CP34 –
Scope and Requirements Document

Updated: 2020-09-08 10:15 PM

Below is the scope document for the project involving Optimal Path for Drone Delivery tasks using the Gazebo Simulator. Contained in this document is the client expectations for the semester.

Contents

Scope & Requirements

This project is broken into two challenge and problem components that are closely related.

Challenge & Problem 1: Simulator Tasks.

Challenge & Problem 2: Optimal Path for Drone Delivery.

The goal of this project is for students to leave with an excellent understanding on simulation in the drone community and algorithm design for path finding. Students will also gain valuable project management, industry and professional experience.

Scope and Requirements Document
USYD Capstone 2020

High-Level Scope Summary

This project assumes we are flying drones at high and low altitudes. At high altitudes, the drone is expected to be constantly calculating battery level, nearest charging station and distance to destination. The final goal of the project is to allow a fleet of drones to deliver any number of parcels to destinations in the most optimal way autonomously.

A key component for simulation is modelling the battery life, weather conditions and accurate distances. This project is not too concerned about object detection but focused on the algorithms for path finding. Collision Avoidance between drones and buildings should be considered.

- **Milestone 1 – Simulator Tools & Components (Week 3)**
    1. Install and Configure Gazebo Simulator Tools with ArduPilot
    2. Model charging station as a marker (landing zone) in Gazebo simulator
    3. Show how to control the drone using code/scripts (backend)
    4. Show how to create worlds and drop assets
- **Milestone 2 – Simulator Improvements & Further Work (Week 6)**
    1. Build a large simulated city / CBD environment in the simulator (open to suggestions)
    2. Install and Configure Gazebo Simulator Tools with PX4
    3. Documentation for placing objects (signs, traffic lights, objects)
    4. Autonomous drone flight possible and demonstrated.
    5. Multiple drones in the simulator at the same time.
- **Milestone 3 – Sign & Object minor sign detection (Week 6)**
    1. Show primitive detection / model working on initially provided 4 signs (stop, turning, parking, traffic lights).
    2. Autonomous flight with detection active.
    3. Demo in simulator (different environments / worlds)
- **Milestone 4 – Sign & Object major detection (Week 9)**
    1. All outlined signs and objects (total 12 unique objects in table) detected and corresponding action response from the drone.
    2. Demo in the simulator (different environments)
    3. Show working for both PX4 and ArduPilot environments
    4. Simulator improvements completed and demo / video showing process to import new objects and create worlds.
- **Milestone 5 – Completed Solution with Demo and Documentation (Week 12)**
    1. Usage documentation (as per each Milestone)
    2. Demo of working solution in simulator
    3. Meets all ‘test conditions’ defined below
    4. Deliverables as per ‘Hand-over deliverables’ section
- **Extended**:
    1. Students welcome to create extra worlds for the purposes of testing in the Gazebo simulator.
    2. Students welcome to model extra objects in the simulator.

Further Requirements and Conditions

Optimal is defined in three ways:

1. **Time** – deliveries should be prioritised before charging. It is vital that all deliveries are made in the shortest amount of time.
2. **Distance** – how far the drone must fly
3. **Remaining power** – how much range is left at the end of flight.

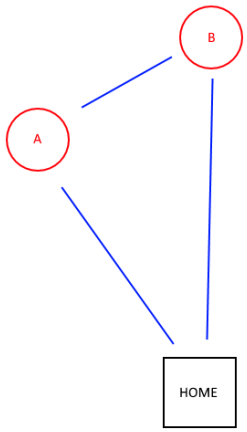
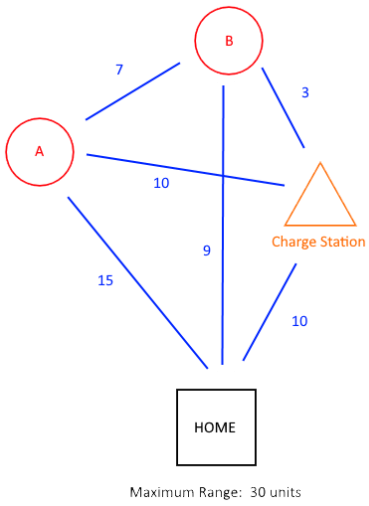
Assumptions & Conditions & Rules:

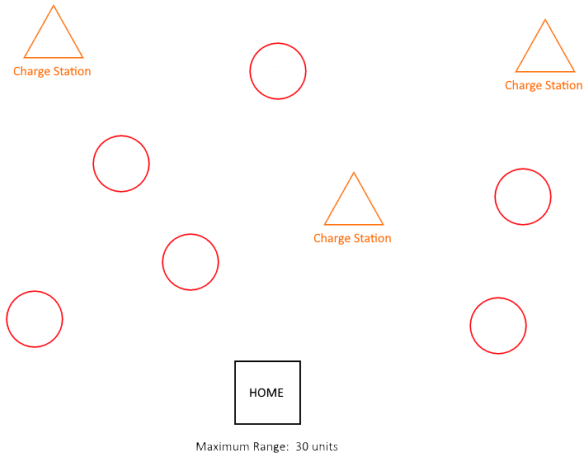
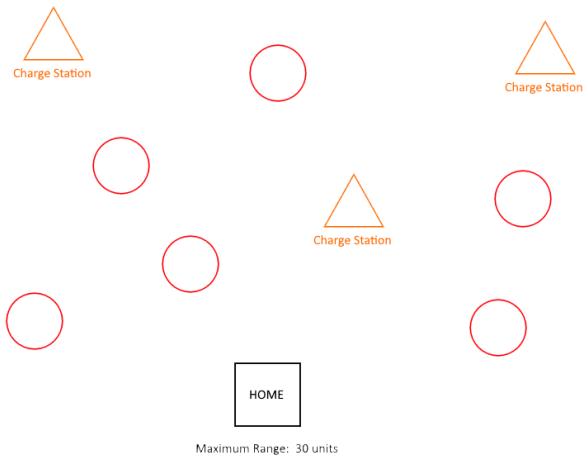
- **Units**: There are no defined distances or times now until the simulation is setup, and these can be worked out in metric units later. For now, everything is in relative units for example purposes.
- **Range**: This is the distance or amount of time the drone can fly before falling out of the sky. For the purpose of these examples, the range has been fixed at 30 units. In the demos provided for the project, the battery life will be reported in milliamp-hours (mAh) or a percentage. This will be related to a distance in metres based on consumption and flight conditions.
- **Charging Stations**: These are locations where a drone can replenish its battery supply and restore range. There is a time penalty for stopping to charge a drone. This is going to be set at 20 units. However, a drone can be partially charged and leave early from a charging station. Charging will be linearly related to range for simplicity.
  Example:

Charge Percentage	Time Elapsed (units)	Range (units)
0%	0	0
25%	5	7.5
50%	10	15
75%	15	22.5
100%	20	30

- **Standby Drones**: In the more complicated examples, 'standby drones' are used. These are drones that are parked at a charging station and able to take over the journey of a drone that may not have enough charge. These have a time cost of 5 units when utilised. The reason for the delay is transferring the parcel.

Name	Description
Point to Point – single drone	Fly a drone in the simulator from a defined point A to defined point B within the range of the drone. Then fly home.
Point to Point – multiple drones	Fly a drone in the simulator from a defined point A to defined point B within the range of the drone. Then fly home. At the same time, launch a second drone from point A to fly to defined point C. Then fly home.
Range	Show that the maximum range of the drone can be reached. Show that the drone can no longer fly when it runs out of battery.

Two Destinations	Assumption: a drone has multiple packages and enough battery life to complete the journey. Two destinations. Autonomous / Algorithm used to show a drone decide which will be an optimal path for delivery and then returning home. In this case, it should complete a circuit based on distance.  This is the simple test case.
Two Destinations – Charge Station without standby drone	Assumption: a drone has multiple packages and not enough battery life to complete the journey. It will need to stop to charge. Two deliveries. Autonomous / Algorithm used to show a drone decide which will be an optimal path for delivery and then returning home. The drone will need to recharge after set distance; however, it should decide what is optimal.  Maximum Range: 30 units Note: route needs to be 'optimal' as per above definition.

Multiple Destinations – Charge Station(s) and Standby drones	Scale the previous example up to any number of deliveries / destinations with any number of charge stations. Include variables: - Weather impacting on performance - Single Drone all parcels / deliveries at the start - Standby drones as relief for starting drone  Note: route needs to be 'optimal' as per above definition.
Multiple Destinations – Charge Station(s) and Standby drones	Scale the previous example up to any number of deliveries / destinations with any number of charge stations. Include variables: - Weather impacting on performance - Multiple Starting Drones with random number of deliveries to make at the start - Standby drones as relief for starting drones - More complex battery life calculations based on the simulator's tools  Note: route needs to be 'optimal' as per above definition.

Scope and Requirements Document
USYD Capstone 2020

Client Expectations

General

This project is broken into two major components each with two milestones. This scope document will outline the expected delivery of each milestone as per the *High-Level Scope Summary* list found below. Students are also expected to do lots of research to ensure the best solutions are found, with research recorded in the appropriate repository and format. Documentation must be included with all work completed in a format that is legible and understandable.

The client reserves the right to amend the scope of the project throughout the semester based on team progress and unforeseen challenges that may arise. The client reserves the right to communicate problems with tutors to be raised with the course coordinator to take academic action if deemed essential for lack of progress or not meeting expectations.

Contribution

The client (Robotics Masters) expects that students meet and exceed the scope and requirements outlined in this document to achieve the best grade in the course. Each individual student in the team are expected to contribute to the project. Contribution and time expected per a week on this project is approximately 15+ hours per week for each student, excluding meetings. That is a total contribution of 60 to 90+ hours per week for each team.

Contribution to the project will be tracked through Bitbucket (there should be constant daily commits being made by all team members), communication (activity on Discord Server) and through all updates provided to the client. This information will be used at the end of the semester when it comes to evaluating everyone's contribution and final grade for the capstone unit.

Meetings

Each team is expected to attend two Zoom meetings per week. The times have now been sent to everyone.

Each team member is expected to submit and present a 1-minute summary of their contribution for each week on Monday/Tuesday meetings. These videos should be uploaded to YouTube as an unlisted video, presented during each Monday/Tuesday meeting and shared to the client after the meeting. Feedback will be given after each individual video during the meeting.

Any technical questions that arise should be asked during these meetings.

Each team is to present and submit the plan for next 'sprint' period at the end of each meeting. This should contain a list of tasks that each team member is working on for the next time period and outline what is expected to be delivered at the next meeting.

Communication

Email – the client prefers email communication for official documents, scope questions and communicating with tutors. The client will also respond to technical and scope questions via email, however, would prefer that the Discord Server (Sydney 2020) is used.

Discord – the client has selected Discord and email as the preferred tool to use throughout the semester for questions and notices. Teams will also be collaborating/discussing/sharing ideas with other teams run by Ben Sand (another client). There is a total of 15 capstone teams taking part in this initiative. This kind of collaboration has been done before and it has been very positive for

students and opens several different opportunities, such as meeting tech influencers, established developers and other technology specialists.

Events

Students are expected to participate in the following activities throughout the semester as part of the core component of this project and to ensure understanding of the project

Venture Café - Students will be required to participate and present at two Venture Café sessions throughout the semester. It is an online meeting community where different creators, start-up owners meet and discuss what they are doing with their own projects. It works like a 'drop-in session' where people may only join for a short amount of time or could stay for longer. Teams will be sharing your progress and discoveries through this platform. Teams are welcome to attend Venture Cafe on any Thursday to get a feel for the platform and chat with other technology people.

Hand-over Deliverables

Documentation – All documentation is to be created and written in Markdown, then build with Readthedocs for portability and usability.

Maintainability – The client would like the code developed to be able to be integrated into the PX4 and ArduPilot repositories cleanly on completion. Students are expected to document the changes and improvements and leave the code in a way that it can be merged by pull request to the corresponding repository.

Test Code – Where students have developed code for testing detection, this is to be uploaded onto Bitbucket and provided to the client at the end of the unit. It is preferable that students include intrinsic documentation within each file explaining its purpose.

Reference Material

All reference materials will be provided by the client to teams via Discord or email.

End of Document