Multi-Agent Aircraft Boarding Simulation

Sicheng Jia
504803461
Computer Science Department
University of California, Los Angeles
jsicheng@cs.ucla.edu

Daisy Zheng
905526314
Computer Science Department
University of California, Los Angeles
dayz@cs.ucla.edu

Brandon Hua
804595738
Computer Science Department
University of California, Los Angeles
huabrandon0@gmail.com

I. INTRODUCTION

With advancements in agent-based modeling and multi-agent systems in the past decade, complex agent interactions and crowd modeling have produced realistic simulations of crowd movement within an environment. These simulations have become increasingly important in the game development field and for crowd flow analysis, where the goal is to simulate crowd behavior under different environments such as emergency evacuations. Many previous works have explored various components of crowd modeling such as collision avoidance, individual agent personalities, and lane formation [1] [2] [3]. We plan to apply these ideas in the scope of multi-agent aircraft boarding simulation. While some works have explored optimal methods for aircraft boarding, to our knowledge, none have modeled aircraft boarding with multi-agent physical simulations [4].

II. PROPOSAL

We propose to incorporate various crowd simulation ideas, such as collision avoidance and queue formation, to the multi-agent simulation of aircraft boarding. We plan to simulate known boarding strategies such as the back-to-front and window-middle-aisle. We will use statistical methods as a basis to compare the performance of each boarding strategy. Lastly, we will reason about the results based on our observations of the simulated agents' behaviors and tendencies.

III. RELATED WORKS

Patil et al. have developed an interactive algorithm to direct and control crowd movement in simulations [3]. By using navigation fields, the authors are able to conduct goal-directed simulations of heterogeneous agents that are able to avoid obstacles and collisions with each other. Users draw a guidance field, which consists of lines that specify the intended crowd traffic flow. Guidance fields can also be extracted from video by detecting motion patterns. These guidance fields are then used to compute navigation fields on a gridded environment by using a variant of Dijkstra's algorithm.

By drawing influence from Personality Trait Theory, Guy et al. were able to control the perceived personalities of individual agents in a crowd simulation [2]. Based off of the Eysenck 3-Factor personality model, Guy et al. were able to map Psychoticism, Extraversion, and Neuroticism to perceived personalities such as aggressive, shy, tense, assertive, active, and impulsive behaviors. Using randomized simulation parameters such as each agent's preferred speed, distance to other agents, and agent radius, Guy et al. were able to realistically model the various behaviors of individual agents within a crowd setting.

Mas et al. have found that the total boarding time is dependent on the number of boarding interferences [5]. A boarding interference is defined as an instance where a passenger blocks another passenger from reaching their seat. Mas et al. have also found that the boarding interferences is dependent on the number of carry on items each passenger has. Simulation of the plane boarding was done using numerical calculations in Visual Basic and Excel. Surprisingly in their simulation, Mas et al. have found that the random boarding method performed the best.

In 2008, Steffen discovered an optimal airplane boarding strategy by using Markov Chain Monte Carlo simulation [6]. In the optimal method, called the Steffen's Perfect boarding method, window seat passengers are boarded every other row on one side first before doing the same on the other side. This process is then repeated for the middle then the aisle seats. This method combines elements of the back-to-front boarding method and the window-middle-aisle (Wilma) boarding method. The intuition behind the Steffen's boarding method is to minimize the luggage stowing obstruction occurrences. Since the Steffen's Perfect boarding method might be too complex for passengers to follow, Steffen also introduced a modified version where every other row on one side is boarded at once.

Iyigunlu et al. have run simulations on different boarding methods on the Boeing 777 (42 rows) and Airbus 380 (48 rows) aircrafts [4]. These aircrafts contain three aisle of seats, where each aisle contains three columns of seats. Iyigunlu et al. have found that the reverse pyramid boarding method (back-to-front) performed the best on the Boeing 777, while the Steffen's boarding method performed best on the Airbus 380.

Cimler et al. have also run numerical simulations on plane boarding methods in NetLogo. Cimler et al. observed that as the number of rows increased, the differences in the boarding methods also became more apparent. Cimler et al. also observed that as the number of rows increased, the time it took for the Steffen method of boarding grew linearly, while other methods did not scale as well for larger number of rows.

Lastly, Cimler et al. tested the effects of late passengers. For some methods, this caused a large increase in boarding time, while for others, it did not affect boarding times significantly.

IV. Methodology

In order to physically simulate plane boarding methods, a physics engine is needed. We chose to use Unity3D as our simulation platform. Unity offers a variety of tools for physical simulations and goal-oriented agents. We initially attempted to develop our passenger AI with ML-Agents, Unity's machine learning toolkit. However, upon further investigation, we decided it more efficient and time-friendly to use Unity's NavMesh system instead, as it offers a simple interface for specifying walkable areas, creating moving agents, and defining obstacles.

A. Environment Setup

In our first simulation setup, we attempted to recreate the scenarios in previous works. Our environment consisted of the plane seats with the passengers already queued in line to board. However, since the focus of our work is the physical simulation of plane boarding, we decided to also include the queuing process in our simulation.

Our final environment consists of an airport terminal gate, the walkway to the gate, and the airplane itself. The airport terminal gate contains 144 seats (colored red) in the main lobby, and 12 seats in a first class lounge. The seats are gridded in a manner that encourages the passenger agents to walk freely in the seat aisles rather than squeeze in between seats. To simulate real world airport terminals, the path leading up to the walkway is taped off for some length to encourage line formation. This also helps discourage the agents from crowding around the narrow walkway when trying to enter. The walkway itself is a narrow corridor with a single 30° bend in it. The bend in the walkway is inserted to add extra complexity into the environment and to test the collision avoidance between the agents and the corridor walls. Lastly, the walkway leads to the plane itself, which is loosely based on the Boeing 737 Commercial Aircraft. In the very front of the plane is the pilot cockpit, which is sealed off to the passengers. To the right of the walkway are the passenger seats, which are led by three rows of four first class seats. The economy class seats consist of 24 rows of 6 seats, for a total of 144 economy class seats. A single aisle runs down the plane, in which the passengers travel toward their seats and stop to stow luggage. Figure 1 shows the layout of the entire simulation environment, including the terminal, walkway, and airplane.

B. Simulation Director

In order to control the parameters of the simulation, we created a controller that we call the "simulation director". The simulation director handles simulating the airplane boarding process using a set of specified parameters. At the start of the airplane boarding process, the simulation director generates all passengers in the airport lobby. Upon being generated,

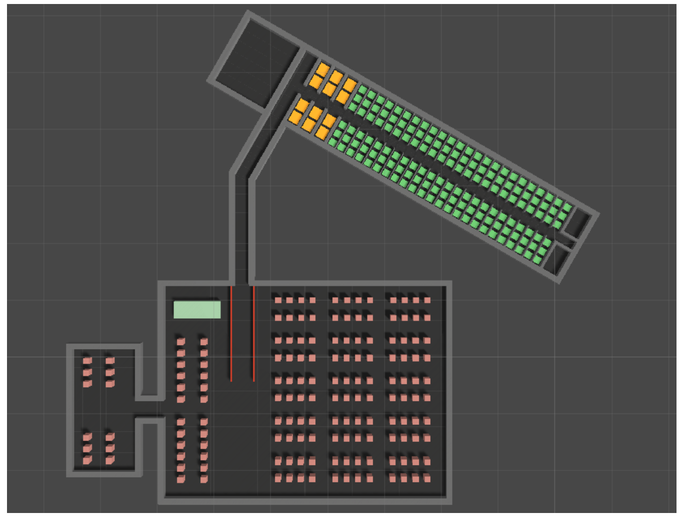


Fig. 1. Layout of the simulation environment. The lobby seats are in red, the first class seats are in yellow, and the economy seats are in green. Starting from the lobby seats, passengers must navigate to their assigned seats in the airplane.

the passengers are assigned a random "goal" seat, which corresponds to a specific seat in the airplane. All passengers must reach the center of their assigned seats for the simulation to end. Passengers are called up to board the plane based on the various simulation parameters. The parameters for the simulation are as follows:

- Boarding Type - Selects which boarding methods to use. The boarding methods implemented are *random*, *front-to-back*, *back-to-front*, *window-middle-aisle*, *Steffen's Perfect*, and *Steffen's Modified*.
- Number of Rows to Board - Applies only to the *front-to-back* and *back-to-front* boarding methods. This parameter selects how many rows to call up at one time.
- Wait Time Per Passenger - Specifies how long to wait between the boarding of each passenger. If multiple passengers are called up at once, the wait time is scaled to how many passengers were called.
- Min and Max Time to Stow - Controls the minimum and maximum time a passenger can spend stowing their luggage on the airplane. The stowage time for each passenger is chosen randomly between the minimum and maximum times. Note: We use a random number generator that samples from a uniform distribution.
- Number of Trials - Specifies the number of times to run the simulation.

Other parameters control small aspects of the simulation, but do not significantly impact the boarding times.

Based on the boarding type and wait time per passenger, the simulation will call up a group of passengers to board the plane at regular intervals. Passengers arrive at the row of their seats, and spend a random amount of time stowing their luggage before going to their seat. A round of simulation ends when all passengers have reached their goals. The time from the first call to when all passengers have reached their goals is recorded. For each boarding method, we recorded times for 20 rounds of simulation.

C. *Passenger Agents*

1) ML-Agents: The first iteration of our passenger agent was implemented using The Unity Machine Learning Agents Toolkit (ML-Agents). ML-Agents allows for the training of intelligent agents using PyTorch and reinforcement learning. In the original environment, we trained one passenger to reach their goal seat, then incrementally added in more passenger agents. However, once we switched environments and seated the passengers in the airport lobby, ML-Agents was not able to train the agents to navigate the narrow walkway. We attempted to add checkpoint goals in the terminal and walkway, but the agents were not able to consistently navigate to their seats. With much longer training times, the agents may be able to learn to board the airplane, however due to time limitations, we began to explore other solutions.

2) NavMesh: We then took some inspiration from the real world: Passengers who are boarding the plane already know approximately where their seats will be. The passengers do not have to learn how to navigate from scratch, but rather we can guide them to their respective seat rows, then have them find their seat. We implemented this idea using Unity's Navigation Mesh System (NavMesh). The NavMesh System consists of three main components [7]:

- NavMesh - Describes the area walkable by the agents.
- NavMesh Agent - The agent itself that navigates the walkable surface to the goal.
- NavMesh Obstacle - A moving (or static) obstacle that the NavMesh Agent will avoid while navigating the environment.

Each passenger in the simulation is given their own NavMesh Agent, which performs the pathfinding and obstacle avoidance necessary to reach their goal. Internally, NavMesh Agents use a path search algorithm called A*. Each NavMesh Agent will also attempt to avoid collisions with each other during navigation. However during luggage stowing, this behavior becomes unreliable as agents will often walk through the stowing passenger. To fix this issue, when a passenger is stowing, we disable the passenger's NavMesh Agent and change the passenger to a NavMesh Obstacle. Since the stowing passenger is now an obstacle, other passengers will avoid the stowing passenger. This behavior is shown in Figure 2.

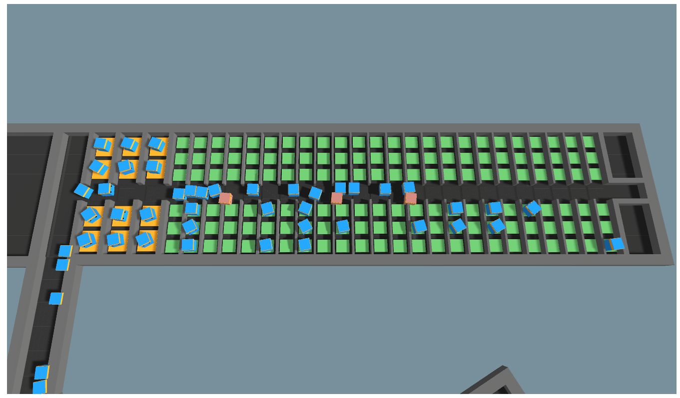


Fig. 2. NavMesh Agents and Obstacles. Passengers are NavMesh Agents (in blue) navigating to their seats. When a passenger reaches their row, the passenger's NavMesh Agent is disabled and replaced with a NavMesh Obstacle (in red) to simulate luggage stowage. Notice that the non-stowing passengers walk around the stowing passengers instead of clipping through them.

3) Queuing: Since airplane aisles are typically very narrow, the luggage stowing process is still not representative of real world scenarios. Although two passengers may squeeze past each other when walking, this is unrealistic during luggage stowage where a passenger will block the entire walkway aisle. In order to simulate this behavior, we implemented a queuing system specifically for luggage stowage. This is done by checking if each passenger's path is blocked (either by a stowing passenger or an already "in-queue" passenger) for each timestep. If a passenger's path is blocked and "blocker" is within a certain threshold range, then the "blockee" will be put into the queue. Being put into the queue includes disabling the NavMesh Agent and enabling a NavMesh Obstacle, similar to a stowing passenger. This behavior will propagate backwards until the last passenger in line, resulting in the formation of a queue. This behavior is shown in Figure 3 below.

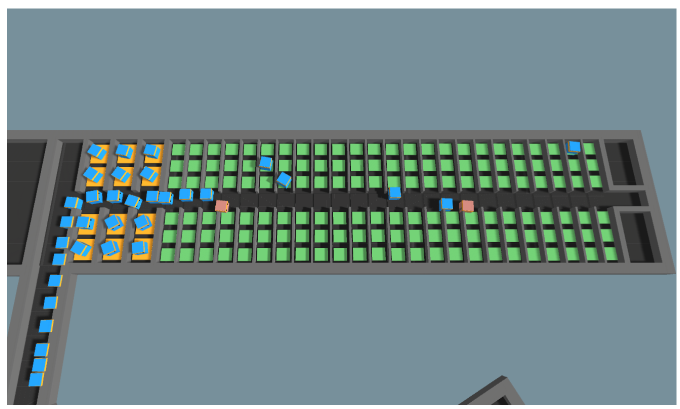
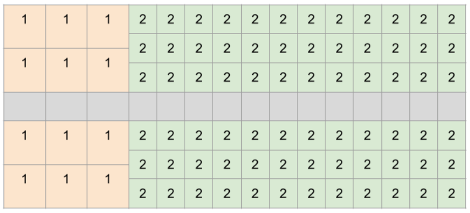


Fig. 3. Queuing behavior during luggage stowage. Instead of pushing through, passengers will now wait behind the stowing passenger. Subsequent passengers will follow closely until within a queuing threshold range, leading to the formation of a queue.

V. Boarding Methods

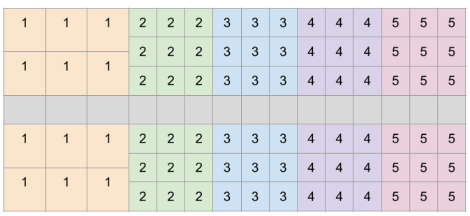
Below we will discuss the various boarding methods and their advantages and drawbacks. In all boarding methods, first class passengers have the highest boarding priority.

A. Random Boarding



In random boarding, all economy class passengers have the same priority. Passengers are called up individually at random to board the plane. The drawback of random boarding in the real world is that it doesn't board passengers that are travelling together at the same time. The random boarding method will serve as the baseline for our simulation.

B. Front-to-Back



In front-to-back boarding, passengers are boarded row-by-row (or block-by-block) from the front of the plane to the back of the plane. Front-to-back boarding is often found to be the slowest method since the passengers in the front of the aircraft tend to block the passengers that need to navigate to the back. The advantage of front-to-back boarding is that it allows passengers that are travelling together to board at the same time.

C. Back-to-Front

In back-to-front boarding, passengers are boarded row-by-row (or block-by-block) from the back of the plane to the front of the plane. Back-to-front retains the previously mentioned advantage of front-to-back boarding, and also reduces luggage stowing obstructions by boarding the back of the plane first. Theoretically, this method allows for parallel luggage stowing as the passengers in the front can reach their row at the same time the back passengers are stowing.

D. Window-Middle-Aisle

In window-middle-aisle boarding (Wilma), passengers who have window seats are boarded first, then those with middle seats, and finally those with aisle seats. The advantage of window-middle-aisle is that it significantly reduces the occurrences where one passenger has to climb over another to reach their seat. However, this method does not attempt to reduce luggage stowing obstructions.

E. Steffen's Perfect

In Steffen's Perfect boarding, passengers board every other row on one side starting from the back. The other side is then boarded in the same manner. Window seats are filled first, then middle, and aisle. The idea behind Steffen's Perfect is to combine the benefits of back-to-front and window-middle-aisle boarding. Passengers are also boarded every other row to further encourage parallel luggage stowing. However, Steffen's Perfect boarding method is quite complicated for passengers to follow. Slow passengers and passengers that are boarding out of order can reduce the efficiency of the method.

F. Steffen's Modified

In Steffen's Modified boarding, passengers with left-side even seat numbers are boarded, then right-side even seat numbers, then left-side odd seat numbers, and finally right-side odd seat numbers. Steffen's Modified simplifies Steffen's Perfect by removing the window-middle-aisle and back-to-front aspects of the original method, and boarding only in alternating rows and sides. This method also has the added benefit of allowing passengers who are travelling together to board at the same time.

VI. RESULTS

Each boarding method was simulated 20 times. To make the simulations run faster, we simulated at time scale 10. For each round of simulation, we record the elapsed time (in seconds) from the first call to when all passengers have reached their goals. Figure 4 summarizes the results.

Boarding Method	Mean	Standard Deviation
Random	1359.852	76.562
Front-to-Back	2263.430	123.521
Back-to-Front	2239.238	106.137
Window-Middle-Aisle	1169.300	47.175
Steffen's Perfect	927.285	29.690
Steffen's Modified	1149.634	51.879

Fig. 4. Table summarizing the statistical results of simulation (time in seconds).

Additionally, we performed pairwise t-tests to determine if there exists statistically significant differences between the mean times of each boarding method (see Figure 8). Almost all pairs obtained low p-values (≤ 0.05), so we claim that, for these pairs, their means are a valid base for comparison. The two pairs whose p-values did not make the cut were (front-to-back, back-to-front) and (window-middle-aisle, Steffen's Modified), both of which had p-values equal to 0.68.

VII. DISCUSSION

While many of our results (such as the superior performance of Steffen's Perfect boarding) are in line with theoretical expectations, there are some surprising things to note regarding some of the other methods. Most notably, using the random boarding method as our baseline, we find that two of the simulated methods–front-to-back and back-to-front–actually performed worse than if passengers were simply boarded at random. Ironically, the back-to-front method is likely the boarding method most commonly encountered in the real world, despite our results suggesting that it is roughly tied for worst boarding method out of the ones we tested, given that there was no significant statistical difference between the mean boarding times of back-to-front and front-to-back.

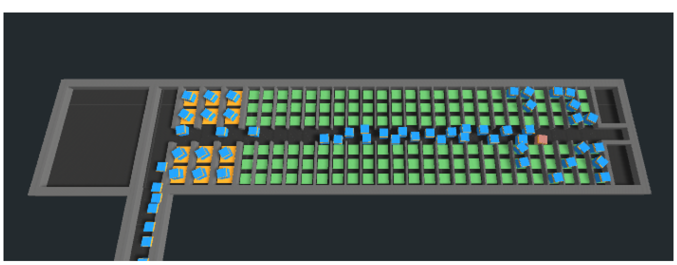


Fig. 5. A single passenger holds up the entire line while stowing during a back-to-front simulation. The second boarding group has been called and is seen entering the plane, but the line from the first boarding group is still backed up nearly to the front of the plane, preventing the second boarding group from stowing their luggage and reaping the expected benefits of parallel stowing alongside the first boarding group.

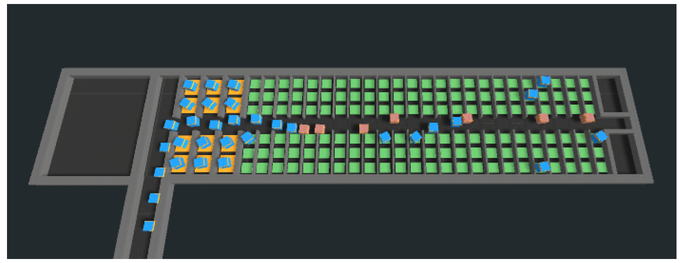


Fig. 6. Multiple passengers are able to stow their luggage at once in this instance of random boarding simulation. While the occurrence of this many parallel stows is admittedly rare, random boarding does lend itself to an increased chance of parallel stowing compared to back-to-front or front-to-back. Cases of 3-4 passengers stowing at once is commonly observed throughout the course of the simulation.

Looking at the simulation, we can see some possible explanations for these results. Despite intuition telling us that back-to-front should avoid stowing obstructions as passengers at the back board and stow their luggage first, the simulation reveals that there is actually significant obstruction and minimal parallel stowing occurring, even from very early on in the simulation. This is illustrated in Figure 5. Due to every passenger trying to reach a seat located in roughly the same portion of the plane, it turns out that there is actually an increased chance of one passenger obstructing another and thus increasing the boarding time. On the other hand, Figure 6 shows how the random boarding method lends itself to increased opportunities for high levels of parallel stowing, as the passengers' seats are spread out throughout the length of the plane. In lucky cases, this can sometimes lead to several passengers (7 or more, even) stowing their luggage at the same time, by chance spread out perfectly to utilize the entire length of the plane.

Another interesting result is that window-middle-aisle boarding performs slightly better than random boarding. In our simulation, window-middle-aisle works similarly to random in that passengers are randomly distributed along the length of the airplane. However, window-middle-aisle boards in layers, starting from the windows towards the aisle. Perhaps the performance boost comes from "flattening" the distribution of randomness. For instance, in random boarding, up to 6 passengers can be waiting to board a given row. In window-middle-aisle boarding, this is limited to 2 passengers. In a sense, window-middle-aisle boarding more uniformly spreads the distribution of boarding passengers than random boarding does. Thus it prevents passengers from targeting one specific area to board, which we previously saw (in the cases of back-to-front and front-to-back) caused an increased number of stowage obstructions.

Observing Steffen's Perfect boarding, we can see that there are consistently high numbers of parallel stowing passengers throughout the simulation. Given the high impact of stowing obstructions on boarding time that were seen in other simulations such as back-to-front, it follows that Steffen's Perfect ended up being the optimal method according to

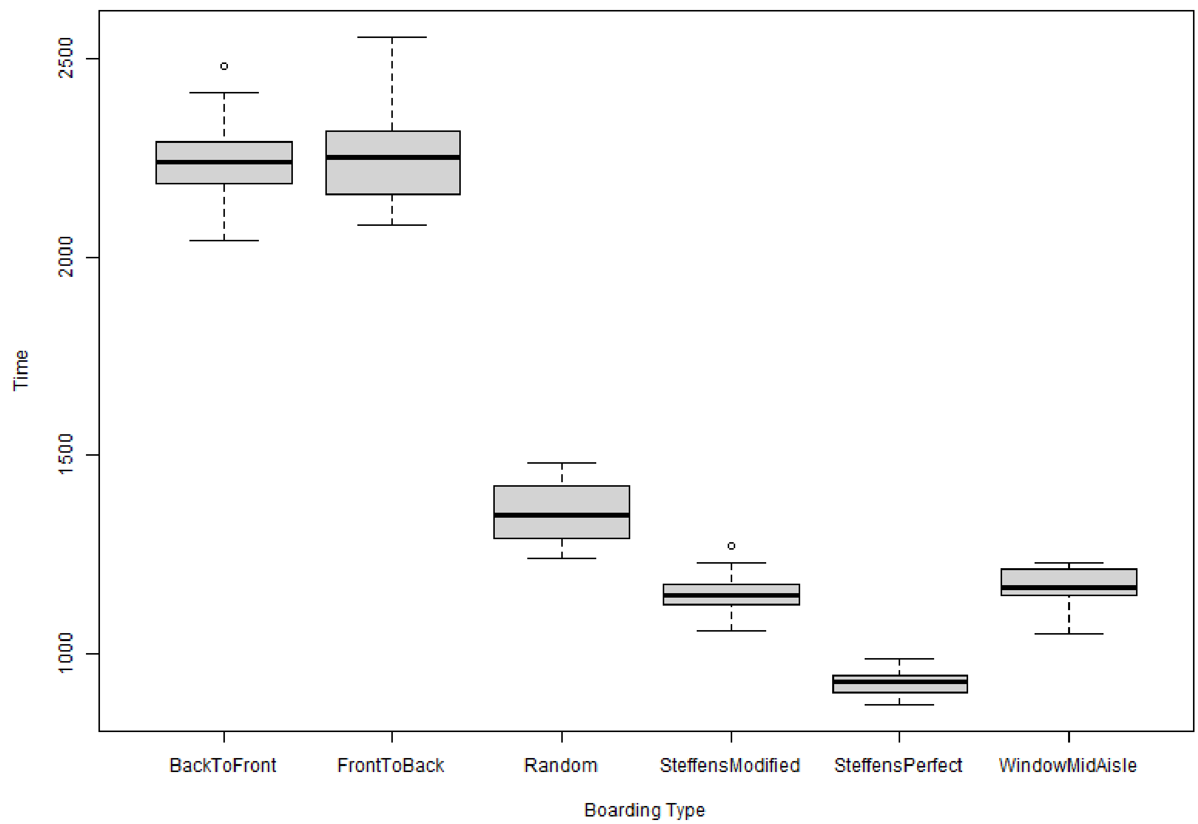


Fig. 7. Boxplot of the recorded times (time in seconds).

	Back-to-Front	Front-to-Back	Random	Steffen's Modified	Steffen's Perfect
Front-to-Back	0.68	-	-	-	-
Random	<2e-16	<2e-16	-	-	-
Steffen's Modified	<2e-16	<2e-16	8.0e-13	-	-
Steffen's Perfect	<2e-16	<2e-16	<2e-16	7.9e-14	-
Window-Middle-Aisle	<2e-16	<2e-16	3.4e-11	0.68	1.5e-15

Fig. 8. Pairwise t-tests between each of the boarding methods.

our simulation. Since our simulation is designed to more accurately model real life, the order of Steffen's Perfect ends up not being completely perfect because passengers are called up to board from various random locations in the gate lobby. Because of this, occasionally we can see that some passengers who are called later actually reach the plane first because they were seated closer to the gate. Despite this, Steffen's Perfect was still able to perform the best out of all the methods.

VIII. Conclusions

Given the prevalence of air travel and the high levels of stress and impatience that can be triggered by inefficient boarding processes, understanding the hallmarks of a good boarding method could do wonders in smoothing out the ordeal for everyone involved. To this end, we implemented physical simulations for six different potential boarding methods using Unity's NavMesh system. Just as in real life, passengers stop to stow their luggage before entering their seat, and passengers behind them stop and form a queue until the stowing is complete and the line can proceed forward.

From our simulation trials, we find that the optimal method, as expected, is the Steffen's Perfect method, followed by Steffen's Modified and window-middle-aisle statistically tied for second place. Our baseline method, random boarding, came in third, while back-to-front and front-to-back took last place. While each method has some benefits and drawbacks associated with them, there were notable differences in boarding times between most of these methods, mostly due to the differing probability of boarding obstruction due to luggage stowage.

By watching the simulations progress for each of these different methods, we are able to visually see and compare the number of obstructions or, conversely, the level of parallel stowing that occurs during boarding. This allows us to better understand why some methods perform better than others. We can visualize what kind of events and situations are encountered as the boarding progresses–observations that are lacking in, for example, purely numerical simulations.

This simulation can also be easily extended in future works. One interesting addition would be to introduce a range of different agent personalities. For instance, some passengers might tend to run late or line up at the gate early, or perhaps have a tendency to push past other passengers or let others pass first before stowing their luggage in order to prevent blockage. Adding these varying qualities would be valuable in better simulating a real life situation, where passengers encounter others with all different kinds of quirks.

References

[1] H. Yeh, S. Curtis, S. Patil, J. van den Berg, D. Manocha, and M. Lin, "Composite agents," *SCA*, Jan. 2008.

[2] S. Guy, S. Kim, M. Lin, and D. Manocha, "Simulating heterogeneous crowd behaviors using personality trait theory.," Aug. 2011, pp. 43–52. DOI: 10.1145/2019406.2019413.

[3] S. Patil, J. van den Berg, S. Curtis, M. Lin, and D. Manocha, "Directing crowd simulations using navigation fields," *IEEE transactions on visualization and computer graphics*, vol. 17, pp. 244–54, Mar. 2011. DOI: 10.1109/TVCG.2010.33.

[4] S. Iyigunlu, C. Fookes, and P. Yarlagadda, "Agent-based modelling of aircraft boarding methods," Jul. 2014. DOI: 10.5220/0005033601480154.

[5] S. Mas, A. Juan, P. Arias, and P. Fonseca i Casas, "A simulation study regarding different aircraft boarding strategies," Jun. 2013, ISBN: 978-3-642-38278-9. DOI: 10.1007/978-3-642-38279-6_16.

[6] J. H. Steffen, "Optimal boarding method for airline passengers," *Journal of Air Transport Management*, vol. 14, no. 3, pp. 146–150, May 2008, ISSN: 0969-6997. DOI: 10.1016/j.jairtraman.2008.03.003. [Online]. Available: http://dx.doi.org/10.1016/j.jairtraman.2008.03.003.

[7] A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, and D. Lange, *Unity: A general platform for intelligent agents*, 2020. arXiv: 1809.02627 [cs.LG].