

Task 5 - Algorithms

Instructions to run:

Simply run the following command

```
python3 ./main.py
```

verify.py

It's a simple script I created to ensure that the solutions follow constraints, in terms of overbooking

Rough Logic:

The python file main.py contains the code. It uses modified DFS to an extent of 3 iterations account for a maximum of 3 layovers, and tries to discover the arrival airport of the flights that have been cancelled, while also satisfying conditions such as the layover time should be atleast an hour, and few obvious real life constraints.

Designing the Metric for Choosing the Flight:

I needed a metric to decide the rating of a flight, and the parameters are

Since the priorities are given as 1) The travel time should be as close as possible to the destination time 2) As less layovers as possible 3) Better earlier than later..

I didn't have a lot of time on hand to craft a better function so I have directly used

$$F = \text{abs}(\text{time_diff}) * 100 + \text{num_layovers} * 10 - \text{signum}(\text{time_diff})$$

which is virtually just a handy way of saying *I want my most important factor to be the time, then num, and then higher priority to arriving earlier*. This function is not very realistic since ideally, it anyone would prefer to take one layover lesser but reach an insignificant time earlier or later, but I used this for simplicity.

Roadblocks:

Since this is a real life scenario, an objective solution is hard to find. One of the main roadblocks I faced was deciding the order in which flights would be used for accomodation. My thought process was to try to maximize this even if it took more compute since there are many routes that use the same flight and so the flight ranking metric was used to sort through them to try to find the best flight suited for travel.

Metric for Choosing A Better Solution:

Ideally, this is what we want - The more the Passenger Satisfaction, the higher the metric function - As many people re-allocated as possible, the better, so higher the metric function should be - As many people from the same cancelled flight clubbed together into the same flight, the better, so again, the metric function should be higher, but this is not as important of a factor as the other two

For a real life scenario it's not ideal to rank these in a strict order globally, and that's why a mathematical function would be better suited here too!

Developing such a function just from it's general characteristics is hard but here's the one I thought of:

Here's a simple, less computationally heavy method I thought of:

A good way to decide whether moving one passenger to accomodate few others would just be using a passenger satisfaction metric. You want to keep all of them as happy as possible as well as try to keep their variance (a metric of how scattered the data is). A cancelled flight could have a satisfaction of -1, while a rescheduled person's satisfaction would be like

$$(k * (1 - time_diff/cancellation_threshold))^{layovers+m}$$

to make the flight as convenient as possible.

However, I was unable to implement finding the best score in the given time . Ahead is a deeper (but probably pointless) mathematical analysis I did, but it plays no role here.

A suitable metric for passenger satisfaction is given by the scoring method above. We need to normalize this data to 1, we could do that by taking a function which linearly decreases upto a threshold hours to hit 0, and raise that to the power of number of layovers.

$$P_{satisfaction} = \left(\max \left(\left(1 - \frac{|time_diff|}{s} \right), 0 \right) \right)^{layovers}$$

This is normalised between 0 to 1. We would ideally want a better representation of the people's satisfaction by proly taking a statisical mean or so.

Taking up the weight for number of people allocated: Let the fraction of people allocated be x . Thinking about the characteristics of x , we would want it to be really small, killing any effect of the passengers satisfaction if x is really less. Both of the factors are pretty important, but naturally the passenger satisfaction would also decay if the number of allocated passengers is less. So I would attribute an arbitrary weight to both. The function representing the number of people could be a suitable power of x .