Mphasis

INTER IIT TECH MEET MADRAS

Quantum Computing Challenge

Team 80

Problem Statement

Objectives

01 Minimizing unaccommodated passengers

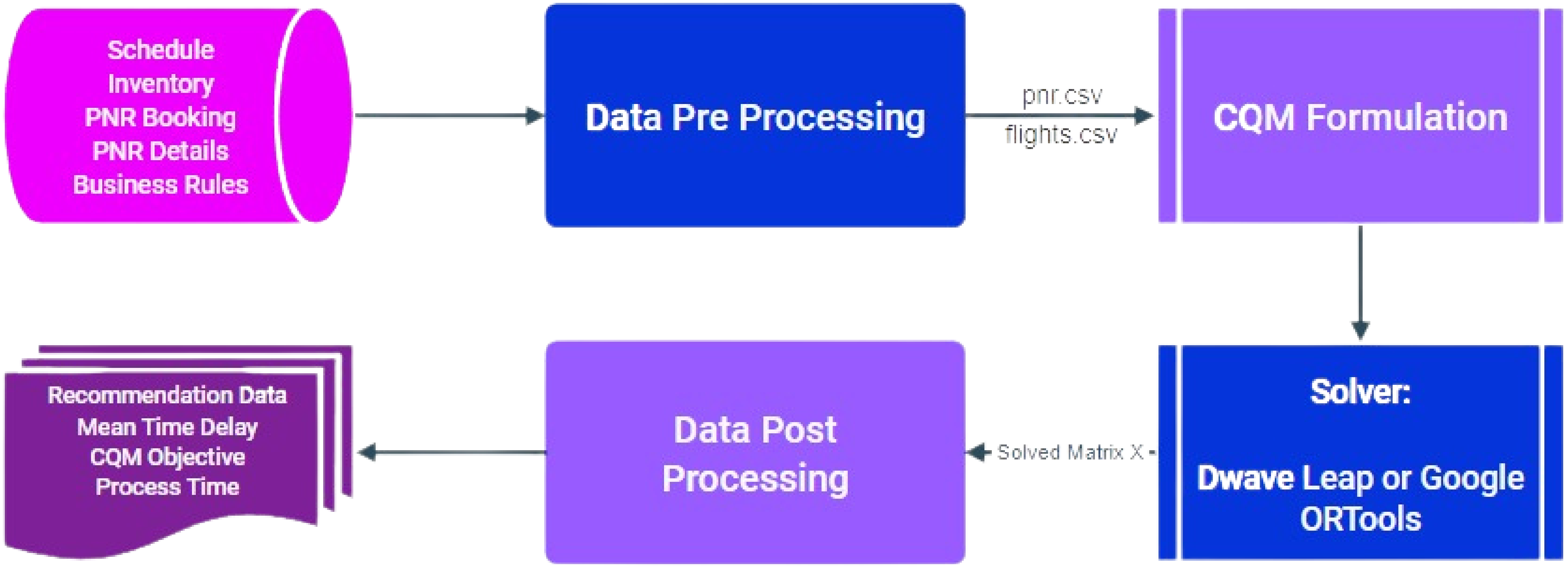
02 Minimizing Mean Arrival Time Delay

03 Prioritizing higher value passengers for better route alternatives

04 Prioritizing alternatives with fewer stops

05 Prioritizing allocation of majority of passengers to the same flight

Solution Pipeline



Data Preprocessing

Passengers

Calculating the SCORE for each passenger on the basis of the business rules of the airline.

Allocating UID of the original flight option to the passenger

```
IDX,PNR,KEY,CNT,SCORE,FK,UID
0,DRGS80,ZZ20240403BLRCCU2504,3,8400,0,480
1,YEZQ47,ZZ20240403BLRCCU2504,4,8450,0,480
2,JDDM40,ZZ20240403BLRCCU2504,2,6100,0,480
3,AVFF16,ZZ20240403BLRCCU2504,1,3550,0,480
4,DBLE79,ZZ20240403BLRCCU2504,2,3350,0,480
5,HBRA44,ZZ20240403BLRCCU2504,4,12450,0,480
6,EOWG16,ZZ20240403BLRCCU2504,2,5350,0,480
7,AVDY52,ZZ20240403BLRCCU2504,4,6200,0,480
8,XKCU70,ZZ20240403BLRCCU2504,1,3050,0,480
9,TOGO39,ZZ20240403BLRCCU2504,4,7950,0,480
10,POTO89,ZZ20240403BLRCCU2504,2,1800,0,480
11,XJFX88,ZZ20240403BLRCCU2504,1,3800,0,480
12,QUEI19,ZZ20240403BLRCCU2504,1,1850,0,481
```

Flights

Creating 4 Flight Options from each flight
Converting departure and arrival time to epochs

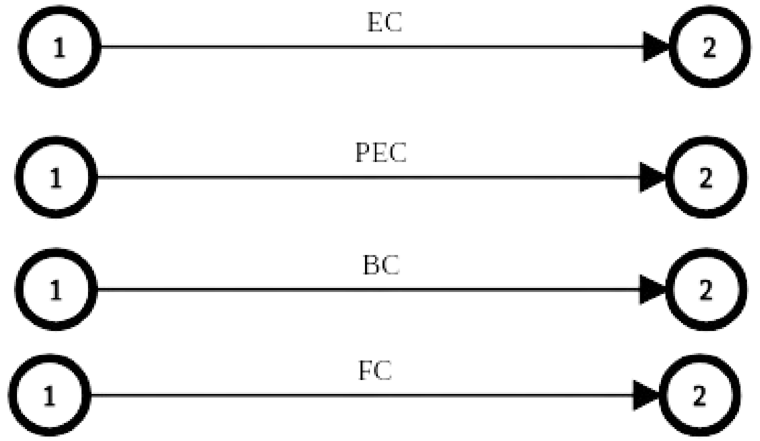
```
UID,KEY,DEP,ARR,CLASS,CAPACITY,DEP_TIME,ARR_TIME
0,ZZ20240505AMDHYD2223,AMD,HYD,FC,42,1714906320,1714937400
1,ZZ20240505AMDHYD2223,AMD,HYD,BC,85,1714906320,1714937400
2,ZZ20240505AMDHYD2223,AMD,HYD,PC,127,1714906320,1714937400
3,ZZ20240505AMDHYD2223,AMD,HYD,EC,170,1714906320,1714937400
4,ZZ20240506AMDHYD2223,AMD,HYD,FC,42,1714992720,1715023800
5,ZZ20240506AMDHYD2223,AMD,HYD,BC,85,1714992720,1715023800
6,ZZ20240506AMDHYD2223,AMD,HYD,PC,127,1714992720,1715023800
7,ZZ20240506AMDHYD2223,AMD,HYD,EC,170,1714992720,1715023800
8,ZZ20240507AMDHYD2223,AMD,HYD,FC,42,1715079120,1715110200
9,ZZ20240507AMDHYD2223,AMD,HYD,BC,85,1715079120,1715110200
10,ZZ20240507AMDHYD2223,AMD,HYD,PC,127,1715079120,1715110200
11,ZZ20240507AMDHYD2223,AMD,HYD,EC,170,1715079120,1715110200
12,ZZ20240508AMDHYD2223,AMD,HYD,FC,42,1715165520,1715196600
13,ZZ20240508AMDHYD2223,AMD,HYD,BC,85,1715165520,1715196600
14,ZZ20240508AMDHYD2223,AMD,HYD,PC,127,1715165520,1715196600
15,ZZ20240508AMDHYD2223,AMD,HYD,EC,170,1715165520,1715196600
16,ZZ20240512AMDHYD2223,AMD,HYD,FC,42,1715511120,1715542200
17,ZZ20240512AMDHYD2223,AMD,HYD,BC,85,1715511120,1715542200
18,ZZ20240512AMDHYD2223,AMD,HYD,PC,127,1715511120,1715542200
```

Methodology

Definitions

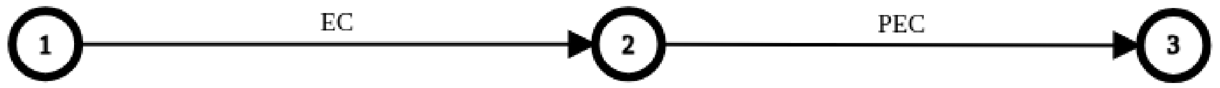
Flight

Flight Options



A flight option is a flight decomposed into its class ratings

Route



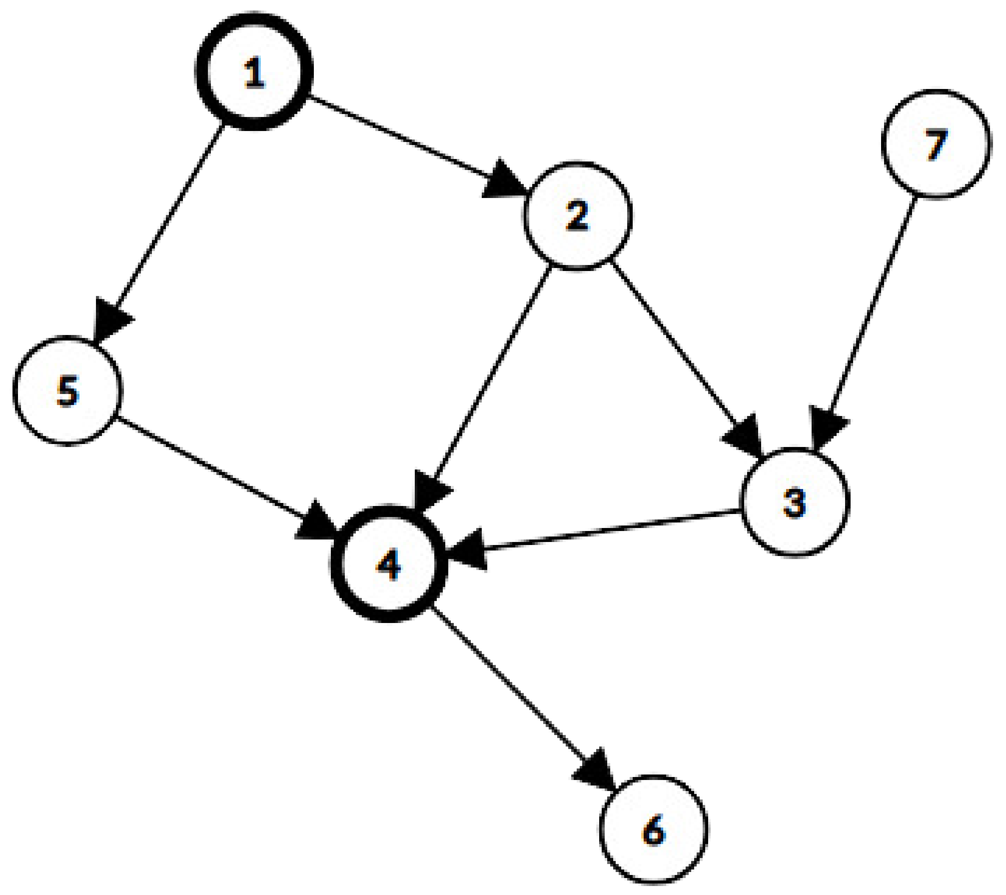
A route is a sequence of flight options

Viable Alternative Routes

Start a DFS Search at the source node with a preset radius R

Ensure that all connecting flights lie in an acceptable time range

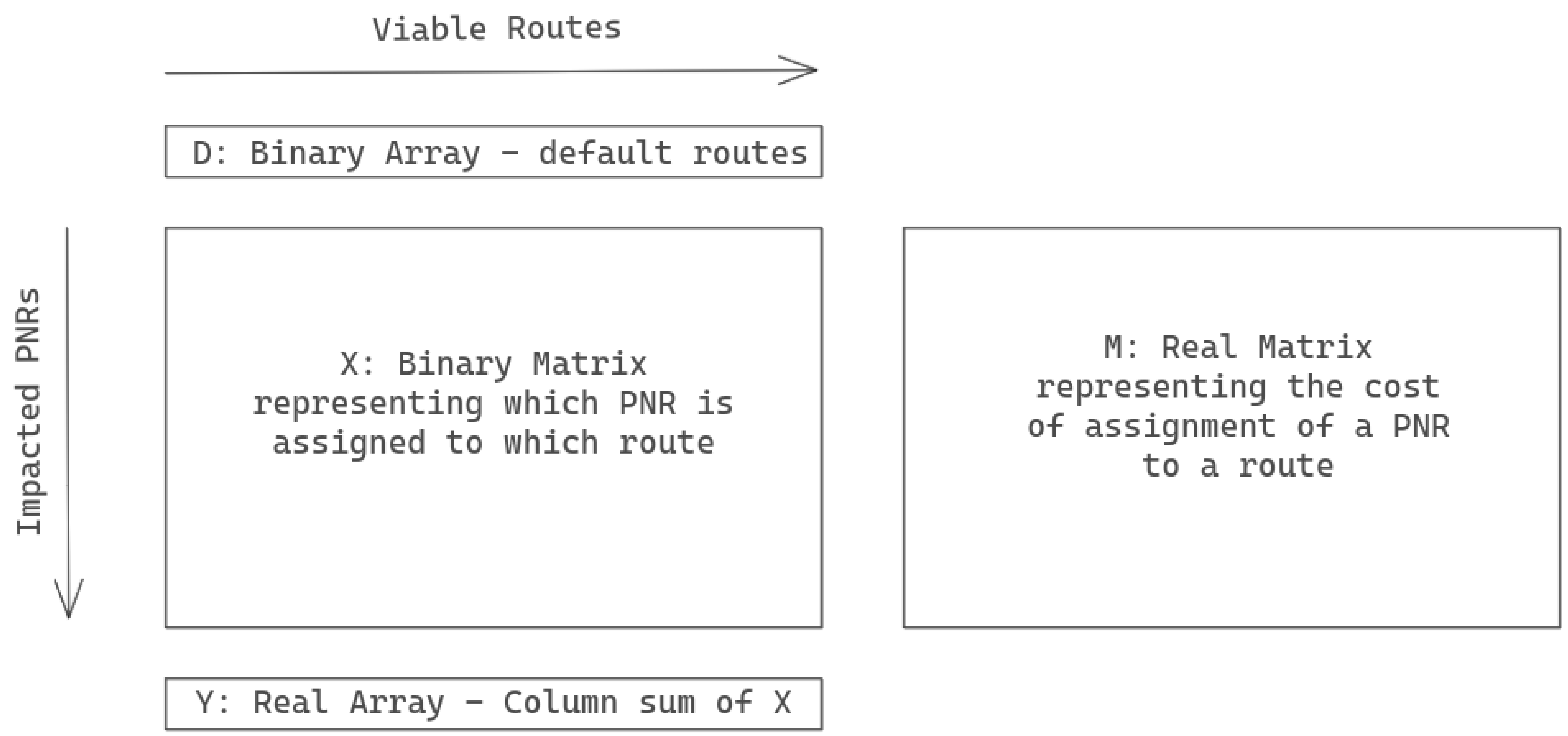
The current defaults are R = 2, and a minimum layover of 1 hour, and a maximum layover of 5 hours



CQM Formulation

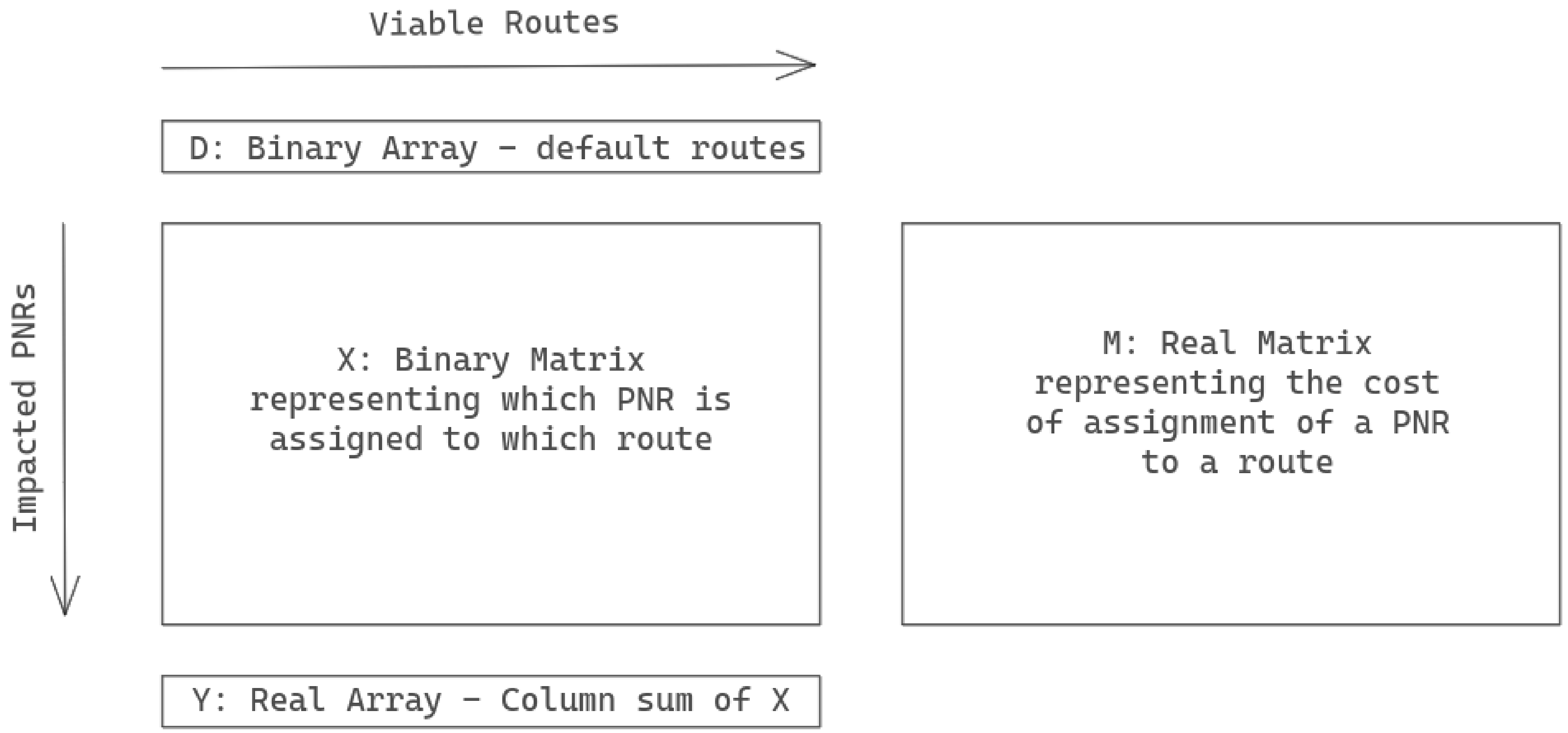
Formulation of CQM

- We reduce our problem to a Constrained Quadratic Model (CQM).
- For each impacted flight 4 data structures are created: 2 matrices M and X and 2 arrays D and Y.



Formulation of CQM

- Are created for each impacted flight.
- Each row corresponds to an impacted PNR of the flight.
- Each column corresponds to a viable replacement route.
- An extra “dummy” column is added for the case of the PNR not being re-accommodated.



Matrix X

- Binary decision matrix, with cell (i, j) = 1 if PNR i is assigned route j. If this cell is in the last column and is 1, that corresponds to the case of the PNR not being re-accommodated.
- X encodes the final solution.

$$
\begin{array}{c}
\begin{array}{cccccccc} \text{Route 1} & R2 & R3 & R4 & R5 & R6 & R7 & \text{Dummy} \end{array} \\
\begin{pmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{pmatrix}
\begin{array}{l} \text{PNR 1} \\ \text{PNR 2} \\ \text{PNR 3} \\ \text{PNR 4} \\ \text{PNR 5} \end{array}
\end{array}
$$

Constraints

Constraint 1: One PNR is to be accommodated in one route only.

$$\sum_j X[k][i][j] = 1 \ \forall \ k, i$$

```
for all i, j, k:
    for all f in R(j):
        Z[f] += (X[k][i][j] * CNT[i])
```

Constraint 2: The number of passengers accommodated on a flight option f must not exceed its capacity to carry C_f.

$$Z[f] \leq C_f \ \forall \ f \in F$$

X for Impacted Flight 1

$$\begin{array}{c} \begin{array}{cccccccc} \text{Route 1} & R2 & R3 & R4 & R5 & R6 & R7 & \text{Dummy} \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{array}{l} \text{PNR 1} \\ \text{PNR 2} \\ \text{PNR 3} \\ \text{PNR 4} \\ \text{PNR 5} \end{array} \end{array}$$

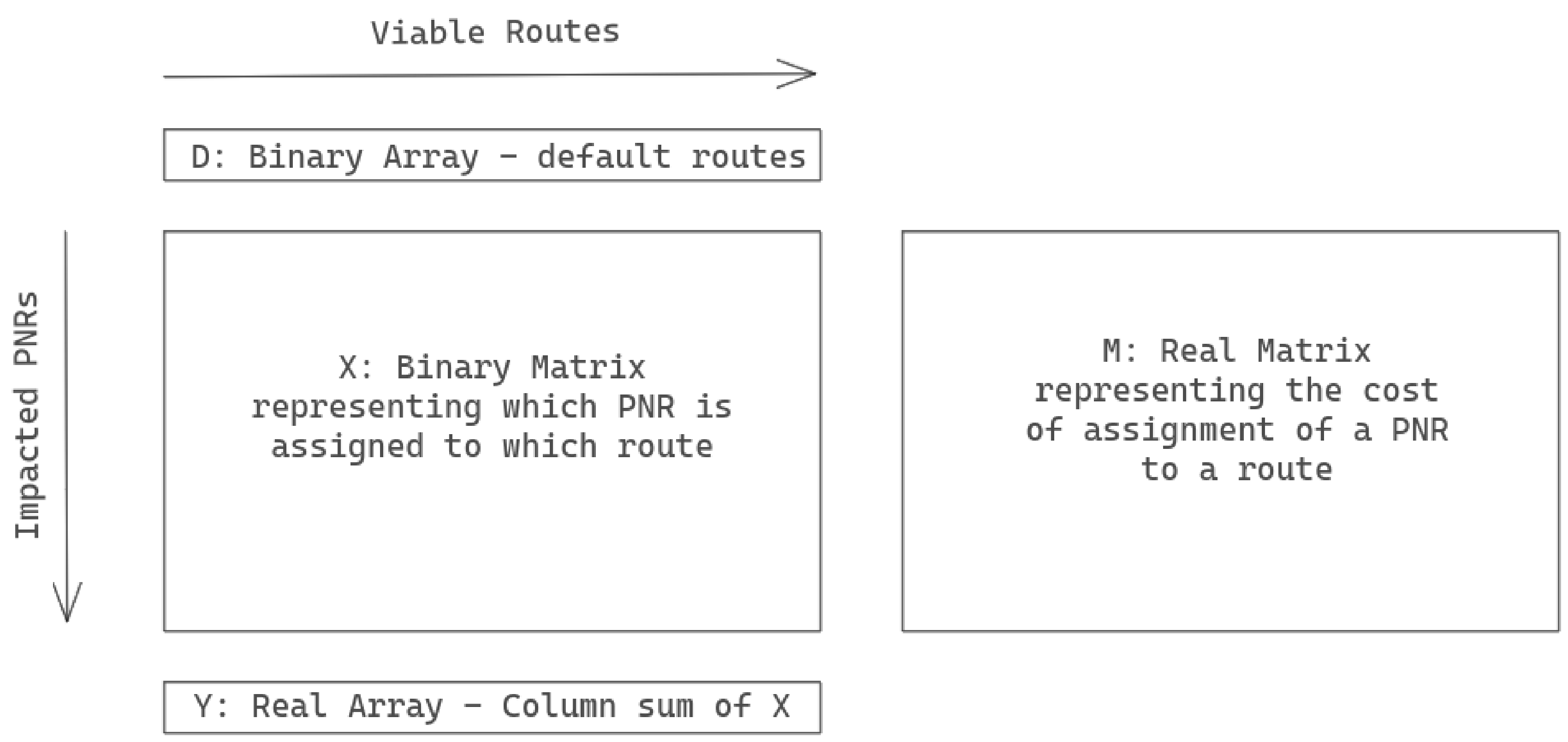
X for Impacted Flight 2

$$\begin{array}{c} \begin{array}{ccccc} R2 & R6 & R8 & R9 & \text{Dummy} \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{array}{l} \text{PNR 1} \\ \text{PNR 2} \\ \text{PNR 3} \\ \text{PNR 4} \\ \text{PNR 5} \\ \text{PNR 6} \end{array} \end{array}$$

Matrix M

- Real matrix, representing the cost of assignment of a PNR to a route.

$$M[k][i][j] = \begin{cases} S_i \times (t_j - t_o) \times \left(1 + \frac{C_j}{\beta O_i}\right) \times \left(1 + \frac{l_j}{\gamma}\right) \times CNT[i] & j < n \\ S_i \times \infty & j = n \end{cases}$$



Objective

```
for all i, j, k:
    S += X[k][i][j] * M[k][i][j]
```

- Sum over all cells of the Hadamard Product of M and X for all impacted flights.
- To be minimized.

X for Impacted Flight 1

$$\begin{array}{cccccccccc} & \text{Route 1} & R2 & R3 & R4 & R5 & R6 & R7 & \text{Dummy} & \\ \left(\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right. & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}1\\0\\1\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\1\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\1\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\1\\0\\0\\0\end{array} & \left.\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right) \begin{array}{l}\text{PNR 1}\\\text{PNR 2}\\\text{PNR 3}\\\text{PNR 4}\\\text{PNR 5}\end{array} \end{array}$$

X for Impacted Flight 2

$$\begin{array}{ccccccc} & R2 & R6 & R8 & R9 & \text{Dummy} & \\ \left(\begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}\right. & \begin{array}{c}0\\0\\0\\0\\1\\0\end{array} & \begin{array}{c}1\\0\\0\\0\\0\\1\end{array} & \begin{array}{c}0\\0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\1\\1\\1\\0\\0\end{array} & \left.\begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}\right) \begin{array}{l}\text{PNR 1}\\\text{PNR 2}\\\text{PNR 3}\\\text{PNR 4}\\\text{PNR 5}\\\text{PNR 6}\end{array} \end{array}$$

M.X for Impacted Flight 1

$$\begin{array}{cccccccccc} & \text{Route 1} & R2 & R3 & R4 & R5 & R6 & R7 & \text{Dummy} & \\ \left(\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right. & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}a\\0\\b\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\c\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\\infty\\0\\0\\0\end{array} & \left.\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right) \begin{array}{l}\text{PNR 1}\\\text{PNR 2}\\\text{PNR 3}\\\text{PNR 4}\\\text{PNR 5}\end{array} \end{array}$$

M.X for Impacted Flight 2

$$\begin{array}{ccccccc} & R2 & R6 & R8 & R9 & \text{Dummy} & \\ \left(\begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}\right. & \begin{array}{c}0\\0\\0\\0\\e\\0\end{array} & \begin{array}{c}d\\0\\0\\0\\0\\f\end{array} & \begin{array}{c}0\\0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\\0\\0\\0\end{array} & \begin{array}{c}0\\\infty\\\infty\\\infty\\0\\0\end{array} & \left.\begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}\right) \begin{array}{l}\text{PNR 1}\\\text{PNR 2}\\\text{PNR 3}\\\text{PNR 4}\\\text{PNR 5}\\\text{PNR 6}\end{array} \end{array}$$

$$S = a + b + c + d + e + f + 4 \times \infty$$

Revisiting Objectives

01 Minimizing unaccommodated passengers

02 Minimizing Mean Arrival Time Delay

03 Prioritizing higher value passengers for better route alternatives

04 Prioritizing alternatives with fewer stops

05 Prioritizing allocation of majority of passengers to the same flight

Matrix M Calculation

- t_j is the time of arrival of route j, t_o is the time of arrival of the original flight. $t_j - t_o$ is the time difference - lower the better.
- S_i is the PNR score, higher implies more priority.
- From the rearrangement inequality, the lowest objective will be obtained when higher S_i is assigned to the lower $t_j - t_o$ which is exactly what we require.
- By setting the cost to infinity when a PNR is not re-accommodated, this case is always avoided unless no other option exists.

$$M[k][i][j] = \begin{cases} S_i \times (t_j - t_o) \times \left(1 + \frac{C_j}{\beta O_i}\right) \times \left(1 + \frac{l_j}{\gamma}\right) \times CNT[i] & j < n \\ S_i \times \infty & j = n \end{cases}$$

Further M Calculation

- Yellow section: multiplies by passenger count, helps in prioritizing families and groups.
- Green section: Penalizes routes with multiple layovers (L_j: no of legs of j'th route, γ: multi leg constant)
- Blue section: Penalizes downgrading of class (C_j: average of classes of j'th route, O_i: original class, β: class constant)

$$\boxed{M[k][i][j] = \begin{cases} S_i \times (t_j - t_o) \\ S_i \times \infty \end{cases}} \boxed{\times \left(1 + \frac{C_j}{\beta O_i}\right)} \boxed{\times \left(1 + \frac{l_j}{\gamma}\right)} \boxed{\times\, CNT[i]} \quad \begin{matrix} j < n \\ j = n \end{matrix}$$

Default Prioritization

01 Minimizing unaccommodated passengers

02 Minimizing Mean Arrival Time Delay

03 Prioritizing higher value passengers for better route alternatives

04 Prioritizing alternatives with fewer stops

05 Prioritizing allocation of majority of passengers to the same flight

Default Routes

- Default route - for every impacted flight, we have one default route. $D[k][j] = 1$ if the j'th viable route for the k'th impacted flight is the default solution for this impacted flight, else zero.
- Y is the column sum over X taking into account passenger multiplicity.

$$
\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \text{D}
$$

$$
\begin{array}{c}
\begin{array}{cccccccc} R1 & R2 & R3 & R4 & R5 & R6 & R7 & \text{Dummy} \end{array} \\
\begin{pmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{pmatrix}
\end{array}
\begin{array}{l} \\ \text{PNR 1} \\ \text{PNR 2} \\ \text{PNR 3} \\ \text{PNR 4} \\ \text{PNR 5} \end{array}
\qquad
\begin{array}{c} \text{CNT} \\ \begin{pmatrix} 2 \\ 3 \\ 2 \\ 1 \\ 1 \end{pmatrix} \end{array}
$$

$$
\begin{pmatrix} 0 & 4 & 1 & 0 & 1 & 0 & 0 & 3 \end{pmatrix} \text{Y}
$$

Final Constraints

Constraint 1: One PNR is to be accommodated in one route only.

$$\sum_j X[k][i][j] = 1 \ \forall \ k, i$$

Constraint 2: The number of passengers accommodated on a flight option f must not exceed its capacity to carry C_f.

$$Z[f] \leq C_f \ \forall \ f \in F$$

Constraint 3: There is exactly one default solution for each impacted flight.

$$\sum_i D[k][i] = 1 \ \forall \ k$$

Modification of Objective

- Lower half of the objective is new; as the solver minimizes the objective, it will set $D[k][j] = 1$ for the highest $Y[j]$.
- As this part is subtracted and the solver minimizes the objective, it rewards a higher value of Y for the default solution, which is exactly what the last goal is.
- α is called **default constant**, which is chosen based on the airline's business policy.

```
for all i, j, k:
    S += X[k][i][j] * M[k][i][j]

for all j, k:
    S -= D[k][j] * Y[k][j] * alpha
```

Conflict between goals

At times, various goals can conflict with each other. For example, consider goals (2) and (5).

1. Minimize number of un-accommodated passengers
2. Minimize mean arrival time delay
3. Prioritize higher value passengers getting better alternatives
4. Prioritize alternatives with less layovers
5. Prioritize allocating a majority of passengers to the same route

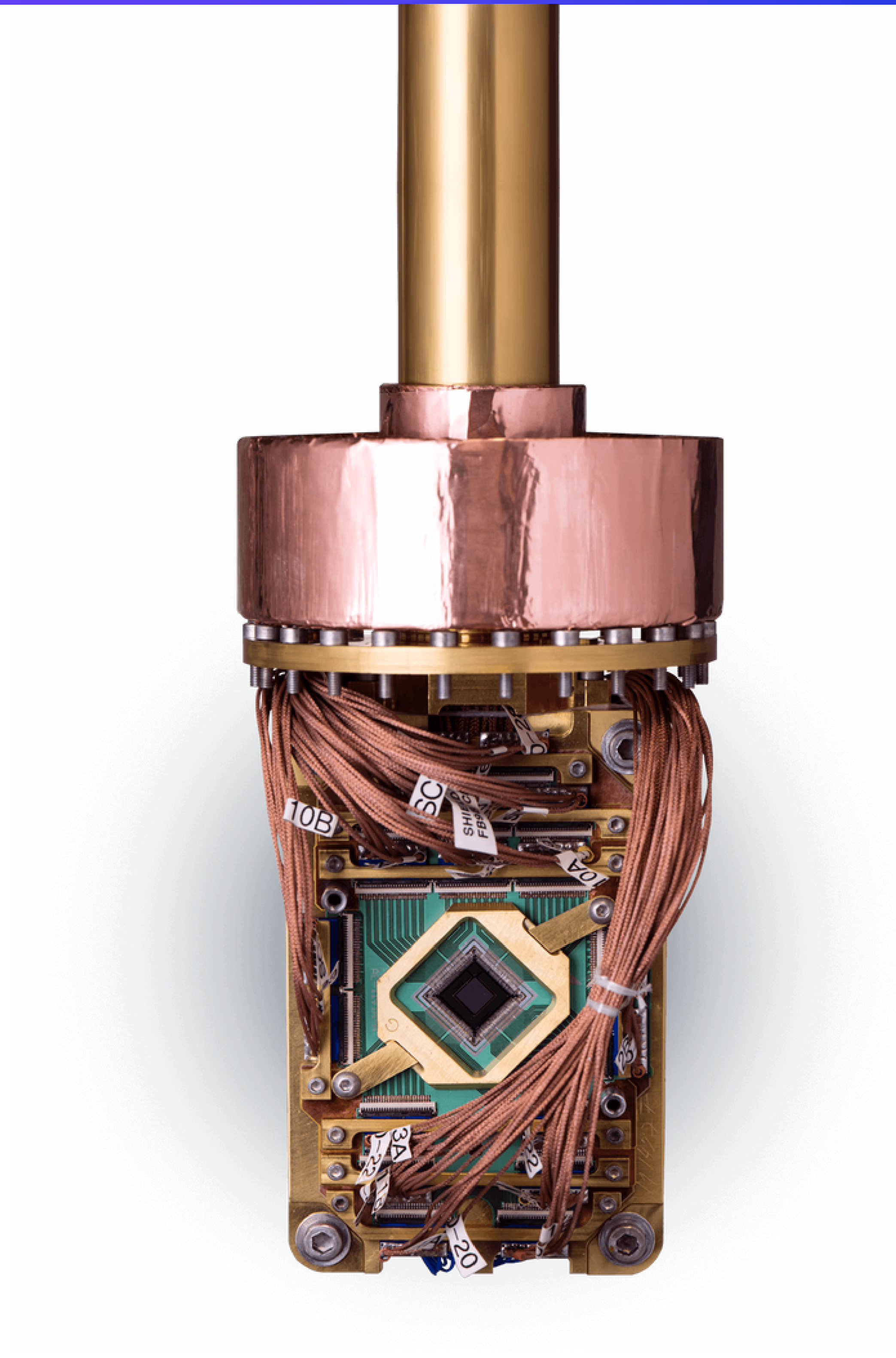
If, for example, the most optimal flight has very few empty seats, then (2) and (5) can conflict. Which goal to prioritize more ultimately comes down to the business policy of the airline, hence there exist various constants which can be tuned by the airline based on their policy.

Various constants used

Symbol	Range	Name	Description
α	$10^6 - 10^{15}$	Default Constant	Higher value prioritizes more % of PNRs under default solution
β	$1 - 100$	Class Constant	Higher value means less of a priority to maintaining /upgrading class
γ	$1 - 100$	Multi Leg Constant	Higher value means multiple-leg journeys are penalised less
T_{mL}	$3600 - 72000$	Min Layover Time	Value in seconds
T_L	$3600 - 72000$	Max Layover Time	Value in seconds
T_D	$3600 - 259200$	Max Departure Delay	Value in seconds
L_m	$1 - 5$	Max No. of Legs	Number of legs in replacement route
∞	$10^9 - 10^{18}$	Infinity	Sufficiently large finite value

The constants can be tuned by the airline based on their business policy.

DWave Leap Solver



Quantum Annealing

This Quantum Stochastic Approach maps the cost function of our problem to a potential energy profile of the solution space.

Starting from a ground state, the system evolves through time dependent Schrodinger equation

D-Wave’s Leap Hybrid Solver

DWave provides one the most accessible quantum computers to solve combinatorial optimisation problems using quantum annealing. Of various solvers, leap hybrid solver for CQM is best suited to our needs

Results and Comparisons

Classical vs Quantum (Time)

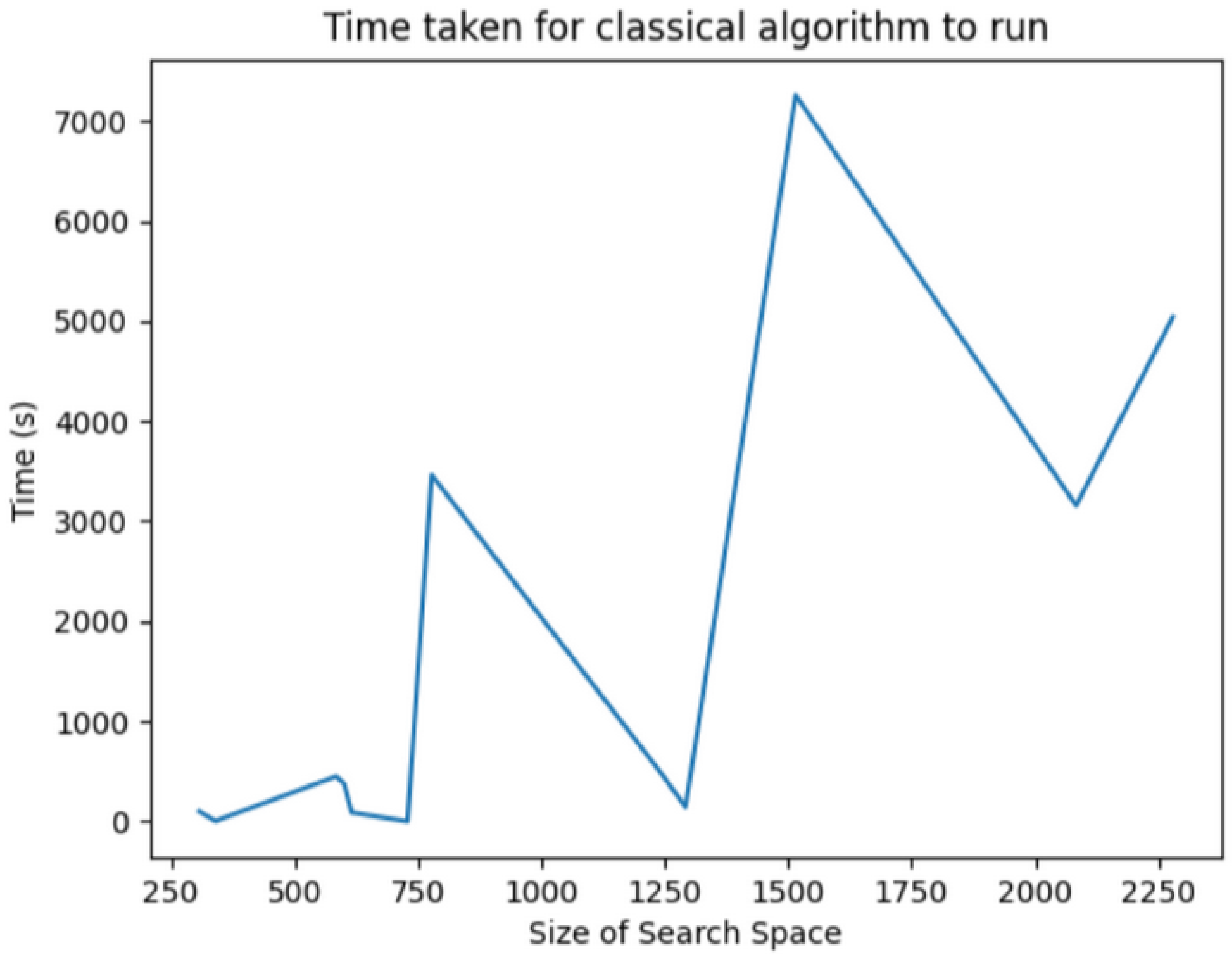
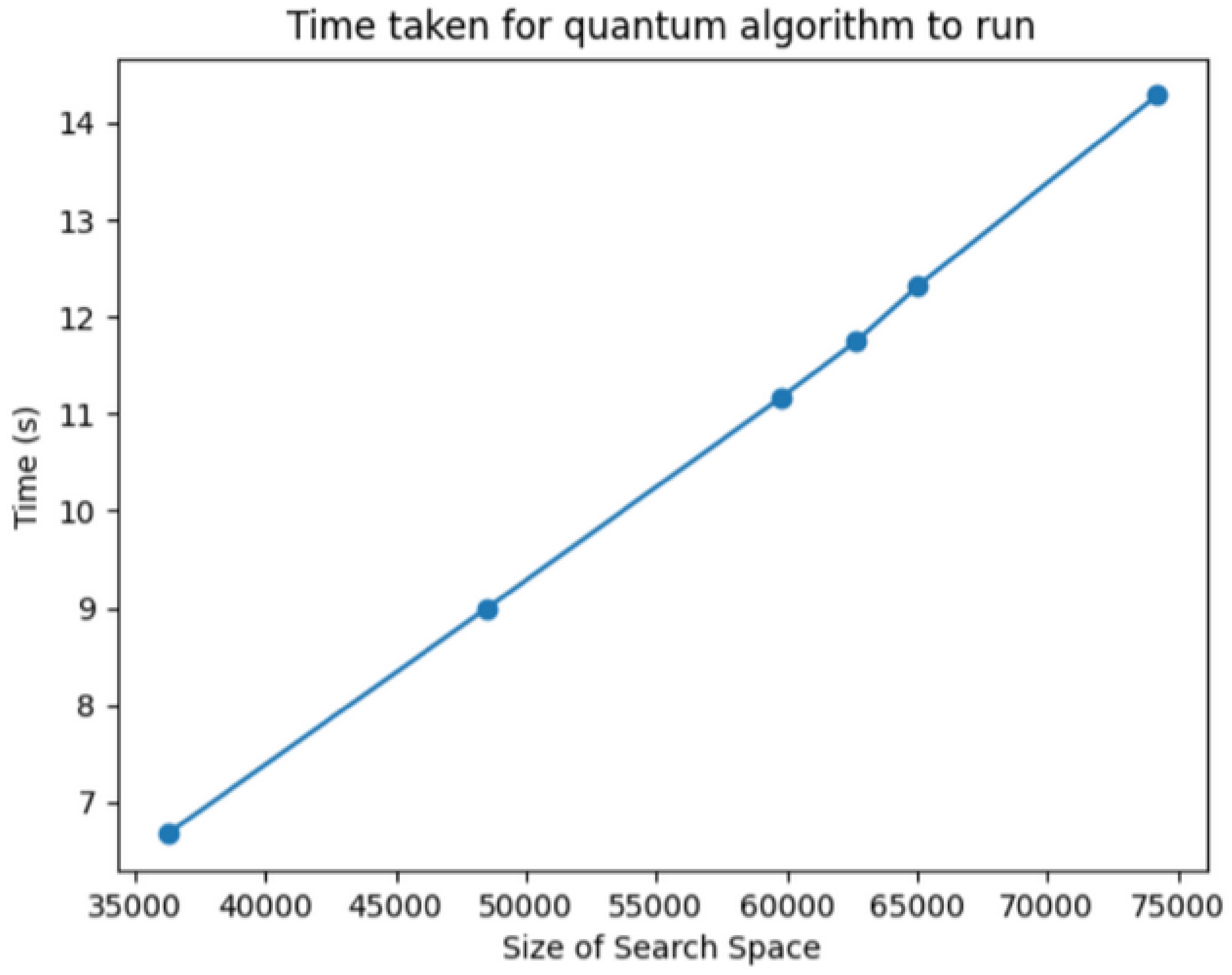


Figure 12: Run on 11th Gen Intel® Core™ i7-11800H, 16 threads, 16GB RAM. Process time is measured.

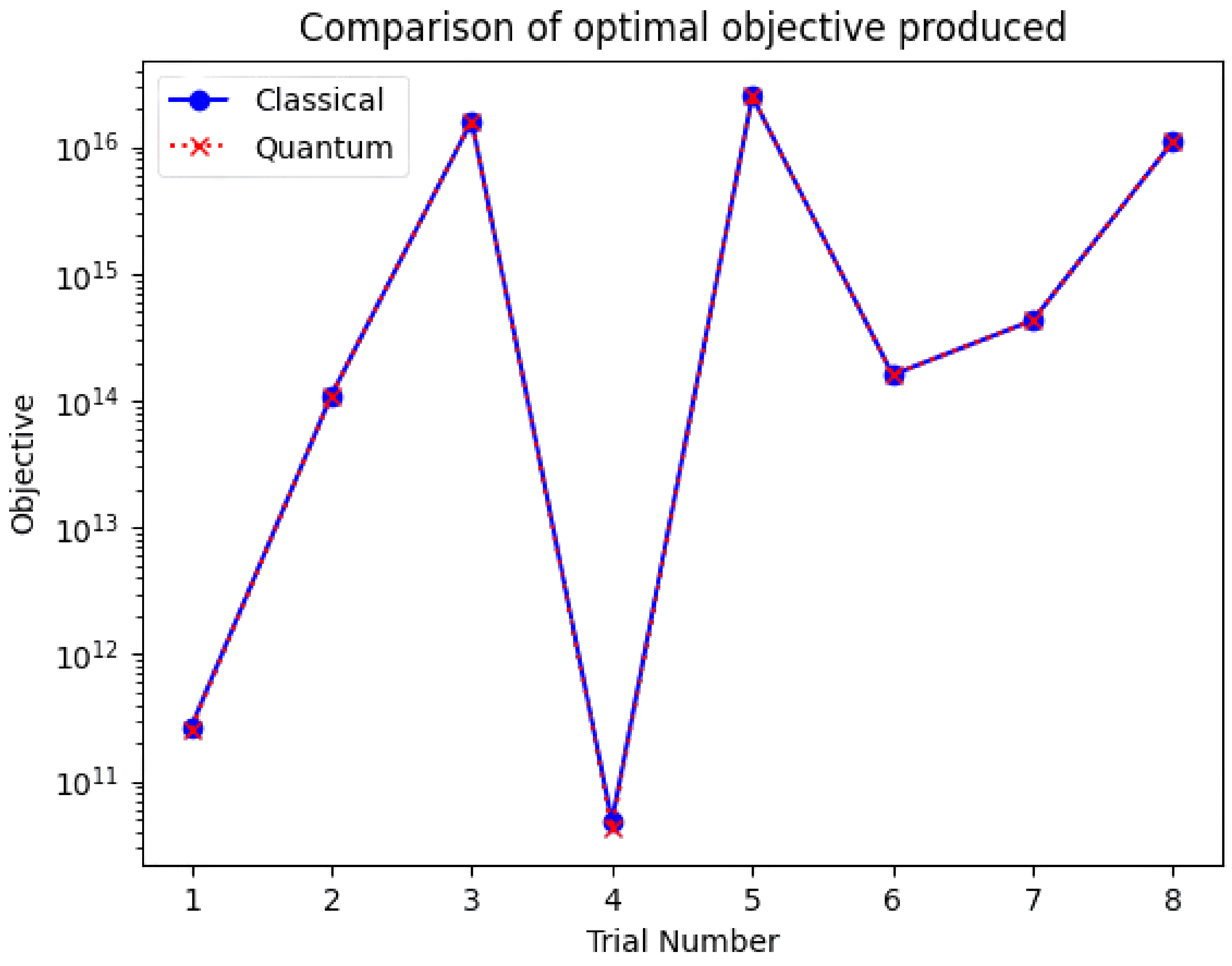


Quantum algorithm is about ~25,000 times faster. Plots have different scales.

The Quantum Solution is significantly faster than the Classical Solution.

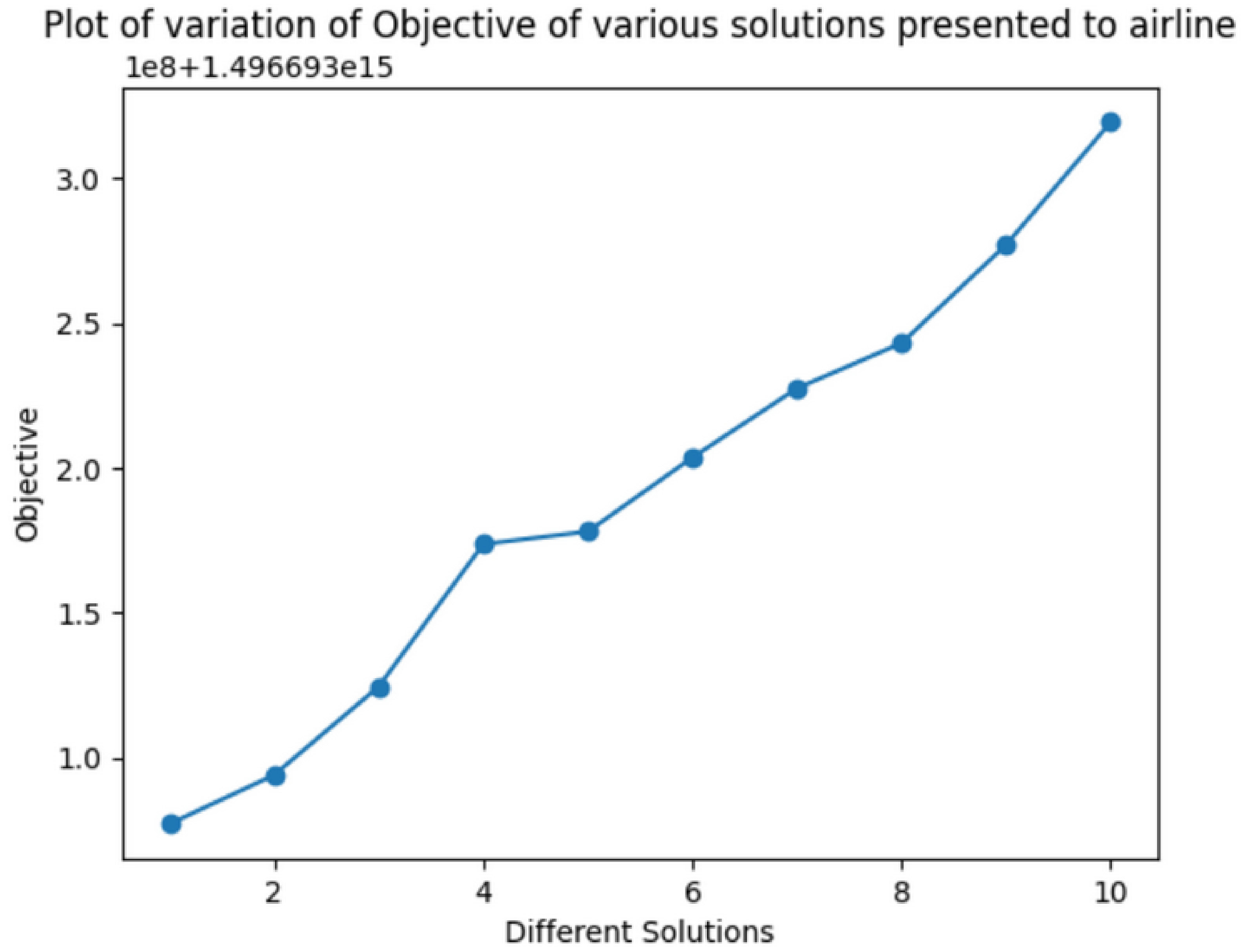
Usually such an increase in speed comes with a trade-off in the accuracy.

But the quantum solution generates results on par with those of Google OR Tools



Plot of variation of objective

Below is a graph showcasing the variation of the objective for the various solutions presented to the airline.



Business Flexibility

Different Scenarios

Aircraft Type Change

Usually leads to overbooking in one or more classes.

We treat the overbooked flight option as a cancelled flight, but add this flight option to the list of available routes as well

Flight Addition

Does not change anything in the dataset, apart from providing an additional solution for overbooked flights

As long as the flight is added to the INV and SCH files, our code deals with it

Time Change

Essentially, this is cancelling a flight at its original time, and adding the flight back with the changed time.

As long as the above two conditions are appropriately conveyed via the dataset, we can deal with it

Schedule Compression

Equivalent to cancelling multiple flights at once.

In our GUI, if all the departure keys of the cancelled flights are added, the code can deal with it similar to how it deals with single / multiple flight cancellations