EPSRC

Engineering and Physical Sciences
Research Council

InFoMM
Industrially Focused
Mathematical Modelling

EPSRC Centre for Doctoral Training in Industrially Focused Mathematical Modelling

Dynamic sectorisation over the West End region of UK airspace

Lingyi Yang

UNIVERSITY OF
OXFORD

NATS

Contents

1. Introduction

The increasing demand for travel and transport by air adds pressure on the air traffic industry to increase capacity whilst still maintaining safety standards. Initiatives have been started across the globe to facilitate the modernisation of practices within this field. NATS is an Air Navigation Service Provider (ANSP); they provide Air Traffic Control (ATC) services within the controlled regions of the UK airspace as well as at many airports both in the UK and internationally.

Air Traffic Control

The UK airspace can be viewed as a three dimensional jigsaw puzzle.

The UK airspace is divided into three Flight Information Regions (FIRs): London, Scottish and Shanwick Oceanic, see Figure 1. These FIRs are then further divided into local areas as shown in Figure 2. The areas are split into even smaller parts, which we call elementary sectors. Regions of the airspace can have different classes, each with different rules and procedures: for example, they can be uncontrolled or controlled. In the uncontrolled parts of the airspace, the responsibility for safety is placed on the pilot.

Figure 1 – The three FIRs for the UK. From *www.nats.aero*.

If an aircraft wants to fly through controlled airspace, they must file a flight plan prior to their departure with their intended reference points to fly over as well as the time and height at which they wish to do so. When aircraft pass through the controlled regions of the UK airspace, they are managed by air traffic controllers (ATCOs). Aircraft being managed have to maintain a minimum horizontal separation of 5 nautical miles as well as at least 1000 ft vertically. The training of ATCOs are specific to the areas they will control. ATCOs work in pairs and together they supervise a portion of the area that they are trained on; this can be just one elementary sector or a group of elementary sectors depending on how busy the traffic is. We call the area that is being controlled by one pair of ATCOs a 'controlled sector'.

During the night, many airports in the UK (particularly the London airports e.g. Heathrow, Gatwick) operate under a night quota period, where the movement, take-off, and landing of aircraft are restricted. As a result, the airspace is a lot quieter during night hours (typically 23:30-06:00). During these low traffic times, it is therefore efficient to merge a greater number of elementary sectors together to form larger controlled sectors compared with the daytime.

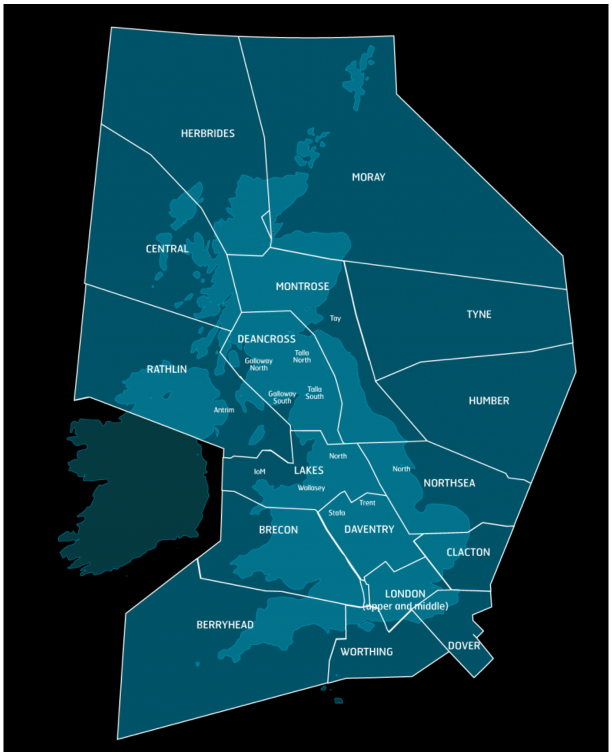

Figure 2 – Further splitting of the UK airspace into local area groups. From *www.nats.aero*.

This merging process is known as 'band-boxing'. When the traffic increases again (as the morning arrives), the combined sectors will once again be split to form smaller controlled sectors, should the workload become too much for one pair of ATCOs to manage. We call this process of band-boxing and splitting the elementary sectors (i.e. changing the configuration) 'sectorisation'.

Workload and Complexity

Workload within the context of ATC is usually defined as the mental and physical strain placed on the ATCO. This is, therefore, a subjective measure based on the experience and health state of the ATCO as well as the physical components of the airspace and aircraft. The complexity of the airspace is the term which encompasses the factors that can typically be physically measured, for example, movement of an individual aircraft and how they interact with one another. Hence workload can be thought of as a function of ATCO-specific features as well as the complexity of the airspace.

Due to the subjective nature of the workload, there is no unified model to define workload or complexity. There are some intuitive ideas which are often used in arguments. As the number of aircraft in a sector increases, the number of potential interactions/conflicts also increases, therefore we expect that the workload on the ATCOs will also increase. However many other factors can play important roles as well and these may overshadow the number of aircraft. For example, take the situation where 10 planes are flying in parallel across the sector. Since there is no risk of collision, this situation may be far less complicated than if 4 planes are flying towards each other. From this example, can we say that the number of different directions the aircraft are flying matters more? If so, how much more? When we think on a three-dimensional level, we have to factor in the changing heights of aircraft as well. Having aircraft flying at different speeds would be more complicated in terms of tracking multiple flights at the same time. The geometry of the sector can be of importance too. One type of

There is disagreement in the field as to what defines workload and complexity.

behaviour may cause no issue in one sector but be a big problem in another.

Another way in which the capacity of airspace is limited is that, as an aircraft leaves one controlled sector and approaches another, the pilot will make voice communication with the new controller. This ATCO then has to respond as soon as possible. The speed at which aircraft can enter depends on the physical limit of how quickly the ATCO can respond.

Dynamic Sectorisation

How can we automate changing the configuration of the airspace?

Currently, there are some computational tools available that provide limited predictions about how busy the airspace is likely to be, so that supervisors can plan the configurations of the day with the ATCOs. With the development of technology, the question is whether the splitting and merging of the elementary sectors throughout the day can be automated through the use of an algorithm. We call this dynamic sectorisation or dynamic airspace configuration. There has been a wide range of research on this topic over the past few decades. Dynamic sectorisation is split into two parts, the first part is determining the elementary sectors and the second is finding the optimal configuration, i.e. the elementary sectors that should be grouped to form controlled sectors.

For the elementary sectors, we can either adopt the existing ones used by the air traffic controllers or create new ones. In the past, a range of methods have been used to create new elementary sectors such as splitting the (2-D) map into hexagonal cells. A variety of methods have also been proposed for finding the optimal configuration, such as integer programming and in more recent years machine learning.

Our aim is to develop an algorithm to automate the process of choosing the airspace configuration in order to best manage the upcoming flight traffic.

Glossary of terms

- **Elementary sectors:** The smallest defined regions that the airspace is divided into.
- **Controlled sector:** A region that is composed of one or more elementary sector that is being controlled by one pair of ATCOs.
- **Dynamic sectorisation:** The process of changing the controlled sectors based on the demands on the airspace.
- **Workload:** Physical and mental strain placed upon the ATCOs.

2. Methods

We use the elementary sectors defined by NATS and apply two mathematical techniques to this problem and compare how they perform against the actual configurations chosen on the day. From the variables of workload/complexity, we use the number of aircraft entering and leaving elementary sectors, and the elementary sectors they have entered from and left to. As an approximation, we say that we want our methods to balance work by minimising the difference in the numbers of aircraft dwelling in each controlled sector. We also want to minimise the coordination workload between controllers. This means we don't want unnecessary duplication of work so we want to clump more elementary sectors together when we can in order to reduce the number of ATCOs that have to communicate with the same aircraft. The coordination effort is defined to be the number of aircraft leaving controlled sectors plus the number of aircraft entering controlled sectors. To simulate realistic conditions, we do not want the number of controlled sectors to exceed the number of available ATCOs. We split time into discrete intervals and process the information in each interval separately. We use the elementary sectors defined by NATS and apply two mathematical techniques to this problem, namely integer programming and greedy algorithms.

Integer Programming

Integer programming is a method of optimisation for problems where some or all variables concerned are integers. Here, we have a discrete number of elementary sectors and hence we have a finite number of possibilities for the airspace configuration. We define the objective function for this problem for each time period to be:

min(total coordination effort + the max difference in # aircraft within controlled sectors).

Using information from NATS, we construct a list of valid configurations. We need to ensure that certain constraints are satisfied, for example, the number of ATCOs required for the configuration cannot exceed the number of ATCOs on duty. The objective value is computed for each configuration at each time period and the configuration with the lowest value for the objective, where the constraints are satisfied, is chosen.

Greedy Algorithms

A greedy algorithm, as the name suggests, selects an option which seems to be the best locally, but it does not necessarily find the best overall solution. A greedy algorithm that frequently impacts everyday life is the Cashier's algorithm for finding the least amount of change. The largest coin value less than the change required is selected repeatedly until there is no remainder left. This is always optimal for the British coin system.

In the area of ATC, there are restrictions as to what configurations we can switch to from the current configuration. For example, we do not want four controlled sectors to be combined into one controlled sector all of a sudden. If these restrictions are imposed as extra conditions, then the optimal solution will need to look further ahead in time to find the best sequence. We can impose these conditions with a greedy algorithm but not the integer programming method since it does not easily allow these types of conditions. One greedy method would be to look at the predicted information only until the next checkpoint, much like the integer programming approach, and choose the best allowable configuration from this information based on our requirements in terms of workload. At each time step, our algorithm allows ATCOs to hand some of their elementary sectors to another controller (who is allowed to be idle at the moment). Once ATCOs are involved in one action, either in giving elementary sectors or receiving elementary sectors, they cannot take another action until the current time period is over. We assume that, if an action has occurred in this time period, then it is more expensive to take actions in the following couple of time periods. This is because we want to discourage changes from occurring too frequently.

3. Results

We apply our methods described in Section 2 to the West End region of the UK. This region is composed of 7 elementary sectors. Initially, we assume that we have perfect future knowledge and know the accurate times that the aircraft enter and exit elementary sectors. We use this information to generate a sequence of configurations. We perform side-to-side comparisons of the results of the two methods against the actual configurations (used by ATCOs on the day). The sequences of configurations are compiled into a video; still image examples are seen in Figures 3 and 4, with the titles of the graphs indicating the time of the day and method used. Each node in the graph represents an elementary sector. Adjacent elementary sectors are joined by a line. Each colour in the figure represents a controlled sector. We see that the two methods can behave similar to each other and/or to the configuration chosen by controllers. The integer programming method (IP) changes very erratically based on whatever is best given the latest traffic prediction. Due to the additional objective function terms and constraints we add to the greedy algorithm, it is a lot more stable. We run the algorithms for a one-day prediction as well as a one-week prediction. We check the agreement between the configurations generated from our methods with the real configuration used. For the one-week prediction, the integer programming has 9% agreement whereas the greedy algorithm is around 15%. If we were only to look at the one-day prediction, then the integer programming sees 17% agreement whereas the greedy algorithm is 34%.

We apply our methods on a section of the UK airspace. The resulting configurations are compared with the actual configurations.

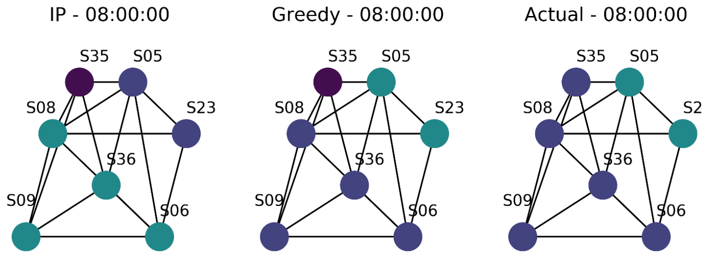

Figure 3 – The two methods can agree with each other but differ from the actual configuration.

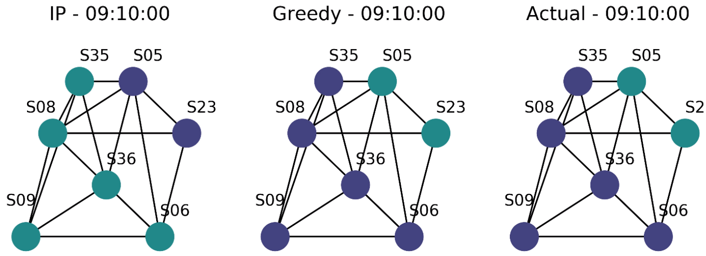

Figure 4 – At times both methods are in agreement with the actual configuration.

Using planned data or sampled data does not seem to affect the overall agreement hugely.

Next we perform the same analysis but with the planned entry and exit times specified on the flight plans. We find that the accuracy of the one-day and one-week predictions are similar for both the integer programming and the greedy algorithm. One possible reason why we do not see much of an improvement using actual data instead of planned data is the time when aircraft get transferred is not the exact time at which they enter a new sector. They typically transfer to the next controller a few minutes before they enter the next sector which enables extra time for the next ATCO to resolve any conflicts that may arise as a result of the new aircraft. Hence, knowing when the aircraft geographically entered into the elementary sector does not necessarily help us to plan any better compared with a guide time. Since many flights are handled by NATS daily, we also look at what happens when we take samples of the data. Are our algorithms sensitive to this decreased size? Does it decrease the agreement of the predictions? We find that, although slightly different configurations are chosen, there is no significant decrease in agreement when we take samples of at least 50%.

We then test the greedy algorithm on a wider area comprised of 19 elementary sectors. The integer programming is too slow to be applied on this region. We see significant slow down with the greedy algorithm even though we initialise with the actual configuration. The agreement of the configurations produced by the greedy algorithm compared with the actual configuration used is 11% and 12% for the actual and planned information, respectively. We obtain similar agreement when a 50% sample of the information is taken. However, if we are only interested in predicting one day ahead, then the agreement is around 20%.

We also make a few adaptations to the objective function to show how the greedy algorithm can make use of the complexity model currently used by NATS. We add a few of the variables from the NATS model to show that this can be done in principle.

4. Discussion, Conclusions & Recommendations

Travel and transport by air has been on the increase over the past few decades, reaching the limit of current practices. Utilising technology and automation is one way to increase capacity. We examined two methods for the dynamic sectorisation problem. The integer programming formulation simplified the problem but it is quite restrictive in the constraints that can be applied as well as being computationally expensive. This lead us to use greedy algorithms instead. While these also suffer from computational cost issues when the problem is scaled up, the cost is lower and implementation is more adaptable to our needs.

Although our predictions do not agree exactly with the configurations used on the day, it is worth noting that for a large proportion of the time, our predictions are 'similar' to the configurations used. The purpose of our methods is not to match the actual configurations but to determine the best configuration given the objective. There is also a subjective element to choosing the configuration: one shift supervisor may choose to sectorise differently to another. Hence a domain expert should be consulted to determine the performance of our methods.

We have shown as a proof of concept that the greedy algorithm can be modified to simulate the current model used at NATS. Further refinements should be made so that the coefficients and variables in the NATS model are fully implemented.

We can develop this work further by using other modelling ideas, machine learning or uncertainty quantification.

There is a lot of scope for further development, for example through other modelling techniques, machine learning, or uncertainty quantification. We have based our study on a list of airspace configurations allowed under the current model of air traffic control which may be subject to change under new initiatives. Therefore one possibility for the future is to investigate other new configurations. We can change the way the greedy algorithm optimises at each time period, for example by using an infectious diseases model where controlled sectors can infect elementary sectors neighbouring it. To reduce the computational costs, we should impose a limit on the time for which the algorithm is searching for a better configuration.

We have also looked at the planned entry and exit times for elementary sectors in the flight plan of the aircraft, as well as the actual times on the day. We have seen that they lead to different configurations being generated by the algorithm. There are often unexpected flight delays which can be due to a multitude of reasons, for example weather or technical issues. This means that there is a degree of uncertainty when using the planned data since we will never have perfect future knowledge. An approach in which the data is updated at each time period (which allows for adjustments in expectations for future times) should be explored.

Can air traffic control ever be fully automated? An interesting concept to explore as we reach the limits of human ATCOs managing the airspace is whether a machine can manage the airspace better. In the decade where we have seen Deepmind's AlphaGo defeat the world champion at Go, first by learning from past games, then through just exploration by itself, can we 'teach' a machine the goals of air traffic control? We have used some heuristic ideas about the workload. However, as numbers of procedures get automated, perhaps many things such as the different directions and speed will not affect the workload if it is down to a machine to process and resolve conflicts.

5. Potential Impacts

Airspace configuration and dynamic sectorisation are very relevant topics amid changes to the air traffic management industry. With new concepts such as Free Route Airspace being implemented, it is possible that our greedy algorithm can be adapted to suit new types of traffic as one possible alternative to the model currently in use by NATS.

Richard Cannon, Commercial Research Lead at NATS, said: *"This short study is a welcome revisiting of the traditional airspace sectorisation problem for air traffic control and NATS. The report demonstrates that a simple and concise treatment of the problem, executed in collaboration with the industry partner, can provide much valued direction and insight to our current research programmes.*

As we investigate the potential of automation in our industry we must first try to represent and model, if possible, the current operational baseline; seeking a strong scientific basis upon which to carefully migrate functions from advisory to authority. This study contributes to our first step towards this goal. It demonstrates clearly that if we can better model the actors in our systems, and their needs, we can take real advantage of - and thus capitalise from - traditional and more modern approaches to mathematical optimisation."