COMP3888 - phase 3

September 2020

1 Introduction

This document outlines the design of taking weather impact into algorithm. The core thinking is to convert the effect into a change in the length of a path which will be reflected in Dijkstra's.
IMPORTANT: this step shall be conducted before battery check algorithm

2 Assumption

- The weather shall be in multiple forms and have different impact towards drone performance respectively. (Hint: possible polynomial classes will do the trick
- Weather is limited to affect a specific area, represented by a shaped 3D object. (Hint: possibly cubes with 8 coordinates like

  $$C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7$$

)
- The affect of weather is computable, ideally, can be converted into a function which takes in some parameters and output optimal result.

  $$f(length, ...) = ... =?$$

3 Reduction

- Given a weather object and a path, calculate the length of this path which is within the effective range of weather. This should be computable, given a line and a space.
- Feed this length into a function and retrieve the equivalent length to fly within this area. For example a rainy path with a length of 300 meters can be converted into a usual path with a length of possibly 400 meters.

  $$f(300) = ... = 400$$

- Reflect this change in the original path. For example if this is a 1km path with 300 meters affected by rain, it can be updated to

$$NewLength = NoneAffectedLength + f(AffectedLength)$$

$$NewLength = 700 + f(300) = 700 + 400 = 1.1km$$

  Note: not necessarily all weather has negative affect on performance. E.g. A correct wind direction

- Repeat the steps on every combination of path and weather affected areas

4 Complexity

Assuming there are m weather affected areas, and within each step, computation and updating edges can be done in $O(c)$, the total running time would be

$$O(m) \times O(n^2) \times O(c) = O(mn^2)$$