OPTIMIZING AIRCRAFT SEATING ARRANGEMENT FOR THE AIRBUS A380 FLIGHT SQ322

by

Amy Kong, Inzaghi Moniaga, Derryl Sayo

Grade Proposal

We would like to propose a grade of 90/100 for this project. Our work demonstrates a well-constructed mathematical modeling process that brought together the concepts we have learned in class as part of the solution. We have also made significant efforts to iteratively improve our model, incorporating various techniques to achieve the best possible solution to the problem. This commitment to continuous improvement is evident in both our clear oral presentation and the detailed report that follows.

Abstract

In this paper, we consider the problem of finding the optimal seating arrangement for the Airbus A380 aircraft for the SQ322 flight path from Singapore Changi Airport in Singapore to Heathrow Airport in London. Methods such as combinatorial greedy fill, linear programming, integer programming, and integer programming with regularization were considered and solved using Python. A mixture of seat classes using regularization as a soft constraint for seating partitions was found to yield the best balance of well-placed seat classes and high revenue following simulated ticket buying trials.

Contents

List of Figures

List of Tables

1. Introduction

In this project, we investigate how mathematical modeling can be used to improve cabin layout or seating arrangement for the Airbus A380 aircraft with data based on flight SQ322 from Singapore Changi Airport in Singapore (SIN) to Heathrow Airport in London (LHR).

We detail the process and methods to develop a mathematical model that identifies the ideal distribution of seats within each class and placement within the aircraft to achieve the highest possible profit margin for the flight utilizing the latest available data and manufacturing design constraints.

1.1 Problem Statement

Singapore Airline just bought a fresh A380 plane from the market and needed help deciding on the seating layout to maximize profit for a flight flying from Singapore to London. What are the optimal combinations for allocating economy, premium economy, business class, and first class seats on the Airbus A380 for Singapore Airlines to fly between Singapore to London to maximize return profits?

1.1.1 Variables

Future references to a variable in the mathematical model can be found in the following.

Let:

- n : The total number of sections in the plane.

- i : The class where $i \in$ {Economy class, Premium economy class, Business class, First class}
- j : The section of the plane.
- s_{ij} : The number of seats in section j of the plane and in class i which is the decision variable.
- c_i : The cost for class i.
- a_i : The area required for building a single class i.
- A_j : The max area available in section j of the plane.
- w_i : The weight for passenger seating in i class.
- W : The max weight allowed to fly and land the plane.
- D_i : The demand for class i (amount filled for each class).
- p_{ij} : The penalty value for each class i if it is assigned in specific section j.

1.1.2 Assumptions

To build our mathematical model, we gathered data on plane specifications, ticket demand, and ticket prices. We then made the following assumptions.

Area

In order to constrain the seating arrangement of plane, we split the Airbus A380 interior into 24 indexed sections as seen in Figure 1.1. Moreover, we define two 0.51 meter wide pathways on both floors for movement inside the plane. This design focuses on having enough distance for people to travel, enough food for long flights, placing emergency exits safely within reach of everyone, and providing enough washrooms to prevent long wait lines.

The total area of each section was calculated using the general aircraft dimensions located in (Airbus, 2021, Section 2-2-0, p. 2-3). The maximum area limits for each area are shown in Table 1.1.

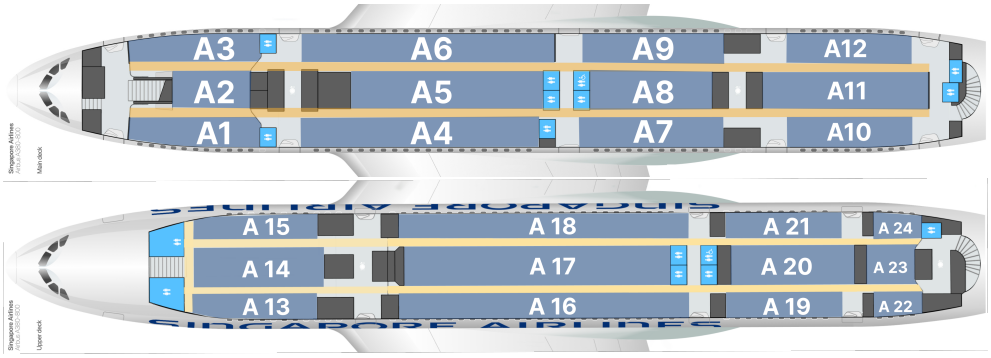

Figure 1.1: Seating sections for the Airbus A380

Section	Width (m)	Length (m)	Area (m^2)
1	1.57	10.18	15.9826
2	2.14	7.64	16.3496
3	1.57	10.18	15.9826
4	1.57	14.56	22.8592
5	2.14	12.14	25.9796
6	1.57	16.18	25.4026
7	1.57	12.06	18.9342
8	2.14	10.85	23.219
9	1.57	12.06	18.9342
10	1.57	8.89	13.9573
11	2.14	7.64	16.3496
12	1.57	8.89	13.9573
13	1.07	14.62	15.6434
14	2.14	15.62	33.8954
15	1.07	14.62	15.6434
16	1.07	19.36	20.7152
17	2.14	17.42	37.2788
18	1.07	19.36	20.7152
19	1.07	8.89	9.5123
20	2.14	8.89	19.0246
21	1.07	8.89	9.5123
22	1.07	4.44	4.7508
23	2.14	4.44	9.5016
24	1.07	4.44	4.7508

Table 1.1: Seating section area for the Airbus A380

The area of each class was calculated using plane cross section data from (SeatGuru, 2024) converted to meters. The results are shown in Table 1.2.

Cabin	Width (m)	Length/Seat Pitch (m)	Area (m^2)
Economy	0.4826	0.8128	0.392
Premium Economy	0.4953	0.9652	0.478
Business Class	0.762	1.397	1.064
First Class	0.889	2.0574	1.829

Table 1.2: Class seating areas for the Airbus A380

Weight

The maximum weight for the plane to fly is 395,000 kg according to (Airbus, 2021, Section 2-1-1, p. 1-2) general aircraft characteristics data. We also assume that each passenger has an average weight of 50 kg and carry 14 kg of luggage for a total of 64 kg per person.

Ticket Pricing

The ticket price is based on Google Flights data and was obtained by a mixture of API calls to SerpAPI's Google Flights API as well as manual inspection on the Google Flights website for the SQ322 flight code. We decided to use the worst case scenario: everyone purchased at the lowest price possible. The costs used are shown in Table 1.3.

Cabin	Ticket Price
Economy	$995
Premium Economy	$2,219
Business Class	$6,145
First Class	$12,966

Table 1.3: Ticket prices by seating class

Ticket Demand

An upper bound was found to be necessary to prevent a single class from dominating the seating arrangement. As ticket buying data are not publicly available, we created two scenarios where one followed the Zipf distribution and the second followed the current seating arrangement for the number of premium, business and first class with no upper bound for economy (SeatGuru, 2024).

The Zipf distribution, sometimes referred to as the zeta distribution, is a discrete

distribution commonly used in linguistics, insurance, and the modelling of rare events (Weisstein, n.d.). The Zipf distribution aligns well with how airplane ticket classes are inherently ranked; first class has a higher monetary value and occupies a smaller section of the plane compared to economy class. Hence, it is reasonable to assume that the frequency tickets bought for each class are inversely proportional to its value when ordered from least expensive to most expensive.

1.1.3 Constraints

Area Constraints

For each section in the plane, we constrained the total area taken by seats to the maximum section area:

$$\sum_{i \in \text{class}} s_{ij} a_i \leq A_j \quad ; \forall j \in \text{sections}$$

Weight Constraints

In order for the plane to fly, the total sum of weight of passengers per seating assignment is constrained by the maximum allowable plane weight stated above:

$$\sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} w_{ij} \leq W$$

Demand Constraints

We capped the demand for each seating class to prevent any single class from dominating the seating arrangement. The rationale behind this constraint will be explained in the discussion of the results.

$$\sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} \leq d_i$$

2. Methods and Results

Solutions were found using Python with the Google OR-tools suite (Perron & Furnon, 2024) as the backend to solve the linear and integer programming formulations for the problem stated above. A combinatorial solution using greedy fill was also employed. Full source code is included as a separate Jupyter Notebook attached with this report.

Furthermore, two scenarios were considered with differing demand constraints:

- The first scenario used the assumption that ticket buying frequencies followed the Zipf distribution to determine the maximum possible revenue when the plane's area is fully maximized with minimal unused space.
- The second scenario follows the Singapore Airlines seat map for the Airbus 380 layout 1 (SeatGuru, 2024) with no upper bound in economy to compare how much more potential revenue our model can produce given our constraints.

Below we discuss the iterative process to refine our mathematical model given the constraints of each setup and discuss the seating distributions found for each of the scenarios defined above.

2.0.1 Combinatorial Solution with Greedy Fill

The initial idea was to sort the area from smallest to largest and sort the demand (size of each class) weight vector based on the index of the highest profit to lowest profit. Then, iteratively fill the area with the new demand starting from the highest cost to lowest cost while checking the weight, area, and demand constraints. Each iteration reduced the demand until no more seats fit based on the constraints.

This resulted with a maximum revenue of $\$1,510,514$. Table 2.1 displays the number of seats for each class and Figure 2.1 depicts the areas where each class is assigned.

Class	Number of Seats
Economy	482
Premium	250
Business	50
First	24

Table 2.1: Greedy Fill Results

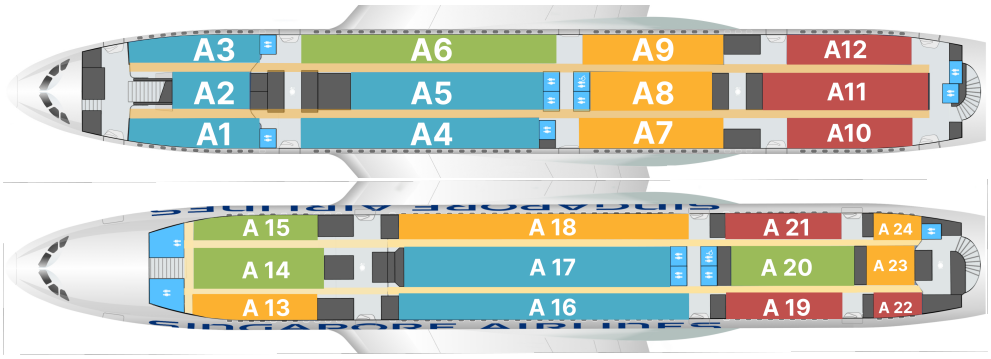

Figure 2.1: Greedy Fill Plane Assignment

2.0.2 Linear Programming

We formulate the linear programming problem as following:

$$
\begin{aligned}
\max_{s} \quad & \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} c_i \\
\text{s.t.} \quad & \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} w_{ij} \leq W \\
& \sum_{i \in \text{class}} s_{ij} a_i \leq A_j \quad ; \forall j \in \text{sections} \\
& \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} \leq d_i \\
& x \geq 0
\end{aligned}
$$

An initial solution using linear programming was found for both scenarios.

In scenario 1, based on the Zipf distribution, the maximum revenue was found to be $1,564,737 with total seats found in Table 2.2 and seating arrangements found in Figure 2.2.

Class	Number of Seats
Economy	544.7579
Premium	250
Business	50
First	24

Table 2.2: Linear Programming Seating Counts (Scenario 1)

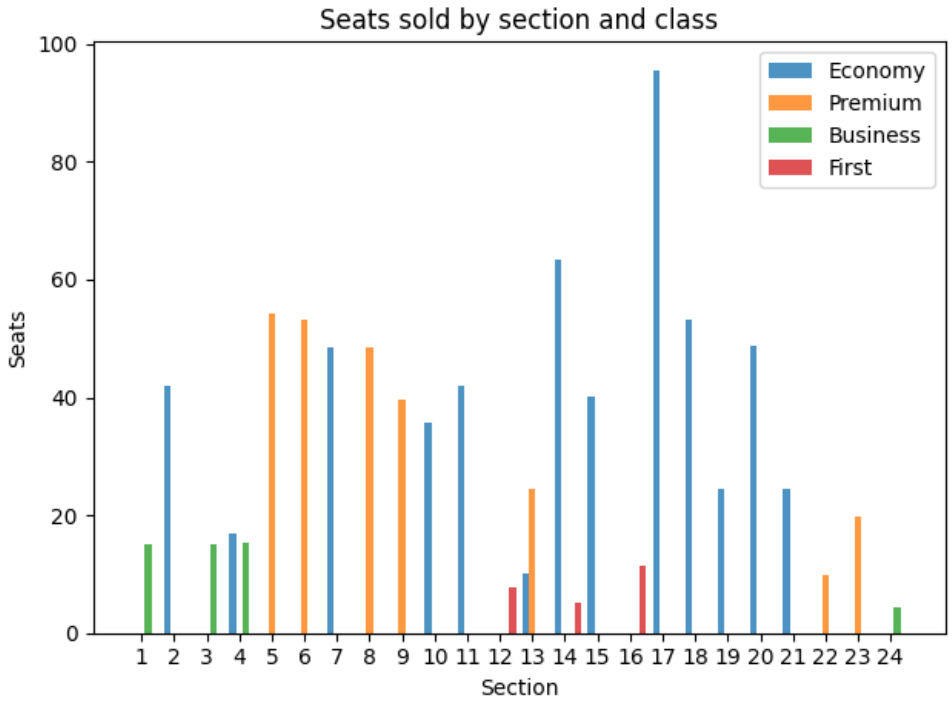

Figure 2.2: Linear Programming Seating Arrangement (Scenario 1)

In scenario 2, where there is no upper bound for economy, the maximum revenue was found to be $1,290,349 with total seats found in Table 2.3 and seating arrangements found in Figure 2.3.

Class	Number of Seats
Economy	833.6912
Premium	38
Business	60
First	12

Table 2.3: Linear Programming Seating Counts (Scenario 2)

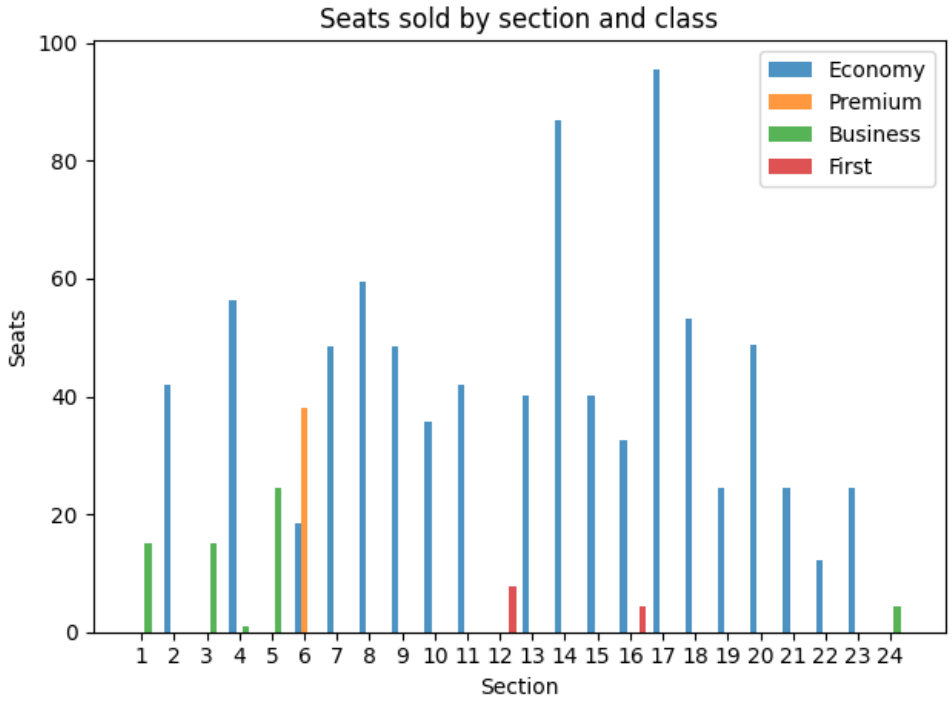

Figure 2.3: Linear Programming Seating Arrangement (Scenario 2)

Although linear programming is sufficiently fast to compute using tools such as OR-tools, the result contained fractional decision variables which doesn't translate well to real-world scenarios; it not possible to have fractional seating or have people paying for a fraction of a seat. Furthermore, we observed that in some sections mixed different classes.

2.0.3 Integer Programming

An integer programming approach was used to ensure that there are no fractional seat assignments. The problem was then reformulated to restrict the assignment to integer

values with the standard form.

$$
\begin{aligned}
\max_{s \in \mathbb{Z}} \quad & \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} c_i \\
\text{s.t.} \quad & \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} w_{ij} \leq W \\
& \sum_{i \in \text{class}} s_{ij} a_i \leq A_j \quad ; \forall j \in \text{sections} \\
& \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} \leq d_i \\
& x \geq 0
\end{aligned}
$$

Theoretically, the integer programming solution has a unique optimal solution. However, in practice, the solver was extremely slow and got stuck refining the objective value to reach the theoretical optimum. The results shown below were from running the OR-tools SAT solver for integer programming with a timeout of five minutes. Although not technically optimal, the seating counts and objective value were deemed suitable for comparison.

Solving the integer programming formulation using demand from scenario 1, we get a max revenue of $1,563,218 with seating counts and arrangements found in Table 2.4 and Figure 2.5.

Class	Number of Seats
Economy	543
Premium	250
Business	50
First	24

Table 2.4: Integer Programming Seating Counts (Scenario 1)

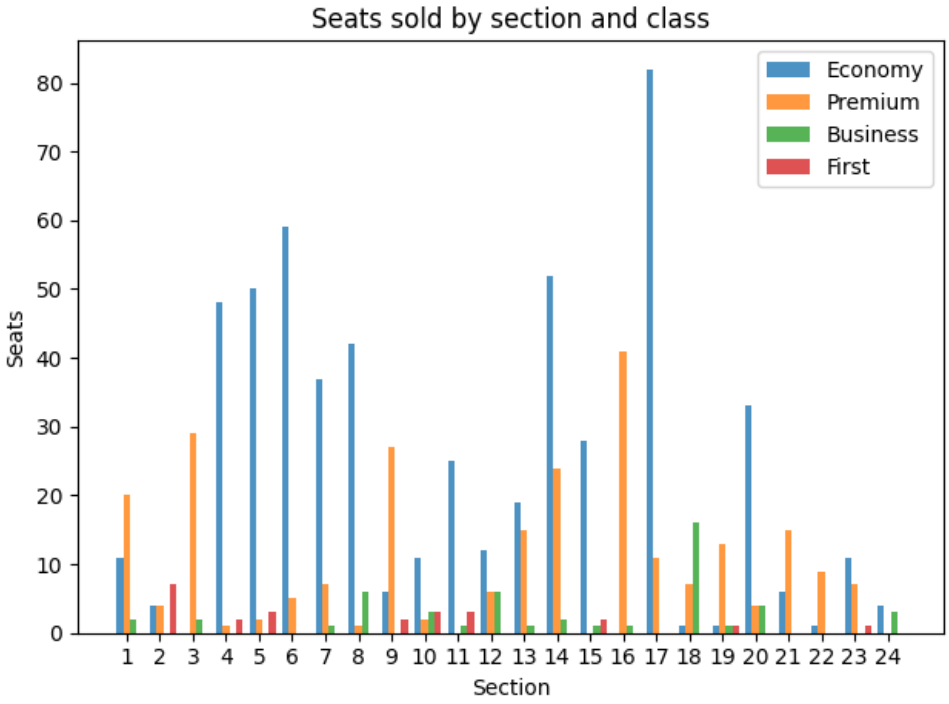

Figure 2.4: Integer Programming Seating Arrangement (Scenario 1)

The integer programming solution for scenario 2 resulted in a max revenue of $1,287,160 with seating counts and arrangements found in Table 2.5 and Figure 2.5.

Class	Number of Seats
Economy	831
Premium	38
Business	60
First	12

Table 2.5: Integer Programming Seating Counts (Scenario 2)

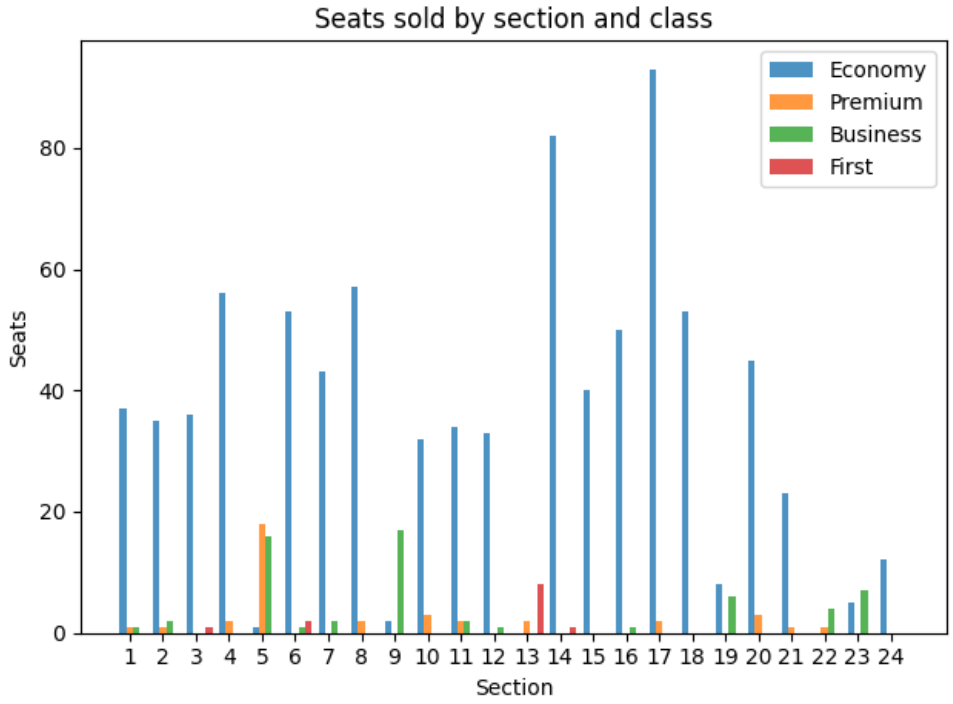

Figure 2.5: Integer Programming Seating Arrangement (Scenario 2)

Although the integer programming formulation has an optimal seating arrangement, the runtime is much slower in comparison with the other methods and the halted runtime gives non-deterministic seating arrangements with the same seat counts. Similarly to linear programming, we observed that there are still sections where different classes sit together.

2.0.4 Integer Programming with Regularization

To resolve the mixed seating arrangement into more well-partitioned sections, we used a method inspired by regularization to introduce a penalty parameter to the objective function. The formulation is similar to the integer programming formulation, but with an additional penalty, p_{ij}, for each class i if it is assigned in specific section j.

$$
\begin{aligned}
\max \quad & \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} c_i - p_{ij} s_{ij} \\
\text{s.t.} \quad & \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} w_{ij} \leq W \\
& \sum_{i \in \text{class}} s_{ij} a_i \leq A_j \quad ; \forall j \in \text{sections} \\
& \sum_{i \in \text{class}} \sum_{j \in \text{sections}} s_{ij} \leq d_i \\
& x \geq 0
\end{aligned}
$$

The idea is we want to penalize the objective value if a certain fare class is assigned outside of some defined optimal location. In our case, we tailored the penalty to assign first class to the first three sections, business class to the next three sections, premium economy to the next six, and economy to the last twelve.

Solutions were found efficiently by assigning a value of 0 if the seating class is in the "correct" section and a penalty of $\$10,000$ if assigned otherwise. Our cost penalty matrix P is then element-wise multiplied by each corresponding s_{ij} in our objective function:

$$
P = \begin{bmatrix}
10000 & \dots & \dots & 0 \\
10000 & \dots & 0 & 10000 \\
10000 & 0 & \dots & 10000 \\
0 & \dots & \dots & 10000
\end{bmatrix}
$$

Note: The penalty matrix heavily depends on the demand vector and can be adjusted depending on different scenarios.

We see in Figures 2.6 and 2.7 for scenarios 1 and 2, respectively, that the seating arrangements are partitioned as set in the penalty matrix. The seating counts are shown in Tables 2.6 and 2.7.

Class	Number of Seats
Economy	511
Premium	218
Business	50
First	24

Table 2.6: Integer Programming w/ Regularization Seating Counts (Scenario 1)

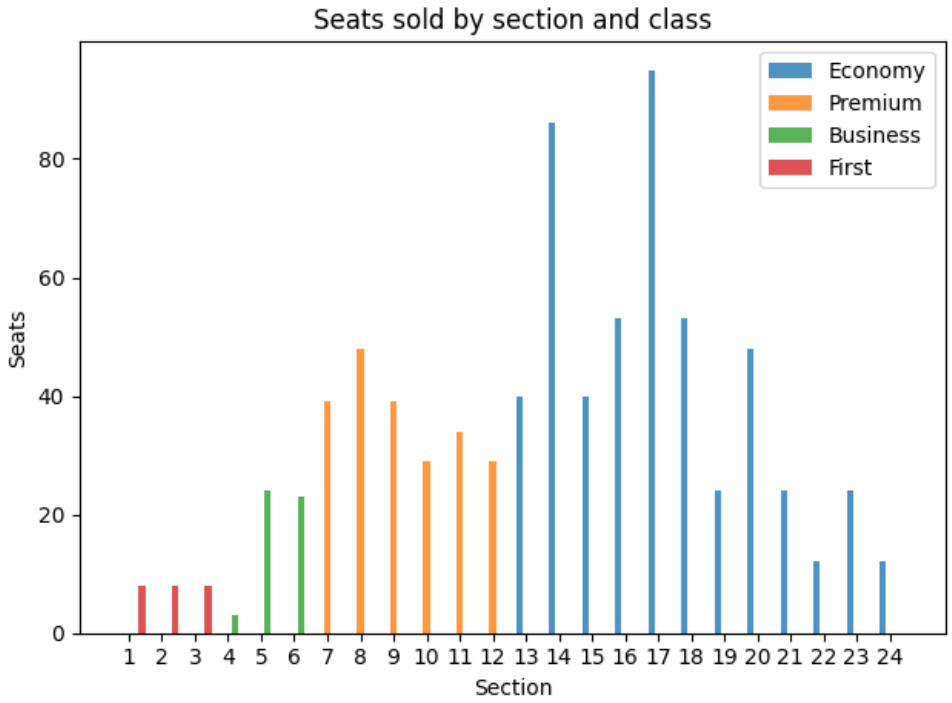

Figure 2.6: Integer Programming w/ Regularization Seating Arrangement (Scenario 1)

Class	Number of Seats
Economy	777
Premium	38
Business	60
First	12

Table 2.7: Integer Programming w/ Regularization Seating Counts (Scenario 2)

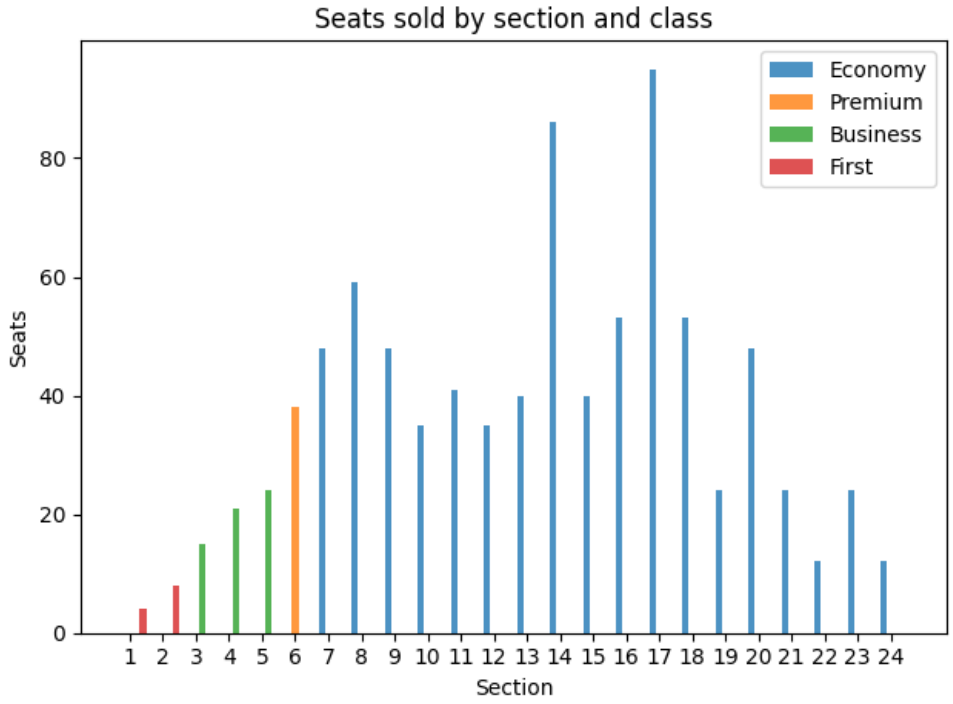

Figure 2.7: Integer Programming w/ Regularization Seating Arrangement (Scenario 2)

There are some other points to note about the regularization solutions. Since we can reduce this formulation to the standard IP form by setting all penalties to zero, the maximum revenue is bounded above by the corresponding standard IP solutions. However, encoding the penalties to the objective function rather than hard constraints allows for greater flexibility in dictating the final layout by making the partitions more or less strict. Moreover, future experiments can use the regularization term to find solutions under unconstrained demands, similar to the Log-Barrier problem for interior point methods.

2.1 Simulated Ticket Buying

To determine the arrangement with the best revenue, we ran simulations for a set of customers buying tickets for the flight. We took the average of 1000 iterations, where we used a random total demand from sampling the log-normal distribution with an underlying normal distribution mean of $\mu = \log(800)$ with variance $\sigma = 0.2$, and split into the four fare classes sampling from the Zipf distribution. A comparison of results

between the two scenarios and single class assignments are shown in Figure 2.8.

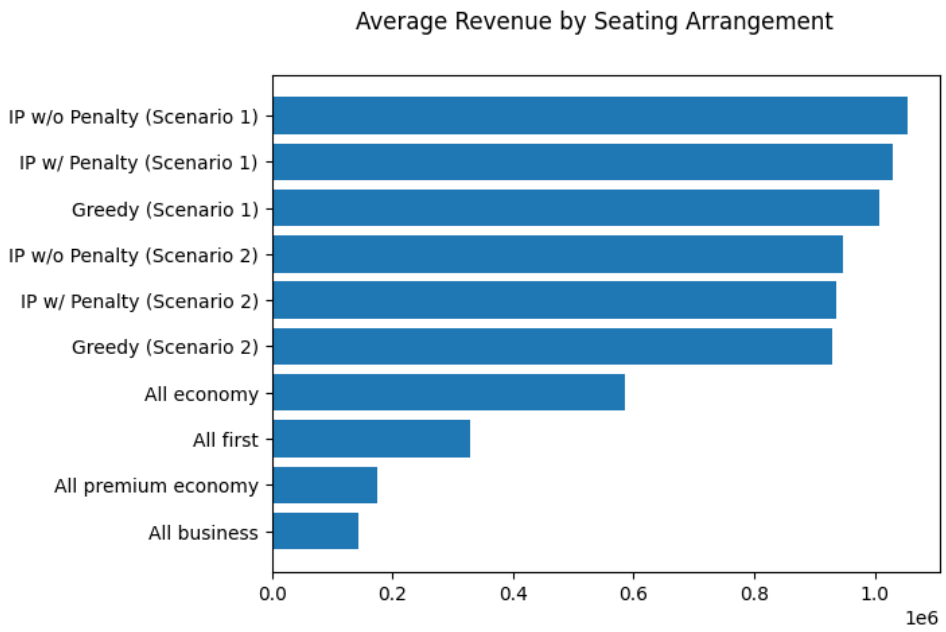

Figure 2.8: Average Revenues by Seating Arrangement (Scenario 1)

The resulting average revenues are shown in Table 2.8

Method	Average Revenue
Integer Programming w/o Penalty (Scenario 1)	$ 1,052,606.475
Integer Programming w/ Penalty (Scenario 1)	$ 1,031,274.711
Greedy (Scenario 1)	$ 1,006,775.681
Integer Programming w/o Penalty (Scenario 2)	$ 945,133.86
Integer Programming w/ Penalty (Scenario 2)	$ 936,015.292
Greedy (Scenario 2)	$ 928,576.793
All Economy	$ 591,199.776
All First	$ 334,341.144
All Premium Economy	$ 173,185.064
All Business	$ 143,066.475

Table 2.8: Average Revenues by Seating Arrangement (Scenario 1)

Table 2.8 illustrates the need for the demand constraint. Although filling the entire plane with the highest fare class may give the highest theoretical maximum revenue, under

any reasonable distribution of seating arrangements (like the simulation parameters set above), it will fall behind a mixed class seating arrangement. So, a cap to each fare class was needed, and the closest assumption we could make was to match the demand upper bounds to the distribution of seats used in this simulation. As a result, the mixed seating arrangements using integer programming resulted in the highest average revenue with the regularization solution slightly worse in terms of monetary value but much better in terms of layout. Hence, we concluded that the integer programming with penalty solution is the best method to solve the airplane seating assignment problem for a fixed demand assumption.

2.2 Comparison to Singapore Airlines Seating Layout 1

As an extension to our original results, we computed optimal seating arrangements for demand matching the Singapore Airlines A380 Layout 1 (SeatGuru, 2024) with an increased economy seating demand. The results are shown in Figure 2.9 with values in Table 2.9.

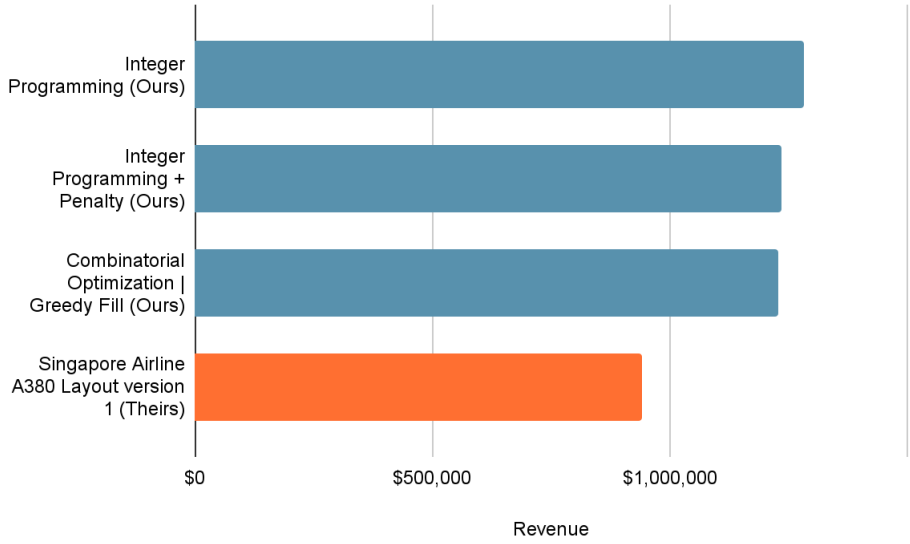

Figure 2.9: Average Revenues by Seating Arrangement (Scenario 1)

Method	Revenue
Integer Programming	$1,282,840
Integer Programming w/ Penalty	$1,234,718
Combinatorial Solution	$1,227,544
Singapore Airlines Layout 1	$ 939,949

Table 2.9: Average Revenues by Seating Arrangement (Scenario 1)

We see that under our airplane space assumptions and from the documented seating arrangements above, we found that we can fit around double the amount of economy seats, which increases the total revenue for a fully booked plane significantly.

3. Conclusion

This project presented an iterative approach to refining a mathematical model used to represent and solve the optimal airplane seating arrangement problem for an arbitrary aircraft partitioned into seating area sections with a set demand. In order to capture well-defined seating partitions, an integer programming approach with an additional regularization penalty term allowed for the highest revenue while balancing both numerical and subjective placement constraints.

Further improvements can be done to represent more complex constraints, most notably the demand. The reality in airplane ticket buying is that tickets can be bought in a wide window before some closing date in which passengers may choose to upgrade or cancel. This uncertainty may be better represented by other stochastic programming approaches. Additionally, grouping the section into rows of three would give something more like what we see in the real-world.

References

Airbus. (2021). Airbus Aircraft AC: A380 [Accessed: February 16, 2024]. https://www.airbus.com/sites/g/files/jlcbta136/files/2021-11/Airbus-Aircraft-AC-A380.pdf

Perron, L., & Furnon, V. (2024, March 7). *Or-tools* (Version v9.9). https://developers.google.com/optimization/

SeatGuru. (2024). Singapore Airlines Airbus A380 Seat Map [Accessed: February 16, 2024]. https://www.seatguru.com/airlines/Singapore_Air/Singapore_Air_Airbus_A380_C.php

Weisstein, E. W. (n.d.). Zipf distribution [Accessed: April 20, 2024]. https://mathworld.wolfram.com/ZipfDistribution.html