6th OCCAM Graduate Modelling Camp

Efficient airline boarding strategies on the Airbus A380

Authors:
Ana Avdzhieva (Sofia University)
Angelika Manhart (University of Vienna)
Armin Krupp (University of Oxford)
Hendrik Brackmann (University of Oxford)
James Mathews (University of Cambridge)
Sam Kamperis (Oxford Brookes University)

Problem presented by:
Prof. Nilima Nigam (Simon Fraser University)

Abstract

In this report we examine and find boarding strategies for a two-aisle model on the Airbus 380. Results for this type of airplanes are largely missing from the literature. A powerful MATLAB simulation tool was built in which many relevant parameters such as loading time, walking speed, etc. can be included as deterministic or random variables. We used this tool to test existing strategies for efficiency and robustness and to obtain insight into the effect of different variables on the total boarding time. Additionally we apply a Genetic Algorithm and obtain first results hinting that also existing boarding strategies can be further improved. Lastly we obtain combinatoric results for certain asymptotic regimes.

April 27, 2014

Contents

1 Introduction

The aircraft Airbus 380 has two plane-length decks and both of them have two aisles. The number of seats in the plane are between 850 and 1000. Even for smaller planes the boarding of the passengers is typically the most time consuming event while the plane is not flying. Many people have experienced themselves how a tiresome boarding also affects their satisfaction with the airline.

Figure 1: Airbus A380 interior

Our task is to propose a way of boarding the plane that is both realistic and quick. Unrealistic ways of boarding the plane would be to tell each passenger when they are allowed to board the plane and not bothering with assigning seats (since they are needed for things such as meals). By reducing the boarding time it reduces the costs for airlines, increases customer satisfaction and also would increase airport capacity. The way we board the plane will be referred to as the boarding strategy.

Definition 1 (Boarding strategy)**.** *We define a boarding strategy to be how passengers with specific assigned seats are assigned to a boarding group.*

Examples of existing boarding strategies include the back to front strategy (where for example we load the back five rows of the plane first, then the five rows in front of those, and so on), the front to back strategy (the same but starting from the front), the Wilma strategy (loading people sitting in the window seats first, then the middle seats, then the aisle seats) and the Steffen strategy (the same as Wilma but we do every other window seat first, then we fill the remaining window seats, then every other middle seat and so on). These strategies are detailed more in Section 4, and in various literature, such as [2], [3] [5] and [6].

1.1 Prior work

We started by researching the problem and looked at the articles detailed in the appendix. The common theme of the literature is that it has always involves a single aisle, and there has been no mention of what happens with two aisles.

In [1], the author uses Monte Carlo simulations of an agent based model to calculate the optimal boarding strategy, which he calls the Steffen strategy (see Section 4). He goes on to compare it with previous strategies and deduce it is the best strategy based on his input variables to the problem. In [2] the same author goes on to test his strategy and others experimentally, and again concludes that the Steffen strategy is again the best. The nature of his experiments is discussed in Section 4.1, and due to privacy concerns about filming on aircraft, it remains one of the few experiments done.

In [3] the authors set up the problem as an non linear assignment problem, which has quadratic and cubic terms in the objective function, and thus is an NP hard problem. This assignment is solved using a mixed integer non-linearly constrained optimization solver, and then they compare boarding strategies. They find for their model that in fact the Wilma strategy (see Section 4) optimises the boarding time, and they recommend it should be implemented on America West flights. Implemented this on the America West flights showed a time saving compared to the traditional back to front boarding method.

An alternative approach based on queuing theory is discussed in [4], although the results are only applicable to a primitive model of a single aisle with a single row per seat. We discuss more about this approach in Section 2.

This is not the first time the problem of aircraft boarding has been considered in a study group. Two previous reports from a Duke study group are listed in the references ([5] and [6]). Although some of the ideas are novel, such as using a genetic algorithm, they only focus on the one aisle problem.

1.2 Outline of work

Based on our research of the problem, we set out to do a number of things. We began by making some basic modelling decision about the problem, such as we would use an agent based model and various other simplifications to the deck of the Airbus, which are described in Section 1.3. Other models such as fluid based model were quickly dismissed since we felt they were not applicable since the passengers were not dense enough.

We briefly investigated a combinatoric approach in Section 2, which was for the very simplified case of one aisle and one row per aisle and gave some results for certain asymptotic regimes. This approach also used the basics of queuing theory.

We then made some brief real life experiments and determined and measured variables which would be inputs for our model, such as the approximate walking speed, loading time and loading space.

The other input into our model would be a boarding strategy, which we have already defined. Having decided it was necessary to built a simulation tool so that we could test strategies, we implemented this tool in MATLAB which is described in Section 3. We then used this tool to test existing strategies, and examine their robustness to randomness in Section 4. At the same time, we also developed a Genetic algorithm optimizer in Section 5 which used this simulation tool to find the optimum strategy.

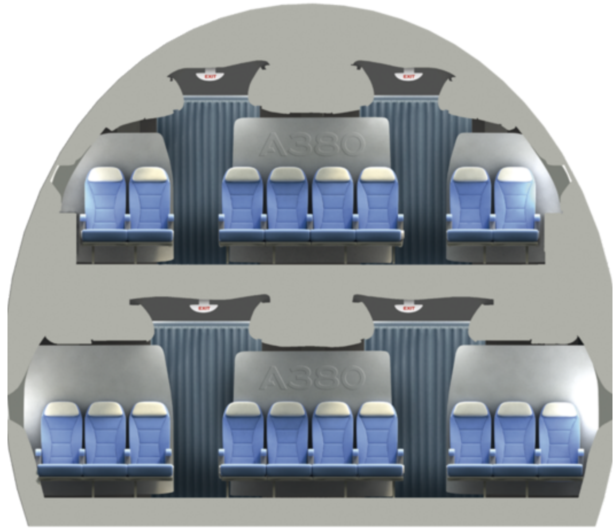


Figure 2: Seating configuration of upper and lower deck from `http://www.airbus.com/typo3temp/pics/5528c3a5d7.png`

1.3 What is different about the Airbus A380?

The Airbus has two decks, with the lower deck seating 538 passengers and the upper deck seating 315. We also have two aisles on both decks, with the stairs locating near the aircraft doors. Although the Airbus A380 has more than one boarding door, unfortunately most airports only have single-door boarding capacity, and due to the length of boarding bridge passengers board from the front.

We can effectively think of the two decks as separate problems, so we only consider the lower deck from now on since it has more passengers. We will assume it has 54 rows of 10 seats in each aisle to simplify matters slightly, and the seating configuration is four central seats and three seats either side of the aisle, as can be seen in Figure 2.

Three differences between a two aisle problem and a one aisle problem are summarised below:

- Even if we divide the two aisle problem into two one aisle problems, we end up with an unsymmetrical one aisle problem with two seats on one side and three seats on the other. This case has not been considered in any literature so far.
- Passengers in the middle aisle can walk down either aisle, either accidentally (such as families grouping together or people not following instructions) or deliberate (since it might be quicker of easier to send passengers down one of the aisles to the other).
- The final difference is whether to fill the middle seats or the outer seats first, and whether this makes any difference.

1.4 Optimal strategies and robustness

Given the variables (see Section 3.1), we can (in theory) create an **optimal boarding strategy**, which minimises the total boarding time, and we explain how to do this in Section 5 using the genetic algorithm. Once we have an optimal boarding strategy we can test it for **robustness**, and see what happens to the total boarding time when

- we change the order within boarding group
- we have passengers being in the wrong boarding group (families sticking together, people not paying attention to when they are told to board)
- passengers go down the 'wrong' aisle to get to their seat

If our optimal strategy is robust, then it will behave "well" under these occurrences, but defining "well" is a hard question, but essentially we want a strategy that has a low expected boarding time and small variance. In this report we test for robustness using the first case, but given more time we would like to investigate the robustness in the other two cases. We note that the third test of robustness is unique to the two aisle problem, and for now we only assume that it is people in the middle row that might go down the wrong aisle, while the people at the two ends always go down the correct aisle.

2 Combinatorial analysis

2.1 Definition of the model

To understand the behaviour of different boarding strategies, we developed a simple agent based combinatorial model. We assume that all agents in the system behave identical and deterministic. Their behaviour is determined by time they need to advance one row and the time they need to store their luggage. With this simplification, we can understand how the order of the passengers entering the plane affects the expected boarding time. The model is as follows: The corresponding

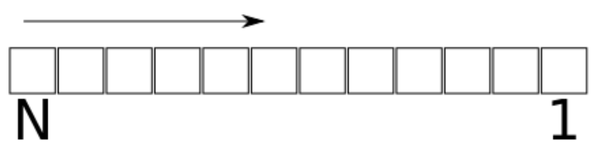


Figure 3: Include single aisle model here

aircraft has only one aisle and only one seat per row. The number of rows is N. The passengers enter the plane from the left to the right, where the seats are numbered from N to 1 from left to right. All passengers behave uniformly and it takes them a time t_a to advance from one row to the other. Also, the loading of their hand-luggage takes them a time t_s. Furthermore, we assume that the distance between any two passengers has to be at least the distance between to consecutive rows and that a single row is sufficient to put the luggage into the overhead lockers.

Since we only have one seat per row, every passenger is uniquely identifiable by its row. Thus, we can use assignments of rows to boarding groups as boarding strategies. We assume that the order of passengers within a boarding group is random.

From this we can infer what the optimal boarding strategy looks like: We take groups of size 1 and let the passenger sitting in row 1 (let us call him passenger 1) enter first, then passenger 2 and so forth. The total boarding time T then amounts to

$$T = N \cdot t_a + t_s. \tag{1}$$

Naturally, we will not always be able to have boarding groups of size 1. Let us therefore analyse two different strategies which deal with boarding groups of size at least two.

2.2 Back-to-front

We can use the optimal strategy to derive the more general back-to-front strategy for boarding groups of size g: The first boarding group consists of passengers $1, ..., g$ and so on, i.e. the k-th boarding group consists of passengers $(k-1)g+1, ...kg$.

For a permutation $\sigma : \{1, ..., g\} \to \{1, ..., g\}$ let us define its **defect** $d(\sigma)$ by

$$d(\sigma) = \max_{i=1,...g} |i - \sigma(i)|. \tag{2}$$

The defect denotes the maximum between the actual position of the passengers within the boarding group and their position if they were in perfect order (i.e. in increasing order). The defect takes values $d = 0, ..., g-1$.

Now consider a specific instance of a back-to-front boarding strategy with boarding group size $g > 1$. Assume that the highest defect among all boarding groups is k. Then the queue will halt for the first time when the person corresponding to that defect is at its place. However, all passengers behind him who are also having a defect of size k, will be able to take their seat as well. Thus, the maximum defect among all groups will be $k-1$ afterwards. In the worst case, we will have first a defect of size g, then $g-1$ and so on. This corresponds to a total boarding time of

$$T = N \cdot t_a + g \cdot t_s. \tag{3}$$

If we keep g fixed and let $N \to \infty$, then this is going to happen with probability 1.

2.3 Back-to-front with gaps

A possible way to deal with these defects is to leave gaps of size l, that is to let passengers $1, ..., g$, then $g+l+1, ..., 2 \cdot g+l$ and so forth enter the plane in the first batch and the remaining passengers in the second batch. If we choose l to be at least $g-1$, the defects of one group do no longer affect the boarding groups behind. To compute the expected boarding time in this instant, we first have to compute the expected time it takes a single boarding group to get seated. We can calculate this time numerically, let us denote it by $S(g)$. Then, the expected boarding time for all passengers is going to be

$$T = \left(\frac{g}{g+l}\right) N \cdot t_a + S(g) + N \cdot t_a + S(l). \tag{4}$$

This is usually a lot slower than the back-to-front strategy without gaps.

3 Description of simulation tool

To gain a better understanding the determinants of the boarding time, a simulation of the boarding procedure was implemented. This section gives a detailed description of the assumed model.

3.1 Model of the passengers

We assume that there are N passengers, each described by the following four variables

- **Seat.** The seat on which the passenger is supposed to sit.
- **Walking speed.** The speed a passenger would walk in the aisle assuming that there are not obstacles.
- **Stopping distance.** The minimum distance left between this passenger and the upfront passenger. This might be interpreted as the physical space a passenger is occupying.
- **Loading space.** The space which is occupied by a passenger while he is loading.
- **Sitting time** The time it takes to put away luggage and sit down. This has two factors, the first one being the time to put away luggage and will vary for each passenger. The second component, how long it takes to sit down will only depend on the configuration of the passengers sitting down and in the way.

All passengers are initialized by a call to the corresponding *get_* function. This allows users to easily adjust the simulation.

3.2 Model of the plane

The plane is assumed to have M rows and SR seats per row, such that $M \cdot SR = N$. Further we assume there are two aisles, which are assumed to be continuous. The seats are assumed to be at equidistant points in the aisle, starting at zero. The length of the aisle is assumed to be long enough such that the passenger at the back has always enough loading space. The *seating table* keeps track of the seats which have already been occupied. The time it takes to sit down once a passenger arrived at his seat is given by a customizable function of the *seating table*.

3.3 Model of the the queue

The queue consists of all variables which change during time. There is one queue for every aisle and it is initialized by the passengers sorted in an order which is determined by a call to the *strategy* function, which represents the boarding strategy. For each passengers it keeps track of three variables

- **Status.** Determines whether the passenger has not entered the plane (NI), is walking (WA), is loading his bags and about to sit down (LO) or is already sitting (SI).
- **Position.** Determines the position of the passenger within the aisle.
- **Waiting time.** Determines the amount of time left for a passenger who is loading until he is seated.

3.4 Description of logic

For the simulation, we only consider discrete time steps with distance Δt. At each time step, we update the queue by updating the variables for each passenger depending on his state

SI A passenger in this state is not updated.

LO The waiting time of this passenger is decreased by Δt. If his waiting time is negative, his status is set to SI and the seating table is updated.

WA The passenger walks Δt * *walking speed*. His position is updated in a way which ensures he keeps *stopping distance* from the next passenger. Once the passenger reaches his seat, it is checked whether he has enough *loading space*. If that is the case he is going to occupy this space in a way which ensures that he uses most space towards the back of the plane. His waiting time is updated accordingly and his status is set to LO.

NI If there is space in the aisle, the passenger is set to WA. In case the passenger is sitting in the first row, he immediately starts to unload.

The simulation then consists of iterating over time steps until all passengers are seated.

3.5 Usage

The simulation can be customized by using different implementations of the functions *get_loadingspace*, *get_seatingtime*, *get_walkingspeed*,*get_stoppingdist* and *strategy*. Using implementations which include stochastic effects and averaging over simulation outcomes approximates expected boarding times in these cases.

3.6 Visualisation

A video showing a random boarding strategy of an early version of the code for one aisle is available to view at `https://www.youtube.com/watch?v=Nf04c4iucLQ`. A screenshot from the visualisation tool is shown in Figure 4. We hope to produce a similar visualisation for the two aisle problem with the final version of the simulation tool.

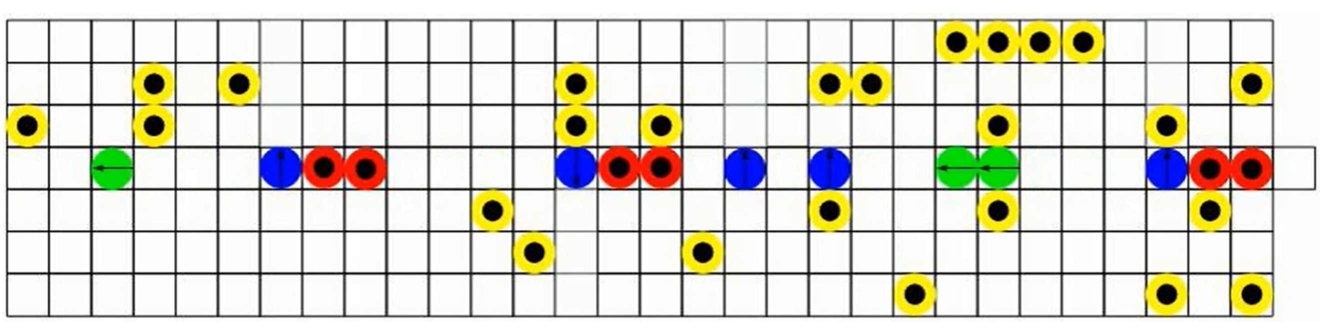
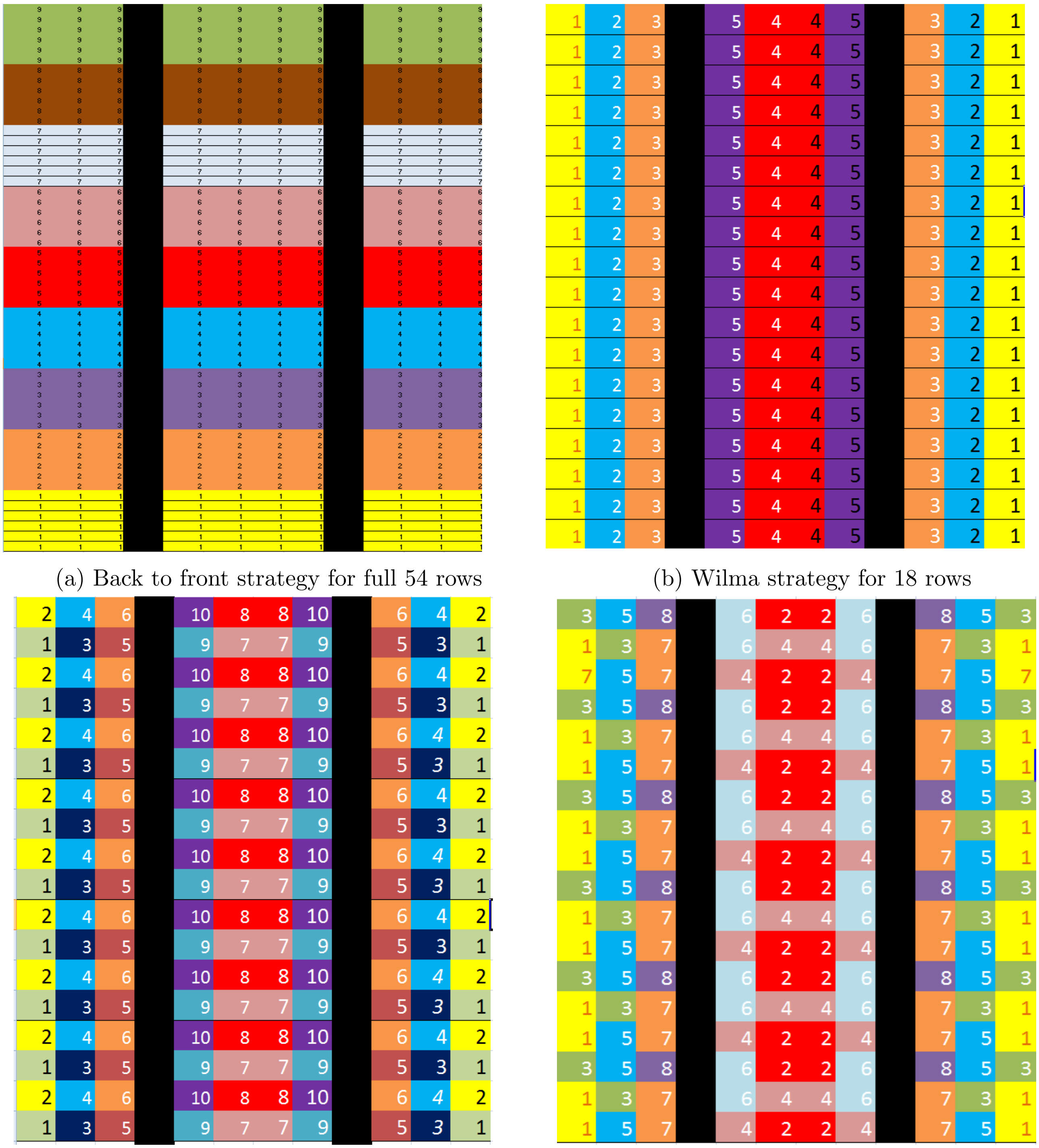


Figure 4: This picture taken from the visualisation video shows a one aisle model, where passengers enter from the right. Walking passengers are shown in green, those loading their luggage in blue, those having to wait in red and finally the ones that have already been seating in yellow.

4 Example and testing strategies



(a) Back to front strategy for full 54 rows

(b) Wilma strategy for 18 rows

(c) Steffen strategy for 18 rows

(d) James strategy for 18 rows

Figure 5: Illustration of different boarding strategies. The numbers (and different colours) refer to the different boarding groups, with boarding group 1 first to board the plane, followed by boarding group 2, all the way up to the last boarding group

Having built a simulation tool, we now want to test some existing strategies to see how they perform. In Figure 5 we recap what some of the existing strategies are, and we also include a new strategy, denoted James. Our brief experiment in de terming parameters suggested the loading space should be 1.5 rows, and the James strategy is designed to take this into account. We have only shown the Wilma, Steffen and James strategies for 18 seats, but it is obvious how to extend them to 54 seats. The four boarding strategies we have shown range from having five boarding groups to ten boarding groups, which would certainty not be unrealistic on an Airbus A380.

We fix the parameters to be the same deterministic value for each passenger, which are given by Table 1, which are based on our brief experiment and literature such as [5].

Walking speed	0.5 m per sec
Stopping distance	0.4 m
Loading space	1.2 m (1.5 rows)
Sitting time	3s plus 5s per passenger that has to move out the way

Table 1: Parameter values, note that each row is 0.8m

We now test these strategies, as well as the random strategy, to the robustness of the order of the boarding groups, doing five hundred enumerations for each strategy.

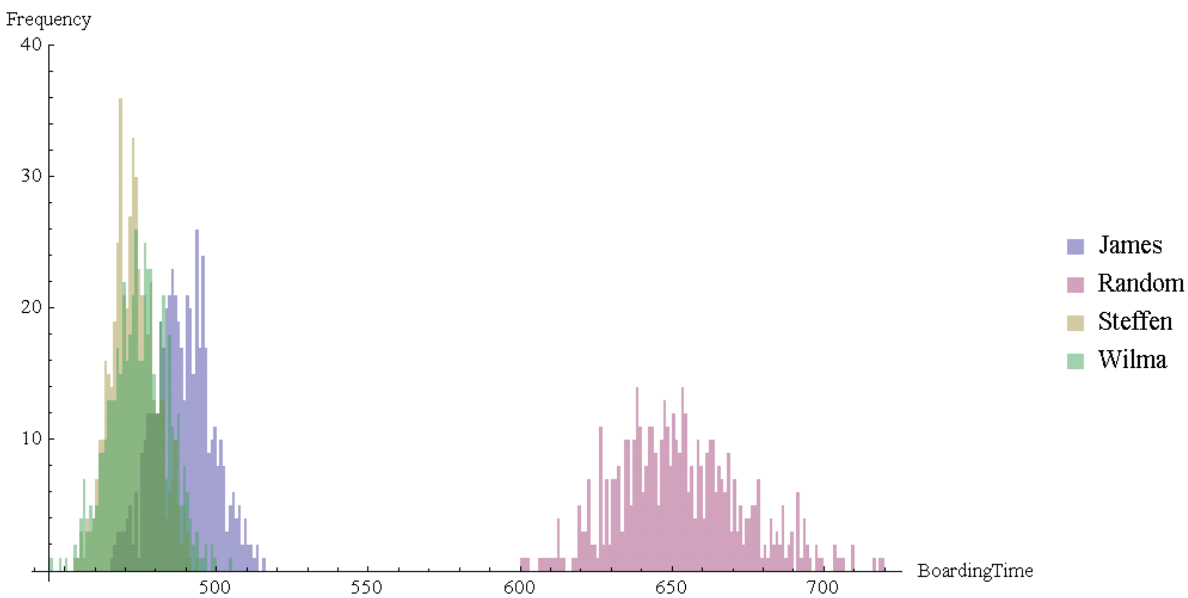


Figure 6: Histogram showing simulated boarding times of three strategies and random boarding

In Figure 6 we see that the methods James, Wilma and Steffen all board faster than complete randomness. There is also less variance in the boarding times. Steffen appears to have the most robust and optimal performance. This method also had the largest number of distinct boarding groups. The passenger boarding order would only randomize inside these groups and if there are more boarding groups, less randomisation will occur.

In Figure 7 we have the same histogram but with the results of the back to front strategy plotted. This is clearly performing worse than complete random boarding in the simulation. The number of boarding groups used was nine (Steffen implemented with ten). The cause of this is most likely aisles being blocked with passengers waiting to get to the back of the plane and no passengers passing more forward seats earlier being seated.

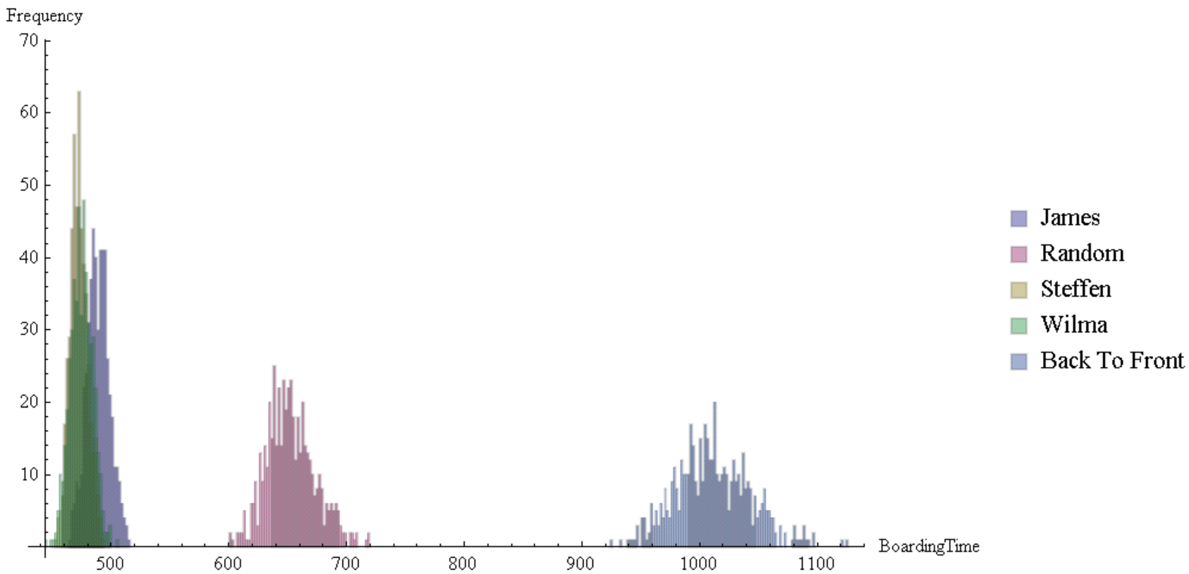


Figure 7: Histogram showing simulated boarding times of four strategies and random boarding

4.1 Lack of Experimental Data

In [2] an experimental comparison of various boarding methods was performed. This was conducted in a mock Boeing 757 fuselage at the Air Hollywood Soundstage in Southern California. This consisted of 12 rows of six seats with a single aisle. The 72 passengers were volunteers and Hollywood extras. The paper states "Each passenger was given a set of five tickets with seat assignments and passenger order/boarding group". An argument is made that "fatigue with boarding the aircraft several times could mitigate the benefit of practice". However, this was put forward with regards to boarding with the same luggage. A similar issue could be raised with regards to boarding the same seating arrangement several concurrent times. If the passengers are aware of the seat numbering when boarding then it could possibly speed up the process. There is also no consideration for passengers boarding as a family which would necessarily require that some small groups of passengers in adjacent seats would board together. This is something that would result in significant changes to the order of boarding for methods like Wilma and particularly Steffen, possibly leading to some of their efficiency being lost.

4.2 Effect of parameters

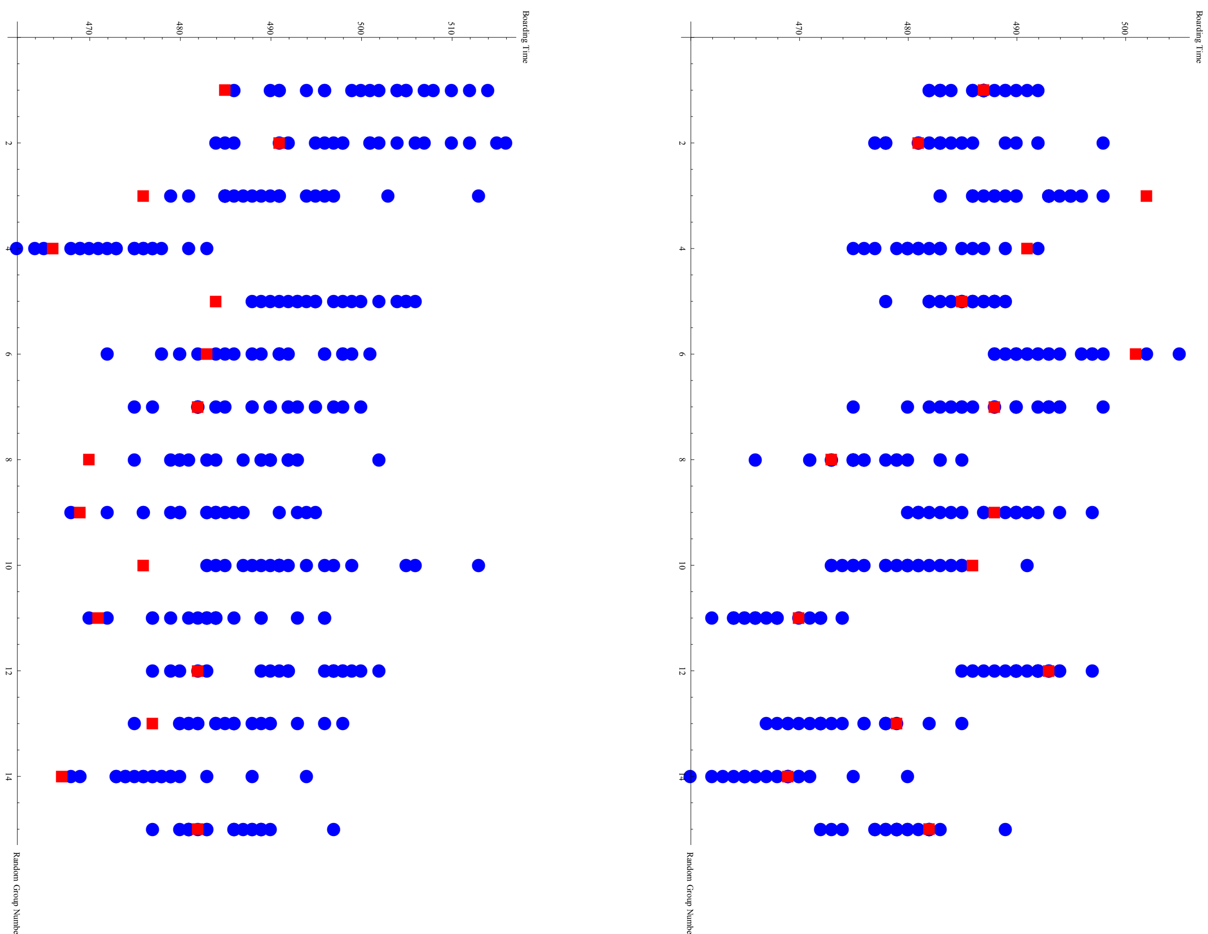
We choose the Wilma strategy with 5 boarding groups (see Figure 5b) and see what happens as the parameters varies stochastically for each passenger. We assume that all the other parameters are fixed (deterministically). We assume that all the passengers follow the instructions, but we allow the order for each group to vary.

For each iteration, we randomise the order of the group, fix it and then perform 20 iterations with the stochastic parameter. We also perform one iteration with this parameter fixed to the usual deterministic value for comparison. We then change the order of the groups and repeat again fifteen times.

We choose values that have been studied in literature and from our own (brief) experiment. Distributions from variables come from literature. Table 2 summarises the stochastic distributions in comparison to the previous deterministic values. In what follows $N(a, b)$ is a normal distribution with mean a and standard deviation b, $U(c,d)$ is a uniform distribution on (c,d) and $\exp[f]$ is an exponential distribution with rate parameter f.

Parameter	Deterministic	Stochastic
Walking speed	0.5 m per sec	$\max(0.1, N(0.5, 0.15))$ m per sec
Stopping distance	0.5 rows	$U[0.125, 1.75]$ rows
Loading space	1.5 rows	$U[1, 2]$ rows
Sitting time	3s plus 5s per passenger that has to move out the way	$\exp[3]$s plus 5s per passenger that has to move out the way

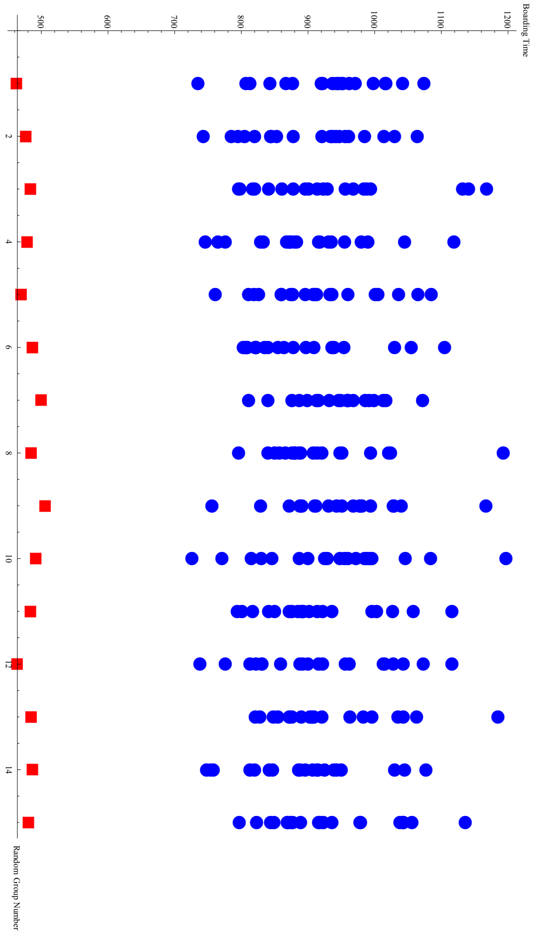
Table 2: Stochastic parameter values, note that each row is 0.8m



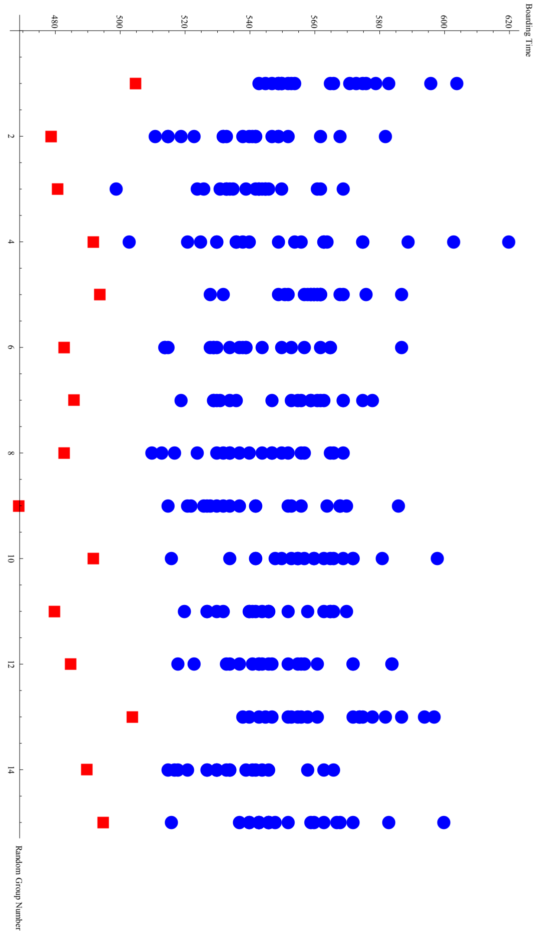
(a) Effect of the **stopping distance** (b) Effect of the **loading space**

Figure 8: We generate 15 random orders for the passengers, which are on the y axis. We then plot on the x axis in red the total seating time with the deterministic value, and in blue twenty runs with stochastic values

From Figure 8 we see that adding stochastic variations to the stopping distance and loading space does not have too much of an effect of the total seating time, since in some cases it increases the



(a) Effect of the **walking speed**



(b) Effect of the **seating time**

Figure 9: We generate 15 random orders for the passengers, which are on the y axis. We then plot on the x axis in red the total seating time with the deterministic value, and in blue twenty runs with stochastic values

seating time and in some cases it decreases the seating time. However, in Figure 8 we see that adding stochastic variations to the walking speed and sitting time has a dramatic effect on the seating time. In all cases we plotted the seating time went up, and in the case of the walking speed went up by a significant amount. This is because as soon as we have a slow person (either slow at walking or slow at storing their luggage) they will hold up everyone else in the aisle, who will be unable to do anything.

5 Genetic Algorithm

A genetic algorithm is an optimization method based on the idea of natural selection. In nature a population of organisms can evolve, i.e. become better adapted to their environment, through rounds of selection and subsequent changing of the genetic material. Selection in this context means that only the *fittest* individuals of each generation survive and are able to produce offspring. The genetic material of the offspring is created by small changes, *mutations*, of the genetic materials of their parents and by possible mixing, *cross-overs*, of the genetic material of two parents. In nature this means, given a long time and many generations, the population will increase its fitness. A very simple example of this process would be population of bacteria in a petri-dish filled with a media containing small amounts of an antibiotic. As the generation time is low and mutation rate quite high, one can observe within a short time that the bacteria population will develop a resistance to the antibiotic, i.e. will become more fit.

5.1 How do we use the Genetic Algorithm

The simulation tool presented above has the capability if test existing strategies for efficiency, robustness, etc. However it can also be used to find new boarding strategies when combined with the GA. We translate the above biological terms to our problem

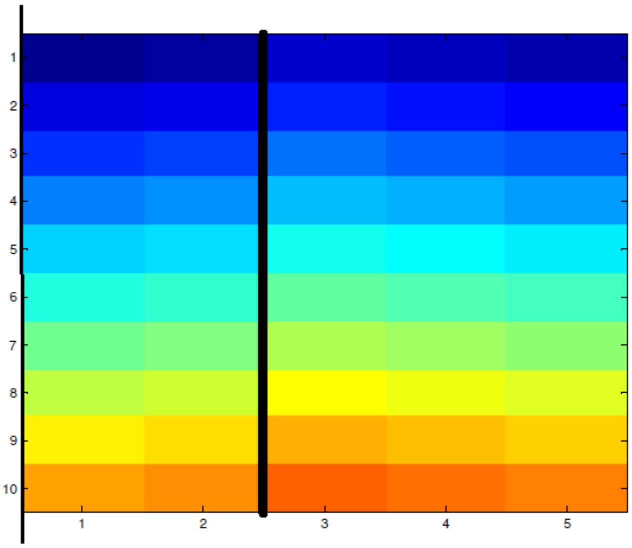
- **Genomes:** Where before an individual was described by its DNA sequence, we will now identify each boarding strategy with an individual. At this point we searched for an ideal dictator strategy strategy (i.e. dictating the exact order in which passengers enter). However the algorithm could easily be adapted to search for boarding strategies with larger boarding groups. Also it is reasonable to assume that, having found a good dictatorship strategy, the corresponding strategy with boarding groups would be a good candidate for a boarding strategy with boarding groups.

- **Fitness:** We need to define a fitness function which we want to maximize. In our case we can use the simulation tool which outputs the total boarding time for a given boarding strategy. Therefore the smaller this time is, the fitter our individual. Note that for boarding groups one could also use other objective functions, for example include the variance in boarding time or the average waiting time per passenger.

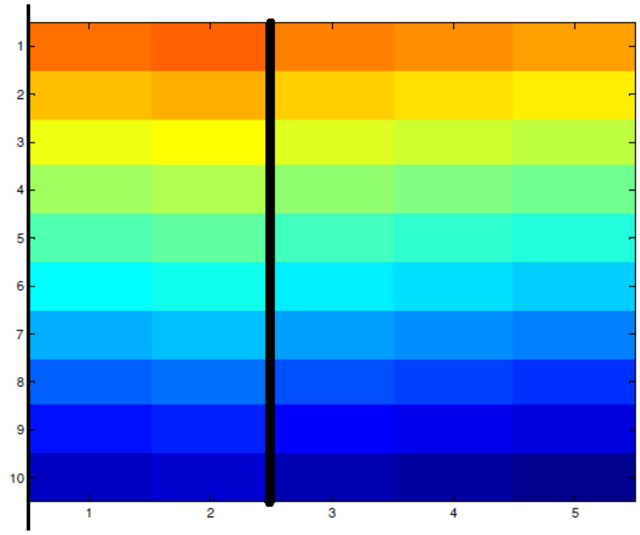
- **Mutation**: Mutations in our case means to change the order of two or more passengers. An alternative would be to let groups of passengers change order. Note that one has to define the mutations carefully in order not to have a hight probability of decreasing the fitness of the offspring.

- **Crossover:** It is not trivial how to define the mixing of two boarding strategies. We haven't included it into the GA yet, however it would be worth considering it in future work.

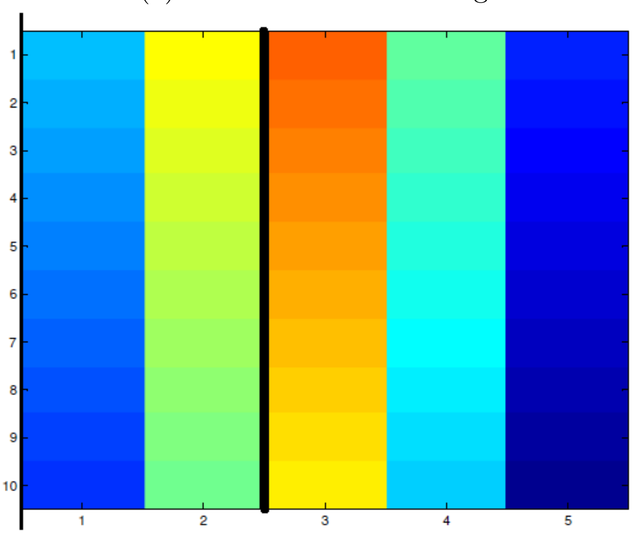
Parameter Space: Note that for dictator strategy boarding strategies the space which we search has size $N!$ (N being the number of passengers). However many of these strategies cannot lead to efficient boarding because they lead to seat interference, i.e. the process of an already seated passenger being in between an incoming passenger and his seat. We exclude these strategies as possible individuals and also only mutate in a way that avoids seat interference. Practically this means ordering the passengers in each half row so as to avoid seat interference. As a second



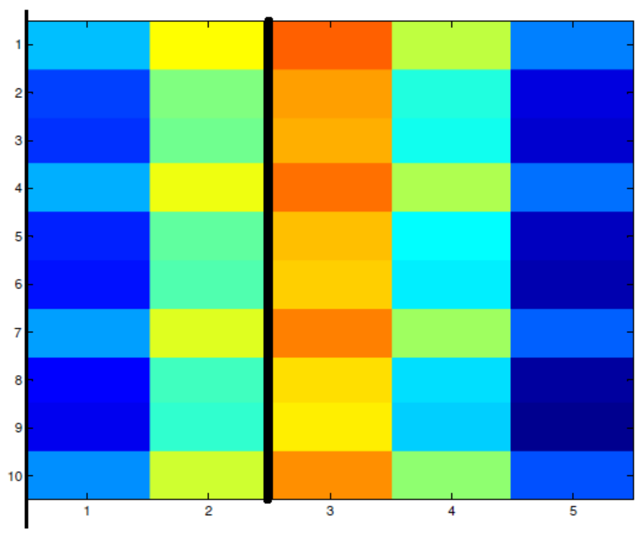
(a) Front to Back Boarding



(b) Front to Back Boarding



(c) Wilma Boarding



(d) Steffen (1.5 loading space) Boarding

Figure 10: Strategies used for seeing. The colour indicates the boarding order (blue: first, orange: later). The vertical black line represents the aisle. The Steffen strategies was always adapted to the loading space used.

reduction of space size we look at the symmetric two aisle problem, i.e. only need to consider one aisle with two seats on one side and three on the other side.

5.2 Implementation and Parameter Details

Seeding: We need to define with which strategies we want to start, i.e. *seed*. We use Front-to-Back, Back-to-Front, Wilma and Steffen (with 1.5 loading space) together with sixteen randomly generated strategies.

No. of rows	10
No. of seats left of aisle	2
No. of seats right of aisle	3
Total population size	20
Parent population size	5
Generation no.	200
Total length of airplace	100 m
Walking speed	0.5 m per sec
Stopping distance	0.4 m
Loading space (ls)	0.8 m, 1.2 m and 1.6 m
Sitting time	3s plus 5s per passenger that has to move out the way

Table 3: Parameter values.

Mutation: We mutate by letting two passengers swap places and subsequently order the passengers in each half row to avoid seat interference.

Parameters: Table 3 summarizes the parameters used for the GA and the simulation tool. The The GA was used for a loading spaces (ls) of $0.8m, 1.2m$ and $1.6m$. This corresponds to $1, 1.5$ and 2 rows respectively.

5.3 Results

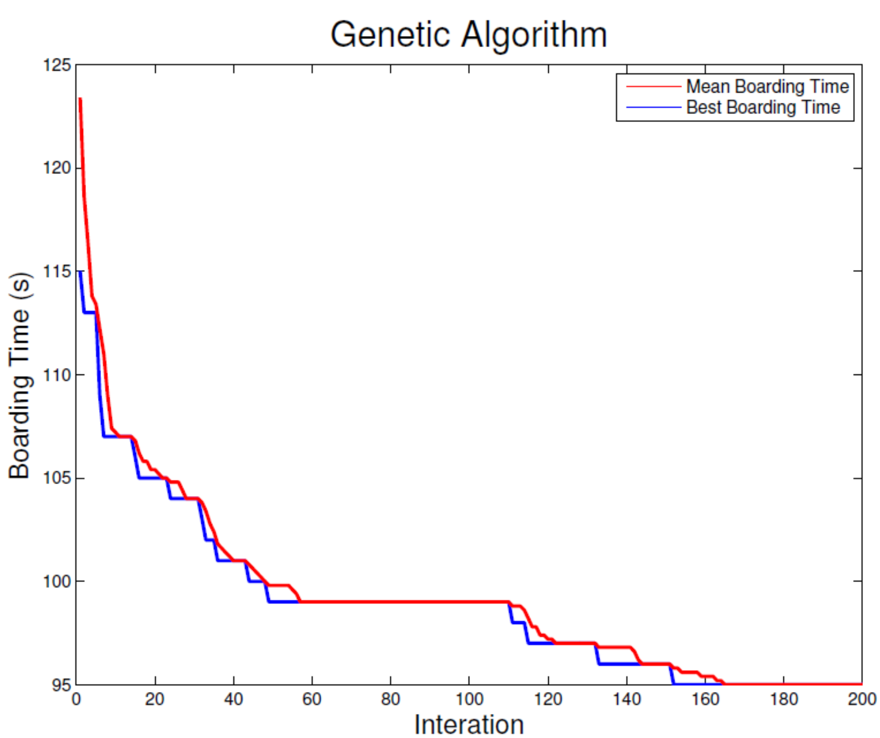


Figure 11: Fitness evolution of the GA.

Figure 11 shows how the fitness of the population and of the fittest individual changes from one iteration to the next.

Figure 12 compares the total boarding time of the seeds with the resulting strategy after 200 iterations. It can be seen that the GA will always give something better that what was started with. Note also that the Wilma strategy is not a good strategy as soon as loading space is larger that 1.

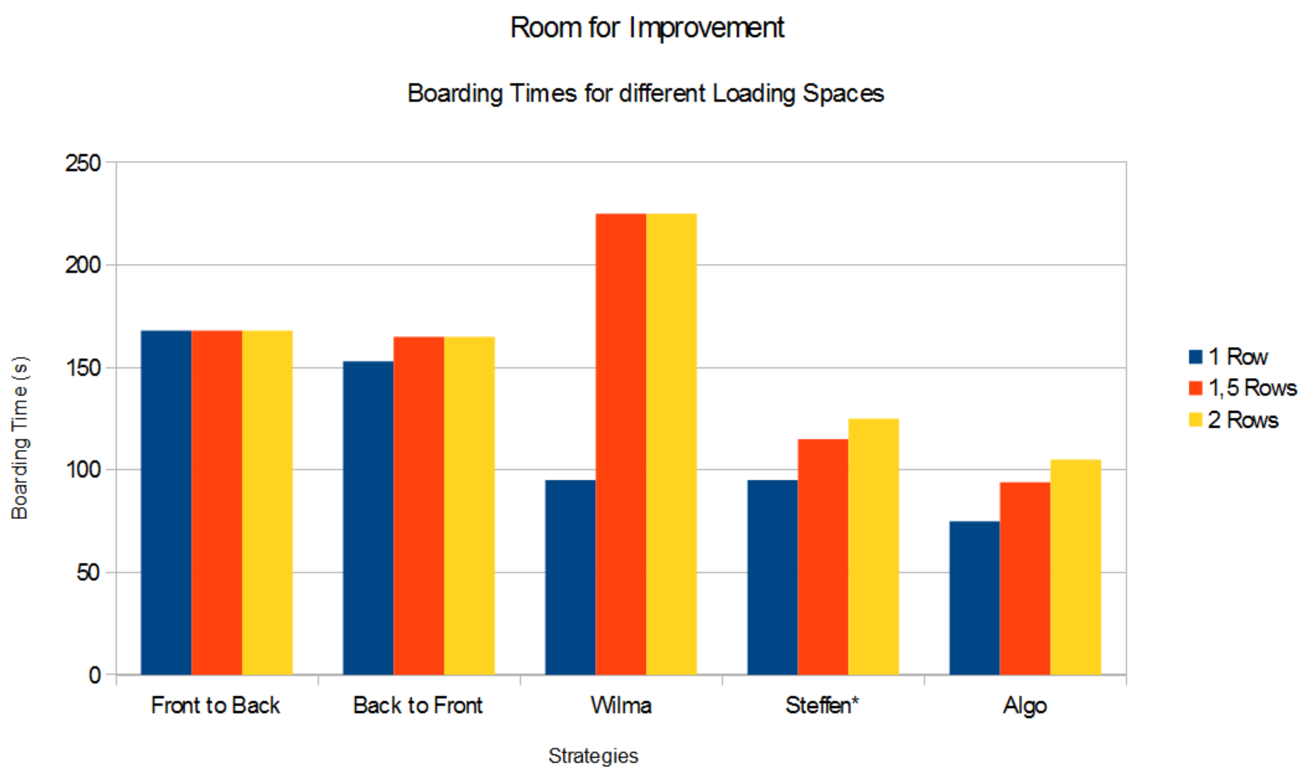
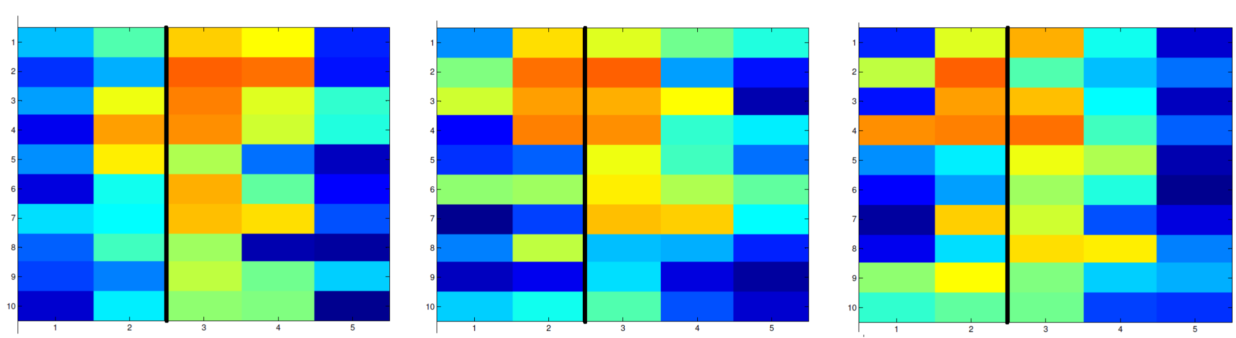


Figure 12: Strategy Comparison.

Next we can examine the strategies the GA gives to see if any pattern can be inferred from it. Figure 13 shows the best boarding strategies after 200 iterations. We can observe a combination of an outside-in behaviour with a back-to-front tendency.



(a) ls: 1 row (b) ls: 1.5 rows (c) ls: 2 rows

Figure 13: GA boarding strategies for varying loarding spaces

6 Further Work

This is of course not the end of the story. Many of the ideas we had weren't developed to their full potential. The following is an (incomplete) list of next steps and extensions:

- **Simulation tool:** The developed simulation tool has not been used to its full power. It is equipped to test various scenarios such as randomized walking speed, passenger dependent stopping distance and many more. It would be interesting to test known and new strategies with this tool and examine further how their efficiency and robustness is affected by various variables.
- **Visualisation:** To better observe where for example congestions occur it would be useful to include a visualization of the full two-aisle model. The visualisation that was done so far gives a good impression about the direction in which this would go.
- **Genetic Algorithm:** The genetic algorithm needs to be tested with boarding groups and more sources of randomness. Also it would be interesting to include cross-overs in the inheriting process. The resulting strategies also need to be examined closer.
- **Combinatorial Results:** There is room for extensions of the combinatorial results to other asymptotic regimes.

References

[1] Steffen, J. H. *Optimal boarding method for airline passengers*. Journal of Air Transport Management, 2008.

[2] Steffen, J. H. and Hotchkiss, J. *Experimental test of airplane boarding methods*. Journal of Air Transport Management, 2012.

[3] Van Den Briel, Menkes HL, et al. *America west airlines develops efficient boarding strategies.* Interfaces 35.3 (2005): 191-201.

[4] Bachmat, Eitan, et al. *Optimal boarding policies for thin passengers.* Advances in Applied Probability 39.4 (2007): 1098-1114.

[5] Team 2056, *In the Zone: Novel Approaches to Airplane Boarding* Duke Study Group 2007

[6] Team 2055, *Boarding at the Speed of Flight* Duke Study Group 2007