# Mphasis

Team 69

15 December 2023

**Abstract**

The aviation industry encounters a significant operational challenge in the effective re-accommodation of passengers faced with disruptions caused by flight delays or cancellations. This paper addresses the intricate task of reallocating and re-accommodating passengers impacted by delayed or cancelled flights. Our approach involves the development of an algorithmic solution to efficiently address this challenge. While a standard approach might involve exhaustive iteration over all conceivable passenger-to-flight assignments, such a backtracking methodology proves impractical due to its exponential time complexity, reaching factorial magnitudes relative to the number of passengers. The impracticality arises from the exponentially growing nature of possibilities, rendering the conventional approach unfeasible for large-scale scenarios. This paper presents an algorithmic based approach to tackle the above problem.

## 1 Introduction:

Upon an initial examination of the provided problem statement and dataset, our focus gravitated towards addressing the challenge of flight reallocation and passenger re-accommodation. Our approach began with the consideration of PNR (Passenger Name Record) ranking, aiming to retain the original class for passengers within the same flight. If the designated class was unavailable, our strategy shifted to exploring alternative flights within the same class. Recognizing the significance of passenger experience, we deemed it unfair to cancel bookings solely due to class unavailability, prompting us to explore the possibility of Downgrading/Upgrading passengers' class, provided an available seat in another class on the same flight. Another dimension of our strategy involved prioritizing the time sensitivity of passengers, seeking to re-allocate them to flights that would expedite their arrival at the destination. We identified key parameters, such as cabin, class, and SSRs (Special Service Requests), and devised algorithms based on two distinct approaches.

- **Maximizing Passenger Comfort:** This approach aims to optimise passenger comfort by prioritising the reallocation of passengers to the same class within the redesigned schema.To put it simply, it attempts to reallocate the maximum number of passengers while considering their priority.
- **Maximizing Seat Reallocation:** The second approach focuses on maximizing the efficient utilization of available seats on flights, giving utmost priority to flights with earlier arrival times. This time-enteric approach places a premium on the expeditious re-accommodation of passengers with high priority.

In pursuit of near-optimal solutions, we have developed algorithms that form the cornerstone of our reallocation schemas. These algorithms incorporate a balance between passenger comfort and efficient seat utilization, contributing to a comprehensive and effective strategy for addressing the complexities inherent in the task at hand.

These Algorithms are based on these 2 factors, first based on priority of passengers and the second one based on maximum seat allocation to minimize net delay through upgrade and downgrade of classes

Further there were 4 basis of reallocating flights to the disrupted passengers. The nomenclature of the approaches is as follows:

1. One–One solution
2. One–M solution

3. M–One solution
4. M–M solution

## 1.1 One-One: Single Impacted Flight - Simple Solution:

- **Situation:** Only one flight in the PNR is impacted (e.g., cancelled, delayed).
- **Non-impacted flights:** Other flights in the PNR itinerary are unaffected and remain as booked.
- **Alternate solution:** The airline offers a single new flight to replace the impacted one. This new flight ideally covers the same route and timings as the original, minimising travel disruption.
- **Benefit:** This option is straightforward and maintains the original booking for the non-impacted flights.
- **Drawback:** Finding a suitable single flight replacement might be challenging, especially for last-minute disruptions.

## 1.2 One-Multi: Single Impacted Flight - Multi-Flight Solution:

- **Situation:** Same as One-One, only one flight in the PNR is impacted.
- **Non-impacted flights:** Remain unchanged as in One-One.
- **Alternate solution:** Instead of one replacement flight, the airline offers a combination of new flights to cover the entire journey. This might involve connecting flights or alternative routes.
- **Benefit:** This option provides more flexibility in finding suitable alternatives, especially for complex travel plans.
- **Drawback:** The travel time and route might be significantly altered compared to the original booking, requiring additional adjustments.

## 1.3 Multi-One: Impacted Flight + Downline Flight - Single Flight Replacement:

- **Situation:** One flight in the PNR is impacted, and it has a connecting flight (downline flight) later in the journey.
- **Non-impacted flights:** Same as One-One and One-Multi, unaffected flights remain unchanged.
- **Alternate solution:** Unlike other scenarios, both the impacted flight and the downline flight are replaced with a single new flight that covers the entire journey.
- **Benefit:** This option simplifies the rebooking process and minimises connection disruptions.
- **Drawback:** Finding a single flight covering the entire journey might be difficult, especially for complex routes or long distances.

## 1.4 Multi-Multi: Impacted Flight + Downline Flight - Multi-Flight Replacement:

- **Situation:** Similar to Multi-One, one flight is impacted along with its downline flight.
- **Non-impacted flights:** Same as all other scenarios, unaffected flights remain unchanged.
- **Alternate solution:** Similar to One-Multi, the airline offers a combination of new flights to replace both the impacted flight and the downline flight. This allows for more flexibility and potential optimization of the travel plan.
- **Benefit:** Provides the most flexibility in finding suitable alternatives for complex travel disruptions.
- **Drawback:** The travel time and route might be significantly altered compared to the original booking, requiring additional adjustments.

## 2 Preprocessing:

The following steps outline the key procedures undertaken to prepare the data for effective passenger re-accommodation:

### 2.1 Selection of Affected PNRs List:

Utilising both the PNR booking & passenger dataset and the flight schedule dataset, we identify and compile a comprehensive list of Passenger Name Records (PNRs) that are adversely impacted due to the disruption of a specific flight.

### 2.2 Identification of Possible Flights for Reaccommodation:

Leveraging the schedule dataset, we filter out flights connecting the same airports affected by the disruption and that arrive within a time frame of 72 hours of actual arrival. Additionally, we identify airports featured in the last segment of itineraries for passengers affected by flight disruptions.

### 2.3 Integration of Schedule Dataset with Inventory Dataset:

Merging the schedule dataset with the inventory dataset, we amalgamate data pertaining to class-wise seat availability for specific schedule IDs and flight dates. This integration facilitates an informed assessment of seat availability for each flight earmarked for re-accommodation.

### 2.4 PNR Ranking:

The priority order of flights is determined by sorting the flight dataset according to the arrival times at the destination airport. This strategic ranking ensures that high-priority passengers are accommodated on flights reaching their destinations earlier, optimizing the overall re-accommodation process. These preprocessing steps collectively lay the foundation for our algorithms, enabling a systematic and data-driven approach to passenger re-accommodation in the face of flight disruptions.

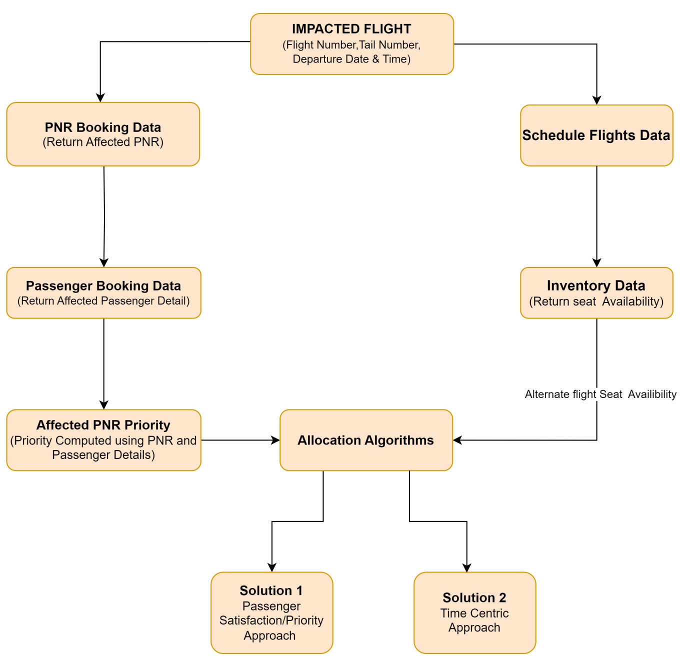


Figure 1: Algorithmic flowchart

# 3 Knapsack Similarity:

*We map our problem to the 0-1 knapsack algorithm with some modification and leverage it into our solution.*

The initial approach to allocate disrupted passengers one by one to available flights bears a time complexity of O(m*n), where 'm' represents the number of passengers, and 'n' denotes the total available seats across all flights. This method, although practical, falls short of providing an optimal solution, introducing the possibility of internal fragments within flights. Recognizing the suboptimal outcome, the intuitive recourse to address this challenge was the application of backtracking. The conceptual alignment of our problem with the well-known 0-1 Knapsack problem prompted the exploration of a distributed allocation approach, drawing parallels to the process of object distribution in a knapsack.

Post the preprocessing phase, two distinct strategies were devised for passenger allocation:

## 3.1 Flight-Priority Allocation:

This strategy entails a systematic allocation process prioritising flights. Disrupted passengers are allocated to available flights, considering individual flight constraints and seat availability. The knapsack algorithm is employed to optimise the allocation process, ensuring the best fit across all possible combinations. This process is iteratively applied for each class until passengers requiring placement in the same class level are accommodated. Subsequently, for remaining passengers, consideration is given to upgraded/downgraded class allocations according to predefined rules. If any passengers remain unallocated, they are deferred for allocation in subsequent flights.

## 3.2 Passenger-Priority Allocation:

In this alternative approach, each cluster of Passenger Name Records (PNRs) belonging to the same class is systematically allocated seats in their original class using the knapsack solution. If the desired class is unavailable in the initial flight, subsequent flights are considered up to a defined threshold of 6 hours. These strategies represent refined and algorithmically sound methodologies for passenger reallocation, leveraging the principles of knapsack problem-solving to optimise the allocation process and enhance the overall efficiency of the re-accommodation task.

## 3.3 Graph Application:

In addressing the passenger-accommodation challenge, a graphical approach has been employed to generate solutions for both the 1-Multi and Multi-Multi scenarios. The 1-Multi approach involves accommodating passengers from a cancelled flight by assigning them to a single alternative flight. To enhance flexibility, passengers are now presented with multiple flight options to reach their destination.

The initial step in the 1-Multi approach involves identifying the affected passengers by extracting information such as the cancelled flight's number, date, and tail number. Subsequently, an iterative process is undertaken through the PNR-Booking file to locate all relevant Passenger Name Records (PNRs).

To streamline the search complexity for generating flight schedules, a decision has been made to allow only one connecting flight. The search process involves filtering flights based on the condition that the departing airport or arriving airport matches that of the cancelled flight. This results in two distinct data frames: one containing flights with the same departure airport and another with flights arriving at the cancelled flight's destination.

Further refinement is applied by eliminating flights departing more than three days(72 hours) after the cancellation of the original flight. This ensures a timely and feasible alternative for the affected passengers.

Subsequently, the Knapsack Approach is employed to efficiently allocate passengers to the available flight options within each data frame, maximising accommodation while adhering to constraints. This methodological approach aims to strike a balance between computational efficiency and optimal passenger accommodation in the event of flight cancellations.

Below is the generated One-Multi solution for the following Cancelled Flight:

- Cancelled Flight Number = 2019

- Cancelled Flight Tail Number = VT-7102
- Departure Date = 04/21/2024

The Flight that was cancelled was From BOM to DEL. Now from 1-Multi solution we had Flights available from BOM to AMD and then from AMD to DEL.

The first file is the Flight Allocated to PNR from BOM to AMD. The second file show the Flight Allocated for PNR from AMD to DEL.

| PNR | InventoryID | New_ClassName | DT | AT | AA |
|---|---|---|---|---|---|
| HYTZ86 | INV-ZZ-8238948 | FC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| NHIY83 | INV-ZZ-8238948 | FC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| HJCQ26 | INV-ZZ-8238948 | BC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| XLES89 | INV-ZZ-8238948 | BC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| IWRM28 | INV-ZZ-8238948 | BC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| ZGRB68 | INV-ZZ-8238948 | BC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| QPXJ84 | INV-ZZ-8238948 | BC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| NJUG57 | INV-ZZ-8238948 | PC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| WXXO13 | INV-ZZ-8238948 | PC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| ZEOJ18 | INV-ZZ-8238948 | PC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| TIKK22 | INV-ZZ-8238948 | PC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| OCCI26 | INV-ZZ-8238948 | EC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| SXCZ87 | INV-ZZ-8238948 | EC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| STKN67 | INV-ZZ-8238948 | EC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| HYFY52 | INV-ZZ-8238948 | EC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |
| NYQT79 | INV-ZZ-8238948 | EC | 25-04-2024 01:10 | 25-04-2024 07:07 | DEL |

Figure 2: Schedule of PNR from Connecting to Destination Airport

| PNR | InventoryID | New_ClassName | DT | AT | AA |
|---|---|---|---|---|---|
| NHIY83 | INV-ZZ-2376230 | FC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| HYTZ86 | INV-ZZ-2376230 | FC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| QPXJ84 | INV-ZZ-2376230 | BC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| ZGRB68 | INV-ZZ-2376230 | BC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| IWRM28 | INV-ZZ-2376230 | BC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| XLES89 | INV-ZZ-2376230 | BC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| HJCQ26 | INV-ZZ-2376230 | BC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| TIKK22 | INV-ZZ-2376230 | PC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| ZEOJ18 | INV-ZZ-2376230 | PC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| WXXO13 | INV-ZZ-2376230 | PC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| NJUG57 | INV-ZZ-2376230 | PC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| NYQT79 | INV-ZZ-2376230 | EC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| HYFY52 | INV-ZZ-2376230 | EC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| STKN67 | INV-ZZ-2376230 | EC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| SXCZ87 | INV-ZZ-2376230 | EC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |
| OCCI26 | INV-ZZ-2376230 | EC | 24-04-2024 13:45 | 24-04-2024 19:31 | AMD |

Figure 3: Schedule of PNR From Source to Connecting Airport

# 4 INTEGER PROGRAMMING MODEL

In our second approach, we propose an Integer Linear Programming (ILP) model to accurately describe the passenger recovery problem. This ILP model aims to efficiently handle the allocation of recovery schemes. The objective function of this ILP model focuses on optimizing specific criteria while considering constraints. The column generation framework, previously employed, is restructured in this approach to split the passenger recovery problem into two distinct problems. The first, an Integer Linear Programming (ILP) master problem, deals with the allocation of recovery schemes. On the other hand, the second problem, known as the subproblem, is dedicated to identifying and selecting the most promising recovery schemes. The objective function in the ILP master problem optimizes the allocation of recovery schemes by balancing various constraints and criteria to ensure efficient passenger recovery within the defined limitations. With the aim of selecting passenger recovery schemes,our objective function is as follows:-

$ObjectiveFunction = minimizing(DelayCost + DowngradingCost + CancellationCost)$

In designing the objective function, we utilize the Mixed-Integer Programming (MIP) model provided by Gurobi.

MIP is a mathematical optimization technique used to solve decision problems where some variables are restricted to be integers. It allows for a combination of discrete (integer) and continuous variables, providing a powerful framework for solving various optimization problems.