Aircraft Route Guidance through Convective Weather

Lawrence Li
Department of Engineering Science
The University of Auckland
New Zealand
lli143@aucklanduni.ac.nz

Abstract

When planning and choosing a flight route to take for aircrafts, an important problem is to determine whether or not a pilot should fly through or to fly around convective weather, so that the air travel delays and costs are minimised. One method of determining if a pilot should fly through or not is to find the shortest path which minimises the total distance and risk a pilot would take for a route.

The aim of the research was to determine a set of route choices for guiding pilots through convective weather based on available weather data. The problem is modelled as a bi-objective shortest path problem, which is a shortest path problem with two objectives. A flight network of the airspace is generated firstly as a 2D network, and then a 3D network.

Key words: aircraft, bi-objective shortest path, convective weather, weather avoidance

1 Background

1.1 Introduction

Convective weather is the process of rising air, which is generally warm moist air above cold air, to form clouds. This process is normally associated with heavy rainfall and hail and is known to produce strong, turbulent winds and thunderstorms. This severity is of concern, as rain and clouds can reduce visibility greatly for aircraft pilots while in flight, and the strong winds can reduce or even overwhelm the pilot's ability to fly safely. In addition, hail, icing, and thunderstorms can damage aircraft sensors and mechanical parts, which can endanger the safety of the passengers onboard the aircraft.

There are several methods that are used to find the best route to take based on the available weather data. One method is to minimise the distance an aircraft will travel and to minimise the risk the path imposes to the aircraft, for which a number of optimisation methods can be used. These methods are able to determine the best set of routes to take in terms of distance and safety, allowing the pilots to choose a route that matches their situation and needs. This research aimed to find a set of route choices for guiding pilots through convective weather by modelling the problem as a shortest path with two objectives.

1.2 Weather Risk

The precipitation intensity is one measurement used by flight dispatchers and meteorologists to help determine the severity of weather. In general, it is the amount of liquid within the atmosphere, which includes rain and hail. One method of measuring this is with vertically integrated liquid (VIL), which shows the amount of water contained in a vertical column. This is obtained from measurements of the amount of reflection from air via weather radars. VIL is a measured on a scale from the one to six, with one being the lowest level of intensity, and six representing the highest level of intensity.

Based on observations of true pilot behaviour over a period of 40 days, it was consistently found that aircrafts avoided weather that was at VIL levels from three and above (Kuhn, 2008). At each level below this however, more aircrafts were found to fly through the weather. In some cases, pilots would deviate from their current route even though it was passable, and in other cases, pilots would penetrate through the weather when it was deemed unsafe. This would indicate that there are other factors involved in the decision making process for pilots that has not been taken into account when the flight route was created. This shows that VIL alone makes for a poor predictor in determining a pilots' decision to deviate or penetrate, as there is some uncertainty in determining whether or not a pilot should fly through based solely on precipitation intensity.

The echo top is another measurement used by dispatchers, and currently is a major factor involved in a majority of weather-related deviations (Kuhn, 2008). The echo top shows how high the precipitation or storm reflects and extends upwards; in other words, it shows the height of the top area of the precipitation, relative to the ground. This should not be confused with the height of the clouds or storms, which is generally higher, as it generally does not contribute towards the storm intensity.

Echo top is of great value in the aviation field, as these heights provide an indication of pilot's intent to penetrate weather, relative to the altitude of the aircraft. This is because echo tops are useful in identifying areas of air currents, where high intensities of air currents would indicate the likelihood of severe turbulence if flown through.

Echo tops alone however cannot identify all severe weather, and are hence interpreted together with other measurements, more notably with VIL. Together with VIL, the intensity of precipitation and the movement of vertical air can be determined, allowing for a more comprehensive overview of the weather severity.

1.3 Problem

The size and quality of the search space presents a limitation on which an algorithm can perform, as larger search spaces in terms of the number of nodes and arcs would increase the memory usage and computational time to find efficient paths.

This project aimed to determine a set of route choices for guiding pilots through convective weather based on the real weather data, and utilised the bi-objective shortest path (BSP) problem to address to find these sets of efficient paths. The flight network in which the aircrafts flew was generated to try and best model the conditions the aircrafts could fly in, in order to achieve the most efficient paths possible. Routes obtained by the BSP program would be compared against different models of realism to determine the flight networks' capabilities in determining the route choices for guiding pilots through convective weather.

1.4 Mathematical Formulation

The problem can be formulated as a bi-objective shortest path problem. The problem can be described as follows:

Let the digraph, or directed network, be represented as G = (N,A), where the set of nodes is N = {1,...,n}, and the set of arcs, or directed edges, is A = $\{(i_1,j_1),...,(i_m,j_n)\}$, which joins the nodes in N. Also, we let P_{ot} represent the set of all possible paths, p, from the origin,o, to the target node, t.

For each arc (i,j) ∈ A, two costs (r_{ij}, d_{ij}) are associated with it, which in this case represents the risk involved and the distance travelled when an aircraft travels on that arc. The distance travelled along a path from the origin node to the destination node can be obtained by summing the length of the arcs, a, and the risk along a path by summing the risk scores.

We wish to minimise the sum of all distances for each arc on a path, and also to minimise the sum of all risk scores along this path. Each path belongs to a set of all possible paths connecting the origin to the destination.

Minimise $d(p) = \sum_{a \in p} d_a$
Minimise $r(p) = \sum_{a \in p} r_a$
subject to $p \in P$

With this, the goal is to obtain routes where it is not possible to obtain one with a better objective value in one aspect without compromising the other. This will allow paths with trade-offs between the most safest and the most direct routes to be found, and subsequently allow pilots to choose generated routes depending on their preference, circumstances, and experiences.

2 Methodology

2.1 Flight Network

The flight network represents the airspace in which the aircraft travels. To model the aircraft as it travels through the airspace, the area in which it travels can be split and discretised into grids of squares. These squares represent a node, or a point, at which the aircraft would be within the airspace at any given time or place. Each of these nodes are connected to other neighbouring nodes by arcs, which represent where the aircraft travels in the airspace from one point to another, like a series of waypoints.

The modelling approach proceeds in two states for the flight network, where each state progressively adds more realism to the problem. Within a 2D environment, we consider the altitude and weather as static; in other words, the plane is flying through static, non-moving weather at a fixed altitude. A model such as this is best used when the aircraft is moving very fast relative to the airspace area, as it is unrealistic otherwise to assume weather being static and not move at all. In a 3D environment, we consider only the weather being static, with the plane being allowed to change its altitude to avoid convective weather.

This setting can be seen as an improvement towards the 2D model, as the aircraft can now change its flight level to compensate for any severe weather it encounters. In this case, real weather data was available for use during the generation of the flight network, however in a real-time planning environment, it should be noted that weather forecasts are utilised.

2.2 Distance and Time measure, d_{ij}

Distance is an important measure in this model. For airlines, the distance their flights have to travel approximately equates to the time it takes to travel to a destination, and also the amount of fuel used. It is in the airlines' best interest to reduce the distance it takes to travel, as this will allow them to save on operating costs, and even on the hourly wages it pays to flight attendants.

2.3 Weather Risk Measure, r_{ij}

Weather, and the risk this imposes to the aircraft in the airspace, is another important measure in this model. The weather for this model is based on real weather data that had been obtained. The airspace in which the weather was obtained from is a set airspace size of 336 by 364 kilometres squared. Each kilometre square has weather data pertaining it with VIL and Echo Top data, measured with weather radar. The VIL and Echo top data represent the different levels of the precipitation and heights of the clouds.

To determine the risk involved for an aircraft travelling within the airspace, another set of weather data is used, which is the occupancy. The occupancy is the likelihood that any aircraft will travel in an area relative to VIL and echo top data in that area. When VIL and Echo Top was relatively high, the occupancy was relatively low, and vice versa. In between, and with an extreme of one particular weather data, the occupancy was more varied in nature, and was not as clear.

To compute the weather risk rij, the VIL, and echo Top values are determined based on the position in the Cartesian coordinate system. From this, the values are then looked up against the occupancy table to determine the likelihood of an aircraft travelling through the area. High occupancy values are given low penalties, and likewise lower occupancy values are given increasingly higher penalties. The penalties themselves are arbitrary values determined to best reflect what is believed the aircraft should be avoiding and not avoiding. With empirical calibration though, it can be expected that the paths obtained would give much better results that have a high correlation with actual flight paths that are tracked.

2.4 Flight Network Generation

The flight network is generated with the use of MATLAB to create and represent the airspace in which the aircraft travels within two dimensional, three dimensional, and four dimensional space. The flight network grid is a necessary component in this model, as this defines the airspace and available nodes in which the aircraft can travel to and from.

To recreate the airspace in which the aircraft travels in 2D, we first take the size of the weather air space into account, as this determines where the aircraft can travel. Afterwards, we use this to set the upper limits onto which the aircraft can travel up to within the airspace.

2.5 Two-Dimensional Space

To generate the flight network in two-dimensional airspace, the following pseudo-code given in algorithm 1 details the basic idea behind the generation of the network.

Algorithm 1 – 2D Flight Network Generation

1. Input – weather data, source and destination, occupancy data, altitude
2. Initialise – Distance values, search parameters, altitude height
3. *for* y = 1 to maximum longitude
4.   *for* x = 1 to maximum latitude
5.   *for* all possible headings (North, NE, East, SE, South, SW, West, NW)

```
            if heading is possible 1 step ahead
                Calculate echo top index
                Calculate weather penalty by looking up occupancy
                Output current (x,y), new (x,y), distance, and weather penalty
            end if
        end for
    end for
end for
```

2.6 Three-Dimensional Space

To generate the flight network in three-dimensional airspace, the inclusion of another nested loop before the longitude in algorithm 1 to take into account the altitude would be required. As a result, the possible headings in which a node can connect to increases as well, as it can go either above or below the current node. However, the speed at which the network was generated would slow down considerably.

The algorithm was improved by reducing the number of nested loops to improve the speed of the network generation. Before the improvement, the generation would have been estimated to take several days for 3D depending on the altitude, and could well take over a couple of weeks for 4D for at least 20 units of time. With the reduction of nested loops to just one loop, the generation of a 3D network dropped drastically to around two hours on a standard Intel Quad-Core computer.

The improvement made was to cycle through all the possible nodes in the airspace, given the size of the weather and altitude, and then re-converting it back to Cartesian co-ordinates, and can be demonstrated in algorithm 2.

Algorithm 2 – 3D Flight Network Generation

```
Input – weather data, source and destination, occupancy data, altitude
Initialise – Distance values, search parameters
for index = 1 to maximum nodes
    Convert to Cartesian co-ordinates
    for all possible altitude levels (current, above and below)
        for all possible headings (North, NE, East, SE, South, SW, West, NW)
            if heading is possible a pre-defined step ahead
                Calculate echo top index
                Calculate weather penalty by looking up occupancy
                Output current (x,y), new (x,y), distance, and weather penalty
            end if
        end for
    end for
end for
```

2.7 Output

The output of the files utilises the Forward and Reverse Star representation to store arcs that come from the nodes into a single array structure. The advantage of this method is that it saves space, is efficient for manipulation, and is suited for dense and sparse networks (Zhan, F., 1998). Each row represents an arc, and the four columns represent the starting node of the arc, the ending node of the arc, and the respective costs for distance travelled and the weather risk at the ending node of the arc.

The size of the problem is dependent on the size of weather data provided. For an airspace of a given size, by increasing the number of weather points available and making the “resolution” of the weather clearer, the number of points increase dramatically. The number of nodes and arcs will increase along with the number of weather data points available, which makes for a more refined path for the plane to follow. For a given problem of the a size of roughly 336x364 with a point of data in between, the total possible number of paths go up to 974236 for a 2D environment.

2.8 Spatial Size

The network size can be reduced, to speed up and simplify the time required to generate the flight network. The end result shows the grid being similar in looks but at a reduced quality.

The cell sizes are discretised such that for each defined area of space in the original airspace, say 3x3 for example, this now becomes 1 square in the smaller version. The underlying VIL and echo top data from the original airspace are also averaged for the new cell, which is averaging the value of the cells. This is one method that can be used to improve computational speed and efficiency.

This effectively reduces the network size, however of note is that the distances have to be compensated by a factor of three for accuracy.

2.9 Bi-objective Shortest Path Algorithm

The bi-objective shortest path (BSP) algorithm is a label setting algorithm, which is an extension of the single objective label setting; Dijkstra's Algorithm. An implementation of this algorithm, in the form of a program by Andrea Raith (Raith & Ehrgott, 2008), was used to help find efficient paths for this model. A number of different algorithms are available for the user to select from when running the BSP program.

The program was coded in the C programming language, and runs on Linux. The parameters the program takes to run is the input file, and a number corresponding to the algorithm to be used, along with any subsequent parameters that need to be specified with the algorithm, if any.

3 Result

The implementation of the weather avoidance system is compared as viable tools for the pilots and dispatchers. The models compared will be the 2D environment, which is with static weather and a fixed altitude, the 3D environment, which is with static weather.

The input weather given for this comparison of the program is of the same weather data as the 2D to provide a grasp of the differences the algorithm has to account for with the inclusion of altitude.

It can be noted that the algorithm produces paths that are in line with the 2D paths generated, however, as the emphasis on safety increases, the utilization of altitude becomes more apparent.

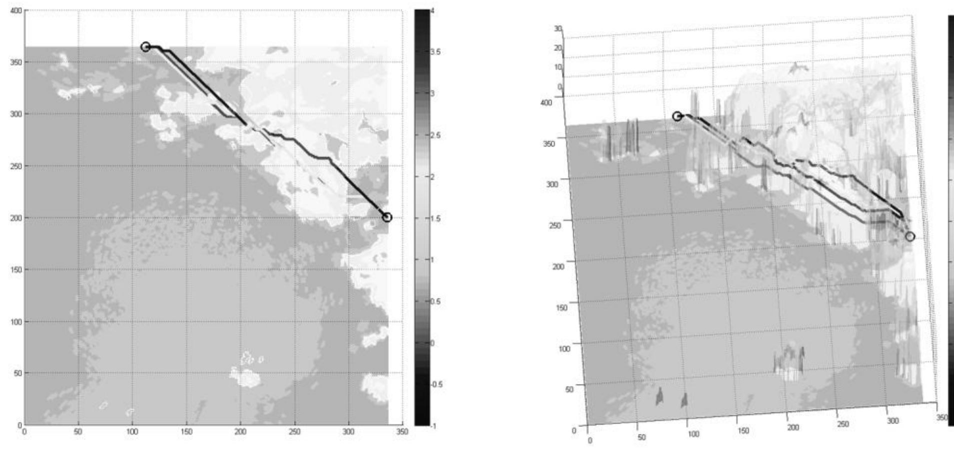

Figure 1. Representation of weather over an airspace, with VIL data over echo top data for cloud heights and severity, and with efficient flight paths overlaid on top.

It can be seen that from figure 1, the paths follow a similar path from the origin to its destination. When looked at from a slightly angled view of the paths, the routes show differences in altitude. Those closer to the ground show more emphasis towards the shortest distance, whereas those higher up, in an attempt to fly above the clouds, show more emphasis towards safety.

What is peculiar is that the paths do not ever leave the weather at all. It can be deduced that once a particular deviation is reached, anything above that will result in a compromise in the distance travelled that is so great that it outweighs the distance.

4 Conclusion

The project aimed to determine a set of route choices for guiding pilots through convective weather based on the available weather data given. This was modelled as a bi-objective problem in order to give pilots the freedom of choice of routes between minimizing risk and maximising efficiency. In particular, flight networks of a two-dimensional, and three-dimensional setting were explored to help achieve this.

5 Future Work

Future work expands to a wide variety of possibilities which have yet to be explored, such as inclusion of aircraft performance constraints like angles of attack and headings for realism, the implementation of 4D with respect to time for dynamically changing weather and comparing actual flight paths to efficient paths obtained with the program. Further work would also include improving the solution algorithm and allowing multiple aircrafts to be routed in the same airspace, instead of just one.

Acknowledgments

I would like to acknowledge Andrea Raith and Kenneth Kuhn for the tremendous support and help that they have given me throughout the year. Additional thanks to Christopher Vogel, David Ryan, Imran Ishrat, and Ming Cen for their help on aviation and simulation related problems.

6 References

Kuhn, K. (2008). "Analysis of Thunderstorm Effects on Aggregated Aircraft Trajectories." Journal of Aerospace Computing, Information, and Communication 5, 108-119 (2008).

Raith, A. and M. Ehrgott (2009)."A comparison of solution strategies for biobjective shortest path problems." Computers & Operations Research 36(4): 1299-1331.

Zhan, F., (1998). "Unit 064 - Representing Networks" Retrieved 8th August 2010, from http://www.ncgia.ucsb.edu/giscc/units/u064/