Vrije Universiteit Amsterdam

Royal Schiphol Group

Master Thesis

Dynamic Gate Planning: A Linear Approach

Author: Bas van der Bijl (2672884)

1st supervisor: Ger Koole
daily supervisor: Sebastiaan de Vries (Schiphol)
2nd reader: Senja Barthel

A thesis submitted in fulfillment of the requirements for the VU Master of Science degree

July 20, 2022

Acknowledgements

This thesis is submitted as part of the completion of the Master's degree program in Business Analytics at VU Amsterdam. The research described herein was commissioned by the Data Science and Engineering department at Royal Schiphol Group between January 2022 and July 2022.

In this study, an integer linear programming approach is proposed to solve the airport gate assignment problem at Schiphol. The model focuses on incorporating the human aspect of the gate assignment process to support the gate planners in their decisions. The Python code that was used in this study is available for members of the Data Science and Engineering department at Schiphol via a GitLab repository.

I would like to use this opportunity to thank the Data Science and Engineering department at Schiphol for the opportunity to write my thesis in their company. In particular, I would like to thank my supervisors at Schiphol, Sebastiaan de Vries and Philip Kerbusch. Thank you for your dedicated support, the critical feedback and for the challenges to maximize the result of my internship. Moreover, thank you for the recognition of my hard work.

Additionally, I would like to thank my supervisor from the the Faculty of Science at Vrije Universiteit Amsterdam, Ger Koole. Thank you for the advice and support throughout the Master Project Business Analytics. I greatly appreciate the biweekly feedback sessions in which we discussed my progress.

I would also like to thank Senja Barthel from VU Amsterdam for assessing my work, as well as Frank Claessen from Schiphol for reviewing my thesis.

Bas van der Bijl

Amstelveen, July 19, 2022

Management Summary

Schiphol aims to make the transition towards an autonomous airport by 2050. This means that the human interaction for their day-to-day operational activities on the airport is minimized. However, that transition towards autonomy cannot be achieved at once. Schiphol chose to make the transition towards this goal by aiding humans in decisions-making using supporting algorithms (augmented decision making). One of the processes that is currently dependent on humans, is the gate planning. This study proposes an integer linear programming model which integrates in the gate reassignment procedure of the gate planners.

The proposed method is able to optimize the schedule of the flights of 60 ramps for six hours in advance within 30 seconds. The model uses information about the initial scheduled ramp, the airline, the responsible ground handler and the position of the connected flight of every flight to optimize the quality of a schedule. The incorporation of the previous position of the flight reduces the amount of adjustments of the decisions of the model. The solutions of the model fit into the workflow of the gate planners since the suggestions remain comprehensible. When comparing the decisions of the model to the decisions of the gate planners, it can be observed that the gate planners do not always meet the constraints that are considered by the model. When using the model, it can be guaranteed that all the hard constraints of a gate planning are met. Finally, this study proposes two methods to be able to generate multiple solutions simultaneously to fulfill the final requirement. It was discovered that the best way to achieve this, was through the adjustment of the weights since the solutions of the different models can be calculated in parallel.

The results of the proposed method in this study show a lot of potential, but there are several components that should be implemented before the solutions of the model can be adopted in reality. With respect to the existing literature, this study demonstrates how a linear programming model can be used to support humans in their decision making without them having to adjusting their way of working.

Contents

Chapter 1

Introduction

1.1 Background

Amsterdam Airport Schiphol (Schiphol) is the largest airport of The Netherlands and one of the largest airports in Europe. With the connection to 316 direct destinations, 71.7 million passengers per year and 1.57 million tons of freight a year, Schiphol plays a major role in the Dutch economy. The core business of Schiphol is to facilitate the airlines, passengers and the handlers in the process of everything associated with arriving and departing aircrafts.

Between the time an aircraft lands at the airport and the time it departs again, many procedures have to be executed. One of the most obvious things is that the passengers need to disembark the aircraft. Moreover, the aircraft needs to be refueled and cleaned, supplies have to be replenished and new passengers need to board. All these actions are part of the turnaround process. The actions that affect the aircraft are taken care of by the so-called *ground handlers* which are employed by the airlines. All of the actions of the turnaround process take place while the aircraft is parked at a ramp (or stand).

The aircraft stand allocation plan, for short *gate plan*, plays an important role in the efficiency of the aircraft handling process. Gate planning is an important process to Schiphol because it is one of the few processes that is fully controlled by Schiphol. Furthermore, the quality of the gate planning determines the efficiency of execution of many other airport processes. For example, a single adjustment in the gate plan can cause an increase in travel time of the ground handlers, which results in a longer handling time for the aircraft. Since the aircraft is then required to occupy the ramp for a longer period of time, the ramp may still be occupied upon arrival of the next flight. Thus, this next aircraft might have to be moved to another ramp which causes the passengers to go through another security filter. Because the security planning did not anticipate this amount of passengers, the waiting times at the security filters may be higher. This example illustrates that all the processes at Schiphol are connected and that the gate planning plays an important role in balancing all the different processes.

Creating a gate plan is currently a largely manual task, executed by dedicated gate planners. The planners have a Gate Management System (GMS) at their disposal which is able to generate a gate plan automatically. However, the generated schedules often do not satisfy the criteria of the gate planners. The GMS meets physical constraints such as that ramps can only handle flights operated by aircrafts of certain sizes, that ramps can only handle flights for certain origins/destinations and that two large aircrafts cannot be assigned to adjacent ramps at the same time. However, the schedule generated by the GMS has to be adjusted by the planners according to their knowledge and experience, since it does not take into account, for example, experiences on certain flights often arriving too early or departing too late.

The first version of the gate plan is constructed a day in advance. The gate planners adjust the planning throughout the day when new insights or requests are obtained. Although the GMS is also able to solve conflicts during the day by adjusting the gate plan automatically, the planners use their knowledge and experience in combination with new insights to adjust the gate plan. The process of making and adjusting the gate schedule is dependent on the human ability to oversee the whole airport and think of the consequences of their decisions. This is almost impossible because of the size and complexity of the processes at Schiphol. For this reason, the whole process is considered to be vulnerable and sub-optimal. The inefficiency in this process can lead to more peak capacity moments on the airport and thus an increase in the amount of safety risks and delays.

1.2 Problem Statement

Schiphol aims to make a transition towards an autonomous airport for their day-to-day operational activities by 2050. Chances are that gate planning is one of the processes that will be largely automated by that time. However, that transition towards autonomy cannot be achieved at once. Schiphol chose to initiate the transition towards this goal by aiding humans in their decision-making using supporting algorithms (augmented decision making). When such an algorithm proves its value and robustness, Schiphol can increasingly rely on the algorithm. Since the gate assignment process heavily depends on the insights and experience of humans, an intelligent gate planning algorithm to support their decisions would be an important step towards this goal. This research contributes to the goal of Schiphol becoming an autonomous airport by answering the following research question:

How can a mathematical model optimize the gate planning on the day of operations by making adjustments in the gate planning in order to help the gate planners?

The first step towards augmented decision making can be taken by helping the gate planners in their day-to-day work. Since the gate planners are the people that will adopt the solutions of the model, there are several requirements that have to be met in order to maximize the usability of the model and its suggestions.

The first requirement the model has to meet, is that it should make its decisions in a similar way as the gate planners. To meet this requirement, it is crucial to know how the quality of a schedule can be measured. When this is known, it can be determined what components should be incorporated in the model.

Secondly, the model should respond quickly to the changing situation at the airport. Incorporating the size and restrictions of the airport makes the optimization of the gate planning a complex task. Finding the optimal solution at that moment in time requires a lot of computing power. Therefore, it is important to know how much time the model needs to find a solution.

Another requirement is that the solutions of the model remain comprehensible to the gate planners, such that they can incorporate them into their workflow. When the gate planners make adjustments in the gate planning, they only make a few changes at the same time to improve the schedule, while maintaining a mental model of the gate planning in heir mind. To match this way of working, the solutions of the model should remain comprehensible.

The last requirement stated by the gate planners, is that they want be able to choose from multiple solutions. Since the gate planning procedure is such a complex operation, it is almost impossible to incorporate all the knowledge of the gate planners in the gate planning. By presenting multiple solutions for the same situation to the gate planners, the chance of suggesting a viable solution increases. One of the alternative solutions could incorporate a for the model unknown component which matches the needs of the gate planner perfectly.

1.3 Literature

The gate assignment problem (GAP) is an important research area in air transportation planning and optimization. Depending on the airport and its characteristics, many variants of the GAP have been researched and different solution methods have been suggested. Diepen et al. [1] proposed an integer linear programming formulation that is based on gate plans based on the situation at Schiphol. The linear programming relaxation is solved through column generation. Due to the size and complexity of Schiphol, a lot of variables and constraints are involved. The column generation approach helps to reduce the solving time of the system of equations to a few minutes. The aim of of the research is to generate a robust schedule one day in advance in order to reduce the physical conflicts throughout the day. The robustness is achieved by maximizing the time between two consecutive flights at the same gate. In 2008 Dorndorf et al. [2] evaluate a solution based three different criteria: number of split flights, total preference score, and robustness. The criteria are aggregated in a linear objective function, which is then solved through a heuristic based on the clique partitioning problem; by varying the relative weights of the criteria, an approximation of the Pareto optimal frontier is determined. In a follow-up paper, Dorndorf et al. [3] consider two further criteria: the number of unassigned flights and the minimal deviation from a reference schedule.

A slightly different research area is the gate reassignment problem (GRP). Papers in this area focus on adjusting the flight-gate assignment schedule in response to disruptions in the day-of flight process. Maharjan et al. [4] formulated a binary integer program in response to flight delays. The optimization criterion in this paper is passenger comfort, which is measured by minimizing the total walking distance. Yan et al. [5] further analyze the relationship between planned and real-time gate assignment using a simulation framework. A mixed integer program is formulated that minimizes the deviation of reassignment from the original planned assignment considering the arrival time distribution for stochastic flights. Zhang and Klabjan [6] propose a gate re-assignment methodology to deal with the disruptions, in which the objective function is to minimize the weighted sum of the total flight delays, the number of gate re-assignment operations and the number of missed passenger connections.

In both research areas, mathematical programming techniques are commonly used. Gosling [7] states that these traditional approaches utilizing classical operations research techniques have difficulty with uncertain information and multiple performance criteria and do not adapt well to the needs of real-time operations support. Therefore, more recent papers focus on other algorithms. Genç et al. [8] proposed a method that combines the benefits of heuristic approaches with a stochastic approach instead of using a purely probabilistic approach to a top-down solution of the problem. Cheng et al. [9] proposed a hybrid approach based on simulated annealing and Tabu search to solve the gate assignment problem. Deng et al. [10] introduce a genetic algorithm (GA) approach for solving multi-objective gate assignment. This paper describes an adaptive Particle Swarm Optimization (PSO) algorithm to improve the convergence speed and enhance the local and global search ability.

The analysed literature focuses on the technical aspect of the gate planning where it is assumed that the quality of a planning can be quantified by a formula. At Schiphol, the gate planners are aware that this is not possible yet. They are able to give an indication what components are taken into account, but it is unknown how these components are weighted. As this research aims to support the gate planners in their day-to-day work, a model that incorporates the human aspect of the gate planning procedure is introduced. This translates into a rescheduling model that minimizes the amount of rescheduling actions. Another way how this is research differs from the literature, is that different techniques on how multiple solutions can be generated are analysed. Presenting multiple solutions to the gate planners enables them to choose the best option based on their experience.

1.4 Reading Guide

The outline for the remainder of this report is as follows: After the introduction, the required context of the problem will be given in Chapter 2. Chapter 3 gives an elaborate description on the proposed model to solve the problem. Chapter 4 gives an explanation of the testing procedure and how the results are gathered. In Chapter 5, the experimental results are discussed and the last two chapters contain the discussion and conclusion of the research.

Chapter 2

Problem Analysis

The gate planner plays a crucial role in the formulation of an appropriate gate plan. While making such plan, the gate planner considers many requirements of different parties into account. This makes the process of making and adjusting the gate planning very complex. When analyzing the flight data from the Central Information System Schiphol (CISS), it can be concluded that approximately 42% of the flights are assigned to a different gate than initially was planned. 64% of these changes are made on the day-of-operations with the aim to improve the gate schedule. Due to the size and complexity of Schiphol, it is hard to trace back how the adjustment improved in the schedule and why they are necessary. According to the gate planners, there are three main improvements that gate changes can contribute to. This chapter provides general insights in the ramp schedule and analyses the reasons behind these changes. The analyses in this chapter are performed over the flights between October 2021 and January 2022 to obtain trustful and robust insights.

2.1 Gate Scheduling Procedure

In November 2021, every day on average 1100 commercial flights a day arrive and depart from Schiphol. Each flight is allocated to one of the 140 ramps. The first planning one day in advance is based on the scheduled block times of the flights. To get a better understanding of the size and complexity of this task, Figure 2.1 shows the initial schedule of the flights assigned to the ramps at the E-pier at November 1, 2021. The bars in the plot represent the time the ramps are occupied by aircrafts. The E-pier is with 51 flights divided over 15 ramps one of the more quiet piers. When considering the whole airport with constantly changing conditions, making an appropriate gate planning becomes very complex.

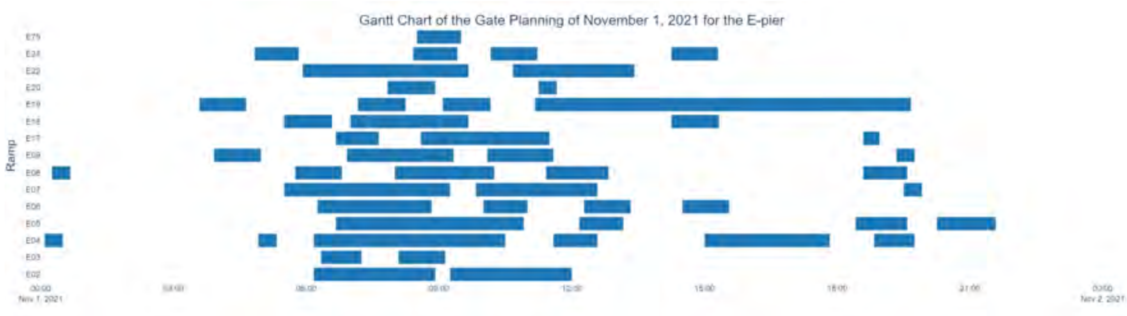

Figure 2.1: Gantt chart of the scheduled block times for flights assigned to ramps at the E-pier for November 1, 2021.

2.2 General Gate Reassignments

Before the reasons behind the gate changes are indicated, having a better understanding of the gate reassignment process is crucial to fully understand the problem. The gate reassignment process is analysed by visualising the moments and characteristics of the gate changes in order to find patterns. A large part of the gate reassignments are executed throughout the day of operations. The insights of this analysis can be incorporated in the gate reassignment model.

First, the timing of the gate changes is analysed. Finding patterns in the timing of the gate changes can give more insights in the decisions of the gate planners. The timing of the gate changes is analysed by calculating the amount of time between the moment of the gate change and the expected block time of the flight. Figure 2.2 shows the distribution of timing of gate changes in minutes, for executed flights. From the plot it can be seen that gate changes are executed mostly between two to three hours in advance of the expected block time of the flight. Furthermore, it can be seen seen that the shape of the distribution looks similar to the shape of an exponential distribution.

Every change in the flight data is stored as a new individual record in the CISS flights table. This make it seem like the gate planners execute one gate change at a time, but the gate planners indicate that they often execute multiple changes at the same moment. Because the timestamp of the moment when the change is executed is stored in the data, it is possible to calculate the time between two gate changes. When looking at the histogram of the number of seconds between two gate changes in Figure 2.3, it can be observed that more than 50% of the changes occur within a single second of the previous gate change. Since only one or two gate planners work on the gate schedule at the day of operations, it is unlikely that changes within a second are not part of the same decision of a gate planners. When assuming that the gate planners are able to draft multiple changes in their GMS before the GMS publishes the changes to CISS, changes within such a small amount of

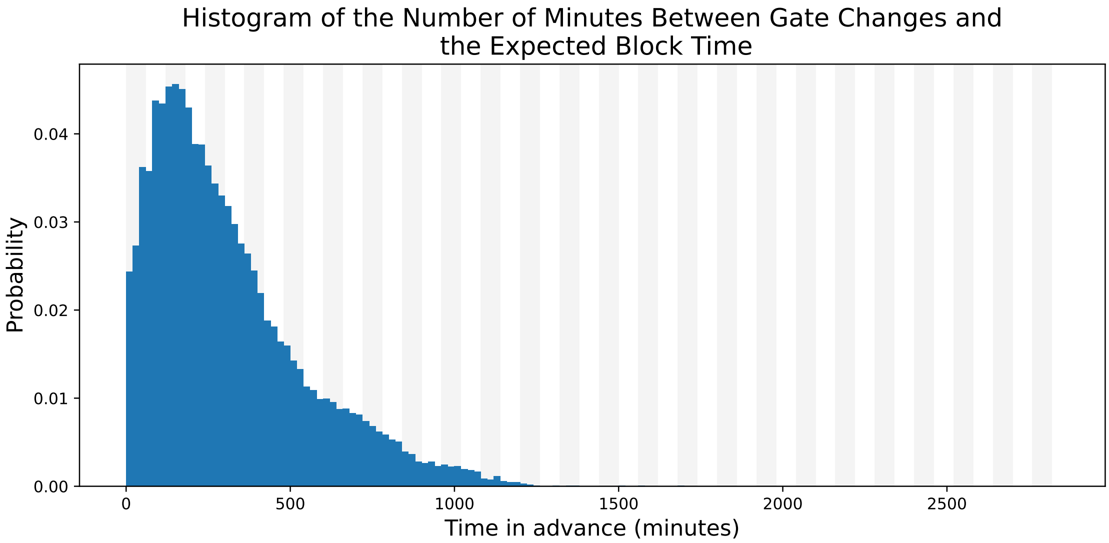

Figure 2.2: Histogram of the distribution of the timing of gate changes.

time can be clustered. Gate changes in the same cluster are considered as being part of the same decision of the gate planner.

After applying the proposed logic to the gate changes, the distribution of the clusters, based on the number of changes in a cluster, can be seen in the histogram in Figure 2.4. By looking at the plot it can be seen that the vast majority of the decisions consist clusters with less than five changes. On average, a cluster consists of 2.2 gate changes. This seems logical since this amount of changes remains comprehensible.

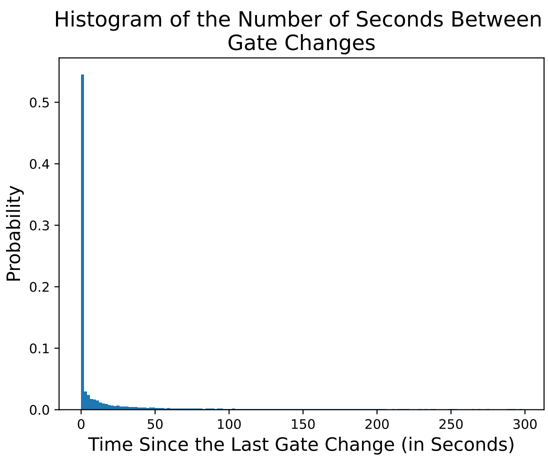

Figure 2.3: Histogram of the time between individual gate changes.

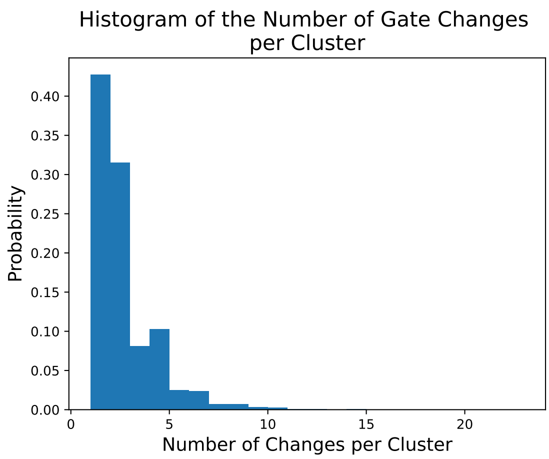

Figure 2.4: Histogram of the size of gate changes clusters.

Other insights are obtained by analyzing the difference in gate changes between characteristics of flights. When looking at the arriving and departing flights, we can see that the probability of a gate change of an arriving flight (30%) is higher than a gate change for a departing flight (25%). A similar difference can be noticed when comparing the Schengen flights to the Non-Schengen flights. A Schengen flight is a flight arriving from or departing to a country that is part of the Schengen agreement. Passengers from countries that are not part of this agreement, require additional security checks compared to passengers on Schengen flights. 29% of the flights arriving from or departing to Schengen countries experience a gate change. This is slightly higher than the Non-Schengen flights, 25% of these flights are rescheduled to another ramp.

Other interesting characteristics that could give more information about the reason behind gate changes, are the scheduled pier and the airline of the flight. A pier is a part of the terminal which consists of a set of gates. Figure 2.5 and Figure 2.6 show the proportion of the total flights where the gate is changed throughout the day, and the figures are split by different characteristics. When looking at Figure 2.5, the proportions are split by the pier where the flight was scheduled initially. Some piers at schiphol are arranged for different types of flights. In the previous paragraph, we touched upon the Schengen/Non-Schengen flights. At Schiphol, the B-pier and the C-pier are not able to execute the checks for Schengen flights and only the D-pier is able to handle both the Schengen and Non-Schengen flights. The heterogeneity in the piers, make the process of making and adjusting an appropriate gate planning for Schiphol complex. This could be a reason for differences in the proportion of the flights that is assigned to another ramp. Nevertheless, Figure 2.5 does not show large differences in the proportions which can be explained by the characteristics of the piers. When going one step further by looking at the origin and destination of the gate change, it is observed that a large part of the gate changes the pier is not changed. For 74% of the gate changes, the flight is assigned to a ramp on the same pier as before the gate change.

In Figure 2.6, the proportions for the five airlines that are responsible for the most flights at Schiphol are shown. These flights are together responsible for 76% of the total amount of flights. It can be seen that the proportion of the flights for Air France that are rescheduled is much lower than the other four airlines. The case that KLM is in the top of the list is very interesting since KLM alone is responsible for 49% of the total flights at Schiphol. KLM is the most important customer for Schiphol and therefore receives some privileges over other airlines. KLM is, for example, allowed to publish and adjust what incoming flight and outgoing flight are connected at a later moment. This will be further analysed in Section 2.4.

Looking at the gathered statistics of the gate changes, no clear patterns can be identified. The proportion of the flights where the gate is adjusted show differences between the

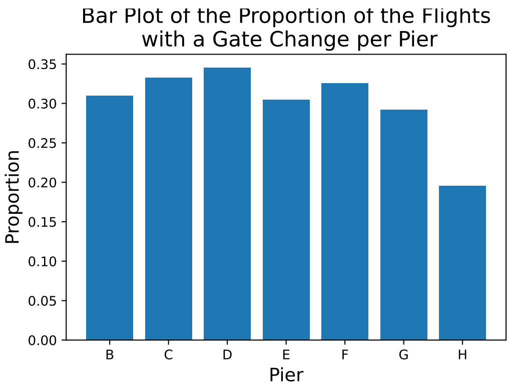

Figure 2.5: Bar plot of the proportion of the flights that is scheduled on a pier undergo a gate change.

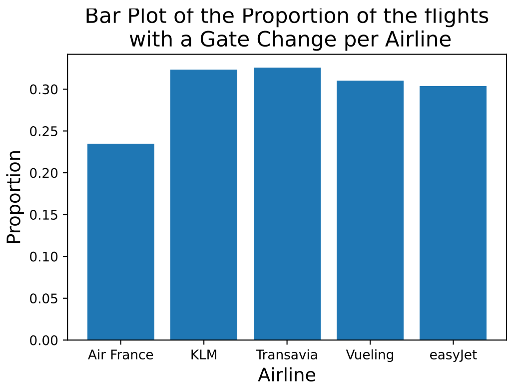

Figure 2.6: Bar plot of the proportion of the flights with a gate change of the 5 airlines with the most flights at Schiphol.

characteristics of the flights, but do not give an explanation for the changes. For the explanation behind the gate changes, other aspects of the environment have to be analysed.

2.3 Handler Clustering

The gate planners indicate that the first possible reason for a gate change is the possibility to improve the clustering of ground handlers of two flights. The turnaround process is executed by the ground handlers that operate in small teams. These teams travel between ramps to handle different aircrafts. Since the ground handlers deal with a limited capacity, an increase in travel time could result in too little time to process all the flights, potentially resulting in flight delays. To prevent this, the gate planners try to schedule the aircraft such that two consecutive flights at the ramp are operated by the same handling agent (clustering). Another advantage of clustering handlers, is that the airline-related processes can be streamlined with the result that the turnaround process takes less time. When analysing the data, it can be seen that in approximately 55% of the gate changes, the cluster criterion does not change after the gate change. Moreover, for 22% of the gate changes, the cluster criterion even improves. Even though the clustering might not be the main reason for the gate changes, it definitely is an important criterion for the decisions of the gate planners.

2.4 Connecting Flight Changes

Another cause for gate changes, are changes in the connecting flights. As mentioned earlier, KLM has the privilege to change the connection between an arriving flight and a departing flight throughout the day, this happens on average for 20% of the flights a day. Changes following from this process could lead to an increase in the amount of tows to move an aircraft to the appropriate ramp. Adjusting the ramp of the arriving flight to the same ramp as the connected departing flight could prevent this.

Figure 2.7 gives a better understanding of when the connected flights are adjusted by visualising the time in minutes between the moment of the change and the expected block time. By considering the chart, it can be observed that the probability of a connecting flight change is rather constant in the last nine hours in advance of a block time. Other interesting observations are the two major drop-offs at nine hours and at fourteen hours in advance of the flight. This can be explained by the fact that the certainty of the flight information increases as the flights approach their block times. The airlines anticipate on this increasing certainty by constantly optimizing their flight configuration.

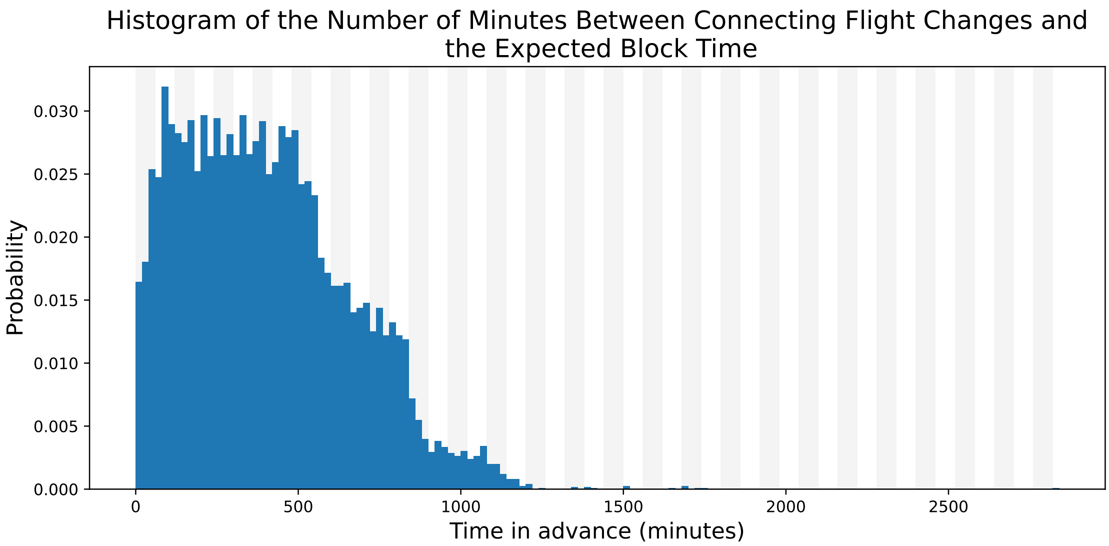

Figure 2.7: Histogram of the distribution of the timing of connecting flight changes.

A next step in the analysis is identifying the gate changes caused by adjustments in the connecting flights. Since the gate changes are done manually and the gate planners need time to react on a change in the environment, we look at the time between the last change of the connecting flight and the moment of the gate change. The histogram in Figure 2.8 shows the distribution of the number of minutes between these two events. It can be seen that 36% of the gate changes are executed within 20 minutes after an adjustment of the connecting flight. When assuming that all the gate changes within the 20 minute window are due to the change in connecting flight, it can be concluded that 36% of the gate changes are caused by these events.

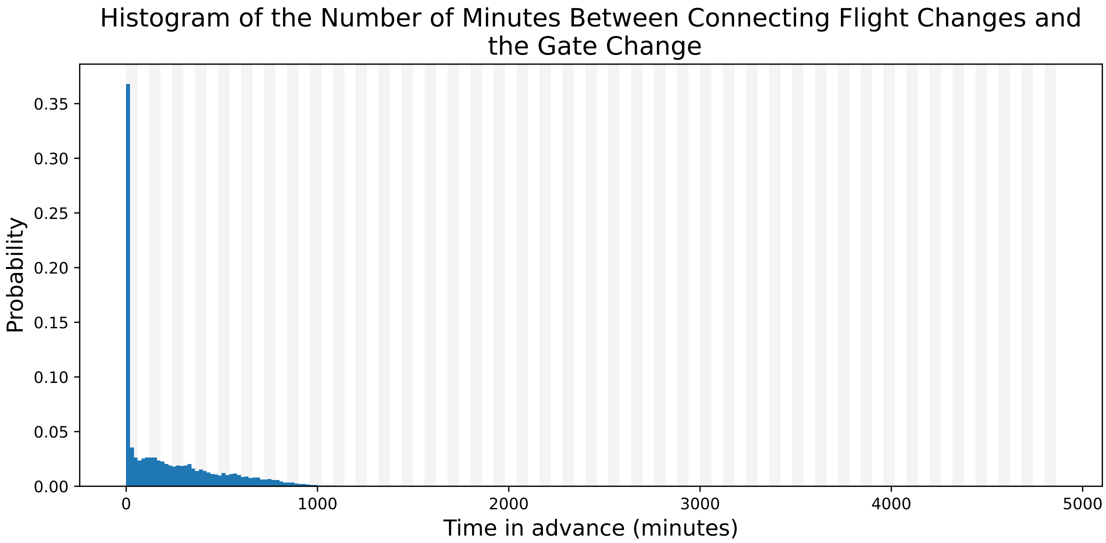

Figure 2.8: Histogram of the distribution of number of minutes between the last connecting flight change and the gate change.

2.5 Block Time Deviation

The third reason indicated by the gate planners, are the gate changes caused by deviating block times of aircrafts compared to their scheduled block times. The first version of the gate planning is based on the scheduled block times of flights. When an aircraft arrives for example earlier than planned, it could be possible that the ramp is still occupied (a gate conflict). By changing the ramp of either of the involved aircrafts, the congestion can be avoided, preventing delays.

From the CISS data, it can be deduced that 12% of the gate changes are executed to solve the gate conflicts due to deviation of the block times. To obtain more insights about the deviation of the actual block times, Figure 2.9 and Figure 2.10 compare the actual block times to the scheduled and expected block times, respectively. When looking at Figure 2.9, it can be seen that the majority of the flights arrive or depart within the 20 minute interval of the scheduled block time. This seems to be rather accurate, but even such a small deviance can cause gate conflicts on tightly scheduled gates.

Figure 2.10 shows box plots for the difference between most recently updated expected block time at that moment in time and the actual block time. Airlines and handlers are able to update the time an aircraft is expected to arrive or depart throughout the day based on new information. When the actual block time of a flight approaches, the uncertainty reduces. This can also be observed from the plot, where the size of the boxes reduces as the number of minutes between the moment in time and the actual block time decrease. A smaller box means that the spread of the deviation is smaller, which implies a higher accuracy. Furthermore, it can be seen that for expectations further ahead, the distribution of the deviations of the block times are slightly skewed towards the right as there are seem to be more positive outliers.

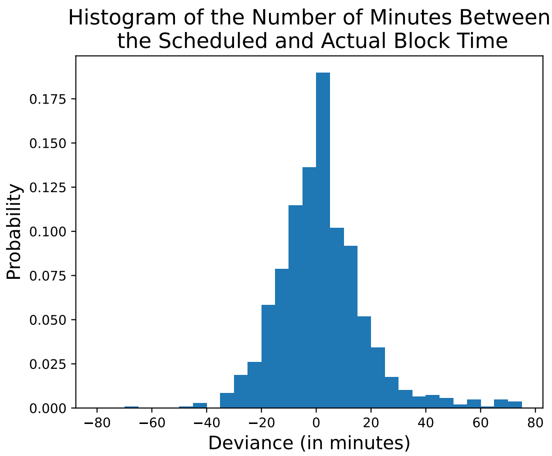

Figure 2.9: Histogram of the deviations between the scheduled block times and the actual block times.

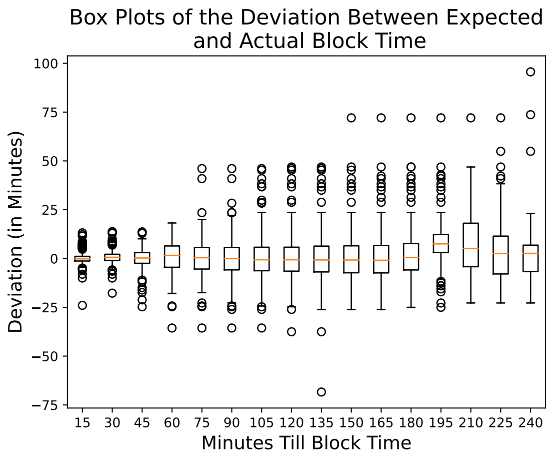

Figure 2.10: Box plots of the deviation between the most recent expected block times and the actual block times at a 15 minute interval.

2.6 Concluding Remarks

In conclusion, the gate planners indicate the presence three main reasons for gate changes, namely: handler clustering, connecting flight changes and gate conflicts due to block time deviations. During the last part of the problem analysis, these reasons are analysed by identifying the presence of these reasons for past changes. From the analysis it could be concluded that these three reasons are clearly present of a majority of the gate changes and should therefore be incorporated in the model.

Furthermore, the CISS data provides a lot of insights in the process of the gate planners. Even though every change is stored as an individual record, a set of changes can be clustered together by using the input moment of the record. The clusters contain on average 2.2 records when clustering the individual gate changes together if the changes occur in short succession. This seems logical since in this way the adjustments remain comprehensible.

When analyzing the characteristics of the gate changes, no clear patterns could be recognized. This could be caused by the fact that the gate reassignment process is triggered by random events. Solving the events can be done in several ways where it is unknown what the best solution is. It is known what considerations are made by adjusting the gate planning, but how they are balanced is dependent on the experience and personal preference of the responsible gate planner.

Chapter 3

Modelling

A good schedule is a trade-off between several components while meeting some hard requirements. Such a schedule can be found when all these aspects are translated and combined into a mathematical model. This chapter elaborates on what aspects are modelled and how these aspects are translated into mathematical formulas.

3.1 Restrictions

The main challenge of the gate assignment process is caused by the limited space and time at an airport. For this reason, the model has to deal with many restrictions, documented in the Regulation Aircraft Stand Allocation Schiphol (RASAS) document [11]. This section further elaborates on what restrictions a gate planning has to meet to make it appropriate.

The first restriction for an appropriate gate planning, is that every flight needs to be assigned to a single ramp. In addition to this restriction, only one flight can be present at a ramp at the same time. This means that the interval between the in- and off-block times two flights at the same ramp cannot overlap.

Another restriction for the allocation of an aircraft at Schiphol is caused by the heterogeneous ramps at the airport. Ramps can handle only flights operated by aircrafts of certain sizes. Aircrafts are divided in categories from one to ten based on their length and wingspan. The aircraft categories are used by Schiphol to indicate which aircraft fit on the ramps. This is achieved by indicating what the maximum aircraft category is for a ramp. However, there are some ramps where this restriction is more complex due to technical or safety reasons. There are some ramps that exclude some aircrafts from the indicated maximum aircraft category or include aircrafts from a higher aircraft category. These exceptions are clarified by a free text field in the aircraft stand configuration. Due to the lack of consistency of this free text field, it is inconvenient to make a perfect rule set for this restriction during this research. This is solved by increasing the maximum aircraft category for a ramp when there are aircrafts from a higher category included.

The final restriction that an appropriate gate assignment has to meet, has to take the origin of a flight into account. In Chapter 2, the difference between Schengen and non-Schengen flights was already mentioned briefly. In accordance with the agreements between the Dutch government and foreign governmental authorities concerning the free movement of goods and persons, the terminal of Schiphol has been divided into different zones. These zones distinguish between Schengen passengers and non-Schengen passengers. The separation between the passengers means that only Schengen flights can be assigned to ramps that are connected to the Schengen zone and vice versa.

3.2 Objective

The objective which determines the quality of a gate schedule is less deterministic compared to the restrictions. Chapter 2 already elaborated on some of the reasons on how a gate change improves the schedule. This section discusses the different components that are incorporated in the model and explains why they are important for the gate planners.

Chapter 2 showed that the clustering of the handlers and the reduction of the number of tows are taken into account while making adjustments in the gate schedule to improve the turnaround process. To increase the quality of the decisions of the model, it is important to incorporate these components in the model. Since these are not hard constraints, this will be achieved by adding them to the objective function of the model.

Another component that should be incorporated in the objective, is the pier preference of the airlines. When all the restrictions are met, airlines are able to indicate the preferred pier of their flights. The gate planners choose whether to grant the preferences of airlines if this fits in the demand at that moment at the airport. The preferences of the airlines can be seen in the distribution of the distribution of the flights of the airlines over the piers. Section 2.1 highlighted that the flights of the more expensive airlines like KLM are assigned more often to the B-pier while flights of the cheaper airlines like EasyJet are assigned more often to the H-pier.

The final requirement for the decisions of the model, is that the solutions remain comprehensible to the gate planners. When the model suggests a set of gate changes that adjusts the planning for the whole airport, the gate planners become suspicious because it is almost impossible to keep track of why and how the changes improve the schedule. Furthermore, the gate planners incorporated a lot of their knowledge and experience in the schedule. Undoing their decisions increases the chance of making unfavourable suggestions in the eyes of the gate planners. In addition to that, does the limited amount of adjustments also matches the current process of the gate planners. Their process of improving the schedule can be compared to local optimization problems where the majority of the schedule remains the same.

3.3 Mathematical Notation

The literature study showed that the GAP can be solved using different techniques. A frequently used method is translating the restrictions and the objective of the problem to a set of mathematical formulas. The advantage of these type of algorithms is that there is a lot of control in the shape and size of the solution space. When a set of solutions is excluded from the solution space by one or more constraints, it is certain that the solution is viable since it must meet the requirements.

The indicated components that are included in the model require the incorporation of information about the sequence of the flights at the ramps. When it is unknown to the model what the previous flight of another flight is, it is unable to detect ground handler clustering or required tow movements of flights. Maharjan et al. [4] implemented during their research such an integer linear program (ILP) at the George W. Bush Intercontinental Airport. The researchers introduced a sequence variable to prevent the block times of two flights at the same ramp to overlap. Even though this variable has no further functionality in their model as they seek to minimize the walking distance of the passengers from the original to the new ramp. By adjusting their model such that the sequence parameter is used in the objective, it fits the situation at Schiphol well. How the adjusted model for this research is implemented is discussed thoroughly in this section.

3.3.1 Sets

To make the linear model more comprehensible, several sets are initiated. This subsection elaborates on what sets are used for the ILP.

- F^t= Set of actual flights with arrival times greater than current time t
- V_P= Set of virtual arrival flights preceding the first arrival of a flight
- V_L= Set of virtual flights following the last departure of a flight
- K= Set of all gates available for gate reassignment
- U^t= Set of all the flights (both actual and virtual), $F^t \cup V_P \cup V_L$
- U^t_P= Set of the union of the actual with the preceding flights, $F^t \cup V_P$
- U^t_L= Set of the union of the actual flights and the virtual last flights of the day, $F^t \cup V_L$

3.3.2 Parameters

In addition to knowing what flights should be scheduled, additional parameters of the flights are introduced to meet the restrictions. The parameters in this study are treated as deterministic values at each point in time, even though the values are known to vary throughout the day. The updates on the parameters cause the equations of the model to change which could result in a new optimal schedule. Which parameters of the flights are required to define the ILP, are explained in this subsection.

- a_i^t= Expected arrival time of flight i at time t, $i \in U^t$
- d_i^t= Expected departure time of flight i at time t, $i \in U^t$
- h_i^t= Responsible handler for flight i at time t, $i \in U^t$
- c_i^t= The connecting flight of flight i at time t, $i \in F^t$
- Ω_i= Set of appropriate gates for flight i, $i \in U_V^t$, $\Omega_i \in K$
- s_{ik}^t= A binary indicator equal to 1 if and only if flight i is originally scheduled at gate k, $i \in F^t$, $k \in \Omega_i$
- l_{ik}= The preference score when flight i is assigned to gate k, $i \in F^t$, $k \in \Omega_i$

3.3.3 Variables

The values of the variables of a model have a direct impact on the quality of the solution in linear programming. The optimal objective value is achieved by finding the best schedule, which is created or adjusted by changing the values of the variables. This subsection elaborates on which variables are used to define the model. Since the model only uses binary variables, the model can be described as a ILP.

- x_{ik}= Binary variable equal to 1 if and only if flight i is assigned to gate k, $i \in U_V^t$, $k \in \Omega_i$
- z_{ijk}= Binary variable equal to 1 if and only if flight i and j are assigned to gate k, and j immediately follows flight i, $i \in U_p^t$, $j \in U_l^t$, $k \in \Omega_i \cap \Omega_j$

3.3.4 Mathematical Model

The mathematical model itself incorporates all the requirements while optimizing the quality of the schedule. While constructing the mathematical model, the system is intentionally kept linear. Systems of linear equations have been proven to lead to fast solving times compared to a non-linear model. Another benefit of a linear model, is that the linearity of the equations make it easier to explain its decisions. The gate scheduling model is composed of the following equations.

$$\max \quad \sum_{i \in F^t} \sum_{k \in \Omega_i} \left(\alpha_1 s^t_{ik} x_{ik} + \alpha_2 z_{ic^t_i k} + \alpha_3 \sum_{j \in F} \left(I_{ij} \sigma \left(d^t_j - a^t_i \right) z_{ijk} \right) + \alpha_4 l_{ik} x_{ik} \right) \tag{3.1}$$

$$\text{subject to:} \quad \sum_{k \in \Omega_i} x_{ik} = 1 \qquad \forall i \in F^t \tag{3.2}$$

$$\sum_{j \in U^t_L, a^t_j \geq d^t_i} z_{ijk} = x_{ik} \qquad \forall i \in U^t_P, k \in \Omega_i \tag{3.3}$$

$$\sum_{i \in U^t_P, d^t_i \leq a^t_j} z_{ijk} = x_{jk} \qquad \forall j \in U^t_L, k \in \Omega_j \tag{3.4}$$

$$x_{ik} = s^t_{ik} \qquad \forall i \in F^t, k \in \Omega_i, a^t_i \leq t \tag{3.5}$$

$$x_{ik}, z_{ijk} \in \{0, 1\} \tag{3.6}$$

Where,

$$I_{ij} = \begin{cases} 1, & \text{if } h_i{=}h_j \\ 0, & \text{otherwise} \end{cases} \tag{3.7}$$

$$\sigma(x) = \frac{e^{4-x/6}}{e^{4-x/6} + 1} \tag{3.8}$$

Objective Function

The objective function of the model given by Equation (3.1) denotes a linear combination of four components. The model seeks to maximize this objective by assigning the flights to the ramp which returns the highest reward.

The first component in the objective measures the amount of flights where the ramp of the initial planning is not adjusted in the new planning. For every flight that is rescheduled to a new ramp, the model is penalized by reducing the score of this component by one. Incorporating this component in the objective function prevents the model from making redundant adjustments in the planning. Since this property reduces the number of the schedule, it ensures to take the requirement of comprehensible solutions into account.

The tow score is incorporated by the second part of the objective. When two connecting flights are assigned to the same ramp without another flight in between them, the aircraft does not need a tow. To be able to detect this, variable z is introduced. When the value of z for two connected flights at any ramp is equal to one, the aircraft does not need a tow. Incorporating the values of these variables in the objective, gives the model a reward when there is no tow necessary.

The formula for third part of the objective is more complex than the other components. In order to approximate the effects of clustering of the handlers, two additional parameters are introduced. I_{ij} is an indicator equal to one if the ground handlers of flights i and j are the same. This parameter ensures that the model is only rewarded when two consecutive flights at a ramp are handled by the same ground handler. The second parameter, $\sigma(x)$, determines the amount of reward based on the amount of time (in minutes) between two consecutive flights. The added value of two flights handled by the same ground handler is approximated using Equation (3.8). The used equation is a transformed sigmoid function where the value for two flights with the same ground handlers close to each other is higher than the value of two flights with a more time between them. Figure 3.1 shows what reward is given for the amount of minutes between two flights. It can be seen that the reward for a flight with the same ground handler as the previous flight arriving within ten minutes is higher than 0.8. While arriving later than 30 minutes after the previous flight reduced the reward to less than 0.4.

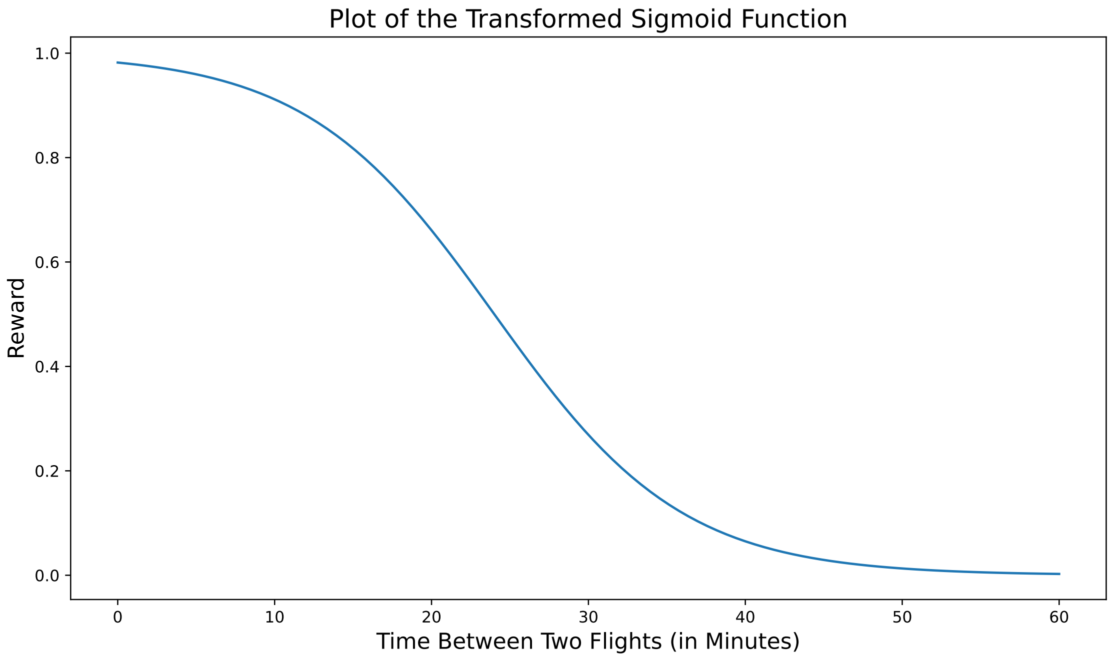

Figure 3.1: Plot of the transformed sigmoid function as formulated in Equation (3.8).

The final component of the objective improves the solutions of the model by increasing the objective when an aircraft of a particular airline is scheduled on the preferred pier. This helps the model to distinguish what airlines are usually located on what ramps. Since a large part of the flights is scheduled on the E-pier, the model is rewarded when those flights are scheduled on that pier.

The overall objective score is a trade-off of the different components. Making an extra adjustment to the schedule reduces the overlap score, but could improve the cluster score or reduce the number of required tows. The model deciding whether that adjustment is worth it depends on the weights of the individual components. The weights α_o determine how much each component contributes to the overall objective. The weights of the parameters are crucial for the practical usability as different weights result in other solutions.

Constraints

The constraints in the proposed model ensure that the solutions of the model meet the requirements of the gate planning. Individual constraints, or combinations thereof, ensure that these requirements are met. The first constraint in Equation (3.2) implies that every flight i must be scheduled to a single ramp. The second and third constraints (Equations (3.3) and (3.4)) expand the decision variables x_{ik} and x_{jk} in terms of decision variable z_{ijk} to represent the sequence of two flights at gate k. This functionality is achieved because the two equations enforce that every flight has a preceding and a consecutive flight. The conditional sums in these constraints prevents that the block times of two flights at the same ramp overlap.

3.4 Solver

The mathematical model is implemented and solved using the programming language Python. Python provides numerous packages on how to implement and solve linear programming models. Packages like PuLP, Pyomo and Google OR Tools provide frameworks to gradually define a model and solve it with different solvers. These solvers can be divided in two groups; the open source solvers and the premium solvers. The open source solvers have the advantage of being free to use while the premium solvers require a paid license. Premium solvers like Gurobi and CPLEX are known for their good performance on large problems. By using a research licence during this research, it is possible to compare the results of the different solvers. An example of a well-performing open source solver, is the COIN-OR Branch and Cut (CBC) solver. During this research the performance of Gurobi and the CBC solver are compared.

3.4.1 Solving Strategies

When composing the proposed linear model for the whole airport for an average day, the model consists of approximately 50 billion variables. Finding the optimal solution for a model takes a lot of time, even with the premium solvers. Taking into consideration that the model should be able to respond quickly on changes on the airport, the long solving times are not desirable. The reduction of the number of variables can be achieved by restricting the space or the time dimensions. The space dimension can be reduced by only considering a subset of the ramps of the airport. The reduction of the time dimension can be reduced by allowing the model to adjust only the flights within a certain time window.

The disadvantage of reducing the space and time dimensions of the model, is that the solutions of the model become sub-optimal compared to a model that is able to adjust the whole airport. Therefore, a trade-off between the performance and the solving time of the model has to be made.

Chapter 4

Evaluation

To evaluate the performance of the model, the model is tested in different scenarios. This chapter gives an elaborate description of what testing scenarios are selected and how the evaluation procedure is executed.

4.1 Evaluation Set

To reinforce the confidence in the model, the performance indicators of the model should reflect the behaviour of the model in real-world cases. More valuable results are obtained by evaluating the model on historical days at the airport. To get a better understanding of the decisions of the model on different days are analysed. The set of evaluation days can be separated in three categories. For every category, two days are selected and added to the evaluation set.

The first two categories of the evaluation set can be distinguished by the number of flights on a day. Days with a lot of flights and fewer flights both have their own challenges. Days with a lot of flights result in a tighter schedule because the number of ramps at the airport is fixed. Because there is less time between two flights at a ramp, a small delay can already have a lot of impact on the schedule. Since all the ramps have the same problem, solving such conflicts requires more gate changes thus becomes more complex. Days with fewer flights result in more space at the ramps, which increases the amount of viable solutions. The challenge for these days is for the model to choose the best solution. Selecting these days is done by looking at Figure 4.1 where the number of flights per day between January 1, 2019 and April 18, 2022 are visualised. When looking at the figure, the drop in number of flights due to the Corona pandemic can be noticed. The last few months show a recovery in the number of flights, but this is still not on the same level as before the Corona pandemic.

For the set of evaluation days, two days during the summer break before the Corona pandemic and two days during the Corona pandemic are added to represent the busy and

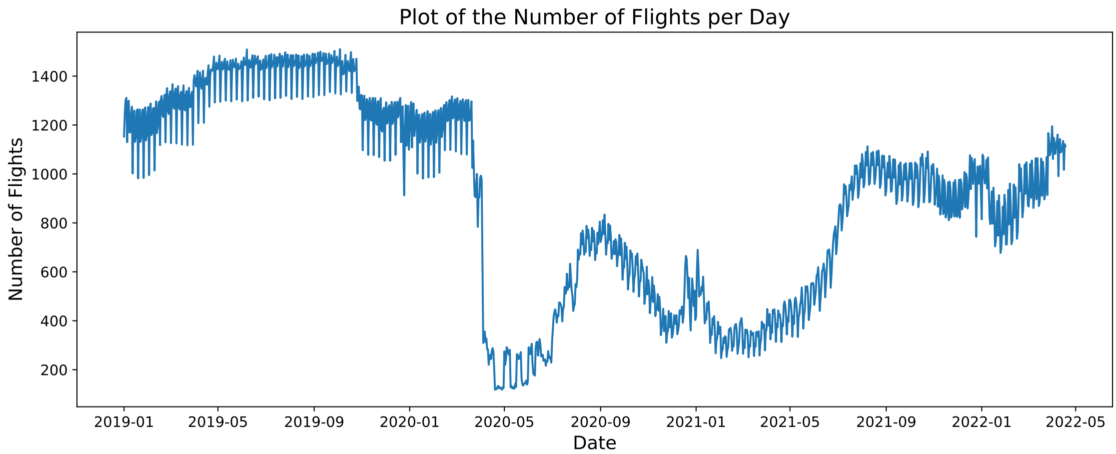

Figure 4.1: Line plot of the number of flights per day.

more quiet days. June 7, 2019 and July 26, 2019 are two busy days in the summer break where 1509 and 1497 flights departed, respectively. Two days with fewer flights are May 27, 2021 and April 4, 2021 with 344 and 402 flights, respectively. These four days are all added to the evaluation set.

The third category of days selected for the evaluation set, are days with many delayed flights. Days with a high percentage of delayed flights increases the chance of physical conflicts. Testing the model on such chaotic days gives insights whether the solutions of the model are consistent, even when unusual events occur. A Key Performance Indicator (KPI) for Schiphol that can be used to identify such days, is the on-time performance (OTP). The OTP is the percentage of commercial flights where the actual departure time compared to the scheduled departure time is less than or equal to 15 minutes. The OTP is important to Schiphol because this gives information about the performance of the turnaround process. If aircrafts arrive at the scheduled time and there are no disruptions in the turnaround process, the flights should be able to depart at the scheduled time. When there is for example a disruption at the start of the day, delays later of the day stack up which result in a low OTP. Schiphol has the long-term ambition to reach a stable OTP of 80%. Figure 4.2 shows the values for the OTP per day. It can be seen that the goal of Schiphol to reach an OTP of 80% was achieved during the Corona pandemic due to the reduced number of flights per day. As the number of flights per day increases, the OTP becomes less stable.

The plot shows a lot of major drops in the value of the OTP. The majority of the drops occur during the winter when there is for example a heavy snow storm. One reason for the delays during this type of weather, is the fact that additional actions have to be taken during the turnaround process to prevent the wings from freezing during the flight. The all-time low at February 7, 2021, where only two flights have departed on time, was

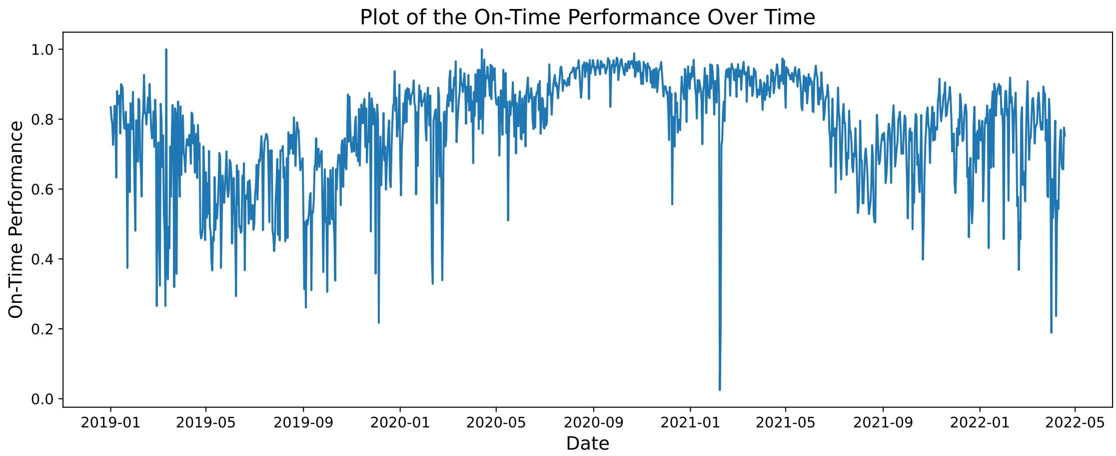

Figure 4.2: Line plot of on-time performance per day.

caused by such a storm according to Schröder [12]. More interesting days with a low OTP happened more recently; April 1, 2022 and April 27, 2022 achieved an OTP of 18.8% and 23.7%, respectively. The low OTP on these days were caused by the unfavorable wind conditions. Due to these conditions, the capacity of the runways could not handle the amount of flights.

4.2 Evaluation Procedure

As mentioned earlier, every change in the flight data is stored in the CISS flights table. Because the time when the change occurred is stored in the data, it is known what information was available at every moment in time. When the flight data is updated because of new insights, it can be detected what impact these updates have on the overall schedule.

Besides the possibility to solve overlapping block times of flights at the same ramp, the model seeks to minimize the number of tows since these are incorporated in the objective function. Therefore, the solutions of the model are tested based on two types of triggers. The first trigger detects when a delay in the flights results in overlapping block times in the schedule at that moment in time. The other trigger detects when a change in the connected flights results in an increase in the number of tows. If this is the case, it is assumed that the schedule at that time became sub-optimal. When either one of the detection mechanisms is triggered, the model is run in order to find the optimal schedule at that moment in time. How often the detection mechanisms are triggered is shown in Table 4.1.

Comparing the number of trigger moments of the busy days and the quiet days the amount of flights appear to have a lot of impact on the number of evaluation moments. This can be explained by the fact that when there are more flights, there are more objects that can be affected. When there are more flights, more connecting flights can be adjusted

Table 4.1: Number of evaluation moments per day in the evaluation set.

Category	**Date**	**Increased tows**	**Physical conflicts**	**Total evaluation moments**
Busy days	2019-07-07	93	103	196
	2019-07-26	119	174	283
Quiet days	2021-05-27	25	12	37
	2021-04-04	8	13	21
Disturbed days	2022-04-01	243	237	480
	2022-04-27	288	250	538

which results in more moments where the number of tows is increased. Another cause of the large number of flights, is that there are more ramps occupied at the same time and that there is less time between two consecutive flights at the same ramp. When there is less time between two flights, a deviation in the block time of a flight sooner causes a physical conflict.

Besides that, it can be observed that the detection mechanisms are most frequently triggered on the disturbed days in the evaluation. This is caused by the fact that the connection between the flights is changed a lot by the airlines in order to reduce the delays as much as possible. So even though these days have less flights than the busy days, the changing circumstances result in more evaluation moments than the other categories in the evaluation set. This means that the model will run most frequently when collecting the results for the disturbed days.

Chapter 5

Results

The performance and the capabilities of the model are tested and evaluated on a set of historical days. Chapter 4 discussed that the evaluation set is divided into three categories; a day with a lot of flights, a day with fewer flights and a day with a lot of disturbances. To get a better view on how well the model fits the different requirements of the gate planners, several experiments are executed. To get an indication of the scheduling capabilities of the model, a stress test is executed. The performance of the model is analysed by adjusting the weights in the objective function of the model. Finally, the results of an alternative technique to generate multiple solutions with the model for the same weights at the same moments are analysed. This chapter discusses the results of the different experiments to obtain a complete view of the proposed method.

5.1 Stress Test

Subsection 3.4.1 introduced two ways to reduce the number of parameters in the model. To analyse how the performance of the model is affected by restricting the time and space parameters, the solving times of the model with different combinations of parameters are collected. For every combination of parameters, the optimal schedule is calculated for six fixed moments per day with two hours among the moments. The first moment of the day the optimal schedule is calculated, is at 06:00 because this is often just before the first flight of the day departs or arrives. The process has been executed for each day in the evaluation set. Figure 5.1 and 5.2 show the average solving times for both the CBC solver and the Gurobi solver, respectively. When looking at the plots, a similar trend can be observed. As the number of variables increase due to the space and time parameter, the solving times increase significantly. Furthermore, it can be seen that there are a few empty spots. This is caused by the limitations of the solvers, for those configurations the solvers are unable to find a solution because of the size of the model. When comparing the actual solving times between the different solvers, it can be seen that the solving times of the CBC solver are much higher than the ones of the Gurobi solver.

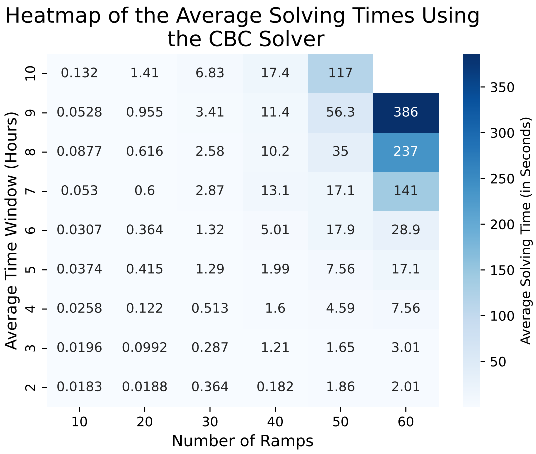

Figure 5.1: Heat plot of the average solving times for the different scenarios using the CBC solver.

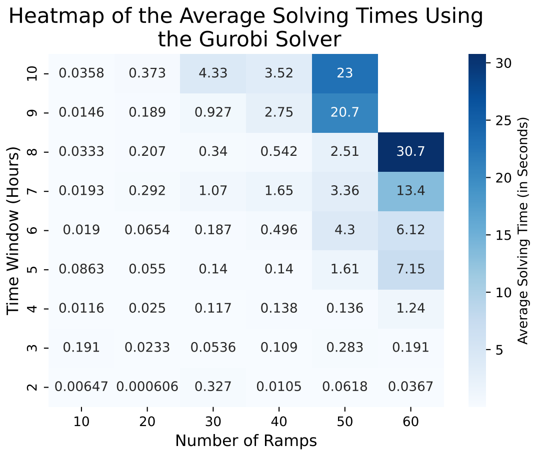

Figure 5.2: Heat plot of the average solving times for the different scenarios using the Gurobi solver.

For the remainder of the research, the results of the model that incorporates the flights of 60 ramps and flights for the next six hours are gathered since this gives the model a lot of flexibility with a reasonable solving time.

5.2 Model Performance

One of the main challenges during this research, is that the optimal schedule yet cannot be quantified. Some of the components that are important for the gate planners are incorporated in the model, but the weights between these components are unknown. Even though the solutions of the model are going to be used by the gate planners, the model should not mimic their solutions since it is unknown if these solutions are truly optimal. For this reason the solutions of three variants of the linear model are compared. The three variants can be distinguished by the weights of the components in the objective function (α_o).

- The first variant is the *baseline model*. The weights of the components of this model are all set to one. This model is mainly used to give more context to the other models. Comparing their solutions gives a better view on the impact of the adjustments in the weights of the objective.
- The next variant is called the *tow reduction model*. The model aims to reduce the number of required tows of its solutions by giving the model an extra reward for

schedules with less required tows of aircrafts to other ramps. The weight of the tow score component (α_2) is increased to five while the weights in the objective stay one.

- The *handler clustering model* is rewarded when a schedule contains a lot of consecutive flights with the same ground handler. The reward for the weight of the handler clustering score component in the objective (α_3) for this model is increased to five while the other components are equal to one.

These three variants of the model are run according to the described logic in Section 4.2. The results of the models are discussed per category of the evaluation set. This enables one to analyse whether the behaviour of the models is consistent over the different categories.

5.2.1 Quiet Days

The first category that is analysed are the days of the quiet category. The analysis is started by comparing the score of the different components of the objective. This has been done using the radar chart in Figure 5.3. The difference between the score components of the different models and the baseline model are plotted in the same radar chart.

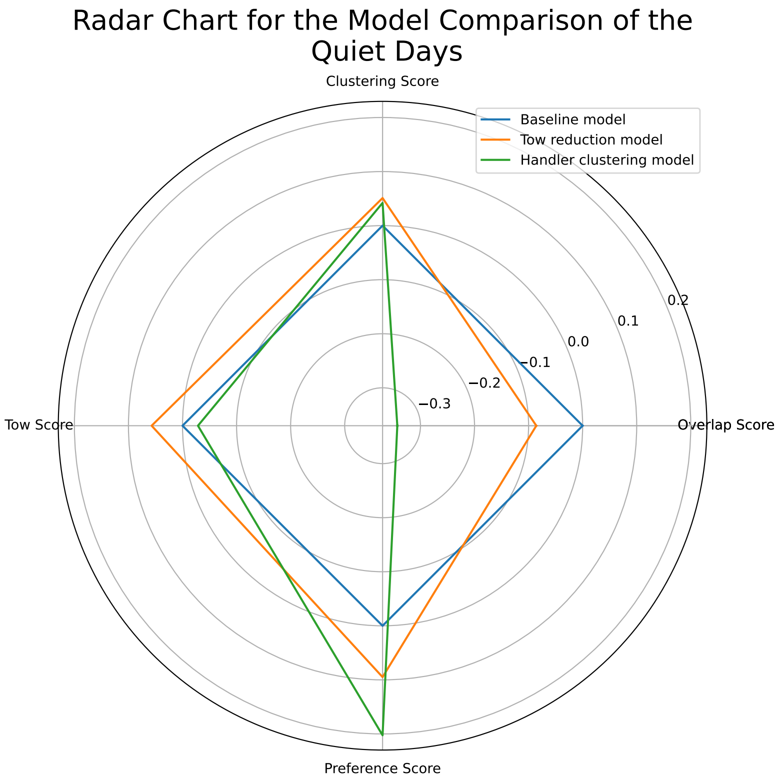

Figure 5.3: Radar chart of the comparison of the different models for the quiet days.

Based on Figure 5.3 it can be concluded that the performance of the three models is similar since the differences are small. One of the larger differences between the models can be seen in the overlap score. This score gives an indication of the complexity of the solution as a lower overlap between the initial schedule and the new schedule means that there are more flights rescheduled. The tow reduction model and the handler clustering model on average make more changes than the baseline model. Comparing this to the average of 1.37 changes per moment of the baseline model, the increases are rather small. This difference can be explained by the fact that the baseline model does not make any change for 68% of the moments, while the tow reduction model and the handler clustering model do not make changes for 55% and 43% of the moments, respectively. Furthermore, it can be observed that the tow reduction model exceeds the cluster score and tow score of the other models. The fact that the tow reduction model outperforms the handler clustering based on the clustering score was not expected since the handler clustering model receives an additional reward for this component.

The difference in the components of the objective scores between models suggest that the models have different solutions. However, looking at the components does not give insights in how the solutions differ from each other. For example, when the baseline model reschedules an aircraft to ramp D47 and the tow reduction model reschedules the same aircraft to D49, it is a different solution. But since it is the neighbouring ramp, it is considered by the gate planners to be a similar solution to the solution of the baseline model. In order to get a better view on the similarity between the different models, the involved flights and the destination pier of those flights are studied.

When both the baseline model and the tow reduction model decide to make adjustments to the planning, 85% of the same flights are involved. These flights are moved to the same pier in 100%. When comparing the decisions of the handler clustering model to the decisions of the baseline model, 40% of the flights match. Just as in the comparison between the baseline model and the tow reduction model, are the same flights in the handler clustering model always moved to the same pier as the baseline model.

Comparing the solutions of the model with the decisions of the gate planners, it can be observed that the scores for the preference score and the tow score are rather similar. The models do significantly better in terms of ground handler clustering. The baseline model scores 15.8 points higher compared to the decisions of the gate planners. This is achieved by making on average 0.6 more changes to the previous schedule. Another difference between the model and the gate planners, is that the gate planners react on average after 8 minutes while the model responds on average in 3 minutes.

5.2.2 Busy Days

The busy days of the evaluation set are analysed in the same way as the quiet days. Similarly to Figure 5.3, Figure 5.4 shows the performance of the models with different weights per objective component.

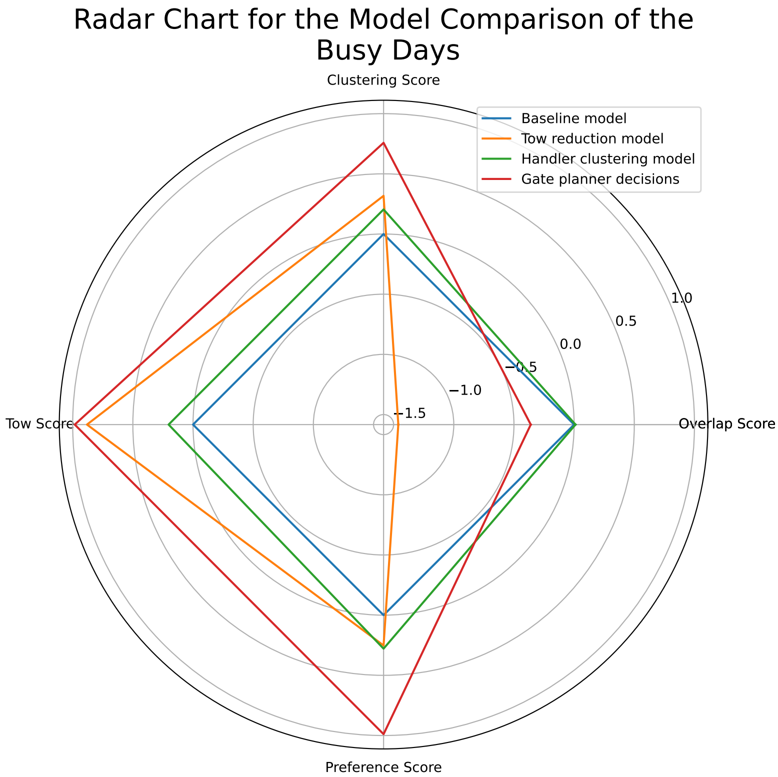

Figure 5.4: Radar chart of the comparison of the different models for the busy days.

When comparing the three models in the radar chart in Figure 5.4, it can be seen that the baseline model is outperformed by the other two models by a small margin. The handler clustering model makes slightly less changes than the baseline model, but scores approximately 0.2 points higher on the other components. The shape of the tow reduction model differs from shapes of the other models. The average of 3.9 changes per moment of the tow reduction model, results in a significant increase of the tow score compared to the other models.

The differences of the shape of the results of the models in the radar chart, indicate that the decisions of the tow reduction model deviates notably from the decisions of the other models. This is confirmed by the overlap of the involved flights between the baseline model and the tow reduction model. The tow reduction model and the baseline model reassign 62% of the same flights. However, the deviation in piers is much smaller since 99% of those flights is assigned to a ramp at the same pier. Even though the shape of the baseline model and the handler clustering model is more similar, the overlap of the involved flights in the decisions is 76%. These flights are moved to the same ramp in 98% of the cases.

The shape of the performance of the decisions of the gate planners is most similar to the shape of the handler clustering model. The additional number of changes result in an increase of at least 0.6 point on the other components. This could be caused by the fact that the model is limited to rescheduling flights in the next six hours for a subset of ramps while the gate planners are able to adjust any flight of the day. Another observation that can be made when comparing the solutions of the gate planners to the solutions of the models, is that the gate planners make changes which are not allowed for the model. For example, the gate planners decide to schedule a Schengen flight on a Non-Schengen ramp and solve this by transferring the passengers of that flight to a Schengen ramp with busses to reduce the number of tows. Due to the additional complexity and larger amount of evaluation moments compared to the quiet days, the gate planners respond on average in 16 minutes. Even though the solving time of the model increased to 8 minutes, it is still able to respond faster to the changing situations.

5.2.3 Disturbed Days

The final category of evaluation days are the days with a lot of disruptions. How the different models and the decisions of the gate planners perform on the disturbed days is visualised in the radar chart in Figure 5.5.

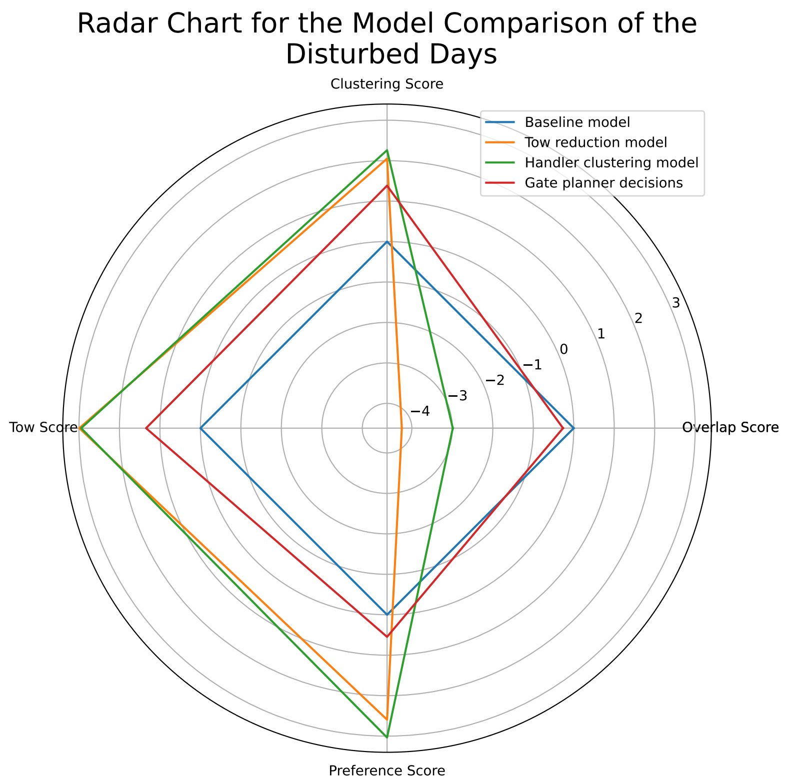

Figure 5.5: Radar chart of the comparison of the different models for the disturbed days.

The radar plot in Figure 5.5 shows that adjustments in the weights in the objective function have a lot of impact on the performance of the decisions of the model for disturbed days. This can be seen by the fact that the scores of the different components of the tow reduction model and the handler clustering model differ notably from the shape of the baseline model. As the models increase the amount of changes, the other components increase as well.

This is confirmed by analyzing the overlap of the involved flights of the different models. The decisions of the tow reduction model and the handler clustering model overlap for 56% and 41% of the flights with the decisions of the baseline model, respectively. The destination of these overlapping flights does not differ a lot since for both models 98% of the overlapping flights are moved to the same ramp as the baseline model.

The scores of the decisions of the gate planners do not match the patterns of any of the models. The gate planners make on average comparable number of changes as the baseline model, but achieves a higher clustering score and tow score. These differences can be explained by the same reasons as mentioned earlier, the gate planners are not as restricted as the decisions that are made in the model, which results in different solutions. Furthermore, the gate planners do not always follow the rules as stated in the RASAS document [11], resulting in solutions that would otherwise be infeasible. Due to the additional responsibilities of the gate planners, the average response time is 17 minutes while the model responds in 6 minutes.

5.3 Generating Multiple Solutions

The final requirement for the model is to give the gate planners the option to choose from multiple solutions. The previous section showed that this could be achieved by adjusting the weights of the components in the objective function. An alternative is to run a model with the same weights in the objective multiple times and excluding the found solution from the solution space after each run. This is achieved by adding constraints to the model that sets the variables x_{ij} for the flights that are rescheduled to a new ramp by the model equal to zero. This forces the model to find a new solution for every iteration. The new solution has a lower score than the previous solution, but the newly generated solution could be a better solution according to the gate planner because it could coincidentally take components into account which are not incorporated in the model.

The results for the analysis of the alternative solutions are gathered in a similar way as the results of the alternative models. The same moments as described in section 4.2 are used, except only the moments where the models in the previous section decided to adjust the initial planning are evaluated. This exception is included since for these moments, it is known what constraints should be added to force the model to find a new solution.

For the last part of the results, only the tow reduction model and the handler clustering model are evaluated since they outperform the baseline model on most of the components for every day category. This makes it more interesting to analyse what the impact of generating alternative solutions for these models is. For each moment, both models tried to find ten unique solutions. It could be possible that the additional restrictions limit the solution space too much such that the problem becomes infeasible.

5.3.1 Quiet Days

The first days that are analysed, are the day in the quiet category of the evaluation set. Since the models did not make any changes for the moments on the quiet days, the amount of moments is reduced to 26 for the tow reduction model and 33 for the handler clustering model. This is a considerable lower amount than the number of moments for the other categories. This number is lowered even more since 58% of these moments the tow reduction model was able to generate ten solutions. When looking at the remaining 42% of the moments, it was noticed that for the majority of the cases (67%) the problem became infeasible after six iterations of generating a new solution. For the handler clustering model this was less of a problem since the model was able to produce ten solutions in 76% of the moments. To get a better understanding on the quality of the different solutions, the average reward per iteration for the tow reduction model and the handler clustering model are plotted in Figure 5.6 and Figure 5.7, respectively.

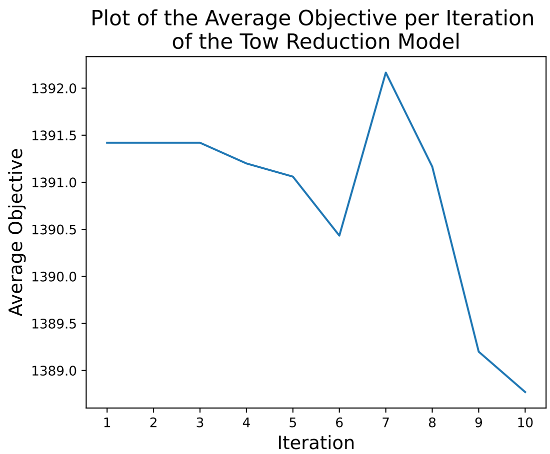

Figure 5.6: Plot of the average objective per iteration of the tow reduction model for the quiet days.

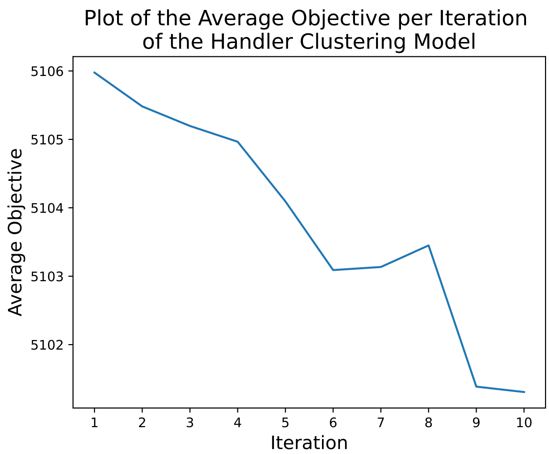

Figure 5.7: Plot of the average objective per iteration of the handler clustering model for the quiet days.

From looking at Figure 5.6 and Figure 5.7 different patterns can be observed. The drop in the average objective of the tow reduction model in Figure 5.6 at the sixth iteration, can be explained by the fact that the tow reduction model was able to produce six results for a large part of the moments. This is confirmed by looking at the objectives of these moments. The tow reduction model solved these moments by making more adjustments than the other moments, which results in a lower objective.

A similar trend between the two plots, is the drop in the objective value after the seventh iteration. The drop in the objective value could suggest that these solutions of the model differ more from the first solution than solutions with a similar objective value. This is tested by measuring the distance between two solutions using the percentage of the same flights that are involved in the solutions of the models. When comparing the overlap of the involved flights of the solutions of the model for every iteration in Table 5.1 to the solutions of the first iteration, a similar trend can be observed. Even though the amount of changes does not increase, the overlap of the involved flights drops from approximately 80% at the seventh iteration to an overlap of 64% of the involved flights at the eighth iteration for both models. The reduction of the overlap continues to 51% at the tenth iteration.

Table 5.1: The percentage of overlapping flights of the solutions of the first iteration and the other iteration on quiet days.

	Iteration								
model	2	3	4	5	6	7	8	9	10
Tow reduction model	82%	78%	76%	76%	72%	81%	64%	63%	62%
Handler clustering model	86%	86%	83%	81%	80%	82%	64%	63%	63%

5.3.2 Busy Days

The same analyses as for the disturbed days are executed for the busy days of the evaluation set. Even though there are more ramps occupied at the same time on busy days than the quiet days, both models were able to produce ten solutions in 99% of the evaluation moments. How the objective value is affected over the iterations for the models, is visualised by plotting the mean of the objective per iteration. Figure 5.8 shows the average objective value for the tow reduction model and Figure 5.9 shows the same for the handler clustering model.

In contrast to the quiet days, where the average objective value dropped for solutions for higher iterations, the average objective value on the busy days appears to decrease linearly for both models. Although the average objective value for both models decrease steadily, the plots show a sudden increase at one point. Where the increase in the average objective value occurs at the third iteration for the tow reduction model, the mean objective value of the handler clustering model increase at the eighth iteration. This disruption of the trend could be caused by the model being stuck in a local optimum where the models escape from because of the added constraints.

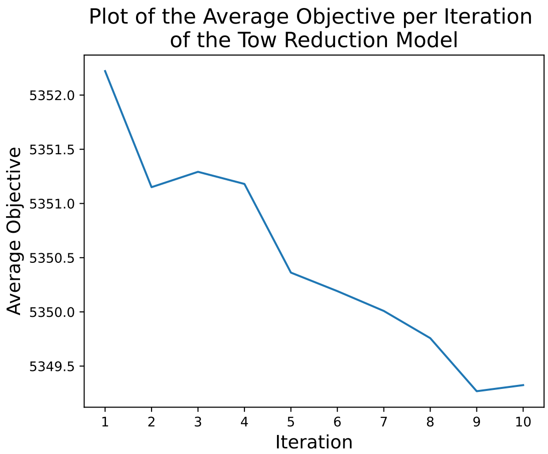

Figure 5.8: Plot of the average objective per iteration of the tow reduction model for the busy days.

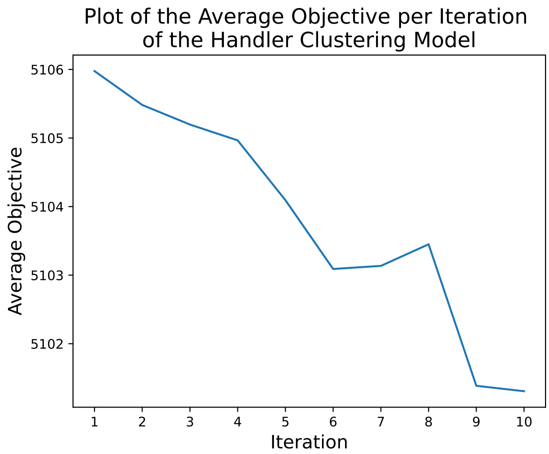

Figure 5.9: Plot of the average objective per iteration of the handler clustering model for the busy days.

To obtain more insights in the similarity of the solutions, the overlap between the involved the flights of the decisions of the first iteration (the optimal solution) and the involved flights of solutions of other iterations are calculated. The percentages are shown in Table 5.2. Comparing the trend of the percentages to the trends of the objective values, it can be concluded that they roughly follow a similar trend. As the average objective value decreases, the overlap percentage decreases as well. Furthermore, it can be noticed that the overlap percentages are relatively low. The second iteration already have overlap percentages of 62% and 69% with the first iteration of the handler clustering model and the tow reduction model, respectively.

Table 5.2: The percentage of overlapping flights of the solutions of the first iteration and the other iteration on busy days.

	Iteration								
model	2	3	4	5	6	7	8	9	10
Tow reduction model	62%	54%	50%	46%	54%	51%	48%	46%	42%
Handler clustering model	69%	59%	56%	70%	65%	56%	55%	56%	56%

5.3.3 Disturbed Days

The final category of the evaluation set, are the days with a lot of disruptions. Starting with the amount of solutions the tow reduction model and the handler clustering model are able to produce, it is observed that both models are able to produce ten solutions for almost all the evaluation moments (92% and 91%, respectively). For the moments where the model is unable to generate the ten solutions, still eight solutions were generated before the problem became infeasible. To get a better understanding on the quality of the different solutions, the average reward per iteration for the tow reduction model and the handler clustering model are plotted in Figure 5.10 and Figure 5.11, respectively.

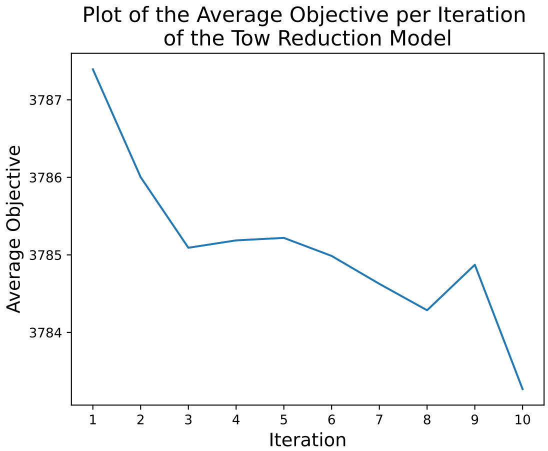

Figure 5.10: Plot of the average objective per iteration of the tow reduction model for the disturbed days.

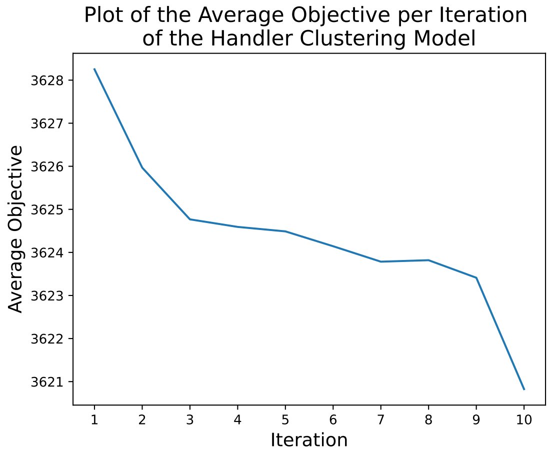

Figure 5.11: Plot of the average objective per iteration of the handler clustering model for the disturbed days.

Figure 5.10 and Figure Figure 5.11 show a similar trend. The objective value decreases faster for the solutions of the second and third iteration. Thereafter, the average objective value is steadily decreases until the tenth iteration. The average objective value of the last iteration shows a drop off comparable to the first few iterations.

Comparing the pattern of the average objective values to the overlap percentages between the first iteration and the other iterations in Table 5.3, some similarities can be observed. The overlap percentages of the handler clustering model show a similar drop off for the second and third iterations from 82% to 70%. However, while the objective is further reduced, the overlap remains approximately 70%. This is similar for the average objective values and the overlap percentages of the clustering model. The overlap between the involved flights for the solutions of the first and second iteration is already dropped to 71%, but stays about the same for the remainder of the iterations.

Table 5.3: The percentage of overlapping flights of the solutions of the first iteration and the other iteration on disturbed days.

	Iteration								
model	2	3	4	5	6	7	8	9	10
Tow reduction model	82%	70%	70%	69%	66%	70%	63%	62%	64%
Handler clustering model	71%	72%	75%	74%	74%	71%	69%	67%	68%

Chapter 6

Discussion

This study aims to extend the existing literature of the GRP. In this study, an ILP was used to limit the number of adjustments to a gate schedule created by human gate planners. The model focuses on incorporating the human aspect in the gate assignment process to support the gate gate planners in their decisions. The method was tested on six historical days which van be separated in three categories; a day with a lot of flights, a day with fewer flights and a day with a lot of disturbances. The variation in the evaluation days help to get a better understanding of the performance of the model in different scenarios.

In this section, the key results are interpreted, after which the limitations of these results are discussed. In addition, a number of recommendations for future research are given.

6.1 Interpretation of Results

As mentioned, an ILP was used to find the optimal gate schedule. In Section 5.1, it was shown that the Gurobi solver is able to find the optimal schedule faster than the open source solver CBC. Furthermore, it shows correlation between the number of variables and the solving time, logically, more variables result in a more complex model, and thus a higher solving time. For the model to make a suggestion to the gate planners on how a conflict could be solved, it needs to be done with its calculations before the gate planners make a decision themselves. On the other hand, it is important for the quality of the solutions of the model to provide enough flexibility in the solution space. Considering this trade-off between the solving time and the flexibility of the model, a model that incorporates the flights for the coming six hours for 60 ramps is chosen to be a best fit since the CBC solver was able to find the optimal solution after on average 29 seconds. As this run-time is an average, in reality it could be the case that the model runs for a long time, which means that the gate planners could have already found a better solution in the meantime, thus it could be valuable to introduce a time limit. With this time limit it is possible to determine a maximum run-time in which the model is restricted to find a solution. When the time limit is reached, the model will return the best found solution.

By comparing the results of the solutions of the model variants in Section 5.2 to the solutions of the gate planners, it can be concluded that incorporating the previous schedule in the objective limits the number adjustments of the model. When making other components in the objective more important by increasing their weights, it can be seen that the model makes more adjustments to the initial planning. Therefore, it is important that the contribution of the overlap between the previous planning and the adjusted planning to keep the decisions of the model comprehensible for the gate planners.

However, when comparing the scores of the solutions of models on the different components to the scores of the scores of the solutions executed by the gate planners, less similarities can be observed. This indicates that their solutions notably differ. The difference in the solutions is caused by different reasons. As mentioned earlier, the model is limited in what flights can be rescheduled while the gate planners are able to reschedule every flight on a day. Another reason behind the difference between the solutions of the gate planners and the model is the difference in what components are considered in making a decision. The components incorporated in the model are a foundation to measure the quality of the solutions, but there are more components that should be incorporated to increase the reality and usability of the solutions of the model. Besides that, the optimal set of weights should be found to further improve on these points.

The requirement of generating multiple results at the same moment is tested using two different techniques. The first technique to generate multiple results, is to change the weights of the components in the objective function. As mentioned earlier, the general effect of increasing the weights of the tow score and the handler clustering score is that the scores of every performance component increase by making more adjustments. How much the solutions of the models differ from each other is expressed in the overlap of the flights that are reassigned to another gate compared to the initial planning. When comparing these percentages to the overlap percentages for the alternative technique where the model with static weights generates multiple solutions, overall similar results are obtained. Nevertheless, when comparing the results of the different iterations over the different categories no clear trend can be observed. Therefore, this method is considered to be inconsistent and the method of adjusting the weights in the objective function is preferred. Another benefit of adjusting the objective weights, is that the models can be solved in parallel to obtain the different scenarios. When using the iterative approach, the previous iteration has to be finished before a new solution can be generated because based on the previous solution new constraints are added. Therefore, the parallel approach saves time and reduces the chance that the gate planner already found a solution by himself.

6.2 Limitations

The results of the model are gathered from six days. Even though the days are selected from three categories to obtain insights in the behaviour in different scenarios, six days may be considered as not enough days to obtain reliable results. The gathered results can be considered as a starting point from which the effect of adjustments can be observed. Before the gate planners can rely on the solutions of the model, the model needs more testing. Another way the quality of the model can be tested and more insights in the decision process of the gate planners can be obtained, is by comparing the solutions of the model to the solutions of the gate planner together with the gate planners. The gate planners can tell why their solution is better than the solution of the model.

Another limitation of the model, is that the model is limited to reschedule the flights for the next six hours for a subset of 60 ramps. When there is a conflict for flights that arrive or depart more than six hours ahead, the model is not able to support the gate planner with a solution. When flights further into the future or flights from a larger subsets of ramps are considered, the model needs more time to find a solution which is not desirable.

6.3 Recommendations

Even though the results of this study are regarded to be successful, this model should be considered to be a basis of which several components should be added before it can be used by the gate planners. Moreover, this study acts as a benchmark for future research, for which a number of recommendations are given below.

As previously mentioned, the solutions of the model cannot be incorporated in the gate assignment process yet. The quality of the solutions is assumed to be expressed as a linear combination of different components. Since this is also unknown for the gate planners themselves, it could be possible that another form is a better fit to the needs of the gate planner. For further research it would be valuable to see whether a non-linear objective has a better fit on the decisions of the gate planners.

Aside from the form of the objective function, a good schedule takes several other components into account. Components that are taken into account by the gate planners, but not are not incorporated in the model are components like the impact of the schedule on the number of passengers at the security filters. Other components that would improve the quality of the gate planning, is adding a robustness parameter to reduce the future work due to new conflicts caused by further delays of the block times of flights. Another component the gate planners take into account while making a planning, is the so-called 'best fit'-principle. With this principle, the gate planners aim to assign larger aircrafts to larger ramps. A final factor the gate planners take into account while making a gate planning, is to prevent a large aircraft to be handled directly after a small aircraft is handled at the same ramp.

From the analysis of the solutions of the gate planners, it could be concluded that the gate planners fail to comply with the rules as defined in the RASAS document quite frequently. Which would lead to infeasible solutions in the model. One way to tackle this, could be to introduce soft constraints. This means that the rules could be violated, however this results in a penalty in the objective function. With this, it is also possible to differ in the weights, which enables the user to penalize particular violations more than others.

When all the components are added to the model and the form of the objective function is found, it is important to balance the components to obtain reliable and realistic results. This study has shown that the weights of the different components impact the decisions of the model. Finding the optimal weights is an iterative task where the weights should be adjusted based on the feedback of the gate planners on the solutions of the model.

Currently, as stated earlier, one of the limitations is that the model becomes computationally too heavy to run when more than 6 hours are considered. As a final addition to the current approach, it could be of value to run the model multiple times for connecting periods, this allows the user to create the schedule further ahead in the future. Considering a run-time of 30 seconds for six hours on average, finding the schedule using connecting periods would still be acceptable in terms of run-time.

Aside from potential improvements of the objective function of the model, it could be possible that an ILP might not be the best method to solve the gate planning. The deeper the knowledge which is extracted from the gate planners, the more knowledge about the gate planning process is gathered. This knowledge also comes with an increasing amount of exceptions. The Schengen/Non-Schengen constraints seem to be hard constraints in the beginning for example, but the gate planners eventually noted that it is possible to schedule a Schengen flight to a Non-Schengen ramp and transfer the passengers to the correct location using busses. As more of such cases appear, the gate planning might be better solved using a heuristic approach. Comparing the solutions of such model to the proposed method in this study could answer what approach is a better fit.

The implementation of these enhancements may lead to more realistic and reliable solutions which would increase the added value when the solutions of the model are incorporated in the gate scheduling process of the gate planners. On the other hand, these enhancements could lead to larger solving times because of the added complexity. This could lead to a further reduction of the solving space to keep the short solving times. Therefore, it is important to test the impact of the adjustments before incorporating the model in the gate scheduling process.

Chapter 7

Conclusion

In Chapter 5, an integer linear programming (ILP) model was evaluated on the different requirements in order to answer the research question that was introduced in Chapter 1:

How can a mathematical model optimize the gate planning on the day of operations by making adjustments in the gate planning in order to help the gate planners?

Considering the results presented in this study, it can be concluded that the proposed model, the ILP model, is a good method to solve this problem, since the model is able to optimize the plan for flights of 60 ramps for the next six hours within a reasonable solving time. Since this is a large part of the airport, the method meets the requirement of the solving capabilities. In addition, the solutions of the proposed model are considered to be comprehensible because the number of adjustments of the planning were comparable to the solutions of the gate planners. This was achieved by incorporating the overlap between the new schedule in the objective function. Finally, the requirement of generating multiple suggestions at the same moment is tested using two methods; changing the weights in the objective function and an iterative approach. In the iterative approach for each iteration the previous solution is excluded from the solution space by adding constraints, resulting in multiple different solutions. The methods showed comparable results, but the main advantage of changing the weights in the objective function over the iterative approach is that it can be executed in parallel. Furthermore, changing weights enables the gate planners to focus on a particular component on one day, for instance the number of tow movements, while on other days another objective can be prioritized.

However, the proposed model is not ready to be adopted in the gate planning process yet since a not all the decision components of the gate planners are incorporated in the model. Moreover, this study should be considered as a benchmark where an integer linear program is introduced that can be easily expanded and adjusted to better fit the requirements of the gate planners.

References

[1] G Diepen, JM Van Den Akker, JA Hoogeveen, and JW Smeltink. **Finding a robust assignment of flights to gates at Amsterdam Airport Schiphol**. *Journal of Scheduling*, **15**(6):703–715, 2012. 3

[2] Ulrich Dorndorf, Florian Jaehn, and Erwin Pesch. **Modelling robust flight-gate scheduling as a clique partitioning problem**. *Transportation Science*, **42**(3):292–301, 2008. 4

[3] Ulrich Dorndorf, Florian Jaehn, and Erwin Pesch. **Flight gate scheduling with respect to a reference schedule**. *Annals of Operations Research*, **194**(1):177–187, 2012. 4

[4] Binod Maharjan and Timothy I Matis. **An optimization model for gate reassignment in response to flight delays**. *Journal of Air Transport Management*, **17**(4):256–261, 2011. 4, 17

[5] Shangyao Yan, Ching-Hui Tang, and Yu-Zhou Hou. **Airport gate reassignments considering deterministic and stochastic flight departure/arrival times**. *Journal of advanced transportation*, **45**(4):304–320, 2011. 4

[6] Dong Zhang and Diego Klabjan. **Optimization for gate re-assignment**. *Transportation Research Part B: Methodological*, **95**:260–284, 2017. 4

[7] Geoffrey D Gosling. **Design of an expert system for aircraft gate assignment**. *Transportation Research Part A: General*, **24**(1):59–69, 1990. 4

[8] Hakki Murat Genç, Osman Kaan Erol, Ibrahim Eksin, Mehmet Fatih Berber, and Binnur Onaran Güleryüz. **A stochastic neighborhood search approach for airport gate assignment problem**. *Expert Systems with Applications*, **39**(1):316–327, 2012. 4

[9] Chun-Hung Cheng, Sin C Ho, and Cheuk-Lam Kwan. **The use of meta-heuristics for airport gate assignment**. *Expert systems with applications*, **39**(16):12430–12437, 2012. 4

[10] Wu Deng, Huimin Zhao, Xinhua Yang, Juxia Xiong, Meng Sun, and Bo Li. **Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment**. *Applied Soft Computing*, **59**:288–302, 2017. 4

[11] Schiphol. **Regulation Aircraft Stand Allocation Schiphol (RASAS)**, Nov 2020. 15, 33

[12] Rico Schröder. **Terugblik: De Zware Sneeuwstorm van Februari 2021**, Feb 2022. 25