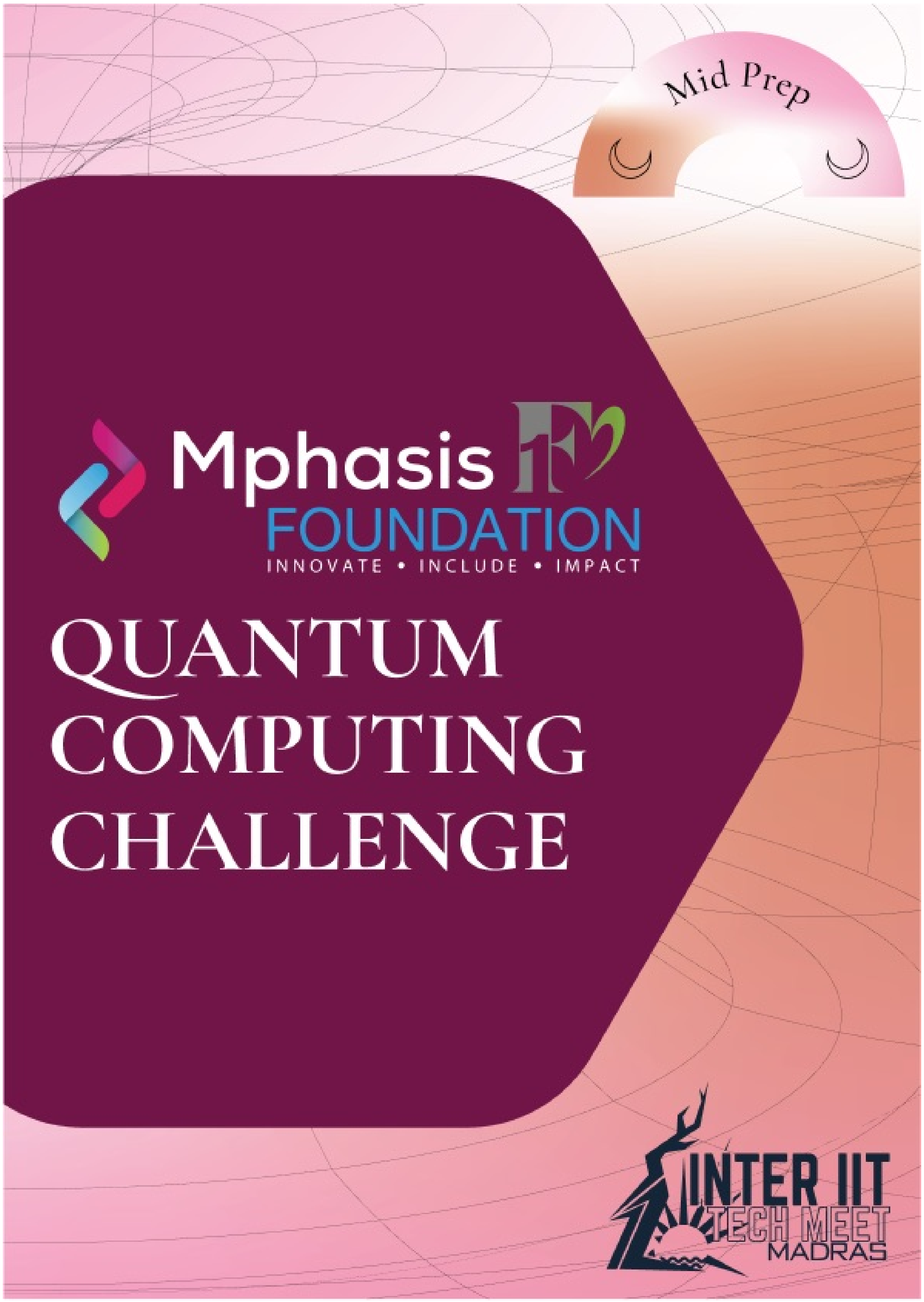
Dynamic Re-accommodation: Automated Tools for Adapting to Airline Schedule Shifts

Team 99

MPhasis Foundation Mid-Prep Quantum Computing Challenge Report



Team 99

Table of Contents-

Abstract

This report presents an innovative automated solution to address the challenges of routine schedule changes in the airline industry. Airlines frequently adjust their flight schedules due to seasonal demands, new routes, daylight savings, flight numbers, and operating frequency modifications. These changes can impact passengers, requiring them to be re-accommodated on alternate flights. Our proposed system focuses on efficiently identifying affected flights and passengers, optimizing alternate flight solutions, ranking them based on various factors, and performing Business Profiling. We have optimized the data sets given to us by the Mphasis Foundation with the parameters and weights assigned to us in the rulebook and have reviewed the potential applications of Quantum Computing in this specific problem.

Problem Statement

Airlines undergo frequent adjustments to their flight schedules for various reasons, such as changes in seasonal demand, the need to identify new routes, adjustments for daylight savings time, modifications to flight numbers, operational frequency, and other considerations. These alterations in itinerary can have significant repercussions for numerous passengers, necessitating the creation of new re-accommodation schedules.

To address these challenges, airlines require a robust solution to assess the implications of the scheduled itinerary modifications for their passengers. This entails an automated system that intelligently determines the most suitable alternative flights for affected passengers, considering availability, convenience, and efficiency. The process involves finding replacement flights and adjusting to accommodate the changes *seamlessly*.

In addition to re-accommodating passengers, airlines must evaluate the broader impact on their flights and the associated logistical elements. This includes assessing the availability of aircraft, adjusting operational frequencies, and managing the overall timing of flights. The goal is to optimize the scheduling changes to minimize passenger disruptions and the airline's operational efficiency.

The complexity of this task underscores the need for a sophisticated system that can analyze various interconnected factors simultaneously. This system should be capable of identifying potential issues, proposing solutions, and making informed decisions about the most appropriate course of action. By automating this process, airlines can streamline their operations, enhance customer satisfaction, and navigate the dynamic and ever-changing landscape of the aviation industry without difficulty.

Project Objectives

- Develop a system that organizes flights according to arrival time and the impact on various supplementary services purchased.

- Assess and prioritize passengers for re-accommodation by considering factors such as being an unaccompanied child, an employee on duty, customer loyalty ratings, and the paid class of service. This involves creating an appropriate assessment framework that dynamically adapts to changing scenarios as given by the user.

- To showcase the system's capability to modify business rules governing passenger and flight rankings dynamically. This ensures flexibility in responding to evolving business needs, providing a solution that can easily adjust to changes in criteria and preferences.

- Implement a ranking system that objectively evaluates and prioritizes different solutions. This involves considering various parameters and criteria and ensuring the ranked solutions align with the airline's goals and priorities.

- Create the files for the flight and passenger re-accommodation solutions, then rank them. Also, to perform the Business Profiling of the entire procedure. Adding an alternative emailing option for affected passengers to choose the flight is a bonus.

- Set clear and actionable guidelines regarding the expiration of specific data or decisions. This ensures that information remains relevant and up-to-date, contributing to the overall efficiency and effectiveness of the system.

Procedure

- ***Platform Choice***: We opted for Google Colab due to its collaborative nature and seamless integration with Google Drive, enabling simultaneous code and folder updates. Files like Inventory_dataset, SCH-ZZ_data, and Passenger_Booking_data from the Mphasis data models (Google Drive) were moved to a new folder accessible in Colab, overcoming the view-only file limitation.

- ***Data Processing***: The above-mentioned Excel files were transformed into DataFrames (inventory_df, flight_df, and passenger_df) for ease of modification. Specific functions were defined to calculate scores, such as calculate_passenger_score for passenger_df.

- ***Passenger Score Calculation***: The function calculates scores based on SSR, cabin, class, downline, and pax. Class-to-cabin mapping is defined in the ruleset, and class scores are assigned alphabetically. SSR data is assumed to be 0 or 1(when this report is written). The total score is the sum of cabin, class, SSR, downline, and pax scores.

· **For passenger_df, we use function(calculate_passenger_score):**

- We can find out the PNR-based score given in the rule set. So, we define a function that calculates the score based on ssr, cabin, class, downline, and pax.
- From the class given, we can find the respective cabin for which the mapping is provided in the ruleset.
- Some classes are not assigned to any cabin, so the remaining classes are assumed to be given to cabin 'Y.'
- The scores for cabins J, K, and Y are given in the ruleset, but for class, the range in which the score lies is given, i.e., 500 to 1000. So, we have assigned a score of 500 to class A, which keeps increasing alphabetically until a score of 1000 is assigned to class Z.
- For SSR, we took it as either 0 or 1.

  The total score is calculated by the sum of cabin+class+ssr+downline+pax scores.

  Passengers for an impacted flight will be assigned to an alternate flight based on the total score obtained for an individual PNR. (A higher score will be assigned first)

- ***Flight Data Processing*: ** For flight_df, an impacted flight is selected using ScheduleID and DepartureDate. Constraints from the ruleset are applied, removing flights with ETD > 72 hours. Total scores for each flight are computed using the calculate_flight_score function, considering factors like arrival_delay, SPF, equipment matching, city pairs, and stopovers.
- **Methodology for flight_df**

1. Suppose we select an impacted flight to sort out the available alternate flights. In that case, we can give the ScheduleID and DepartureDate of the flight as input to extract the impacted flight data (unique for each flight).

2. Now, we have a constraint in the Ruleset. If ETD is >72 hours for the proposed flight, we reject this flight. So, if the Departure time of the proposed flight is > 3 days away from the impacted flight, they are removed.

3. For the above-proposed flights, the total score for the flight is computed. We use the function (calculate_flight_score) for each flight to compute the total score.

4. Arrival_delay is calculated from the difference in the impacted and proposed flight departure dates. According to the ruleset, the scores are assigned based on the arrival_delay.

5. Similarly, SPF is also graded according to the ruleset, i.e., the difference between the arrival time and departure time of a flight.

6. We assumed the Equipment No of a flight as ScheduleID of the same flight. So, the score is calculated if the equipment no of the impacted flight matches the proposed flight according to the ruleset.

Scores based on *citypairs* and *Isstopover* will also be computed. We used function assign(grade) to assign grades based on their total score:

This is used to eliminate low-grade flights from the solution.

- ***Inventory Data Processing*****:** For single impacted flight passengers allocated to a single alternate flight, inventory data for the proposed flight is obtained, and available seats are distributed equally among classes. Passengers are assigned based on their total score, considering class seat availability. Unallocated passengers go to the exception list.
- **Methodology for inventory_df:**

  For single impacted flight passengers allocated to a single alternate flight case:
- In this instance, we select one of the suggested flights and acquire the inventory information for the suggested ScheduleId and DepartureDate (ScID & Ddate are unique for an inventory).
- Oversold (FC_Oversold) indicates the number of first-class tickets that are still available. Additionally, FC_CD provides the total number of seats in a first-class together with the corresponding classes ('F', 'P'). However, the number of seats available in each class is not stated. Therefore, we assumed the seats would be the same for all classes.
- To ensure that no class in the first class has more seats than available, function(distribute_equally) distributes the seats across all classes equally, likewise for the other classes.
- Based on their overall score, passengers will be assigned. If seats are still available in the requested class, passengers will be assigned to that class. If not, they may be placed in other classes, as specified in the Ruleset, if there are open seats.
- Passengers who are not assigned will be added to an exception list.

For single impacted flight passengers allocated to multiple alternate flight cases:

- This method is effective only when multiple flights are being considered. The underlying logic remains the same; however, in cases where passengers remain in the exception list after being assigned to a single aircraft, these passengers are distributed among the other proposed flights.

- In the case of several impacted flights, passengers are assigned to a single alternate flight.

This is done by inputting the Schedule ID and Departure Date of the affected flights. In this case, passenger data is aggregated based on the number of affected flights, while everything else stays the same.

- In the case of multiple impacted flights, passengers allocated to multiple alternate flights case can be done using the same logic as stated above.

- ***Handling Multiple Scenarios*****:** The process is adapted for cases where passengers are allocated to multiple alternate or impacted flights.
- ***Solution Ranking*****:** Solutions are ranked based on low scores for a high exception list, lower mean arrival delay, and other weights specified in the ruleset.

  The solution is ranked based on the following factors:
  - Low score for high exception list.
  - More score for less mean arrival delay.
  - The remaining weights are given as mentioned in the Ruleset.

Results-

We only conducted the 1-1 case. We have to provide input to choose a flight that has been affected.

- The selected_DEP_KEY provides a distinct identifier that pertains to the affected flight data. This is the sole input required to determine an affected flight.
- The PNR numbers are taken from the PNRB-ZZ-20231208_062017.csv file based on the DEP_KEY that matches the selected_DEP_KEY. Thus, we obtain the passenger data at the PNR level for those affected.
- The passenger-level data, including names and phone numbers, is extracted from the Passenger_Booking_data file using the PNR-level data.
- The PNR and passenger-level data columns have been combined to obtain the necessary data in a unified data frame. Originally, we possessed coach data for passengers, categorised as F, J, Q, and so forth. Currently, we only possess class-specific data, such as First class and Business class. Each passenger will be assigned coaches randomly, depending on their class. If a passenger has a first-class seat, they will be assigned an F or P coach. The same applies to passengers in other classes.
- By utilising this consolidated data frame, scores are computed. For example, scores such as Class, PAX, and Downline scores.
- Using selected_DEP_KEY, we can get the Available_flight data, which has ETD < 72 hours and (DepartureAirport and ArrivalAirport) same as the impacted flight.
- These flights are scored accordingly (Arrival Time delay score, SPF score, etc).

By utilizing the Available_flight data, we can extract the inventory level data from the INV-ZZ-20231208_041852.csv file by using the Available_flight data.

- As the distribution of oversold seats among coaches is unknown, we allocate them evenly, as done with the prior data set. (It is located in the preceding process)
- Additionally, we own both passenger data and inventory data. Passengers are assigned seats based on the availability of seats in their coach type. If seats are unavailable in their coach type, they are checked for seat availability in their suggested class. They are placed on the exception list if seats are unavailable in either their coach type or proposed class. The allotted passengers are derived by excluding those listed in the exemption list from the initial affected passengers. (The same methodology was applied to the prior data set

as well).

- Currently, the solutions are prioritized according to the total score obtained by combining the exception list score and the average arrival time delay score. The score for the exception list is calculated as 100 minus the number of passengers on the exception list.
- The mean arrival time delay score is calculated by subtracting the arrival time delay from 100. The passengers, exception list, available inventory, and solution ranking of the available flight data are stored in the **Solutions.xlsx** file.
- Example-

ScheduleID	ierC	tNum	craftTy	aftTailNur	DepartureAirport	ArrivalAirport	DepartureTime	ivalTi	rtD	dDa	tatu	ur	Dat	val_d	ture	Arrival Delay score	SPF score	y pairs sc	uipment sc	Total Score	Grade	Exception	Rank
SCH-ZZ-5716505	ZZ	4335	Airbus	VT-2116	HYD	AMD	2023-12-16 14:19:00	###	04	06/1	Sch	['	04/	48	48	30	40	50	0	120	D	72	70
SCH-ZZ-6469147	ZZ	2299	Airbus	VT-6642	HYD	AMD	2023-12-16 01:59:00	###	04	05/3	Sch	['	04/	48	48	30	40	60	0	130	D	71	70
SCH-ZZ-5716505	ZZ	4335	Airbus	VT-2116	HYD	AMD	2023-12-16 14:19:00	###	04	06/1	Sch	['	04/	24	24	40	50	50	0	140	D	78	60
SCH-ZZ-6469147	ZZ	2299	Airbus	VT-6642	HYD	AMD	2023-12-16 01:59:00	###	04	05/3	Sch	['	04/	24	24	40	50	60	0	150	C	70	60

ailableInv	TotalInven	bokedInve	C_Oversol	ailableInv	FC_CD	BC_CD	PC_CD	EC_CD	FC_CD_a	BC_CD_a	PC_CD_a	EC_CD_a
29	94	55	25	39	{'F': 12, 'P'	{'C': 8, 'J':	{'Q': 12, 'R	{'Y': 2, 'A':	{'F': 6, 'P':	{'C': 4, 'J':	{'Q': 3, 'R':	{'Y': 2, 'A': 2, 'B': 1, 'D': 1, 'E': 1, 'G': 2, 'I': 1, 'K': 2, 'L': 1, 'N': 1, 'O': 1, 'U': 1, 'V': 1, 'W': 1, 'X': 1}
32	89	49	31	40	{'F': 12, 'P'	{'C': 6, 'J':	{'Q': 10, 'R	{'Y': 0, 'A':	{'F': 7, 'P':	{'C': 5, 'J':	{'Q': 4, 'R':	{'Y': 3, 'A': 2, 'B': 1, 'D': 1, 'E': 1, 'G': 2, 'I': 1, 'K': 2, 'L': 1, 'N': 1, 'O': 1, 'U': 2, 'V': 1, 'W': 1, 'X': 1}
30	98	58	30	40	{'F': 13, 'P'	{'C': 8, 'J':	{'Q': 12, 'R	{'Y': 3, 'A':	{'F': 7, 'P':	{'C': 5, 'J':	{'Q': 4, 'R':	{'Y': 2, 'A': 2, 'B': 1, 'D': 1, 'E': 1, 'G': 2, 'I': 1, 'K': 2, 'L': 1, 'N': 1, 'O': 1, 'U': 2, 'V': 1, 'W': 1, 'X': 1}
34	102	58	30	44	{'F': 13, 'P'	{'C': 6, 'J':	{'Q': 12, 'R	{'Y': 3, 'A':	{'F': 7, 'P':	{'C': 6, 'J':	{'Q': 4, 'R':	{'Y': 2, 'A': 2, 'B': 1, 'D': 1, 'E': 1, 'G': 2, 'I': 1, 'K': 2, 'L': 1, 'N': 1, 'O': 1, 'U': 2, 'V': 1, 'W': 1, 'X': 1}

Available inventory as shown above, FC_CD shows the total seats in First class and its coaches. FC_CD_a shows the available seats computed from Oversold data. Passengers are allocated using these available seats only. It is also done similarly for other classes.

RECLOC	ATION_DT	STOMER_	AST_NAM	RST_NAM	ATIONALIT	DOB	TACT_PH_	NTACT_EM	DOC_TYPE	DOC_ID	L_NAN	RIG_C	EST_C	EP_D	P_DTI	_D	_D	_D	isseng	Cabin	abin_sco	lass_scor	wnline_sc	AX_score	SR_score	otal Scor	Proposed_Class
VDJD92	3/13/2024	GPN9737	Baral	Krish	IN	8/6/1	(592) 440-	krishbaral	Passport	797042483		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	F	600	2700	100	100	0	3500	F
VDJD92	3/13/2024	NGG2246	Brady	William	CA	9/18/	(346) 230-	williambra	Driving Lic	5.5014E+11	NRPS	HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	F	600	2700	100	100	0	3500	F
ATNB22	3/13/2024	GXQ2050	Sampath	Sumer	IN	5/18/	(345) 440-	sumersam	Govt-Id	8343859372		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	F	600	2700	100	50	0	3450	F
UEOO75	3/13/2024	RYK7104	Montoya	Robert	CA	4/22/	(759) 860-	robertmo	Passport	147406370		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	Z	1000	2000	100	200	0	3300	Z
UEOO75	3/13/2024	RFE2786	Jones	Hannah	CA	1/20/	(224) 499-	hannahjor	Govt-Id	9881235118		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	Z	1000	2000	100	200	0	3300	Z
UEOO75	3/13/2024	FQF4263	White	Danielle	US	6/5/1	(859) 634-	daniellewl	Govt-Id	7838639746		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	Z	1000	2000	100	200	0	3300	Z
QQVP20	3/13/2024	BMT7482	Garcia	James	CA	3/30/	(950) 104-	jamesgarc	Driving Lic	2.3284E+11	NRPS	HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	Z	1000	2000	100	150	0	3250	Z
ULEJ53	3/13/2024	OFI7100	Wagner	Amber	CA	5/6/1	(629) 833-	amberwag	Govt-Id	6275679964		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	T	880	1500	100	450	0	2930	T
ULEJ53	3/13/2024	CVW7248	Adkins	Lorraine	CA	7/22/	(508) 246-	lorrainead	Driving Lic	8.1029E+11	NRPS	HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	T	880	1500	100	450	0	2930	T
QQVP20	3/13/2024	WCV3401	Knapp	Sandra	US	1/13/	(500) 496-	sandrakna	Passport	191529664	NRPS	HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	J	680	2000	100	150	0	2930	J
QQVP20	3/13/2024	MNQ3180	Jordan	Rebecca	US	8/4/1	(613) 335-	rebeccajo	Passport	749099825		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	J	680	2000	100	150	0	2930	J
ULEJ53	3/13/2024	WJT8568	Ghose	Fateh	IN	4/27/	(197) 546-	fatehghos	Driving Lic	2.2964E+11		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	T	880	1500	100	450	0	2930	T
ULEJ53	3/13/2024	HDN9246	Clark	Judy	US	10/4/	(625) 390-	judyclark5	Passport	226096094		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	S	860	1500	100	450	0	2910	F
ULEJ53	3/13/2024	KCZ9478	Watts	Colton	US	8/25/	(250) 391-	coltonwat	Govt-Id	8380502608		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	S	860	1500	100	450	0	2910	F
ULEJ53	3/13/2024	MFV6280	Cantrell	Heather	CA	4/11/	(619) 636-	heatherca	Passport	16704132	NRPS	HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	S	860	1500	100	450	0	2910	S
ULEJ53	3/13/2024	ZHJ6710	Martinez	Elizabeth	CA	5/27/	(557) 438-	elizabethr	Passport	486834314		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	Q	820	1500	100	450	0	2870	Q
UEOO75	3/13/2024	OLE4934	Wali	Ehsaan	IN	12/26	(826) 978-	ehsaanwa	Driving Lic	6.9708E+11		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	C	540	2000	100	200	0	2840	C
FZHS34	3/13/2024	HYU4702	Mandal	Elakshi	IN	5/9/1	(714) 149-	elakshima	Passport	583059926		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	J	680	2000	100	50	0	2830	C
PPOR96	3/13/2024	LNB4727	Vance	Nicole	CA	8/6/1	(990) 339-	nicolevan	Driving Lic	4.8562E+11		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	J	680	2000	100	50	0	2830	O
NKGU25	3/13/2024	BZL3869	Stevenson	Jessica	US	12/14	(236) 434-	jessicaste	Passport	1.8751E+11		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	J	680	2000	100	50	0	2830	I
ULEJ53	3/13/2024	DIG7045	Joshi	Zoya	IN	12/24	(798) 706-	zoyajoshi	Govt-Id	3686860899		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	M	740	1500	100	450	0	2790	M
ULEJ53	3/13/2024	IYI3462	Ravi	Indrans	IN	4/13/	(632) 230-	indransra	Passport	473271782		HYD	AMD	4/28/	4/28/	4/2	4/2	4/.	Passe	M	740	1500	100	450	0	2790	M

The allocated passenger's data contains the score of passengers, their initial class before and allocated class (proposed class) after, etc. The exception list is prepared similarly.

Passenger Emailing Facility

We have created an additional facility to email the passengers about the re-scheduled flight using the Gmail API. The automated email system employed by the team utilises the Google Cloud Platform and integrates the Gmail API. This sophisticated system incorporates HTML templates for various elements such as plane tickets, email formats, and other appropriate send-out features. The solution file is downloaded to a server (facilitated by a laptop). Subsequently, the system dispatches updates to all passengers via email, keeping everyone informed of updated information.

Business Profiling

Business Concept: The application is designed to address the challenges associated with planned schedule changes in the airline industry. The primary objective is to identify optimal and best alternate flight solutions for all impacted passengers, ensuring compliance with a set of provided rule sets. It goes beyond mere rebooking by using advanced optimization algorithms and ranking mechanisms to enhance the overall passenger experience. Overall, our application is positioned to revolutionize the way airlines handle planned schedule changes, providing a comprehensive and efficient solution that prioritizes passenger satisfaction while adhering to airline-specific rule sets and policies.

Target Audience: An ideal target audience for this application consists of various commercial airliners. This will satisfy the need of airliners to accommodate passengers in the case of rescheduling, which is becoming necessary in the modern, volatile world where everyday uncertainties can cause trouble for an airline seeking positive customer feedback.

Market analysis and overview: The industry is scattered with various other airline reservation software like TravelPerk, myCWT, Deltamatic (used by American airliner Delta Airlines), Amadeus (operated by American budget airline Southwest), SAP Concur, and so on. While these reservation systems went along without problems for the most part since its establishment, they happened to backfire heavily during the pandemic. Several passengers were hit with rescheduling conflicts, which went wrong and left disgruntled customers behind. This leaves a permanent scar on the face of the airline's reputation- a lost customer would never come back to the same airline, considering the industry's competitiveness. This is where our application can go into place- an accurate application that leaves the airline satisfied with satisfied customers and assures the return of the customer if they were to fly again.

Marketing and Sales Strategy: The idea would be to focus on local airlines first, build a strong reputation then focus on the larger ones. This is because these airlines require more customer satisfaction to remove possible fliers from the giant airliners. Our product should be able to satisfy and convince them to buy it- the catch being, that there is a limited amount of airlines- so an initial loss (selling the product at below-margin rates) to rake in more airline customers.

Potential Applications of Quantum Computing in this problem with a study on its Effectiveness

Theory-

In classical computing, a bit represents information as either $|0\rangle$ or $|1\rangle$ in Dirac notation, corresponding to the column vectors (1 0)’ and (0 1)’, respectively. In quantum computing, a qubit is the fundamental unit of information expressed as $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$ in superposition. Here, $|\psi\rangle$ is the quantum state, a and b are complex numbers, and the normalization condition is maintained $a^2 + b^2 = 1$. When measuring $|\psi\rangle$, the probability of observing 0 is a^2, and the probability of observing 1 is b^2.

1. **Score Calculation using Quantum Computing-** We first attempted to calculate the total scores (passenger score and total flight score assuming these weights and constraints) using Quantum Computing. The technique used was Quantum Fourier Transformation, which has proven to be the simplest way to perform arithmetic on Quantum Computers using the least number of qubits. We performed a sample addition and a multiplication operation, and the code can be viewed here as Arithmetic.ipynb .
2. Simulating the multiplication of large numbers on quantum computers demands a substantial number of qubits, and the simulation runtime surpasses one minute. Even with the IBM Eagle providing a maximum of 127 qubits and scalable technologies struggling to exceed 5000 qubits, executing intricate arithmetic operations like calculating the total scores in the flight re-routing problem on quantum computers appears impractical. In such cases, classical calculations are preferred due to current limitations in quantum

computing capabilities.

3. Quantum Machine Learning- Quantum Approximation Optimization Algorithm and Quantum Annealing

To offer flexibility in business rules, instead of the classical optimisation techniques, we could use Quantum Machine Learning (QML), which uses the quantum properties of particles to implement ML and optimisation algorithms and shows higher success with more constraints.

The first step is formulating the algorithm, considering all the constraints and weights. If we don't have pre-defined data (for testing flexibility in business rules), we could transform this formulation into a QUBO (Quadratic Unconstrained Binary Optimization) or an Ising Hamiltonian, depending on the type of problem. This technique is used for combinatorial optimization problems.

Conversion to QUBO

The following code below converts the problem into QUBO form using the QUBOvert package in Python. The QUBO form can then be optimised based on the company's requirements using techniques like Quantum Annealing, with the use of simulated annealers like PyQubo (D-Wave Quantum Annealer)

QUBOvert.ipynb

The QUBO is then converted to the Hamiltonian energy form. After obtaining the Hamiltonian, the next step is optimisation to find the lowest possible energy, representing the optimised function analogous to the ground state in quantum machine learning. Traditional Quantum Approximate Optimisation Algorithm (QAOA) might not be ideal for this problem due to its classical-hybrid nature and challenges in ranking alternative solutions. Instead, Quantum Annealing (QA) offers a promising approach. Similarly, we could also convert the problem to Ising Hamiltonian Formulation using *dimod* or *PyQubo* packages.

Quantum Annealing involves using a specialized quantum annealer, such as the D-Wave

Annealer, to find the energies of the Hamiltonian. Multiple rounds of annealing can be executed to discover the lowest energy state, considering all parameters as the optimized cost function. QA has an advantage in increasing the probability of reaching a global minimum, providing more accurate results. D-Wave provides the PyQubo package, facilitating the direct conversion of the problem into the Ising Hamiltonian and optimization. Additional samples can be generated to identify the second-lowest energy state, representing the second-best alternative, enabling the ranking of solutions.

Furthermore, QA can be leveraged to determine the ranking by addressing the requirement to rank alternative flight routes and present choices to customers. The solution with the second-lowest energy becomes the second-best, the third-lowest for the third-best, and so on. This ranked information can be used to offer customers alternatives in a prioritized manner. This technique can only be used if we wish to test the flexibility of any algorithm for different weights and cannot be implemented for pre-defined data, as given in this problem statement.

Conclusions

Thus, this elaborates our solution for the flight-rescheduling problem, and the solution files, code, and other necessary files have been zipped. We have used classical optimization to work on our solution based on certain assumptions. The applications of Quantum Computing have been mentioned, and business profiling and the emailing facility have been included in the report file.