About Aircraft Boarding Processes: Methods, Simulations, and Conclusion

BAO Cheng (1335784)
ZHANG Dinggen (1336574)
Advisor: Dr.Omar Dib

Abstract This research report examines the inefficiencies inherent in current aircraft boarding processes and investigates various strategies, simulations, and potential improvements. The problem of prolonged boarding times has been exacerbated by the rising demand for air travel and limitations in infrastructure, primarily due to aisle and seat interference. The report outlines several boarding methods, including random, back-to-front, unassigned seating, 'WILMA', 'Stephen', and dynamic optimization methods, analyzing their theoretical efficiencies and practical challenges. Utilizing the Mesa framework, simulations were conducted to compare these methods, revealing that while some theoretical models show promise, practical implementation often faces significant hurdles, primarily due to economic considerations and passenger preferences. The report concludes that airlines prioritize revenue generation over boarding efficiency, suggesting that a balance between the two is necessary to enhance the overall boarding experience.

Keywords: boarding methods;simulation;divide and conquer;priority selection

1. Introduction

The quality of in-flight services often faces criticism: meals are subpar, cabin temperatures fluctuate between extremes, and seating is so cramped that passengers may question whether the seats have been designed for comfort. However, the only issue that both airlines and passengers can agree upon is the excessive duration of the boarding process(CNBC, 2023). When boarding notifications are announced, passengers rush to the gate as if responding to a starting gun(CGP Grey, 2019). Once aboard, they navigate slowly through the narrow aisle, struggling to find space for their luggage, which often results in others becoming stuck in the middle, unable to move forward or backward. If passengers sit on the aisle seat board first, the situation can become even more complicated, as they must stand to allow others to pass, thereby blocking those queuing behind them. This raises the question: why can't the boarding process be simplified to alleviate the burden on passengers(CGP Grey, 2019)?

In the 1970s, the boarding process typically took just 15 minutes; however, it now averages at least 30 minutes, with some instances taking even longer.

This increase in boarding time can be attributed to the rapid rise in the number of passengers utilizing airplanes for long-distance travel, which has outpaced the construction of necessary infrastructure, such as airports and aircraft, as well as the expansion of flight routes(BAI et al., 2022). Two primary factors contribute to the prolonged boarding times(Caroline, 2020):

1. **Aisle Interference**: Passengers blocking the pathway while stowing luggage create delays for everyone.

2. **Seat Interference**: Passengers seated in aisle positions must move to accommodate those in middle or window seats, further congesting the narrow space.

In this report, we will explore the methodologies being developed to address this problem and how these methods can be simulated using computational tools. Although this issue is intertwined with numerous unknown variables and subjective factors, we aim to summarize the research findings of other scholars and present our viewpoints. The results of our investigation are both intriguing and thought-provoking.

2. Boarding Methods Overview

Drawing from our practical experiences, extensive research, and in-depth analysis of relevant literature, we identify several different boarding methods currently in use within the aviation industry, which are either widely adopted or continuously emerging.

2.1 Random & Back-to-Front Methods

Many airlines within China and internationally employ random seat assignments as their passenger boarding strategy(BAI et al., 2022). During check-in, passengers receive fixed seat assignments generated by the system. Subsequently, during boarding, economy class passengers queue in a random order to board the aircraft. It can also be considered as the baseline of all boarding methods below.

Conversely, the back-to-front boarding method appears to be more scientifically grounded, allowing rear-seat passengers to board first. However, this approach has also proven ineffective, as it often results in passengers congregating at the rear of the aircraft, leading to aisle congestion and diminished boarding efficiency.

2.2 Unassigned Seating Registration Method

Since its inception, Southwest Airlines has employed a random, unassigned seating registration method. Passengers are allowed to check in online 24 hours before their flight, at which point the system assigns a boarding order number (e.g., A1, B20) based on a first-come, first-served principle. Those who opt for priority boarding are permitted to board first, while other passengers are categorized into A, B, and C groups. During boarding, announcements are made to call groups in sequential order, allowing passengers to queue according to their assigned number and select their seats freely upon entry. This method enhances passenger flexibility, minimizes aisle congestion, and facilitates smoother luggage placement.

2.3 The "WILMA" Method

In 2017, United Airlines introduced the "WILMA" method, which stands for "Window-Middle-Aisle". By this method, passengers seated in window positions are allowed to board first, those who sit in middle seats followed, and finally the passengers in aisle seats. Priority boarding and business/first-class passengers continue to board early.

2.3.1 Mathematical Description

The priority value for each passenger in the WILMA method is calculated using the following equation:

$$P_i = \begin{cases} 1, & \text{if passenger } i \text{ has a window seat} \\ 2, & \text{if passenger } i \text{ has a middle seat} \\ 3, & \text{if passenger } i \text{ has an aisle seat} \end{cases}$$

Where:

- P_i is the priority value for passenger i

The WILMA boarding method follows this priority order:

1. Passengers with window seats board first, in order of their priority value (lowest to highest).
2. Passengers with middle seats board next, in order of their priority value (lowest to highest).

3. Passengers with aisle seat board last, in order of their priority value (lowest to highest).

This boarding sequence is designed to minimize the number of times passengers need to cross the aisle, reducing congestion and interference during the boarding process. The rationale is that window seat passengers can board without disrupting the aisle, followed by middle seat passengers, and finally, aisle seat passengers.

2.3.2 Summary

According to data from United Airlines, this method can reduce boarding time by up to two minutes per flight, resulting in approximately $200 savings per flight in labor, ground services, and aircraft idle time. Given that United operates nearly 5,000 flights daily, this could lead to a daily savings of around $1 million(Caroline, 2020).

2.4 The "Stephen" Method

While the "WILMA" method effectively addresses the issue of aisle passengers needing to make way for those seated in inner positions, it does not resolve the challenge of luggage placement blocking access. Researchers have long sought ways to streamline the boarding process. In 2005, astrophysicist Jason Stephen(*Steffen Boarding Method*, 2024), frustrated by lengthy boarding lines at Seattle Airport, developed what he believed to be the most effective boarding method. His approach resembles "WILMA" but is more systematic(Steffen and Hotchkiss, n.d.). This method involves categorizing passengers by their seat assignments—window, middle, and aisle—while ensuring that boarding occurs in alternating rows. Initially, passengers from odd-numbered rows queued, followed by those from even-numbered rows. Window seat passengers from alternating sides board first, followed by other passengers, gradually filling middle and aisle seats.

2.4.1 Mathematical Description

The priority value for each passenger in the Steffen Method is calculated using the following equation:

$$P_i = \begin{cases} 2i - 1, & \text{if } i \text{ is odd} \\ 2n - 2i + 1, & \text{if } i \text{ is even} \end{cases}$$

Where:

- P_i is the priority value for passenger i
- i is the index of the passenger, where $i = 1, 2, 3, ..., n$
- n is the total number of passengers

The Steffen Method follows a specific alternating pattern to board passengers based on their priority values:

1. Passengers with odd priority values board first, starting from the highest value.
2. Passengers with even priority values board next, starting from the highest value.

This alternating pattern is designed to minimize the number of times passengers need to cross paths in the aircraft aisle, reducing congestion and interference during boarding.

The mathematical formulation of the Steffen Method is relatively simple. Still, it has been proven to be one of the most efficient boarding strategies to reduce boarding time and cabin congestion. The method's effectiveness has been demonstrated through various simulation studies and real-world implementations.

2.4.2 Summary

Stephen's method effectively reduces aisle congestion and ensures that each passenger has adequate space. Simulations indicated that this approach is 20-30% faster than random boarding(Steffen and Hotchkiss, n.d.). However, theoretical advantages do not always translate into practical applications. Implementing this method would necessitate either assigning seats based on queue order or boarding passengers according to their seat numbers, both of which present challenges. Boarding by seat number may lead to confusion due to incorrect announcements, while queue-based seat allocation could frustrate passengers who are assigned less desirable seats. Significantly, passengers typically prefer to sit with friends or family rather than be separated for efficiency. Consequently, Stephen's method has not achieved widespread adoption.

2.5 The Dynamic Optimization Boarding Method

Subsequently, Zeineddine proposed the Dynamic Optimization Boarding method(Hassan, 2017). This approach focuses on grouping traveling companions together and automatically categorizing passengers during check-in, notifying each group of their designated boarding time. The system assigns priority based on seat location—window seats receive the highest priority, aisle seats are the lowest, and rear seats rank higher than front seats. For group boarding, passengers can queue together regardless of their seat priorities. However, practical scenarios reveal complexities. For instance, if four passengers occupy seats 20A, 20B, 20C, and 20D, the initial priority appears straightforward; however, seat 20D blocks access for passengers seated in 20E and 20F.

2.5.1 Mathematical Description

The priority value for each passenger can be calculated using the following equation:

$$P_i = w_1 \cdot d_i + w_2 \cdot c_i + w_3 \cdot l_i$$

Where:

- P_i is the priority value for passenger i
- d_i is the distance between passenger i's seat and the door
- c_i is the number of passengers currently in the aisle between passenger i's seat and the door
- l_i is the time required for passenger i to stow their luggage
- w_1, w_2, and w_3 are weight factors that determine the relative importance of each component in the priority calculation

The distance component d_i represents the physical distance the passenger needs to travel to reach their seat, contributing to the overall boarding time. The congestion component c_i accounts for the interference caused by other passengers already in the aisle, which can lead to delays and slower movement. The luggage component l_i considers the time required for the passenger to stow their luggage, which can also impact the boarding process.

By optimizing the boarding sequence based on these priority values, the Dynamic Optimization Boarding Method aims to minimize the total boarding time by reducing aisle congestion, passenger interference, and luggage storage delays.

The weight factors w_1, w_2, and w_3 can be adjusted to prioritize specific aspects of the boarding process, depending on the particular needs and constraints of the airline or the aircraft configuration.

2.5.2 Summary

More intricate scenarios arise when group members cannot cluster together and must move back several rows equal to their group size. For example, if three passengers are seated in 20A, 20B, and 20C, subsequent passengers would need to occupy seats no closer than row 17 to prevent boarding congestion. Despite its complexities, this method effectively addresses the issue of families or friends boarding separately. While its efficiency is comparable to that of Stephen's method, neither approach has gained widespread traction, with many airlines still employing sub-optimal boarding strategies(BAI et al., 2022).

2.6 Graphical Shortest Paths Method

In exploring the optimization strategy for the boarding problem, we employ the concept of graphs to model an aircraft's seating layout and passengers' boarding process. This approach allows us to analyze and solve the boarding efficiency problem systematically and quantitatively. We can consider the seating layout of an airplane as a graph, where each seat is a node, and the paths of passengers from the gate to their seats are edges. We aim to find a boarding sequence that minimizes the total cost for all passengers to reach their seats.

2.6.1 Definitions

- **Node Definition**: in the graph, nodes represent each seat in the airplane. Each node contains information about the location of the seat, such as row and column numbers.
- **Edge Definition**: an edge represents the potential path of a passenger from the gate to their seat. Each edge has a weight representing the distance or time required for the passenger to travel from the gate to the seat.
- **Diagram Construction**: When constructing a diagram, the physical layout of the airplane needs to be considered, including the arrangement of the seats and the structure of the corridors. This is accomplished using an adjacency list, where each entry represents a connection between two nodes and the corresponding weight.

2.6.2 The Shortest Path Algorithm: Dijkstra

Dijkstra's algorithm is a greedy algorithm for computing single source shortest paths. It selects the next nearest node by maintaining a priority queue and updating the distance of its neighboring nodes. Following is the implementation:

1. **Initialization**: the boarding gate is set to the source node, and the distances of all other nodes are initialized to infinity, except for the source node, which has a distance of zero.
2. **Update Distance**: for each node, check all its neighboring nodes and update the distance to the adjacent node if the distance to reach the neighboring node through the current node is shorter.
3. **Repeat the process**: repeat the above process until all nodes have been visited or the distances of all nodes have been determined.
4. **Path length ordering**: Once the shortest paths for all passengers have been computed, the passengers are sorted according to the length of these paths. This can be done by comparing the total weight of each path.
5. **Prioritization**: Sorting ensures that passengers with shorter paths (i.e., those closer to the gate) board first, reducing their waiting time and congestion at the gate.
6. **Boarding simulation**: Add passengers to the boarding queue in sorted order.

2.7 Dyanmic Programming Method

To solve the boarding problem using Dynamic Programming (DP), we can think of it as an optimization problem in which we want to minimize the total boarding time for all passengers. We can think of the airplane seating layout as a graph, where nodes represent seats and edges represent the paths passengers take from the gate to their seats. We aim to find a path that minimizes all passengers' boarding time.

In a typical boarding process, there is only one boarding gate. The design of the dynamic programming algorithm needs to be simplified accordingly, as we do not need to consider different gate options. Each passenger will board directly from the unique gate, so we only need to consider how to sequence the passengers to minimize the total boarding time. In this case, we can characterize the problem using some greedy strategy, i.e., prioritizing the passenger furthest from the gate to board first to reduce the waiting time of other passengers. This approach is called the "farthest-from-first" strategy.

The design of the algorithm is shown below:

1. **GetAllSeats**: defines a list seats of all available seats on the airplane, seats consist of row and column numbers.

2. **Initialize the cost table**: the cost array stores the minimum total time to reach each seat.Initialization to 0 means that the boarding time is 0 when there are no passengers.

3. **Calculate Distance**: Prioritize seats that are farther away based on the distance from each seat to the gate and sort the seats based on the distance.

4. **Update the cost table**: Iterate through the sorted seats and update the cost array to ensure that the boarding time for each seat is the current minimum total time plus the distance to that seat.

5. **Reconstruct the optimal solution**: reconstruct the optimal boarding path from the cost array, recording each seat and corresponding passenger ID.

3. Boarding Methods Simulation

In exploring the aforementioned boarding methods, we aim to simulate the aircraft boarding process using Python(Moreira et al., 2023). To facilitate this, we identified a GitHub repository that utilizes the Mesa framework to construct an agent-based model designed to evaluate and compare different boarding algorithms(Jerzy and Konrad, 2020). By analyzing and modifying the code within this repository, we can observe the performance of these methods in a real-world simulation.

3.1 The Mesa Framework

The Mesa framework was employed in this project to simulate various aircraft boarding strategies, which enables:

- users to swiftly create agent-based models using built-in core components (such as spatial grids and agent schedulers) or customized implementations;

- visualize these models through a browser-based interface;

- analyze the results using Python's data analysis tools.

The Mesa framework provides a flexible, scalable, and easily visualizable platform for simulating aircraft boarding strategy projects, facilitating the efficient simulation, analysis, and demonstration of the effects of different boarding strategies.

3.2 The Agents

In our project, we have defined two types of agents:

1. **PassengerAgent**: Represents an individual passenger. Each passenger is characterized by a unique ID, seat position, boarding group, and number of bags. These attributes influence the passenger's boarding behavior and timing.

2. **PatchAgent**: Represents the seats, aisles, and walls within an airplane. These proxies are utilized to depict the physical layout of the aircraft and provide information regarding seat adjustments during the simulation.

3.3 The Plane Model

The Plane Model serves as the core of the simulation and inherits from Mesa's Model class. It encompasses the aircraft's grid layout (MultiGrid), which allows multiple agents to occupy the same cell, and a QueueActivation that regulates the sequence in which the agents are executed.

3.4 Implementing the Boarding Algorithm

We abstract each boarding algorithm as a function that accepts a model as an argument. These functions are responsible for creating the corresponding passenger agents within the model and adding them to the boarding queue.

3.5 Passenger Boarding Process Visualization

Our codes provide a visual interface that enables users to interactively select the boarding algorithm, adjust parameters (e.g., whether to permit seat adjustments and baggage size), and observe the simulation in real-time.

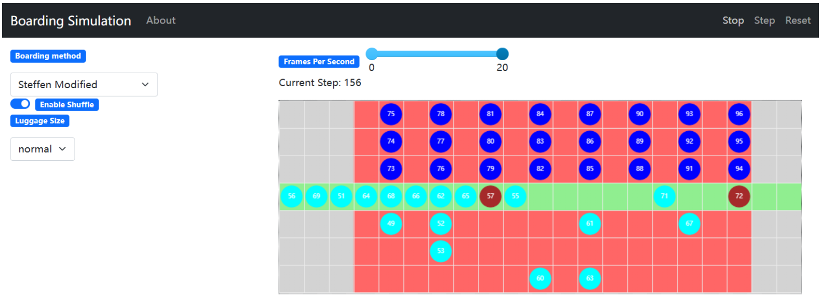

Figure 1
Boarding Simulation Snapshot

1. The PlaneModel initializes the aircraft's mesh layout, generating proxies for walls, seats, and aisles.

2. The designated boarding algorithm is invoked to create passenger agents and assign them to the boarding queue.

3. Each passenger agent determines its actions based on its current state and the surrounding environment, as provided by the PatchAgent. For instance, if the aisle in front of the passenger is unobstructed and they have not yet boarded the plane, they will advance.

4. As passenger agents navigate toward their designated seats, they attempt to sit down upon arrival. If a seat is occupied or unavailable, the passenger will wait until it becomes accessible.

5. Additionally, passenger agents simulate baggage handling times, which may influence their boarding duration.

4. Boarding Algorithm Analysis

We used the following key metrics for our algorithmic analysis:

4.1 Key Metrics 1: Average Boarding Time

This is the main indicator of the efficiency of the boarding algorithm. Collecting the total time required for boarding in each simulation run and then calculating the average of these times helps us understand the average performance of different boarding methods over multiple runs.

4.2 Key Metrics 2: Kernel Density Estimation(KDE)

The basic idea of kernel density estimation is to place a kernel function around each data point, and the superposition of these kernel functions forms a probability density estimate for the entire data set. The kernel functions are usually symmetric, centered on the data points, and the specific form of the kernel function determines their shape.

Kernel density estimation is a non-parametric way of estimating the probability density function of a random variable, which can help us understand the shape and density of the distribution of the data.

4.3 Key Metrics 3: Time Complexity and Space Complexity

When designing the boarding process, airlines need to find a balance between time complexity and space complexity. For example, an algorithm may effectively reduce boarding time (low time complexity). Still, if storing additional data (high space complexity) requires a lot of memory, this may not be optimal.

Understanding an algorithm's time and space complexity can help airlines optimize their boarding strategy, for example, by reducing unnecessary data storage or improving the algorithm's logic to reduce complexity.

4.4 Analysis Conclusion

Figure 2 presents a comparative analysis of various airplane boarding strategies. The x-axis represents the time elapsed during the boarding process, while the y-axis depicts the density or occupancy of the aircraft cabin. Several boarding strategies are shown, including Random, Front-to-Back, Back-to-Front, "WIMA" and variations of the "Steffen" method.

The data suggests that the Steffen Perfect and Steffen Modified strategies demonstrate the highest efficiency, with lower cabin density and a more gradual increase in occupancy over time compared to the other methods. Conversely, the Random and Front-to-Back approaches exhibit more pronounced peaks in cabin density, indicating more significant congestion during the boarding process.

This visualization provides valuable insights into different airplane boarding techniques' relative performance and effectiveness. The data can inform airlines and industry stakeholders in their efforts to optimize the boarding process, reduce passenger wait times, and enhance the overall passenger experience.

The conclusion of multiple boarding strategies allows for a comprehensive

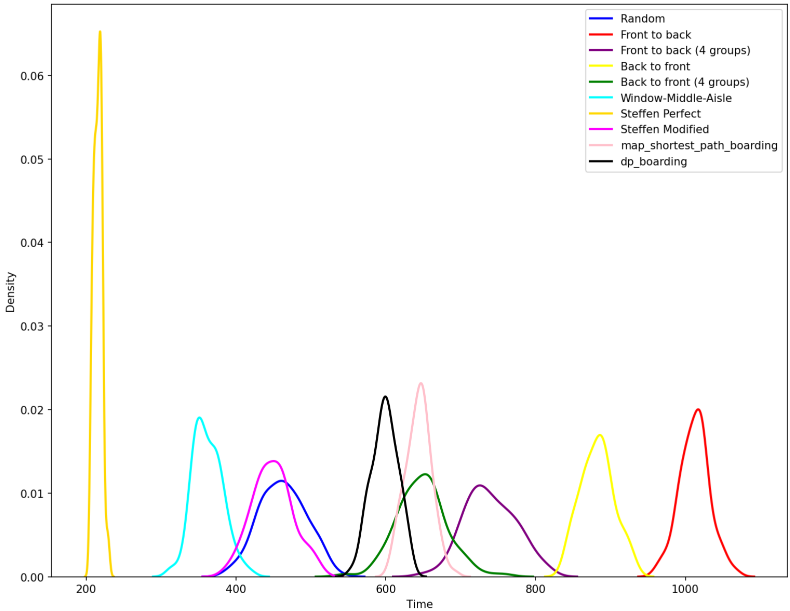

Figure 2
Diagram of Each Boarding Method

comparative analysis, enabling researchers and practitioners to evaluate the trade-offs between factors such as boarding time, spatial constraints, and passenger flow. This type of quantitative analysis can contribute to developing more efficient and passenger-centric boarding procedures in the commercial aviation industry.

- **Highest Efficiency**: As can be seen from the graph, the "Steffen Perfect" method has the lowest average boarding time, indicating that it is the most efficient method in the simulation environment.
- **Second best choice**: "Steffen Modified" and "map shortest path boarding" follow closely behind, showing better performance, with an average boarding time of less than 600 seconds.
- **Medium performance**: "Back-to-front (4 groups)" and "dp boarding" are in the middle of the pack, with average boarding times between 600 and 700 seconds.
- **Needs Improvement**: "Random" and "Window-Middle-Aisle" have

longer average boarding times of more than 700 seconds, indicating that these methods are less efficient in the simulation.

- **Worst Performance**: The "Front-to-back" method has the highest average boarding time, close to 1000 seconds, which could mean that this method is the least efficient in the simulation environment.

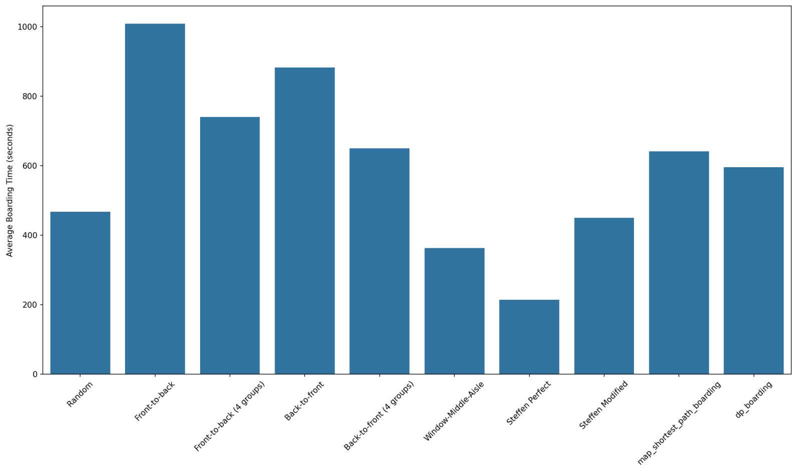

Figure 3
Average Boarding Time

Figure 3 shows that a well-designed boarding strategy can significantly reduce the average boarding time. "Steffen Perfect" method performs the best in the simulation, while the "Front-to-back" method performs the worst. This suggests that choosing a boarding strategy is critical to improving boarding efficiency. Airlines can learn from these simulation results to optimize their boarding processes to reduce boarding times and increase passenger satisfaction. In addition, these results provide a basis for further research to explore how these strategies can be adapted to different flight sizes, aircraft layouts, and passenger behavior patterns.

Regarding time and space complexity, it isn't easy to analyze for the previous algorithms because it is composite and contains grouping logic and some other logic that makes it challenging to quantify. However, for our implementation of graph shortest path and DP algorithms, we can describe their efficiency well in terms of time and space complexity.

For the graph shortest path algorithm and the DP algorithm, the time &

space complexity are reflected as follows, Where V is the number of vertices, N is the number of passengers:

	Graph Construction	Dijkstra's Algorithm	Sorting Passengers
Time Complexity	$O(1)$	$O(V^2)$	$O(NlogN)$
	Graph Storage	Dijkstra's Algorithm	Passengers List
Space Complexity	$O(V)$	$O(V)$	$O(N)$

Table 1
Graph Shortest Path Algorithm Analysis

	List Creation	Distance Calculation	Sorting Seats	Filling DP Tables
Time Complexity	$O(N)$	$O(N)$	$O(N)$	$O(N)$
	Seats List	Distance Dictionary	DP table	Optimal Path
Space Complexity	$O(N)$	$O(N)$	$O(N)$	$O(N)$

Table 2
Graph Shortest Path Algorithm Analysis

The graph shortest path algorithm may be less efficient in calculating the shortest DP path because it needs to run Dijkstra's algorithm individually for each passenger, which leads to higher time complexity when the number of passengers is high. The DP algorithm determines the optimal boarding order by calculating the distance from all seats to the gate at once and then using dynamic programming, which may be superior in terms of time complexity, especially when the number of passengers is high. The space complexity of both algorithms depends mainly on the number of passengers. Still, the graph shortest path algorithm may require more memory to store the graph structure and the auxiliary data structures of the Dijkstra algorithm.

5. Conclusion: Theory vs. Reality

As mentioned, enhancing boarding efficiency could improve passenger experience and yield significant cost savings for airlines, creating a win-win scenario. However, airlines have not uniformly adopted such approaches. While increased efficiency may boost revenue, there are instances where a slower boarding process can be more profitable(Thompson, 2024). Passengers often appreciate early boarding, and airlines have found that many are willing to pay for this privilege. Priority boarding facilitates faster entry and enhances passengers' feelings of exclusivity, generating additional revenue opportunities. Airlines can offer this service as part of a paid option, a loyalty program benefit, or bundled with other premium services.

Theoretically, the dynamic optimization boarding method has demonstrated exceptional performance in simulations. However, its practical implementation presents several challenges. This method necessitates increasing ground staff to assist passengers, leading to heightened personnel costs. In contrast, current random boarding methods require minimal management and resources, as passengers typically queue independently with simple announcements and few ground service staff. Introducing a new boarding method would entail redesigning boarding processes, adjusting gate assignments, and educating passengers about the new rules. For global airlines, such coordination could prove to be prohibitively expensive.

Conversely, Southwest Airlines, recognized for its distinctive unassigned seating approach, plans to abandon this method in 2025 in favor of system-assigned seats (Meyer, 2024). Despite its commitment to democratizing air travel as America's leading low-cost airline, Southwest acknowledges that 80% of customers and 86% of potential customers prefer assigned seating.

At first glance, Southwest's decision appears to respond to customer demand; however, it may also reflect a strategic move for survival. In the short term, selling priority boarding and charging for baggage fees generates more immediate revenue than merely focusing on reducing boarding time and costs. The indirect savings from decreased boarding bridge fees are negligible compared to the direct income derived from boarding privileges. Airlines are not incapable of enhancing boarding efficiency; instead, they have opted for the model that currently maximizes revenue. The equilibrium between time and efficiency ultimately hinges on passengers' willingness to pay for privileges and experiences rather than altering their habits to save a few minutes in queuing (Meyer, 2024).

Approaches similar to Southwest's, which promote baggage check-in and provide incentives, may effectively alleviate boarding congestion(Meyersohn, 2023). Furthermore, passengers who plan their luggage and minimize the frequency of overhead bin openings could enhance the overall boarding experience. Ultimately, this balance between efficiency and experience may depend significantly on individual choices. (Caroline, 2020)

References

BAI, Q., WU, S., CAO, R., MENG, S.-y. and XU, Z.-m. (2022). Review on passenger boarding process optimization at civil airports, **22**(4): 68–88.
URL: https://transport.chd.edu.cn/cn/article/doi/10.19818/j.cnki.1671-1637.2022.04.005

Caroline, D. (2020). Math proves the fastest, most efficient way to board a plane. Section: Science.
URL: https://www.popularmechanics.com/science/a30549336/math-boarding-plane/

CGP Grey (2019). The better boarding method airlines won't use.
URL: https://www.youtube.com/watch?v=oAHbLRjF0vo

CNBC (2023). Why airlines won't fix inefficient boarding.
URL: https://www.youtube.com/watch?v=iuGEqnmySvo

Hassan, Z. (2017). A dynamically optimized aircraft boarding strategy, **58**: 144–151. Publisher: Pergamon.
URL: https://www.sciencedirect.com/science/article/abs/pii/S0969699716302071

Jerzy, L. and Konrad, S. (2020). Wo0kie3/simulation-of-the-plane-boarding-strategies: Me and my friend, artificial life project.
URL: https://github.com/Wo0kie3/Simulation-of-the-plane-boarding-strategies

Meyer, J. (2024). Why southwest airline's big seating change is sparking controversy.
URL: https://www.explore.com/1631580/reason-why-southwest-airline-seating-change-open-assigned-controversial/

Meyersohn, N. (2023). Boarding a plane is a nightmare. here's how we got here | CNN business.
URL: https://edition.cnn.com/2023/09/16/business/airline-boarding-process-explained/index.html

Moreira, H., Ferreira, L. P., Fernandes, N. O., Silva, F. J. G., Ramos, A. L. and Ávila, P. (2023). A simulation study of aircraft boarding strategies,

11(20): 4288. Number: 20 Publisher: Multidisciplinary Digital Publishing Institute.
URL: https://www.mdpi.com/2227-7390/11/20/4288

Steffen, J. H. and Hotchkiss, J. (n.d.). Experimental test of airplane boarding methods.
URL: http://arxiv.org/abs/1108.5211

Steffen Boarding Method (2024). Page Version ID: 1233705966.
URL: https://en.wikipedia.org/w/index.php?title=Steffen_Boarding_Method&oldid=1233705966

Thompson, J. (2024). There are quicker ways to board a plane–so why don't airlines use them?
URL: https://www.scientificamerican.com/article/there-are-quicker-ways-to-board-a-plane-so-why-dont-airlines-use-them/