Maastricht University

School of Business and Economics

Slot pair optimisation at Amsterdam Airport Schiphol

Q.H. Vû

Supervisor UM: C.P.M. van Hoesel
Supervisor AAS: R. van Schaik

MSc Econometrics and Operations Research

Electronic copy available at: `https://huyvu.nl/research/`

Schiphol, October 17, 2016

Schiphol
Amsterdam Airport

Foreword and acknowledgements

This thesis is written as partial fulfilment of the requirements for the degree of Master of Science in Econometrics and Operations Research at the Maastricht University School of Business and Economics.

I would like to thank Amsterdam Airport Schiphol (AAS) and Schiphol Group, especially the department Aviation Marketing and my sub-department Traffic Analysis and Forecasts for the possibility to do my graduation thesis here. I had a great time at the department, with very helpful colleagues, nice discussions about aircraft, a nice view on the runways and the funny activities we did at the airport. I would like to thank in special my supervisor Ramon van Schaik to set-up this assignment and the valuable feedback and support he gave during my time at the department.

I would like to thank University Maastricht and my supervisor Stan van Hoesel for his feedback and guidance for this thesis.

Huy Vû

This master's thesis has been adapted for online publication. Parts may be removed or replaced with fictitious names or numbers.

Electronic copy available at: `https://huyvu.nl/research/`

Table of Contents

Chapter 1

Introduction

Amsterdam Airport Schiphol (IATA code: AMS) is the biggest airport in The Netherlands and the fifth biggest in Europe with 58.2 million passengers in 2015 and 450.679 flight movements [1, 2]. In 2015 from Schiphol 322 destinations with 108 airlines could be reached with direct flight, including cargo [3].

According to the market forecast of the two biggest aircraft manufacturers, Airbus and Boeing, in the next 20 years the demand for air transportation will rise 4.7% a year for passenger traffic and 4.2% for freight traffic while air traffic is expected to double in 20 years [4, 5].

The growth in aviation will have its limitations: the demand at airports might be bigger than the available capacity (also at Amsterdam Airport Schiphol). In the Netherlands the Omgevingsraad Schiphol (previously the Alderstafel) discusses the developments in aviation and the environment of Schiphol Airport. Up to year 2020, flight movements (arrival and departures) are limited to 500.000 a year (whereof 32.000 during night between 23:00 and 07:00 local time (LT)) that is known as the Aldersakkoord [6] which is set due to operational, but mainly because of environmental capacities to limit the noise emissions to the environment.

1.1 Problem statement

Since the capacity is limited at Schiphol Airport, slots are required. Airlines need a slot for arrival (landing) and a slot for departure (take-off) to be able to operate at the airport. An arrival and departure slot combined is called a slot pair.

Airlines prefer a consistent flight schedule, that is a flight schedule where the scheduled arrival and departure times are the same throughout the season (perfect slot pairs). Since the number of perfect slot pairs at Schiphol Airport are limited, slot requests of airlines will be rejected or slots will be returned by the airline, since they have to fly at undesirable times in case the requested slot pairs are not available.

This thesis will investigate what the influence on the number of possible slot pairs will be when the requirements of perfect season-round (operating one day-of-week throughout the season) slot pairs are relaxed, so when (small) deviations in flight schedules are allowed.

The problem in this thesis can be divided into three parts:

- First will be analysed how airlines use their flight schedules. Do airlines use all-season-round a slot pair at the same time, how does an airline judge a deviation in a season constant slot usage? With this part the 'value' of a slot for an airline will be determined and how much deviation for a schedule is allowed.
- Each airline will have its own preferences. When for a flight a slot pair is requested and given that this is not available season-round, what is the best alternative for this flight? And in case that multiple flights want the same slots, what is the most optimal method to allocate the available slots, how do flights influence each other?
- Finally, given the preferences of an airline, how many possible slot pairs are left that satisfies these preferences?

This thesis will use data of summer season 2015 (S15) and winter season 2015/2016 (W15), since the data for these seasons are final. By changing the input data, the model will be applicable to other seasons.

1.2 Thesis outline

The thesis is built-up as follows: first an introduction to airports slots will be given in chapter 2, what are slots, when are slots required and how are slots allocated. In chapter 3 two related articles for the slot allocation at Schiphol Airport will be presented and discussed. In chapter 4 the first part of the problem statement will be addressed: how do airlines use their schedules?

In chapter 5 an introduction to linear programming will be given with exact solving methods, because in chapter 6 a linear program will be formulated to determine the best alternative flight schedule, which is the second part of the problem statement. In chapter 7 the exact results of the model will be demonstrated and discussed.

In chapter 8 the last part of the problem statement will be addressed: how many slot pairs are possible? Finally in chapter 9 some cases for which the model has been used at Schiphol will be given and in chapter 10 will be ended with a conclusion and recommendations for future research.

Chapter 2

An introduction to airport slots

In this chapter an introduction will be given what an airport slot is, when a slot is required and how slots are allocated.

2.1 Slot allocated airports

The International Air Transport Association (IATA) has three levels for airport coordination [7]. Coordination involves the allocation of constrained or limited airport capacity to airlines and other aircraft operators to ensure a viable airport and air transport operation. Coordination is also a process to maximize the efficient use of airport infrastructure. However, coordination is not a solution to the fundamental problem of a lack of airport capacity [7]. IATA states that coordination should be seen as an interim solution to manage congested infrastructure until the longer term solution of expanding airport capacity is implemented. The objective of airport coordination is to ensure the most efficient use of airport infrastructure in order to maximize benefits to the greatest number of airport users. The three airport levels are:

- Level 1 airports (non-coordinated airports) have no capacity constraints, the airport is able to meet the demand of airport users at all time
- Level 2 airports (schedule facilitated airports) have potential congestion during some periods (day/week/season) that can be solved by scheduling adjustments in agreement between airlines and the airport facilitator
- Level 3 airports (fully coordinated airports) have exceeding demand and no solution can be expected in short time. At these airports a slot coordinator will allocate slots to airlines and other aircraft operators that are using or planning to use the airport as a means of managing the declared capacity

For a Level 3 airport slots are allocated to operate. The definition of an airport slot according to IATA is: *"An airport slot (or 'slot') is a permission given by a coordinator for a planned* *operation to use the full range of airport infrastructure necessary to arrive or depart at a Level 3 airport on a specific date and time."* [7]

On the 10th of February 2016, there are 179 coordinated airports and 122 schedule facilitated airports [8]. In The Netherlands there are three fully cordinated airports: Amsterdam Airport Schiphol (AMS), Rotterdam-The Hague Airport (RTM) and Eindhoven Airport (EIN) [8, 9]. Groningen Airport Eelde (GRQ), Enschede Airport Twente (ENS), Maastricht-Aachen Airport (MST) and Lelystad Airport (LEY) are non-coordinated airports [9]. There are no Level 2 airports in The Netherlands. In The Netherlands the slot coordination is done by Stichting Airport Coordination Netherlands (SACN) [9].

2.2 Demand and capacity analysis

Schiphol Airport and Luchtverkeersleiding Nederland (LVNL) determines the declared capacity based on environmental and operational capacity. Environmental capacity is determined by the noise limitations and the operational capacity is determined by the capacity of the runways (number of runways, configuration, weather, etc.), terminals (gates, baggage handling, etc.) and Air Traffic Management (ATM) (separation between aircraft) and is given by the maximum number of aircraft movements that the airport can handle within a given timespan, which is called a bracket. Each possible movement in a bracket is called a slot. The number of slots per bracket at Schiphol Airport can be found in appendix A.

2.3 Key principles of slot allocation

To operate at a fully coordinated airport, the airline or aircraft operator should have an allocated slot for each movement. Airlines can request slots at the slot coordinator and the coordinator will allocate those in a neutral, transparent and non-discriminatory way.

A series of slots is at least five slots requested for (approximately) the same time on the same day-of-week, distributed regularly in the same season if possible [7]. If a flight operates on more than one day-of-week, then each day-of-week is considered as a separate series of slots.

When an airline operates (and can demonstrate that) a series of slots at least 80% of the time during the period allocated in the previous equivalent season, the airline is entitled to retain this series of slots on basis of historic precedence (also called grandfather rights (GFR) or the 80/20 rule). The slot coordinator is not allowed to withdrawn these slots to accommodate new entrants or any other category of aircraft operator. Slots that are allocated on an ad hoc basis are not eligible for historic precedence. Slots that were requested as series, but allocated on ad hoc basis, and forms a series by the end of the season, have a possibility to be eligible for historic precedence.

The series of slots held on the Historic Baseline Date (HBD), on 31 January for summer season and 31 August for winter season, is used as the basis for determining qualification for historic precedence. Slots returned after this date, are considered as unused.

Slots that are not allocated via historic precedence, including new slots, will be put in a Slot Pool. In this Slot Pool, 50% of the slots must be allocated to new entrants (unless there are less than 50% requests by new entrants). New entrants are airlines that have less than 5 slots at that airport on the day of requesting slots.

2.4 Slot allocation priorities

Slots are allocated according to a priority scheme:

1. First slots are allocated according to historical precedence
2. A change to a historic slot will have priority over new requests for the same slot within the available capacity. These historic precedence ensures that airlines can operate services in the long term
3. When historic slots and changes to historic slots are allocated, a Slot Pool will be established with the remaining slots. These are allocated in the following priorities:
    - Introduction of year-round operation: new operations that extend an existing operation into a year-round operation (for example, an airline has a flight service during winter season and wants to continue with this flight in the summer season) have higher priority over other new operations. This is due to the interest of schedule stability and therefore have a higher priority [10].
    - Also a schedule that will be effective for a longer period of operation in the same season will have a higher priority. Regular scheduled operations will have priority over ad-hoc operations.

Additional criteria for slot allocation are (not in priority order) [7, 11]:

- Effective period of operation
- Type of service and market
- Competition
- Worldwide scheduling constraints e.g. curfews
- Requirements of the travelling public
- Frequency of operations
- Local guidelines

If the requested slot is not available, the coordinator will allocate another time slot, but the adjustments should be mutually agreed between the coordinator and the airline. Airlines can ask to keep their requested slots on a waiting list instead of refusing the schedule adjustment.

An airline may hold allocated slots if they intend to operate, transfer, exchange or use it in shared operation. In case the airline will not use the allocated slots, the airline must immediately return them to ensure that scarce capacity is not wasted. These returned slots can be reallocated to other airlines. Series of slots that airlines will not use, should be returned before the Slot Return Deadline (SRD) (15 January for summer and 15 August for winter season). According to IATA guidelines, airlines that intentionally return series of slots after SRD will receive a lower priority by the coordinator during the Initial Coordination of the next equivalent season.

When airlines use their slots, they may not intentionally operate services at a significant different time or use slots in a significantly different way than allocated by the coordinator. Slot times (in UTC) are based on the planned arrival (on-block) and departure (off-block) times. Actual times of arrival and departure may very due to operational factors. Slots are not route, aircraft or flight number specific and may be changed by an airline from one route type of service to another.

2.5 The IATA Slot Conference

The main objective of the Slot Conference (SC) is to agree on the slot allocations for the coming season between airlines and airport coordinators around the world.

The slot allocation itself is done in the following way. Airlines can request slots, before a date (33 days before SC), airlines must submit their planned operations to the coordinators of the airport. In the following two weeks, the appointments calendar are opened to coordinators to make appointments with airlines (14 days before SC). 12 days before the SC, the coordinators have to distribute the results of the Initial Coordination to all airlines (Slot Initial Allocation List Deadline (SAL)). After this deadline, airlines have the opportunity to make appointments with the coordinator (8 days before SC). Then the IATA Slot Conference takes place. This is a forum in Montreal, Canada where all airport coordinators and airline representatives are available. The forum is intended to allocate and managing slots at Level 3 airports and discussing schedule adjustments at Level 2 airports. The forum enables discussions between coordinators and airlines with the aim of agreeing on arrival and departure slots and officially allocate them. The advantage to have a forum is that all coordinators and airlines are in the same room and switching between airports and airlines can done quickly. Airlines can also discuss to other airlines to swap or transfer slots. After the Slot Conference, airlines can start making their flight schedules with the allocated slots.

When winter or summer season has arrived, the airline can use their obtained slots to fly at an airport and this process starts over again for the next winter/summer season.

Chapter 3

Literature related to the slot allocation problem

In this chapter two articles related for the slot allocation problem for Schiphol Airport will be discussed.

3.1 Economic efficiency and fairness of slot allocation

The article "Airport slot allocation in Europe: economic efficiency and fairness" by Castelli, Pellegrine and Pesenti [12] is one of the very few articles about the airport slot optimisation problem that contained a mathematical model. In the article the authors considered that there is an interdependence of the slots between different airports in Europe. The grandfather rights (as discussed in section 2.3) leads to a loss in economic efficiencies, because inefficient use of these grandfather slots will not lead to loosing them (80/20 rule).

A quantitative analysis has been done on the economic impact of the grandfather rights by comparing airlines' cost, when these rights are either enforced or not. The cost due to the non-optimality of the final schedule for each airline will be considered.

For each flight, an airline needs a feasible combination of slots at the origin and destination airport. A mathematical model is provided for slot allocation that is coherent with the preferred schedule from the very beginning. In the model an airline will obtain for each flight a departure (or arrival) slot if at the destination (or origin) airport it receives a slot which is compatible with the flight time of one of the possible routes connecting the two airports.

Since airlines prefer an ideal slot pair of arrival and departure slots for each of its flights. Given that this ideal slot pair is not available, the airline has to change the arrival time (and as a result the departure time), which leads to an undesirable cost to the airline, which is called 'shift cost'.

The model tries to minimize these shift costs. Each airline is allowed to indicate a maximum shift that it can bear for each flight. A deviation from the optimal arrival and/or departure time, will cause a revenue loss to the airline.

The efficiency in slot allocation could be increased by removing the restrictions enforcing the existence of the grandfather rights. Two cases were tested; one case where grandfather rights are included, and one with free allocation (without grandfather rights).

The model also contained a compensation mechanism: grandfather rights and free allocation may penalize just a few airlines to the benefit of the whole system. Additional constraints were added to increase the fairness of the final solution through a monetary compensation mechanism. Airlines that need to shift their schedule will receive a compensation that is equal to the maximum cost the airline is ready to bear. Multiple methods are proposed for the compensation mechanism and can be found in the article [12].

The model proposed by Castelli and Pelligrini has similarities to the problem at Schiphol Airport to solve the slot pair scarcity, but the main difference is that the model proposed optimises for multiple airports and airlines. It requires an agreement for all airlines on the compensation method, which is impossible. For example KLM has a very tight schedule, and is probably not going to pay other airlines because they have the 'ideal slots' obtained via grandfather rights. Applying this model for Schiphol Group is difficult, since airlines do not want to publish their schedules and revenues per flight.

3.2 What is the value of a slot for an airline?

Cao and Kanafani developed a model to asses the value of a slot in the article "The value of runway time slots for airlines" [13]. They analyse the relationship between rescheduling of flights and the profit of that flight. To assess the impact of flight rescheduling a minimum-cost flow model is constructed. When this model is solved, an optimal schedule can be obtained under the condition of rescheduling specific flights at specific time slots. Based on this optimal schedule the value of specific time slots can be determined. The model can be used for congestion pricing, slot auctioning of adjusting airline scheduling schedules to accommodate the airport capacity constraints.

3.2.1 Impact of flight reassignment

During the optimisation of an airline's schedule, they allocate available aircraft to flights, on the basis of their network structure and frequency plan, time dependent demand and the available fleet, with the objective to maximize profit or revenue and minimize cost. Due to airport capacities, the most optimal schedule will not always be possible. Therefore flights have to be shifted or cancelled to other brackets or assigned to other flight destinations. This will lead to different slot valuation for flights or airlines. In figure 3.1 a schedule for two connecting flights is shown. On the vertical axis, the time is indicated and the demand for flight 1 and flight 2 are shown on the left and right respectively. When the departure from airport 1 is shifted, the arrival (and departure) at the second airport has to be adjusted as well. In the case the flight is shifted to an earlier bracket, the demand at airport 1 will decrease. But the aircraft will arrive earlier at airport 2 and this provides a possibility to depart at airport 2 earlier in case there is a higher demand for it. When the departure at airport 1 is shifted to a later bracket, the demand will be higher at airport 1, while flight 2 will get a lower demand due to the delay in departure at airport 2. How to determine the optimal schedule (with a mathematical model) is left as extra reading material in [13].

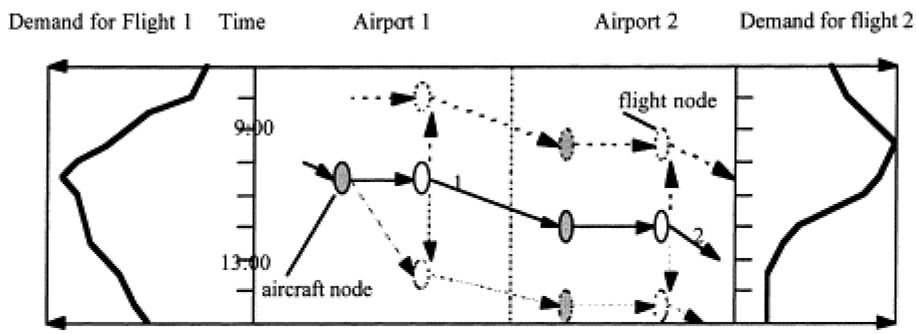

Figure 3.1: Impact of flight reassignment [13]

The value of the requested slot is determined as the profit impact to the airline if this slot for flight i is allocated to the airline. The value of the requested slot can be considered as the profit difference of the requested schedule and the optimal schedule of cancelling a flight i on that slot. The optimal alternative schedule should be determined when flight i is cancelled. When flight i is cancelled, the aircraft will become surplus and can be assigned to other flights (destinations) or ferried to other airports. The optimal schedule without flight i but with a surplus aircraft is given by the new total profit R_c and the value of the requested slot is $V_e = R_0 - R_c$ where R_0 was the expected profit under the requested schedule with flight i. Because R_0 was optimal (without bracket capacity constraints) in de requested schedule, V_e will not be negative.

The value of another slot can be determined as follows. Consider flight f is allocated to slot k and the new total profit of the new schedule is R_k, then the value of slot k for flight f will be $V_k = R_k - R_c$ where R_c is the maximum profit under cancelling flight f.

An overview of rescheduling and the value of slots is given in figure 3.2. R_0 original profit (without bracket capacity constraints) and suppose flight f has allocated slot j in the original schedule.

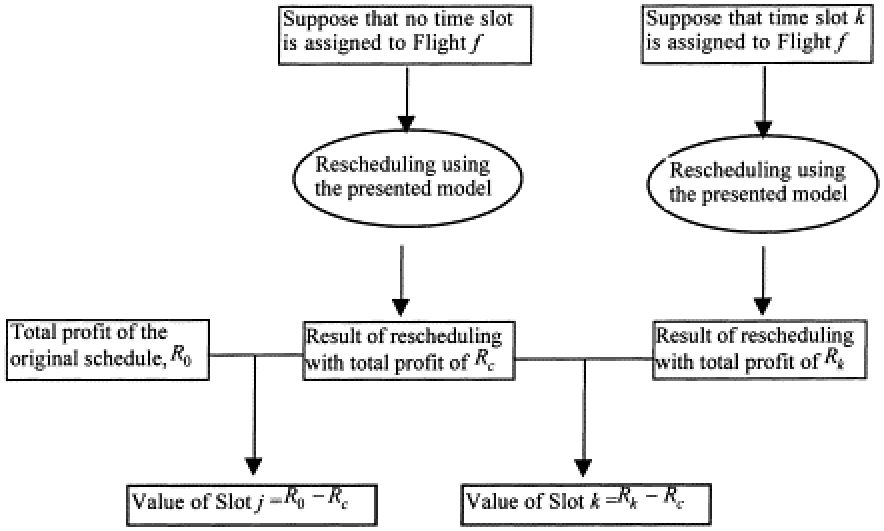

Figure 3.2: Valuation of slots [13]

As can be seen in figure 3.1 a shift in arrival or departure, will cause a series of effects on the airline's schedule. For the valuation of slots, Schiphol Group needs to know exactly what the flight rotations are of an aircraft for an airline. This is usually confidential information of an airline and for the slot allocation model at Schiphol no good input data is available.

Chapter 4

Slot usage of airlines

In this chapter the schedules (and so slot usage) of airlines will be investigated to determine the cost for the model (shift and omitting) for the model.

The original idea was to do an analysis on a year-round basis, but during the process it turned out it is better to do an analysis on a season-round basis, since the model will mainly be used per season and the winter and summer schedules can be analysed independent of each other.

4.1 Data files

To determine how airlines use slots for their flights, data analysis is done with the flight schedule of summer season 2015. Airlines use Slot Information Request messages (SIR) to inform about their flight schedules. An example of some SIR messages is given below. In a SIR message [14], the first characters indicates the action code (in this case an arrival or departure) with a flight number, the date range the flight will operate (can also be a single day, e.g. `25AUG25AUG`), the day(s) of operation (1 = Monday, 7 = Sunday), the number of seats with aircraft type (IATA 3 letter aircraft code), the destination (IATA airport code) with scheduled time (in UTC) and the type of flight (J = scheduled passenger flight).

```
H KL1497 29MAR26JUL 1234507 13273W 1445HUYHUY J
H KL1497 27JUL04SEP 1234500 13273W 1445HUYHUY J
H KL1497 06SEP23OCT 1234507 13273W 1445HUYHUY J
HKL1498 29MAR31JUL 1234507 13273W HUYHUY1725 J
HKL1498 03AUG04SEP 1234500 13273W HUYHUY1725 J
HKL1498 06SEP23OCT 1234507 13273W HUYHUY1725 J
```

In the SIR messages above there are two flights in summer season 2015, departure flight KL1497 (14:45 UTC) and arrival flight KL1498 (17:25 UTC) with destination airport HUY (Humberside

Airport). This flight is operating 6 days per week (not on Saturdays), and 5 days per week between 27 July and 4 September (not on Saturdays and Sundays).

All SIR messages (from now on called SIR file) will be converted to one movement per line and in local time (LT) as shown in table 4.1, because it is easier to do an analysis per movement.

Table 4.1: SIR files converted to movement per line

A/D	Scheduled datetime	Flight number	Destination	PAX	Aircraft	Flight type
DEP	29-03-2015 16:45 LT	KL1497	HUY	132	73W	J
ARR	29-03-2015 19:25 LT	KL1498	HUY	132	73W	J
..	..	..	..	..	..	..
DEP	23-10-2015 16:45 LT	KL1497	HUY	132	73W	J
ARR	23-10-2015 19:25 LT	KL1498	HUY	132	73W	J

A second file is used to indicate in what market segment a flight is [15]; hub, Europe, intercontinental, leisure, low cost or cargo. The hub segment are all flights that are served by SkyTeam airlines and KLM codeshare partners. A European flight is a non-hub carrier (not a SkyTeam airline or KLM codeshare partner) to a European destination with over 10,000 business passengers (outbound) per year. The intercontinental segment are non-hub carriers to an intercontinental destination with over 10,000 business passengers (outbound) per year. Leisure flights are non-hub carriers on European or intercontinental destinations with less than 10,000 business passengers (outbound) per year. Cargo flights are non-hub full freighter traffic. For low-cost there is no good definition available at Schiphol.

The flight numbers of the two files will be linked as shown in table 4.2. The market segments are used to average the values of all individual flight numbers. With the result of these segments, a better preference can be determined to new slot pair requests for flights. When no match can be found between flight number and market segment, the flight number will not be considered.

From now on, when an airline or flight preference is mentioned, the market segment of the flight is meant.

Table 4.2: Market segment for KL1497 and KL1498

Flight Number	A/D	Carrier	Market segment
KL1497	DEP	KL	hub
KL1498	ARR	KL	hub

4.2 Data analysis

Each flight that arrives (landing) or departs (take-off) has a flight number, will have a slot (scheduled time) with destination (from or to) and will be executed by an airline (carrier) with a specific aircraft type. These variables will be used to determine the 'preference' of a market segment.

In the year 2014-2015 (winter season 2014 and summer season 2015) 458,106 movements took place [16]. Since there are many flights that are only executed a limited time, a minimum is set before a flight number is taken into account. Flight numbers that occur less than 24 times (80% of the number of weeks in summer season 2015) will not be used for the analysis of preferences.

4.2.1 Preprocessing

In table 4.1 an example flight schedule is given. Not all flights are operated all day-of-week, and therefore the analysis is done per day-of-week (DOW: Monday is day 1, Sunday is day 7). A flight number is normally executed once a day (one arrival or one departure), except flights that, for example use their sixth freedom of the air (flights from country A to country B (The Netherlands) to country C, with origin or termination in own state [17]) or cargo flights can have the same flight number for the arrival and departure flight (for example flight GA88 of Garuda Airlines from CGK (Jakarta) to AMS (Amsterdam) to LGW (London Gatwick)).

At Amsterdam Airport Schiphol time brackets of 20 minutes are used during day time (between 06:00 LT and 22:59 LT), and during night (between 23:00 LT and 05:59 LT) brackets of 1 hour, as can be seen in the bracket list in table A.1. Due to the 'fire break' (a bracket to change between an arrival peak and departure peak or vice versa), there is also a bracket of 10 minutes between 07:50 LT and 07:59 LT. Since the flight schedule contains scheduled times up to 5 minute precision, these times are converted to a bracket time (for night brackets HH:00-HH:59 and day time brackets HH:00-HH:19, HH:20-HH:39 or HH:40-HH:59) since the exact scheduled time is not that important, only in which bracket the flight is scheduled. For this analysis, the 30 and 10 minute bracket (07:20-07:49 LT and 07:50-07:59 LT) will be considered as two 20 minute brackets (07:20-07:39 LT and 07:40-07:59 LT respectively). For example, for the flights in table 4.1, KL1497 will depart at 16:40 LT and KL1498 will arrive at 19:20 LT.

4.2.2 Changes in flight schedules

In the flight schedules, changes in scheduled times can be detected. Per day-of-week, each flight number (and per movement: arrival or departure due to the reason mentioned above) is compared to the previous flight on the same day-of-week. A change is when there is a change in bracket time (so a difference in time in the same bracket is not considered as a change). In this thesis a difference of one bracket, is called one shift (earlier or later). So when the bracket time difference is one hour (during day time), the difference is three shifts. Per flight number, the number of shifts is counted. Note that the number of minimum shifts is chosen for a change. For example, if a flight operated last week at 20:00 LT and this week at 08:00 LT, there is a difference of exactly 12 hours. Assume a flight is moved to an earlier bracket, then the number of bracket shifts is 36 (12*3, brackets of 20 minutes) and when the flight is moved to a later bracket (overnight), the number of bracket shifts is 22. This is because during night there are brackets of one hour. So 22 shifts will be counted in this example.

Further a distinction is made between 'temporary' and 'permanent' changes. A temporary change is a bracket change that will return to the original bracket within four weeks. The bracket time of a flight before a change on the same day-of-week is considered as original bracket. Changes longer than four weeks are considered as permanent change. In figure 4.1 an example is shown what is considered as a temporary or permanent change. In week 3 the flight is moved to a different bracket (20 minutes later) and in week 4 returning to the original bracket and is considered as a temporary change. In week 6 the scheduled time is moved to an earlier bracket, and will return to the original bracket in week 11. However, this change lasts longer than 4 weeks, and therefore it is considered as a permanent change and not as a temporary change.

Note that for a temporary change two changes are detected (to a later (or earlier) bracket and then returning to the earlier (or later) original bracket), in figure 4.1 in week 3 and 4). However this will be counted as one change, while the same case longer than 4 weeks will result in two changes (in figure 4.1 in week 6 and 11). So figure 4.1 contains in total 1 temporary change and 3 permanent changes.

When the analysis is done for a year instead of one season, there will be a shift for intercontinental flights in the first week of the second season due to the winter or summer time changes on the northern hemisphere. For example from winter to summer season: a flight will depart at the same time in Peru (where no winter/summer time is), but due to summer time in Europe it will arrive an hour earlier at Schiphol Airport and vice versa.

4.2.3 Omissions in flight schedules

Omissions (not flying) are detected in a similar method as explained above for changes. Per flight number and day-of-week is compared to the previous flight on the same day-of-week. This method only counts the omissions between the first and last flight date per day-of-week (when for example a flight is only operated for half of the season, the other half of the season is not considered as omissions). If the difference between the previous flight and the current flight is seven days, it means that the flight was operated in the previous week and a difference of 14 days means that a flight was not operated the previous week, etcetera.

Per flight number, again the number of omissions are determined. A short period omission is considered when one week is not flown (per day-of-week), and a long period omission when a flight is not operated for two or more weeks (per day-of-week).

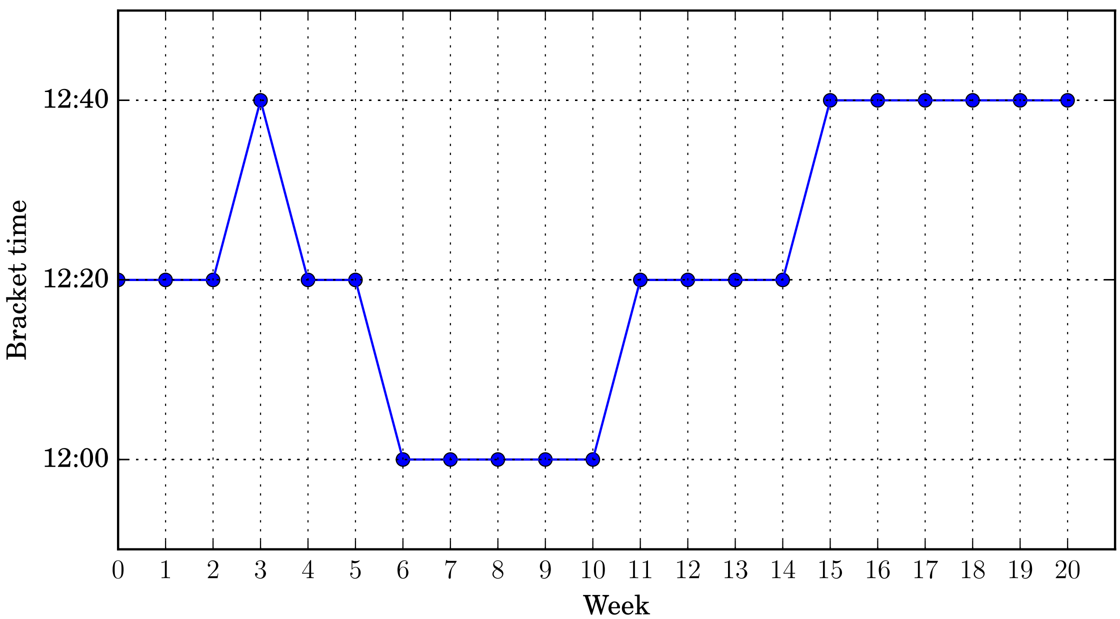

Figure 4.1: Example of changes, a blue point indicates an allocated slot

Flight KL1497 in section 4.1 is not operated on day-of-week 7 (Sunday) between between 27 July and 4 September (5 weeks), which will be counted as five omissions and is considered as one long period omission.

4.3 Results & Analysis of results

In the SIR file of summer 2015 there are 286,118 movements, whereof 267,244 movements are used (due to minimum of 24 flights). In table 4.3 and 4.4 the results for four flights are shown. In table 4.5 the aggregated values for the segments are shown and only a small number, 5000 flights (2%), could not be allocated to a market segment. In this table multiple segments* (Euro, intercontinental, hub, leisure, low-cost) are aggregated for confidentiality reasons. An extra row example** is given for further explanation in Table 4.6.

The results of the non-aggregated values are as expected. When a flight more in a segment towards the hub functionality of Schiphol Airport, the percentage of omissions (with respect to total flights) decreases. In the segment leisure, it is allowed to shift a lot before a flight will be omitted (they fly when there is demand, for example charter flights), while for the segment Europe, hub and intercontinental the schedules are much 'tighter' and therefore less shifts are allowed before a flight will be omitted. The freight segment has the highest percentage of shifts (153% w.r.t. total flights) and omissions (28% w.r.t. flights) and it can be concluded that they are willing to fly when they can fly (which is indeed the case, because they only fly when there is demand for air freight).

Table 4.3: Results of analysis (part 1)

Flight Number	A/D	Flights	Changes			
			Temporary	Permanent	Total	Shifts
KL1497	DEP	175	0	0	0	0
KL1498	ARR	175	0	0	0	0
ZZ1234	DEP	27	1	0	1	16
ZZ1235	ARR	27	0	0	0	0

Table 4.4: Results of analysis (part 2)

Flight Number	A/D	Omissions				Market Segment
		Short	Long	Total periods	Count	
KL1497	DEP	0	1	1	5	hub
KL1498	ARR	0	1	1	5	hub
ZZ1234	DEP	2	0	2	2	freight
ZZ1235	ARR	3	0	3	3	freight

Table 4.5: Results of analysis per market segment

Segment	Flights	Changes				Omissions			
		Temp.	Perm.	Total	Shifts	Short	Long	Total	Count
Undefined	5052	34	135	169	2665	153	155	308	1044
Aggregated*	254,151	1207	1774	2981	34,223	2044	1023	3067	8782
Freight	8041	327	597	924	12,304	544	294	838	2274
Example**	200	15	5	20	40	12	8	20	75

4.4 Costs for a shift or an omission of a flight

For the model presented in chapter 6, the number of shifts or omissions should be minimised. In the linear program, different flights (per market segment) will have different cost for a shift or an omission and the model will minimise these costs. A perfect flight schedule, without any shifts or omissions, will have a cost value of infinity (set to a large number) for a deviation, so making a shift or an omission definitely not allowed. Because flights are aggregated per market segment, this will not occur (because bracket shifts or omissions will happen in any case).

The total number of flights per segment, the total number of shifts and the total omission periods (short and long) are calculated. The sum of shifts and omission periods is called total deviations. The number (count) of omissions is not used, because this number can be higher than the number of shifts (for example due to flight KL1497 and KL1498 in table 4.3 and 4.4, which do not operate on Sunday for five weeks (so count of omissions is five) while no bracket change has occurred) and in the model this will result that an omission would be better than a shift. The model will than not allocate an alternative slot, but will omit the flight immediately. Dividing the total deviation by the number of flights, gives a ratio how constant a segment is. If this ratio is low, the schedule is 'tight'.

[Removed for online publication.]

Applying these calculations to the results in section 4.3, the cost for shift and omission in table 4.6 are obtained.

Table 4.6: Cost for shifts and omissions

[Removed for online publication.]

Chapter 5

Linear programming and exact solving techniques

In this chapter an introduction will be given to linear programming and how to solve these programs, since the model that will be formulated for slot allocation in chapter 6 will be a linear program.

5.1 Introduction to linear programming

An optimisation problem can be formulated by a linear programming (LP) problem. An LP tries to find a set of values for continuous variables (called decision variables: for example $x_1, x_2, ..., x_n$) that minimizes or maximizes an objective function, while satisfying a set of linear constraints (system of linear equations and/or inequalities). A basic LP can be expressed as in equation 5.1, where the goal is to optimize a function with a constraint that restricts the upper bound.

$$\begin{array}{rll} \text{Maximize:} & \sum_j \quad c_j \cdot x_j & \\ \text{Subject to:} & \sum_j \quad a_{ij} \cdot x_j & \leq b_i \end{array} \tag{5.1}$$

An integer (linear) program (IP) is an LP in which all decision variables are integers, in the case of the LP in equation 5.1, the variables x_j would be integers. An LP with a combination of decision variables that are integers and continuous is called a mixed integer programming problem (MIP).

From an IP an LP can be obtained by relaxing the integer requirements in the IP. The resulting LP is called the LP-relaxation of an IP. An IP problem in which the decision variables are restricted to zero or one is called a binary integer program (binary IP or BIP) [18].

5.2 Exact solving techniques

In this section two popular exact solving techniques for solving LP problems will be discussed, the simplex method and the brand-and-bound algorithm.

5.2.1 Simplex method

A classic algorithm for solving LPs is the simplex method by George Dantzig. The simplex method is an iterative process that uses a sequence of LP relaxations. The simplex method for upper bounded variables is used for reducing the problem size by implicitly handling the upper and lower bounds on variables. This explanation will use the primal form of an LP. Using the simplex method for the dual form is effective for re-optimising the current optimum when additional constraints are added, without resolving the augmented LP problem from scratch [19].

[Removed for online publication.]

5.2.2 Branch and Bound

Branch and Bound (B&B) is a 'divide-and-conquer' framework for solving discrete optimisation problems. B&B is a heuristic method for global optimization for non convex problems, that maintain a lower and upper bound on the global objective value. The performance of B&B is often slow and in worse case it will be exponential [20].

[Removed for online publication.]

Chapter 6

Linear programming model for slot allocation

When an airline requests a slot pair for a flight and given the current slot usage at Schiphol Airport this is not possible, what will be the best alternative slot pair for this flight? In this chapter an integer programming model will be presented to determine the best alternative for a flight.

6.1 Relevant models

The model that will be presented in this chapter for slot allocation has similarities to a rostering problems, for example an examination scheduling/timetabling problems (ESP) or nurse scheduling problems (NSP). In both problems there is an allocation of exams or nurses to time slots over a period subjected to constraints.

In an examination scheduling problem, exams should be allocated into a number of periods within a defined examination session while satisfying constraints. McColum, McMullan et al. [21] defines an ESP with the following points which could be translated to the slot allocation model:

- Hard constraints must be satisfied
- Soft constraints which contribute to a cost if they are violated
- A set of exams should be scheduled into periods
- A set of rooms with individual capacities are provided
- Weighting values for soft constraints
- The capacity of individual rooms is not exceeded at any time throughout the examination session
- Satisfaction of period related constraints, Exam_A after Exam_B

- Exams are allocated to at most one room (cannot be split) and allocated to at most one period.

The authors formulated a mathematical model with a weighted-sum approach and not a multi-objective for simplicity. The authors mentioned that a weighted-sum approach is not ideal because it may have adverse side effects for certain individual students.

The problem can be translated to the slot allocation problem as follows: a set of flights should be allocated to brackets, while the bracket capacities may not exceed. An arrival should take place before a departure, flights cannot be split over multiple brackets and cannot be allocated to more than once a day. These points are the hard constraints and should be satisfied. The soft constraint is that each flight has a preference and not satisfying these preferences will have a weighting value for violation.

The nurse scheduling problem is similar to the examination scheduling, but for the NSP there are legal requirements, for example the number of nurses (depending on qualifications), maximum working times, etc.

Mathematical formulations in ESP and NSP are used as guidance to develop the model for slot allocation.

6.2 ILP model for slot allocation at Schiphol Airport

6.2.1 Notation and convention

First some notation and convention will be introduced to simplify the problem statement:

Sets

- I: set of flights ($i = 1, ..., I$)
- W: set of weeks ($w = 1, ..., W$) (week number of the year, starting on Sunday*)
- D: set of days ($d = 0, ..., 6$) (day of week, starting on Sunday = 0, Saturday = 6*)
- S: set of brackets per day ($s = 1, ..., 58$) (1 = 00:00h-00:59h LT, etc. (bracket table, appendix A))
- M: set of movement types $m = \{ARR, DEP\}$ (arrival (ARR) or departure (DEP))

* Since seasons always start on the last Sunday of March or October, it is easier to model with week numbering and day of week starting on Sunday. For readability, in the input and output the system Monday = 1 and Sunday = 7 is used.

Parameters

Airport parameters:

- C_{sm}: capacity of bracket s for movement m (bracket table, appendix A)
- $T_{N_{ARR}}$: total arrival capacity during night for season (capacity declaration, appendix B)
- $T_{N_{DEP}}$: total departure capacity during night for season (capacity declaration, appendix B)
- T_N: total arrival and departure capacity during night for season (capacity declaration, appendix B)
- T_T: total capacity for season (capacity declaration, appendix B)

Flight dependent parameters:

- D^i: set of days that flight i is operating
- Q^i_m: requested bracket for movement m for flight i
- R^i_m: set of excluded brackets for movement m for flight i
- TAT^i_{min}: minimum turnaround time for flight i
- TAT^i_{max}: maximum turnaround time for flight i
- P^i_{shift}: penalty cost for a shift (one bracket earlier or later) for flight i
- P^i_{omit}: penalty cost for omitting a flight i

This model only focuses on the slot availability and its optimisation. Airport capacities (such as gate availability, baggage handling capacity, aircraft size, etc.) are not taken into account. Further is assumed that an aircraft will depart on the same day as arrival, so it is not possible to depart the next day (so called stopover flights). Stopovers are admitted to many regulations, which are not possible to implement in the model.

Decision variables

The following variables should be determined by the ILP:

- $X^i_{wdsm} = \begin{cases} 1 & \text{if flight } i \text{ will be allocated in week } w\text{, day } d\text{, bracket } s \text{ and is a movement } m \\ 0 & \text{otherwise} \end{cases}$
- e^i_{wdm}: number of brackets earlier than the requested bracket Q^i_m for flight i in week w and day d for movement m
- l^i_{wdm}: number of brackets later than the requested bracket Q^i_m for flight i in week w and day d for movement m

  With e^i_{wdm} and $l^i_{wdm} \in \mathbb{Z}^+$.

6.2.2 Current bracket usage

Using the converted SIR file as given in table 4.1, which was used to determine an airline's schedule, the remaining capacity per bracket per day and per movement is determined. The flights in the SIR file will not be loaded in the model, because this will result in over 15 billion decision variables for summer 2015 (for each week, each day, each bracket and each movement only one of these decision variables will equal to one).

Since in the SIR file no slot pairs can be identified, all flights will be treated as individual flights. All flights n in the SIR file should be allocated to exactly one bracket as given in 6.1:

$$\sum_{w \in W} \sum_{d \in D} \sum_{s \in S} \sum_{m \in M} X^i_{wdsm} = 1 \qquad \forall i \in \{1, ..., n\} \tag{6.1}$$

Next, the information of a flight (arrival or departure, date with time) will be converted to the model's parameters. For example, the conversion of the flights in table 4.1 is shown in table 6.1, whereof the last column will be used for further calculations:

Table 6.1: Conversion schedule to model variables

A/D	Scheduled datetime	i	w	d	s	m	X^i_{wdsm}
DEP	29-03-2015 16:45	1	13	0	39	DEP	$X^1_{13,0,39,DEP} = 1$
ARR	29-03-2015 19:25	2	13	0	47	ARR	$X^2_{13,0,47,ARR} = 1$
..	..	..	..	..	..	..	..
DEP	23-10-2015 16:45	$n-1$	43	6	39	DEP	$X^{n-1}_{46,6,39,DEP} = 1$
ARR	23-10-2015 19:25	n	43	6	47	ARR	$X^n_{43,6,47,ARR} = 1$

With the variable in the last column, the capacity per bracket per week, per day, per movement can be determined quickly. Introduce the new variable C_{wdsm} as given in equation 6.2:

$$C_{wdsm} = C_{sm} - \sum_{i \in I} X^i_{wdsm} \qquad \forall w \in W, \forall d \in D, \forall s \in S, \forall m \in M \tag{6.2}$$

C_{wdsm} is the capacity for week w, day d, bracket s and movement m. All flights of the SIR file are incorporated in these bracket capacities.

6.2.3 Instance for new flights

For new flights, for which a optimal slot pair should be found, the instance is given by a 10-tuple:

i	D^i	Q^i_{ARR}	Q^i_{DEP}	R^i_{ARR}	R^i_{DEP}	TAT^i_{min}	TAT^i_{max}	P^i_{shift}	P^i_{omit}
1	2	36	40	34	42,43	2	5	1.5	5.0
2	5,7	15	18	12,13	19,20	1	4	1.2	7.5

The instances also contains additional information to create a SIR file, but this is irrelevant for the model and not shown here. This example instance contains two flight pairs:

- Flight 1 wants to operate a flight on day 2 (Tuesday) with an arrival in bracket 36 (15:40-15:59 LT) and a departure in bracket 40 (17:00-17:19 LT). For arrival, bracket 34 may not be used and for departure brackets 42 and 43 may not be used. The minimum turnaround time is 2 brackets (40 minutes) and the maximum turnaround time is 5 brackets (119 minutes). Shifting a bracket from the preferred bracket will have a cost of 1.5 per bracket and omitting the flight will have a cost of 5 per movement. So instead of shifting more than 8 brackets (earlier or later, which gives a cost of 11.5), it is more beneficial to omit the flight (cost of 10).
- Flight 2 wants to operate a flight on day 5 and 7 (Friday and Sunday) with an arrival in bracket 15 (08:40-08:59 LT) and a departure in bracket 18 (09:40-09:59 LT). For arrival, brackets 12 and 13 may not be used and for departure brackets 19 and 20 may not be used. The minimum turnaround time is 1 bracket (20 minutes) and the maximum turnaround time is 4 brackets (99 minutes). Shifting a bracket from the preferred bracket will cost 1.2 and omitting a movement will cost 7.5. So instead of shifting more than 12 brackets (earlier or later, which gives a cost of 15.6), it is more beneficial to omit the flight (cost of 15).

Note that when a movement is omitted (e.g. arrival), the other movement (e.g. departure) on that day will also be omitted, otherwise no feasible pair can be created. This is the reason that for a flight twice the number of shifts is allowed before the flight will be omitted. Further in the example above, during omitting the turnaround times are not taken into account.

6.2.4 Objective function

The objective is to meet as many soft constraints as possible, so minimize the total cost of shifting a flight from the requested slot pair and the cost of omitting a flight (when no slot is available that satisfies the conditions, or the shifting cost are bigger than the omission cost).

$$\text{Minimize} \sum_{i \in I} \sum_{w \in W} \sum_{d \in D^i} \sum_{m \in M} \left(P^i_{omit} \cdot (1 - \sum_{s \in S} X^i_{wdsm}) + P^i_{shift} \cdot (e^i_{wdm} + 1.05 \cdot l^i_{wdm}) \right) \tag{6.3}$$

In equation 6.3, the shifting cost is a linear function (for example, one bracket shift will have a cost of 5, two bracket shifts will have a cost of 10, etc.). l^i_{wdm} is multiplied by 1.05 to avoid symmetry (first use the earlier bracket and then the later bracket).

6.2.5 Constraints

Schiphol Airport and Luchtverkeersleiding Nederland (LVNL) determines per season the operational capacity and with this, a capacity is declared. The declared capacity gives the number of maximum flights per season (winter or summer) and night period (arrivals, departure or both) and is given in appendix B.

Inequality 6.4 states that the capacity of a bracket per week per day and per movement as calculated in equation 6.2 should not be exceeded. In inequality 6.5 the total (seasonal) capacity is given (day and night). Inequalities 6.6 and 6.7 states respectively the total arrival and total departure capacity that are allowed during night. And finally in inequality 6.8 the total (seasonal) night capacity is given.

As can be seen in the declared capacity in appendix B, the night capacity for arrival, departure or both are not declared for all seasons. In summer 2015 (S15) no total night capacity is declared, while for winter 2015 (W15) and summer 2016 (S16) only the capacity for departures and total are declared. If no capacity is declared, the value of the capacity will be set to a large number (bigger than the largest capacity).

$$\sum_{i \in I} X^i_{wdsm} \leq C_{wdsm} \qquad \forall w \in W, \forall d \in D, \forall s \in S, \forall m \in M \tag{6.4}$$

$$\sum_{w \in W} \sum_{d \in D} \sum_{s \in S} \sum_{m \in M} X^i_{wdsm} \leq T_T \qquad \forall i \in I \tag{6.5}$$

$$\sum_{w \in W} \sum_{d \in D} \sum_{s \in S^{ARR}} X^i_{wds,m=ARR} \leq T_{N_A} \qquad \forall i \in I, S^{ARR} = \{1, ..., 9, 10, 58\} \tag{6.6}$$

$$\sum_{w \in W} \sum_{d \in D} \sum_{s \in S^{DEP}} X^i_{wds,m=DEP} \leq T_{N_D} \qquad \forall i \in I, S^{DEP} = \{1, ..., 9, 57, 58\} \tag{6.7}$$

$$\sum_{w \in W} \sum_{d \in D} \sum_{s \in S^{ARR}} X^i_{wds,m=ARR} + \sum_{w \in W} \sum_{d \in D} \sum_{s \in S^{DEP}} X^i_{wds,m=DEP} \leq T_N$$
$$\forall i \in I, S^{ARR} = \{1, ..., 9, 10, 58\}, S^{DEP} = \{1, ..., 9, 57, 58\} \tag{6.8}$$

To avoid multiple arrivals and departures per day, the number of arrivals is restricted to one per day by inequality 6.9 (and so departure, since a pair is required):

$$\sum_{s \in S} X^i_{wdsm} \leq 1 \qquad \forall i \in I, \forall w \in W, \forall d \in D^i \tag{6.9}$$

As mentioned earlier, it is assumed that stopovers are now allowed. When an aircraft arrives, it should be followed by a departure on the same day. To create a slot pair on the same day constraint 6.10 applies:

$$\sum_{s\in S} X^i_{wds,m=A} - \sum_{s\in S} X^i_{wds,m=D} = 0 \qquad \forall i \in I, \forall w \in W, \forall d \in D^i \tag{6.10}$$

Between an arrival and a departure, a minimum time is required to disembark and board the passengers and unload and load luggage, which is called the turnaround time (TAT). An aircraft that is not flying, is not generating revenue for the airline, so there is also a maximum turnaround time (which airlines want to minimize, except there is no demand or the times are unfavourable). Inequality 6.11 and 6.12 states the number of brackets between arrival and departure should be larger than the minimum and smaller than the maximum turnaround time respectively. The left-hand side determines in which bracket the arrival and departure is (if $X^i_{wds,m=D} = 1$) and the right-hand side is the TAT, multiplied whether a flight is operated (if $X^i_{wds,m=D} = 1$).

Inequality 6.11 will also ensure that an arrival will take place before a departure, since the minimum turnaround time is at least one bracket.

$$\sum_{s\in S} s \cdot X^i_{wds,m=D} - \sum_{s\in S} s \cdot X^i_{wds,m=A} \geq \sum_{s\in S} X_{wds,m=D} \cdot TAT^i_{min} \qquad \forall i \in I, \forall w \in W, \forall d \in D^i \tag{6.11}$$

$$\sum_{s\in S} s \cdot X^i_{wds,m=D} - \sum_{s\in S} s \cdot X^i_{wds,m=A} \leq \sum_{s\in S} X_{wds,m=D} \cdot TAT^i_{max} \qquad \forall i \in I, \forall w \in W, \forall d \in D^i \tag{6.12}$$

Given an airline only wants to fly on certain days, then all other days can be excluded by equation 6.13:

$$X^i_{wdsm} = 0 \qquad \forall d \notin D^i \tag{6.13}$$

For a flight, there is the opportunity to exclude the allocation to some brackets. Therefore only brackets that are not blocked may be used and this is given by equation 6.14 and 6.15:

$$X^i_{wds,m=ARR} = 0 \qquad \forall s \in R^i_{ARR} \tag{6.14}$$

$$X^i_{wds,m=DEP} = 0 \qquad \forall s \in R^i_{DEP} \tag{6.15}$$

Assign a flight to the requested brackets if possible, else choose another bracket. Deviation from the requested bracket leads to a cost value that are determined in section 4.4.

$$\sum_{s\in S} s \cdot X^i_{wdsm} = Q^i_m \sum_{s\in S} X^i_{wdsm} + l^i_{wm} - e^i_{wm} \qquad \forall i \in I, \forall w \in W, \forall d \in D^i, \forall m \in M \tag{6.16}$$

6.2.6 Output

The model will output a list with the decision variables: X^i_{wdsm} to which bracket a flight is allocated, the number of brackets earlier e^i_{wdm} and the number of brackets later l^i_{wdsm} than the requested slot pair and the objective value.

All these variables are converted to a readable table like in table 4.1 and to a SIR file that can be imported by the systems of Schiphol Airport as shown in section 4.1.

Chapter 7

Results exact method

In this chapter the results of some example flights will be presented. The total output will be too long to present here (summer 2015 lasts 30 weeks and results in 210 days, so 420 movements per flight) and therefore only one day will be analysed (but the model will optimize for the whole season). At the end of the chapter a graphical view of a schedule flight will be shown.

7.1 Programming language and solver details

The model is written in programming language Python [22] using the library PuLP [23, 24]. The default solver of PuLP is CBC (COIN-OR Branch and Cut) developed by Computational Infrastructure for Operations Research (COIN-OR) [25]. CBC is an open-source mixed integer programming solver that uses primal and dual simplex algorithms and relaxations, branch-and-cut methods and cutting plane implementations (combinatorial cuts, flow cover cuts, Gomory cuts and lift-and-project cuts) [26].

The PuLP library can easily interface with popular commercial solvers, like IBM CPLEX [27], Gurobi [28] or FICO Xpress [29] by changing one parameter in the script. In this section the default solver CBC is used, unless the running time of the default solver exceeds 3600 seconds (1 hour). In that case the commercial solver Gurobi (with an academic license) will be used.

The calculations are executed on a computer with an Intel i5-3470 (3.20 GHz) processor and 24 GB RAM memory.

7.2 Example result

The results shown below are based on the day with the most flights in 2015, namely 20 July 2015. On that day there were 1460 movements (arrivals and departures). This does not mean that on that day the most passengers travelled via Schiphol Airport, the busiest day at Schiphol in passenger numbers was 2 August [30] and this day had 100 movements less than 20 July.

7.2.1 Bracket capacities

In table 7.1, the bracket capacities for 20 July that are left are shown. For arrivals one movement per bracket is overbooked, while for departures this is not the case. This change is made to visualise the changes that will be made by the model, otherwise most of the flights will be allocated to their requested bracket.

Table 7.1: Bracket capacities for 20 July 2015

	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
ARR	1	1	0	0	5	3	1	0	1	1	4	1	0	0	1	1	7	2	8
DEP	1	2	1	0	1	0	0	1	8	1	1	2	7	3	6	1	0	0	1

7.2.2 Example flights

The instance below contains six flights. These flights are set-up in such a way to show what the effect is when multiple flights request the same slot pairs (flights $i = 1, 3, 5$). An explanation of the instances for the model can be found in subsection 6.2.3.

i	D^i	Q^i_{ARR}	Q^i_{DEP}	R^i_{ARR}	R^i_{DEP}	TAT^i_{min}	TAT^i_{max}	P^i_{shift}	P^i_{omit}
1	1,2,3,4,5,6,7	12	17	*,13	*,19	3	8	165	377
2	1,2,3,4,5,6,7	24	30	*	*	2	6	6	82
3	1,2,3,4,5,6,7	13	17	*,15	*,18	1	7	139	642
4	1,2,3,4,5,6,7	13	29	*	*	12	20	40	296
5	1,2,3,4,5,6,7	12	17	*	*	1	15	37	415
6	1,2,3,4,5,6,7	13	24	*	*,25,26	8	12	25	30
7	1,2,3,4,5,6,7	15	21	*	*	3	15	95	512
8	1,2,3,4,5,6,7	17	28	*	*	5	15	153	568

* All brackets that are not listed in table 7.1: bracket numbers 1-11 and 31-58.

7.2.3 Results

In table 7.2 the results of the instances above with the bracket capacity on 20 July. The runs are done in the following way: first only the schedule for flight 1 is optimized, followed by flights 1 and 2, followed by flights 1, 2 and 3, etcetera. The slot pair per flight is written as tuple (ARR, DEP).

Table 7.2: Results of example instance for 20 July 2015

Requested slot pair:	(12,17)	(24,30)	(13,17)	(13,29)	(12,17)	(13,24)	Running time (s)
Flight $i = 1$	(12,16)						13
Flights $i = 1, 2$	(12,16)	(26,30)					44
Flights $i = 1, ..., 3$	(12,16)	(26,30)	(13,19)				56
Flights $i = 1, ..., 4$	(12,16)	(23,27)	(13,19)	(16,30)			295
Flights $i = 1, ..., 5$	(12,16)	(23,27)	(13,19)	(16,30)	(16,20)		1051
Flights $i = 1, ..., 6$	(12,16)	(23,27)	(13,19)	(16,30)	(16,20)	(- -)	405
Flights $i = 1, ..., 7$							948
Flights $i = 1, ..., 8$							1877

Results analysis

Every decision that is made by the model can be explained:

- If only flight 1 is added, the requested arrival bracket is available, but for departure, bracket 17 is not available. The closest available bracket is bracket 16 and will be assigned to flight 1
- When flight 2 is added, no conflict will occur, because a slot pair is requested at a different part of the day. Because bracket 24 is not available, bracket 26 is chosen. However, bracket 23 will lead to a lower shift cost (only one difference), but this will violate the maximum TAT (or the departure has to shift more, leading to higher costs).
- When flight 3 is added, flights 1 and 3 both want to depart in bracket 17, which is not available. For arrival, for both flights there is one slot available. The closest departure brackets to 17 that are available are 15 (1 earlier) or 19 (2 later). Since flight 1 has a higher shift cost, bracket 15 is allocated to flight 1 and bracket 19 for flight 2
- When flight 4 is added, flight 1 will keep the same slot pair (because the highest shift cost, other flights are shifted first). Flights 2 and 4 want to depart in bracket 30 and 29 respectively, however around that bracket only one slot is left (in bracket 30). Since flight 4 has a higher shifting cost, flight 4 will get a slot in bracket 30. The closest departure to 30 for flight 2, is bracket 27. However, the minimum TAT for flight 2 is 2 brackets, so an arrival slot before bracket 26 should be chosen, which is 23. Flights 1, 3 and 4 want to have an arrival in bracket 12 or 13, but only 2 slots are available. Because flight 4 has the lowest shifting cost of these three flights, flight 4 has to shift its arrival to bracket 16
- When flight 5 is added, it will compete with flights 1 and 3 (for arrival and departure) and flight 4 (for arrival). Because flight 5 has the lowest shifting cost, flight 5 will use slots that are available and shifting this flight results in the lowest cost
- Flight 6 has a low shift/omit ratio compared to the other flights. This means that it does not allow shifting a lot before omission (maximum 2 shifts in this case). For the departure, there are enough slots in that bracket, but for arrival the slots are scarce. If this flight is assigned to (13, 24), flight 3 has to move its arrival to bracket 16 what will cost 417

(3*139), while omitting flight 6 will cost 380 (2*195). Therefore flight 6 is omitted for that day

- Flight 7 and 8 are added for evaluating the running time and will not be considered here

In figure 7.1 the schedule for flight 5 per day-of-week is shown (flight 6 is excluded for this run). In this figure can be seen that for Friday, Saturday and Sunday the requested arrival slot is available for all weeks, while for the other days one or two days a different bracket is allocated. The departure for flight 5 is the most consistent on Sunday, while for Friday and Saturday bracket 17 is not available often (many shifts).

Running time

In the last column of table 7.2, the running times are shown. These values are calculated by averaging the running times of three runs. As can be seen, the running time increases when flights are added, but it is also dependent on the constraints (preferences of a flight). From the results, no correlation between number of flights and running time can be proved. When flight 5 was included, the running time was four times higher than the case with 4 flights. However, when flight 6 was added, the running time decreases again. Therefore flight 7 and 8 are added to show whether flight 5 or flight 6 is an outliner. As can be seen in table 7.2, the running times of flight 7 and 8 increases, and flight 5 is probably an outliner. When all eight flights are optimised, the model returned 1680 slot pairs (3360 movements). The default solver was able to optimise these eight flights in 1877 seconds, while the commercial solver Gurobi was able to solve the model in 302 seconds, a factor of 6 faster.

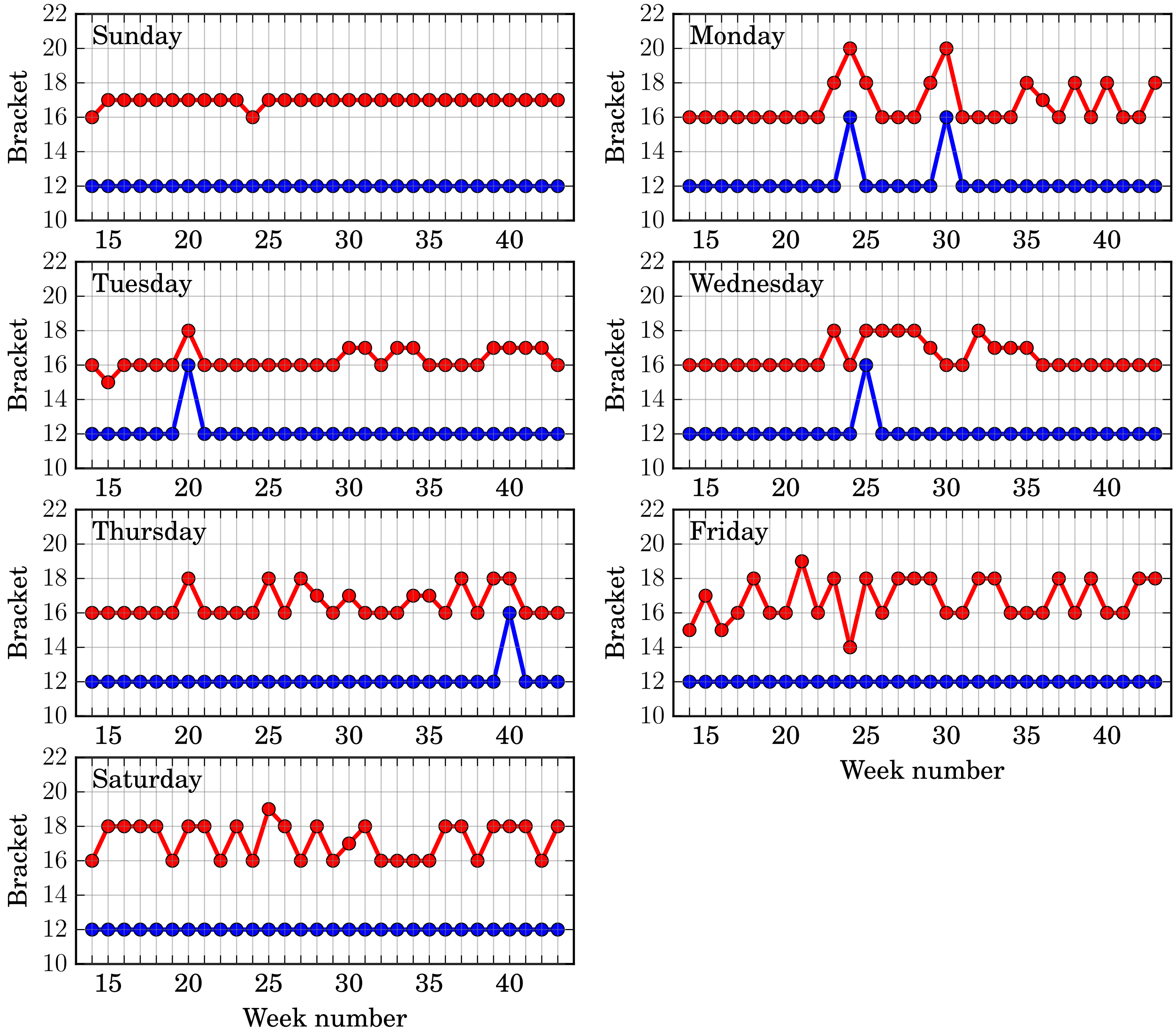

Figure 7.1: Schedule for flight 5 per day-of-week. Blue line is arrival (bottom line), red line is departure (upper line). Each point indicates an allocated slot. Requested slot pair is (12, 17)

Chapter 8

Number of possible slot pairs

The last part of the problem statement will be addressed in this chapter, how many slot pairs are possible when the preferences of a market segment are taken into account.

In section 4.4 the preferences of the market segments are determined and in chapter 6 a model is developed to determine the best alternative schedule that can be offered for a flight given the flight preferences and the current slot usage. Using the same model from chapter 6, the number of possible slot pairs that satisfies the preferences of a flight in a certain segment can be determined.

8.1 Available slot pairs with no flexibility

Let start determining the number of possible slot pairs without any flexibility (no shift or omissions allowed) for season-round (operating one day-of-week throughout the season). A table with season-round arrivals and departures can be determined very quickly from the capacities that are also calculated for the model in subsection 6.2.2. For each day-of-week, the minimum number of arrival and departure per bracket for the whole season is determined (easily by choosing the overall minimum value for that bracket for all weeks).

In table 8.1 the number of arrival and departure slots that are available per bracket for a season-round Sunday is given (summer 2015). A possible slot pair can be determined by looking to the minimum number of arrival slots and the departure slots for a given turnaround time.

Using table 8.1 as example, if a flight wants to arrive at bracket 16 (09:00 LT) and have a turnaround of 3 brackets (60-79 minutes), then the model will look up how many slots are left in arrival bracket 16 and how many departure slots are left in bracket 19 (13:00 LT) and choose the minimum of these two. There is one slot for arrival in bracket 16 and two slots for departure in bracket 19, so one slot pair for season-round is possible. Using this method for all brackets and different turnaround times, will lead to the number of possible slot pairs which is shown in table 8.2, where 90 slot pairs are possible (with an arrival between bracket 13 and

28). A TAT of 2 is a turnaround time between 40-59 minutes, TAT of 3 between 60-79 minutes etcetera. This method will be applied to all day-of-week, and using the minimum for all days for a certain bracket, all-season-round (operating all days) slot pairs can be determined. In that case, for a TAT of between 2 and 6 brackets, there are only 16 slot pairs available (with an arrival between bracket 13 and 28).

Table 8.1: Availability season-round slots for Sunday

Bracket	..	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	..
ARR	..	4	1	0	1	1	5	1	1	1	1	2	0	1	1	1	5	..
DEP	..	1	1	1	1	2	1	2	9	4	3	1	4	2	4	2	1	..

Table 8.2: Slot pair possibilities for different turnaround times on Sunday

Bracket	..	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	..
TAT 2	..	1	1	0	1	1	5	1	1	1	1	2	0	1	1	0	3	..
TAT 3	..	1	1	0	1	1	4	1	1	1	1	2	0	1	0	1	5	..
TAT 4	..	2	1	0	1	1	3	1	1	1	1	2	0	0	1	1	1	..
TAT 5	..	1	1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	..
TAT 6	..	2	1	0	1	1	4	1	1	1	1	0	0	1	1	1	2	..

8.2 Possible slot pairs with flexibility

For this part only one sample flight (that is representative for a market segment) is required which contains information about the cost for shifting and omitting a flight and the maximum shift and omitting percentage.

Restricting brackets is flight specific and will not be considered, so no brackets will be excluded. The maximum turnaround, is determined by the number of shifts it is allowed to make before a flight is omitted, as is explained in subsection 6.2.3.

The simple method described in section 8.1 will not work when shifts or omissions are allowed and therefore the model of chapter 6 will be used. Given a sample flight, the model should keep adding flights that satisfy the maximum shift and omitting percentage for each day-of-week individually and for different turnaround times. This is an iterative process whereby during process the quality of the schedule decreases till it violates the constraints.

The following example instance, with two different preferences, will be used for calculations:

i	P^i_{shift}	P^i_{omit}	MAX^i_{shift}	MAX^i_{omit}
1	1.0	2.29	0.60	0.29
2	1.0	11.80	0.87	0.05

These flights are chosen such that flight 1 has a low shift versus omitting cost, which means that flight 1 does not want to shift much and will be omitted quicker, while the second flight has a high shift versus omitting ratio, which means that this flight allows more shifting to avoid an omission. If you compare the omitting percentages between the two flights, it is visible that flight 1 has relatively more omissions than flight 2 and that flight 2 has relatively more shifts than flight 1.

With the explanation above, the number of possible slot pairs for flight 1 on Sunday is in total 419 (for different turnaround times between 2 and 6 brackets and an arrival between bracket 13 and 28). For flight 2, there are 539 slot pairs possible as shown in table 8.4 (again for different turnaround times between 2 and 6 brackets and an arrival between bracket 13 and 28).

Table 8.3: Possible slot pairs for flight 1

Bracket	..	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	..
TAT 2	..	5	4	3	5	5	11	7	4	4	7	5	5	3	2	2	8	..
TAT 3	..	5	5	3	6	11	7	5	3	6	5	6	2	2	2	3	14	..
TAT 4	..	5	5	4	8	6	7	5	6	4	6	5	2	2	3	12	8	..
TAT 5	..	5	6	9	7	5	7	7	4	6	5	4	1	3	5	3	7	..
TAT 6	..	6	10	6	5	4	8	5	5	3	4	4	3	4	3	7	5	..

Table 8.4: Possible slot pairs for flight 2

Bracket	..	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	..
TAT 2	..	6	6	5	6	8	13	7	6	5	8	7	7	3	3	3	11	..
TAT 3	..	6	6	5	8	13	8	6	5	8	7	8	4	3	3	6	16	..
TAT 4	..	6	6	8	10	7	8	6	7	6	8	6	3	3	4	15	11	..
TAT 5	..	6	8	11	7	6	8	8	5	8	6	5	2	4	8	4	9	..
TAT 6	..	8	11	7	7	5	8	6	7	5	5	6	6	6	4	4	9	..

8.3 Conclusion

As shown in table 8.3 and 8.4, when the requirements are relaxed, the number of possible slot pairs increases as expected.

Chapter 9

Cases

The model presented in chapter 6 has already been used for Schiphol Group in two cases. In this chapter these two cases will be discussed.

9.1 Gebruiksprognose 2017

For the 'Gebruiksprognose 2017' (the expected usage at Schiphol and noise emissions), Schiphol Group used the SIR files of summer 2016 and coming winter 2016/2017. To these SIR files flights are added that are expected to operate in winter 2016/2017 and summer 2017. In the SIR file, the number of arrival and departures were not balance (more departures), so some departures were swapped to arrivals. To ensure that the bracket capacities are not exceeded, the model in chapter 6 was used without many constraints (for example, arrival should happen within a certain timespan, while the departure flight was not considered). For summer 2017 additional flights are added, which should have a possible slot pair, again without many constraints.

In total, roughly 12,000 flights are added over the two seasons. For summer 2017, 7502 movements (19 flights) were added. The instance that was used, is shown below. The default solver of PuLP was not able to return a solution within 60 minutes, while the commercial solver Gurobi returned a solution in 78 seconds.

9.2 Additional arrivals or departures in the morning

During daytime, the brackets are usually 20 minutes, except two brackets in the morning: 07:20-07:49 LT and 07:50-07:59 LT. Schiphol Group and Luchtverkeersleiding Nederland (LVNL) want to change this to brackets of 20 minutes each. This change gives the opportunity to add more arrivals or departures to one of these brackets. As part of the decision making, the model was used with the possible slot pairs at these times like in chapter 8. With the possible slot pairs, a quick overview is given which movement is scarce. The model is used to investigate slot pairs for different turnaround times, are there (good) slot pairs possibilities for the different market segments when more arrivals are added? The model answered this question.

i	D^i	Q^i_{ARR}	Q^i_{DEP}	R^i_{ARR}	R^i_{DEP}	TAT^i_{min}	TAT^i_{max}	P^i_{shift}	P^i_{omit}
1	2,3,5,7	43	56	**	**	9	18	4.6	460
2	2,4,5,7	11	26	**	**	15	21	4.6	460
3	2,4.6	44	49	**	**	5	12	4.6	460
4	1,2,3,4,5,6,7	11	18	**	**	12	27	4.6	460
5	2,3,5,6,7	13	33	**	**	12	24	4.6	460
6	1,2,3,4,5,6,7	31	44	**	**	12	18	4.6	460
7	1,2,3,4,5,6,7	30	48	**	**	1	50	4.6	460
8	1,2,3,4,5,6,7	30	48	**	**	1	50	4.6	460
9	1,2,3,4,5,6,7	13	25	**	**	1	50	4.6	460
10	1,2,3,4,5,6,7	36	53	**	**	1	50	4.6	460
11	1,2,3,4,5,6,7	14	18	**	**	4	12	4.6	460
12	1,2,3,4,5,6,7	26	35	**	**	12	18	4.6	460
13	1,2,3,4,5,6,7	13	21	**	**	8	18	4.6	460
14	1,2,3,4,5,6,7	19	20	**	**	1	2	1	100
15	1,2,3,4,5,6,7	28	29	**	**	1	2	1	100
16	1,2,3,4,5,6,7	37	38	**	**	1	2	1	100
17	1,2,3,4,5,6,7	50	51	**	**	1	2	1	100
18	1,2,3,4,5,6,7	47	48	**	**	1	2	1	100
19	1,2,3,4,5,6,7	41	42	**	**	1	2	1	100

** All brackets that requires a night slot: brackets 1-10 and 58 for arrivals and brackets 1-9, 57 and 58 for departures.

Chapter 10

Conclusion & Recommendations for future research

In this thesis, a model is formulated to determine the best alternative slot pair given that the requested slot pair is not available. This is done by analysing the flight schedules of airlines. This analysis focuses on the number of changes and omissions per flight number per day-of-week. From these results, the cost for a deviation, a shift or omission, is determined. Using these costs in the model, the model will allocate the closest slots to the requested ones and will omit the flight if the deviation from the requested pair is too big. Because of the different costs for flights, flights with a higher shift or omission cost, will have priority during the slot allocation.

For solving the model an open-source MIP solver is used, which has acceptable running times (less than 60 minutes) for small instances ($\sim$ 5-10 flights, depending on constraints). In most cases this default solver can be used, because Schiphol Group is not going to add thousands of movements at once. However, when the instances are bigger, the running time increases quickly (also depending on the constraints in the instance). A heuristic can be developed to decrease the running time, but the quality of the solution will also decrease. Since Schiphol Group needs optimal schedules that could be presented to airlines, it is decided that the highest quality solution should be obtained, which can be solved by the exact method. When a heuristic is used to obtain the exact solution, the running time increases and the advantage of obtaining an acceptable solution (but not exact) within a reasonable time for a complicated problem disappears.

The possible slot pairs, which are calculated in chapter 8, it is demonstrated that when the preference of flying season-round at the same time is relaxed, the number of possibilities increases. Also, it is important to mention that the number of slot pairs is decreases during the season. At Slot Return Date (before the season starts), enough perfect slot pairs are possible (all-season-round at same times), but during the season this will decrease quickly, mainly because many ad hoc flights are requested. These ad hoc flights reduce the number of perfect slot pairs, because they do not request slot series.

10.1 Recommendations for future research

The current model is written for flights of airlines that are not based at Schiphol Airport. An aircraft that is based at an airport, will be positioned at the airport at the end of the day and starts its schedule the next day from that airport. For an airline is would be interesting to know how many rotations it can make on a day (number of departures and arrivals) with that aircraft given certain destinations, flight times and turnaround times.

Finally, for Schiphol Airport it is important to have a forecast how schedules are developing in the future. Depending on the segment of a flight, flights are added to current schedule. Given the market demand, preferences of airlines, which brackets will be filled first? This will be similar to the case in section 9.1, but more accurate.

Bibliography

[1] Schiphol Group (by A. Kappers), "Transport and Traffic statistics Amsterdam Airport Schiphol," http://www.schiphol.nl/SchipholGroup/Company1/Statistics/TransportAndTrafficStatistics.htm, Accessed: 08 Apr 2016.

[2] Amsterdam Airport Schiphol, "Schiphol Traffic Review 2015," https://www.schiphol.nl/B2B/RouteDevelopment/NewsPublications1/RouteDevelopmentNews/SchipholTrafficReview2015.htm, Accessed: 01 Apr 2016.

[3] Amsterdam Airport Schiphol, "Airport Facts," http://www.schiphol.nl/B2B/RouteDevelopment/AirportFacts2.htm, Accessed: 20 Mar 2016.

[4] Airbus, "Globlal Market Forecast 2016-2035," http://www.airbus.com/company/market/global-market-forecast-2016-2035/, Accessed: 04 Mar 2016.

[5] Boeing, "Traffic and Market Outlook," http://www.boeing.com/commercial/market/long-term-market/traffic-and-market-outlook/, Accessed: 04 Mar 2016.

[6] Schiphol Werkgroep Amstelveen/Buitenveldert, "Schiphol mag minder groeien," http://www.stichtingswab.nl/schiphol-mag-minder-groeien, Accessed: 01 Mar 2016.

[7] International Air Transport Association, *Worldwide Slot Guidelines (Effective August 2015)*, International Air Transport Association (IATA), 2015, 7th Edition English Version.

[8] International Air Transport Association (IATA), "Worldwide Slot Guidelines - Latest list of all level 2 and level 3 Airports," http://www.iata.org/policy/infrastructure/slots/Pages/slot-guidelines.aspx, Accessed: 08 Mar 2016.

[9] Stichting Airport Coordination Netherlands, "About SACN - Introduction," http://www.slotcoordination.nl/about-sacn, Accessed: 08 Mar 2016.

[10] Czerny, A., *Airport Slots: International Experiences and Options for Reform*, Ashgate Publishing, Ltd., 2008.

[11] Airport Coordination Limited (ACL), "UK Slot Allocation Process and Criteria," http://www.acl-uk.org/UserFiles/File/UK%20Slot%20Allocation%20Process%20and%20Criteria%20V7.pdf, Accessed: 24 Mar 2016.

[12] Castelli, L., Pellegrini, P., and Pesenti, R., "Airport slot allocation in Europe: economic efficiency and fairness," *International Journal of Revenue Management*, Vol. 6, 2016, pp. 28–44.

[13] Cao, J.-M. and Kanafani, A., "The value of runway time slots for airlines," *European Journal of Operational Research*, Vol. 126, No. 3, 2000, pp. 491 – 500.

[14] Stichting Airport Coordination Netherlands, "SCR Crash Course," `http://www.slotcoordination.nl/crash-course`, Accessed: 25 Apr 2016.

[15] Rijksoverheid, "Advies over uitvoering selectiviteitsbeleid Schiphol," `https://www.rijksoverheid.nl/documenten/rapporten/2012/12/17/rapport-inzake-advies-over-uitvoering-selectiviteitsbeleid-schiphol`, 10 December 2012, Accessed: 12 Apr 2016.

[16] Bewoners Aanspreekpunt Schiphol (BAS), "Baangebruikcijfers gebruiksjaar 2015," `http://www.bezoekbas.nl/userfiles/Baangebruikcijfers-gj2015.pdf`, Accessed: 05 Mar 2016.

[17] International Civil Aviation Organization, *Manual on the Regulation of International Air Transport*, International Civil Aviation Organization (ICAO), 2004, Doc 9626, Part 4.1.

[18] Carleton University - Systems and Computer Engineering (by John W. Chinneck), "Practical Optimization: A Gentle Introduction," `http://www.sce.carleton.ca/faculty/chinneck/po.html`, Accessed: 12 Jul 2016.

[19] Chen, D., Batson, R., and Dang, Y., *Applied Integer Programming: Modeling and Solution*, Wiley, 2010.

[20] Stanford University (by John Duchi), "EE364b: Convex Optimization II: Branch and Bound Methods," `https://web.stanford.edu/class/ee364b/lectures/bb_slides.pdf`, Accessed: 24 May 2016.

[21] Mccollum, B., Mcmullan, P., Burke, E. K., Parkes, A. J., and Qu, R., "The Second International Timetabling Competition: Examination Timetabling Track," September 2007.

[22] The Python Software Foundation, "The official home of the Python Programming Language," `https://www.python.org/`, Accessed: 01 Aug 2016.

[23] PythonHosted, "Optimization with PuLP," `https://pythonhosted.org/PuLP/`, Accessed: 01 Aug 2016.

[24] Optimization Online (by Stuart Mitchell and Michael O'Sullivan and Iain g), "PuLP: A Linear Programming Toolkit for Python," `http://www.optimization-online.org/DB_FILE/2011/09/3178.pdf`, 5 September 2011, Accessed: 01 Aug 2016.

[25] COmputational INfrastructure for Operations Research, "Open source for the operations research community," `http://www.coin-or.org/`, Accessed: 01 Aug 2016.

[26] COmputational INfrastructure for Operations Research, "COIN-OR Branch-and-Cut MIP Solver," `https://projects.coin-or.org/Cbc`, Accessed: 01 Aug 2016.

[27] IBM CPLEX Optimizer, "CPLEX Optimizer," `https://www.ibm.com/software/commerce/optimization/cplex-optimizer/`, Accessed: 01 Aug 2016.

[28] Gurobi Optimization, "State of the Art Mathematical Programming Solver," `http://www.gurobi.com/products/gurobi-optimizer`, Accessed: 01 Aug 2016.

[29] FICO, "FICO Xpress Optimization Suite," `http://www.fico.com/en/products/fico-xpress-optimization-suite`, Accessed: 01 Aug 2016.

[30] Schiphol Group, "Zomer op Schiphol: Ruim 10,6 miljoen reizigers in juli en augustus," `http://nieuws.schiphol.nl/zomer-op-schiphol-ruim-106-miljoen-reizigers-in-juli-en-augustus/`, 30 June 2015, Accessed: 29 Jul 2016.

[31] Stichting Airport Coordination Netherlands, "Declared Capacity," `http://www.slotcoordination.nl/declared-capacity`, Accessed: 14 Mar 2016.

[32] Stichting Airport Coordination Netherlands, "Night regime AMS," `http://www.slotcoordination.nl/allocation-night`, Accessed: 17 Mar 2016.

Appendix A

Bracket list Schiphol Airport

In table A.1 the bracket list at Schiphol Airport for summer 2016 is given. Each bracket is allowed to overbook with two movements. For the computations in this thesis, an overbooking of one slot per bracket is used.

Table A.1: Bracket list summer 2016 [31]

Bracket nr.	**Time from (LT)**	**Time to (LT)**	**# Arrivals**	**# Departures**
1	00:00	00:59	24	25
2	01:00	01:59	24	25
3	02:00	02:59	24	25
4	03:00	03:59	24	25
5	04:00	04:59	24	25
6	05:00	05:59	24	25
7	06:00	06:19	8	10
8	06:20	06:39	8	10
9	06:40	06:59	8	*14
10	07:00	07:19	12	25
11	07:20	07:49	18	20
12	07:50	07:59	11	6
13	08:00	08:19	23	13
14	08:20	08:39	23	13
15	08:40	08:59	22	12
16	09:00	09:19	23	13
17	09:20	09:39	12	25
18	09:40	09:59	12	25
19	10:00	10:19	12	25
20	10:20	10:39	12	25
21	10:40	10:59	12	14
22	11:00	11:19	23	13
23	11:20	11:39	23	13

24	11:40	11:59	12	25
25	12:00	12:19	12	25
26	12:20	12:39	12	25
27	12:40	12:59	12	14
28	13:00	13:19	23	13
29	13:20	13:39	22	12
30	13:40	13:59	23	13
31	14:00	14:19	12	24
32	14:20	14:39	12	25
33	14:40	14:59	12	25
34	15:00	15:19	12	14
35	15:20	15:39	23	13
36	15:40	15:59	23	13
37	16:00	16:19	23	13
38	16:20	16:39	12	25
39	16:40	16:59	12	25
40	17:00	17:19	12	24
41	17:20	17:39	12	25
42	17:40	17:59	12	25
43	18:00	18:19	12	14
44	18:20	18:39	23	13
45	18:40	18:59	23	13
46	19:00	19:19	22	12
47	19:20	19:39	23	13
48	19:40	19:59	23	13
49	20:00	20:19	12	25
50	20:20	20:39	12	25
51	20:40	20:59	12	24
52	21:00	21:19	12	25
53	21:20	21:39	12	*25
54	21:40	21:59	12	14
55	22:00	22:19	12	14
56	22:20	22:39	12	14
57	22:40	22:59	12	8
58	23:00	23:59	24	25

*In winter 2014, summer 2015, winter 2015 the capacity for bracket 9 was 10 departures and the capacity for bracket 53 was 13 departures.

Appendix B

Declared capacity at Amsterdam Airport Schiphol

In table B.1 the declared capacity of Amsterdam Airport Schiphol per season is given. Some of the capacities in table B.1 are left empty, since they were not declared. So for summer 2015 (S15), for the night, only the number of arrival slots and the numbers of departure slots were given but no overall number, while for winter 2016/2017 (W16) only the overall number of night slots were given.

The definition of a night slot is [31, 32]:

- An arrival between 23:00 and 07:19 LT
- A departure between 22:40 and 06:59 LT

A night slot that is operated between 23:00 and 06:59 LT (runway times) becomes night movement.

With the definition above and the bracket list in table A.1, a night slot is a slot in the following bracket numbers:

- Arrival: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 58
- Departure: 1, 2, 3, 4, 5, 6, 7, 8, 9, 57, 58

Table B.1: Capacity declaration per season [31]

Season	Total slots	Night slots for arrival	Night slots for departure	Total night slots
W14	188.600	8.080	3.320	-
S15	275.800	14.100	8.580	-
W15	188.600	-	3.320	11.400
S16	300.000	-	8.866	23.436
W16	185.000	-	-	10.735

In the SIR file, more movements are allocated than the declared capacity. This is because the slot coordination expects that not all allocated slots will be used. For the model in chapter 6, the declared capacities are increased with 10%.