as a baseline for node development, it's useful for testing odometry calculations, however for other aspects of the project, real-world tests will be much more useful.

1.2 Key Objectives

The key objectives of this project can be seen below. The first five are the base line goals, minimum goals to be completed by the due date. Then six and seven are additional goals based on progress/success of previous goals. Current progress on key objectives is discussed in the progress section.

Key Objectives:

1. To develop separate maps of the Automower's environment and position using Odometry, GPS and Boundary Sensor data.
2. Combine and contrast previous methods to measure performance (precision and accuracy of generated map and Automower location). This will allow for evaluation for which methods worked best and why. And also which combinations work best together.
3. Develop a mobile application to interact and display mapping information, the application will be sent the mapping information using ROS over the network.
4. Using generated maps dynamically limit the Automower to or from an area (within mobile application).
5. Using generated maps to increase the complexity in the Automower's path planning algorithm, yielding more efficient (than random) solutions to path inside the area.
6. Additionally develop additional localization and mapping from a visual based SLAM method using a front mounted camera on the Automower.
7. Depending on precision and accuracy of best resulting mapping methods, from a given image, to be able to mow the image into the mowable area.

2 Motivation and Related Works

2.1 Motivation

There are many different SLAM (simultaneous localisation and mapping) methods, and they each aim to output the same general result, to provide information of the autonomous robot's position and to create a map of the corresponding area. However each method will have its own strengths and weaknesses. The main focus of this project is the evaluation of different SLAM methods, and how each of them compares in an outdoor environment, and additionally how they can be fused together, in order to increase accuracy. This project will cover the use of wheel encoders (for odometry), GPS, boundary loop sensors (magnetic field sensors), and a visual SLAM method (to commence in the project's second phase). As a result, this will provide a comparison between the methods (and combinations of the methods) allowing for an objective analysis of what any given system could use, providing an increased knowledge about which methods would be most suited to a given situation.

In its factory state, the Husqvarna Automower navigates in a simple manner (to mow the lawn), simply moving forward, until a boundary loop is detected, then changing to a new direction away from the loop, and then repeating this process again. This solution is a simple solution to the mowing of a given lawn. With the knowledge gained using the aforementioned SLAM methods, practical applications using these methods can be implemented.

Firstly the sophistication of the navigation algorithm can be increased. These SLAM methods provide more information to the Automower about its position and local area, allowing for more intelligent decisions to be made navigating the lawn (and thus mowing it). With an accurate map of the lawn, and Automower localisation, it can now know where it has and hasn't been. With this information, it can mow areas of the lawn that have been left out (where it hasn't been), and avoid areas already sufficiently mowed (where it has been). Secondly, with knowledge of the Automower's own location and its local area, the user can then be allowed to impose specific limits on where the Automower can and cannot go in its local area also, allowing users to do so will yield further practical applications for the Automower.

2.2 Related Works

In the initial phase of this project, odometry, GPS, and the Automower's boundary sensors will each be used for the purpose of SLAM. Odometry is a very well documented method [10], using wheel encoders to make position estimates over-time. The results expected to be seen with GPS are expected to follow those outlined in section 4.2 and will compared as such against them. For the Automower's boundary sensors, [1] makes use of an Automower, and sets the goal to follow the boundary loop at an arbitrary distance using the Automower's boundary sensors. One particular section of the paper highlights the expected boundary sensor readings in a given range of distances from the boundary loop. When conducting sensor measurements and implementing SLAM using the boundary sensors in this project, these results will be used a basis to compare to.

After a map has been generated, functionality will be developed using these maps inside the mobile application, with the intent to allow an end user to restrict/forbid the Automower to/from a specific area of that map. Similar functionality can be found in the indoor autonomous vacuum cleaner, Botvac D7 from Neato [13]. This autonomous vacuum cleaner creates a map of its environment, displays which areas have and haven't been vacuumed and allows the user to impose a no-go zone, which the robot should stay away from. The idea for the Automower is similar in nature, however for an outdoor environment.

3 Specification

This section will outline the requirements for each area of this project, in order for completion of the key objectives outlined in the introduction.

3.1 Hardware

Some of the hardware requirements have been completed by myself in a previous project, however they are worth listing them anyhow, as it shows what has been added to the Automower from its default factory condition. The Automower contains a Raspberry Pi inside it, for computational purposes. Installation of Ubuntu Mate 16.04 on the Raspberry Pi will allow ROS kinetic to be installed, which further allows for the installation of HRP and the required packages for this project to run. Additional hardware required is the webcam, as it is a requirement for visual SLAM, and the GPS receiver is also required for the GPS section of this project.

Requirements:

1. The Automower should have a Raspberry Pi 3 Model B with Ubuntu Mate 16.04, ROS kinetic, HRP and any other ROS packages required in other aspects of this project installed.
2. The Raspberry Pi 3 Model B should have access to power from inside the Automower, ideally from the Automower's own batteries.
3. The Raspberry Pi 3 Model B should have a data connection to the Automower.

4. The Automower should have a webcam to output HD video.
5. The Automower should have a GPS receiver that can receive GPS updates in real time.

3.2 SLAM

Below are the listed requirements for the methods and sensors used for SLAM, the methods are each discussed in detail in section 4. The requirements below are tied directly to the key objectives (objectives one, two and six) outlined in the introduction section of this report.

Functional Requirements:

1. Localisation and mapping implemented with odometry (using Automower wheel encoders)
2. Localisation and mapping implemented using GPS.
3. Localisation and mapping implemented using the Automower's boundary sensors.
4. Visual SLAM researched and implemented.
5. Sensor fusion techniques employed to combine the above methods (e.g. kalman filter)

Non-Functional Requirements:

1. Methods should be implemented in an efficient manner, to reduce delay, for a more optimal solution.
2. Implemented methods should be reliable and robust.

3.3 Automower Navigation & Area Restriction

The requirements below can be split into three sections: navigation using generated maps, as outlined in key objectives five in the introduction. Restricting/forbidding the Automower from a specified area (detailed in key objective four). The final part is the processing of an image to determine which areas need to be mowed more or less, and to imprint the image into the grass (detailed in the key objective seven from the introduction).

Functional Requirements:

1. The Automower should be able to extract data from generated maps.
2. The Automower should be able to make decisions based on map data for where to go.
3. The Automower should be able to plot a path to in the generated map.
4. The Automower should be able to receive position updates in order to execute the necessary movement commands to navigate this path.
5. The Automower should be able to receive updates for areas of the generated maps to be restricted to/forbidden from.
6. The Automower should act on these updates, making sure it does not enter areas it is forbidden from and remains in areas it is restricted to.
7. The Automower should be able to integrate images from the mobile application, so that the Automower mows certain areas more, and other less, so that an imprint of the image is left in the grass.

Non-Functional Requirements:

1. The processes should run efficiently, reducing unnecessary computation, with the intention to minimise delay.
2. The implementation should be reliable and robust.

3.4 Mobile Application

The mobile application will be a way to view and visualise the methods and features outlined above, however the mobile application will also be required to facilitate some of the key objectives outlined in the introduction. Firstly interaction with an image received over the local network to highlight areas that the Automower is restricted/forbidden from (Outlined in the requirements section 3.3 and in the introduction (key point 4)). Also the mobile application should allow for publishing an image to the Automower, so that it can be imprinted into the grass by mowing certain areas more and other less (key point 7).

Functional Requirements:

1. The mobile application should have ROS integrated into it.
2. The mobile application should be able to subscribe to ROS topics elsewhere on the current network.
3. The mobile application should be able to receive images (of generated maps) and display images to the application from a ROS topic.
4. The user should be able to interact with the image, and to highlight particular areas of the image.
5. The mobile application should be able to publish the interactions the user made back to the Automower.
6. To be able to insert images into the mobile application, so that they can be integrated into the generated map.
7. The mobile application should be able to publish images to the Automower.

Non-Functional Requirements:

1. The mobile application should be user-friendly and easy to understand.
2. The mobile application should be reliable and robust.

4 Methodology

This section will discuss each of the methods/sensors used for SLAM, talking about their roles, strengths and weaknesses, algorithms used, error etc.

4.1 Odometry

Odometry is the process of using the wheel encoders of a robot to calculate position and orientation estimates over time. Wheel encoders are fitted to each wheel, each of which measures the amount the wheel has turned. This information can then be used to make predictions about how far the robot has travelled and in which direction. Odometry is a useful technique, however due to its nature, errors that occur, due to wheel slippage, or slightly mismatched wheels, or any other error sources will build up and accumulate over time. As a result, the error in the odometry calculations is unbounded.

In this project, an outdoor environment is the target environment, the more unpredictable terrain will further increase the error the odometry yields. This error will need to be tested to gauge its significance. It can also be compared to an indoor environment to see how the much the outdoor environment affects the calculations.

The Husqvarna Automower is differential drive robot, this means that it satisfies the conditions for the following odometry equations to apply [10].

$$p = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{4.1}$$

$$\Delta x \ = \ \Delta s \sin\left(\theta \ + \frac{\Delta\theta}{2}\right) \tag{4.2}$$

$$\Delta y \ = \ \Delta s \cos\left(\theta \ + \frac{\Delta\theta}{2}\right) \tag{4.3}$$

$$\Delta\theta \ = \frac{\Delta s_r - \Delta s_l}{b} \tag{4.4}$$

$$\Delta s \ = \frac{\Delta s_r + \Delta s_l}{2} \tag{4.5}$$

In the equations above, *p* represents the position, made up of x,y and θ. Δs_l and Δs_r are the readings taken from each respective wheel encoder. *b* is the wheel base (distance between centre of the two drive wheels). In [3] similar equations for calculating the position and orientation of differential drive robots, however the ones currently listed are more simple and easier to implement.

4.2 GPS

One of the advantages of an outdoor environment is that GPS is much more reliable (in comparison to indoors). GPS will mainly allow for predictions for position. Additionally direction can also be calculated, using previous points determined to be accurate to produce heading.

GPS position updates will have a degree of error to them, in an open area the average smartphone using GPS will be accurate within 4.9m[8] of the true value, however the distribution of readings are not linear, as can be seen in the figure below. Most of the measurements are accurate to 1-2 metres (95% less than ~1.9m). This data will be compared against readings done with the GPS receiver connected to the Automower, and thus be able to see if it will fare better or worse than expected.

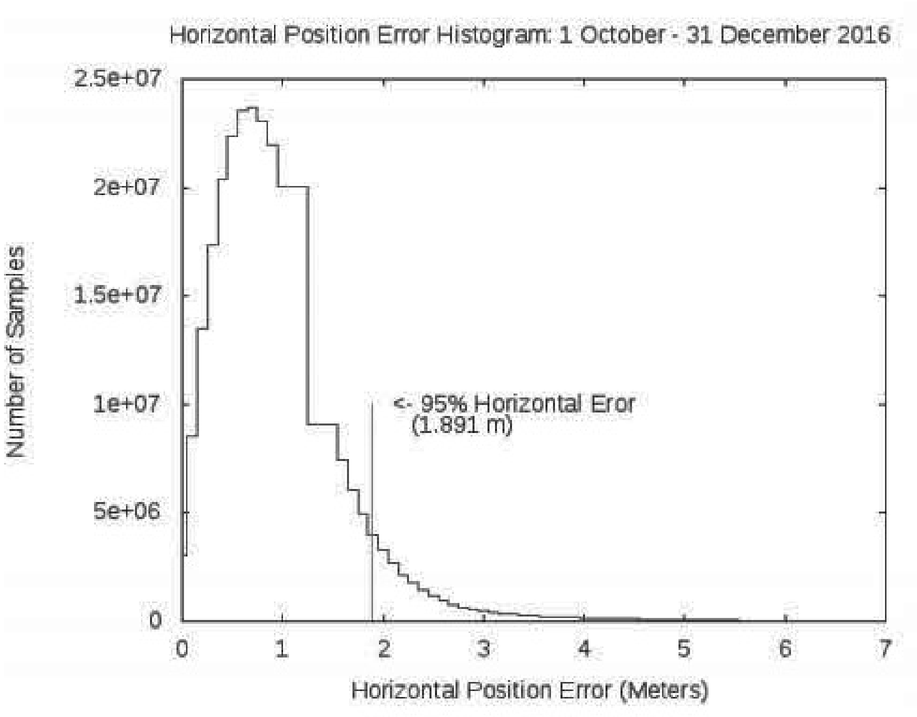

Figure 4: Global horizontal error histogram [9]

4.3 Boundary Sensors

In its factory setting the Automower uses its boundary sensors to detect the boundary loop and to restrict itself inside the lawn, these boundary sensors measure magnetic field strength, more specifically the magnetic field strength of the field produced by the boundary loop. As the Automower moves closer to the boundary wire, the measured field strength increases, up to a point close to the wire where the polarity of the magnetic field switches, as can be seen in the figure below. This switch means that for any given measured magnetic field strength, there can be two corresponding distances from the loop that will yield that result, this increases the difficulty of localisation using the boundary sensors, as we may be one of two distances away. However the polarity switch occurs on top of the loop, so if we can identify if we are sufficiently close to the loop, we can distinguish between the two distance values. One way to distinguish this is by measuring the rate of change of measured magnetic field strength, near the loop the rate of change is at its peak, and then can be identified.

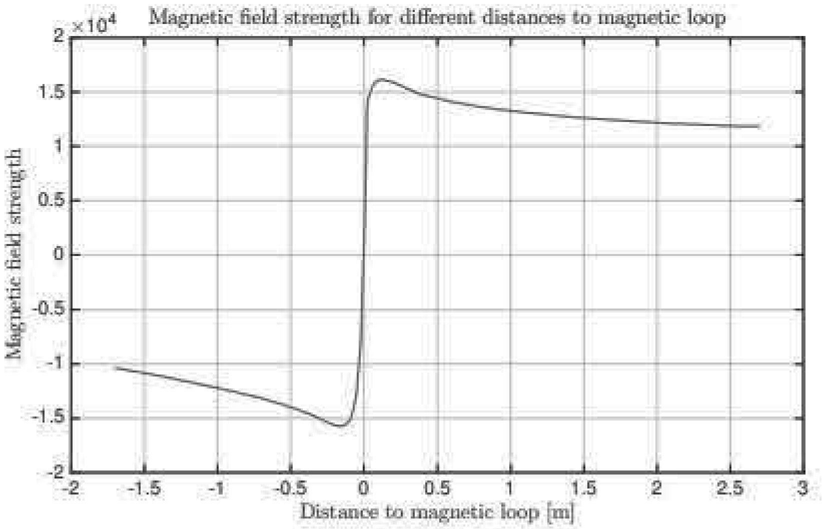

Figure 5: Magnetic field strength against distance to magnetic loop [1]

This same test to measure magnetic field strength will be done with our Automower as the results will be key for using the boundary sensors for localisation.

4.4 Visual SLAM

This area of this project is to use a camera mounted onto the Automower for localisation and mapping. It would be able to recognise features in the environment and then use them to localise against. The research for this part of the project is due to start over Christmas and begins after the previous methods have been completed.

4.5 Combining Methods

In addition to implementing the previous methods individually, they will also undergo sensor fusion, this is where each of the sensors are taken into account (the proportionally dependant on the error they produce). This will be done using a kalman filter[6], as it is an optimal sensor fusion technique (on average). Different combinations of sensors will be tried and then evaluated to see how effective they each are, for example, comparing only odometry to odometry and the boundary sensors, and measuring the improvement.

5 Implementation

This section of the report will detail the currently implemented sections, and outline how upcoming sections to implement will be implemented. Some areas have had progress limited, risk assessment and subsequent safety precautions are still being processed, this will soon be resolved, and testing and further progress can be made.

5.1 ROS and Current Nodes

ROS has been installed on my own personal computer, the mobile application and the Raspberry Pi on the Automower. The installation was the same for both my personal computer and the Raspberry Pi[4], however for the mobile device the installation is different ([12] section 6.1). The non-mobile device parts of this project are being developed as a ROS package called Automower_slam[11], this package currently contains each of the developed nodes/scripts. ROS packages are a way of grouping and organizing nodes together, using launch files for custom launch configurations and to provide an easy manner for software reuse. The package currently contains the nodes or scripts found below:

5.1.1 Odometry

The odometry node subscribes to the “wheel_encoder” topic for wheel encoder readings and using the equations shown in section 3.1 makes position and orientation predictions. The results are then written to file (so they can be plotted by the plot data node), or they can be used to create a map, displaying the path that the Automower has taken (relative to its starting position).

5.1.2 GPS Nodes

The package contains two GPS nodes, one subscribes to the topic “slam/gps_receiver” to read in GPS points (latitude and longitude), and converts them to a local (x,y) co-ordinate system and generates plots the output overtime (the plots are also saved as images). In the future the images will be published to the mobile application. This can be done using the method shown from [12] section 6.6.4.

The second node is a used for testing purposes, so GPS points from file (.gpx files) can be used as a source for GPS for the previous GPS node. gpx files are a type of xml file, so they can be parsed as such. For each GPS point in the file, the latitude and longitude is extracted, along with the time of day the point was recorded at, and converts it from hh:mm:ss.ms to seconds. The node then calculates the time between the current point and the previous point, then sleeps the thread for that time difference, and then publishes the point. This means that each point in the file is published at intervals according to the times they were recorded at, simulating the GPS points as if they were live readings.

5.1.3 Boundary Sensors

Currently without access to the Automower in the outdoor setting, and the boundary loop in the simulator not functioning as intended (readings having no bearing to the boundary loop visualised in the lawn, additionally with no crossover point where the reading was negative). The purpose of the current node is to record the readings from all the Automower’s boundary sensors, and write them to file. The goal will be to take different measurements at different distances to the loop and compare them to the results shown in section 3.3, and to also measure the amount of error in the sensors.

5.1.4 Plot Data

This script is used for testing purposes, where data logged in other nodes and was written to file, this node parses the file and plots the data, and saves the plot as an image. It is done this way as the plotting can be computationally expensive. As the Raspberry Pi has less computational power, it makes more sense to record the data first, and then plot when finished.

5.2 Mobile Application

At this stage of this project a basic mobile application has been created, with ROS support. The mobile application has support to receive images from a ROS topic, currently none of the nodes send images to the mobile application, however once the images have been generated this will be relatively trivial.

5.3 Hardware

In the last project with the Husqvarna Automower[12] various hardware was installed onto the robot. This includes a Raspberry Pi 3 Model B (Ubuntu Mate 16.04 for operating system), Raspberry Pi power pack (for battery), and a webcam. The Raspberry Pi is running ROS Kinetic and can run the automower_slam ROS package. The webcam will be used for the visual SLAM component of the project, as webcam or video recorder is necessary for it. Additionally a GPS receiver (GlobalSat BU-353-S4, compatible with the Raspberry Pi) has been ordered and will be set up and tested after the interim report deadline.

6 Progress

6.1 Project Management and Progress Reflection

The first area of the project was to gain the respective knowledge via research. In previous projects/modules I have had experience working with odometry and GPS, hence why it was not specifically scheduled in. However I was unfamiliar with how the Automower's boundary sensors functioned, so this was scheduled into the plan. The next area of the project is more modular, working individually on SLAM for odometry, GPS and the boundary sensors. Initially scheduled was the testing of the Automower in an outdoor environment, this is key phase, to learn more about the sensors, expected error, for testing implementations and obstacles to progress. However due to unforeseen delays due to risk assessment and safety measures that must go through the school this has not been possible yet. So this has had a significant effect particularly on the boundary sensor area and GPS area (as the outdoor environment contains the boundary loop and GPS is not accurate indoors). However this roadblock should be alleviated soon after this report. These delays have been factored into the updated Gantt chart below.

Even without the unforeseen delay, I would say that I didn't quite allocate enough time for each SLAM method, as each method needs work on the mapping after this report and over Christmas, implementation for localisation was more successful however. Although it is hard to say, potentially with access to real tests in the outdoor environment, progress may have been quicker/more efficient.

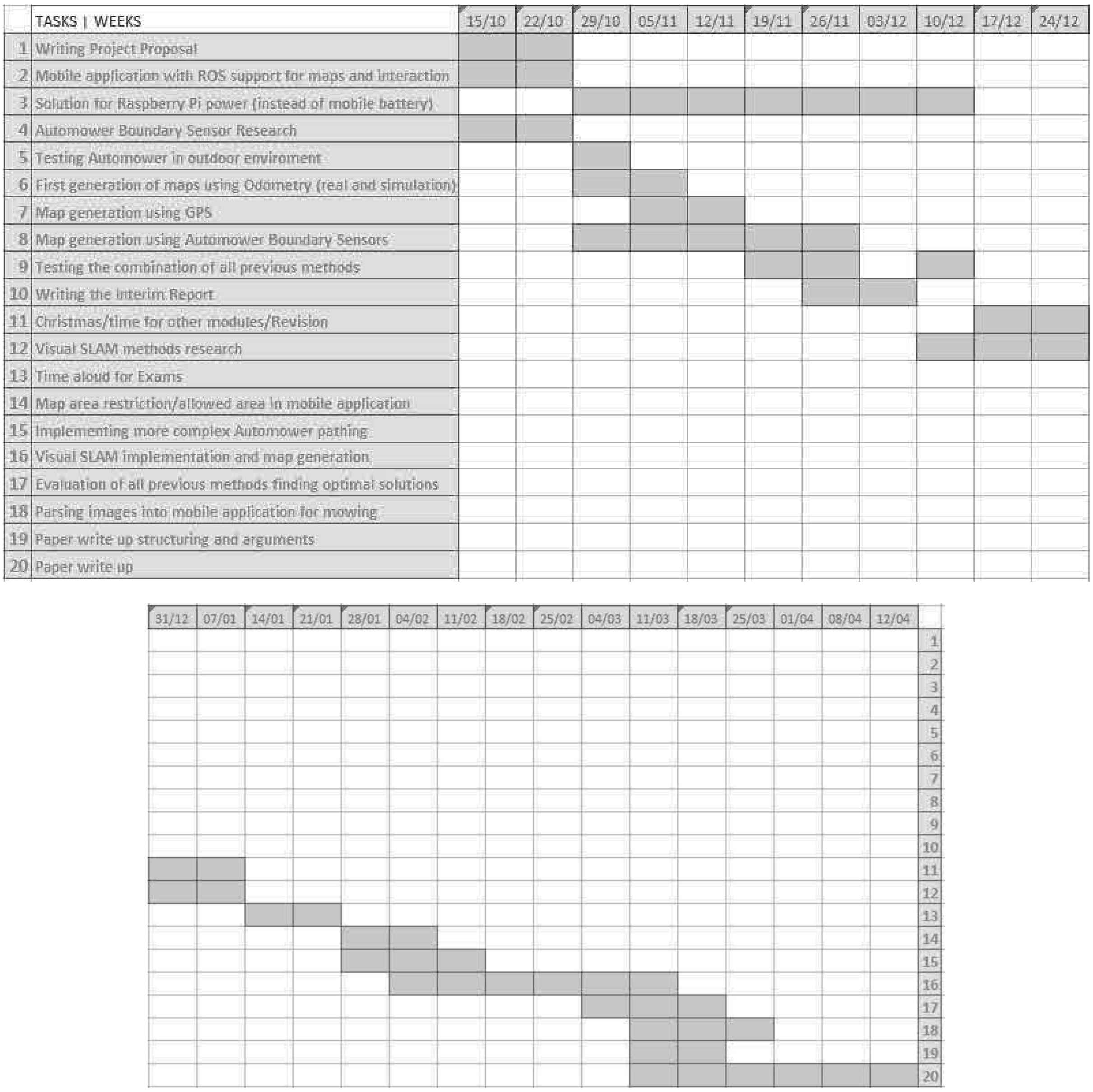

Figure 6: Initial Gantt chart produced for the project proposal

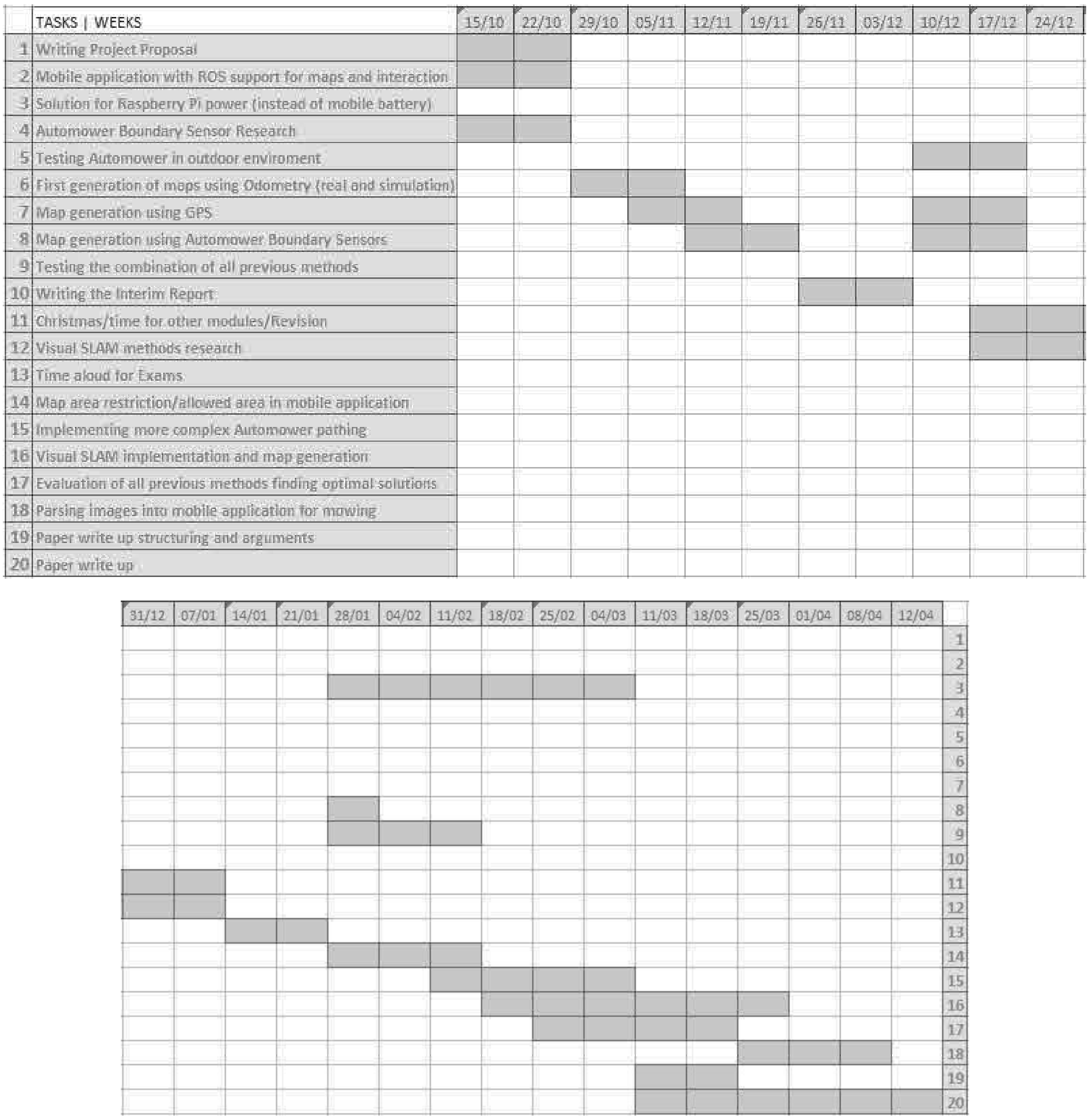

Figure 7: Updated Gantt chart as of the interim report.

6.2 Moving Forward

The interim report is an important point of the project to assess this project's progress and development. As discussed earlier, there was an unforeseen delay relating to the outdoor testing of the Automower, which has had a knock on effect on the other tasks. This has been represented in the updated Gantt chart. Additional time has been granted to the development of the GPS and boundary sensors areas (more specifically, the mapping component for both, and localisation for the boundary sensors), which has in turn pushed back some of the areas of the project (especially parts dependant on their completion). The additional goals of the project are still scheduled (Visual SLAM and then the parsing of images into the mobile application for mowing), however the time to complete them has been reduced, and will depend on the progress made in the core elements of this project over the coming months. To alleviate the effect of the delay, additional time has been freed up over Christmas.

To conclude this section, the delay experienced has had a knock on effect on the work plan of this project. However I believe that I have made the best of the situation, completing work without the Automower, and research completed in the meantime. Also the risk assessment is seemingly coming to an end. Once it has, testing of the sensors and currently implemented areas of the project can then be completed. Allowing the project to move forward and the next areas to begin.

Bibliography

[1] Hannah Dysenius, Nonlinear automatic control of autonomous lawn mower. Linköping University.

[2] Husqvarna Automower 430X, https://www.husqvarna.com/us/products/robotic-lawn-mowers/automower-430x/967646205/ (accessed 05/12/18)

[3] A Tutorial and Elementary Trajectory Model for the Differential Steering System of Robot Wheel Actuators, http://rossum.sourceforge.net/papers/DiffSteer/#d6 (accessed 02/12/18)

[4] ROS Kinetic (Robot operating system), http://wiki.ros.org/kinetic (accessed 05/12/18)

[5] Husqvarna Research Platform , https://github.com/HusqvarnaResearch/hrp (accessed 03/12/18)

[6] Howie Choset, Kevin M Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia R Kavraki, Sebastian Thrun. (2005). Principles of Robot Motion: Theory, Algorithms, and Implementation. MIT Press. p5.p269-p289.

[7] Husqvarna. Husqvarna Automower 420/430X/450X Manual, Chapter 7: Yard Examples, 15/03/2016.

[8] Frank van Diggelen and Per Enge. (2015). The World's First GPS MOOC and Worldwide Laboratory using Smartphones. Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2015).

[9] William J Hughes Technical Center. (2017). Global Positioning System (GPS) Standard Positioning Service (SPS) Performance Analysis Report. Federal Aviation Administration.

[10] Roland Siegwart, Illah R. Nourbakhsh. (2004). Introduction to Autonomous Mobile Robots. Massachusetts Institute of Technology. p187.

[11] automower_slam ROS package, https://github.com/EoinDuffill/automower_slam (accessed 04/12/18)

[12] Eoin Duffill. (2018). Robotics: Control and Interaction via Mobile Devices.

[13] Gizmodo, The Best Thing to Ever Happen to Robot Vacuums is This New Feature http://www.gizmodo.co.uk/2017/08/the-best-thing-to-ever-happen-to-robot-vacuums-is-this-new-feature/ (accessed 06/12/18)