Flight Scheduling Optimiser

H446, 2022

CENTRE NUMBER:

Hollie Lumsden

Candidate number: 5118

CENTRE NAME: WYCOMBE HIGH SCHOOL

Analysis

# Problem Identification

Flight scheduling considers the arrival/departure times and block times for a specific flight. However, many other factors also affect a flight’s schedule, such as the specific airport, connecting flights, turnaround time, flight route, other airlines and destination. These flight schedules are created by software months in advance by the airline so that flights can run smoothly and with minimum disruptions.

However, unexpected events can happen, such as bad weather or a health problem holding a plane up. This can mean that the original scheduling done in advance needs to adapt and change. This is done by a human team of schedulers. This takes time and means the airlines must employ people to do this task, costing them money that could be used elsewhere. The airline will also further lose money for staying on the aircraft apron docked into a gate as it has exceeded the time it was allocated. Therefore, airlines want disruptions to be handled has quickly and as efficiently as possible, which can only be done to a limit with a human team.

Instead, a solution to this problem of dealing with unexpected events could be computational. This is because it will be faster to solve, meaning that scheduling changes can be made with minimum disruptions to other airlines. Also, this will reduce the long-term costs for the airlines, as they wont have to pay wages for people to do the scheduling, and also the costs for disruptions when at the airport.

# Stakeholders

The clients for this project will be airline owners and managers in charge of current flight scheduling, so will respond well to how the project is solving the problem of efficiency and time management. Not many other users with have access or use this software, so the demographic for this project is quite small. The clients will use the software in airports or in airline businesses that help schedule the flights for specific companies.

The stakeholders should represent the opinions of different airlines, and how they would want a computational solution to deal with different situations.

How will stakeholders use this? Explain further.

My stakeholders are Becky, Giacomo and William, who are all graduate airport planners for Atkins. Atkins is a British multinational engineering, design, planning, architectural design, project management and consulting services company. They will be helping me with the areas to include with the flight optimiser, as there are many factors I could include and will help me make choices in that area. They are representative of my demographic as they are people who could handle the software that is being made.

# Why is it suitable to a computational solution?

The solution is suitable to become computational because of the implemented use of algorithms. The solution will use an algorithm to schedule the flights for the airport, considering different variables, which the algorithm can solve faster than a human. It is also suitable for a computational solution because the software will be able to change variables and update the schedule with a higher accuracy and efficiency rate than currently can be done, which will aid to solve the problem of having to wait for flights to be rescheduled.

# Computational methods that will be used

Explain and justify this stuff !!!!!!!!

## Problem Decomposition

This problem can be divided into steps that then can further be divided into sub-problems (see Divide and Conquer). My original plan to decompose the problem is:

1. Take in the flight data
2. Schedule a flight with this data
3. Check for any external variables that have been added
4. Move any flights that are affected

This should mean that the program runs smoothly, and that each variable is checked and there are no clashing flights. It should also mean that the programs runs more efficiently than if the whole problem was solved as one as variables could get missed.

## Divide and Conquer

This method is divided up into three sections; divide, conquer and combine. Divide is when the problem is split into two or more sub-problems that are similar to the original problem but are smaller in size. Conquer is when the sub-problems are solved recursively and then the problems are combined to create a solution to the original problem.

Some of the sub-problems will include:

* Creating a working button for the instructions menu
* Create a function that puts all the flights into a list
* Check the user interface in functional and working correctly
* Create an algorithm that can organise the flights

Using this method means that again, like problem decomposition, everything is checked and that each aspect of the problem is looked at and solved. It also means that the problem is solved faster whilst still being accurate.

## Representational Abstraction

This is when unnecessary detail is removed to provide a simpler version of a complex problem.

Some details that can be removed using representational abstraction are:

* Physical details of the planes (e.g. mass, colour)
* Some external factors (this still needs to be decided which ones will be removed)
* The exact amount of plane docking areas there are on the apron in representation of an airport imitation (e.g. not copying Heathrow Airport exactly)

This is key to making the problem computational, as having to code every detail would be impossible to achieve in the time-frame set. This will also mean that the focus can be on more important

## Information Hiding

Information hiding is when the complexity of the code is hidden behind interfaces or encapsulated in classes.

Ways that information hiding could be used:

* GUI that the user will interact with to use the program (e.g. PyQt, Kibby)
* Using object-oriented programming to utilise encapsulation

# Success Criteria - First Draft

|  |  |
| --- | --- |
| **Criteria** | **How to Evidence** |
| Shows the flight schedule on main window | Screen shot of the main window with schedule on it through use of GUI |
| Visual representation of the airport | Screen shot of the airport visual representation through use of GUI |
| The option to change environmental/disaster variables | Screen shot of the buttons that can change the variables |
| The option to stop simulation | Screen shot of the buttons that can stop/pause the simulator |
| Able to add/delete flights | Screen shot of the buttons that can change the number of flights in the schedule |
| Shows that the software is fast and efficient | Show an efficient use of an algorithm (screen shot the code) |
| Instructions with how to use the software through a pop-up window | Screen shot the options to view the instructions and the instructions button |
| Settings menu were the changing variable and adding/deleting flights option will be stored | Screen shot of the menu and the settings button |
| Simple, usable design | Screen shot of the main window which shows a simple design were everything is clearly labelled and everything is explained |
| All functions are working | Screen shot evidence of testing, with all the inputs and outputs that are made |
| GUI is interactable | Screen shot of the simulation in use |

# Research

## Factors affecting flight scheduling

The airport that the flight is arriving or departing from could affect the flight schedule. This is due to the capacity or the airport and how busy it is, which is caused by:

* Limited slots available for airplanes on the tarmac
* Different window allowances by airport
* How many gates are available and at what times they are available
* How many runways are available
* Is there any construction occurring at the airport

Flights don’t always go from A to Z, as connecting flights can occur. This would mean that the schedule would need to take into account the time it takes for passengers to depart the first flight, walk to the connection gates and go through customs.

Turn-time is the amount of time it takes the plane to

* Turn-time
  + the amount of time it takes the plane to empty out and the crew to [clean and board the next flight](http://www.cntraveler.com/stories/2016-07-11/how-and-how-often-airplane-bathrooms-are-cleaned), as well as whether there are crews that need to be swapped out who might be flying in from another destination.
* Time plane is in the air
  + Air traffic, so might opt for night-flights when going a bust route to avoid traffic
* Marketing factor
  + Is the airport a certain hub for an airline
  + Whether there is a morning or evening departure depending on destination
* Weather
* Maintenance/Re-fuelling
* Unexpected events
* ‘Padding’

A flight delay is when an airline flight takes off and/or lands later than its scheduled time. The [Federal Aviation Administration](https://en.wikipedia.org/wiki/Federal_Aviation_Administration) (FAA) considers a flight to be delayed when it is 15 minutes later than its scheduled time. A cancellation occurs when the airline does not operate the flight at all for a certain reason.

In the [European Union](https://en.wikipedia.org/wiki/European_Union), [Flight Compensation Regulation 261/2004](https://en.wikipedia.org/wiki/Flight_Compensation_Regulation_261/2004) states that flight delays for over three hours, cancellations and denied boarding entitles passengers to a compensation from €250 up to €600 per passenger from the airline.

## Different transportation systems

The spoke-hub distribution paradigm is a form of [transport topology optimization](https://en.wikipedia.org/wiki/Topology_optimization) in which [traffic planners](https://en.wikipedia.org/wiki/Transportation_planning) organize routes as a series of "[spokes](https://en.wikipedia.org/wiki/Spokes)" that connect outlying points to a central "hub". Simple forms of this distribution/connection model compare with [point-to-point transit](https://en.wikipedia.org/wiki/Point-to-point_transit) systems, in which each point has a direct route to every other point. Because of the efficiency (and relative inflexibility) of the model, it requires that the items (or people) being distributed must be routed through a central hub before reaching their destination. Airlines have extended the hub-and-spoke model in various ways. One method is to create additional hubs on a regional basis, and to create major routes between the hubs. This reduces the need to travel long distances between nodes that are close together. Another method is to use [focus cities](https://en-academic.com/dic.nsf/enwiki/1027777) to implement point-to-point service for high traffic routes, bypassing the hub entirely.

Benefits

\* For a network of "n" nodes, only "n"- 1 routes are necessary to connect all nodes; that is, the upper bound is "n"- 1, and the [complexity](https://en-academic.com/dic.nsf/enwiki/28509) is O("n"). This compares favorably to the ("n"("n"- 1))/2  routes, or O("n"2), that would be required to connect each node to every other node in a  point-to-point network.

\* The small number of routes generally leads to more efficient use of transportation resources.  For example, aircraft are more likely to fly at full capacity, and can often fly routes more than  once a day.

\* Complicated operations, such as package sorting and accounting, can be carried out at the hub, rather than at every node.

\* Spokes are simple, and new ones can be created easily.

\* Customers may find the network more intuitive. Scheduling is convenient for them since there  are few routes, with frequent service.

Drawbacks

\* Because model is centralized, day-to- day operations may be relatively inflexible. Changes at the hub, or even in a single route, could  have unexpected consequences throughout the network. It may be difficult or impossible to  handle occasional periods of high demand between two spokes.

\* Route scheduling is complicated for the network operator. Scarce resources must be used  carefully to avoid starving the hub. Careful traffic analysis and precise timing are required to keep the hub operating efficiently.

\* The hub constitutes a bottleneck in the network. Total cargo capacity of the network is limited  by the hub's capacity. Delays at the hub (caused, for example, by bad weather conditions) can  result in delays throughout the network. Delays at a spoke (from mechanical problems with an  airplane, for example) can also affect the network.

\* Cargo must pass through the hub before reaching its destination, requiring longer journeys than direct point-to- point trips. This is often desirable for freight, which can benefit from sorting and consolidating  operations at the hub, but not for time-critical cargo and passengers.

Point-to-point transit is a transportation system in which a plane, bus, or train travels directly to a destination, rather than going through a central [hub](https://en.wikipedia.org/wiki/Transport_hub). This differs from the [spoke-hub distribution paradigm](https://en.wikipedia.org/wiki/Spoke-hub_distribution_paradigm) in which the transportation goes to a central location where passengers change to another train, bus, or plane to reach their destination.

One of the biggest advantages of the point-to-point model is reduced travel time. Passengers could save time without having long layovers between connecting flights, and wouldn’t have to deal with the consequences of a delayed flight.

Point-to-point flights reduce total travel time, primarily by eliminating the intermediate stop, but also by avoiding circuitous routings and increasing aircraft block speed

The point-to-point model also decreases airport dependency. Many hub-and-spoke airlines and their alliance partners operate flights through a limited number of continent hubs, each of which represents a high share of business. The dependency on these airports reduces the number of new routes that can enter the airport, limiting the entrance of new airlines or the addition of new routes. However, the point-to-point model reduces this dependency greatly, as every route from each airport is actively important.

In addition, the elimination of a “dependent airport” reduces the risk an airline faces if a route fails. Point-to-point airlines simply eliminate the struggling route, while hub-and-spoke airlines may be forced to make dozens of accommodations for passengers affected by the route change. The elimination of hub-and-spoke routes might also severely damage the profits of the airport, which depends of the hub-and-spoke airlines for a majority of its business.

## Using SimPy

Why?

# Interview

Needs to be done when email set up with stakeholders for initial meeting

# Features of Proposed Solution

Ideas and then explain limitations

# Stakeholder Feedback

After meeting for interviews

# Requirements

Hardware and Software and then those proposed from the stakeholders

# Success Criteria

The criteria and how to evidence that in the project