Technische Universiteit Delft

Evaluation of pairing requests in dynamic airline crew rostering

Thesis report

M. Beulen

TU Delft

Thesis report

Evaluation of pairing requests in dynamic airline crew rostering

by

M. Beulen

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday February 15, 2019 at 10:00 AM.

Student number:	4133714	
Thesis committee:	Prof. dr. R. Curran,	TU Delft
	Dr. ir. B. F. Lopes dos Santos,	TU Delft
	Dr. ir. A. Cervone,	TU Delft
	Ir. drs. L. Scherp	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Preface

It was not until I reached the age of sixteen when I boarded an aircraft for the very first time. The flight took me from Düsseldorf in Germany to Birmingham in England, and from that very short flight onward, I was drawn to everything involved in the process of making that flight possible. Still, today, walking around airports across the world or being on flights, long or short, always amazes me in many ways. First of all, I just love to be among the atmosphere of passengers coming from and going to destinations all over the world, realizing that every individual has a unique story that brings them to something or someone far away from home. Second of all, I enjoy the multidisciplinary nature of such places in which a demanding operational environment requires not only all technical processes to run smoothly but also brings about countless challenges for societies and commercialism. Third and most important of all, it is the pragmatic problem solver in me that gets excited when being confronted with such a vast amount of operational challenges in this high stakes environment.

This latter driver has taken me to where I am today, finalizing my Master of Science in Aerospace Engineering with a graduation research project at the intersection of the academic world and the airline industry practice. I have had the opportunity to perform my research as part of a longer-term TU Delft research project in collaboration with the Flight Operations department of KLM Dutch Royal Airlines. This both lifted research to a higher level and confirmed the relevance of my work. My research concerned the topic of crew preferences in an airline crew scheduling context which presented challenges that I had not considered before, walking across airports or being on flights. While crew scheduling might seem like just another allocation problem, it quickly became clear to me that it is *people* that are being scheduled. With these people come personal stories and because of this, crew scheduling will always require both a technical and a social approach. In my opinion, this has been the learning experience that completed my studies. Operations optimization in airline crew scheduling is powerful and indispensable but will always need to be contextualized and nuanced from a social perspective. Without a doubt, on all the aircraft I will board in the future, this will come to my mind.

The project would not have been the experience it was without a few very important people on my side whom I would like to acknowledge. First of all, I would like to thank Lennart, my daily supervisor, for being the absolute best guidance I could have hoped for in the process of my MSc thesis research. Whether it was well-organized databases, critical feedback on content and progress in weekly meetings or a personal conversation, you never ceased to provide the boundary conditions for me to thrive. Your research is still to be continued, but I cannot stress enough that I have the utmost respect for the way you manage the project. Second of all, I would like to thank Bruno, my professor and supervisor, for his critical attitude towards the academic scope of the project, the involvement of KLM and for his care for the prosperity of the team. You were the one who could always promptly point out the issues or questions that still needed attention. Third of all, I would like to thank my colleagues at KLM who have been involved in the project, always finding a moment to support my work and challenge my assumptions and findings. Thank you Nico Scheeres, Marco van Vliet, Karel Bockstael, Thea Groot, Shirah van den Hoek, Taco Eyck and Leon Ceelen for sharing your many insights from your years of experience. Fourth of all, I would like to thank my fellow TU Delft team members Richard Janssen, Toine Hooijen, Joey van Kempen and Michiel van Amerongen from the research project team at KLM's Operations Control Center. You guys have been the best sparring partners for research directions and crazy venture ideas at our desk isle with whizzing computers, which grew bigger every few months. And last of all, I would love to thank my wonderful friends and family who have supported me every step of the way. You brought the company, the joy and the wise words that I needed and I cannot thank you enough for that!

M. Beulen
Amsterdam, January 2019

Executive summary

In operational airline crew scheduling, one of the major tasks is to allocate flight activities to individual crew members. This task is referred to as the crew rostering problem; an optimization problem that aims to assign flights to crew members in a cost-effective way, while also considering preferences of crew members.

Flights are combined into multiple-day activities called pairings that consist of flight duty days and rest period days. For intercontinental long-haul destinations specifically, pairings may take more than a week. During flight duty days of these long-haul pairings, crew members are not situated at the airline base location. During rest period days, however, they have the opportunity to be at their homes or with their families. The pairings that crew members on long-haul crew divisions operate, thus, directly impact their life apart from work as well as the locations at which they are situated during flight duty days. Therefore, crew preferences are highly valued by crew members, as these provide their only means to influence duties. This is a critical reason for crew preferences to be a recurrent topic in negotiations on collective labour agreements between crew unions and airlines that influence crew resources. As crew costs are the biggest expense for airlines next to fuel (Belobaba et al., 2015), crew preferences are considered in an environment that is predominantly designed to manage crew resources as efficiently as possible.

For long-haul cockpit crew in European airlines, crew preferences can be expressed via requests for the operation of specific pairings. However, the problem with these pairing requests is that the requests concern pairings that commence months later than the pairings considered by the crew rostering problem. This problem is typically solved only one month before operation of the pairings. It is impractical to consider the pairing requests as part of the crew rostering optimization problem, since the problem size of the optimization problem becomes too large to be solved within practical computation time. In other words, a decision on whether or not to grant a certain pairing request has to be made with limited information available on the consequences of that decision since it concerns a part of the schedule that is not optimized for. When a pairing request is granted for a crew member, this invokes a pre-assigned pairing for this crew member. Such pre-assigned pairings constrain the construction of this crew member's roster in the crew rostering problem, as only a limited number of options exist to efficiently construct this crew member's roster. When many of these pre-assigned pairings exist in the schedule, this leads to inefficiencies in the schedule. Such inefficiencies cause loss days without productive activity in the schedule, which are an indicator of inefficient use of crew resources leading to higher crew costs. The process of evaluating pairing requests, thus, impacts the amount of crew resources required to operate a schedule.

Although the relevance of crew preferences in airline scheduling practices is evident, the topic has received little attention in literature. Literature on crew preference management in airline scheduling is focused on formulating crew satisfaction objectives as part of the optimization objective of the crew rostering problem. With such an approach, it is assumed that crew preferences concern pairings that are assigned within the crew rostering problem. As was stated before, this is not the case for pairing requests that commence in a wider planning horizon than the crew rostering problem. Moreover, current literature approaches the crew rostering problem as a static problem with a deterministic input, rather than a dynamic problem of which the input is subject to change over time. The problem with approaching the crew rostering problem as static is that variability of resources over time is not considered. In addition, this static approach does not provide a method for modeling pairing requests as a means of expressing crew preferences. Therefore, modeling pairing requests within a crew rostering problem has not been considered before. This thesis challenges the static approach to airline crew rostering and provides the first integrated means for evaluating pairing requests in a broader planning horizon. The objective is to make recommendations on methods to evaluate pairing requests while capturing the dynamic nature of the crew rostering problem.

The research has been captured in a design framework that consists of three major components. Firstly, a static rostering model has been developed to measure the effect of pairing requests on crew resources in the crew rostering problem. In the static rostering model, the optimization model is constrained by a minimum of pairing requests that need to be granted. Secondly, a dynamic rostering model has been developed to capture the dynamic nature of the crew rostering problem. This dynamic model is the first airline crew rostering model to adopt a rolling rostering approach. With this approach, the existing schedule is appended with a small time horizon of activities for each time iteration, while taking into account pre-assigned pairings that already exist in the schedule. The dynamic model can be used for simulations on dynamic crew rostering problems with incoming pairing requests that need to be evaluated. In these simulations, pairing requests are submitted by crew members and evaluated by the third major component in the research. Thirdly, three pairing request evaluation algorithms have been developed. These algorithms have been integrated in the dynamic rostering model.

The first algorithm, the random-based pairing request evaluation algorithm, is based on stochastic modeling of the pairing request evaluation decision. The second algorithm, the rule-based pairing request evaluation algorithm, has been inspired by current scheduling practices in which pairing requests are evaluated manually. The length of the void in the schedule that is induced by pre-assigning a pairing is an important driver for the evaluation of pairing requests in this manual process. The rule-based algorithm is designed to resemble and automate this manual request evaluation decision. The third algorithm, the classification-based pairing request evaluation algorithm, is designed to incorporate a feedback mechanism in the request evaluation decision. A supervised machine learning classification model has been trained with data on pairing requests with their eventual impact to the rostering problem. To model a crew rostering scenario that resembles airline practices, the input of both the static and dynamic rostering model have been driven by analyses of historical airline data from a major European airline. A case study, based on airline crew rostering practices has been formulated to illustrate the viability and practical use of the models and algorithms.

Results of multiple experiments have resulted in recommendations on how pairing requests can be evaluated in dynamic airline crew rostering. The effects of pairing requests to the number of required FTEs to operate all the pairings in a schedule have been investigated. The number of required FTEs represents the crew resources that an airline needs for the feasible operation of the pairings within a crew division. These crew resources can be translated into financial resources required to operate a schedule feasibly. In the experiments that were formulated, the linear programming rostering model was constrained to a minimum of to be granted pairing requests. Varying this minimum of pairing requests resulted in insights on the (financial) effects of pairing requests. A level of granted pairing requests can be determined at which the available set of crew members is not sufficient to cover all the pairings in the schedule. For a workforce of 72 FTEs, the level of granted pairing requests at which this workforce was not sufficient to operate the pairings was determined at ≥ 31 granted pairing requests out of the 71 pairings in the schedule. This approach can be used by airline scheduling departments to explore the allowance of pairing requests in crew preference management.

For the comparison of methods for pairing request evaluation, pairing request data that was available in a case study with a major European airline served as a benchmark for comparison of the methods for evaluating pairing requests that have been developed in this research. For the case study airline, 72 FTEs are required for feasible operation of the pairings while 15.51 pairing requests can be granted each week, on average. With the implementation of the random-based pairing request evaluation algorithm, 72.57 FTEs are required for feasible operation of the pairings while an average of 7.31 pairing requests can be granted each week. With the implementation of the rule-based pairing request evaluation algorithm, 74.67 FTEs are required for an average of as much as 16.02 granted pairing requests each week. This shows its potential for practical implementation, as the rule-based approach is inspired by current scheduling practices. Lastly, with the implementation of the classification-based pairing request evaluation algorithm, 71.48 FTEs are required for an average of 5.59 granted pairing requests each week. Although this number of granted pairing requests is relatively low compared to the other methods, the classification-based algorithm is the only method for which the provided set of crew members of 72 FTEs is sufficient. Training the classification-based algorithm with simulation-based rostering data has proven to be an effective approach for a pairing request evaluation method.

It can be concluded that the pairing request evaluation algorithms are viable methods for the evaluation of pairing requests. Selection of an appropriate method depends on the incentives for pairing request evaluation within an airline practice. In airline practices, the algorithms can be used as a decision mechanism on top of the crew rostering software that is used by an airline. This implementation step has the potential to mitigate hours of manual trade-off making for pairing request evaluation by scheduling personnel.

List of Tables

List of Figures

List of Symbols

$c_r^l(t)$ cost of roster $r \in R_l(t)$ for crew member $l \in L(t)$

$(n_v^i)_r^l$ number of voids v in personalized roster $r \in R_l$ for crew member $l \in L$ with a void length of i number of days

$(n_v^i)_r^l(t)$ number of voids v in personalized roster $r \in R_l(t)$ for crew member $l \in L(t)$ with a void length of i number of days

$C(t-1)$ set of carry-in pre-assigned activities as a boundary condition for time iteration t

c_p penalty cost for assigning a covered pairing

c_r^l cost of roster $r \in R_l$ for crew member $l \in L$

c_r^s cost of roster $r \in R_s$ for slack crew member $s \in S$

$c_r^s(t)$ cost of roster $r \in R_s(t)$ for slack crew member $s \in S(t)$

c_v unindexed cost for a void in the roster

c_v^i indexed cost of void in the roster with a void length of i number of days

c_{BSA} cost for a base arc

c_b bonus cost for assigning a requested pairing

c_{PA} cost for a pairing arc

c_{PEA} cost for a pairing end arc

c_{PSA} cost for a pairing start arc

c_{SEA} cost for a schedule end arc

c_{SSA} cost for a schedule start arc

c_{VA} cost for a pre-assignment arc

c_{VEA} cost for a pre-assignment end arc

c_{VSA} cost for a pre-assignment start arc

e_p^l pairing $p \in P$ requested by crew member $l \in L$

$e_p^{r,l}$ pairing $p \in P$ chosen for personalized roster $r \in R_l$ for crew member $l \in L$

H planning horizon spanning $[t_0, t_{max}]$

k_{index} cost index

L set of crewmembers of the crew type considered

l crew member $\in L$

$L(t)$ set of crew members of the crew type considered at time iteration t

$l_{carry-in_{length}}$ carry-in pre-assignment length

n_q number of assigned pairings in a roster that have been requested

n_y number of submitted pairing requests per crew member per week

n_{FTE} number of required FTEs for feasible operation of the pairings in a roster

P set of pairings to be covered by the crew type considered

p pairing $\in P$

$P(t)$ set of pairings to be covered by the crew type considered at time iteration t

pa_{dest} pairing destination ID

pa_{id} pairing ID

$pa_{p_{request}}$ pairing request probability

pa_{rp} pairing rest period days

pa_t pairing departure time

q minimum desired number of requested and assigned pairings in the schedule

$Q(t-1)$ set of pre-assigned activities as a results of granted pairing requests as a boundary condition for time iteration t

q_r^l number of requested and assigned pairings in personalized roster $r \in R_l$ for crew member $l \in L$

$q_r^l(t)$ number of requested and assigned pairings in personalized roster $r \in R_l(t)$ for crew member $l \in L(t)$

r a personalized roster for one crew member

R_l set of personalized rosters for crew member $l \in L$

$R_l(t)$ set of personalized rosters for crew member $l \in L(t)$ at time iteration t

R_s set of personalized rosters for slack crew member $s \in S$

$R_s(t)$ set of personalized rosters for slack crew member $s \in S(t)$ at time iteration t

S set of slack crew members of the crew type considered

s one slack crew member $\in S$

$S(t)$ set of slack crew members of the crew type considered at time iteration t

t time iteration $\in H$

t_0 initial time iteration

t_i time iteration

t_{max} maximum time iteration

x_r^l personalized roster $r \in R_l$ chosen for crew member $l \in L$

x_r^s personalized roster $r \in R_s$ chosen for slack crew member $s \in S$ if true

Pr(X=x) A pseudo-random number X. It is either granted $(X = 1)$ or rejected $(X = 0)$

Contents

1

Introduction

In operational airline crew scheduling, an important task is to ensure that all the flights are operated by the set of crew members that is available to the airline. Therefore, an airline scheduling department needs to solve an assignment problem in which work rosters with flight activities are constructed for all crew members. This is achieved by solving the crew rostering problem; an optimization problem for which the objective is to assign flights to crew members in a cost-effective way, while also considering preferences that crew members have regarding their rosters.

When operating long-haul destinations, crew members can be away from the airline base location for multiple consecutive days. In between duties, they have the opportunity to be at their homes or with their families. The flights that a crew member on long-haul crew divisions operates, impact the locations at which he or she is situated in between these long-haul flights. This can be irregular and challenging and therefore, crew preferences are highly valued by crew members, as these provide their only means to influence duties. Because of its relevance to crew members as individuals, the topic of crew preferences can be crucial in negotiations on collective labour agreements between crew unions and airlines.

One of the possible means that airlines can provide for crew members to express preferences is to allow for requests for the operation of specific flights activities. Such requests, however, may concern flights that depart much later than the activities that are being scheduled at the moment of requesting. To further illustrate this, Figure 1.1 shows a schedule for three crew members that operate long-haul destinations. Within this schedule, several time horizons and milestones are highlighted. The rosters for the three crew members with multiple activities are visualized in a Gantt Chart representation. In the example, these activities are limited to pairings; multiple-day activities that have been created by combining flight activities. These pairings consist of flight duty days followed by rest period days. For intercontinental long-haul destinations, pairings may take more than a week, as is the case in the example.

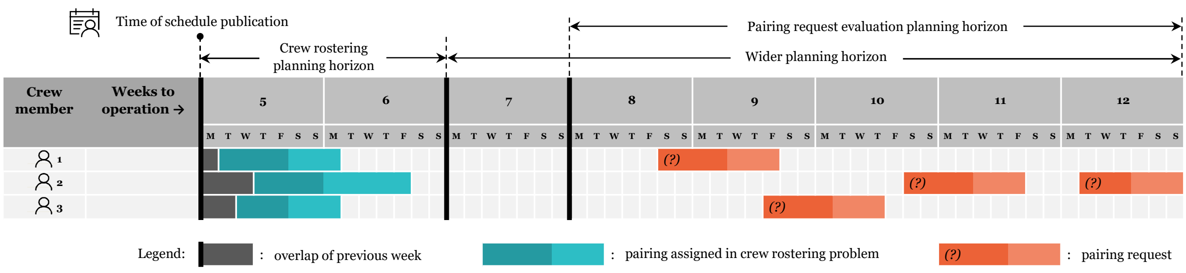


Figure 1.1: Example schedule for 3 crew members in a total planning horizon of 12 weeks

Pairing requests that need to be evaluated are visualized in the figure in orange. The evaluation of a request concerns the decision on whether to grant the request or reject the request for a crew member. To understand the challenge of this evaluation, the time horizons of the example are addressed below:

- *The crew rostering planning horizon* - A roster is constructed for each of the crew members by solving the crew rostering problem. In this planning horizon, a roster is finalized for publication to the crew members. After solving the rostering problem for a specific crew rank, all pairings need to be covered by exactly one crew member. The completed schedule, consisting of the finalized rosters for all the crew members, is published at the time of schedule publication (i.e. 4 weeks before operation in the example).

- *The wider planning horizon* - It is desired to already assign certain types of activities that take place in a wider planning horizon than the crew rostering planning horizon. Pairing requests are an example of such an activity type, as well as holiday leaves or longer-term crew retraining periods.
- *The pairing request evaluation planning horizon* - An airline can enable crew to submit pairing requests for a given time horizon. Given that pairing requests are evaluated in the week of submitting, pairing requests concern those pairings that commence within the pairing request evaluation planning horizon.

Crew costs are the biggest expense for airlines next to fuel costs. What is key to a crew rostering process, therefore, is that crew resources are used efficiently. An example of a desired situation in Figure 1.1 is to assign to the first crew member a pairing that starts on the day right after the pairing that is already assigned. Voids in the rosters are undesirable, as these eventually lead to loss days that have no productive activity. If too many of these inefficiencies exist in the roster, this could lead to the undesired situation of a higher demand for crew members. When a pairing request is granted for a crew member, this invokes a pre-assigned pairing for this crew member. Such pre-assigned pairings constrain the crew rostering problem, as only a limited number of options exist to efficiently construct each crew member's roster. The problem of evaluating pairing requests in a wider planning horizon is that limited information is available on the schedule at the time of the pairing activity. Referring to the figure to give an example; on what grounds should a decision be made for granting or rejecting the pairing request for the first crew member? Therefore, the research that is presented in this thesis report concerns the following research question:

How could pairing requests in the airline crew rostering problem be evaluated?

The objective of the research is to make recommendations on methods to evaluate pairing requests while capturing the dynamic nature of the airline crew rostering problem. To explore the effect of pre-assigned activities that are induced by pairing requests, a rostering model is designed with which this effect can be captured. The output of this model is a solution to the crew rostering problem, i.e. a schedule that covers the pairings in the crew rostering planning horizon. A dynamic rostering model has been developed to capture the dynamic nature of the crew rostering problem. In addition, activities within the wider planning horizon are also considered without considering them as part of the optimization problem what would be impractical because of problem size. This dynamic rostering model is the first airline crew rostering model to adopt a rolling rostering approach. The existing schedule is appended with a small time horizon of activities for each time iteration, while taking into account pre-assigned pairings that already exist in the schedule. The dynamic model is designed for simulations on dynamic crew rostering processes with incoming pairing requests that need to be evaluated. Pairing requests are submitted by crew members in these simulations. To evaluate the pairing requests, three algorithms have been developed that have been integrated in the dynamic rostering model. Experiments have been designed to explore the effect of pairing requests on a crew rostering model and to compare the performance of the pairing request algorithms. Based on the results of these experiments, recommendations are made on capturing the dynamic nature of the airline crew rostering problem and methods to evaluate pairing requests within this problem.

The thesis report is structured as follows. In Chapter 2, an extensive literature review is presented on the broader topic of crew preference management in an airline crew rostering context. The topic is explored and the research gap and relevant methodology to address this gap are identified. Next, the design of the research is presented in Chapter 3. Here, the scope of the research and a framework for addressing the research objective are presented. In Chapter 4, the design of the static rostering model is presented. Following, the dynamic rostering model is presented in Chapter 5. The evaluation methods; the pairing request evaluation algorithms are presented in Chapter 6. Further, the experiments that have been performed in this research as well as the results and discussion on the experiments are presented Chapter 7. To further explore the performance of the models and algorithms, a sensitivity analysis on certain aspects of the models and algorithms are presented in Chapter 8. Finally, Chapter 9 concludes this thesis report with final conclusions on the results of the research as well as contributions to research and recommendations for future research.

2

Literature review

The literature review to which this Chapter is dedicated, contributes to an extensive study on crew preferences as a concept within the airline crew rostering problem. The objective of this literature review is twofold. The first objective is to determine state of the art and to identify a research gap regarding crew preferences in the field of (airline) crew rostering. The second objective is to present and reflect upon literature that can support possible problem approaches that can be chosen to close this research gap and to add novel methodology to the body of research. In this Chapter, Section 2.1 and 2.2 address the first objective of this literature review. Section 2.1 positions the topic of crew preferences in the crew rostering problem of the airline schedule planning process. Section 2.2 covers the concept of crew preferences as handled in the airline industry and other industries and concludes with a gap in the body of research. Subsequently, Section 2.3 covers the second objective of this literature study. Here, an overview of the methodology to address the research objective is presented. The Chapter is concluded with Section 2.4 that provides a synthesis of presented problem approaches and methodologies throughout the Chapter.

2.1. Airline crew rostering in an airline scheduling context

The scheduling of airline crew is recognized to be the last phase of the overall airline schedule planning problem. This problem consists of multiple sequential schedule planning problems, which have all been widely studied in the literature. This Section positions the concept of airline crew rostering and crew preferences into the airline schedule planning problem, to get a better understanding of the problem scope. Note that this Section will not focus on modeling methods, but rather on characteristics of the relevant problems within the airline scheduling domain. A top-down approach is used to reflect the sequential nature of the multiple schedule planning problems. Section 2.1.1 addresses the overall airline scheduling planning problem followed by an introduction of the airline crew scheduling problem which in turn can be further broken down into subsequent planning problems. Firstly, in the airline crew pairing problem, cost-effective flight duties for the crew are produced. Secondly, in the airline crew rostering problem, individual crew members are assigned to these flight duties. Since crew preferences are expressed on an individual level, they are taken into account in the airline crew rostering problem. The emphasis of this overall Section is therefore on the airline crew rostering problem. Characteristics of the airline crew rostering problem are discussed in Section 2.1.2 and approaches to the airline crew rostering problem in Section 2.1.3, while also offering context for the concept of crew preferences.

2.1.1. Positioning of the airline crew rostering problem

The airline industry is a driver for optimisation problems being widely studied within the industry. A tightly competitive market characterizes the industry, along with many rules and regulations, high operational costs and a high level of uncertainty in many operational aspects (Eltoukhy et al., 2016). These same characteristics indicate the complexity of the airline schedule planning problem. Efficient and effective management of resources is crucial, and schedule planning optimisation has therefore been extensively studied in the airline operations research domain (Sherali et al. (2013); Eltoukhy et al. (2016)). The airline schedule planning problem involves both the design of aircraft schedules and crew schedules with an overall objective of maximizing airline profitability.

In line with the problem complexity lies the sheer size of the problem which has been the major reason for the body of research not being able to solve, or even formulate, the airline schedule planning problem as a whole. This has resulted in the decomposition of the airline schedule planning problem into four subproblems that reflect the airline schedule planning subproblems that are solved sequentially (Barnhart et al. (2003a); Belobaba et al. (2015)). These four problems are as follows, in decreasing order of time to operations:

- **Schedule design** - Defining which markets to serve, with what frequency, and how to schedule flights to meet these frequencies (typically 1 to 2 years before operation for airline network decisions, 6 to 12 months before operation for timetable design)
- **Fleet assignment** - Specifying aircraft to assign to each flight (typically 6 to 12 months before operation)
- **Aircraft maintenance routing** - Determining the routing of aircraft to ensure satisfaction of maintenance requirements (typically 1 to 2 months before operation)
- **Crew scheduling** - Selecting which crews to assign to each flight to minimize crew costs (typically 1 month before operation)

The application of operations research techniques has been instrumental in being able to meet the challenges of schedule optimization at airlines. However, the sequential nature of the schedule planning process limits the optimization effects. The different subproblems assume the previous subproblems to be solved and require the solution to previous subproblems as an input. In practise, the problems are very interrelated which indicates that simultaneous optimization would yield a more desired solution to the overall scheduling problem. However, the nature of the problems and their timing in the scheduling chain require that decisions need to be made at different moments in time, up to 2 years in advance. The crew scheduling subproblem is the problem that comes last and it can, in turn, be decomposed into sequential subproblems.

Airline crew scheduling

Crew costs represent a significant portion of an airlines operational expenses, second only to fuel costs. The slightest improvements in utilization can lead to significant cost reductions, and it is, therefore, one of the most studied problems compared to other scheduling planning problems in airline planning going back as far as 1969 (Arabeyre et al., 1969).

Within the airline crew scheduling domain, two types of crew can be identified. The cockpit crew is responsible for flying the aircraft, and the cabin crew is responsible for in-flight passenger service and safety. Together, the cockpit crew and the cabin crew form the the crew complement for a flight which is defined as the required crew for that flight concerning crew type and rank. The two types of crew are scheduled differently since some distinguishing factors are relevant for the crew scheduling problems to be solved. Firstly, cockpit crew is typically qualified to operate only one aircraft fleet type or fleet family at a certain moment in his or her career. Periods of training enable cockpit crew to transfer to operating another aircraft fleet type. Due to strict work qualification rules, cockpit crew usually is inflexible to work across fleet types. Cockpit crew can, therefore, be generalized as divisions or cohorts of crew per fleet family and rank (i.e. captain, first officer, second officer or flight engineer). Cabin crew, on the other hand, is more flexible when it comes to the operation of different fleet types. Moreover, the number of cabin crew members required for a flight varies per flight and passenger count, while the number of cockpit crew is independent of these factors. Therefore, scheduling on a more individual level is more common for cabin crew. Secondly, cockpit crew members are typically paid a substantially higher salary than cabin crew members. Many of the optimization problems in the airline crew scheduling domain have therefore focused on cockpit crew.

Airline crew scheduling assumes a solution for all the previous problems in the planning process that was described in Section 2.1.1. The crew scheduling subproblem is in turn further broken down into two subproblems that are typically solved sequentially because of size, complexity and tractability issues (Barnhart et al. (2003a); Belobaba et al. (2015)). These two subproblems are as follows:

- **Crew pairing problem** - Combining flight legs and layover days into unassigned multiple-day duty periods called pairings
- **Crew rostering problem** - Combining pairings and other activities into individual lines of work and assigning these to crew members

The solution to these two subproblems leads to an overall crew schedule. When discussing airline crew scheduling, it is notable that the terminology for crew schedules and crew rosters is inconsistent in literature. Therefore, this literature review defines schedules as the total solution for all crew members and rosters as individual lines of work. To clarify terminology, Figure 2.1 shows an example of a schedule visualization for a 14-day period for m crew members. The rows in the schedule represent the lines of work for each crew member in a Gantt Chart representation. These lines of work are constructed by combining different types of activities, of which the figure addresses five examples. The first of the two activities are the flight duty days and layover days. Flight duty days represent the days that a crew member is operating flight legs. For long-haul crew, these flight duty days can in practice constitute outbound flight legs from base to destination, recovery time at the destination, and inbound flight legs from destination to base. Layover days, as another type of activity, represent the days following the flight duty days which allow for rest and recovery for the next flight duty days. The pairings are represented by the combination of the flight duty days and the layover days in this visualization. To summarize, duties constitute consecutive flight legs that comprise a working day for a crew member and layovers represent a rest period between these duties. A pairing starts and ends at the same base, and it typically lasts 1-5 days for short- and medium-haul problems and possibly longer for long-haul problems (Kasirzadeh et al., 2017). For example, crew member 1 is assigned to one pairing from day 1 up to and including day 7, and one pairing from day 8 up to and including day 14. The three other types of activities in the figure are typical activities that are relevant when constructing lines of work. Leave of absence represents the days reserved for yearly leaves or planned days off. Simulator training sessions and medical checks represent recurrent moments of evaluation required for crew members to stay qualified for operation. Note that this visualization does not capture the complexity of an actual crew schedule. In practice, many more details yield complexity. Examples of such details are duty start and end times, rules and regulations that limit feasible succession of activities, or crew that operates on a part-time basis.

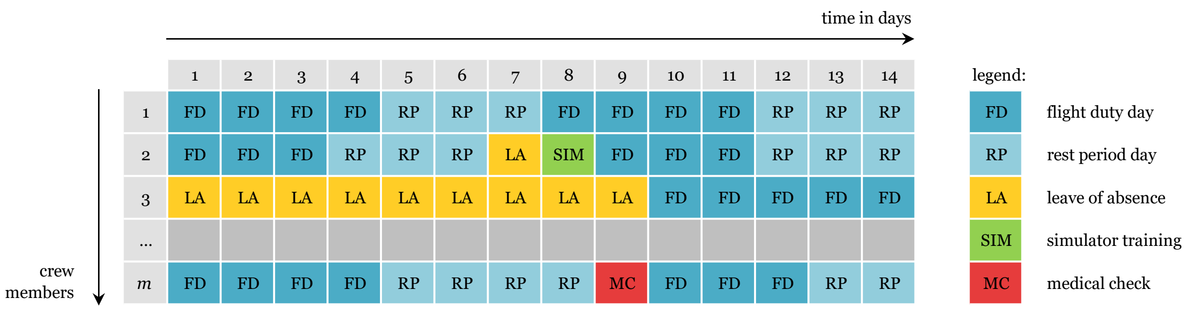


Figure 2.1: Example of a crew schedule visualization for a 14-day period for m long-haul crew members

Airline crew pairing

The crew pairing problem is concerned with the generation of multiple-day duty periods called pairings, in which sets of flight duty days and layover days are sequenced. The problem has been described, among others, in the works of Barnhart et al. (2003b), Belobaba et al. (2015) and Eltoukhy et al. (2016). The crew pairing problem is solved per fleet type, which is in line with the fleet-specific employment of cockpit crew. The objective of the crew pairing problem is to cover all the flights in a schedule exactly once, at minimum cost, while also satisfying all applicable rules and regulations. The solution to the crew pairing problem comes in the form of unassigned pairings that serve as a major input for the subsequent crew rostering problem, where the pairings are assigned to crew.

Work rules and regulations that are enforced through collective labour agreements are of major influence in the airline crew pairing problem. The duties should be sequenced in such a way that the schedule can be operated without violating the agreements. Examples of factors to consider within a certain period are minimum rest time and maximum flying (block) time.

When considering the cost minimization objective, it can be stated that the calculation of crew cost varies widely per airline (Belobaba et al., 2015). Most European, Asian and South American airlines pay a fixed monthly salary with additional bonuses or premiums, depending on the assigned rosters. When using this approach, it is in the interest of an airline to minimize the headcount of the workforce as a whole. Most

Australian, Canadian, UK and USA airlines handle a crew cost function based on production. When using this approach, different pay rates can be handled for flying time and other types of duties, such as reserve duties. The airline interest and cost minimization objective, in this case, is highly dependent on the cost structure of this production-based approach. Although cost structures vary per airline, the crew pairing problem can be solved using tangible costs associated with operating a flight schedule (Medard and Sawhney, 2007). As an input to the crew pairing problem, the flight schedule of an airline is taken to provide the flights that need to be covered and sequenced. Figure 2.2 shows an example of a set of flight legs, represented in a time-space network. In the figure, airport A represents the crew base which is the starting and ending location of each pairing. Airlines either have a single crew base or crew bases on multiple locations. Flight legs in a pairing must sequence each other in both time and space. In the figure, such a pairing is represented by the shaded blue V-shape, consisting of one outbound flight from the base airport A to airport C and one inbound flight from airport C to the base airport A.

Within the airline crew scheduling domain, the crew pairing problem is the subproblem that has received the most attention. Contributions to the recognition of the problem are its significance to total crew cost, the variety of cost structures throughout the industry and the many regulations and contractual issues that increase the modeling complexity.

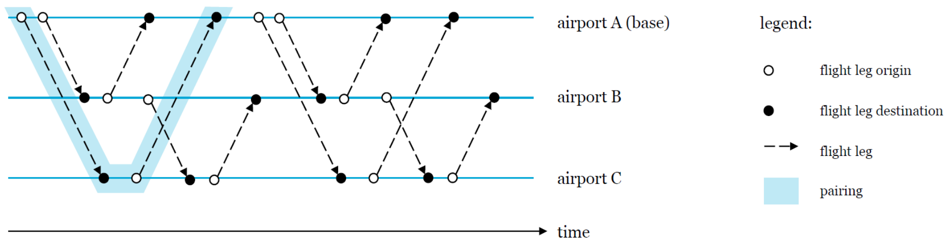


Figure 2.2: Example of a time-space network schedule visualization for three airports connected by flight legs

Airline crew rostering

The crew rostering problem is the follow-up subproblem of the crew pairing problem. It is concerned with creating rosters by combining unassigned pairings and other activities into individual lines of work (Belobaba et al., 2015). Solving this crew rostering problem usually takes place 2-6 weeks before operation of the associated flights and is usually done on a monthly basis. A feasible work schedule, or roster, is constructed for each crew member. The crew rostering problem is also referred to as the crew assignment problem in literature (Kohl and Karisch, 2004) but for reason of consistency, the crew rostering designation will be used in this thesis report. The following two Sections will elaborate on the characteristics of the problem and approaches to the problem while addressing its relevance to the topic of crew preferences. Methodology for solving the airline crew rostering problem will be addressed in Section 2.3.

2.1.2. Characteristics of the airline crew rostering problem

Kohl and Karisch (2004) have given a comprehensive description and definition of the crew rostering problem. The authors are affiliated with the Carmen Crew Rostering system which is used at major European airlines such as Alitalia, British Airways, KLM as well as the German state railway company Deutsche Bahn. Four main categories of information are presented that together provide the input to the crew rostering problem. These four categories are discussed below:

1. **Crew information** - For each crew member, a record is kept containing historical flights, qualifications, due dates of training sessions or medical checks, planned vacation days, preferred flights or holidays and pre-assigned activities such as simulator training sessions that are already scheduled.

2. **Activities** - Pairings, reserve duties, instruction flight legs, office duties, ground duties, reserve duties, unassigned training sessions or courses and unassigned medical checks are all examples of activities to be fed to the crew rostering problem.

3. **Objectives** - Typically, there are four types of objectives that are used in the crew rostering problem that are given below. Often, these objectives are combined such that multiple objectives are pursued. The

interpretation of Kohl and Karisch (2004) has been adopted to some extent, while also giving examples and interpretations not presented:

(a) *Objectives related to real costs* - Minimizing real costs is a major objective of the airline crew rostering problem. This objective can be pursued by minimizing open time. In airline crew rostering, open time is the term for unassigned activities. A solution to a crew rostering problem, a schedule for all crew members, can still be feasible with a high amount of open time in the schedule. For example, there are multiple solutions for dealing with a pairing in a schedule that is unassigned. Options are to ask crew to work overtime, to convert reserve duties into flight duties or simply to cancel a flight. This latter option is strictly undesired, so the objectives related to real costs are explored to prevent this from happening. Costs for not assigning an activity or for leaving gaps in rosters can usually be calculated. Examples of real cost drivers in a schedule are gaps that are not assignable (gaps without any activity), overtime payments, hiring extra freelance crew, deadheaded flights to relocate crew and paying out premium days or fees when no reserve crew is available.

(b) *Objectives related to individual preferences* - Maximizing crew satisfaction is relevant for an increasing number of airlines. Crew preferences are taken into account in the crew rostering problem, and preference endorsement is considered to be a means for sustaining or increasing crew satisfaction. Means to express preferences come in multiple forms which are addressed in Section 2.2.

(c) *Objectives related to equality* - Maximizing a global equality criterion is an important objective when considering individual crew members concerning the whole workforce. For instance, equal rostering can be measured for duty hours, days-off, early duty start times and layovers. Usually, these objectives are measured on a yearly basis rather than in each planning period. As an example, practical implementation can be achieved by penalizing deviations from average values. In airlines with a fixed monthly salary this objective is especially important. For example, a crew member might have operated more flight duty hours than the workforce average in the first quarter of a certain year. In the second quarter of that year, assigning long duties to that crew member could be penalized based on an equality principle.

(d) *Objectives related to solution attractiveness* - Minimizing unattractive solutions in a schedule is an objective for the airline crew rostering problem in which roster patterns that might cause problems are penalized. Kohl and Karisch (2004) consider this type of objective to be related to the robustness of the solution. However, it can be argued that this type of objective is a way to solve the problem within margins that have been obtained empirically and with experience. For example, a solution for a feasible roster might comprise a set of duties that is more demanding for crew to operate than other duties. Penalizing this type of activity sequencing, allows induced fatigue to not vary too much across duties.

4. **Rules and regulations** - Rules and regulations are a major topic in the crew rostering problem limiting the solution space and defining the conditions for a legal schedule. Rules are recorded by agreements between airlines and employee unions, legislation and governmental agencies or the airline itself. The rules and regulations applicable to the crew rostering problem overlap with the set of those applicable to the crew pairing problem. For the crew rostering problem, they are enforced on a more individual level, and the level of detail and complexity is, therefore, higher. For example, a crew member might be transferred to another fleet family division where other rules apply. This results in scheduling challenges in the fleet transition period for that crew member. Complexity for airline crew rostering rules with respect to airline crew pairing rules lies in the variety of individual cases. Typically, there are three types of rules and regulations that are handled by the crew rostering problem. Horizontal and vertical rules refer to the Gantt Chart representation of a roster as was shown in Figure 2.1 in which time is represented along the horizontal axis and lines of work for crew members are represented along the vertical axis.

(a) *Horizontal rules* - Apply to rosters, or single lines of work, and do not consider the rosters of other crew members. Examples are rules for qualifications, minimum rest time between tasks, maximum accumulated duty hours or avoiding pairings to conflicting time zones.

(b) *Vertical rules* - Apply to either a set of multiple rosters or the whole schedule. Examples are crew complement that needs to operate each flight, experience of crew complement, language qualification for each flight, training programs or global constraints on crew satisfaction.

(c) *Artificial rules* - Additional constraints outside of legislation and contractual agreements that are usually enforced by airlines to improve robustness or to omit unattractive solutions. These rules are also called quality rules. Examples are constraints to prohibit unacceptably long layovers or built in margins for flights that are more likely to be delayed.

When comparing the airline crew rostering problem to the airline crew pairing problem, the latter has received more attention in the literature. The most important reason for this is that most of the cost savings in operational productivity can be achieved in the crew pairing problem. As was stated before, the cost function of the crew pairing problem is quite well monetarily defined (Kohl and Karisch, 2004). Cost can be expressed through metrics such as workforce headcount, the amount of reserve crew needed, and productive flying time. The crew rostering problem, however, comes with a cost function that is less tangible. Where the crew pairing problem does not take individual crew members into account, the crew rostering problem does. It is in this phase of the airline schedule planning problem, that individual crew members are assigned to duties. These individual crew members come with all sorts of requirements and demands that need to be taken into account in the crew rostering problem. In line with this more individual approach lies the social value of the problem. A roster can be considered to be of higher (social) quality whenever the preferences of the crew member have been taken into account. An incentive to strive for this higher (social) quality is to increase or sustain satisfaction of the crew members. Individual satisfaction of crew members or satisfaction of the workforce as a whole is a concept that is challenging to capture using operation research techniques. Firstly, it is a challenge to capture the concept of crew satisfaction in a (financial) cost function. Secondly, to scale cost minimization objectives and crew satisfaction maximization objectives together, a trade-off must be made concerning a cost-effective solution and a high crew satisfaction as defined by the satisfaction performance metrics of consideration. For crew satisfaction especially, determining these metrics is a challenge. This illustrates that the airline crew rostering problem and management of crew satisfaction go hand-in-hand. This will further be characterized in Section 2.2.

2.1.3. Approaches to the airline crew rostering problem

Regarding the characteristics of the airline crew rostering problem, it can be stated that the problem can be approached in many different ways. Airlines and airline divisions vary in terms of strategies and objectives, and so does the airline rostering problem. The literature on the airline crew rostering problem reflects this and efforts have been made to develop multiple approaches to create rosters. Moreover, the consideration of objectives related to individual crew preferences varies across different rostering problems encountered in literature. In some crew rostering approaches, crew preferences are incorporated into the objective function of the problem whereas, in other approaches, crew preferences are considered after the rosters have been created. In terms of methodology, the multiple methods will be discussed more comprehensively in Section 2.3. There is no consensus on the definition of the approaches since they are typically modified with respect to different airline strategies and objectives. However, the differences between the approaches are relevant to scope research as crew preferences are handled differently in each approach. Three types of approaches to the airline crew rostering problem can be identified from the literature, which are presented and summarized below:

1. **Bidlines approach** - The bidlines approach is performed in two steps. In the first step, unassigned lines of work, called bidlines, are created. These bidlines are presented to the crew members who can then express preferences in the form of bidding on these lines of work. In the second step, these bidlines are assigned to individual crew members based on the relative preference that each crew member expresses for each of these bidlines (Kohl and Karisch (2004); Belobaba et al. (2015)). With this approach, a crew member will know exactly what he or she will get if the bid is granted, namely the complete lines of work that are assigned to that crew member.

2. **Personalized rostering approach** - With the personalized rostering approach, lines of work are created directly for individual crew members instead of first creating unassigned lines of work which are assigned later on. Rosters for each crew member are constructed while being able to conform to each crew member's individual requirements. This allows for attributes such as pre-assigned activities and preferences of each crew member to individually be taken into account. Some authors (Belobaba

et al. (2015); Medard and Sawhney (2007)) identify the rostering approaches as one single approach, simply referred to as rostering. However, two distinct approaches can be identified, which can also be used in a hybrid approach:

(a) *Personalized rostering approach with preferential bidding* - The preferential bidding approach allows crew to express preferences for certain attributes of their rosters (Kohl and Karisch (2004); Stojković et al. (2009)). These preferences are reflected by weighted bids and can vary widely. Examples of such bids are bids on duties to preferred geographic areas, bids on preferred pairing length and bids on preferred duty start times. With this approach, a bidding score can be evaluated for each roster as a measure of quality of the rosters for each individual crew member. Gamache (1998) presented a model for assigning pilots using the preferential bidding approach. In this model, a problem is solved for each crew member sequentially and on seniority order aiming to maximize the bidding score for each roster. All the remaining activities to be assigned to the remaining crew members are taken into account in each subsequential subproblem for each crew member. The problem is thus solved starting with the most senior crew member and ending with the most junior crew member. In response, Achour et al. (2007) presented an exact method for the preferential bidding approach where the assignment of a roster with maximum bidding score to a more senior crew would be delayed until there would only be one feasible roster left for that crew member. This approach accounted for a better bidding score for more junior employees.

(b) *Personalized rostering with pre-assigned activities* - In the personalized rostering approach with pre-assigned activities, preferences are expressed for specific activities (Stojković et al., 2009). A specific pairing or day-off can be requested by a crew member, and the request is typically handled on a strict seniority basis (Eltoukhy et al., 2016). Certain quality criteria can be considered such as rostering on a fair share basis or to take into account crew preferences that are considered during the roster creation (Kohl and Karisch, 2004). The set of pre-assigned activities can be considered as a set of boundary conditions to which the roster that is created should comply. Examples of pre-assigned activities are planned pairings with a training incentive, planned leaves of absence, or planned simulator training sessions (Kasirzadeh et al., 2017). This category has received relatively much attention from a modeling perspective.

The bidlines approach is adopted by many North American airlines whereas the personalized rostering approaches are adopted by most airlines in Europe and the rest of the world (Belobaba et al., 2015). According to Kohl and Karisch (2004), crew members benefit most from the bidlines approach as they will know exactly what roster they get. However, high costs for airlines can occur due to a high likelihood of conflicts with pre-assigned activities. The bidlines can thus often not be assigned to individuals in their entirety, leading to additional steps in the creation of the roster. The goal for all three methods is to minimize costs while considering roster quality criteria for both individual crew members and the workforce as a whole. To conclude, the models differ mainly in the way how the objective function is formulated and how they can be practically integrated into an airline rostering process.

A final way of approaching the airline crew rostering problem is not to consider it as a separate problem within the airline crew scheduling problem. This approach is to integrate it with the crew pairing problem and to formulate it as an integrated crew scheduling problem (Eltoukhy et al., 2016). One can distinguish between partial integration and full integration. In the case of partial integration, the problem is still solved sequentially by first creating pairings and then assigning them. However, the problem is formulated as a single problem. The model presented by Guo et al. (2006) first selects a set of crew pairings, with which the model provides an individual crew schedule. The major drawback of this approach is the quality of cost estimation and the quality of the solution concerning individual practicality. In the case of full integration, the crew pairing problem and crew rostering problem are solved simultaneously. The model presented by Zeghal and Minoux (2006) provides such a fully integrated solution, which is evaluated especially on computational metrics rather than on the solution quality from an airline crew rostering perspective. This limited solution quality on an individual crew member level currently is the major drawback of integrated approaches. This limits the approaches to be used within a practical crew rostering process. Concerning the investigation of crew preferences that are evaluated on an individual level, the non-integrated approaches are more applicable for research purposes.

2.2. Defining the research gap and objective

Crew preferences are handled in crew rostering phase of the airline crew scheduling problem. Approaches to handling crew preferences vary per airline but almost always, there is a process in place that incorporates these preferences into crew rostering (Kohl and Karisch, 2004). To address these different approaches to the topic of crew preferences and to emphasize the stand-alone topic, the term *crew preference management* is adopted in this thesis report.

Crew preference management
The strategy that an airline carries out to satisfy the objective of managing the satisfaction of individual crew members and the workforce as a whole about the rosters that constitute the solution to the crew rostering problem.

To further describe crew preference management, multiple characteristics of crew preferences have been identified in Section 2.2.1. Based on these characteristics, the relevance of crew preferences is identified and discussed in Section 2.2.2. A review of approaches to crew preference management within the airline industry and within scheduling problems in other industries is presented in Section 2.2.3 and Section 2.2.4, respectively. From this review, a resulting research gap is identified that will outline the research objective on the topic of airline crew preference management. This research gap and research objective are covered in Section 2.2.5 and Section 2.2.6, respectively.

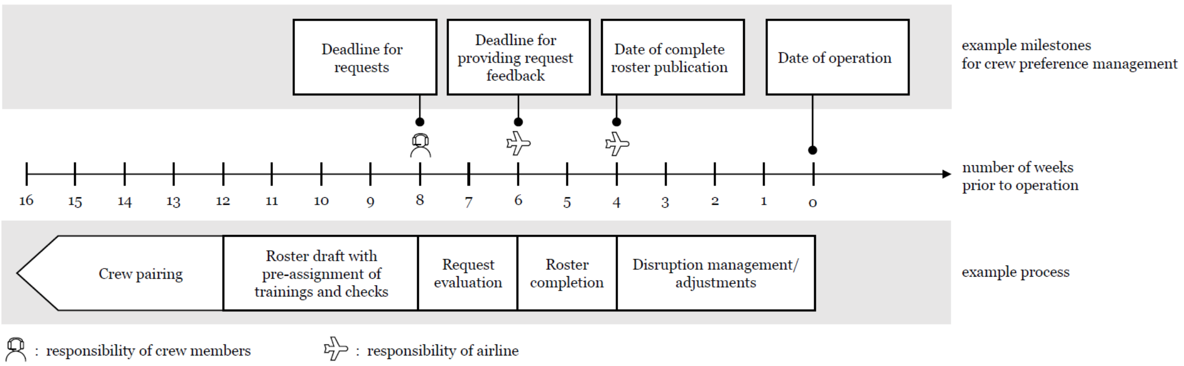


Figure 2.3: Example of a crew preference management request process in the airline crew scheduling process

2.2.1. Characteristics of crew preference management in airline crew rostering

When discussing preferences of airline crew, it is necessary to get a sense of what it is that crew members prefer in their rosters. It is, therefore, useful to elaborate on what the rosters are made up of. The building blocks of a roster are referred to as roster attributes (Caprara et al., 1998). Examples of roster attributes are duty start times, duty end times, pairing length, pairing destination, specific pairings with fixed start and end date, multiple day leave of absence, flying East- or Westbound. These attributes are not always mutually exclusive. A pairing, for example, can have the following attributes simultaneously; starting on a Monday, in the morning, with a length of six days. While one crew member might prefer operating flights on a Monday regardless of the other attributes, another crew member might prefer this pairing specifically. The roster attributes for which preferences can be expressed vary per airline. The different means of expression of crew preferences can be subdivided into the following two categories:

- **Bids** - Preferences for roster attributes where a relative weight is given concerning other bids. For example, points need to be distributed among the placed bids of a crew member to express the relative importance of each bid to that crew member.

- **Requests** - Preferences for very specific roster attributes such as specific pairings or specific days off where no relative weight is given concerning other requests.

Crew preference management not only varies across different airlines. A different strategy might be carried out for different airline divisions such as different fleet families or short-haul versus long-haul flights.

Whenever the crew rostering problems are solved separately within an airline, crew preference management can be approached differently. To get a sense of crew preference management within a scheduling process, an example approach is shown in Figure 2.3. In the figure, a request process is illustrated. The bottom bar represents the phases in the process plotted against decreasing time to operations. The top bar represents milestones in this process with an indication of which stakeholder (crew members or the airline) is responsible for this milestone.

Typically, the process for crew management is integrated into the crew rostering process and system of an airline. It is, therefore, challenging to isolate the stand-alone topic of crew preference management. From the available work on airline crew preference management, however, the following three main characteristics can be identified:

1. **Social nature** - The first characteristic of crew preference management is its social nature. Rosters of airline crew can be demanding, fatiguing and irregular. Moreover, the rosters are published only a few weeks before operation, making it challenging for the airline crew to arrange for their personal lives. Crew preferences are a means for airlines to acknowledge and address this social aspect of the crew rostering phase for individuals. However, the satisfaction of crew preferences can be accounted for only to a certain extent. Airlines need to identify and make a trade-off of satisfying the needs of the crew that are of this social nature. The trade-off between sustaining or increasing the level of satisfaction while simultaneously suppressing the cost impact on the crew rostering phase is a challenge that airlines face. The social nature of crew preference management, therefore, emphasizes the multiple objectives that compete in airline crew rostering at airlines. The definition of this trade-off is challenging to formalise. For example, simply maximizing the number of granted crew requests across the workforce does not necessarily capture the morale or other social dynamics of the workforce. Medard and Sawhney (2007) discuss that a human resource management perspective needs to be taken into account in the crew rostering problem.

2. **Negotiation nature** - The second characteristic of crew preference management is the negotiation nature of the topic. In line with the social nature, crew preference management influences individual rosters, and in addition to that the highly valued personal lives of crew members. It is, therefore, a recurring topic in negotiations between employee unions and airlines. Different aspects of a crew preference management strategy can be leveraged to accomplish negotiation objectives from both sides. However, the financial or social impact of making changes in a crew preference management strategy or from granting or rejecting certain preferences is often unclear which is underexposed in literature.

3. **Crew prioritization** - The third characteristic of crew preference management is the role of crew prioritization. A common way of crew prioritization is crew seniority. The preference of the most senior crew member will often be given more weight than that of the second senior crew member (Achour et al., 2007). In the case of strict seniority, infinitely more weight is given to preferences of more senior crew members (Hojati, 2010). The consequence of this (strict) seniority enforcement is twofold. Firstly, the aggregation of popular preferences takes place with crew members. For example, junior crew members might never be able to operate certain destinations due to some destinations being highly desired and always preferred by more senior crew members. Secondly, the enforcement of seniority restrictions is practically very hard to change since more senior crew members will uphold the position that they have reached a rank in seniority within their career stage from which they derive certain rights from. In some airline divisions, another way of crew prioritization is maintained based on a fair-share principle. A combination of both can also be employed where some roster attributes are divided based on a fair-share principle whereas others are divided based on seniority order. This latter option can also be done based on a rolling seniority principle, where a dynamic seniority rank is handled rather than a rank based on employment length only.

2.2.2. Relevance of crew preference management in airline crew rostering

A roster is sufficient for operation if all the pairings are timely covered by the crew members (Day and Ryan, 1997). Adopting this perspective, there is no direct interest for the airline in which flight is operated by which crew member, as long as all rules and regulations are met and productivity is guaranteed. However, when considering the main characteristics of crew preference management that were discussed above, its relevance is evident which changes the perspective. In the studies that have been addressed up to this point, different drivers and factors of crew preference management are presented. Examples from the presented work are optimizing for crew satisfaction, fairness across the workforce, equality of employees or balancing out unpopular rosters. The ways that companies around the world deal with employment relationships is reflected in their labour relations and human resource management. Given its service-intensive nature, this employment relationship is especially important in the airline industry (Barnhart et al., 2003a). The high level of union representation of the workforce, especially in large airlines, and the high ratio of labour cost concerning total costs are reasons why employees are in the position to affect an airline performance significantly. Collective bargaining is an important means to demand higher wages or mutations to rules and regulations. Moreover, high additional costs can be imposed by strikes or other service disruptions. On the other hand, the crew members play a critical role in the achievement of quality and productivity. To balance these dynamics, management of the workforce has played a central role in airline management. In this context, the relevance of crew preference management is different from a crew member perspective and an airline perspective.

From a crew member perspective, the benefits from granted preferences can be mainly expressed as the gratification of personal or social needs. With bids or requests, crew members can influence duty days, duty times, days off, vacation days and flight destinations. The incentives for expressing these preferences can vary widely, such as the examples that follow. A short-haul crew member could have a preference for an early morning duty so he or she can be home for family dinner. A crew member who has planned a weekend away with friends could prefer to have both the Saturday and Sunday off for a specific week. A crew member could prefer visiting the newly added attractive destination to the airline network without really minding when this will take place. With bids, these incentives can be measured to some extent by the indicated relative importance of preferences. With requests, these incentives are often unclear. Being informed about a roster as early as possible enables crew members to adopt their work schedule to their personal lives to a better extent. A monthly roster publication moment is common for most airlines which makes it possible that crew are informed about their rosters, and the feedback on their preferences, only a few weeks in advance (Gamache, 1998). Requests for multiple days yearly leaves can be evaluated on a larger planning horizon than that of the crew rostering problem.

From an airline perspective, preference management is a means to accommodate for and acknowledge the challenging work-life balance of their crew. Within the scheduling process, however, minimizing costs can be considered the most important objective. However, it is unclear what the financial impact of preference management on the costs and feasibility of a schedule is. It is therefore difficult to live up to a specific strategic goal or objective with respect to crew preferences and to define a maximum or acceptable degree to which preferences are satisfied. Having more insights on the financial aspects of preference management in the crew rostering phase could lead to a better ground for managing these preferences. Moreover, better estimates could be made to quantitatively measure the effect of rules and regulations with regards to preference management. This could lead to a stronger negotiation position of airlines when the crew preference management strategy is leveraged concerning its cost or impact. Furthermore, an airline can have an indirect advantage from being able to create rosters that make the workforce happier and healthier. High workload rosters can lead to an increase in absenteeism and a decrease in productivity (Hanne et al., 2009).

2.2.3. Approaches to crew preference management in airline crew rostering

As stated in Section 2.1, the crew rostering problem has received less attention than the crew pairing problem due to its lower financial impact on productivity. However, many approaches to the problem have been presented by the body of research over the years. This Section will discuss the literature on the approaches that have considered preferences as part of the problem objective. Since crew preference management is inherent to the crew rostering problem approach, the work is categorized and discussed per type of rostering approach in the following Sections. Crew preference management in the bidlines approach is discussed firstly, followed by the preferential bidding rostering approach and the pre-assigned activities rostering approach.

Crew preference management in a bidlines rostering approach

Figure 2.4 shows an example of a bidlines process within a scheduling process. The layout is similar to Figure 2.3 where the bottom bar represents the process phases and the top bar represents milestones in this process with an indication of the responsible stakeholder. What can be seen in the figure is that in this example, the airline is responsible for generating the lines of work or bidlines that are published to the crew, in this case, 8 weeks before operation of those bidlines. It can be argued that when using the bidlines approach, the quality of constructed bidlines is measured on the regularity of the working patterns. Crew preferences are considered in the next step after bidlines construction which is also clear from the figure. The concept of crew preference management is acknowledged by Jarrah and Diamond (1997) who perform a study on bidlines generation. However, more emphasis lies on the step before the actual crew preferences are expressed. In the creation of bidlines, namely, it is already considered that the set of generated bidlines should reflect an equality principle. This equality principle can be modeled through the cost function of a roster. Target values are defined for each crew member using historical data for each crew member. The targets are continually devised for each planning period because many targets are determined on a broader planning horizon than that of roster publication. The objective then is to minimize deviation to the target of each crew member when selecting a set of rosters that covers all the pairings. This is achieved via a heuristic that selects a set of feasible bidlines and checks whether all activities are assigned to one and only one bidline. Other than this work, there is no relevant work available on crew preference management when using a bidlines approach. Based on this, however, an opportunity for crew preference management might be to use the data of the crew bids to account for the bidlines generation process being able to match these preferences better. However, a difference in desirability for each bidline is inevitable in practice.

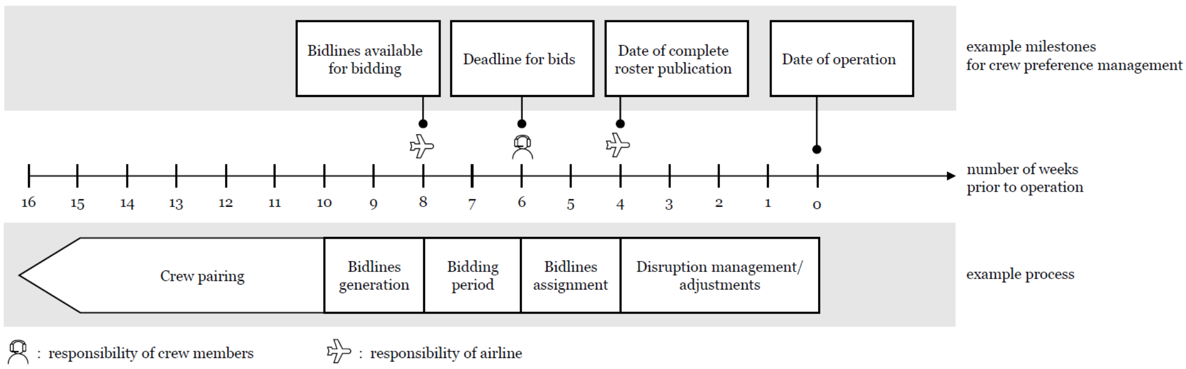


Figure 2.4: Example of a crew preference management bidline process in the airline crew scheduling process

Crew preference management in a preferential bidding rostering approach

Approaches to crew preference management within a preferential bidding rostering approach has been more widely addressed. It is notable that most approaches consider preferences from a different perspective than other authors and do not continue the work of others. The work of Gamache (1998) and Achour et al. (2007) are the only exception to this rule. Gamache (1998) presents a heuristic rostering model for the preferential bidding system at Air Canada. This work is recognized by the majority of the authors within the airline crew rostering domain. In the presented model, cockpit crew members are assigned to pairings while aiming to maximize the bid score of the individual crew members by solving a linear integer program for each crew member on seniority order from the most senior crew member to the most junior crew member. For each crew member, the sets of yet unassigned pairings and remaining crew members are determined, and a so-called residual problem is solved by column generation embedded in a branch-and-bound tree. The objective of the residual problems is formulated in Equation 2.1, where c_s represents the bidding score of roster s if $s \in \Omega_e^k$, where Ω_e^k is the set of residual schedules for crew member e that are available in iteration k and Y_s represents the binary variable that takes the value one if roster s is chosen for crew member e and zero otherwise. In the residual problems, the constraints that the problem is subject to are subdivided into local constraints to create feasible rosters and global constraints to create feasible overall schedules. The local constraints concern flight hours, maximum consecutive working days, minimum rest periods and a cumulative fatigue factor for the rosters of each crew member. Global constraints concern the set partitioning constraints which guarantee assignment of each pairing. Other global constraints concern a maximum of assigned rosters with limited flight hours and a maximum of open time schedules that cannot be assigned to a crew member schedule and are thus assigned to a fictitious crew member. In the experiments, Gamache (1998) presents two categories of problems; a large-scale and a small-scale monthly rostering problem. The problems are based on the crew divisions at Air Canada at the time. The large-scale problems deal with experiments ranging from size in 46 crew members and 267 pairings to 82 crew members and 602 pairings, for which the lowest and the highest computation time were found to be 64 seconds and 481 seconds, respectively. In comparison, the small-scale problems deal with experiments ranging from size in 18 crew members and 103 pairings to 48 crew members and 178 pairings, for which the lowest and the highest computation time were found to be 3 seconds and 13 seconds, respectively.

$$\text{Maximize} \sum_{s\in\Omega_k^k} c_s Y_s \tag{2.1}$$

As an addition to the previous work, Gamache et al. (2007) presents a graph-colouring model for a feasibility problem in monthly crew scheduling. The model of Gamache (1998) is used as a reference, and a limitation to this model is pointed out. This limitation is that it is possible that a roster is selected for a crew member where the more junior employees cannot cover the residual pairings. In this case, backtracking of the solution process is needed. A feasibility test is proposed that uses a graph colouring method together with a tabu search algorithm to avoid this backtracking. The work focuses on validation of the tabu search algorithm and the graph colouring model especially, which are stated to be suitable for integration within the model. Integration of the method with the model from Gamache (1998) is presented as a recommendation for future research rather than as a result of the work. The achievements of this work are relevant regarding the improvement of an established preferential bidding rostering model but do not contribute considerably to the field of crew preference management. Being affiliated with the same research group, Achour et al. (2007) presented an exact solution approach to solve the crew rostering problem for the first time. This model adds another method to the approach of Gamache (1998). In this model, a method for delaying the selection of a best-score schedule for a senior crew member is used. Best-score schedules are explicitly enumerated and the delay of assignment takes place until only one feasible schedule is left for a senior crew member that yields his or her best score together with the best scores for the subsequent crew members. The method relies on using a longest resource-constrained path algorithm to enumerate the best-score feasible schedules for each crew member. The quality of the solution is substantially improved with similar computing times. Moreover, this new approach often reduced the number of backtracks that were needed to solve for feasibility. In addition to the objective of Gamache (1998) in Equation 2.1, Achour et al. (2007) present two other ways to express the objective of a preferential bidding problem. The objective in Equation 2.2 is based on the work of Sherali et al. (2013) where a set of weights $\{\lambda_1, \lambda_2, \ldots, \lambda_m\}$ is used to reflect the priority of the m different objectives to each of the m crew members. The bidding scores of more senior crew members can be prioritized using this method. Here, Ω_i represents the set of feasible rosters for crew member i, c_{ij} represents the bidding score of roster j for crew member i and Y_s represents the binary variable that takes the value one if roster j is chosen for crew member i and zero otherwise. The drawback of this approach is that it is impractical in practice because the relative weights can become too large for a computer to process. For example, in the case of considering crew seniority linearly, the weight of the most senior crew member can be (arbitrarily) ten times higher than that of the second most senior crew member. However, for that crew member and his successor, the same rule could apply. In this example, this leads to weights going up to 10^m. The objective in Equation 2.3 represents an approach that preceded the approach of Gamache (1998). In this approach, the constraints ensure that an equal score schedule is assigned to the first $k-1$ crew members when constructing the roster for crew member k. The drawback of this approach is that an integer solution must be found for m large-scale generalized set partitioning problems, which again makes this an impractical problem to be applied in the industry. The experiments in the work of Achour et al. (2007) indicate that that preference scores increase with respect to the work of Gamache (1998) while the computation time increases for most of the experiments. This is due to the required backtracks to find a feasible solution.

$$\text{Maximize} \sum_{i=1}^{m} (\lambda_i \sum_{j\in\Omega_i} c_{ij} Y_{ij}) \tag{2.2}$$

$$\text{Maximize} \sum_{j\in\Omega_k} c_{kj} Y_{kj} \tag{2.3}$$

A crew preference approach related to preferential bidding that was presented before the well-acknowledged work of Gamache (1998) is that of Strevell and Chong (1985). Here, a quantitative method is presented to assign leave to crew members in a military flying unit where the objective was to maximize crew preferences. A point system was used to enable the crew to bid for preferred periods. What is notable about this approach is that it relied on the concept of an open market system. The method for assigning leave periods was straightforward; the highest bidder to a leave period was assigned to that period. A demand schedule for leave periods was published such that crew members could make their trade-offs between period desirability and (point) costs in a rostering phase in which they could react to other crew members' bids. The disadvantage to this system was that, due to being occupied during duty hours, not the whole workforce was able to outbid others and to participate in the full bidding period. To mitigate this drawback, the process was changed to a two-phase bidding period. In the first phase, the crew expressed their preferences for the periods they wanted to be assigned to. A demand schedule was published as an output from this first phase, and in the second phase, the crew was able to express their bids in a closed bid knowing what periods had the highest demand. A conclusion from this experiment in practice was that it proved to be very effective as leave requests notably shifted away from the summer and the Christmas period which resulted in a more stable schedule throughout the year. Although it seems like an attractive approach to crew preference management, the study is incomplete when it comes to presenting methodology and problem size. It appears to be practically executed while not offering a method to capture its novelty or repeatability. The work has not been referred to in the airline rostering domain, and the problem as described here has not been addressed since it has been published.

The topic of rostering days-off or in other words, leaves of absence, is addressed by Day and Ryan (1997) using another method. Day and Ryan (1997) present a two-stage method to allocate days off to flight attendant crew or cabin crew at Air New Zealand while taking into account crew preferences. The work, however, is not focused on crew preference management specifically but addresses multiple optimization methods that have been implemented at Air New Zealand. In the days-off rostering model, firstly, feasible and legal days-off lines of work are generated for each crew member over a two-week rostering period. Here, different patterns of 5 days off and 9 duty days are generated. Secondly, a days-off line of work is assigned to each crew member and the remaining 9 duty days of the 14 days rostering period can be used in rostering to assign work duties to. In the duty allocation phase, quality of lines of work is considered important of which the major measures are crew preferences and equitability. The aspect of equitability is not addressed in other works within the preferential bidding approach. From a solution point of view, the authors do not address it. However, enforcing equitability is a means to spread satisfaction among crew within a crew rank evenly. The cabin crew group had been involved in selecting appropriate weightings for each type of preference. It is acknowledged that the quality of a line of work is very difficult to quantify and that some quality measures compete for importance. Line-of-work costs are determined to comprise a weighed penalty corresponding to different quality measures. These penalties are derived from undesirable roster constructions. Examples are a cost penalty for assigning an early start time where a late start time was requested or a cost penalty for cumulative duty hours very close to the legal limit. The penalties are not always constant but can be expressed with a linear function relating to penalty severeness. Subsequently, the objective then was to minimize the costs of the lines-of-work for the entire schedule. A linear programming relaxation followed by application of branch and bound was used to obtain an integer solution. Solution times for rostering problems deal with experiments ranging from size in 13 crew members and 86 pairings to 80 crew members and 610 pairings, for which the lowest and the highest computation time were found to be 12 seconds and 258 seconds, respectively.

Coming from the same research group and addressing the work of Day and Ryan (1997), Butchers et al. (2001) appoint three measures of quality for rosters. These measures are the fair distribution of work, satisfying the work preferences of individual flight attendant crew members in seniority order and the avoidance of difficult work patterns. The main objective for the national (short-haul) rostering problem is to maximize crew satisfaction and lines-of-work robustness in which cost measures were decided upon in collaboration with crew members. The days-off patterns presented by Day and Ryan (1997) are taken into account here. Since both crew satisfaction and lines-of-work robustness are subjective measures, the system opts for a high-quality solution rather than an optimal solution. In the rostering of international flights rather than national flights, the airline considers all crew members within one rank as equal. The main objective here is to minimize the under-coverage of language requirements. Crew requests are captured in cost penalties. By keeping track of historical data, it is ensured that similar levels of satisfaction are upheld across the workforce and that duties to desirable and undesirable destinations are distributed equally. An important statement is that crew negotiated the set-up of the preferential bidding system as part of the crew contract. The results that are presented indicate that the models and optimization methods that were developed were tailored towards Air New Zealand, along with all the implementation and operational aspects. The results that are presented are focused on cost reductions for the overall crew scheduling optimization efforts. It is stated that, due to the efforts in optimization models, the number of staff needed to solve the crew-scheduling problem at the airline decreased from 27 in 1987 to 15 in 2000. Specific results on crew preference management, however, are not presented.

Crew preference management in a pre-assigned activities rostering approach

In approaches to crew preference management within a pre-assigned activities rostering approach, the elements of the objective function of the problem receive more attention. Maenhout and Vanhoucke (2010) present a model where three objectives are considered; minimizing costs, ensuring impartiality and fairness to all regular crew members, and enabling crew members to express preferences for both specific duties and more general roster attributes. The authors wanted to maximize the quality of the schedule regarding these objectives and used a hybrid scatter search approach that aimed to assign rosters to crew members while deciding if extra personnel or freelance personnel was needed. Decision variables in the objective function indicate whether the crew that is assigned to an activity is of the type regular, extra or freelance. The crew preferences are incorporated into the objective function. A crew aversion score is calculated as an inverse to the crew preference score for a specific roster, and this is considered one of the penalty cost factors in the objective function. This function is relatively complex with respect to objective functions presented in other works. Crew preferences were considered alongside pre-assigned activities, crew information and regulations. The procedure was tested and validated at Brussels Airlines, where one month of preference data was available. Another artificial data set was used in which preferences were generated randomly. When a solution to the problem is found, local search algorithms account for improving the solution to increase the level of granted crew preferences in the solution. Tests were performed with 50 generated problem instances of 50 crew members and 700 activities (600 pairings, 50 reserve duties and 50 pre-assigned activities). Distinguishing between these activities is interesting for future research as the ratio between the multiple types of activities can be adapted. When using the algorithm under investigation in this work, the hybrid scatter search algorithm, the solution process proved to be of similar computation times than with the variable neighbourhood search algorithm that was already in use (127 seconds and 115 seconds, respectively). In 81% of the cases, the algorithm was able to satisfy all crew member's preferences. Concerning solution quality as defined in this work, the hybrid scatter search algorithm outperforms the variable neighbourhood search algorithm with a deviation of 3.2%, which could indicate a benefit of considering evolutionary algorithms in a rostering model.

A model that lies more emphasis on personalized rostering and the concept of crew preferences is presented by Kasirzadeh et al. (2017). The model is presented for the personalized cockpit crew rostering problem. In the model, a subproblem is solved for each crew member in which a personalized roster is constructed. Two types of preferences are considered, for specific flights and vacations, and the problem is solved in the context of short- and medium-haul flights. Monthly schedules can be constructed with a sequential approach based on branch-and-price. Given a set of cockpit crew preferences and yet unassigned anonymous pairings, the objective of the problem is to minimize the total crew costs while satisfying a minimum number of preferences. Hence, costs and preferences are not considered in two separate steps but are rather combined into a single objective function. What is notable from this approach is that the set of preferred flights and preferred vacations are explicitly listed in the model formulation. The objectives of cost minimization and preference maximization are captured in a single objective function while using penalty costs and bonus costs to weight out the effects. The cost function of a personalized roster s for pilot l is presented in Equation 2.4. Here, the $e_p^{s,l}$ represents the binary variable that takes the value one if pairing $p \in P$ is covered by pilot l in personalized roster s and value zero otherwise and C_p represents the cost of pairing $p \in P$. This cost factor is defined in the work of Saddoune et al. (2012), where pairing cost is expressed as a function of waiting time before, in between and after the flight duties, assuming a production based salary for crew members. Subsequently, n_s^l represents the number of preferred flights in roster s for pilot l and c_f^l represents the bonus cost for covering preferred flight f in the set of preferred flights for pilot l. Finally, $v_v^{s,l}$ represents the binary variable that takes the value one if vacation $v \in V_l$ for pilot l is covered by personalized roster s and value zero otherwise and c_v^l represents the penalty cost for not covering preferred vacation $v \in V_l$. The objective function for the model from Kasirzadeh et al. (2017) is given by Equation 2.5. Here, x_l^s represent the binary decision variables of the model that take the value one if schedule s is chosen for pilot l and the value zero otherwise. $\overline{e_p}$ takes the value one if flight f is not covered by the solution and $\overline{C_f}$ represents the cost for not covering that flight. Except for the cost parameter C_p, it is unclear how the cost parameters are determined and if these are based on real costs or airline data. Summarized, this model enables setting a constraint for a minimum required preferred activities. An additional advantage to this model in the context of investigating the effect of preferences is that pre-assigned activities can be used as an input to the model to set a certain degree of flexibility that the model has to work with. The computation time for problem instances ranging from 33 crew members and 172 pairings to 305 crew members and 1648 pairings are 150 seconds and 17535 seconds, respectively.

$$C_s^l = \sum_{p \in P} e_p^{s,l} C_p + n_s^l \cdot c_f^l + \sum_{v \in V_l} (1 - v_v^{s,l}) \cdot c_v^l \tag{2.4}$$

$$\text{Maximize} \sum_{l \in L} \sum_{s \in S_l} C_s^l x_l^s + \sum_{f \in P} \overline{e_p} \overline{C_f} \tag{2.5}$$

Considering crew preference management as a means to account for fairness in the rosters, Doi et al. (2017) presents a two-level decomposition based matheuristic algorithm for solving an airline crew rostering problem. To account for fairness in the rostering problem, penalties are introduced that penalize rosters that deviate from a roster with average working time. Three types of fairness measures in the form of penalties are presented; linear penalty, quadratic penalty and min-max penalty. To improve for equality across the workforce, the linear penalty approach proved most effective in improving fairness across the workforce as measured by the deviation from average working time. A difference between railway crew and airline crew is addressed by the authors, which is that in railway crew rostering, cyclic rosters can be created more easily as opposed to airline crew where irregular activities are often tailored in the rosters such as demand constraints for all crew members and seniority constraints.

2.2.4. Approaches to crew preference management in other scheduling problems

Practices on crew preferences in crew scheduling problems from domains other than the airline industry have been consulted as well. This Section presents the state-of-the-art of crew preferences in a crew rostering context. The railway industry and the healthcare industry are industries in which the scheduling domain is widely studied. Still, crew preference management practices are relatively under-covered. Section 2.2.4 covers the railway industry, Section 2.2.4 covers the healthcare industry and Section 2.2.4 covers other industries.

Crew preference management in the railway industry

The railway industry is similar to the airline industry in many aspects as both have an operational transportation characteristic where a fleet of trains or aircraft needs to be scheduled to serve a certain objective. Scheduling topics that are relevant to the railway industry are the planning of railway infrastructure, maintenance scheduling, and timetable planning (Turner et al., 2016). Crew scheduling is relevant in the last planning phase.

A study that acknowledges crew preferences as an important objective in railway crew rostering is that of Hanne et al. (2009). An optimization-based decision support tool is presented that is used in railway crew rostering. The railway specific rostering problem is addressed along with constraint treatment, the formulation of different objective functions and the consideration of crew preferences. It is argued that it becomes increasingly important to consider employee preferences in scheduling the duties that they operate and that it has become important for the crew to be able to have flexible work schedules according to their wishes. These flexible work times allow employees to get hold of their work-life balance better. For a railway company, two advantages of flexible work schedules are presented. Firstly, skilled staff can be retained, and recruitment costs are reduced. Secondly, another crucial advantage is that the morale of the workforce is increased which is often positively correlated with a reduction in absenteeism. In this study, this assumption is made based on National Australian statistics from a general study on flexible working hours. It is argued that scheduling flexible rosters as opposed to regular cyclical rosters requires a computer-based support system. Concerning the proposed solution approach in this study, crew preferences can either be formulated as objectives that need to be optimized for or as soft constraints that can be violated. What is notable from this study is that it considers preferences as a concept that is not only applied to the crew but also to the railway company. Preferences for the railway company are aspects such as the favorability for specific pre-assigned activities or crew qualifications. In airlines, this is usually formulated as an extra set of requirements rather than as preferences. Moreover, personnel costs might not be fixed when external drivers need to be hired to cover all the duties. The transition from work time to vacation time might also lead to additional employee costs. The preferences of the railway company are expressed in the cost function of the optimization problem. Preferences can also be considered as a way to reduce the search space for the solution. The Equations 2.6 to 2.10 indicate the way in which preferences are handled in this example of the railway industry. In Equation 2.6, the costs of each employee roster are expressed similarly as was done in Equation 2.1 for an airline crew rostering problem. Equation 2.7 represents the degree to which preferences are violated, where the term d_j represents a deviation of actual duty start times with regards to preferred duty start times. Equation 2.8 represents holiday incongruity which is measured by the term h_j, with which deviation of actual holiday start and end dates with regards to preferred holiday start and end dates are expressed. Equation 2.9 represents a penalty for violation of constraints that are built up with w_k being the penalty weight and v_{ik} being a binary penalty violation indicator of hard constraints. Finally, Equation 2.10 represents the objective of the assignment problem. The problem is formulated as a multi-objective problem:

$$C = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{2.6}$$

$$D = \sum_{j=1}^{n} d_j \tag{2.7}$$

$$H = -\sum_{j=1}^{n} h_j \tag{2.8}$$

$$V = \sum_{i}\sum_{k} w_k \cdot v_{ik} \tag{2.9}$$

$$\text{Minimize } f(x) = \begin{pmatrix} C \\ D \\ H \\ V \end{pmatrix} \tag{2.10}$$

For simplification reasons, the study leaves the V objective out of the equation, which makes the multiple objectives easy to visualize as a Pareto front across the three axes of the three remaining objectives; minimizing costs (C), minimizing preference violation (D), minimizing holiday incongruity (H). The Pareto optimization methodology is adopted here to discuss the multi-objective results of the optimization problem. It is notable, however, that the study focuses on presenting the feasibility of the decision support system resulting from the model. No experiments are presented that indicate the problem size or the results of the scheduling model. Therefore, crew preference management can only be reviewed from a modeling perspective which is achieved with preference violation penalties.

A study that focused on a fairness or equality principle rather than individual crew preferences was presented by Jütte et al. (2017). A study was performed that focused on rostering fairness preferences at a European railway freight carrier. Integration of fairness considerations into a crew scheduling optimization approach is performed, and the effects of fairness constraints and unpopularity constraints on the cost of a crew schedule are evaluated. Unpopularity is measured with a score based on two unpopular properties; shifts in which old trains are operated or shifts in which unpopular areas are visited. The formulation that is used is based on the previous work of the author, and a solution is generated based on column generation including both fairness and unpopularity constraints while keeping the formulation linear. Costs that are associated with a schedule are workforce costs and hotel costs when drivers are not at their base for an overnight stay. Often, when solving for minimal costs, the schedule contains duties that are unpopular to the drivers. With regards to fairness, these duties are often distributed unevenly across the workforce, and this is perceived by those who find themselves at a disadvantage. The study integrates fairness conditions into an optimization model, and it analyses the effect of increased fairness on cost. Significant improvement in schedule fairness with only marginal increases in schedule cost was established. The perceived unfairness had led to a decrease in job satisfaction, and it could potentially lead to a decrease in job performance, bickering and increased absenteeism which is an assumption that is based on the study of Bard and Purnomo (2004). These factors are very hard to quantify and cannot be considered dependent on fairness alone. Negative factors from perceived unfairness can be an increase in employee turnover (Smet et al. (2014)) and a trigger for labour strikes (Abbink et al. (2005)). Again, a triad of objectives was formulated to minimize crew schedule costs, minimize the number of unpopular duties, and to establish an even distribution of unpopular duties among depots. Experiments were executed that focused on unpopularity, fairness and a combined strategy. The results from the experiments in a crew base with 17 depots show that a more even distribution of both unpopularity and fairness across the depots was achieved with a cost increase of 1%. It is unclear from the study, whether this cost increase represents real costs or the optimization problem solution. However, it is notable that taking into consideration quality measures such as fairness and popularity of roster attributes comes at a cost.

Crew preference management in the healthcare industry

A so-called preference scheduling approach for nurse scheduling at hospitals that operate around the clock was presented by Bard and Purnomo (2004). The contrast with manufacturing environments is highlighted that operate in standard shifts with standard days off. Fair management of preferences must be guaranteed while ensuring sufficient coverage of the shifts. This is achieved in a multi-objective problem using a column generation approach combining linear integer programming and heuristics. The degree to which the individual preferences of a nurse are violated to determine the objective coefficients. Individual preferences are accommodated by the nurse scheduling process and are measured regarding requests for specific shifts or specific days off. To this preference scheduling problem, a range of exact methods, heuristics, and constraint satisfaction have been applied. Exact methods primarily involve the use of set covering-type models where alternative schedules for each nurse are generated, and one is selected. Minimizing a weighted combination of costs and preference penalties is a common objective in the problem. A quantitative measure includes an exact monetary cost for a certain roster attribute such as an agreement to pay overtime per hour. A qualitative measure includes preference violation that could lead to low morale for the employees. No reference is given to the correlation between crew preference management and the low morale of employees, but the study assumes this. A method to quantify this qualitative measure is to adopt the cost coefficient according to the severity of the preference violation. An exponential function of penalty costs is proven to work best in the model. What is noted by the authors is that the computation time exponentially grows with the number of nurses and days in the planning horizon. The largest experiment was performed with 68 nurses, which required a total of 7347 columns. This averages to just over 108 columns per nurse before converging. The smallest experiment was performed with 20 nurses, which converged with the addition of only 20 new columns or 1 column per nurse on average. A suggestion for future research is only to generate those columns that are considered attractive rosters for the nurses. This might be a desirable situation. However, it is yet unclear how attractiveness is quantified and whether a feasible solution can be found with only columns that are considered attractive.

Another example that has been used in the healthcare industry is a scheduling problem in a home care company by Rasmussen et al. (2012). This problem is an example of a situation where preferences are used to narrow down the solution space. A method is presented to cluster visiting schemes for a home care crew scheduling problem based on visiting preferences. By identifying clusters with preferred shifts first, run times of the set partitioning problem with a branch-and-price solution algorithm decreased significantly. The solution space is narrowed down when focusing on preferences first before continuing to schedule the rest of the shifts. It is stated that loss of quality of the solution was found in a few problem instances only.

Crew preference management in other industries

Crew preference management is addressed in other industries as well. Firstly, work from a scheduling perspective in different domains is presented. These domains are the retail and services industry, scheduling of telephone operators and a more general approach to uncertainty in personnel scheduling. Secondly, when considering work on crew preference management in industries other than the scheduling domains that have been widely studied, another domain should be addressed. Many of the studies that have been presented in the airline scheduling domain and other domains so far claim that crew preference management is correlated with low morale of the workforce which can lead to an increase in absenteeism. This claim is substantiated in this Section by considering crew as human resources and by reviewing the efforts of studies that consider crew preferences from a social perspective.

A trend within the retail and services industry is captured by Mohan (2008) where it is addressed that the size of the part-time workforce is constantly increasing in this industry. With this development, the demand for flexibility in availability and shift preferences has increased. An integer program is proposed to maximize employee satisfaction while meeting demand requirements for each shift. Employee satisfaction is defined as a function of seniority, preferences and availability as indicated in Equation 2.11. Here, the parameter w_i indicates the seniority index of employee i, the parameter p_{ijk} indicates the preference of employee i for shift j on day k in a range $[0,10]$, and the parameter a_{ijk} indicates the (binary) availability for employee i for shift j on day k. The term $(a_{ijk}-1)w_i$ represents a penalty for the situation where an unavailable employee is assigned to shift. In an experiment, the preference of each employee for each shift is randomly chosen, and a branch-and-bound enumeration procedure with additional cuts is used to reduce computing time successfully.

$$\text{Maximize} \sum_i \sum_j \sum_k Y_{ijk}[w_i p_{ijk} a_{ijk}) + (a_{ijk}-1)w_i] \tag{2.11}$$

A preferential bidding system at a telephone company with weekly shift assignment of telephone operators is presented by Hojati (2010). Employees indicate the preference coefficients relevant in this study for earlier and later starts of their shifts. The objective function of the scheduling problem is to minimize the dissatisfaction of employees. This objective function is presented in Equation 2.12 where p_e represents the priority of an employee which takes a lower value whenever the employee e is of higher seniority. Furthermore, D_e represents the set of available working days of employee e, S_{ed} represents the set of eligible shifts for employee e on available day d with $d \in D_e$, dis_{es_d} represents the dissatisfaction of employee e to perform shift s_d and x_{es_d} is the decision variable that takes the value one if shift s_d is assigned to employee e and zero otherwise. The dissatisfaction parameter is defined as a function of a difference in the start time of shift s_d and the desired start time of employee e. Such a difference is straightforward to quantify with respect to shift start time or end time. However, with requests for specific shifts or pairings in the case of airline crew scheduling, it is unclear and mostly ambiguous how a difference can be quantified. A set of 100 experiments are performed with a set of 73 employees and randomly generated preferences over which the computation time is 5 minutes on average, and the dissatisfaction coefficient is 1.45 hours on average. This study indicates that a quantitative metric such as time can be used to express a qualitative metric such as (dis)satisfaction. It can be argued, however, that this metric does not necessarily capture (dis)satisfaction of crew.

$$\text{Minimize} \sum_{e} p_e \left(\sum_{d \in D_e} \sum_{s_d \in S_{ed}} dis_{es_d} x_{es_d} \right) \tag{2.12}$$

A study that addresses the stochastic nature of the crew rostering problem is that of Ingels and Maenhout (2017). Here, the problem is addressed that there is a lack of incorporating uncertainty in personnel scheduling models. A distinction is made between two planning phases; the creation of a medium-term baseline roster where assumptions and predictions about service demand and personnel capacity are made followed by the creation of a short-term roster where variability arises. The variability that arises in the short phase might not be represented in the assumptions and predictions for the creation of the medium-term baseline roster. The study aims to improve the quality of the baseline rosters by focusing on the stability of these rosters. The problem is addressed by presenting a deterministic formulation of a stochastic shift scheduling problem in which capacity buffers are matched to stochastic demand. A trade-off needs to be made between the desired level of stability (the degree to which uncertainty is accounted for) and the corresponding additional costs. The objective of the model formulation is presented in Equation 2.13 which consists of two parts that are summed. The first part represents the minimization of wage cost and preference penalty cost; the second part represents the minimization of understaffing the shifts. In the function, c^{w}_{idj} represents the wage cost of assigning employee i to shift j on day d, p_{idj} represents the preference penalty cost if employee i receives a shift j on day d, x^{w}_{idj} represents the binary decision variable for employee i to receive shift assignment j on day d, c^{wu}_{dj} represents the cost of understaffing employee i to shift j on day d and x^{wu}_{dj} represents the number of employees shortage on day d for shift j. The stochastic element of the problem is captured in the constraints of the problem. Equation 2.14 presents these stochastic staffing requirement constraints, in which x^{w}_{idj} and x^{wu}_{dj} are summed and compared to the value $\widetilde{R^{w}_{dj}}$ which represents the uncertainty in the required number of employees.

$$\text{Minimize} \sum_{i \in N} \sum_{d \in D} \sum_{j \in S} (c^{w}_{idj} + p_{idj}) x^{w}_{idj} + \sum_{d \in D} \sum_{j \in S} (c^{wu}_{idj} x^{wu}_{dj}) \tag{2.13}$$

$$\sum_{i \in N} x^{w}_{idj} + x^{wu}_{id} \geq \widetilde{R^{w}_{dj}} \quad , \quad \forall d \in D, \forall j \in S \tag{2.14}$$

A difference is addressed between the planned cost and the actual cost. The planned cost comprises shortages in shifts and total assignment cost (wages and preference penalty costs). The actual cost includes shortages in shifts, total assignment cost (wages, preference penalty costs and cancelled duties) and the number of changes in the duties of the employees. The wage cost and the cost of understaffing are both predetermined and non-variable in the model. The preference penalty costs are randomly generated within an interval of $[1,5]$ and are made possible for shift-on (desire to work), and shift-off (the desire not to work) requests. In the experiments, different levels of variability in available employees are determined based on the basic probability of absenteeism are distinguished, i.e. $P(X=0) = 2.44\%$ and $P(X=0) = 10\%$. Within these many variabilities of employees, different levels of operational flexibility are distinguished, based on three metrics, i.e. the number of cancellations, the number of swaps in employee shifts, reassignments of an employee with a day off to a shift duty. Smaller actual costs are achieved when using this method, and the method proves especially impactful with low operational flexibility. It can be argued that this is an expected finding. However, the difference in computational time is unclear from the study. As it is stated that the quality of the baseline roster using this method, the methodology used in this study can be explored further to be applied to the airline crew rostering problem.

2.2.5. Research gap

It can be stated that crew preference management is a widely recognized concept in literature to address the social nature of the crew rostering problem. However, the majority of authors focus on solving the rostering problem itself, while taking into account the crew preference management strategy that is handled by the airline with which research is affiliated. This crew preference management strategy is then formulated in the objective problem function, and cost parameters are determined, assumingly based on data from the airline cooperating to the research. What is lacking in these problems, is a critical view on the effect of crew preference management on the solution. What would the solution be if the crew preferences were omitted and what is the sensitivity of the solution to these crew preferences? The opportunities within crew preference management are possibly limited by the fact that the crew preference procedure of an airline is typically fixed and rigid and only altered with small adaptations resulting from negotiations between airline and employee unions (Kohl and Karisch, 2004). The financial effects of crew preference management are, therefore, unclear. It should also be noted that only one author (Maenhout and Vanhoucke, 2010) used one month of airline crew preference data as validation. Other authors used random preference generators to generate crew preference. Moreover, crew preference management strategies that are presented in the literature consider the crew as a collective. The entire set of crew preferences is considered an input to the problem objective, but there are no examples of approaches that handle individual crew preferences. From a social perspective, receiving direct feedback on an expressed request is assumed to be desired by the crew as this allows for clearer expectations and more certainty about the future roster. The approaches presented in the literature do currently not allow for this individual approach. An example of a crew preference management process that captures feedback on crew preference in an early stage of the scheduling problem is presented in Figure 2.5. The current issue of providing early feedback, however, is the uncertainty of how the roster will evolve.

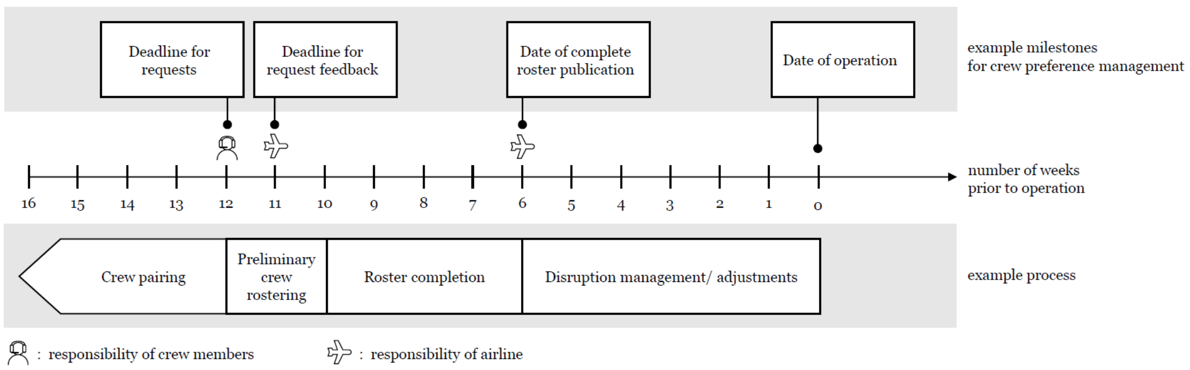


Figure 2.5: Example of a crew preference management request process with early feedback on preferences in the airline crew scheduling process

In addition to that, another missing factor can be identified when considering the rostering problem in which crew preference management is embedded. In the crew rostering problems that take into account crew preference management, the solution process is considered a static problem. In the literature, the process is considered as follows: the activities or roster attributes that can be bid on or requested by crew are published by the airline, crew express their preferences, the input to the crew rostering problem is finalized, the airline crew scheduling department solves the crew rostering problem, and the schedule is published. In airline scheduling practices, the rostering process can be considered as a process that is not deterministic. There exists variability in the demand of crew and the capacity of the crew. Incorporating this variability is essential for disruption management, but in the crew rostering domain, this is underexposed. This variability impacts the practical relevance of the deterministic mathematical models. The schedule under consideration during the rostering phase and the input to the crew rostering problem are, in turn, all but deterministic. There are two main reasons for this. Firstly, there is an overlap of a scheduling period with a preceding and a successive scheduling period. Pairings that commence in the already published scheduling period carry into the scheduling period under consideration. Therefore, disruptions or mutations taking place in the operation of the scheduling period that was considered published can propagate into the scheduling period under consideration. Secondly, all kinds of mutations can take place in the rostering phase other than those that deal with overlap from other scheduling periods. On a weekly or even on a daily basis, the input to the problem can change. Examples of causes that require a schedule to change are activities that are cancelled such as simulator training sessions, swaps of activities among crew or crew that falls out due to longer-term illness.

The rostering problem for long-haul crew divisions usually comes with more rigidness concerning the solution process than that of short-haul crew divisions. The reason for this is that the activity length is longer for long-haul crew divisions. A pairing in a short-haul roster can be as short as one day whereas a pairing in a long-haul crew division can take up to more than one week. This makes that the set of activities that need to be assigned in the long-haul rostering problem is less flexible and more likely to leave open time (unassigned activities) in the rosters. In order to make such a solution with open time operable, these nonassignable activities need to be covered by overtime, extra crew or reserve crew. This should be avoided considering the cost minimization objective of the crew rostering problem. Since it is expected that long-haul rather than short-haul flight rostering problems are affected most by crew preferences, a research scope could be designed to address the long-haul rostering problem first.

When recalling the relevance of crew preference management, the relevance for crew members is mainly from a social nature, while the relevance for airlines is mainly from a human resource perspective to acknowledge this social nature of crew preferences for the crew. However, it is clear from the literature that the cost minimization objective in the airline crew rostering problem is dominant. To capture financial effects of airline crew preferences management is, therefore, a means to improve the crew preference management strategy according to the needs of the airline and to leverage this information in contractual negotiations with the crew. To conclude the above, the following research gaps have been identified from the literature study on crew preference management in an airline crew rostering context. These research gaps are the topics to focus research on:

- **To use a dynamic approach to the airline crew rostering problem** - The crew rostering problem can be approached as a dynamic problem. On a periodic basis, the crew rostering problem can be solved, while the problem input and boundary conditions change over time. Stochastic modeling can be used within simulations that resemble crew rostering processes.

- **To leverage historical airline crew preference data** - Data on crew preferences has been a missing asset in studies on crew preferences. Insights can be obtained from investigating airline crew preference data to select appropriate preference modeling methods and parameters.

- **To capture the (financial) effect of crew preference management on the crew rostering problem** - The sensitivity of (financial) rostering objectives to crew preference management can be explored.

- **To develop a method to evaluate individual preferences** - New insights on the management of crew preferences can be obtained by modeling the evaluation of crew preferences on an individual level. A decision process for evaluating individual preferences can be developed that can be integrated into a crew rostering process.

2.2.6. Research objective and questions

The decision was made to focus research on pairing requests as a type of crew preferences. An important driver of this decision is the applicability of pairing requests to a dynamic rostering environment. Corresponding to the research gap, a research objective can be formulated that indicates the purpose of the research to be carried out. The research objective can, in turn, be captured in a research question. Answering this question will address the research objective and will fill the research gap as described in the previous Section. To make this research question more tangible, subquestions have been formulated. The remaining part of this literature study presents relevant work based on these subquestions that together are sufficient for addressing the research objective. The research objective, question and subquestions that have been identified are as follows:

Research question
How could pairing requests in the airline crew rostering problem be evaluated?

Research objective
To make recommendations on methods to evaluate pairing requests while capturing the dynamic nature of the crew rostering problem.

Research subquestions

1. How could the airline crew rostering problem be modeled to capture the dynamic nature of the problem?
2. How could crew preference management in a crew rostering simulation environment be used to identify and measure the (financial) effect of crew preference management on the crew rostering problem?
3. How could historical crew preference data be leveraged to identify and define parameters for modeling pairing request evaluation?
4. How could the evaluation of pairing requests be modeled and integrated into a crew rostering process?

2.3. Methods for creating and evaluating airline crew rosters with crew preferences

In order to address the research objective that was presented in Section 2.2.5, it is important to get a comprehensive overview of the methods at hand to address this research objective. It should be noted that the focus of Section 2.1 and 2.2 was on problem approaches of the studies that were discussed. Moreover, the studies that were discussed in that part of the literature study did not reflect the (airline) crew rostering domain as it focused only on studies that took up crew preference management as research. In this Section, however, studies from multiple domains are presented while the emphasis is on the presented methods to come to a solution to the problem at hand. The methodology is presented in such a way that it relates to the research subquestions that were presented in Section 2.2.6. Methods for creating and evaluating airline crew rosters are presented in Section 2.3.1, methods for a dynamic approach to crew rostering are presented in Section 2.3.3 and finally, methods for decision mechanisms for evaluating crew preferences are presented in Section 2.3.4.

2.3.1. Methods for creating and evaluating airline crew rosters

As was already clear from the problem approaches to the airline crew rostering problem in Section 2.1.3, a comprehensive literature review for state of the art regarding airline schedule planning is given by Eltoukhy et al. (2016). More specifically for airline crew rostering, Kohl and Karisch (2004) have made an effort to give an overview of methods used for the airline crew rostering problem. In both studies, a deliberate decision is made to not classify approaches according to their solution methodology or formulation method but rather on their problem characteristics (i.e. bidlines approach versus rostering approach). In this thesis report, however, the decision is made to review the literature based on the solution methodology since the solution method for formulating the model for both the airline crew rostering approaches is rather similar. The distinction lies in the solution methods so, therefore, this Section discusses multiple types of methods for the creation and evaluation of airline crew rosters. To illustrate a typical solution process for the airline crew rostering problem, Figure 2.6 shows the building blocks for this crew rostering solution process. In the solution procedure, a distinction is made between construction methods of rosters and improvement methods of rosters. After which a subproblem is selected (e.g. the roster of a specific crew member), feasible rosters for that crew member are generated after which the optimization problem at hand improves the solution of the roster of that crew member and the overall schedule. After which a certain stop criterion such as a threshold in problem size is reached, the solution process is stopped. The rule evaluator is a solution step in which constraints to the problem are relaxed and then enforced again. After a construction method is chosen, an initial solution or schedule is determined first after which improvement methods can increase the quality of this solution. Concerning the improvement methods, many that have been presented in the literature are heuristics which will be covered later on in this Section.

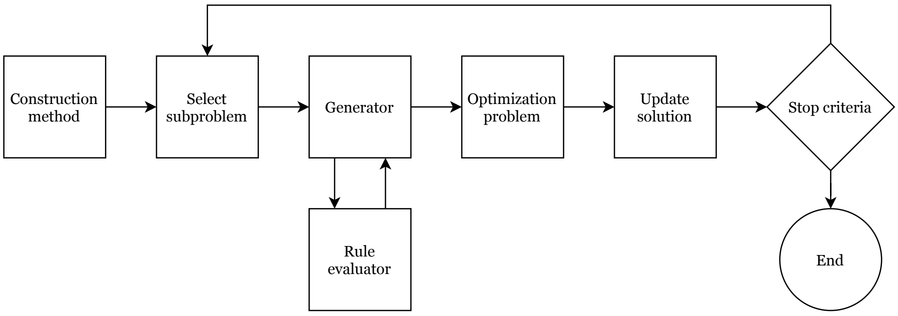


Figure 2.6: Typical solution process for airline crew rostering (based on Kohl and Karisch (2004))

When considering the construction method, a widely used method is to represent the rostering problem in a time-space network graph, since rostering problems deal with timed activities that have to be represented in both time and space. To illustrate this, Figure 2.7 shows an example of such a graph representation which is partly based on the work and representation of Kasirzadeh et al. (2017). In the figure, the source and sink node represent the start of the schedule and the end of the schedule, and the pairings are defined by a pairing start node and pairing end node. The midnight nodes represent the midnight times of the planning horizon. To construct a roster for the full planning horizon, a path can be found between the source node and the sink node via the arcs that connect these nodes. These arcs thus define the possible parts of the path (i.e. the solution of the roster) and are defined in the legend of the figure. Arcs that should be highlighted are the leave of absence preference arc, and the pairing preference arc as these are explicitly modeled in the work of Kasirzadeh et al. (2017). This is a modeling approach that could be adapted in the research following from this literature study.

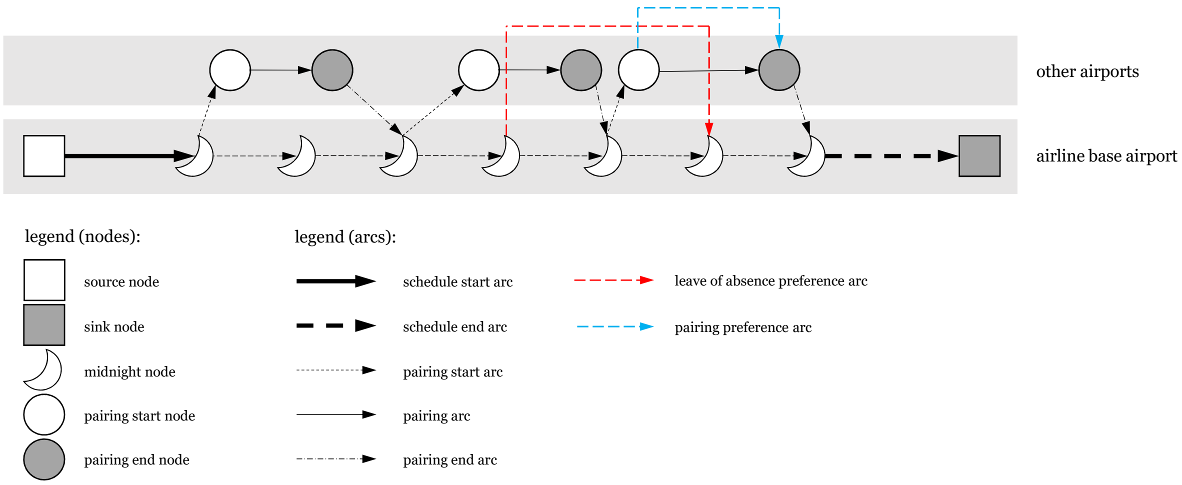


Figure 2.7: Example of a time-space network graph representation for the construction of roster with nodes that indicate instances of time and space and arcs that connect these instances (based on Kasirzadeh et al. (2017))

Methods for modeling multiple objectives in airline crew rostering

When formulating a crew rostering problem, a common way to do so is by modeling it as a 0-1 integer program. In the integer program, the problem can be formulated as a set-partitioning problem where, for rostering, the partitioning represents the distribution of activities among the crew. Adding a constraint that requires all crew members to be assigned to one roster turns such a problem into a set-covering problem. Cappanera and Gallo (2004) present a way to formulate the rostering problem as a 0-1 multicommodity flow problem. The commodities that have to be distributed over the network represent the employees. However, the body of research does not seem to have adopted this approach, so the set-covering problem is leading.

To be able to optimize the problem and come to a solution, an objective function for the rostering problem should be formulated. As was already clear from Section 2.2, the objective of the rostering problem is to optimize for multiple aspects. This Section considers studies that have addressed this from a solution perspective. When it comes to modeling objectives across the workforce, Butchers et al. (2001) indicate that rostering problems can be very different to reflect capabilities and business rules. The study shows that there is a distinction between four separate rostering problems; national pilots, national flight attendants, international pilots, international flight attendants. The rostering models use different quality measures, and the preferential bidding system is used for the international pilot divisions only. The rest of the divisions are assigned according to equitability assignment.

A relatively simple way to address the multiple aspects of the problem objective is presented by Lučić and Teodorović (2007), who address the multi-objective nature with a function that combines three objective values to one objective in a scalarization as shown in Equation 2.15. Here, the weights w_1, w_2 and w_3 indicate the relative importance of the objective values f_1, f_2 and f_3 respectively.

$$F = w_1 f_1 + w_2 1 f_2 + w_3 f_3 \tag{2.15}$$

Extensive work on multi-objective programming and goal programming by Tanino et al. (2003) refers to the above method as the *weighted sums* method and where it is stated that this method is a popular multi-objective approximation for combining two objectives. When considering the concept of optimality, Tanino et al. (2003) cover different types of optimality. Pareto optimality is a form of optimality that is used to describe the front of multiple objective values where it is no longer possible to achieve a better objective value. This does not mean that the achieved solution is the only possible solution to the problem or that an absolute optimum has been reached but at least a Pareto optimal solution has been reached. The solutions in the Pareto front are non-dominant to each other, meaning that achieving a gain in one objective always requires a certain amount of sacrifice in another objective (Konak et al., 2006). This form of optimality is often associated with resource allocation. With lexicographic optimality, a specific sequence in the solution approach has to be followed. In the example of the airline crew rostering problem, a clear example is the enforcement of crew seniority. An optimal solution is found meeting this lexicographic requirement while this could have been entirely different for the non-lexicographic problem.

Other work on scalarization as a means to model multiple objectives into one function comes from Ehrgott (2006) who describes multi-objective integer programming (MOIP) problems based on scalarization which was shown in Equation 2.15. Drawbacks of this method are presented with the lack of applicability in real-world problems being an important one. The method of applying Lagrangian duality to this general scalarization method does not remedy the drawback. Therefore, the method of elastic constraints combines good features of constraints with a weighted sums method to avoid the drawbacks of scalarization.

A way to bring more variety into the employee schedules of railway crew as a quality objective was presented by Abbink et al. (2005). Insights on crew scheduling optimization at the Dutch national passenger railway operator which dealt with severe strikes of employees. An alternative set of scheduling rules was proposed called *sharing-sweet-and-sour* which opted to schedule more variable shifts for the crew while also minimizing costs. The changes were unpopular with traffic management, which had to deal with a much more complex system in case of major disruptions. Experiments were performed with a set of 14678 train trips for which a solution of 982 drivers was found after about 2 hours of computing which converged to 971 drivers after about 4 hours of computing, which then in the last stage converged to; 965 drivers in 14:47 hours of computing. This means that an improvement of only 6% has been made in close to 70% of the total computing time. The railway operator experienced significant changes regarding punctuality of trains and employee costs were reduced by $4.8 million or 1.2% annually. It is stated these saving come from operational optimization and an increased motivation both. It is unclear, however, how this difference is made.

Another way of expressing roster quality is to define a service level for a roster. Castillo et al. (2009) present a solution approach for workforce scheduling in general where cost minimization and service level maximization are considered simultaneously, and an example is given for a call centre environment. The goal of the study is to open a broader workforce paradigm where service quality is taken into the analysis, and the possibility is given to study the interaction between cost and service quality. In the proposed approach, quantifiable criteria such as average waiting time, service level, queue length and personnel utilization are used and averages as well as minimum and maximum values to these criteria. Employee satisfaction, however, is missing from this service quality list.

2.3.2. Heuristics in airline crew rostering

After an initial schedule is constructed, an improvement method can be selected to improve the solution iteratively. Since it is often not possible to find an optimal solution for large-scale rostering problems, heuristic procedures should be leveraged (Lučić and Teodorović, 1999). This Section discusses multiple types of heuristics that appear in the crew rostering domain.

Branching heuristics

Branching heuristics are a common type of heuristic in the crew rostering problem. A problem that was already addressed in Section 2.2, is the flight attendant day-off rostering problem of Day and Ryan (1997), where a method is proposed for constructing rosters for short-haul domestic flight attendants that satisfy employment contract regulations. Recall that days off patterns were allocated to crew in a first subproblem and that duties were scheduled in a second subproblem. For the first subproblem, a generalized set partitioning model with the branch-and-bound procedure is used to solve for a feasible days-off roster maximized for desirability and preferences. Both generation and optimization of a division of 80 crew members with each up to 400 possible lines of work took about 80 seconds to compute. For the second subproblem, a set partitioning model with the branch-and-bound procedure is used. The use of subrosters of a period of the n first days of the 14 days roster period reduces the combinatorial size of the problem. With a solution for the n first days, feasibility is checked for the remaining roster problem, and in case of infeasibility, a step back is taken to the solution for the $n-1$ first days. Using this method, it has been found that a set of feasible rosters can be constructed with m (number of crew members) equal to n (number of rosters). Feasibility of the full schedule is most important in this phase and computation times vary widely for each combination of crew members and duties. For a 14 days roster, computation times varied from 13 crew members, 86 duties and 12.2 seconds to solve for the smallest problem instance to 80 crew members, 610 duties and 257.9 seconds to solve for the largest problem instance.

Another study where the branch-and-bound tree has been implemented is that of Gamache (1998). For the preferential bidding problem of Gamache (1998) that was already presented, recall that days off, annual leaves and training periods are assigned while considering weighted bids of employees for the activities. A schedule is constructed for each employee sequentially from the most senior to the most junior employee. With the remaining set of employees and activities left for each of these employees in a residual problem, an optimal integer solution is obtained. An external branch-and-bound tree is used to links these residual problems and to allow for backtracking to ensure feasibility of the complete schedule. The GENCOL optimizer that uses a branch-and-bound algorithm is used to solve the linear relaxation of the generalized set partitioning problem, and the dual variables are used to price out new rosters. The rosters are generated by solving the subproblems for each employee that are formulated as constrained longest path problems. A novelty to this work was the use of cutting planes in the column generation process at the subproblem level. This is continued in another publication of Gamache (1999) where a similar solution approach is adopted with adaptations to achieve a faster solution process. Here, the objective of rostering cabin crew at Air France is to minimize the duration of uncovered pairings. The cabin crew are qualified to operate different types of aircraft, meaning that there always is more crew available for the demanded workforce. By effectively controlling column generation, the solution procedure could be as much as three times lower than was the case when using standard column generation.

An extension to the branch-and-bound technique in rostering problems was presented by Dawid et al. (2001). It is made possible to downgrade crew to a lower rank in order to find a feasible solution. The algorithm that is introduced is the SWIFTROSTER algorithm in which enumeration takes place and elements of constraint programming are used for the different divisions in the airline involving over 1300 crew members and 6 fleet types. Computation times for a one-month scheduling problem ranged from 1.22 seconds for 25 crew members and 280 duties, to 605.04 seconds for 779 crew members and 1711 duties.

A related branching technique was presented by Hoffman and Padberg (1993). Here, a branch-and-cut approach to the airline crew scheduling problem as one of the firsts. The approach was tested on many large-scale real-world problems of both set partitioning problems and set partitioning problems with side constraints. However, no specific results on the performance of this approach in the crew rostering problem are presented. Another related branching technique, branch-and-price, was presented by Boubaker et al. (2010) who propose a set partitioning problem for bidline scheduling with equity in which a pilot is selected for each pairing given a set of available pilots and pairings. The bidline cost corresponds to the contribution of that bidline to variances of credited flying hours and days-off, thus penalizing highly deviating equity. A standard branch-and-price heuristic was used to solve the problem, and in the linear relaxation, a rounding procedure was used to simply round the relaxed solution to an integer solution. Dynamic constraint aggregation was then used to fasten up the solution of the restricted master problem as this accounted for a large proportion of the total computing time. With this approach, restricted master problems are replaced by aggregated restricted master problems of reduced size which are more likely to produce less fractional results. In computational results, it is made clear that that for large-scale test instances, better quality solutions are produced by in dynamic constraint aggregation heuristic in only a fraction of the time that was required by the standard branch-and-price heuristic. For the biggest problem instance with 564 crew members and 2924 pairings, the solution time of the dynamic constraint aggregation heuristic was 3076 seconds which is only just over 3% compared to that of the branch-and-price heuristic of 95215 seconds.

Metaheuristics based on simulated annealing

A good improvement method that can be used for combinatorial optimization problems is that of metaheuristics. Concerning computational effort, metaheuristics have the advantage that it is not guaranteed that global optimum solution needs to be found (Lučić and Teodorović, 2007). This makes it applicable to problems such as the airline crew rostering problem, where a cost-effective solution is desired, but an optimal solution not necessarily because of problem size or simply because it can be argued that there is not necessarily value in the optimality of a roster which will later be mutated or disrupted. For problems where a large set of feasible solutions is possible, metaheuristics prove to be effective, and it is therefore recognized in the literature as a go-to method. The first metaheuristic under consideration is the one that is based on simulated annealing. Simulated annealing finds its analogy in the process of cooling metal in order to minimize energy levels in the crystal structure and its random search for a global minimum state. One of its characteristics is that it avoids being trapped in local minima by accepting occasional solutions, with a predefined probability, which worsens the objective function. In the algorithm, this is realized with two loops. In the outer loop, a temperature parameter is altered that influences the probability with which a worsening objective value is accepted. In the inner loop, roster perturbations are generated to re-allocate the duties that were assigned in the initial solution. The objective values of the altered solutions and the probability for accepting these solutions are evaluated, and when a solution is accepted, this is considered the current best solution. The algorithm is stopped whenever the temperature parameter has reached a predefined minimum.

Simulated annealing is also applied by Campbell et al. (1997), who built a bidline generator for FedEx. This generator produces a complete set of operable and legal lines of work for an airline's fleet while identifying the remaining unscheduled duties. According to this work, the general goal of generating bidlines is twofold. Firstly, to minimize total cost overall bidlines or simply to minimize the number of bidlines. Secondly, to maximize the line value and line purity. The latter objective was formulated as a discount factor on the cost objective which makes the cost of a perfect line equal to zero. This objective formulation was tested against other candidate formulations and proved to represent the best results. To formulate the problem, a 0-1 integer program was tried at first, but it failed due to the sheer number of possible solutions. Therefore, metaheuristics were relied on. Simulated annealing is chosen as a heuristic for solving this problem, and a local optimum is prevented because occasionally, the heuristic accepts change in the solution that worsens it. An initial solution is generated, and from there, changes are evaluated based on their modified costs and the schedule is updated. Other heuristics such as tabu-search and neural networks have not been tested but could have been used as well. The simulated annealing approach ran into some problems, and too many unassigned duties were left in the (non-optimal) solution. Therefore, in a second pass, valid lines were created by distributing the remaining unassigned duties using greedy heuristics. Run times varied with fleet size. Although the fleet size is not specified, the problems of the smaller fleets needed 20 to 30 minutes to run, and the problems of the larger fleets could take up to 6 to 10 hours. The computer equipment has been specified as well, which is not consequently done throughout studies.

Another heuristic procedure based on simulated annealing is followed by Lučić and Teodorović (1999) for enabling crew members to have approximate equal workloads. This equal workload can be considered as an equal number of weekend days away from the home base, an equal number of very early departures, an equal number of duty hours, etcetera. Firstly, an initial feasible solution is generated. Secondly, the simulated annealing technique is used to improve the solution obtained in the previous step. As a solution, the computation times are stated to be satisfactory. However, the actual computing time results are not presented. The model parameters of the simulated annealing method are tested against multiple problem dimensions, but there is no explicit conclusion regarding these tests. In a later study Lučić and Teodorović

(2007), when multiple metaheuristics were compared, the simulated annealing procedure proved to be the best performer regarding solution quality for a numerical problem example of a small to a medium sized airline with 53 pilots and 422 pairings during a 30 day period. The computing time, however, was highest for solving the problem with the simulating annealing heuristic. This was 20 minutes against 12 minutes and 4 minutes for a genetic algorithm and a tabu search method, respectively. It can be argued that solution times of this order of magnitude can be justified as the crew rostering problem is not necessarily a problem that needs to be solved real time when considering a practical scheduling process.

Metaheuristics based on genetic algorithms

Genetic algorithms are another commonly used type of improvement method in crew rostering. Genetic algorithms find their analogy in the concepts of natural selection and survival of the fittest of Darwin's theory of evolution (Lučić and Teodorović, 2007). A set of initial solutions to the problem are defined, and from this set, the best or *fittest* solutions concerning their objective value are selected as the *initial* population. The remaining solutions are discarded, and the selected solutions undergo a process of reproduction, crossover and mutation. The probability of a solution being selected for reproduction is typically proportional to the fraction of its objective value with respect to the sum of objective values of all solutions in consideration. This proportional selection is also referred to as roulette wheel selection. In the selection for reproduction, two solutions are randomly selected as *parents* and a cut-off point is determined which cuts the solutions into two. In the crossover phase, these cut-off parts of the parents are exchanged which creates two *off-spring* solutions. In the mutation phase, slight changes to the solutions can be made with a minimal probability, with the purpose of preventing irretrievable loss of possible solution outcomes. An example to get to a solution is to define a maximum number of generations in the created family tree and to then select the best-discovered solution up to that point.

A two-phase genetic algorithm is implemented by Christou et al. (1999), who also adopt the concept of line value and line purity. The problem at hand is a large scale bidline generation problem at Delta Air Lines. The first phase of this bidline approach is related to line purity which is a measurement for the quality of a line. As many high-quality lines as possible are generated in this first phase. The second phase of this bidline approach is to construct high-total-value valid lines from remaining open trips, and a genetic algorithm is used in the problem formulation. Strategies such as the roulette wheel mechanism and the aging mechanism are used to enhance the offspring quality in the genetic algorithm and thus to avoid convergence to undesirable points. The maximum computing time for the problem with 230 crew members in 58 minutes. When comparing these generated bidlines with the semi-automatic generation used before, both are of similar quality making the fully automatic approach more cost-effective.

A study that emphasizes that genetic algorithms used for problems with multiple objectives should be tailored accordingly is presented by Konak et al. (2006). Genetic algorithms that are specifically developed for problems with multiple objectives are proposed which differ from traditional genetic algorithms due to the introduction of fitness functions and the promotion of solution diversity. Genetic algorithms are stated to be primary tools in multi-objective optimization problems where consideration of trade-offs is essential. An important notion from this study is that using this method, it is often not needed to map the full Pareto front but to identify Pareto optimal solutions across the range of interest to the user only.

This multi-objective approach is also a reason for Lee et al. (2007) to use a genetic algorithm to improve the robustness of a flight schedule by re-timing its departure times. A practical outcome of the study is that upon termination of the solution process, a set of improved flight schedules is given to a user that comprises the Pareto front; the trade-off surface of the solution. This allows airline schedulers to determine the trade-off of each objective with respect to the other ones. It can be argued that for airline crew rostering, such a solution is desirable when it comes to decision-making processes.

Another problem where genetic algorithms are used is presented in the study of Souai and Teghem (2009), who aims to solve the airline crew pairing and crew rostering problems simultaneously with a solution approach based on a genetic algorithm. The heuristic-based initialization is compared with linear programming based initialization of which the latter performed better. Multi-point crossover is compared with simplified one point crossover of which the multi-point crossover performed better.

In the bus driver scheduling problem with multiple objectives, the metaheuristic genetic algorithms are compared to tabu search algorithms by Lourenço et al. (2001). The concerned objectives are the cost and the service quality of the schedule. The solutions that were found with linear programming found an optimum, but it is stated that these solutions were often not accepted by management due to their impracticalities. The solutions that were found after implementing the multiobjective metaheuristics were accepted much better and were easy use. The objectives were also much easier to use resulting in a competitive advantage in negotiations with unions. For the smallest test problem with 209 employees and 19990 duties, only the genetic algorithms had been tested. The computation time of the genetic algorithms varied from the LP-based solution (for which an optimal solution could be found). This deviation was 70% lower for genetic algorithms and cost were 15% higher for genetic algorithms. For the largest test problem with 348 employees and 74019 duties, the computation time of the genetic algorithms and tabu search procedure, varied from the LP-based solution. This deviation was 80% lower for genetic algorithms and 103% higher for tabu search algorithms. Costs were only 0.4% higher for genetic algorithms and 22% higher for tabu search algorithms. This indicates that for genetic algorithms significantly outperform tabu search algorithms for a large problem instance. This can be considered in research when selecting an appropriate improvement method. When considering the multiple objectives of the rostering problem, it is these kinds of trade-offs between cost, roster quality and computation time that need to be taken into account to select the solution strategy.

Metaheuristics based on tabu search

Tabu search algorithms find their analogy in the prohibition of certain directions in which a local search procedure can direct to (Lučić and Teodorović, 2007). This can be explained more practically with the rostering example. The initial solution is determined by the algorithm described above. Then, a list is created of all the possible pairs of crew members (for instance, crew member 1 and crew member 2). A swap of the solution is made for all pairs, and the objective value after the swap is evaluated. The swaps that are defined as tabu for that iteration are removed from the list of crew pairs. The remaining solutions represent the neighbourhood of the current solution consisting of all the possible other solutions that can be reached from this solution. A frequency-based memory of the solution process penalizes swaps that were made in the recent past. The number of iterations that a particular swap is tabu (i.e. taboo or forbidden) is determined when initializing the algorithm and is fixed. The best solution from the neighbouring solutions is selected, and the algorithm continues until, for example, a specific objective value or a maximum number of iterations is reached. Tabu search algorithms were compared to LP based solution methods and in the previous Section.

A hyperheuristic approach to solving several distinct timetabling or rostering problems is chosen by Burke et al. (2003). Using a hyperheuristic approach, a decision has to be made about which heuristic will be used to solve an optimization problem at hand. Based on reinforcement learning rules, heuristics compete with each other. A tabu list of heuristics is maintained in order to exclude some heuristics from competing at certain times during the search. The goal of this work was to be able to present a solution approach competitive with the state-of-the-art while maintaining a higher level of generality of the problem. This similar level of competitiveness was achieved for both a nurse rostering problem and a university course timetabling problem.

It can be stated that work on tabu search algorithms seems to have been adopted much less in a rostering context than that of genetic algorithms. Genetic algorithms are more widely acknowledged to be suitable for multi-objective problems which are typical for rostering problems (Lučić and Teodorović, 2007). The work of Lourenço et al. (2001) underlines this based on the results presented in the previous Section.

Other heuristics

Apart from heuristics that have been more commonly presented in literature in a crew rostering context, other heuristics have been presented in this context as well. What is notable from the studies at hand, is that the majority of these studies have been published from 2010 onwards, which indicates the interest of the research community of continuing to explore other (types of) heuristics in the crew rostering context.

To both improve and combine solutions to the rostering problem, Maenhout and Vanhoucke (2010) presents different meta-heuristics. A proposed scatter search algorithm is compared with a variable neighbourhood search solution approach and a branch-and-price solution approach in which the efficiency of the proposed algorithm is shown. The variable neighbourhood search approach is outperformed by 3.20% in solution quality, and the proposed algorithm deviates only 1.30% for the solution quality of the branch-and-price solution approach that has average runtimes of over 20 hours as opposed to just over 2 minutes for the proposed algorithm.

Exploiting the neighbourhood structure in the column generation procedure of an airline integrated crew scheduling problem is presented by Saddoune et al. (2011), proposing to use a heuristic based on the constraint aggregation method By aggregating clusters of set partitioning constraints, dynamic constraint aggregation allows to reduce the number of constraints in the column generation master problem. The computation time of seven test cases is reduced by 230%. Two variants of the proposed solution method led to cost savings of 4.02% and 4.76%. The computation time, however, increased with a factor of 3.0 and 3.8, respectively. As a follow up on this work, Saddoune et al. (2012) propose another integrated model for the crew pairing and crew assignment model which continues to compare computational experiments based on the seven test instances from Saddoune et al. (2011), in this instance using dynamic constraint aggregation. With the approach, the average total cost savings of 3.37% can be achieved with 5.54% fewer pilots in the test cases. As a drawback, the computational times were as high as 6.8 times higher on average.

Three other heuristics based on behaviour found in nature are the ant colony optimization technique, particle swarm optimization technique and the artificial bee colony algorithm. These are all covered in separate studies. For the first of these heuristics, Deng and Lin (2011) formulated the airline crew scheduling problem as a travelling salesman problem. To improve the solution, ant colony optimization is the technique adopted in this study which is inspired by the foraging behaviour of ants and the deposition of pheromones they use to exchange information. In the practical example of scheduling, this relates to keeping track of promising routes on the path graph and to update this so-called pheromone information on this graph. In terms of computational time, the ant colony optimization outperformed a genetic algorithm approach that was taken as a reference by 18% on average across all the 23 problem instances. A recommendation for further research involved the incorporation of habits and requirements of crew members to be able to increase the satisfaction of crew members. Azadeh et al. (2013) make use of another metaheuristic called a particle swarm optimization algorithm for the airline crew scheduling problem which owns its analogy to a flock of flying birds that form a swarm. The principle of the algorithm is that the particles (birds) move through multi-dimensional space under defined flight dynamics while communicating about historically found solutions. This algorithm is hybridized with local search heuristics, and the results are evaluated based on randomly generated problems. The hybridized particle swarm optimization technique is leveraged against genetic algorithm and an ant colony optimization technique and, based on the results, it proves to be very effective in terms of computing time and quality which is outperformed for both small and large problem instances. For the third of these nature-based heuristics, an artificial bee colony algorithm is hybridized with a hill climbing optimizer is used by Awadallah et al. (2015) for a nurse rostering problem. The approach is tested in 69 problem instances against eleven other methods in which best-published results were reached in 35 out of the 69 problem instances.

2.3.3. Methods for a dynamic approach to crew rostering

Modeling of dynamic properties in the airline scheduling domain is especially established in the daily operations of airlines, which is the phase after the publication of the schedule. In this phase of the scheduling problem, the purpose is to evaluate plans and to apply recovery policies in a random environment. As many sources of disruption in operations can occur that affect the schedule, the term disruption management is common in literature, and a stochastic modeling approach can be adopted. Rosenberger et al. (2002) present a simulation implementation of a stochastic model used to model airline operations. Disruptions are defined here as events that prohibit airlines from operating their schedules. Recovery is defined here as the way in which an airline reacts to disruptions. It is assumed by the authors that by building a better understanding of airline operations and how plans and recovery policies affect them, airlines could improve their on-time performance, reduce operational cost, and increase customer satisfaction. It can be tested though, whether the disruption approach and related modeling approaches could be adopted in the crew rostering domain. The stochastic model that is presented is a discrete event semi-Markov process which is described in terms of states and transitions (either random or deterministic). The schedule that was published is considered the initial state of the schedule. Information that the multiple states in the model hold are the deviations from the initial state, historical information needed for performance measures (such as possible breaching of rules and regulations) and conditions beyond control such as weather and air traffic control. Transitions between these states are achieved through either events that occur (e.g. departures, arrivals, maintenance events, weather events or airport congestion events) or by decisions made by the operational schedules. This latter transition is relevant for research in crew preference management, as decisions for crew preference management also affect the state of a roster. Examples of other decisions in the operational phase are to cancel flights, to swap aircraft or to call reserve crew to fly pairings. The model presents a method to capture the stochastic nature of these events and performs simulations that simulate the operational process of an airline. A limitation of the model is that the events are modeled independently. Events of weather and airport congestion, however, are sets of event that are likely to be correlated. Within airline crew rostering, this assumption can be challenged as well. The work and the method, however, are well acknowledged by the body of research and forms a promising basis for the implementation of a stochastic modeling approach in airline crew rostering. The authors encourage the model to serve as an assistant in developing airline planning and scheduling models.

This difference, from a modeling point of view, between the operational phase and the scheduling phase is made clear throughout literature. In a study on airline crew recovery, Lettovský et al. (2000) highlight that crew recovery substantially differs from crew scheduling due to the dynamic environment and the requirement to provide solutions with limited impact to the original schedules. However, the assumption can be challenged that the environment of crew scheduling is not dynamic. To that extent, multiple authors acknowledge the limitations of a crew scheduling phase when it comes to its deterministic nature. In a study on airline crew scheduling under uncertainty, Schaefer et al. (2005) address the limitations of assumptions in airline scheduling problems that all pairings will be operated as planned. Yen and Birge (2006) recognize this and addresses that information on potential disruptions is not explicitly included in the crew scheduling problem. However, this is not completely the case. As seen earlier in this literature study, margins can be built in to account for disruptions to occur in a later stage. In the studied that handled such margins, however, none are stated to be modeled stochastically. Guo et al. (2006) address the operational problem of multiple home bases. It is argued that the process of crew scheduling and assignment is very dynamic in practice, with constant changes in crew's availability and changes in pre-assigned activities such as simulator duties, approved requested duties and office duties. All this data should thus be taken into account in pairing generation for which a method is proposed. In addition, Smet et al. (2014) address the fact that it is common in the academic body that schedule periods are isolated when modeling them which does not conform to real-world requirements. Many of the problem constraints relate to following or previous schedules and to continuity in general. This part is often not taken into account in the evaluation metrics of a roster.

2.3.4. Methods for decision mechanisms for evaluating crew preferences

In terms of crew preference management, multiple ways to formalize this into a crew rostering problem have been presented in Section 2.2. However, a method to make decisions on individual crew preferences rather than considering a set of crew preferences is not presented in the literature. A review of possible decision mechanisms and methodology that can contribute to the development of a decision mechanism for evaluating individual crew preference is presented in this Section. Within the airline crew rostering domain, Biskup and Simons (2004) acknowledges this and states that very few models consider learning effects in scheduling. The author distinguishes between autonomous learning (or learning-by-doing) and induced learning. Because of rapid technological progress in terms of computing, the importance of considering learning effects as a competitive advantage in mathematical models is self-evident. The authors use learning effects in the determination of due dates in a scheduling environment. An example of induced learning is presented by Suraweera et al. (2013), who proposes a method that is able to infer constraints from historical crew schedules based on a set of user provided template outlining the general structure of important constraints. Complex multivariate constraints can be induced by the algorithm. To improve the method, it is proposed that more noisy schedules are used as an input to improve the algorithm's performance. Such methods could be tested for elements of crew preference management as well. Relevant constraints of a crew preference management strategy that can be induced which could lead to a better understanding of the sensitivity of the solution to such a strategy. In terms of individual preferences, methods that collect and process information about such preferences on both a schedule level and an individual roster level should be leveraged.

In the nurse scheduling domain, Asta et al. (2016) present a hyperheuristics technique with learning capabilities and self-improvement through a machine learning technique called tensor analysis. This technique enables remembering relevant changes throughout the solution process and is able to present that information throughout the search process. A forgetting effect prevents that too much information is stored and used as a reference.

In a study on the operational airline crew scheduling problem, Stojković et al. (2009) argue that a very relevant asset to research would be the day-to-day decision data of schedulers that make decisions on the scheduling of disturbances. It is then possible to compare results obtained by airline operators with automated decision models. This statement underlines the research gaps of leveraging crew preference management data within a crew preference management model.

When it is assumed that this data on crew preference management is available, or can be generated from simulation models, an interesting methodology for decision making on an individual level is that of supervised machine learning. The goal of supervised machine learning is to reason from externally supplied class labels in a data set what the class label of a future instance might be. To translate that to crew preference management, the hypothesis can be tested that individual crew preferences can be evaluated using supervised machine learning. supervised learning can be divided into the categories; classification and regression. Since an individual crew request decision is a yes-or-no decision, classification is expected to suit the problem specifics more accurately. A typical solution process in supervised learning algorithms is presented in a review on supervised learning techniques by Kotsiantis (2007) and is presented in Figure 2.8. After identifying the problem, a data set should be collected. Within this data set, a suggestion should be made on the attributes that are the most informative of that data set. This could be achieved by having an expert on the data reviewing these features or by brute force and simply measuring everything available to later be able to isolate the relevant features. An important step is the selection of an algorithm to define the classifier. There are multiple types of algorithms that are used in supervised machine learning and depending on the problem specifics; an algorithm should be selected. Since this algorithm selection is an iterative process, only a selection of algorithms is identified in this literature study:

- Decision trees classify instances by sorting them based on feature values. Nodes in a decision tree represent the features and branches represent a value of that feature that a node can assume. Instances are classified starting at the root node.
- Naive Bayes algorithms classify instances based on applying Bayes' theorem with the naive assumption of independence between every pair of features.
- k-Nearest-Neighbours (kNN) algorithms classify instances from simple majority vote of the nearest neighbours of each point; an instance is assigned the data class which has the most representatives within the nearest neighbours of the point.
- Support vector machines (SVMs) classify instances based on a induced boundary that separates these classes by as wide a margin as possible.

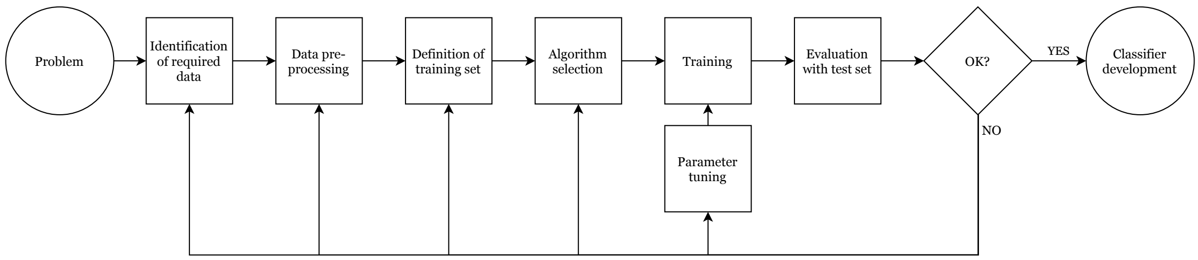


Figure 2.8: Typical solution process of supervised machine learning (based on Kotsiantis (2007))

An example of measuring the performance of the accuracy of a classifier can be evaluated by splitting the data set into the training set and the performance testing sets in multiple different ways. If an error rate is unsatisfactory, the process of should be repeated until a satisfactory classifier is found. From a solution point of view, this supervised learning methodology could be explored further for possible use in a crew preference management context.

2.4. Synthesis of literature review

The approaches and methodology that have been presented in this Chapter form the basis for addressing the research objective that was identified in Section 2.2.6. This Section aims to summarize this into a synthesis of approaches and methodology. This synthesis provides an overview of the available solution strategies. Combining the information of separate parts of the literature review leads to a better overview to scope the research required to address the research objective. Section 2.4.1 presents a synthesis of problem approaches. Here, different approaches to crew rostering and crew preference management will be discussed. These approaches have been presented throughout Section 2.1 and 2.2. Section 2.4.2 presents a synthesis of methodology to address the research objective, which will discuss the methodology for different research sub-objectives, presented in Section 2.3.1. At the end of this Chapter, a table is presented that summarizes the key problem approaches and methodology presented in this literature review.

2.4.1. Synthesis of problem approaches

Considering problem approaches relevant to studying crew preference management, the distinction can be made between approaches to the airline crew rostering problem and approaches to crew preference management.

Approaches to the airline crew rostering problem

In the airline crew rostering problems, the following three types of approaches have been identified:

- Bidlines approach
- Rostering approach with preferential bidding
- Rostering approach with pre-assigned activities

As airlines and airline divisions vary regarding strategies and objectives, so does the rostering problem. An airline can, therefore, adopt different approaches whenever a scheduling problem is solved separately from another scheduling problem which can be the case for different types of crew, different ranks or different fleet types. These approaches differ in the way in which crew preferences are managed. An integrated approach to the crew pairing problem and has also been studied. However, most efforts in research are made to make it computationally viable rather than viewing it from a practical perspective which is desired for the concept of crew preference management.

Approaches to crew preference management across industries

When considering crew preference management in the bidlines approach, literature is very limited. The reason for this might be that managing crew preferences when using the bidlines approach is very straightforward. As crew express their preference for an entire lines of work, the allocation of these lines of work is very straightforward. The seniority principle, in such a case, can even speed up the solution process and allocate, on seniority order, a desired schedule only when possible.

When considering crew preference management in the preferential bidding rostering approach, the heuristic rostering model of Gamache (1998) is widely recognized. Here, a residual subproblem is solved for all crew members and the bid score of each crew members is maximized. Other approaches that have been used are graph colouring model for the feasibility problem in preferential bidding and an exact solution approach to delay the assignment of activities to crew and rosters with a high score to increase the bid score of the overall schedule. The rostering of days off is another approach used in line with the preferential bidding approach. In the case where standard sequences of duty days and days off are sequenced, this has proven to be a promising approach.

When considering crew preference management in the pre-assigned activities rostering approach, the focus is more on the objective function and how this has been formulated. Some authors embed crew preference management in the objective function while others regard them as constraints. Crew preferences are considered alongside pre-assigned activities, crew information and regulations and a trade-off of objectives is made in the problem objective function, based on the multi-objective nature of the problem. When considering crew preference management in other industries, the solution approaches are best to be compared to the pre-assigned activities rostering approach. For example, aspects of crew preference management can be translated into real cost components such as penalty costs for desired roster attributes and bonus costs for undesired roster attributes. The determination of such cost parameters or other parameters within a single problem objective function with multiple objectives is a challenge. Studies do not present methods to identify the cost parameters from a practical point of view which is left open for interpretation of the problem at hand.

Throughout the experiments that have been presented for models that adopt crew preference management, it can be stated that computation times are found to increase exponentially with problem size. This is related to the linear nature of the integer programming methods. However, the problem size concerning crew members was not found to be higher than 400 crew members and most problems dealt with crew divisions of 50 to 100 crew members. Higher computation times are not undesired necessarily as,

Table 2.1: Overview of airline crew rostering problem characterisics, (D = daily, W = weekly, M = monthly, ILP = integer linear program, MCNF = multi-commodity network flow, SP = set-partitioning, the asterix (*) indicates problems dedicated to cockpit crew only)

Author(s) (year)	Planning horizon	Model formulation	Objective function	Solution procedure
Day and Ryan (1997)	W	SP	Minimize rostering costs	Graph based branching heuristic
Gamache (1998)*	M	SP	Maximize roster bidding score	CG embedded in branch-and-bound tree
Gamache (1999)	M	SP	Minimize costs of uncovered pairings	CG
Lučić and Teodorović (1999)*	M	NA	Minimize average and absolute flight time deviation	Algorithm based on pilot-by-pilot and simulated annealing
Dawid et al. (2001)	M	SP	Maximize the total utility value	SWIFTROSTER algorithm
Fahle et al. (2002)*	M	SP	Minimize rostering costs	CG embedded in constraint programming
Cappanera and Gallo (2004)	M	MCNF	Minimize number of uncovered activities	CPLEX
Guo et al. (2006)	NA	NF	Minimize overnight/compensation costs	Rostering heuristic
Zeghal and Minoux (2006)*	W&M	ILP	Minimize total number flight credits	CPLEX and partial tree search heuristic
Achour et al. (2007)*	M	SP	Maximize roster bidding score	CG embedded in exact approach
Souai and Teghem (2009)*	D	SP	Minimize rostering costs and deviations	Hybrid genetic algorithm and local search heuristics
Maenhout and Vanhoucke (2010)	M	SP	Minimize total penalty costs	Hybrid scatter search heuristic, branch-and-price procedure, variable neighborhood search
Saddoune et al. (2011)*	M	SP	Minimize roster and pilot penalty costs	CG and dynamic constraint aggregation
Kasirzadeh et al. (2017)	M	SP	Minimize roster penalty costs	CG embedded in branch-and-bound tree

from a practical point of view, rostering problems are not solved to give an immediate solution. It is desired, however, to have low computational times for rostering problems which can influence rostering decision processes on a more short-term basis rather than having to wait for over more than 5 hours of computing time.

2.4.2. Synthesis of methodology

The problem methodology relevant to studying crew preference management can be subdivided into three main categories. Firstly, methods have been identified for the creation and evaluation of airline crew rosters. Secondly, dynamic approaches to the rostering problem have been identified. Thirdly, decision mechanisms for evaluating crew preferences have been identified. A synthesis of these three categories is given below.

Creating and evaluating airline crew rosters

Crew rostering problems across different domains are described similarly. With a certain set of employees, a set of duties and a set of constraints, a solution needs to be found for individual employee rosters while striving for a specific objective and while meeting a set of rules and regulations that are usually translated in constraints. Problems that take crew satisfaction into the equation, often claim that considering this mimics the industry practice much better. The basic mathematical formulation that is chosen for almost all rostering problems is called the set-partitioning problem (SPP). The objective formulations of different rostering problems presented in the literature vary widely, and the cost structures associated with these objectives also vary widely. In the majority of the problems, multiple objectives are formulated for the problem which regards the cost and quality of the solutions. Table 2.1 gives an overview of crew rostering methods presented in different studies.

When it comes to improvement methods in the rostering problem, multiple heuristics have been presented. Because of the combinatorial nature of the problem and because there are many solutions possible to the subproblems of a rostering model, metaheuristics are the most common type of heuristic used as an improvement method. Authors have been using genetic algorithms to the largest extent, followed by simulated annealing. However, a trend is visible in the fact that authors continue to research possible improvement methods and that heuristics that were not presented before are now compared in literature to genetic algorithms and simulated annealing algorithms. The improvement method suitable for research should be tailored to the problem size, other specifics and desired computation time, which varies widely across studies. When the improvement method is not the topic of research, it might be better to consider heuristics that have been widely applied as a benchmark.

Dynamic approach to crew rostering

Acknowledging the dynamic nature of the crew scheduling problem is especially established in the operational scheduling problem which takes place after the crew roster is published. Disruptions are modeled as stochastic events which need to be resolved in a recovery model. From this disruption management, both approaches and methodology can be adopted. From an initial state of a schedule, successive states can be established using stochastic modeling. Such an approach could also be adopted. The limitations indicate that this is the desired situation which provided opportunities for research in crew rostering that is more related to the industry practice.

Decision mechanisms for evaluating crew preferences

Regarding decision mechanisms that can be adopted to evaluate individual crew preferences, literature is limited. This is mostly due to the fact that the rostering models that are used are suitable for optimizing for the workforce as a whole. In order to make a transition to a more individual approach, a method should be in place to collect data from the state of a schedule and to make a prediction on the decision that should be made. Dynamic modeling and stochastic modeling could be used to define this state representation of the rostering process although the rostering model could still be solved in a deterministic way, with changing inputs in the states that are defined over time. In the context of crew preferences, a prediction could be made on whether or not to grant a certain request and what this would incur in the roster. Using simulations of many rostering processes, data could be collected of the decisions that have been made and how they related to the different model states and objectives. Together with leveraging available airline data, learning techniques that have been presented in this literature study could then be applied to develop an automated decision system for evaluating individual preferences.

2.4.3. Overview of approaches and methodology

Table 2.2 provides an overview of the key approaches and methods that have been presented throughout this literature review. Problem approaches to address the research objective can be obtained from this overview by appropriately selecting an approach or method from each column.

Table 2.2: Overview of the key approaches and methods that have been presented in the literature review

Rostering approaches	**Crew preference management approaches**	**Rostering construction methods**	**Rostering improvement methods**	**Dynamic approach to crew rostering**	**Decision mechanisms**
Bidlines approach	Bids management	Activities in time-space network representation	Branching heuristics	Discrete event semi-Markov process	Induced learning
Rostering approach with preferential bidding	Request management	Linear integer programming in set partitioning problem	Metaheuristics based on simulated annealing	Simulation based modelling with stochastic disruptions	Supervised machine learning classifiers
Rostering approach with pre-assignments	Translation of crew preference management strategy into cost bonuses and penalties	Column generation techniques	Metaheuristics based on genetic algorithms		
	Translation of crew preference management strategy into problem constraints		Metaheuristics based on tabu search		
			Other heuristics		

3

Research design

To establish research that is suitable for answering the research question that was presented in Section 2.2, this chapter is dedicated to the design of this research. Firstly, the scope of the research is explained in Section 3.1 to get a better understanding of the problem within the broader scope of the airline crew scheduling problem. Secondly, the framework for addressing the research objective is presented in Section 3.2, explaining the steps that were required in the research process. Thirdly, the assumptions that have been made in the design of this scope and framework are discussed in Section 3.3.

3.1. Scope of the research project

In the literature review of Chapter 2, the topic of crew preferences in an airline scheduling context has proven to be strictly embedded in the crew rostering problem. In this section, a structured overview of the positioning of crew preferences in the chain of airline schedule planning process is given. This helps in defining and discussing the research with the right scope in mind.

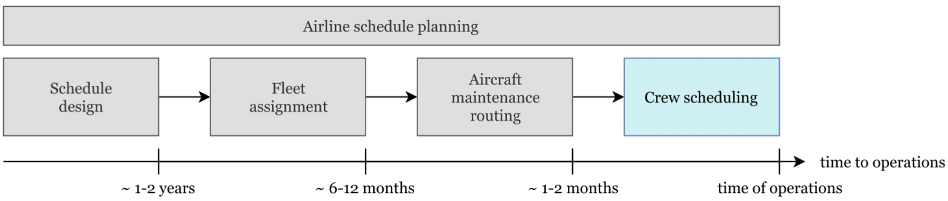


Figure 3.1: Typical process for airline schedule planning with the positioning of the crew scheduling problem indicated in blue

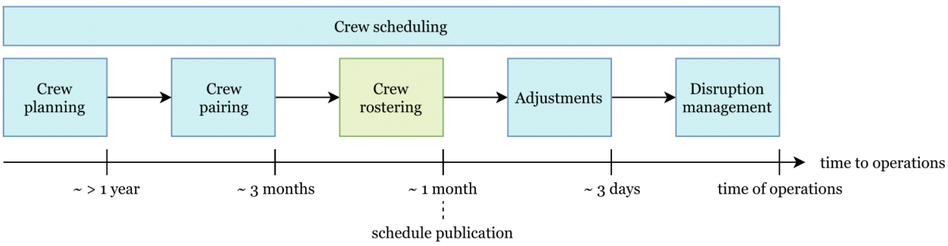


Figure 3.2: Typical process for crew scheduling with the positioning of the crew rostering problem indicated in green

A typical airline schedule planning process is shown in Figure 3.1 to illustrate the overarching process. As was already discussed in Chapter 2, the airline schedule planning process consists of problems that are sequentially solved while often being interconnected. This implies that Figure 3.1 might illustrate the process blocks as successive processes where in practice, feedback loops or interconnections might exist. In this figure, the process of crew scheduling, the last scheduling phase, is indicated with the blue process block. It can be seen that the crew scheduling problem is typically solved in a time horizon from approximately 1 or 2 months before operation up to the time of operation.

Next, the process of crew scheduling is more comprehensively addressed in Figure 3.2 where a typical crew scheduling process is shown. In this figure, the process of crew rostering is indicated with the green box. It is clear that the crew rostering problem is addressed after the crew pairing problem has been solved. The pairings that have been generated are an essential input for the crew rostering problem, in which crew members are assigned to these pairings. Figure 3.2 illustrates that both the crew planning problem and the crew pairing problem can be considered as parts of crew scheduling. Although the time horizon for the crew scheduling problem was stated to be solved 1 to 2 months before operation, the crew planning problem and the crew pairing problem are typically solved in an earlier stage of the process. In crew planning, crew resources are managed such as predictions of supply and demand of workforce, transitions in crew ranks and the establishment of margins to account for absenteeism, reserve duties, and operational disruptions. In crew pairing, flight legs and layover days are combined into unassigned multiple-day duty periods called pairings. In crew rostering, these pairings and other activities are combined into individual lines of work and assigned to crew members. In the adjustment process, crew resource disruptions and operational disruptions are managed to ensure productivity of the schedule in the period directly after schedule publication. In disruption management, operational disruptions are managed to ensure productivity of the schedule in the few days before operations.

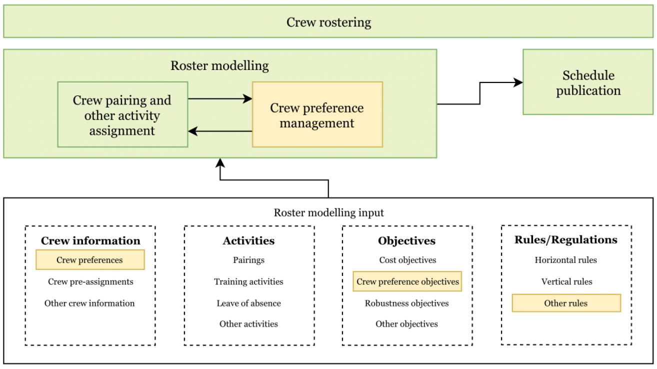


Figure 3.3: Typical process for crew rostering with the positioning of crew preferences indicated in yellow

Crew preferences are taken into consideration in the crew rostering process that is more comprehensively addressed in Figure 3.3. The rostering process consists of roster modeling and crew schedule publication. In the roster modeling process, assignment of activities to crew members is interconnected with the management of crew preferences that need to be taken into account in the activity assignments. A roster is separately constructed for each crew rank and each crew division. In the modeling part, many types of input information need to be taken into account which is indicated in the white block of the figure. These different types of input have already been comprehensively addressed in the literature review of Section 2.1.2. The input information and processes in the figure that are related to crew preference management are indicated with yellow blocks. This figure illustrates that crew preference management is embedded in the crew rostering process. The crew information input comprises the preferences that crew express for certain activities. The objectives input comprises the objectives that are defined for addressing the satisfaction or equity among crew members, and the rules input comprises the rules and regulations that an airline enforces regarding its crew preference management. Now that the topic of crew preference management has been positioned within the airline schedule planning problem, it is important to scope the topic of crew preference management further to clarify the research to be carried out. The different types of crew preferences were identified to be bids and requests. In defining the research objective of this thesis report in Section 2.2, it was pointed out that this research will focus on crew pairing requests. For reason of clarification, the definitions of both crew preference management and crew pairing requests are given below.

Crew preference management

The strategy or policy that an airline carries out to satisfy the objective of managing the satisfaction of individual crew members and the workforce as a whole about the rosters that constitute the solution to the crew rostering problem.

Crew pairing requests
Requests by crew members for the operation of specific desired multiple-day duty periods called pairings.

PAIRING ID	MON	TUE	WED	THU	FRI	SAT	SUN
PA.1	Flight Duty	Flight Duty	Flight Duty	Flight Duty	Rest Period	Rest Period	Rest Period

Figure 3.4: Example of a single pairing composed of four flight duty days and three rest period days

To illustrate the interpretation of a pairing in this research, an example of such a pairing is shown in Figure 3.4. A pairing is composed of flight duty days and rest period days. Pairings can vary in length, and the number of rest period days after a set of flight duty days can additionally vary across contract types. When taking Figure 3.4 as an example, a crew member operating on a full-time basis is eligible for 3 rest period days after the 4 flight duty days. A crew member operating on an 80% basis, meaning that this crew member can be represented in the workforce as 0.8 full-time equivalent or 0.8 FTE, is eligible for 3/0.8 = 3.75 rest period days after the 4 flight duty days in the pairing example. This type of variation in pairing length complicates the crew rostering problem.

PAIRING ID	PAIRING STATUS	MON	TUE	WED	THU	FRI	SAT	SUN	MON	PAIRING REQUEST POSSIBLE?
PA.1	unassigned	FLT	FLT	FLT	FLT	RP	RP	RP		yes
PA.2	unassigned	FLT	FLT	FLT	FLT	FLT	RP	RP	RP	yes
PA.3	unassigned	FLT	FLT	FLT	RP	RP	RP			yes
PA.4	pre-assigned	FLT	FLT	FLT	FLT	RP	RP	RP		no
PA.5	unassigned	FLT	FLT	FLT	FLT	RP	RP	RP	RP	yes

Figure 3.5: Example of a list of five pairings with corresponding pairing status, flight duty days and rest period days, and an indication if the pairings can be requested by crew members

In a typical airline crew request process that was shown in Figure 2.3 in Chapter 2, one of the steps was to collect a set of pairing requests that were submitted by crew members. To facilitate this, an airline typically has a process in place that enables the crew to view the pairings that are up for requesting. To illustrate this, Figure 3.5, shows an example of a list of five pairings indicating whether a crew member can request the operation of each pairing. As can be seen, the fourth pairing with pairing identification *PA.4*, is given the status *pre-assigned*. Requests for the operation of this specific pairing is, therefore, disabled by the airline. Such a pre-assignment can be imposed by the airline for numerous reasons. To give two examples, the flying hours of this pairing might be required for training purposes for a specific crew member or the pairing has been requested and assigned to another crew member in an earlier stage of the scheduling process. In this example, the pre-assignment represents a pairing. However, pre-assigned activities can also represent activities other than pairings such as pre-assigned simulator training sessions, pre-assigned annual leave periods or pre-assigned office duties. A definition that is handled in this thesis report is given below.

Pre-assigned activities
Activities in the crew schedule that have been pre-assigned to specific crew members in an earlier scheduling stage, limiting the solution space of the crew rostering problem.

An important characteristic of pre-assigned activities is their time component. The pre-assigned activities have been executed in an earlier scheduling stage with respect to the assignment problem that needs to consider the pre-assigned activities as a boundary condition. For a crew rostering problem, this means that the pre-assigned activities have been assigned in a scheduling stage prior to the rostering problem. The rostering problem is typically solved one month before operation. However, for the assignment of certain activities, this planning horizon can be considered too narrow. It is this consideration that affects the relationship between pairing requests and pre-assigned activities. An airline can handle a crew preference

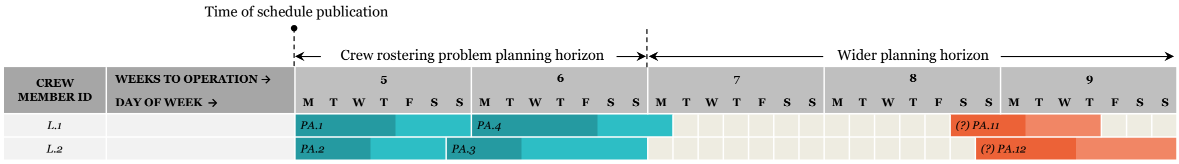
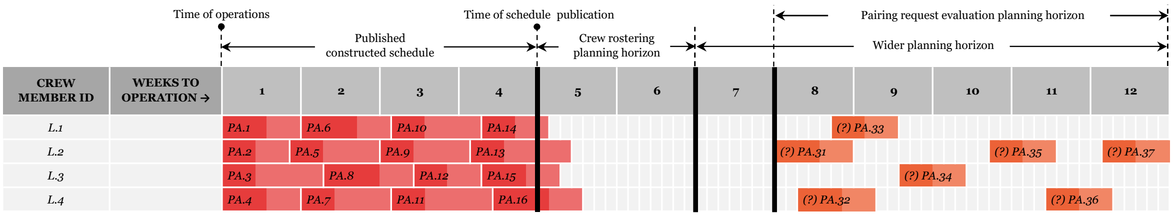


Figure 3.6: Example of two different planning horizons; the crew rostering problem planning horizon in which pairings are assigned and the wider planning horizon for which pairing requests can be submitted

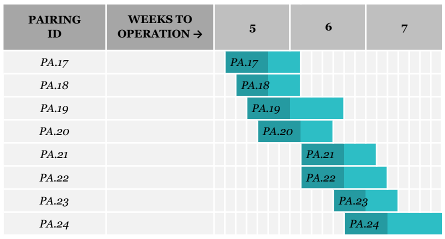
management strategy, in which pairing requests by crew members can be submitted for pairings that take place in a wider planning horizon than that of the crew rostering problem. The current literature does not support such a strategy. Only methods for evaluating crew preferences are presented for preferences that concern the pairings that are assigned in the planning horizon of the crew rostering problem. Consideration of pairing requests in a wider planning horizon is illustrated with an example in Figure 3.6. In the figure, a simplified representation of two rosters is presented for two crew members with crew member ID *L.1* and *L.2*. The planning horizon is subdivided into a planning horizon for which the crew rostering problem is solved, and a wider planning horizon that is not considered in the crew rostering problem. The week numbers in the figure represent the week number relative to the week in which the crew rostering problem is solved. In this example, it is assumed that the rostering problem is solved in the current week 1 for the rosters of weeks 5 and 6. The figure illustrates that for the crew rostering planning horizon, pairings *PA.1*, *PA.2*, *PA.3* and *PA.4* have been assigned to either one of the crew members. For the wider planning horizon, the figure illustrates that crew member *L.1* and *L.2* have submitted a request for the operation of pairing *PA.11* and *PA.12*, respectively. Both of these requested pairings start in week 8. Evaluating these pairing requests is a challenge. As soon as a pairing request by a crew member is approved by the airline, it is transformed into a pre-assigned activity for that crew member. Approved pairing requests, therefore, impose pre-assigned activities in the rosters. It can be argued that increasing the number of pre-assigned activities decreases the flexibility and the solution space of the rostering problem, up to a point at which feasible assignment of all the pairings cannot be guaranteed anymore.
The following two solutions could omit the situation in which pairing requests need to be evaluated in a wider planning horizon with respect to the crew rostering planning horizon:

1. Increase the planning horizon of the rostering problem to a wider planning horizon
2. Evaluate pairing requests in the crew rostering problem planning horizon only

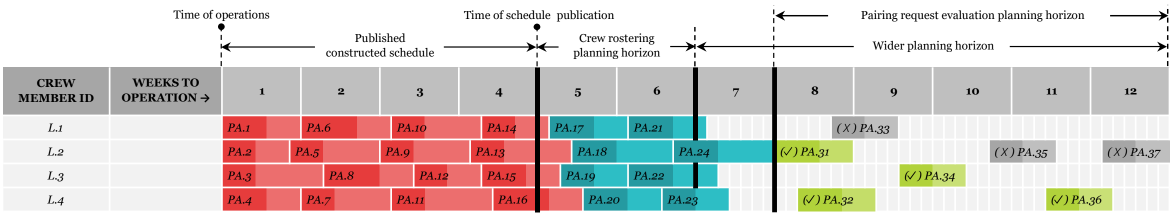
These solutions, however, can be undesired by airlines for different reasons. Firstly, the reason for the first solution not being favorable is mainly from a solution point of view. As the planning horizon increases, so does the problem size and the solution space of the rostering problem. Examples in literature have focused efforts on solving rostering problems for a period of two to four weeks. The planning horizon could even extend further than the planning horizon in Figure 3.6. While the figure illustrates an example in which pairing requests need to be evaluated in week 8, this wider planning horizon could further be extended if the crew preference management strategy demands it. A rostering problem that is solved for a planning horizon of ten weeks is not only impractical in terms of problem size; it also brings about more uncertainty. An example of this is the uncertainty of available workforce at time of operations. The further away from the time of operations, the more uncertainty exists for the projected crew demand due to absenteeism. Secondly, the reason for the second solution not being favourable is from a crew preference management point of view. Crew requests are a means for airlines to enable crew members to influence the pairings they operate. Airlines can handle a policy in which crew members can both submit pairing requests and receive feedback on the decision on their pairing requests within a planning horizon that does not match the planning horizon of the crew rostering problem. The main reason to uphold this policy is to cater to the challenges of irregular and uncertain airline crew rosters. For crew members, it can be desired to receive feedback on pairing requests in an early scheduling stage as it enables crew members to influence their flight duty days and rest period days. For intercontinental long-haul pairings, crew members are not situated at the airline base during flight duty days. During rest period days, they have the opportunity to be at their homes or with their families. Therefore, the pairings they operate directly impact their life apart from work. While crew members can have multiple incentives for requesting a pairing, especially an incentive of requesting from a crew member's personal planning point of view supports the assumption that feedback



(a)



(b)



(c)

Figure 3.7: (a) Example overview of a crew rostering scenario in a wider planning horizon, before solving the crew rostering problem and before evaluating the pairing requests, (b) Example set of pairings to be assigned in the crew rostering problem of this scenario, (c) Example overview of a crew rostering scenario in a wider planning horizon, after solving the crew rostering problem and after evaluating the pairing requests for the pairings starting in week 8

on pairing requests is desired in an early scheduling stage.

To further illustrate this crew preference management approach, an example scenario is illustrated in Figure 3.7. Figure 3.7a shows a crew scheduling scenario in which the week numbers represent the weeks to operation. This suggests that week 1 commences on the next upcoming Monday. From the figure, it can be concluded that the schedules for week 1, 2, 3 and 4 (in red) have already been finalized and published to the crew members. The crew rostering problem will be solved for weeks 5 and 6, for which the pairings are presented in Figure 3.7b. In the planning horizon for pairing request evaluation, Figure 3.7a presents a set of pairing requests (in orange) by the crew members. Figure 3.7c shows the same scenario as 3.7a after the pairings from Figure 3.7b have been assigned by the crew rostering problem and after the pairing requests of pairings that start in week 8 have been evaluated. The request for pairing *PA.33* by crew member *L.1* has been rejected and the requests for pairing *PA.31* and *PA.32* for crew member *L.2* and *L.4* respectively, have been approved. The rejected request does not have to be considered in a later scheduling stage. However, these approved requests are transformed into pre-assigned activities. The pairing request for *PA.31* by crew member *L.2* is approved because this pairing directly succeeds pairing *PA.24* that was assigned to crew member *L.2* in the crew rostering problem, leaving no room for gaps in the roster.

Furthermore, the pairing request for *PA.32* by crew member *L.4* is approved because approving this pairing would leave a gap in the roster of six days between pairing *PA.23* and pairing *PA.32* which could be covered by a six-day pairing that starts on the Wednesday of week 7. The gap of 11 days, possibly caused by approving the pairing request for *PA.33* by crew member *L.1* could be considered too long, and therefore, the request is rejected. It is this type of reasoning that is typical for evaluating pairing requests in the wider planning horizon. However, with an increasing amount of pairings, crew members and pairing requests that are submitted for pairings in weeks even further from the point up to which the rosters are known, the problem of evaluating pairing request becomes increasingly challenging.

This challenge plays a central role in addressing one of the four research questions that were presented in Section 2.2; how could the evaluation of individual crew preferences be modeled and integrated into a crew rostering process? To conclude this section on defining the scope of the research project that can address all research sub-questions, the following elements are part of this research scope:

Research scope

1. To develop a method to model the dynamic nature of the problem
2. To develop a method to model the evaluation of individual crew preferences and integrate this into a crew rostering model
3. To develop an approach to identifying and measuring the (financial) effect of crew preference management on the crew rostering problem
4. To develop an approach to leveraging historical crew preference data to identify and define parameters for modeling crew preference management

3.2. Framework for the research project

The framework for addressing the research scope is presented in Figure 3.8. The framework in the figure shows that the required input for the static rostering model is profoundly driven by historical airline data. Methods for this are described in Chapter 4. The pairing request evaluation modeling is based on historical pairing request data and a corresponding pairing request management policy. The static rostering model is the basis for assigning a set of pairings to a set of crew members in the crew rostering planning horizon. While the static rostering model lays the foundation for the dynamic rostering model, adaptations are made in adopting a rolling roster modeling approach. The dynamic model can be used for simulations on dynamic crew rostering processes with incoming pairing requests that need to be evaluated. The pairing request evaluation algorithms serve as a means to make decisions on these incoming pairing requests. Three pairing request evaluation algorithms are integrated into this dynamic rostering model, which will be explained in the following chapters.

The objective of this research is to make recommendations on methods to evaluate pairing requests while capturing the dynamic nature of the crew rostering problem. As far as literature is concerned, evaluation of pairing requests is a manual process in current airline practices. The recommendations are made based on a case study that is used as an input to the models. This case study concerns a long-haul cockpit crew division of a major European airline. Within the crew rostering process in the case study, pairing requests are currently evaluated manually.

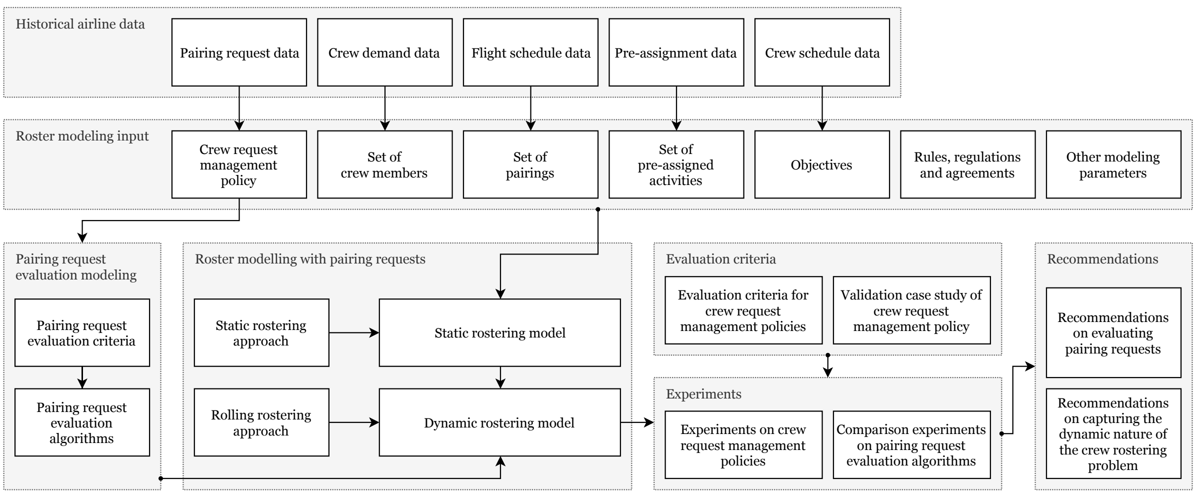


Figure 3.8: Visualization of the framework for the research project

3.3. Assumptions in the research design

The assumptions that have been made in the design of the scope and framework of this research are listed below. As the process of pairing requests is embedded in the modeling the crew rostering problem, that in itself provides many complexities, these assumptions were required in defining attainable project timing:

1. The airline used in the model design uses a single base for its operations.
2. The airline used in the model design is a major European airline regarding size, network, and operational practice.
3. The airline used in the model design uses a fixed salary policy, meaning that crew members are paid a fixed monthly salary based on their rank and contract type, regardless of the flight hours they have operated.
4. The schedule design problem of the airline is already solved, providing a flight schedule of the airline that forms the basis for the creation of a set of pairings departing from the airline base.
5. The crew planning problem of the airline is already solved, providing a crew demand that is required to operate all the pairings in a certain planning horizon. This demand can be expressed in full-time-equivalents or FTEs.
6. The crew pairing problem of the airline is assumed to be solved, providing a set of pairings that all start with a flight departing from the airline base, optionally followed by a set of flights departing from the first destination of the pairing, to then eventually return to the airline base, followed by a corresponding number of rest days.
7. The flight schedule and, therefore, the set of pairings that needs to be operated, is assumed to be the same for each week of operations.
8. In the pairings, the number of flight duty days and the number of rest days, that together comprise the length of the pairings, is represented by rounded integers for simplification purposes. For this reason, the departure time or start time of the pairings and pre-assigned activities is not considered in the model. A pairing or pre-assigned activity in the model starts at 00:00 a.m. on the day of departure or start-day, and ends at 11:59 p.m. on the arrival day or end day. There is thus no room for other activities to be assigned on that day.
9. The information about the pairings does not to change due to operational disruptions. Operational disruptions do not to affect the set of pairings and the pairing information in the crew rostering problem.
10. The model design is applicable for solving a crew rostering problem of a crew division that operates within a single rank within a single fleet family crew division.
11. The crew divisions of the airline in this research are defined by the fleet family that crew members are operating. The crew ranks that are being considered in this research are; captain, first officer, and second officer.
12. The workforce is composed of a set of crew members that all operate based on a full-time contract.
13. The crew rostering strategy of the airline is most closely represented by the personalized rostering approach with pre-assigned activities.
14. The crew rostering strategy of the airline is based on a rolling rostering approach. Adopting this approach means that each week, the existing published schedule is appended with one additionally scheduled week that lasts from Monday to Sunday, four weeks in the future. The schedule of this additionally scheduled week is assumed to be published on the midnight from Friday to Saturday, exactly 30 days before operations of this additionally scheduled week.
15. The model does not account for the effects on rostering of reserve duties and training activities. Flights are used for production only and not used in mixed reserve duties or route instruction.
16. The destinations of the pairings are assumed to be relevant concerning their desirability with respect to pairing requests. For research purposes, the destinations have been given a code name which gives a geographical indication of their locations.

4

Static personalized rostering model

Pairing requests are evaluated within the process of crew rostering. To evaluate the impact of these pairing requests on the airline crew rostering problem, the static personalized rostering model has been developed for this research. In the model, the airline crew rostering problem is solved, while the number of crew requests that should be granted at minimum are constrained. This chapter presents the design of the static rostering model that was introduced in the design of the research framework in Chapter 3. The static personalized rostering model serves as an input to the dynamic rostering model environment. This dynamic rostering model will be addressed in the next chapters. The design of the static rostering model is addressed in Section 4.1. Following, a mathematical representation of the optimization problem underlying the model is presented in Section 4.2. Then, the core model elements will be discussed in Section 4.3. The model input that is driven by historical airline data is discussed in Section 4.4. Following, Section 4.5 presents and discusses the output of the static rostering model. The chapter will be concluded by the verification of the static rostering model in Section 4.6.

4.1. Design of the static rostering model

The static rostering model is a key element of the research framework, as it lays the foundation for pairings to be assigned to crew members. Although being able to assign pairings to crew members is not the main goal of this research, the effects of assignment decisions on the overall performance of rosters and the schedule as a whole can be evaluated using a rostering model. In the design of this rostering model, this research points out a difference between the static rostering model and the dynamic rostering model. The static rostering model, as seen in the research framework of Figure 3.8, serves as a foundation for the dynamic rostering model.

4.1.1. Static rostering approach and rolling rostering approach

The difference between the static rostering model and the dynamic rostering model is the relationship between the models and a time component. The static rostering approach requires a set of crew members and a set of pairings that need to be assigned to these crew members. This set of pairings usually spans a planning horizon of approximately 4 weeks in the airline crew rostering models that have been presented in literature. However, this way of approaching the crew rostering problem assumes that an airline solves the crew rostering problem for a planning horizon of 4 weeks. A disadvantage of this approach is that there is a difference in assigning pairings that are operated in, for instance, 8 weeks with respect to pairings that are operated in, for instance, 4 weeks. An example of such a difference is the stochastic nature of the crew supply as one of the rostering model input sets. There are multiple causes that contribute to variation in crew supply that is available for operating pairings. Examples of causes that are stochastic are long- and short-term illnesses or parental leave. Examples of other causes that contribute to variation in crew supply are the retraining of crew members that shift to other aircraft types and the demand for crew members regarding activities other than pairings. The set of crew members that is available for operating the pairings is more challenging to predict for a time-horizon that is more prospective. In other words, this set of crew members is assumed to be more uncertain 8 weeks before operation than 4 weeks before operation. Another disadvantage of this approach is that airlines need to manage the trade-off between constructing a crew schedule that is prone to disruptions, while also being able to inform crew members about their rosters

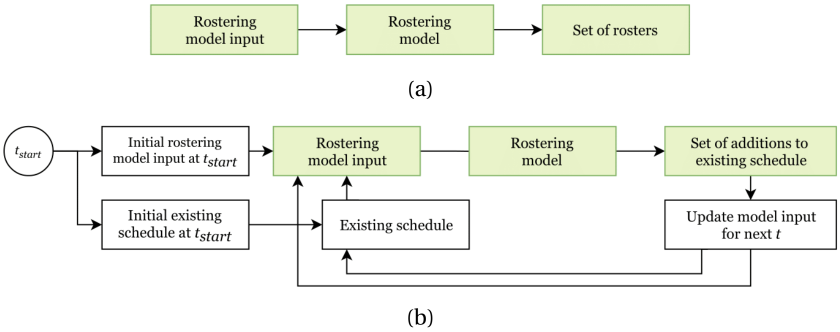


Figure 4.1: (a) Schematic representation of the static rostering approach, (b) Schematic representation of the rolling rostering approach

timely. Although the available literature suggests otherwise, the approach of solving the crew rostering problem on a 4-weekly or monthly basis is not consistently used by airlines.

The different generic definitions for the static and dynamic rostering approach that are used in this research are presented below. The approaches are schematically illustrated in Figure 4.1. Figure 4.1a illustrates the static rostering approach, in which the rostering model constructs a set of rosters for a fixed model input. Figure 4.1b illustrates the rolling rostering approach, in which an existing schedule is initialized, which serves as a starting point for the construction of the roster in multiple time-iterations. The rolling rostering approach will further be discussed in Chapter 5.

Static rostering approach
The approach to roster modeling that requires a set of model inputs and produces a set of crew rosters at a certain moment in time.

Rolling rostering approach
The approach to roster modeling that requires a *reduced* set of model inputs and appends the existing schedule with a small time horizon of activities that is similar to the frequency with which the rostering process is executed.

4.1.2. Design framework for the static rostering model

The static rostering approach is used in the design of the static rostering model. Furthermore, the model is based on a cost-minimizing optimization problem that uses linear programming, which has proven to be an effective method for solving the airline crew rostering problem throughout literature. The model has been visualized in Figure 4.2 for which the process blocks are labeled referring to the static rostering model or SRM.

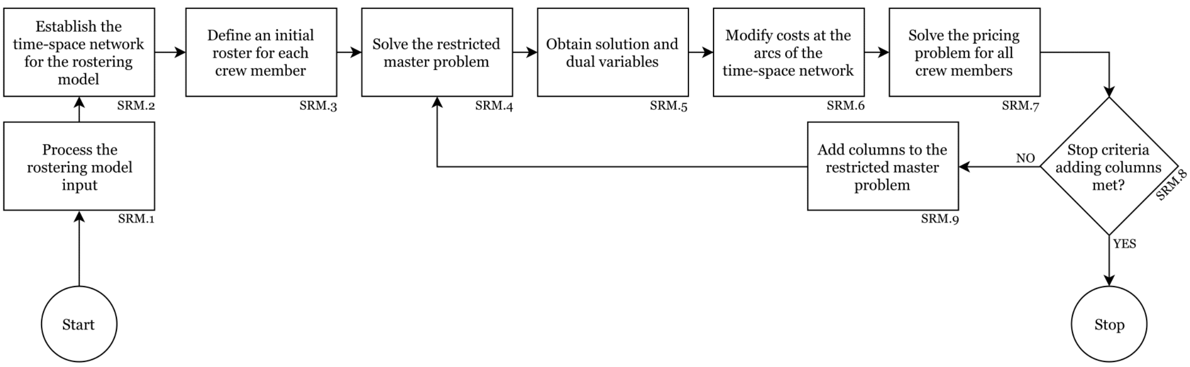


Figure 4.2: Design framework for the static rostering model

The process blocks that are presented in the figure are more comprehensively discussed in the following sections of this chapter, which is indicated for each of the process blocks:

- SRM.1 - The model input and the methods that have been used to process this input are presented in a comprehensive section that is dedicated to the wide variety of model input; Section 4.4.
- SRM.2 - Capturing feasible pairing activity sequencing is achieved by using a time-space network for the rostering model. This has been covered in Section 4.3.1.
- SRM.3 - The initialization of an initial roster for each crew member is required as a starting point to then be able to optimize for more cost-effective rosters. This is explained in Section 4.3.2.
- SRM.4 - Solving the optimization problem for a reduced set of possible rosters in a schedule is an effective way of approaching a crew rostering problem that is based on a linear optimization problem. Process blocks SRM.4 up to and including SRM.9 comprise this so-called column generation algorithm. In this algorithm, possible rosters are added that have the potential to improve the cost-minimization objective of the optimization problem. The initial rosters are a starting point for a so-called restricted master problem that is specified in SRM.4. This is further discussed in Section 4.3.3.
- SRM.5 up to and including SRM. 9 - The remaining process blocks of the column generation algorithm will be covered in Section 4.3.4.

4.1.3. Incorporation of pairing requests into the static rostering model

Modeling of pairing requests as a means of expressing crew preferences has not been incorporated in crew rostering models presented in literature. In the static rostering model in this research, however, this is established by using a data-driven set of input based on historical pairing request data. In an airline practice, the two options for evaluating a pairing request of a certain crew member are to:

1. Grant the request - The pairing request for the requesting crew member is granted; the pairing is pre-assigned to the requesting crew member which is assumed to be irreversible; the pairing cannot be assigned to other crew members in the workforce anymore.
2. Reject the request - The pairing request for the requesting crew member is rejected; the pairing can still be assigned to other crew members in the workforce.

In the static rostering model, a pairing request of a crew member is considered granted whenever the requested pairing is assigned that crew member in the rostering optimization problem. The handling of pairing requests in the static rostering model is achieved similarly compared to the work of Kasirzadeh et al. (2017). When translating the methods of this work to the context of pairing requests, three methods can be used to incorporate crew preferences in the optimization problem of the crew rostering problem:

1. Use a bonus cost parameter (a negative cost parameter) in the objective function for assigning rosters that hold pairings that are requested
2. Constrain the set of assigned pairings that are requested for each roster for each crew member to a predefined minimum *per roster*
3. Constrain the complete set of assigned pairings that are requested to a predefined minimum for the *overall schedule*

In the personalized crew rostering model that is used by Kasirzadeh et al. (2017), the predefined bonus cost parameter reduces the objective value whenever a preferred pairing for a crew member is assigned to that crew member. The drawback of this approach is the valuation of this bonus cost parameter. The sensitivity of the problem solution to the bonus costs could significantly impact the overall costs of the optimization problem. This might not give a clear indication on the trade-off between the costs of a productive and feasible roster on the one hand and the accommodation of pairing requests on the other hand. However, to test this and to both enable and disable the option of using a bonus cost parameter in crew preference management, the bonus cost method has been incorporated into the static rostering model as well. As a model setting, this cost parameter could be set to zero to not have any effect on the roster costs. Using the second and third option is more promising for the static rostering model. Constraining the requested activities to a minimum could be a useful method for experimenting with the sensitivity of the rostering model to pairing requests. Note that these three methods are all integrated into the rostering optimization problem. Other methods for evaluating pairing requests will be discussed later on in this report.

4.2. Mathematical representation of the static rostering problem

The crew rostering problem can be formulated as a mathematical optimization problem. Although several authors have formulated the airline crew rostering problem, efforts to include crew preferences into the problem formulation are minimal. The optimization problem that has been formulated for this research can be used for solving the airline crew rostering problem for a single crew rank that operates a single fleet family. This personalized static rostering optimization problem is presented in this section, that can be solved to construct a schedule that meets the requirements. The problem is described by its required sets, parameters, decision variables, cost function, objective function and constraints that are consecutively presented below.

Sets

- L : set of crew members of the crew type considered
- S : set of slack crew members of the crew type considered
- P : set of pairings to be covered by the crew type considered
- R_l : set of personalized rosters for crew member $l \in L$
- R_s : set of personalized rosters for slack crew member $s \in S$

Cost parameters

- c_b : bonus cost for assigning a requested pairing in the roster
- c_v^i : cost of void in the roster with a void length of i number of days
- c_r^l : cost of roster $r \in R_l$ for crew member $l \in L$
- c_r^s : cost of roster $r \in R_s$ for slack crew member $s \in S$

Other parameters

- $(n_v^i)_r^l$: number of voids v in personalized roster $r \in R_l$ for crew member $l \in L$ with a void length of i number of days
- q_r^l : number of requested and assigned pairings in personalized roster $r \in R_l$ for crew member $l \in L$
- q : minimum desired number of requested and assigned pairings in the schedule
- $e_p^l = \begin{cases} 1 : \text{if pairing } p \in P \text{ is requested by crew member } l \in L \\ 0 : \text{otherwise} \end{cases}$
- $e_p^{r,l} = \begin{cases} 1 : \text{if pairing } p \in P \text{ is chosen for personalized roster } r \in R_l \text{ for crew member } l \in L \\ 0 : \text{otherwize} \end{cases}$

Decision variables

- $x_r^l = \begin{cases} 1 : \text{if personalized roster } r \in R_l \text{ is chosen for crew member } l \in L \\ 0 : \text{otherwise} \end{cases}$
- $x_r^s = \begin{cases} 1 : \text{if personalized roster } r \in R_s \text{ is chosen for slack crew member } s \in S \\ 0 : \text{otherwise} \end{cases}$

Cost function

$$c_r^l = \left(\sum_{i=1}^{j} (n_v^i)_r^l \cdot c_v^i\right) + \left(q_r^l \cdot c_b\right) \tag{4.1}$$

Objective function

$$\text{Minimize} \sum_{l \in L} \sum_{r \in R_l} c_r^l \cdot x_r^l + \sum_{s \in S} \sum_{r \in R_s} + c_r^s \cdot x_r^s \tag{4.2}$$

Constraints

$$\sum_{l \in L} \sum_{r \in R_l} e_p^{r,l} \cdot x_r^l = 1 \quad , \quad \forall p \in P \tag{4.3}$$

$$\sum_{r \in R_l} x_r^l = 1 \quad , \quad \forall l \in L \tag{4.4}$$

$$\sum_{l\in L}\sum_{r\in R_l}\sum_{p\in P} e_p^{r,l}\cdot e_p^l\cdot x_r^l \geq q \tag{4.5}$$

$$x_r^l \in \{0,1\} \quad , \quad \forall l \in L, \forall r \in R_l \tag{4.6}$$

$$x_r^s \in \{0,1\} \quad , \quad \forall s \in S, \forall r \in R_s \tag{4.7}$$

Equation 4.1 represents the cost function of a roster $r \in R_l$ for $l \in L$. It is composed of a component that accounts for void costs and a component that accounts for bonus costs for assigned preferred pairings. Equation 4.2 is the objective function that consists of the sum of the product of the decision variables and their corresponding cost parameter. Equation 4.3 represents the set of constraints that ensures that all of the pairings are covered and chosen for one and only one (slack) crew member. Equation 4.4 represents the set of constraints that ensures that all (non-slack) crew members are assigned one and only one roster. Equation 4.5 represents the set of constraints that ensures that the total number of the assigned pairings that were requested by crew members is greater than or equal to the minimum desired number of requested and assigned pairings in the schedule. Equation 4.6 and 4.7 represent the integrality constraints, which ensure that the value of all the decision variables can be either one or zero.

4.3. Elements of the static rostering model

Referring to the static rostering model framework of Figure 4.2, three critical elements of the model can be identified. Firstly, a time-space network is required to construct a framework for feasible sequencing of activities in terms of time and space (SRM.2). Secondly, a shortest path algorithm is needed to generate sequences of activities, or, in other words, rosters for crew members (SRM.3). Thirdly, integer linear programming in combination with a column generation algorithm is used to efficiently generate multiple possible rosters for crew members to eventually find a solution for each crew member (SRM.4 up to and including SRM.9). These three key elements are discussed in the next sections.

4.3.1. Time-space network representation for the static rostering model

To construct rosters in which the activities feasibly subsequent each other, a method is required to represent these activities in both time and space (SRM.2 in Figure 4.2). The time component is needed for feasible sequencing of activities in terms of time. In a single roster, for example, a pairing that ends on the fourth day of the schedule cannot be directly succeeded by a pairing that starts on the third day of the schedule. The space component is required for feasible sequencing of activities in terms of space. In a single roster, for example, a pairing that ends in North America cannot be directly succeeded by a pairing that starts on in Europe without an activity in between connecting the two locations. This is achieved by representing the activities in a time-space network graph. Such a network graph is composed of nodes (or vertices) that are connected by arcs (or edges). The nodes represent instances of time and space and the arcs represent the possible connections between the nodes. The time-space network that is constructed for the rostering models in this research consists of different types of nodes and arcs that are represented in Table 4.1 and 4.2, respectively. The representations of these nodes and arcs are presented in the tables. To give a visual impression of time-space network using these nodes and arcs, Figure 4.3 shows an example with three pairing activities and one pre-assignment in a 7-day planning horizon. The horizontal and vertical axes indicate the time and space component of the network, respectively. A distance or cost value can be assigned to an arc which indicates the cost of traversing this arc with respect to other arcs in the network graph. In the case of crew rostering, for example, it could be more desired for a crew member to operate a pairing than to be unproductive at the airline base. Such more desirable roster constructions can, among other things, be manipulated by the costs of the arcs.

4.3.2. Shortest path algorithm

The time-space network alone is not sufficient to be able to generate rosters for crew members. In the time-space network of Figure 4.3, multiple possible paths between the schedule source node and the schedule sink node can be identified. However, not all these paths that represent rosters are equally desired in the problem solution as the objective is to minimize costs. Therefore, cost parameters are assigned to the different types of arcs which influence the attractiveness of choosing a certain path or roster above another roster. The time-space network graph is constructed for each crew member and based on this network graph, a set of feasible

Table 4.1: Overview of network nodes in the time-space network representation for the static rostering model

Node type	Node nomenclature	Node time and space representation	Node created for
SON	Schedule source node	Time and space of the beginning of the schedule at the airline base	The schedule as a whole
MNN	Midnight node	Midnight (00:00 a.m.) at the airline base	Each midnight in the schedule planning horizon
PSN	Pairing start node	Departure time of a pairing from the airline base	Each pairing
PEN	Pairing end node	Arrival time of a pairing at the airline base	Each pairing
VSN	Pre-assignment start node	Start time of a pre-assignment at the airline base	Each pre-assignment
VEN	Pre-assignment end node	End time of a pre-assignment at the airline base	Each pre-assignment
SIN	Schedule sink node	Time and space of the end of the schedule at the airline base	The schedule as a whole

Table 4.2: Overview of network arcs in the time-space network representation for the static rostering model

Arc type	Arc nomenclature	Arc representation	Arc created for
SSA	Schedule start arc	Connecting the beginning of the schedule to each midnight node	Each SON-MNN combination
PSA	Pairing start arc	Connecting each midnight node to the pairings start nodes, for the pairings that start on the day of the midnight node	Each feasible MNN-PSN combination
PA	Pairing arc	Connecting beginning and end of a pairing activity	Each corresponding PSN-PA pair
PEA	Pairing end arc	Connecting the pairing activity ending to the airline base	Each corresponding PA-PPA pair
VSA	Pre-assignment start arc	Connecting each midnight node to its corresponding pre-assignment start node	Each corresponding MNN-VSN pair
VA	Pre-assignment arc	Connecting beginning and end of a pre-assigned activity	Each corresponding VSN-VEN pair
VEA	Pre-assignment end arc	Connecting the pre-assigned activity ending to the airline base	Each corresponding VEN-MNN pair
BSA	Base arc	Connecting each midnight node to its subsequent midnight node to represent a non-active day at the airline base	Each consecutive MNN-MNN pair
SEA	Schedule end arc	Connecting the final midnight node in the schedule planning horizon to the end of the schedule	The final MNN to SIN combination only

rosters needs to be constructed for each crew member. Retrieving possible paths through the graph, that represent rosters, is done using Dijkstra's algorithm. This algorithm is a greedy shortest path algorithm for planning a route between two specific nodes in a graph. The six steps of the algorithm are briefly described below:

1. Mark all nodes in the time-space network as unvisited and add to the set unvisited nodes.
2. Assign a tentative distance value to all nodes representing costs in the rostering problem. The start node (schedule source node or SON) is assigned 0; all other nodes are assigned ∞. Set the start node (schedule source node or SON) as the current node.
3. For each unvisited neighbor node of the current node, calculate the tentative distance to these nodes when visited through the current node. Compare the new tentative distance of each unvisited neighbor node to current tentative distance and assign the smallest of the two as updated tentative distance.
4. Mark the current node as visited and remove from the set unvisited nodes.
5. If the end node (or schedule sink node) is marked visited, then stop and finish the algorithm.
6. If the end node (or schedule sink node) is not yet marked visited, then select the neighbor node of the current node with the smallest tentative distance, set it as the new current node and go back to step 3.

To get an impression of how this algorithm functions in the context of the time-space network graph of Figure 4.3, Figure 4.4 shows an example of the shortest path that is found between the schedule source node

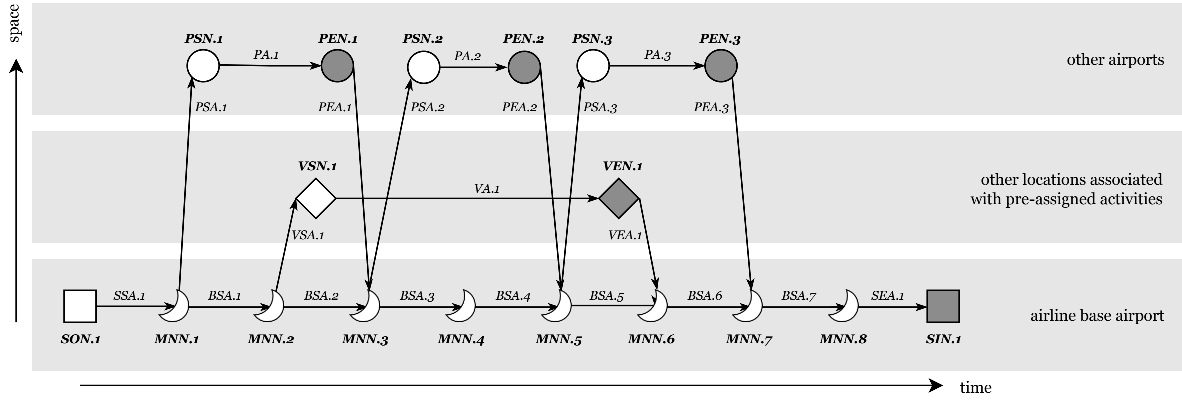


Figure 4.3: Example of a time-space network graph with three pairing activities and one pre-assignment in a 7-day planning horizon

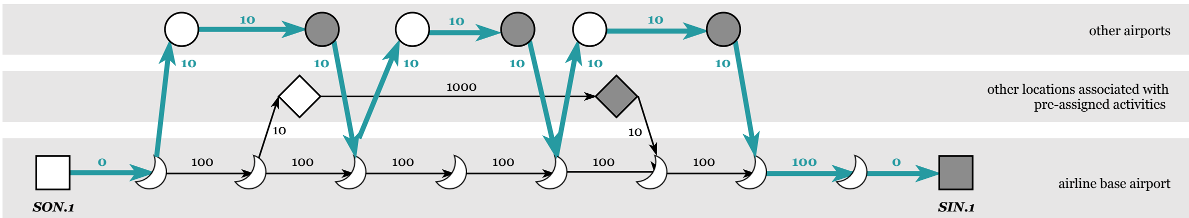


Figure 4.4: Example of the shortest path between the the schedule source node *SON.1* and the schedule sink node *SIN.1*

SON.1 and the schedule sink node SIN.1. In Figure 4.4, this shortest path is highlighted in blue. The shortest path algorithm is also used for defining an initial roster for each crew member (SRM.3 in Figure 4.2).

4.3.3. Integer linear programming

The optimization framework for the static rostering model is based on integer linear programming or ILP. Integer linear programming is a mathematical optimization method to select a solution that performs best in terms of the problem objective, with respect to other alternatives. Characteristic to integer linear programming is that the objective function and the constraints of the optimization problem are linear and all decision variables take on integer values. This is the case for the optimization problem of the static rostering model as is clear from the problem formulation in Section 4.2. In the rostering model, a set of feasible rosters is generated for all crew members. The optimization step that is then required is to minimize the costs of the combined set of rosters that are chosen for the full workforce. The optimization of the integer linear programming problem in the rostering model is achieved using a well-recognized optimization solver (Gurobi Optimization, 2018).

4.3.4. Column generation algorithm

The shortest path algorithm is used to generate the most cost-effective feasible roster for each crew member based on the time-space network graph that was constructed for that crew member. Then, the optimization solver accounts for solving for the optimal solution to the formulated rostering optimization problem. However, as was already clear from the optimization problem formulation in Section 4.2, a set of multiple feasible rosters is required (R_l) for each crew member ($l \in L$) to achieve a feasible global solution to the rostering problem. One of the possibilities is to generate the full set of all the possible rosters for each crew member. In larger-scale rostering problems, hundreds or even thousands of possible rosters may exist for each crew member. However, only one roster can be chosen for each crew member. A full enumeration method would, therefore, lead to many non-basic decision variables with a value of zero. Computationally, such an approach can be very large and impractical. A better approach is to start with a manageable part of the problem and from that point onwards, determine rosters to further explore by analyzing the solution to the partial problem that exists up to that point. Generating the set of feasible rosters in this way can be achieved by the column generation algorithm. Columns refer to the format in which the composition of the decision variables can be visually presented. This is done in Table 4.3, where the columns represent the decision variables and the rows represent the constraints and other model components. The table can be considered as a visual representation of the mathematical model presented in Section 4.2. In the rows, an indication is given on whether a decision variable applies to that row using binary values. For example, the row for pairing constraint 1 indicates that pairing PA.1 is present in the following rosters: r_1 for $l = 1$, r_2 for $l = 2$ and r_1 for $s = s$. The constraint sign (i.e., =) and the constraint right-hand side (i.e., 1) indicate that the sum-product of the binary value indicators in the row and the decision variable solution should be equal to one. This implies that pairing *PA.1* should be chosen for one and only one roster in the problem which was also clear from the pairing constraint from Equation 4.3 in Section 4.2. The representation in Table 4.3 is referred to as the restricted master problem. It is a *master problem* as it holds all the information required to solve the rostering optimization problem and it is *restricted* since the problem is limited to set set of rosters ($r \in R_l, \forall l \in L$). The restricted master problem, thus, does not provide an exhaustive overview of all the feasible rosters for each crew member but it uses a greedy method to add only those columns to the restricted master problem that are expected to reduce costs to the overall solution. This method is called column generation, and it is a recognized method in airline crew rostering since it relies on the idea that only rosters (i.e., columns) which have the potential to improve the objective function should be added to the restricted master problem. The steps of the column generation algorithm were introduced in the design framework for the static rostering model of Figure 4.2 and are repeated and explained below:

- SRM.4 - Solve the relaxed version of the restricted master problem - Relaxing the solution process means not to force the decision variables to be integer values. This results in a solution that would not be feasible in the non-relaxed problem, but this allows for the possibility to retrieve dual variables from the solution to the dual problem which refers to the constraints of the primal problem.
- SRM.5 - Obtain the solution and dual variables - If there is a dual variable that is lower than zero, then there is still an opportunity to improve the objective function of the primal problem.
- SRM.6 - Modify the costs at the arcs of the time-space network graph - The costs of the arcs are modified by adding the (negative) reduced costs represented by the dual variables.
- SRM.7 - Solve the pricing problem for all crew members - The pricing problem represents the generation of new rosters using the shortest path algorithm that was described earlier. A new roster is generated for all crew members. This time, however, a time-space network graph is formulated with modified edges distances or arc costs.
- SRM.8 - Check if the stop criterion for adding columns is met - If the cost of the newly generated roster is lower than the current lowest cost of a roster for this crew member, then the roster (column) is added to the restricted master problem.
- SRM.9 - Add columns to the restricted master problem - This pricing problem and checking whether to add columns is done for each crew member. If the costs for a roster for a crew member are lower than already existing rosters for that crew member, the roster is added to the restricted master problem.

4.4. Model input

In the research design that was presented in Section 3.8, it was indicated that the input to the model is driven by historical airline data. Table 4.4 presents the different types of input to the model, along with an indication of what data has been used to retrieve this input. This section will discuss the methods used for translating the historical airline data to model inputs.

Modeling crew members

In the airline crew scheduling problem, it is a challenge to determine the number of crew members required to operate a given pairing schedule. Many aspects influence the number of crew required to be able to guarantee the productivity of the flights and other required activities. For an airline, there usually is an absolute minimum number of crew members that is required to operate the pairings in a situation in which all crew members would operate on a full-time basis, and all would be dedicated to operating pairings only. This minimum is referred to as the *net crew demand* and it is expressed in full-time equivalents or FTEs.

- **Net crew demand** - L_{dem} - The net crew demand is the minimum crew required to operate the set of pairings for a specific aircraft type for a specific rank for a given period. It defines the set of crew members for productiveness of the pairings. It forms the basis of all the increments that need to be

Table 4.3: Visual representation of the mathematical rostering model in which the rosters for the crew members are represented as columns

Objective function, constraints, cost parameters and decision variables:		**Crew rosters represented by columns in the restricted master problem:**														**Constraint sign:**	**Constraint RHS:**
objective function: Minimize $\sum_{l \in L} \sum_{r \in R_l} c_r^l \cdot x_r^l + \sum_{l \in L} \sum_{r \in R_s} c_r^s \cdot x_r^s$		crew members $l \in L$													slack crew members $s \in S$		
		rosters $r \in R_l$ for $l = 1$			rosters $r \in R_l$ for $l = 2$			rosters $r \in R_l$ for $l = 3$				rosters $r \in R_l$ for $l = l$			slack rosters $r \in R_s$ for $s = s$		
		r_1	r_2	r_3	r_1	r_2	r_3	r_1	r_2	r_3	...	r_1	r_2	r_3	r_1		
pairing constraints:	pairing constraint 1	1	0	0	0	1	0	0	0	0	...	1	0	0	1	=	1
	pairing constraint 2	0	1	0	1	0	0	0	1	0	...	0	0	1	1	=	1
	pairing constraint 3	0	0	1	0	0	1	1	0	0	...	0	1	0	1	=	1
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
	pairing constraint p	1	0	0	0	0	0	1	1	0	...	1	0	1	1	=	1
crew constraints:	crew constraint 1	1	1	1	0	0	0	0	0	0	...	0	0	0	0	=	1
	crew constraint 2	0	0	0	1	1	1	0	0	0	...	0	0	0	0	=	1
	crew constraint 3	0	0	0	0	0	0	1	1	1	...	0	0	0	0	=	1
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
	crew constraint l	0	0	0	0	0	0	0	0	0	...	1	1	1	0	=	1
request constraint:	request constraint 1	0	1	2	0	1	1	0	0	1	...	0	1	1	1	≥	q
cost parameters:	c_r^l , c_r^s	c_1^1	c_2^1	c_3^1	c_1^2	c_2^2	c_3^2	c_1^3	c_2^3	c_3^3	...	c_1^l	c_2^l	c_3^l	c_1^s	**Objective value:**	
cost parameter values:		1000	**800**	750	**1000**	900	850	1200	850	**800**	...	900	**750**	700	Big M	result of objective function: **3350**	
decision variables:	x_r^l , x_r^s	x_1^1	x_2^1	x_3^1	x_1^2	x_2^2	x_3^2	x_1^3	x_2^3	x_3^3	...	x_1^l	x_2^l	x_3^l	x_1^s		
decision variables solution:		0	**1**	0	**1**	0	0	0	0	**1**	...	0	**1**	0	0		

Table 4.4: Overview of the static rostering model input sets and the airline data that has driven these sets

Input set	**Airline data**
Set of crew members	Historical crew demand data
Set of pairings	Historical flight schedule data
Set of pairing requests	Historical pairing request data
Set of pre-assigned activities	Historical operational crew scheduling data
Rules, regulations and agreements	Historical airline collective labour agreement data
Other modeling parameters	Historical operational crew scheduling data

taken into account that increase the required number of crew members for a given period. Examples of such increments for the crew demand for a given period are the part-time percentage of crew members, the expected absenteeism, the expected requirement for other activities (such as training sessions, aircraft type retraining periods, office hours).

When analyzing historical airline data on crew demand, the net crew demand has been found to vary each week. This can be explained by the operation of en-route instruction flights, that are part of the net crew demand. In some weeks, more en-route instruction flights are operated than in other weeks, influencing the net crew demand.

Modeling slack crew members

In the modeling of the slack crew members, the decision has been made to create a slack crew member for each pairing that needs to be assigned in the rostering model. This method has proven to be convenient in analyzing unassigned pairings in the rostering process.

- **Pairing specific slack crew member** - s_{pa} - The slack crew members $s \in S$ in the model are created for each pairing $p \in P$. A time-space network for a slack crew member s is created that connects the schedule source node and the schedule sink node via the pairing arc corresponding to pairing p, only.

Modeling pairings

The airline flight schedule consists of individual flight legs that are generally repeated every week for a given period. These flight legs, together with rest duty days are combined into a set of pairings. Therefore, the flight schedule can be reduced to a pairing schedule, in which a pairing is represented by a reduced set of information on the pairing rather than the individual flight legs in the pairing. The pairings that have been used in this research have a minimum of two flight legs and a maximum of four flight legs. For long-haul cockpit crew, it can be assumed that the crew complement of a pairing does not change during the execution of such a pairing. The following information on the pairings has been collected and combined as the pairing input to the rostering model:

- **Pairing ID** - pa_{id} - All the pairings in the model have been given a unique identifier rather than identifier that is weekly repeated such as a flight number. This is especially important when looking at pairings in different weeks.
- **Destination ID** - pa_{dest} - The destinations in the airline pairing schedule have been given a pseudo-code destination identifier rather than an exact IATA (International Air Transport Association) airport code destination identifier. This has been done to highlight the geographical indication of the destination rather than the exact destination. The destination of a pairing can be a driver for airline crew to request that pairing, so it has, therefore, been considered as input.
- **Departure time** - pa_t - The departure time indicates the start date of a pairing. Since it has been decided not to use the specific start time in this research, the start time of a pairing is assumed to be 00:00 a.m.
- **Flight duty days** - pa_{rd} - The flight duty days indicate the number of days that are required for operating the flight legs in a pairing, including potentially required rest time between the operation of flight legs.
- **Rest period days** - pa_{rp} - The rest period days indicate the number of days that are part of a pairing in which a crew member is eligible for rest time before operation of the next pairing.

Data on the weekly pairing schedule, along with a visualization of the pairing schedule used in this research is presented in detail in Appendix A. It should be noted that the pairing schedule as presented consists of a list of 71 pairings that is operated every week. The pairing identifiers are enumerated accordingly.

Modeling pairing requests

The modeling of pairing requests in the model is a crucial step in the rostering model that focuses on integrating the evaluation of these requests. To get an insight on pairing request behavior in a crew scheduling environment, historical data on pairing requests has been analyzed for this research. This data has not been available in work of other authors. Previous work relied on the generation of random preferences of crew members that was not driven by data. After analyzing historical pairing request data, additional assumptions have been made in the modeling of pairing requests:

1. Requesting of pairings is a random process and should be modeled as such. Three main aspects together comprise the randomness of this process for each crew member. The first aspect is the timing of requests which determines *when* in the scheduling process a crew requests a specific pairing. The second aspect is the decision about which pairing is requested by which crew member. The third aspect is the number of requests that are submitted by each crew member. While the third aspect can vary across crew members, a decision has been made to only model the first two aspects in the interest of this research as model input of stochastic nature.

2. The supply of pairings that can be requested varies for each change in the pairing schedule. The pairing schedule for a specific crew type may vary over time during the year. The reason why this is important when analyzing pairing request data is that the destination of a pairing is one of the incentives to submit a request for operating a pairing. Requests are a type of preference, and a preference for a pairing exists only with respect to other pairings. For this reason, it is considered more accurate to determine the attractiveness of pairings based on the data in the months of operation in which the pairing schedule used for the rostering model is handled, rather than concluding on years of historical pairing request data.

In a real-life scheduling environment, pairing requests are submitted by crew members on a random basis. Some pairings, however, are more likely to be requested. There are multiple reasons for this, such as the desirability of pairings to certain destinations or certain days of the week. From the data analysis, the likelihood with which pairings will be requested in the future has been determined. This likelihood is a relative comparison of the count of requests for certain pairings with respect to the total count of pairing requests. For the set of pairings that have been presented in Appendix A, the so-called *pairing request probability* for each pairing is given.

- **Pairing request probability** - $pa_{p_{request}}$ - Probability with which a certain pairing is requested by a crew member when being prompted to request a pairing from the weekly set of pairings in a given planning horizon.

- **Number of pairing requests per crew member** - $n_{l_{request}}$ - The number of pairing requests that each crew member submits for each week of operations.

Table 4.5: Number of occurrences of each carry-in length corresponding to the pairing schedule of Appendix A

Carry-in length in days	0	1	2	3	4	5	6	7	8	9	10
Number of occurrences	14	8	7	12	8	10	9	1	0	1	1

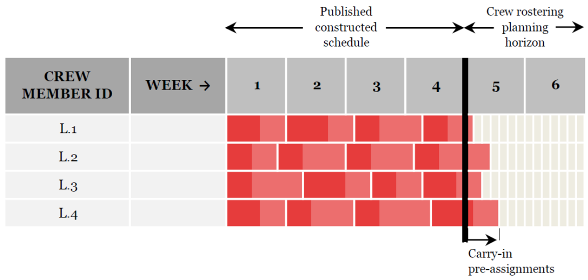


Figure 4.5: Visual representation of carry-in pre-assignments in the crew rostering problem

Modeling pre-assigned activities

An airline schedule is continuous in terms of time. Therefore, overlap of assigned activities from another planning horizon need to be taken into account. The most important reason for this is that continuity of a schedule needs to be considered. This is especially important for activities that span multiple days, which is the case for long-haul pairings described in this research. This overlap of activities across two planning horizons has been described by Medard and Sawhney (2007) as *carry-in* to a planning horizon. In this research, the length of the overlap from the published schedule is referred to as *carry-in pre-assignments*. Figure 4.5 illustrates that the pairings that start in week 4 have an overlap with week 5 that varies in length. Since it is assumed that the pairings of the pairings schedule are repeated on a weekly basis, the number of occurrences of each carry-in length possibility can be determined on a deterministic basis. For each pairing in the pairing schedule from Appendix A, the carry-in length has been determined leading to the frequencies of carry-in length presented in Table 4.5. In the model, a carry-length has been determined for each crew member to account for this overlap effect in the static rostering model in which the rostering model for week 4 is assumed solved. Determining the carry-in length for each crew member is achieved by choosing a random instance from the list of all carry-in length possibilities. If the number of crew members is greater than the number of pairings that need to be covered, the surplus crew members, as well as the slack crew members, will be assigned a carry-in length of zero days.

- **Carry-in pre-assignment length** - $l_{carry-in_{length}}$ - Length of the overlap of activities for a specific crew member from a preceding planning horizon into the planning horizon that is currently considered in the crew rostering problem.

Modeling rules

Rules and regulations are an important input to a crew rostering problem. However, the assumptions that have been made in the design of this research in Section 3.3 allow for not having to test an extensive set of rostering rules explicitly. Rules that apply to rostering in an airline practice mostly concern the working hours of crew members. In this research, the legality of working hours is already captured in the modeling of pairings. Since the pairings are modeled as an activity that captures both the flight duty days and the rest period days, it is assumed that a crew member is eligible to operate the next pairing directly after the previous pairing. In terms of working hours rules, it is assumed that this has been accounted for in the construction of the pairings.

A set of rules that are relevant to this research concerns crew preference management. In the mathematical model of Section 4.2, a constraint that set a minimum (q) to the granted number of requested flights was formulated in Equation 4.5. This number is a parameter for the static rostering model to enforce a higher degree of granted pairing requests which can be varied to test the sensitivity of the model to the enforcement of crew preferences.

- **Minimum of requested and assigned pairings** - q - It is challenging to determine the minimum of requested and assigned pairings in the schedule of the whole workforce. This minimum is usually zero by default since the productivity of pairings is of higher priority. However, the crew preference management strategy of an airline crew division can have the incentive to grant requests to a certain degree. The minimum q can be used to grant requests to a certain degree. It should be noted that this parameter captures the absolute minimum. It is, therefore, sensitive to the number of requests that are submitted by crew members in a crew rostering scenario. For example, if the model settings only allow for one pairing request for each crew member, the chances are low that a feasible solution is found when setting q to a very high value. This parameter can, therefore, be adapted to express a relative service level of preferences to be granted.

Modeling cost parameters

The cost parameters of the static rostering model are highly significant since these parameters are incorporated into the objective function and the heuristic solution process. An overview of the cost parameters of the static crew rostering model is presented in Table 4.7. In the table, the first four cost parameters require some additional explaining that is given below. The other cost parameters in the table refer to the costs for the arcs in the time-space network that was presented in Section 4.3.1.

In the static crew rostering model of this research, a novel method for defining roster costs is introduced that is driven by historical airline data. This method relies on the non-linearity of the costs of voids in the roster.

- **Cost of voids in the roster** - c_v^i - When a roster of a single crew member is created, the desirable output is a roster where activities seamlessly succeed each other. No voids of non-active days exist in such a roster — however, this a challenge when constraining the roster to lengthy activities such as long-haul pairings. In this research, a roster void is referred to as a period of unassigned days in the roster that can vary in length. Airlines can resort to multiple strategies to use voids effectively. Examples of these strategies are:
  - Using voids between long-haul pairings for one-day (simulator) training sessions at the airline base
  - Using voids between long-haul pairings for other activities such as mandatory medical checks, theoretical qualification exams or office duties
  - Using voids between long-haul pairings for paying out a balance of yearly leave days of a crew member

Table 4.6: Percentage of total loss days of total count void days in a void with a length of i days and corresponding index for calculating void costs c_v^i of a void v with void length of i days

Void length in i days	**Percentage total loss days of total void days occurring in a void of i days**	**Cost index (k^i_{index})**
1	17.50%	1.00
2	25.00%	2.85
3	31.77%	5.45
4	19.55%	4.47
≥ 5	29.38%	8.39

Although there are ways to use voids for other activities, airlines usually have to calculate a number of void days that cannot be used productively and should, therefore, be considered as *loss days*. Not only do loss days already exist in the solution of a crew rostering problem, but these loss days are also induced by the planning process of schedule adjustments and disruption management that has been discussed in Section 3.1. In the analysis of voids in the roster using historical airline schedule data, the hypothesis has been tested that argued that the degree to which loss void days occurred in a void, is different for voids of different lengths. A data analysis experiment was designed to test of which the results are presented in Table 4.6. Firstly, operational airline data has been analyzed to count the total number of instances where a void with a length of i number of days occurred in the actual operation of the schedule. Secondly, the total number of loss days in such voids with a length of i number of days was summed. With both the number of instances of voids of length i and the total number of loss days occurring specifically in voids of length i, a percentage of total loss days of total void days has been determined for each void length i. Referring to Table 4.6, this suggests that 10 voids with a length of 1 day eventually result in $10 \cdot 1 \cdot 0.175 = 1.75$ loss days and that 10 voids with a length of 2 days eventually result in $10 \cdot 2 \cdot 0.25 = 5$ loss days. What is remarkable in the table is that the percentage of total loss days of total void days does not increase proportionally with the void length in days. The reason for this is that different types of multiple-day voids can be used for different types of other activities. The reason why a void length of 4 days has a low percentage of loss days is that after 64 hours after the last activities, a certain rest threshold is met for certain trainings. The difference of the impact of each of these voids has been captured in the cost index k^i_{index}. This index indicates the relative costs of a void of i days with respect to the costs of a void of 1 day, adjusted for the percentage total loss days of total void days. Equation 4.8 is then used to calculate the costs of any void in the roster. Here, c^i_v represents the cost of a void v with length i which is the product of a baseline void cost c_v and the cost index k^i_{index} for voids of length i.

$$c^i_v = c_v \cdot k^i_{index} \tag{4.8}$$

This data-driven method of expressing rostering costs has been chosen to not only construct more productive rosters for the crew rostering phase of the airline planning problem but to also capture the impact of the succeeding scheduling phases (adjustments and disruptions managements) on the eventual productiveness of rosters.

- **Bonus costs for an assigned requested pairing** - c_b - Another cost parameter that has been introduced in the model formulation of Section 4.2 is the bonus costs for assigning a pairing to a crew member that has requested this pairing. The main reason for is to slightly influence the solution process to give preference to rosters in which requested pairings are assigned. In the experiments, the sensitivity of this parameter has been tested.

- **Cost for assigning a slack roster** - c^s_r - In order to divert the solution process from a solution in which slack crew members are required to arrive at a feasible solution, the costs of assigning a roster r to a slack crew member $s \in S$ has been made very expensive in terms of costs. This is often referred to as the Big M method.

- **Penalty costs for a covered pairing** - c_p - In order to find rosters with pairings that are not yet covered by non-slack crew members and to divert the solution process from choosing pairings for non-slack crew members that are already covered by multiple other non-slack crew members, penalty costs for choosing covered pairings are introduced.

- **Cost parameters for the time-space network** - c_{VEA} and c_{BSA} - As was already explained in Section 4.3.1, a cost or distance feature can be assigned to each of the arcs in a time-space network graph. This has been done for the time-space network graphs for each crew member as well. As can be seen in Table 4.7, two of the time-space network graph cost parameters are higher than the others. The cost of a pre-assignment arc is higher since it is, therefore, more expensive to choose a path through a pre-assignment arc rather than a path through a pairing arc when having to choose between the two. The pre-assignment arcs are especially important for modeling pre-assigned activities which will be covered more extensively in Chapter 5. The cost of a base-arc is higher since it is, therefore, more expensive to choose a path through a base arc rather than a path through a pairing arc when having to choose between the two. Base arcs are undesirable since any count, and any number of subsequent base arcs determines the voids of a certain length in a roster. These voids induce loss days, which was explained in the establishment of the void cost parameters.

4.5. Model output

In the explanation of the column generation algorithm using Table 4.3, an introduction to the model output was already presented. Since an optimization problem underlies the static crew rostering model, the output of the model comes in the form of a solution to that optimization problem. In Table 4.3, the solution to the decision variables forms the solution to the crew rostering problem, which is illustrated in the bottom row of the table. The decision variables x^l_r indicate which roster r has been chosen for which crew member l. The solution to the rostering problem can be visualized as a roster in the Gantt Chart representation that has been used to represent rosters throughout this report. Figure 4.6 shows an example of such a solution visualization

Table 4.7: Explanation and commonly used values of the cost parameters in the model

Cost parameter	Cost explanation	Initial value
c_v	Unindexed costs for a void in the roster	100
c_b	Bonus costs for an assigned requested pairing	10
c_r^s	Cost for assigning a slack roster	1000000
c_p	Penalty costs for a covered pairing	10
c_{SSA}	Costs for a schedule start arc	1
c_{PSA}	Costs for a pairing start arc	1
c_{PA}	Costs for a pairing arc	1
c_{PEA}	Costs for a pairing end arc	1
c_{VSA}	Costs for a pre-assignment start arc	1
c_{VA}	Costs for a pre-assignment arc	100
c_{VEA}	Costs for a pre-assignment end arc	1
c_{BSA}	Costs for a base arc	100
c_{SEA}	Costs for a schedule end arc	1

	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN
L_001	NA	NA	NA		PA_0040	CA_004	CA_004	RP	RP	RP	RP			
L_002	NA	NA	NA	NA	NA	NA	PA_0067	AF_001	AF_001	AF_001	RP	RP	RP	
L_003	PA_0003	NA_005	NA_005	NA_005	RP	RP	RP							
L_004	NA	PA_0019	AF_001	AF_001	AF_001	RP	RP	RP						
L_005	NA	NA	NA	PA_0038	ME_003	ME_003	ME_003	ME_003	RP	RP				
L_006	PA_0001	ME_003	ME_003	ME_003	ME_003	RP	RP							
L_007	NA	NA	NA	NA	NA	PA_0060	ME_003	ME_003	ME_003	RP	RP			
L_008		PA_0013	ME_003	ME_003	ME_003	ME_003	ME_003	RP	RP	RP				
L_009	NA	PA_0017	AF_002	AF_002	AF_002	AF_002	AF_002	AF_002	RP	RP	RP	RP		
L_010	PA_0005	ME_001	ME_001	RP	RP	RP	PA_0061	CA_003	CA_003	CA_003	CA_003	RP	RP	RP

Figure 4.6: Visualization of an example solution to the static rostering problem

Table 4.8: Overview of performance metrics as additional output to the static crew rostering model

Performance metric	Explanation
$n_{non-slack}$	Number of crew members assigned to a roster [-]
n_{slack}	Number of slack crew members assigned to a roster [-]
n_{FTE}	Number of minimum FTE required to operate the roster [-]
$p_{unassigned}$	Number of pairings that are assigned to slack crew members [-]
n_q	Number of assigned pairings that have been requested [-]
$f_{q,actual/minimum}$	Surplus of assigned pairings that have been requested compared to minimum of assigned pairings that have been requested [-]
$m_{objective}$	Objective value [-]
t_{CPU}	Computation time [s]

for a set of 10 crew members.

Also, the model output can be supported by a set of metrics that illustrate the performance of the crew rostering model. The set of performance metrics chosen for the static rostering model is presented in Table 4.8.

4.6. Model verification

To verify the correct functioning of the model, this section comments on a small scale test that has been performed using the model. In the test scenario, three pairings need to be assigned to three crew members. The pairings are presented in Table 4.9 that shows the specifics of each pairing that are relevant to the problem (e.g., number of flight duty days and rest period days). The pairings are visualized in a Gantt Chart schedule example in Figure 4.7. Table 4.10 lists the crew IDs of the crew members to which the pairings can be assigned to as well as the crew IDs for three pairing-specific slack crew members.

Initialization of pairing requests and carry-in length

Each of the crew members submits a request for one pairing. The pairings that have been requested by each of the crew members are listed in Table 4.11. Pairing *PA.1* has been requested twice, which can be explained by its high request probability that is listed in Table 4.9. The possibilities for the carry-in length in this test case scenario are 0 days, 3 days and 6 days. These lengths correspond to the voids at the beginning of the schedule

Table 4.9: Test pairings

Pairing ID	**Departure time**	**Flight duty days**	**Rest period days**	**Destination**	**Request probability**
PA.1	1-01-18 0:00	3	3	*ME_003*	0.8
PA.2	4-01-18 0:00	3	3	*ME_002*	0.1
PA.3	7-01-18 0:00	3	3	*NA_005*	0.1

Table 4.10: Test crew members

Crew ID	**Slack crew ID**
L.1	*S.PA.1*
L.2	*S.PA.2*
L.3	*S.PA.3*

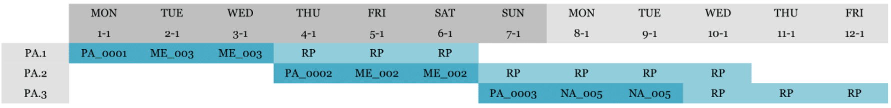


Figure 4.7: Visualization of the test pairings

Table 4.11: Test pairing requests per crew member

Pairing ID	***L.1***	***L.2***	***L.3***
PA.1	X	X	
PA.2			
PA.3			X

Table 4.12: Test carry-in length per crew member

Carry-in length	***L.1***	***L.2***	***L.3***
0	X		
3		X	
6			X

as can be seen in Figure 4.7. The carry-in length that has been randomly assigned to the crew members is listed in Table 4.12.

Solution process

The solution process for the verification test is illustrated in Table 4.13. The rows in the table correspond to the columns in the solution process. In the first iteration, the pairings have all been assigned to the corresponding slack crew members. The regular crew members are assigned a path with base arcs only. In the second iteration, columns are added for each of the crew members in which paths along pairing arcs (PA) are initiated. In the third iteration, an additional and final column is added for crew member *L.1*, and the stop criterion for adding columns is met. The optimizer then finds a solution to the ILP problem. This solution is listed in Table 4.14 and a visualization of how this solution translated to a schedule for the crew members is presented in Figure 4.8. From the figure, the allocation of carry-in pre-assignments and pairings is clear. The green-shaded pairings have been requested by the crew members.

Verification of constraints

Finally, given the solution to the problem, the constraints can be verified. This is done in Table 4.15, where the constraints from Equation 4.3, 4.4 and 4.5 are verified, respectively.

Table 4.13: Solution process of the verification test

Iteration	**Crew ID**	**Decision variable**	**Roster path**													
1	*L.1*	x_1^1	SSA_1	BSA_1	BSA_2	BSA_3	BSA_4	BSA_5	BSA_6	BSA_7	BSA_8	BSA_9	BSA_{10}	BSA_{11}	BSA_{12}	SEA_1
	L.2	x_1^2	SSA_4	BSA_4	BSA_5	BSA_6	BSA_7	BSA_8	BSA_9	BSA_{10}	BSA_{11}	BSA_{12}	SEA_{13}			
	L.3	x_1^3	SSA_6	BSA_6	BSA_8	BSA_9	BSA_{10}	BSA_{11}	BSA_{12}	SEA_1						
	S.PA.1	$x_s^{PA.1}$	SSA_1	PA_1	SEA_1											
	S.PA.1	$x_s^{PA.2}$	SSA_1	PA_2	SEA_1											
	S.PA.1	$x_s^{PA.3}$	SSA_1	PA_3	SEA_1											
2	*L.1*	x_2^1	SSA_1	PSA_1	PA_1	PEA_1	PSA_3	PA_3	PEA_3	SEA_1						
	L.2	x_2^2	SSA_4	PSA_2	PA_2	PEA_2	BSA_{10}	BSA_{11}	BSA_{12}	SEA_1						
	L.3	x_2^3	SSA_6	PSA_3	PA_3	PEA_3	SEA_1									
3	*L.1*	x_3^1	SSA_1	PSA_1	PA_1	PEA_1	BSA_7	BSA_8	BSA_9	BSA_{10}	BSA_{11}	BSA_{12}	SEA_1			

Table 4.14: Solution of the decision variables in the verification test

Decision variable	x_1^1	x_1^2	x_1^3	$x_s^{PA.1}$	$x_s^{PA.2}$	$x_s^{PA.3}$	x_2^1	x_2^2	x_2^3	x_3^1
Solution	0	0	0	0	0	0	0	1	1	1

	MON 1-1	TUE 2-1	WED 3-1	THU 4-1	FRI 5-1	SAT 6-1	SUN 7-1	MON 8-1	TUE 9-1	WED 10-1	THU 11-1	FRI 12-1
L.1	PA_0001	ME_003	ME_003	RP	RP	RP						
L.2	NA	NA	NA	PA_0002	ME_002	ME_002	RP	RP	RP	RP		
L.3	NA	NA	NA	NA	NA	NA	PA_0003	NA_005	NA_005	RP	RP	RP

Figure 4.8: Visualization of the schedule as a solution to the verification test

Table 4.15: Verification of the constraints in the verification test

Constraint				Verified
Pairing constraints	*PA.1*	$x_s^{PA.1} + x_2^1 + x_3^1$	$0+0+1=1$	yes
	PA.2	$x_s^{PA.2} + x_2^2$	$0+1=1$	yes
	PA.3	$x_s^{PA.3} + x_2^1 + x_2^3$	$0+0+1=1$	yes
Crew constraints	*L.1*	$x_1^1 + x_2^1 + x_3^1 = 1$	$0+0+1=1$	yes
	L.2	$x_1^2 + x_2^2 = 1$	$0+1=1$	yes
	L.3	$x_1^3 + x_2^3 = 1$	$0+1=1$	yes
Request constraint	*Q.1*	$x_s^{PA.1} + x_s^{PA.2} + x_s^{PA.3} + x_2^1 + x_2^3 \geq 1$	$0+0+0+1+1 \geq 1$	yes

5 Dynamic personalized rostering model

The static rostering model that has been presented in Chapter 4 fulfills the purpose of assigning pairings to crew members. However, to model the dynamic nature of the crew rostering process, another approach is required. Recall from Section 4.1.1 that the dynamic rostering model is based on the rolling roster modeling approach. This rolling approach requires a reduced set of model inputs and appends the existing schedule with a small time horizon of activities on an iterative basis. In this chapter, the dynamic rostering model that has been developed for this research is presented. This dynamic rostering model with integration of flight request evaluation is a novel construction-based crew rostering method. It allows for evaluating crew preferences in an earlier scheduling stage than is possible with current optimization approaches. The dynamic model better represents airline practices that publish an addition to the schedule on a weekly basis, rather than publishing a schedule for one month as is common in literature. The design of the dynamic rostering model is addressed in Section 5.1. Following, a mathematical representation of the dynamic scheduling environment is presented in Section 5.2. Then, the model elements will be discussed in Section 5.3. The approach to the model input in a dynamic context and the adaptations required to use the static rostering model as the input to the dynamic rostering problem are discussed in Section 5.4. Following, Section 5.5 presents and discusses the output and verification of the dynamic rostering model.

5.1. Design of the dynamic rostering model

The design of the dynamic rostering model is based on iteratively solving the static rostering model within a dynamic scheduling environment. In this environment, decisions on the allocation of requested pairings in a wider planning horizon are made by a pairing request evaluation algorithm. Adopting the rolling rostering approach, presented in Section 4.1.1 and Figure 4.1, implies that the model input is updated and the schedule is appended with a small time horizon of newly assigned activities for each time iteration. The design framework and the incorporation of pairing request evaluation algorithms in this process are discussed in Section 5.1.1 and 5.1.2, respectively.

5.1.1. Design framework for the dynamic rostering model

The dynamic rostering model used in this research has been visualized in Figure 5.1 in which the process blocks are labeled referring to the dynamic rostering model or DRM. Three elements are highlighted in the figure since these are crucial additions to the dynamic rostering model as opposed to the static rostering model. These elements will be discussed more elaborately in Section 5.3. From the figure, it is clear that the dynamic model requires an initial model input (DRM.1) at the initial state of the schedule and from then onward, two modeling loops can be identified. Firstly, the time loop serves to guide the model through instances in time at which the rostering model needs to be solved. If the maximum time iteration stop criterion (DRM.14) is not met, the model continues to solve the rostering problem for another time iteration (DRM.15). These time intervals will be further explained in Section 5.2. When in the next time iteration, the already existing model data should be adjusted for time (DRM.16), and a global database that holds information on all time iterations in terms of model input and output is updated. From this same database, new model input that is required at the new time iteration is retrieved (DRM.18) and processed (DRM.2). Secondly, the rostering loop serves to guide to model through the rostering process for each time iteration.

The rostering process for each time iteration (DRM.3 up to and including DRM.11) is then similar to the process of the static rostering model of the framework that was presented in Figure 4.2. The only exception is an important additional step that is required. This is the feasibility adjustment of the time-space network for the rostering model (DRM.4). In this step, a feasibility check is performed on the time-space network graph for each crew member based on the additional pairing assignments that have been made on top of the pairing assignments by the rostering model. These additional pairing assignments are made by the pairing evaluation process that evaluates pairing requests on a wider planning horizon than that of the rostering problem (DRM.12 and DRM.13).

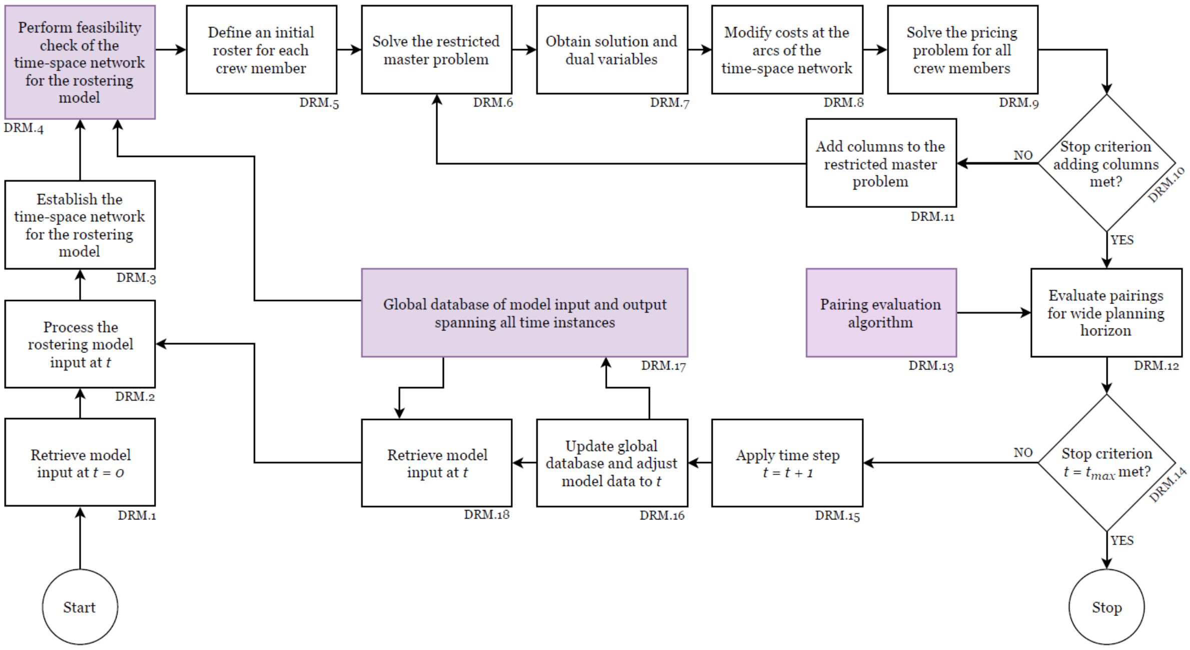


Figure 5.1: Design framework for the dynamic rostering model

5.1.2. Incorporating pairing request evaluation algorithms into the dynamic rostering model

The desired evaluation of pairing requests in a wider planning horizon is enabled by the dynamic rostering model. This evaluation takes place after the rostering problem for a certain time iteration has been solved. To further clarify this, Figure 5.2 illustrates three scenarios that capture the process of evaluating pairing requests in the dynamic rostering scenario. A rostering scenario is visualized where the difference between the separate scheduling stages is highlighted in the top bar of the figure. Time of operation indicates the moment in time that activities in the roster are executed. The published constructed schedule represents the schedule that is published for crew members, and that is not being adapted by the rostering problem anymore. The crew rostering planning horizon represents the time horizon for which the rostering problem is solved. The wider planning horizon represents the time horizon that is further in the future, with respect to the rostering problem that is currently being solved. The pairing request evaluation planning horizon represents the time horizon for which pairing requests can be submitted by crew members which will then be evaluated by the dynamic rostering model. The Figure is split up into three different states of a dynamic rostering scenario; state (a), state (b) and state (c). Also, the figure shows three transition arrows; transition (I), transition (II) and transition (III). These illustrate the transition of corresponding example activities between state (b) and state (c). Both the model states and the different types of transitions will be discussed below.

Three model states:

- **State (a) - Rostering problem for t_0 solved, pairing requests submitted but not yet evaluated:**
  - Example model state at time iteration t_0 after step DRM.10 in the dynamic rostering model framework

  - The rostering problem for t_0 has been solved
  - A solution has been found for the assignment of pairings PA.17, PA.18, PA.19 and PA.20
  - Each of the crew members submitted a set of pairing requests for the weeks 8, 9, 10, 11 and 12 in the wider planning horizon

- **State (b) - Rostering problem for t_0 solved, pairing requests evaluated:**
  - Example model state after step DRM.12 and DRM.13 in the dynamic rostering model framework
  - In addition to the rostering problem that is solved, the pairing requests that have been submitted by the crew members in t_0 are all evaluated
  - The pairings PA.1, PA.2, PA.3 and PA.4 are scheduled to be operated in the upcoming week.

- **State (c) - Time step applied, evaluated pairing requests translated to pre-assigned activities, rostering problem for t_1 solved, pairing requests submitted but not yet evaluated:**
  - Example model state after step DRM.10 in the dynamic rostering model framework
  - The rostering problem for t_1 has been solved
  - A solution has been found for the assignment of pairings PA.21, PA.22, PA.23 and PA.24 to the crew members
  - Each of the crew members has submitted a set of pairing requests for the weeks 8, 9, 10, 11 and 12 in the wider planning horizon
  - The pairings PA.1, PA.2, PA.3 and PA.4 have been operated and are not part of the still to be operated schedule anymore.

Three model transitions:

- **Transition (I) - From upcoming operation to operated**
  - For the time iteration t_0 in model state (b), the pairings PA.1, PA.2, PA.3 and PA.4 in week 1 are considered to commence on the upcoming Monday at 00:00 a.m.
  - For the time iteration t_1 in model state (c), it is assumed that the pairings PA.1, PA.2, PA.3, and PA.4 are currently being operated. From this perspective, the pairings PA.5, PA.6, PA.7 and PA.8 in week 1 are considered to commence on the upcoming Monday at 00:00 a.m.
  - This operation transition clarifies the eventual operation of each week of pairings that have been assigned by the rostering model.

- **Transition (II) - From solution to the rostering problem to addition to the existing schedule**
  - For the time iteration t_0 in model state (b), the pairings PA.17, PA.18, PA.19 and PA.20 that commence in week 5 have been assigned by the rostering model in this time iteration.
  - For the time iteration t_1 in model state (c), the pairings PA.17, PA.18, PA.19 and PA.20 that were assigned to the crew members in the previous time iteration t_0, are now added to the existing schedule.
  - The start of the pairings has shifted a week in time, from week 5 in t_0 to week 4 in t_1.
  - This schedule addition transition clarifies the rolling effect of the dynamic rostering model.

- **Transition (III) - From granted pairing request to pre-assigned pairing**
  - For the time iteration t_0 in model state (b), the pairing requests for the pairings PA.31, PA.32, PA.34, and PA.36 have been granted.
  - For the time iteration t_1 in model state (c), the granted pairing requests are transformed into pre-assigned pairings while also having shifted a week in time.
  - For example, the start of pairing PA.32 has shifted a week in time, from week 8 in t_0 to week 7 in t_1.
  - This pairing request transition clarifies the transformation of granted pairing requests to pre-assigned pairings that eventually need to be considered in the crew rostering planning horizon of week 5 and 6.

Incorporating pairing request evaluation algorithms in the scenario that has just been explained occurs in the change of states between state (a) and state (b) of Figure 5.2. In the process of pairing request evaluation, the job of the pairing request evaluation algorithm that was introduced in the framework of Figure 5.1 is to decide on either granting or not granting specific requests. The algorithms that have been designed to serve this purpose are introduced in the next Chapter 6. From the explanation of a dynamic scenario by discussing different types of states, however, it is already clear that the evaluation of pairing requests and their transformation into pre-assigned pairings has an impact on the solution process of the dynamic rostering problem.

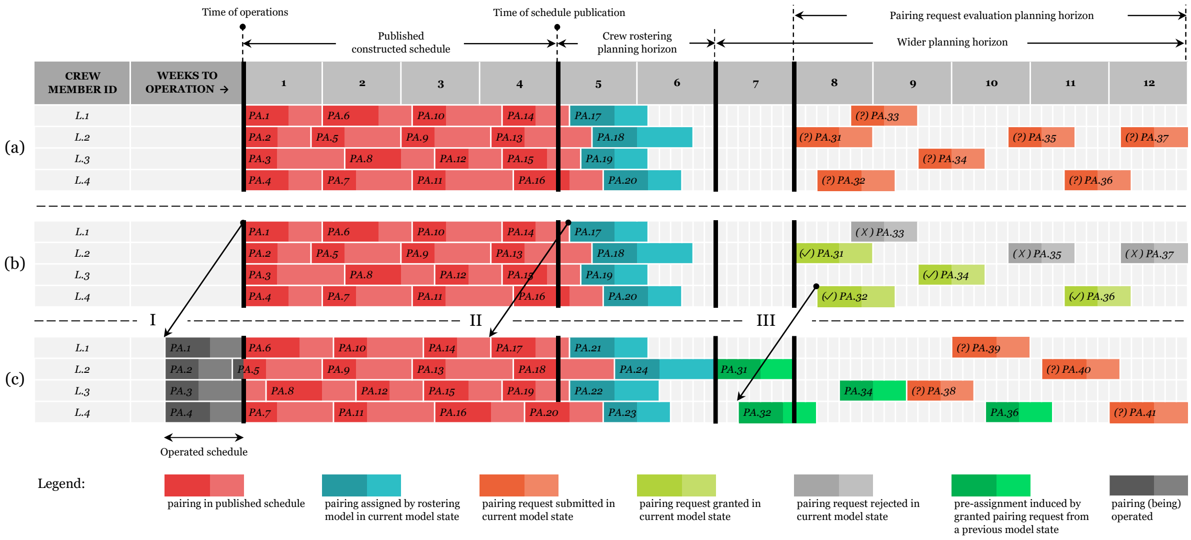


Figure 5.2: Example of three model states (state (a), state (b) and state (c)) and three model transitions (transition (I), transition (II) and transition (III)) in a dynamic rostering modeling scenario

5.2. Mathematical representation of the dynamic rostering problem

The dynamic personalized rostering optimization problem is presented in this section, that can be solved to construct a schedule that meets the requirements. The problem is described by the dynamic modeling parameters, the required sets, parameters, decision variables, cost function, objective function and constraints that are consecutively presented below. It should be noted that many of the sets and parameters in the model are time-dependent (t) since they vary over time in the planning horizon.

Dynamic modeling parameters

- t_0 : initial time iteration
- t_{max} : maximum time iteration
- H : planning horizon spanning $[t_0, t_{max}]$
- t : time iteration $\in H$

Boundary conditions

- Boundary conditions for $t = t_0$:
  - $C(t_0)$: set of carry-in pre-assignments initiated at t_0
- Boundary conditions for $t_0 < t \leq t_{max}$:
  - $C(t-1)$: set of carry-in pre-assignments existing at the end of the previous time iteration $t-1$ as a result of assigning pairings in the crew rostering problem
  - $Q(t-1)$: set of pre-assigned activities existing at the end of the previous time iteration $t-1$ as a result of granting pairing requests with the pairing request evaluation algorithm

Sets

- $L(t)$: set of crew members of the crew type considered at time iteration t
- $S(t)$: set of slack crew members of the crew type considered at time iteration t
- $P(t)$: set of pairings to be covered by the crew type considered at time iteration t
- $R_l(t)$: set of personalized rosters for crew member $l \in L(t)$ at time iteration t
- $R_s(t)$: set of personalized rosters for slack crew member $s \in S(t)$ at time iteration t

Cost parameters

- c_b : bonus cost for assigning a requested pairing in the roster
- c_v^i : cost of void in the roster with a void length of i number of days
- $c_r^l(t)$: cost of roster $r \in R_l(t)$ for crew member $l \in L(t)$
- $c_r^s(t)$: cost of roster $r \in R_s(t)$ for slack crew member $s \in S(t)$

Other parameters

- $(n_v^i)_r^l(t)$: number of voids v in personalized roster $r \in R_l(t)$ for crew member $l \in L(t)$ with a void length of i number of days
- $q_r^l(t)$: number of requested and assigned pairings in personalized roster $r \in R_l(t)$ for crew member $l \in L(t)$
- q : minimum desired number of requested and assigned pairings in the schedule
- $e_p^l(t) = \begin{cases} 1 : \text{if pairing } p \in P(t) \text{ is requested by crew member } l \in L(t) \text{ at time iteration } t \\ 0 : \text{otherwise} \end{cases}$
- $e_p^{r,l}(t) = \begin{cases} 1 : \text{if pairing } p \in P(t) \text{ is chosen for roster } r \in R_l(t) \text{ for } l \in L(t) \text{ at time iteration } t \\ 0 : \text{otherwize} \end{cases}$

Decision variables

- $x_r^l(t) = \begin{cases} 1 : \text{if personalized roster } r \in R_l\text{(t) is chosen for crew member } l \in L(t) \text{ at time iteration } t \\ 0 : \text{otherwise} \end{cases}$
- $x_r^s(t) = \begin{cases} 1 : \text{if personalized roster } r \in R_s \text{ is chosen for slack crew member } s \in S(t) \text{ at time iteration } t \\ 0 : \text{otherwise} \end{cases}$

Cost function

$$c_r^l(t) = \left(\sum_{i=1}^{j} (n_v^i)_r^l(t) \cdot c_v^i \right) + \left(q_r^l(t) \cdot c_b \right) \tag{5.1}$$

Objective function

$$\text{Minimize} \sum_{l \in L(t)} \sum_{r \in R_l(t)} c_r^l(t) \cdot x_r^l(t) + \sum_{s \in S(t)} \sum_{r \in R_s(t)} + c_r^s \cdot x_r^s(t) \tag{5.2}$$

Constraints

$$\sum_{l \in L(t)} \sum_{r \in R_l(t)} e_p^{r,l}(t) \cdot x_r^l(t) = 1 \quad , \quad \forall p \in P(t) \tag{5.3}$$

$$\sum_{r \in R_l(t)} x_r^l(t) = 1 \quad , \quad \forall l \in L(t) \tag{5.4}$$

$$\sum_{l \in L(t)} \sum_{r \in R_l(t)} \sum_{p \in P(t)} e_p^{r,l}(t) \cdot e_p^l(t) \cdot x_r^l(t) \geq q \tag{5.5}$$

$$x_r^l(t) \in \{0,1\} \quad , \quad \forall l \in L(t), \forall r \in R_l(t) \tag{5.6}$$

$$x_r^s(t) \in \{0,1\} \quad , \quad \forall s \in S(t), \forall r \in R_s(t) \tag{5.7}$$

5.3. Elements of the dynamic rostering model

Referring to the dynamic rostering model framework of Figure 5.1 and the newly introduced dynamic features of the rostering model, some key elements of the model can be identified that are discussed in the next sections. Firstly, the elements of the static rostering model that have been adapted to serve the dynamic rostering model are discussed. Secondly, the elements to model the dynamic component of the dynamic rostering model are discussed. Thirdly, the global database that holds the model input and output spanning the full planning horizon H. Finally, the feasibility check of the time-space network is discussed. The pairing evaluation algorithms (DRM.13) as a model element are discussed in Chapter 6.

5.3.1. Elements of the static rostering model to be adapted

The elements of the static rostering model that were discussed in Chapter 4 are the time-space network representation, the shortest path algorithm, integer linear programming, and the column generation algorithm. All of these features are still used in the dynamic rostering model. The time-space network representation is still used for ensuring feasible sequencing of activities in terms of time and space. An extension to the time-space network representation is the added feasibility check that is required to check for feasibility and continuity across different time iterations. The shortest path algorithm is still used to generate minimum-cost columns or rosters for the crew members. Integer linear programming is still the method that is used for solving the optimization problem that was presented in Section 5.2. Although the building blocks of the model are dependent on time iteration t in the planning horizon H, a separate optimization problem using ILP and the column generation algorithm can be solved for each time iteration.

5.3.2. Elements to model the dynamic component

The dynamic modeling parameters that were presented in 5.2 set the boundaries for the planning horizon in which the dynamic rostering model is solved. To be able to grasp the shifts in time in the model, below is an example of how the different time iterations can be interpreted:

- Suppose that the current moment in time is the initial time iteration t_0
- This suggests that the model state can be represented by the state $y(t_0)$
- Assuming that the rostering model is solved for week 5 and 6 in the planning horizon, this suggests that the rostering model is solved for the weeks with identifiers w_0^5 and w_0^6. Here, the subscript 0 refers to the time iteration t_0 and the superscripts 5 and 6 refer to the 5th and 6th week in the future with respect to the current time iteration t_0
- When the rostering model is solved, the pairing requests have been evaluated, and the transitions that were described in Section 5.1.2 are performed, the next iteration in the time loop is started, and the state of the model can be reformulated.
- The state of the model can then be represented by the state $y(t_1)$. The weekly shift ensures that the activities from week w_0^5 in state $y(t_0)$ are now shifted to w_1^4 in state $y(t_1)$.

Regardless of the time iteration, the rostering model is thus solved at each time iteration for the weeks with identifiers w_i^5 and w_i^6. Moreover, after solving the rostering problem at each time iteration, week w_i^5 is added to the published schedule and week w_i^1 has been operated. The above framework is used in the simulations that are executed for the dynamic rostering model.

5.3.3. Global database of model input and output spanning all time iterations

Referring to element DRM.17, to keep track of the right model for each time iteration in the simulation of the dynamic rostering model, a database has been constructed that is initialized at t_0 and updated for the decision that are made in the model. This database holds information about, among other things, the following aspects:

- Pairing information on each unique pairing
- Crew information and up-to-date assignment restriction information for each crew member
- Pairing request information on each unique pairing request
- Performance metrics for each constructed roster and for the schedule as a whole in each iteration

The data has been organized such that the decisions can be derived back to the time iteration and the state of the model in which they have been made.

5.3.4. Feasibility check of the time-space network

Referring to element DRM.4, pre-assigned activities that are induced by pairing requests eventually end up in the crew rostering planning horizon. For this planning horizon, a time-space network graph has been constructed for each crew member. However, the pre-assigned activities induced by the pairing request evaluators, are not yet considered in this time-space network. The most important reason for this is that the time-space network approach is deliberately limited to the two-week crew rostering planning horizon of week 5 and 6 to limit the size of the time-space network. The following approach has been followed as a feasibility check to update the time-space network graph for a specific crew member and make it comply with all the pre-assigned activities in the crew rostering planning horizon:

1. **Step 1** - Translate the pairings assigned in a previous iteration to this crew member to a corresponding carry-in pre-assignment length in week 5 of the current iteration

2. **Step 2** - Update the time-space network for this carry-in pre-assignment length by removing the schedule start arcs (SSAs) that are not applicable to this crew member. Recall that the schedule start arcs are formulated between the schedule start node (SSN) and all the midnight nodes (MNNs) in the schedule. The remaining schedule start arc is thus drawn from the schedule start node to the midnight node from which point onward the crew member is available for operation again.

3. **Step 3** - Check whether there are pairings assigned by the pairing request evaluator that are currently being scheduled in the crew rostering planning horizon. If yes, set this pairing as the pre-assigned pairing under consideration and proceed to step 4.

4. **Step 4** - For the pre-assigned pairing under consideration that is being definitively assigned by the crew rostering model in this iteration, *remove* the following arcs from the time-space network graph:

   - Remove all other pairing start arcs (PSAs) from the network graph that start on the same day as the pre-assigned pairing under consideration
   - Remove all the pairing start arcs (PSAs) from the network graph for which the corresponding pairings overlaps. In other words, remove those pairing start arcs for which the corresponding pairing starts on a day prior to the start of the pre-assigned pairing under consideration but that *ends* after the start of pre-assigned pairing under consideration.
   - Remove all the base arcs (BSAs) from the network graph that start on the same day as the pre-assigned pairing under consideration
   - Remove all the pre-assignment arcs (VSAs) from the network graph that start on the same day as the pre-assigned pairing under consideration

The feasibility check serves two purposes. Firstly, it removes the possibility for choosing a path in the network through those pairings that cannot be operated anymore by a crew member under consideration, due to other pairings that have been pre-assignment to this crew member. Secondly, from the node at which the pre-assigned pairing under consideration starts, it removes all options but the option through the pre-assignment under consideration in finding a path from this node onward to the schedule sink node (SIN). This forces the shortest path algorithm to explicitly include the pre-assigned pairing in each possible roster for this crew member.

5.4. Model input

As was clear from the research design that was presented in Section 3.8, the input to the model is driven by historical airline data. Table 5.1 presents the different types of input to the model, along with an indication of what data has been used to retrieve this input. This section will discuss the methods used for translating historical airline data to model input for the dynamic rostering model.

5.4.1. Model input initialized at initial time

Referring to element DRM.1, the input to the rostering model at the initial time iteration t_0 in the rostering simulation is similar to the modeling of rostering input in the static rostering model. Some additional assumptions are given below.

Table 5.1: Overview of the dynamic rostering model input sets and the airline data that has driven these sets

Input set	Historical airline data
Set of crew members at each time iteration t	Crew demand data
Set of pairings at each time iteration t	Flight schedule data
Set of pairing requests at each time iteration t	Pairing request data
Set of pre-assigned activities at each time iteration t	Operational crew scheduling data
Other modeling parameters at each time iteration t	Operational crew scheduling data

Modeling crew members and slack crew members

For the modeling of crew members and slack crew members in the dynamic rostering model, the initial assumption was made that crew demand for each week can vary since predicted absenteeism and propagation of operational disruptions are considered in establishing the net crew demand. The time component in the set of crew members in the model $L(t)$, however, is still useful to be able to model known longer-term absenteeism (e.g., longer-term illnesses). In this model, the decision has been made to take a non-changing set of crew members as input to the model. The set of crew members $L(t_0)$ is similar to the set of crew members $L(t_1)$. This is necessary for the goal of the dynamic rostering model to continuously append a new set of assigned activities to the already existing published roster for each crew member.

Modeling pairings

For the modeling of pairings, assumption 7 in Section 3.3 suggested that the pairing schedule that needs to be operated is the same for each week of operations. The pairing schedule of Appendix A with pairing start days spanning from Monday up until Sunday, repeats itself every week. Therefore, there are 71 pairings to be assigned at each time-instance of the dynamic rostering model simulation. In each time-instance, not only the pairings to be assigned should be considered but also the pairings in the future weeks should be tracked due to the pairing request evaluation process. A unique identifier has, therefore, been assigned to all the pairings in the model, keeping track of absolute information on the pairing start day and assignments status throughout the simulation. At each time iteration t_i, the pairings starting in week w_i^5 are assigned in the rostering model, while the other pairings in week w_i^6 up to and including week w_i^{12} are considered in the evaluation of pairing requests. This adds up to 8 weeks with 71 departing pairings $(8 \cdot 71)$ that need to be enumerated in each time iteration, for which only the first 71 pairings being assigned by the rostering model. For the whole dynamic rostering model simulation, this adds up to 8 weeks with 71 pairings with departing pairings in which 12 time transitions have taken place in the schedule that all add an additional week of pairings to the end of the roster ((8 weeks in the initial time iteration + 12 additional weeks) $\cdot$ 71 pairings = 1420 pairings).

Modeling cost parameters

The cost parameters are modeled similarly to the cost parameters in the static rostering model. The bonus costs for an assigned requested pairing c_b are zero in the dynamic rostering model since it is assumed that the pairing request evaluator has already made the value judgment whether or not to assign a requested pairing in the model.

5.4.2. Model input at each time iteration

Referring to element DRM.18, the set of pairing requests and pre-assigned activities is update for each time iteration in the dynamic rostering model.

Modeling pairing requests

If a set of pairing requests has been evaluated at the initial time iteration t_0, these requested pairings will re-occur in the next iteration t_1 as pre-assigned pairings. The following assumptions have been made in the modeling of pairing requests in the dynamic rostering model:

- A batch of pairing requests B_i^{week} has been introduced. Such a batch of pairing requests holds the collection of pairing requests for all the crew members in the schedule at a certain time iteration t_i for planning horizon week number $week$.
- In each iteration t_i, pairings in planning horizon week number 8, 9, 10, 11 and 12 can be requested by crew members. Therefore, for each iteration t_i, the following batches of pairing requests are fed as input to the model; B_i^8, B_i^9, B_i^{10}, B_i^{11}, B_i^{12}.

- In each iteration t_i, pairings in planning horizon week number 5, 6 and 7 cannot be requested by crew members anymore.
- All pairings that are requested in a certain iteration t_i are evaluated in that same iteration. The granted requests are transformed into pre-assigned pairings; the non-granted requests are not considered by the model anymore.
- The requesting of pairings can be considered a random process and has, therefore, been modeled as such. The following two parameters that were introduced in:
  - **Pairing request probability** - $pa_{p_{request}}$ - Probability with which a certain pairing is requested by a crew member when being prompted to request a pairing from the set of pairings in a given planning horizon.
  - **Number of pairing requests per crew member** - $n_{l_{request}}$ - Number of pairings within the to be requested planning horizon (week w_i^8 up to and including week w_i^{12}) that are requested by each crew member in each iteration t_i.

  The pairing request probability $pa_{p_{request}}$ serves as the probability distribution with which a random pairing is chosen from the pairings that can be requested in an iteration t_i. A total number of $n_{l_{request}}$ requests are randomly chosen for each crew member. In the case that duplicate pairing requests exist in the requested pairings of a crew member, the process is repeated. The requests that have been submitted in each time iteration are added to the corresponding batch of pairing requests B_i^{week}. This batch holds the pairing request information necessary for the pairing request evaluator to evaluate the request; the combination of crew member ID and pairing ID.

Modeling pre-assigned activities
Recall from the model input for the static rostering model that overlap of assigned activities from a previous planning horizon needs to be taken into account. These carry-in pre-assignments are modeled for the dynamic model as well. However, the approach to modeling the carry-in assignment for the dynamic rostering model is different, which is explained below. The other type of pre-assigned activities are the granted requested pairings that have been transformed into pre-assigned activities.

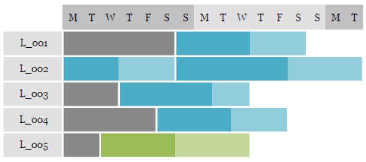
- **Carry-in pre-assignment length** - $l_{carry-in_{length}}$ - Length of the overlap of activities for a specific crew member from a preceding planning horizon into the planning horizon that is currently considered in the crew rostering problem. This can be explicitly retrieved from the activities assigned to a crew member in a previous dynamic modeling iteration. This process was introduced in Section 5.1.2 where transition (II) explained the transition from a solution to a rostering problem to addition to the existing schedule. In terms of the time frame, the solved rostering week w_0^5 in t_0 will become week w_1^4 in t_1. The pairings that start in week w_1^4 can have a residue in week w_1^5. This residue in a certain roster is the carry-in pre-assignment length for a specific crew member.

- **Pre-assigned granted pairing requests** - Pairing requests that have been granted for a crew member are transformed into pre-assigned activities for a crew member. This process causes activities to be assigned starting from an initially empty schedule in the weeks w_0^8 up to and including w_0^{12}. Instead of having to formulate a large time-space network covering all weeks in this wider planning horizon, an algorithm has been used to keep track of feasible assignment in this wider planning horizon with an increasing number of pre-assigned activities. For each crew member, the pre-assigned activities are translated to a list of days at which the crew member is not available for the assignment of potential other pre-assigned pairings. This list serves as the input to a feasibility check for further assignments to this crew member. This feasibility check ensures that no overlapping assignments are impossible without having to capture the full schedule in a very large time-space network graph. This approach will be further explained in the explanation of the pairing request evaluators in Chapter 6.

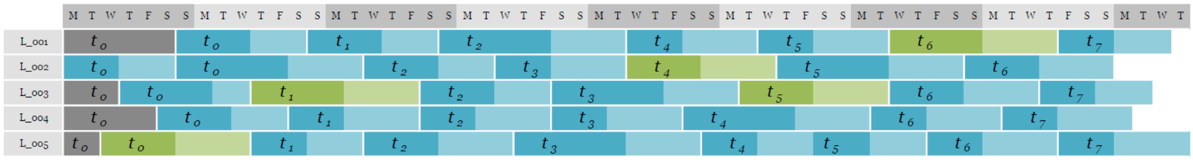
5.5. Model output

As well as in the static rostering problem, an optimization problem underlies the dynamic crew rostering model. Therefore, the output of the model comes in the form of a solution to that optimization problem. For the dynamic rostering model, a solution is found for each iteration in the dynamic modeling environment. Each iteration optimizes for the scheduling weeks w_i^5 and w_i^6 and the solution for each iteration is appended to an overall solution. The decision variables for each optimization problem can be translated into a gradually constructed schedule that spans the full planning horizon. To illustrate this, Figure 5.3 compares the dynamic rostering model output to the static rostering model output. In Figure 5.3a, a visualization of an example solution for two weeks is presented for the static rostering model. In Figure 5.3b, a visualization of an example solution to the dynamic rostering model is presented. In the pairings, the time iteration t_i is indicated, which represents the time iteration in which each specific pairing has been assigned. Note from Figure 5.3b that it presents pairings that have been assigned up to week 7. A full simulation would yield the rolling schedule for 13 weeks.

Furthermore, the model output can be supported by a set of metrics that illustrate the performance of the crew rostering model. The set of performance metrics chosen for the dynamic rostering model similar to the set of performance metrics for the dynamics rostering model that was presented in Table 4.8. For the dynamic model, however, these metrics can be measured for each time iteration in the process.



(a)



(b)

Figure 5.3: (a) Schematic representation of the output of the static rostering model, (b) Schematic representation of the output of the dynamic rostering model

The model output has been verified with a small test case that is similar to the verification of the static rostering model that was presented in 4.6. Figure shows the output of this test case. As can be seen, iteration t_0 considers the initialized carry-in pre-assignments and the following iterations consider the existing pairings as pre-assigned activities.

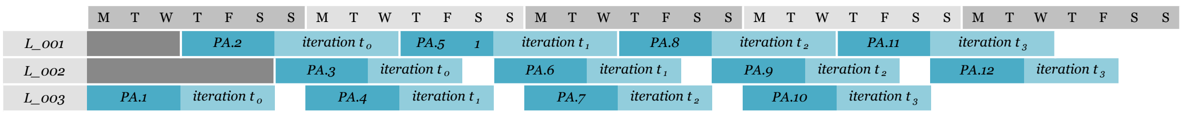


Figure 5.4: Visualization of the schedule as a solution to the verification test

6

Evaluation of pairing requests

In the design framework of the dynamic rostering model that was presented in Figure 5.1 in Section 5.1.1, one of the key elements is the pairing request evaluation algorithm. This algorithm enables pairing requests to be evaluated in a wider planning horizon than the crew rostering planning horizon. In this research, three pairing request evaluation algorithms have been developed. The random-based pairing request evaluation algorithm is presented in Section 6.2, the rule-based pairing request evaluation algorithm is presented in Section 6.3 and the classification-based pairing request evaluation algorithm is presented in Section 6.4. Before discussing the pairing request evaluation algorithms, assumptions on the pairing request policy that is handled are presented in Section 6.1.

6.1. Assumptions on pairing request policy

Pairing requests as a means for airline crew preference management have not been considered in literature. To clarify the pairing request process, this section provides the pairing request policy that is considered in this research. Although some of its content has already been introduced throughout this report, the following assumptions have been made in the policy that is used within the scope of this research:

1. Each full-time based crew member is eligible for a number of granted pairing requests on a yearly basis. This number, q_{max}, is fixed for the full workforce.
2. To check whether a crew member is eligible for the honoring of a pairing request, a counter is kept for each crew member.
    - For each granted pairing request, a crew member's counter decreases with the value 1.
    - For each crew member, the counter will be reset to its yearly maximum q_{max} the at a random yearly recurring moment in the year.
3. Crew members can submit an unlimited amount of pairing requests during the year.
4. A pairing request that is submitted by a crew member cannot be withdrawn.
5. All pairing requests that have been submitted in a certain week will be evaluated in the same week.
6. All pairing requests that have been submitted in a certain week are categorized based on their corresponding planning horizon week. After that, the requests are sorted for priority based on either one of the two approaches:
    - First come first served approach (FCFS)
    - Seniority approach
7. The pairing request evaluation algorithm evaluates the pairing requests for each batch.
8. When a pairing request is granted for a crew member, the pairing activity is pre-assigned to this crew member. This pre-assignment is binding for the rostering problem and cannot be withdrawn anymore.
9. When a pairing request is not granted for a crew member, this crew member can submit a pairing request for this specific pairing again at a later time.

6.2. Random-based pairing request evaluation algorithm

The pairing request evaluation algorithm is executed after a solution to the rostering problem for a time iteration has been found, as was shown in the framework for the dynamic rostering problem that was presented in Figure 5.1 in Section 5.1.1, The first algorithm that is covered in this Section is the random-based pairing request evaluation algorithm. Figure 6.1 presents the flowchart of the random-based algorithm that consists of the following phases:

- **Pre-processing requests (RA.1, RA.2, RA.3)** - The requests are sorted into batches that refer to the corresponding week of the start day of the pairing. Following, the first pairing from the first batch is selected to be evaluated.
- **Feasibility check (RA.4, RA.5)** - The feasibility check is twofold. Firstly, it ensures that CLA agreements are met in terms of equitability across the workforce by checking the request budget for a crew member. Secondly, it ensures that the potential granting of a request does not conflict with the already existing pre-assigned activities in the roster.
- **Random-based request evaluation (RA.6)** - The decision that is taken on granting or declining the request is achieved on a random basis. Equation 6.1 presents the probability distribution that is commonly handled throughout this research. A pseudo-random number X is generated within this probability distribution that is the deciding factor for a pairing request to be granted ($X = 1$) or rejected ($X = 0$).

$$\Pr(X = x) = \begin{cases} p_{grant} & x = 1 \\ 1 - p_{grant} & x = 0 \end{cases} \quad \text{where } (x = 1 : \text{grant}) \text{ and } (x = 0 : \text{reject}) \tag{6.1}$$

- **Post-processing requests (RA.9, RA.10)** - Based on the decision that was made on granting or declining the request, post-processing takes place. In the case of a granted request, the pairing activity is transformed into a pre-assignment. In both the cases of granting or declining a request, the global database should be updated. This is especially important since the process of granting the following requests in the iterative process can be considered as construction-based without backtracking.
- **Iteration (RA.11, RA.12, RA.13, RA.14)** - Once post-processing of a request has taken place, the next request in the batch is selected. If all the requests in the batch are evaluated, the next batch is selected. This process continues until all batches of requests within a time instance have been evaluated.

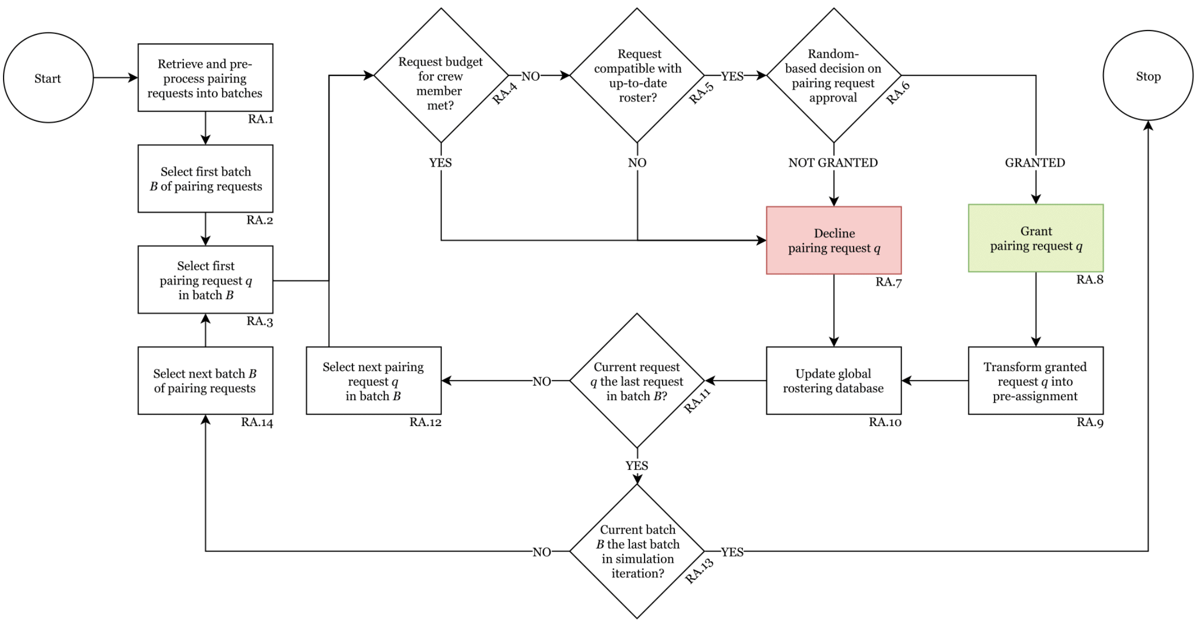


Figure 6.1: Flowchart of the random pairing request evaluation algorithm

6.3. Rule-based pairing request evaluation

The rule-based pairing request evaluation algorithm has been inspired by current practices at an airline at which pairing requests are evaluated manually. In this process, schedulers tend to stick to a set of unwritten rules to make decisions that, by experience, make up for a feasible roster. According to schedulers, one of the most influential metrics in the decision on granting a request is the length of the void in the schedule that is induced by a pre-assignment due to granting a request. The likelihood of the induced void to be covered by another pairing, in a later scheduling stage, is the value judgment that is made in the evaluation of all pairing requests. However, this value judgment is not supported by real-time data but rather by scheduling experience. Moreover, this value judgment is usually based on a tangible planning horizon covering only a few weeks. Roughly estimating the likelihood of being able to cover a one-week void in a roster is a tangible task for a scheduler, but this task is significantly harder for a three-week void. The rule-based pairing request evaluation algorithm is designed to capture this heuristic of void length evaluation. Figure 6.1 presents the flowchart of the random-based algorithm that consists of the following phases:

- **Pre-processing requests (RU.1, RU.2, RU.3)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.
- **Feasibility check (RU.4, RU.5)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.
- **First rule-based evaluation (RU.6 and RU.7)** - The first rule that is checked is whether or not a certain threshold of granted pairing request in the current batch has been met. If not, the remaining pairing requests in the batch are not considered and a short-cut is taken to continue to the next batch represented by the red arrows in the flowchart of Figure 6.2.
- **Second rule-based evaluation (RU.8 and RU.9)** - The second rule that is checked is to measure the length of the void induced by potentially granting the pairing request. The condition for void length induced by potential granting of pairing request q is presented in Equation 6.2. Here, v_{length} represents the measure void length, $p_{length,median}$ represents the median pairing length of the pairings in the pairing schedule (P) and i represents an integer value within a bounded set. This implies that the void length induced by request q should be an integer multiple of $p_{length,median}$ in order for request q to be granted.
$$v_{length} = p_{length,median} \cdot i \quad , \quad i \in [0, 1, \dots, n] \tag{6.2}$$
- **Post-processing requests (RU.10, RU.11, RU.12, RU.13)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.
- **Iteration (RU.14, RU.15, RU.16, RU.17)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.

6.4. Classification-based pairing request evaluation

The classification-based pairing request evaluation algorithm has been developed to incorporate a feedback mechanism in the request process. For decisions to be made within a wider planning horizon, it is useful to understand what has eventually become of these decisions. Consider the following example; a pairing request q has been granted in iteration t_0 for batch B_0^8; the pairing corresponding to this request will be part of the to be assigned set of pairings in iteration t_3. In that iteration, the rosters will be constructed of which the requested pairing is part of. This is also the moment at which cost can be determined for the roster and the schedule in which the pairing of request q exists. This method is experimented with based on the notion that a pairing request can be evaluated based on the projected costs of either granting or declining the request. Figure 6.3 presents the flowchart of the random-based algorithm that consists of the following phases:

- **Pre-processing requests (CL.1, CL.2, CL.3)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.
- **Feasibility check (CL.4, CL.5)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.
- **Classification evaluation model (CL.6, CL.7** - Explained throughout the following sections 6.4.1 up to and including 6.4.4.

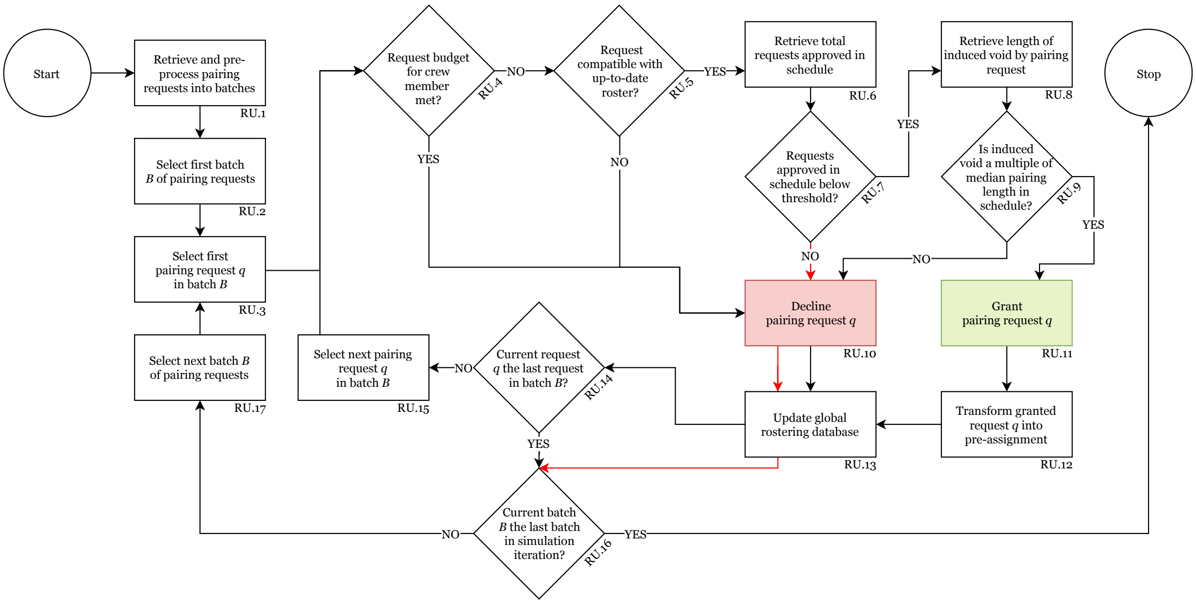


Figure 6.2: Flowchart of the rule-based pairing request evaluation algorithm

- **Post-processing requests (CL.8, CL.9, CL.10, CL.11)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.
- **Iteration (CL.12, CL.13, CL.14, CL.15)** - Similar to the random-based pairing request evaluation algorithm of Section 6.2.

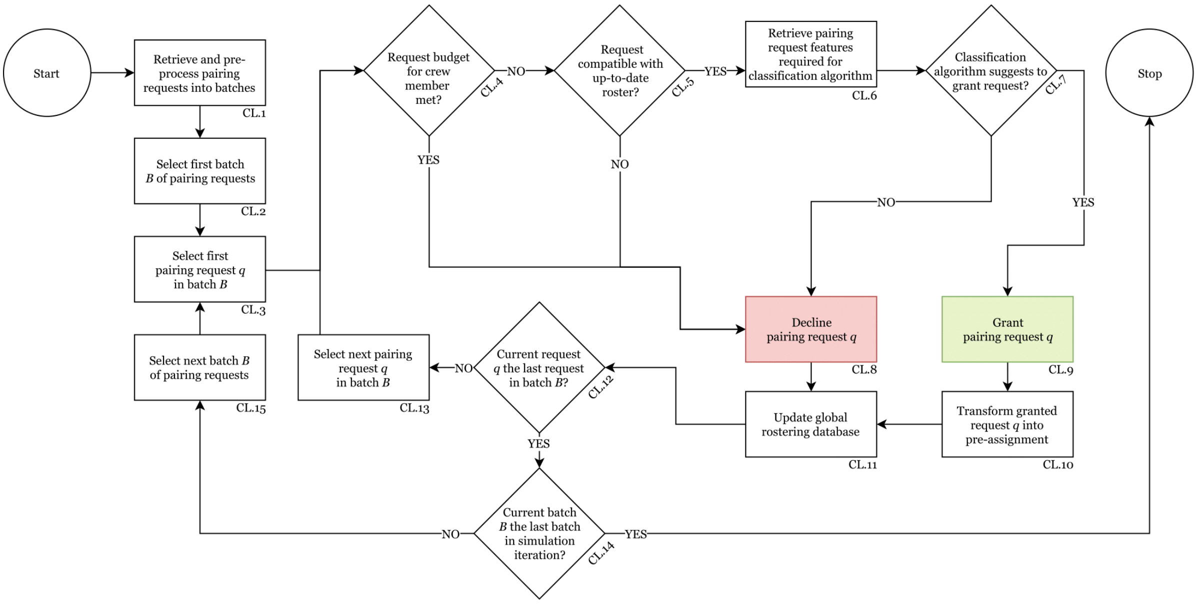


Figure 6.3: Flowchart of the classification-based pairing request evaluation algorithm

6.4.1. Overview of classification model

The classification model that has been developed for this research has been designed to either grant or reject a request. Classification is a *supervised* machine learning method. New observations are classified or categorized based on a training set of data that contains observation features of which the category is known. Supervised learning, in turn, is a subclass of machine learning in which a function is induced or learned from a set of example feature-response pairs that can map a set of features to a response. This process is explained in the functional flowchart of Figure 6.4. The classification model consists of a training part and a prediction part. Firstly, in the training part, data on submitted and already evaluated pairing requests is collected. From this data, certain features from each request can be extracted as well as the corresponding response. In classification terminology, *features* represent the properties of an *observation instance* of training data and the *response* represents the corresponding classification category or *class* of this observation instance. In the context of this research, pairing requests represent *observation instances* that comprise pairing request properties represented by *features* and a resulting evaluation on the granting or declining of the request represented by the *response*. All the observation instances with their corresponding features and responses are fed as input to a machine learning algorithm which is design to induce the function that can map features to a corresponding response. Secondly, in the prediction part, test data is available on which only the features can be extracted. The response is unknown which is for the classifier to predict. The classifier is the algorithm that predicts the response for a new observation instance with a given set of features. In the flowchart of the classification-based pairing request evaluation algorithm, this classifier is represented by the decision block CL.7.

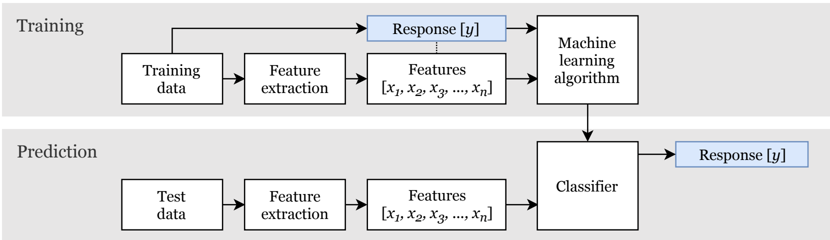


Figure 6.4: Flowchart of the classification model

6.4.2. Selection of pairing request features

Table 6.1 presents the features that have been selected for the classification of pairing requests. The features involve a timing component within the simulation (features (1) and (2)), properties of the pairing itself (feature (3) and (4)) and a environment assessment of the pairing requests that have already been assigned at the moment at which the request evaluation is made (feature (5)). These features have been selected for having an expected impact on the pairing request evaluation. Formulating all the feature possibilities allows for the feature set to hold binary values only. For a new observation instance, each of the 44 feature possibilities is assigned a binary value. The reason for choosing this approach is the nature of the information that the features hold. For example, granting a request for a certain pairing in iteration 1 could have more resemblance to this same decision in iteration 8 than it would have to iteration 2. The same goes for the other feature types in the table. From a collected set of pairing requests, features *for each observation instance* be extracted and translated into the form presented in Equation 6.3. For the training process, the corresponding response request decision is retrieved as well and translated into the form presented in Equation 6.4. The possibilities for this response are binary as well, with 1 representing granting of the request and 0 representing declining of the pairing request.

$$X = [x_1, x_2, x_3, x_4, \dots, x_{44}] \tag{6.3}$$

$$y = [y] \tag{6.4}$$

Table 6.1: Overview of the names and possibilities of the pairing request features

Feature name	**Feature possibilities**												
(1) iteration number (i in t_i)	0	1	2	3	4	5	6	7	8	9	10	11	12
(2) batch number (wk in B_i^{wk})	8	9	10	11	12								
(3) pairing length (days)	5	6	7	8	9	10	11						
(4) pairing departure day	MON	TUE	WED	THU	FRI	SAT	SUN						
(5) count of requests granted at decision moment of request evaluation	0	1	2	3	4	5	6	7	8	9	10	≥11	

6.4.3. Generation of pairing request training data

The dynamic crew rostering model that has been developed in this research has been used as a means to generate pairing request training data. The following steps have been taken to achieve this:

- Define the simulation environment and model settings in which the pairing request data is collected
- In the simulation test environment, collect data on each pairing request and evaluate the request using the random-based request evaluation algorithm. For the generation of request data, the probability distribution that was presented in Equation 6.1 is used. There is a relatively low probability that a pairing is granted. Therefore, the rostering problem is more likely to provide a solution without having to rely on slack variables. This allows for the generation of pairing request data with outcomes that are desirable in a crew rostering problem.
- For each pairing request, evaluate the costs of the corresponding roster $c_{q,roster}$ and the overall schedule $c_{q,schedule}$ in the iteration in which the pairing that was requested is scheduled in week w_i^5. For this to be valid, the following condition should be met:

$$\text{Condition for cost evaluation of request in iteration } t_i\text{: } (q_{iteration} + 3) + (q_{batch} - 8) = i \tag{6.5}$$

- Analyze and filter the collected data to extract examples of correctly identified granted requests and correctly identified rejected requests. The interpretation on the correct identification on either granting or declining a request is based on the confusion matrix which is used in the evaluation of a classification algorithm, presented in Figure 6.5. It is assumed that correctly labeled granted requests (true positive) induce low costs for both the roster and the schedule and that correctly labeled rejected requests (true negative) induce high costs for roster and schedule. The valuation for either low or high costs has been made by analyzing the data and determining a clear cut-off for either low or high costs.

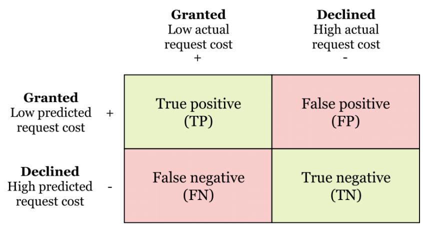


Figure 6.5: Confusion matrix

6.4.4. Selecting a machine learning algorithm and training the classifier

The machine learning algorithm chosen for this problem is the Bernoulli Naive Bayes learning algorithm that is based on the Bayes theorem of Equation 6.6 where $(x_1, \ldots, x_n)$ represents the feature vector and y the response.

$$P(y|x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n|y)}{P(x_1, \ldots, x_n)} \tag{6.6}$$

The naivety of the algorithm lies in the assumption of conditional independence between every pair of features. This independence factor has been accounted for in the selection of the features that were presented in Section 6.4.2. The reason for choosing this algorithm is its high performance with relatively small sets of training data, along with its low computation time for training. This makes it suitable for implementation in crew rostering, in which the training data could be updated for retraining without having to compromise for large training times. The Bernoulli Naive Bayes learning algorithm is one of the established Naive Bayes methods that assumes for its features to be binary-valued (Bernoulli, Boolean), which is the case in this problem. This algorithm explicitly penalizes the non-occurrence of a feature i that is an indicator for a response y. The implementation of the Bernoulli Naive Bayes has been established using the Scikit-learn Python module in which this algorithm is integrated for medium-scale supervised learning problems (Pedregosa et al., 2011).

$$P(x_i|y) = P(i|y)x_i + (1 - P(i|y))(1 - x_i) \tag{6.7}$$

7

Experiments and results

The experiments that have been carried out in this research are designed to answer the research subquestions that were defined in Section 2.2. This chapter covers the explanation of these experiments as well as the results and a discussion of the experiments. Section 7.1 covers the experiment in which the effect of pairing requests on the required number of FTEs in a roster is tested. Section 7.2 covers the experiments that evaluate and compare the different pairing request evaluation algorithms in this research. Furthermore, the validation of the results in the context of the case study airline are covered in Section 7.3.

7.1. Experiment A: Effects of pairing requests on the number of required FTEs

In an early stage of the research, the hypothesis was formulated that pre-assigned activities in the roster decrease the solution space and, possibly, impact the feasibility of the crew rostering problem. The granting of pairing requests, as a cause of these pre-assigned activities has been considered the main reason for this to be further investigated and quantified. The static rostering model has been used as a well-established means of a evaluation to measure the effect of increasing the pairing requests in the problem that need to be granted at the minimum.

7.1.1. Experiment A: Setup

A certain threshold or a pattern of saturation is expected for the workforce being able to cover all the pairings while also, as an airline scheduling department, being able to grant pairing requests. To explore this threshold, the constraint has been investigated that ensures a minimum of requested pairings within the set of assigned pairings in a roster (Equation 4.5 in Chapter 4 and recalled below in Equation 7.1).

$$\sum_{l \in L} \sum_{r \in R_l} \sum_{p \in P} e_p^{r,l} \cdot e_p^l \cdot x_r^l \geq q \tag{7.1}$$

If this minimum cannot be met with the established workforce in the model for a given scenario, it is assumed that additional crew members are required in the case that the minimum q is binding. This supports the notion that the effect of pairing request management could also be measured financially as additional FTEs impose higher crew costs. The experiments are focused on the ability of the workforce being able to cover all the pairings, while also meeting the minimum of requested pairings in the schedule as a whole. Recall that the full workforce works on a full-time basis in this research, meaning that the set of crew members directly translated to full-time equivalent or FTEs. Additional crew members required for an operable schedule are also measured in FTEs. This unit of measurement is a convenient and established way for crew scheduling practices to express the human resources for the workforce that is required for an operable schedule.

In the experiments on pairing request saturation, the static rostering model has been used. As was extensively discussed in Chapter 4, multiple sets of model input have to be provided to this model. The net crew demand required for the schedule to be operable is based on available historical airline data of net crew demand. This data corresponds to the pairing schedule in Appendix A. The net crew demand for the week to which the pairing schedule corresponds is 69.67 FTEs. Since the assumption has been made that all crew members work on a full-time basis and that route days and rest period days have lengths of integer values, the baseline net crew demand for this experiment is a workforce of 70 FTEs.

Two model input types have a stochastic nature. For the full workforce, a set of pairing request has been generated based on the request probability $p_{request}$. Also, the carry-in pre-assignment length is randomly distributed across the workforce, since there is no earlier schedule available to induce it from. These stochastic factors cause the rostering problem to find a different solution for each instance that the rostering model is solved. The modeling environment for experiment A is the static rostering model, covering one week of pairings. Following from a sensitivity analysis that is presented in Section 8.1, the provided set of crew members to the model has a size of 72 FTEs. For the scenario in this experiment, the minimum number of requested pairings to be assigned is varied. The scenario is executed 20 times for each of the possible values for q in the range [minimum of q, maximum of q]. The model settings that have been used are presented in Appendix C.

To interpret the results it is important to understand how the number of required FTEs along the vertical axes of the graphs is established. In the rostering problem, a solution is found for the decision variables for the rosters of crew members (x_l^r) and slack crew members (x_s^r). If the solution for a slack crew member decision variable is non-zero, this means that the roster for this slack crew member holds a pairing that is not assigned to one of the crew members in the set L. In such a situation, the rostering problem has not found a way to assign all the pairings to non-slack crew members. If all solutions to the slack crew member decision variables are zero, this means that the available set of crew members provided to the model was sufficient to cover the pairings while also meeting the other constraints. These situations are formulated below in Equation 7.2

$$n_{FTE} = \begin{cases} n_L + n_{unassigned} & \text{if } \sum_{s\in S, r\in R_s} x_l^s > 0 \quad \text{(i.e., slack crew members required)} \\ \sum_{l\in L, r\in R_l} x_l^r + n_{unassigned} & \text{if } \sum_{s\in S, r\in R_s} x_l^s = 0 \quad \text{(i.e., no slack crew members required)} \end{cases} \tag{7.2}$$

7.1.2. Experiment A: Results

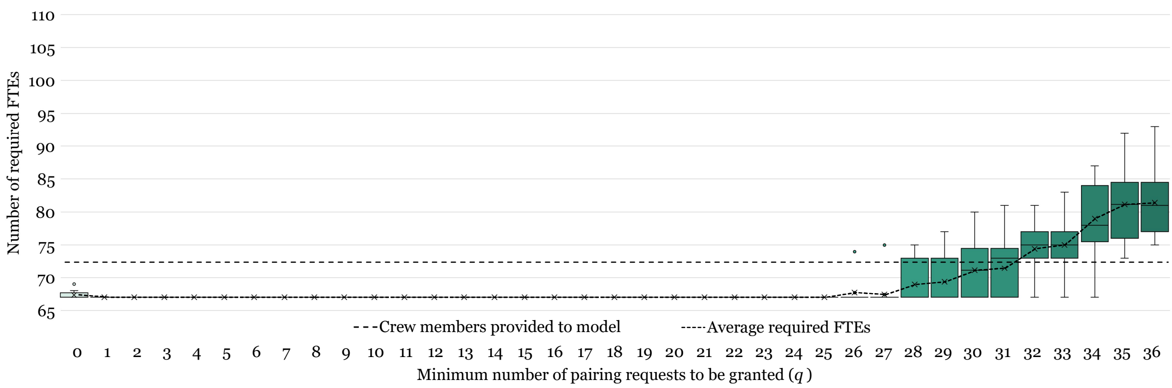


Figure 7.1: Results of experiment A (1/2): the effects of pairing requests on the number of required FTEs with size of L = 72, size of P = 71, step size of varying q = 1 and n = 20

Figure 7.1 shows the results on the number of required FTEs when varying q between a value of 0 and 36 along the horizontal axis. For each of the values of q, 20 experiments have been performed of which the results are presented in the box plots that correspond to each value of q. In addition to these box plots, two dashed lines have been plotted. The horizontal line indicates the level of FTEs that has been provided to the model (i.e., 72 FTEs) and the other line shows the trend of the averages of the box plots. Along the vertical axis, the resulting number of required FTEs (n_{FTE}) is represented. To show the development of n_{FTE} for higher values of q, Figure 7.2 shows the results on the number of required FTEs when varying q between a value of 30 and 50 along the horizontal axis. The set-up of this figure is similar to the set-up of Figure 7.1.

7.1.3. Experiment A: Discussion of results

From Figure 7.1 and 7.2, it is clear that from a certain level of q onwards, there is an upward trend in the number of FTEs that is required for feasible assignment of the pairings in a roster. In Figure 7.1, the 72 crew

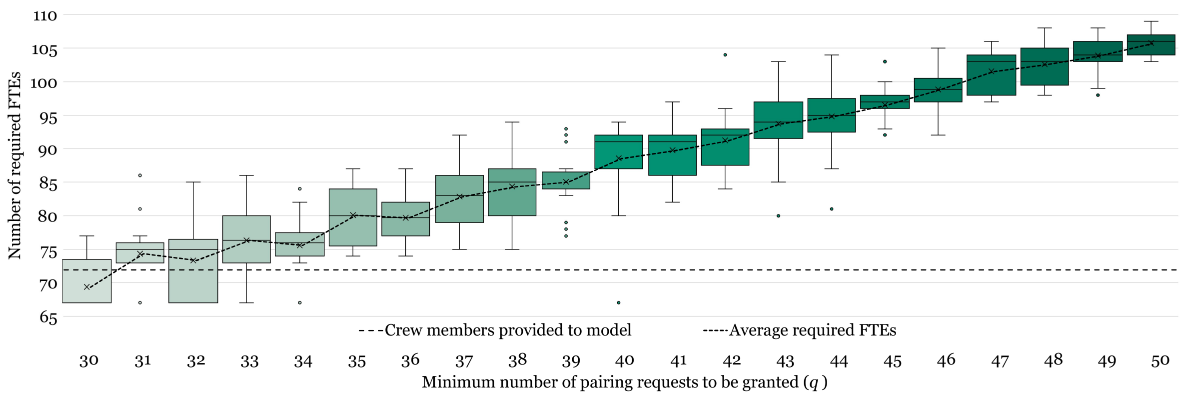


Figure 7.2: Results of experiment A (2/2): the effects of pairing requests on the number of required FTEs with size of L = 72, size of P = 71, step size of varying q = 1 and n = 20

members provided to the model are sufficient for coverage of the pairings up to a value of q of 31. In Figure 7.2, this is the case up to a value of q of 30. The difference between these measures lies in the variability of the solution that is prone to the stochastic nature of the model input. If 30.5 pairing requests in the roster are granted, this implies that just under 43% of the 71 pairings in total in this scenario are both requested and granted without having to resort to additional FTEs. Two assumptions especially limit the validity of this scenario. Firstly, the static rostering model uses an optimization approach for granting pairing requests. This is driven by the constraint of Equation 7.1. The optimization problem starts with a clean slate and builds rosters from scratch without having to consider pre-assigned activities in the rosters. In an airline practice with pairing requests that need to be evaluated in a wider planning horizon, this situation does not hold. Secondly, the assumption is made that a sufficient number of pairing requests are submitted by crew members to test all levels of q. After the threshold of 30 granted pairing requests is reached, the linear upward trend in Figure 7.2 shows that $(105.75 - 74.35)/19 = 1.65$ additional FTEs are required for each additional granted request on average. This forms a critical incentive for airlines to grant pairing requests in an effective and efficient manner.

The results of this experiment clearly indicate that granting pairing requests in the roster decreases the solution space and impact the feasibility of the crew rostering problem. In addition, one of the questions in this research was as follows: "How could crew preference management in a crew rostering simulation environment be modeled to identify and measure the (financial) effect of crew preference management on the crew rostering problem?". This experiment addresses this question and measures the effect of enforcement of pairing request granting to the number of required FTEs. This number can be considered as the resources that an airline. Translating this to a financial or monetary value is airline specific but a higher demand for FTEs induces higher costs for an airline.

7.2. Experiment B: Performance of pairing request evaluation algorithms

The algorithms that have been developed for the evaluation of pairing requests have been tested in terms of performance in the dynamic rostering model. The set-up and results of the random-based, rule-based and classification-based pairing request algorithms are presented and discussed in the following sections.

7.2.1. Experiment B: Setup

The pairing request evaluation algorithms are integrated in the dynamic rostering model. Performance can, therefore, be measured over time in the scheduling process simulation. To investigate the variability, each pairing request evaluation algorithm has been tested 50 times over the full dynamic simulation scenario (i.e., for each of the time iterations t_i in the planning horizon H). Results that correspond to each iteration have been recorded. The remaining model settings that have been used are presented in Appendix C. For each iteration, the required number of FTEs (n_{FTE}) has been measured as well as the number of granted pairing requests in the roster n_q.

7.2.2. Experiment B: Results random-based pairing request evaluation algorithm

Recall that the random-based pairing request evaluation algorithm is based on a feasibility check of the pairing request, after which a random-based decision is made based on granting probability p_{grant}. In the sensitivity analysis that has been presented in Section 8.2.1, an appropriate parameter for p_{grant} has been determined to be 0.04. Figure 7.3 shows the results for the number of required FTEs and Figure 7.4 shows the results for the number of granted pairing requests in the dynamic rostering model with random-based pairing request evaluation. The results of the 50 simulations are presented in the box plots that correspond to each time iteration in the simulation. In addition to these box plots, dashed lines have been plotted. In Figure 7.3, the average number of required FTEs and the crew members that have been provided to the model are plotted. In Figure 7.4, the average number of granted pairing requests is plotted.

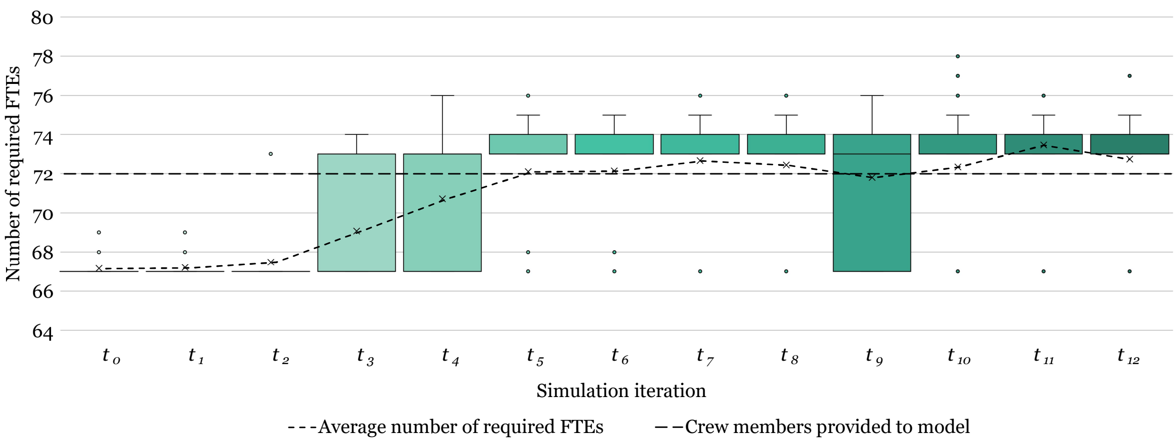


Figure 7.3: Results of experiment B: number of required FTEs in the dynamic rostering model with integration of the random-based pairing request evaluation algorithm with size of $L = 72$, size of $P = 71$ and $n = 50$

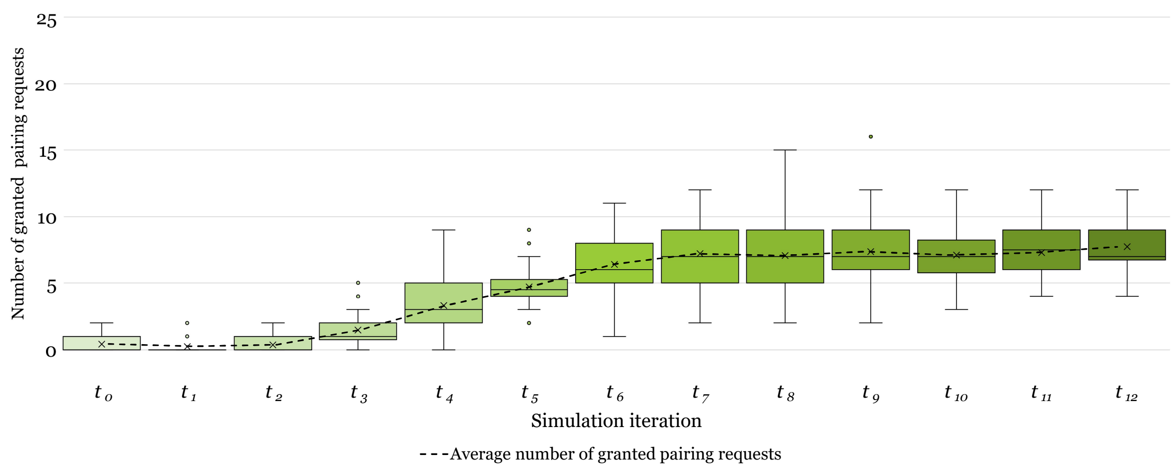


Figure 7.4: Results of experiment B: number of granted pairing requests in the dynamic rostering model with integration of the random-based pairing request evaluation algorithm with size of $L = 72$, size of $P = 71$ and $n = 50$

7.2.3. Experiment B: Discussion random-based pairing request evaluation algorithm

An important remark to make is the upward trend starting at time iteration t_2 in both Figure 7.3 and 7.4. In Figure 7.4, it can be seen that for the time iterations t_0, t_1 and t_2 no pairing requests have been evaluated by the pairing request evaluation algorithms. This is due to the planning horizon for submitting pairing requests in the dynamic model, that commences eight weeks to operation which is separated three weeks from the crew rostering planning horizon. In time iteration t_3, the roster is created for the week that first allowed for the submission of pairing requests. From time iteration t_7 onwards, these initialization effects are not present anymore since pairing requests have been allowed to be submitted during all the weeks of the planning horizon for submitting requests, for pairings that commence in the rosters that are solved in time iterations $\geq t_7$. This remark explains the initialization effect. Note that this also holds for the results of the other two pairing request evaluation algorithms. Based on the results for iterations $\geq t_7$, the average number of required FTEs is 72.57 and the average number of granted pairing requests is 7.31. A main drawback of the random-based pairing request evaluation algorithm is that the probability for granting feasible pairing requests should always be set up in accordance with a given or a predicted number of submitted pairing requests. However, what the performance of this algorithm does show is that with appropriately selected parameters a random-algorithm has the potential to automate pairing request evaluation process.

7.2.4. Experiment B: Results rule-based pairing request evaluation algorithm

The rule-based pairing request evaluation algorithm has been inspired by current airline practices. With this algorithm, the assessment for granting or rejecting a pairing request relied on assessing the length of the induced void in the roster in case of granting the request. If the length is a multiple of the median pairing length in the schedule (+/- a certain deviation), the pairing request is granted. In the sensitivity analysis that has been presented in Section 8.2.3, an appropriate percentage of deviation from the standard median pairing length rule has been determined to be 1%. Figure 7.5 shows the results for the number of required FTEs and Figure 7.6 shows the results for the number of granted pairing requests in the dynamic rostering model with rule-based pairing request evaluation. The results of the 50 simulations are presented in the box plots that correspond to each time iteration in the simulation. In addition to these box plots, dashed lines have been plotted. In Figure 7.5, the average number of required FTEs and the crew members that have been provided to the model are plotted. In Figure 7.6, the average number of granted pairing requests is plotted.

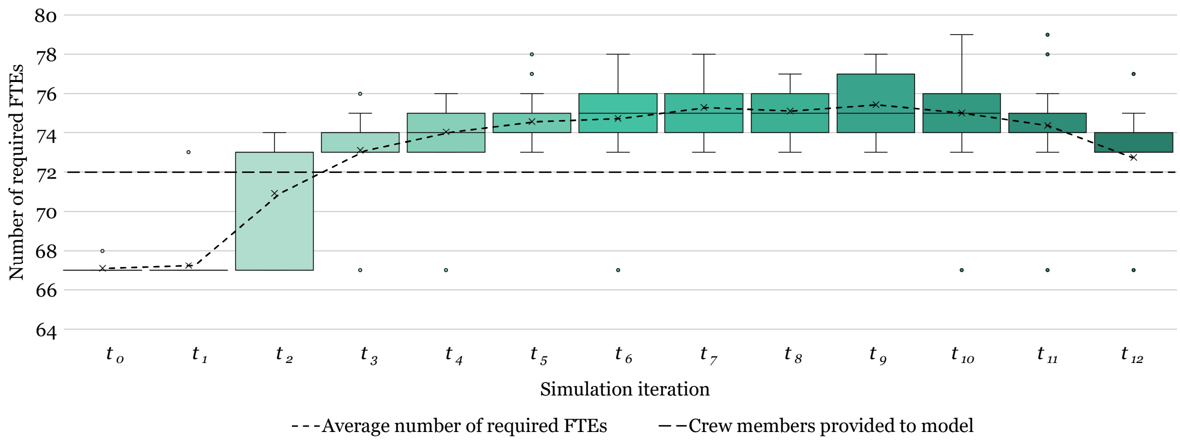


Figure 7.5: Results of experiment B: number of required FTEs in the dynamic rostering model with integration of the rule-based pairing request evaluation algorithm with size of $L = 72$, size of $P = 71$ and $n = 50$

7.2.5. Experiment B: Discussion rule-based pairing request evaluation algorithm

As can be seen in Figure 7.5, the level of crew members provided to the model is exceeded when using this form of the rule-based evaluation algorithm. In the sensitivity analysis, it is clarified that the 1% deviation results in a rule with which too many pairing requests are granted. Based on the results for iterations $\geq t_7$, the average number of required FTEs is 74.67 and the average number of granted pairing requests is 16.02. Restricting the rule(s) in the rule-based algorithm could further be investigated. The algorithms shows promising results in terms of the level of granted pairing requests with 2.67 additional required FTEs on average. Although the variation in the results of the granted pairing requests reached ranges higher than 10 for t_6, $t_1 0$ and $t_1 1$, this variation does not result in high variations of number of required FTEs for these time iteration. Therefore, the sensitivity of reducing the number of granted pairing requests should be further investigated to reduced the number of required FTEs to meet the level of crew members provided to the model. The downward trend in later simulations ($\geq t_8$) can be due to the fact that because of too many granted pairing requests in early time iterations, the feasibility check allows for less pairing requests to be

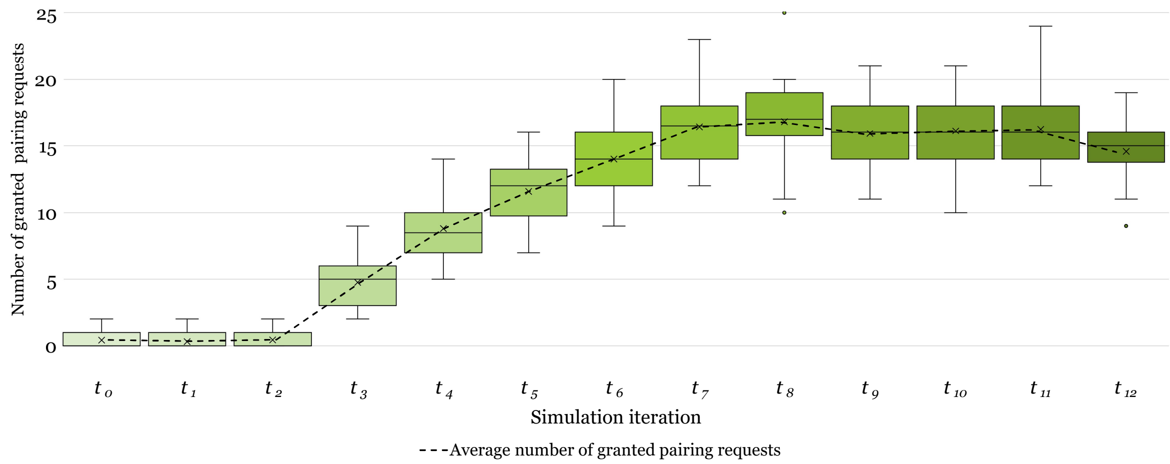


Figure 7.6: Results of experiment B: number of granted pairing requests in the dynamic rostering model with integration of the rule-based pairing request evaluation algorithm with size of $L = 72$, size of $P = 71$ and $n = 50$

assessed by the rule-based evaluation. To improve the performance of the rule-based pairing request evaluation algorithm, two suggestions are made:

- Exploring an enumeration method - The median pairing length rule could be extended by enumerating the full set or a partial set of the possibilities with which to 'fill' the void that is induced by granting the pairing request. Search heuristics could be used to find these possibilities. A better assessment could be made based on relating the granting of requests to the level of possibilities that still exist to efficiently fill the void with activities in the roster.

- Hybridizing with a random-based approach - Although the number of FTEs is exceeded in this experiment with the rule-based pairing request algorithm, the relatively high number of granted pairing requests is an incentive for exploring the effects of applying a correction factor in addition to the rule in this experiment. Since the random-based algorithm is straightforward in use, this approach could be added to a rule-based approach to limit the granting of pairing requests on a stochastic basis.

7.2.6. Experiment B: Results classification-based pairing request evaluation algorithm

Recall that the classification-based pairing request evaluation algorithm is designed to use a feedback mechanism by training a classification model with examples of eventual roster costs of pairing requests. In order to assess the pairings, features of a pairing request are collected and the classification model that is trained with a training data set that holds these features, gives a response on whether or not to grant a request. In the sensitivity analysis that has been presented in Section 8.2.4, an appropriate set of features has been selected for this research. This set of features holds the possibilities for the length of a pairing. Combinations with other types of features such as the departure day and the number of pairing requests already granted did not show promising results at the moment. Figure 7.7 shows the results for the number of required FTEs and Figure 7.8 shows the results for the number of granted pairing requests in the dynamic rostering model with classification-based pairing request evaluation. The results of the 50 simulations are presented in the box plots that correspond to each time iteration in the simulation. In addition to these box plots, dashed lines have been plotted. In Figure 7.7, the average number of required FTEs and the crew members that have been provided to the model are plotted. In Figure 7.8, the average number of granted pairing requests is plotted.

7.2.7. Experiment B: Discussion classification-based pairing request evaluation algorithm

It is clear from Figure 7.7 that the number of required FTEs does not exceed the level of crew members provided to the model in the majority of the cases. Referring to Equation 7.2, the results for n_{FTE} show that the model is at the verge of the decision whether or not to add slack crew members to the problem solution. This is an indicator of desirable result, as this indicates that the resources in the form of FTEs are being used

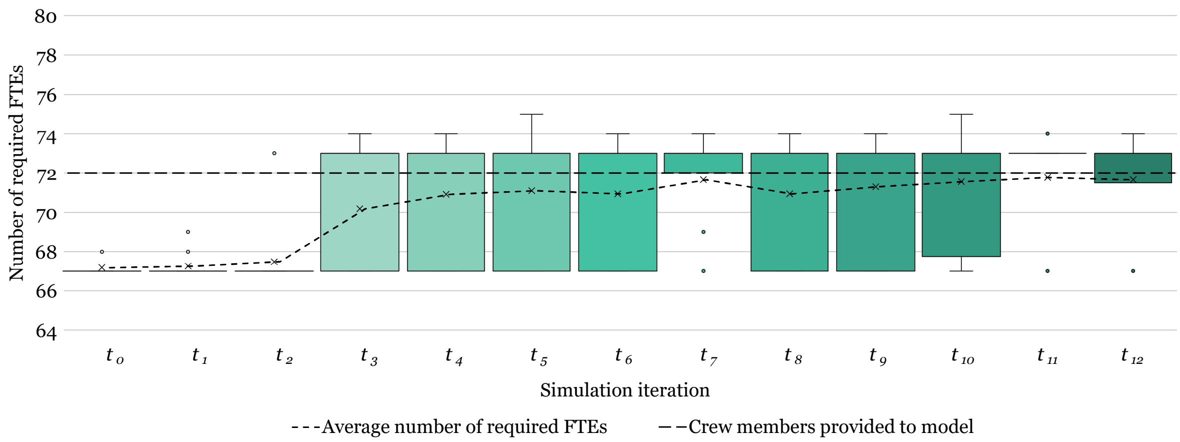


Figure 7.7: Results of experiment B: number of required FTEs in the dynamic rostering model with integration of the classification-based pairing request evaluation algorithm with size of $L = 72$, size of $P = 71$ and $n = 50$

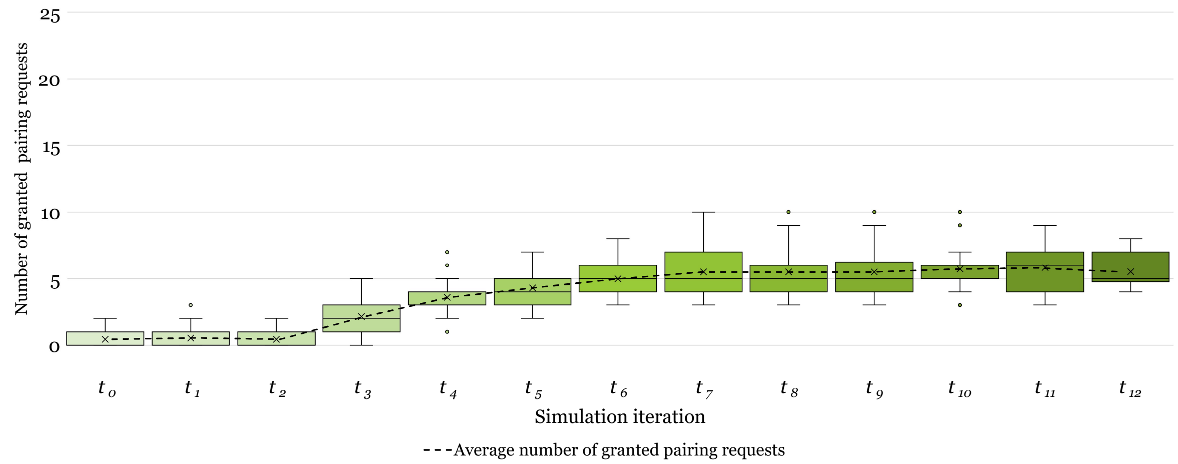


Figure 7.8: Results of experiment B: number of required FTEs in the dynamic rostering model with integration of the classification-based pairing request evaluation algorithm with size of $L = 72$, size of $P = 71$ and $n = 50$

efficiently. Based on the results for iterations $\geq t_7$, the average number of required FTEs is 71.48 and the average number of granted pairing requests is 5.59. This number of granted requests is lower than the granted pairing requests that were granted on average by the random-based and rule-based pairing request evaluation algorithm. However, the fact that the provided set of crew members is not exceeded on average, is a promising result for the classification-based pairing request evaluation algorithm. It can be considered as an indicator that the predicted cost-effectiveness by applying a classification model is an effective method for regulating the number of granted pairing requests. However, this is not backed by a quantitative analysis on the classification model itself in this research. Also, the establishment of the training data should further be investigated.

7.3. Validation of results

In this section, the results that have been presented throughout this chapter are compared to the airline practices at the major European airline that was introduced in the design of this research in Chapter 3. After this comparison, a discussion on the validity of the results within the project scope and within airline practices is presented.

By analyzing historical pairing request data from the case study airline, the average number of submitted pairing requests per FTE per week (i.e., n_y) was established at $n_y = 0.426$. From the total number of pairing requests, a fraction of 0.52 was granted. When eliminating the pairing requests that were indicated as infeasible, this fraction is 0.57 which has served as an input to the sensitivity analysis in Chapter 8.5. For a size of the workforce of 70 FTEs, the average number of total pairing requests that are granted in a week are 15.51. In comparison, Table 7.1 shows the average n_{FTE}, n_q and CPU time for each algorithm for iterations $\geq t_7$. Figure 7.9 shows the comparison of the number of FTEs in each of the algorithms compared to the crew members that have been provided to the model (i.e. the required number of crew members for the case study airline). Figure 7.10 shows a similar comparison for the number of granted pairing requests. Here, it can be seen that although model parameters in the random-based have been driven by data from the case-study airline, the rule-based algorithm is the only algorithm that approximates the average level of granted pairing requests of the case study airline. This, however, comes at the cost of having the highest value for n_{FTE} as presented in Table 7.1. Figure 7.11 shows the comparison of CPU time for each iteration when implementing each of the models. It can be stated that the solutions processes when implementing each of the algorithms is practical in use for both modelling purposes and potential use by schedulers in integrated software. This is especially due to the small planning horizon for the crew rostering model and due to assumptions in the project scope.

Table 7.1: Average of n_{FTE}, n_q and CPU time in the iterations $\geq t_7$ for each of algorithms in the dynamic rostering model and the case-study airline

	Random-based	**Rule-based**	**Classification-based**	**Case study airline**
Average n_{FTE}	72.57	74.67	**71.48**	72.00
Average n_q	7.31	**16.02**	5.59	15.51
Average CPU time [s]	16.96	**15.06**	17.35	-

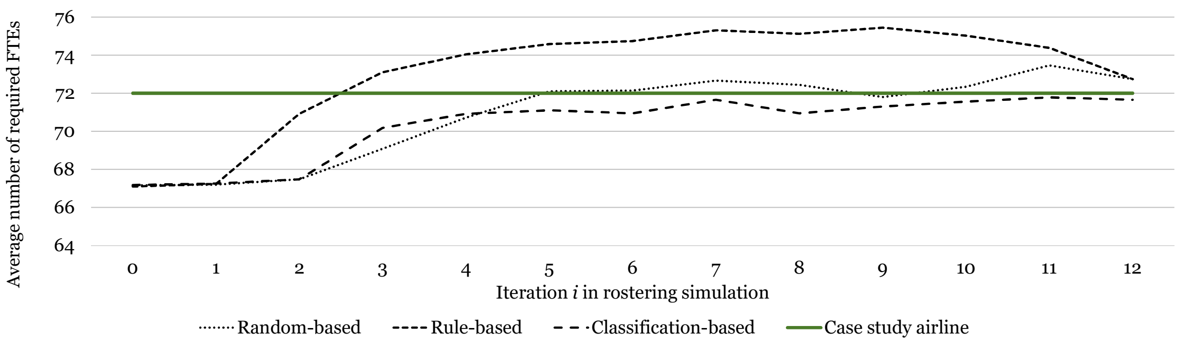


Figure 7.9: Comparison of the results for the average number of required FTEs for the random-based, rule-based and classification-based algorithm, compared to the required FTEs at the case study airline

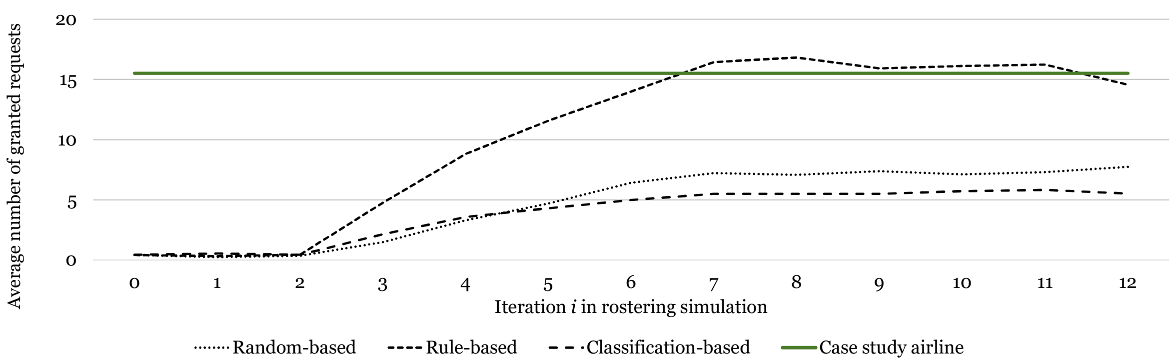


Figure 7.10: Comparison of the results for the average number of granted pairing requests for the random-based, rule-based and classification-based algorithm, compared to the average number of granted pairing requests at the case study airline

Validity of the results within the project scope

The scope of this research project comprised the following four commitments; developing a method to model the dynamic nature of the problem, developing a method to model the evaluation of pairing requests

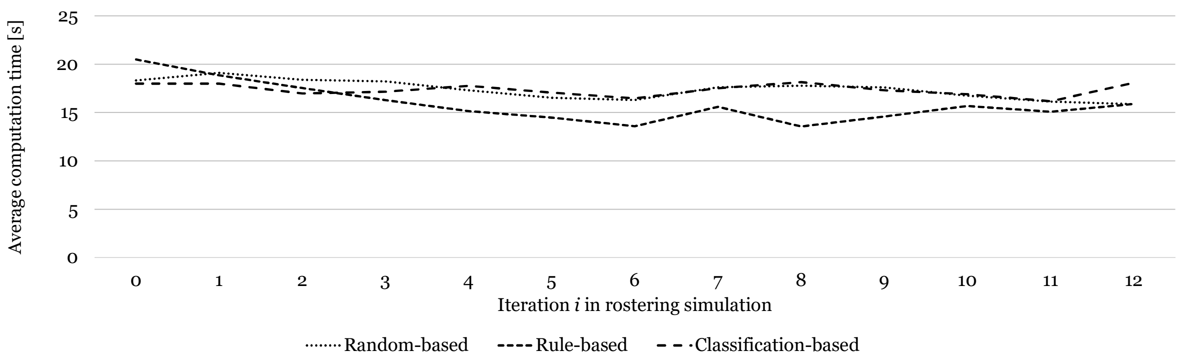


Figure 7.11: Comparison of the average computation time for each crew rostering problem with integration of pairing request evaluation algorithm for the random-based, rule-based and classification-based algorithm

and integrate this into a crew rostering model, develop an approach to identify and measure the (financial) effect of crew preference management on the crew rostering problem and finally, developing an approach to leverage historical crew preference data to identify and define parameters for modeling crew preference management. It can be stated that these four commitments have been successfully performed.

In terms of the validity of the results within this project scope, it can be stated that the results are valid within this scope. The methods that have been developed for evaluating pairing requests are valid methods that have been proven to be usable within a dynamic crew rostering model.

Validity of the results within airline practices

The objective of this research is to make recommendations on methods to evaluate pairing requests while capturing the dynamic nature of the crew rostering problem. In airline practices, methods for the automation of existing manual decision processes can be explored. The evaluation of pairing requests is one of such processes and the algorithms that have been developed can be used on top of the crew rostering software that is used by an airline. However, some limitations to the validity of the results are addressed below:

- Part-time crew members - The model has been designed to suit full-time crew members only to eliminate the many types of activities from the project scope. However, for the model to be valid in airline practices, part-time crew member that comprise around half of the workforce in the case study airline needs to be incorporated as well in the model and the evaluation algorithms.
- Other types of activities - From a crew preference perspective, other types of activities can be incorporated in the dynamic rostering model such as training activities and holidays to create a rostering environment that better matches airline operations.
- Variability in the request submitting behaviour of crew members - The constant level of pairing requests per FTE per week should be challenged as this influences the performance of the algorithms.

Summarized, the methods have the potential to automate the time-consuming process of manually evaluating pairing requests in airline scheduling practices with a rolling rostering approach. For the model to be used in a practical scheduling operations, these assumptions should therefore be challenged and investigated.

8

Sensitivity analysis

The experiments that have been presented in Chapter 6 are preceded by an analysis of the sensitivity of the model parameters and data sets. This chapter presents multiple sensitivity analyses of the models and algorithms to different model input parameters or data sets, as well as findings on the parameters that yield the desired results in the model. In Section 8.1, the sensitivity of the parameters that have been tested in the static rostering model are presented. In Section 8.2, the sensitivity of the parameters that have been tested in the dynamic rostering model are presented.

8.1. Sensitivity of static rostering model parameters

The parameters that have been tested in this section relate to the static rostering model. In this problem, the size of the input sets are crucial to the solution as well as the cost parameters in the generation of rosters. Therefore, the following sensitivity analysis have been performed and discussed; sensitivity to the provided set of crew members is presened in Section and granted requests and sensitivity to cost parameters in the roster.

8.1.1. Sensitivity to the provided set of crew members and requests to be granted

The workforce that is available to cover the pairings, is crucial for airlines to manage. In airline practices, this so-called crew demand can be expressed in gross crew demand and net crew demand. Net crew demand represents the number of FTEs that is required for feasible operation of all the pairings. In this research, only crew members operating on a full-time contract are assumed. Therefore, the provided set of crew members to the model represents the net crew demand in this sensitivity analysis.

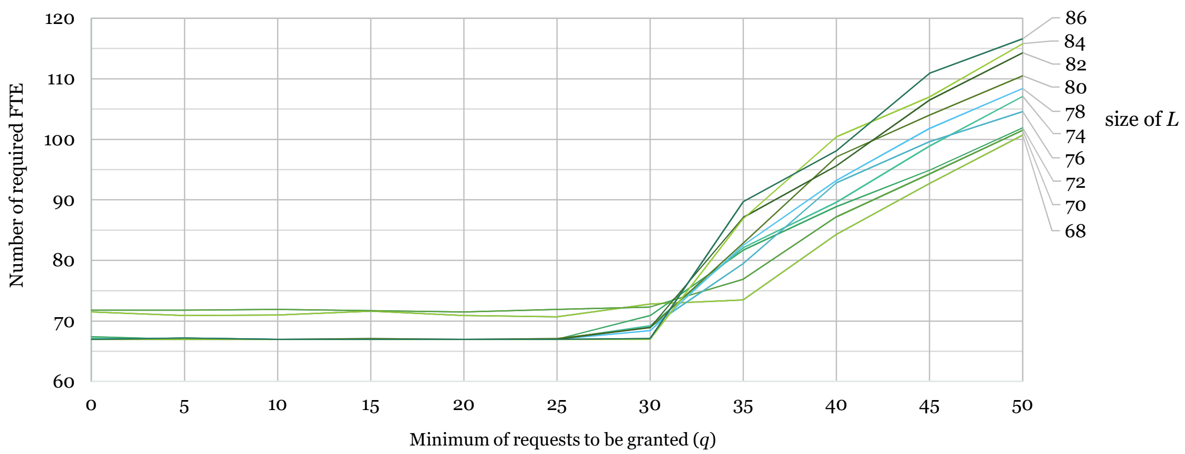


Figure 8.1: Results of the sensitivity analysis of the provided set of crew members to the static rostering model (step size of varying $L = 2$, step size of varying $q = 5$, $n = 10$)

In this sensitivity analysis, the size of the provided set of crew members has been varied in order to determine the minimum of crew members that is necessary to cover the 71 pairings provided to the model. Simultaneously, the minimum number of pairing requests to be granted in the rostering model has been varied. With this variation, the constraint of minimum pairing requests to be granted can be explored. As the model is constrained to a minimum level of granted pairing requests, the provided set of crew members is not sufficient to cover all the pairings at a certain level of granted pairing requests. Therefore, the key questions to this analysis are:

1. What the minimum size of the provided set of crew members for covering all the pairings?
2. At what level of minimum granted pairing requests is the provided set of crew members exceeded to be able to grant the provided this minimum number of pairing requests?

An important assumption in this sensitivity analysis is that sufficient number of requests have been submitted by the crew members for the optimization problem to meet the minimum of pairing requests to be granted. Figure 8.1 shows the results of this sensitivity analysis for 10 experiments for each model setting. Along the horizontal axis are the minimum number of requests to be granted (q) and along the vertical axis are the number of required FTEs to cover the pairings *while* the model is being subject to the minimum number of requests to be granted. In the figure, 10 lines are shown that represent the varied size of L; the provided set of crew members in FTEs. This variation is based on the data-driven net crew demand of size of $L = 70$, required for the original schedule of the case study in this research. It can be seen in the figure that for all sizes of the provided set of crew members a tipping point can be identified around the level of minimum granted requests of $q \approx 32$ after which a linear upward trend follows. A critical note to this tipping point is the step size of 5 for increasing q. A smaller step size would have resulted in a more gradual development of n_{FTE} versus q.

At low numbers of minimum granted requests, two lines stand out as they start at a higher number of required FTEs than the other lines (varying around 72 FTEs). These are the lines for a size of $L = 68$ and a size of $L = 70$. This implies that with sizes of the workforce of 68 and 70, the model is not able to assign all the pairings with the available set of crew members, regardless of the minimum requests to be granted. However, with a workforce size of 72, this is possible for low numbers of minimum requests to be granted. An answer to the first question in this sensitivity analysis is, therefore, that a provided workforce of 72 crew members is sufficient for feasible assignment of the pairing schedule of the case study.

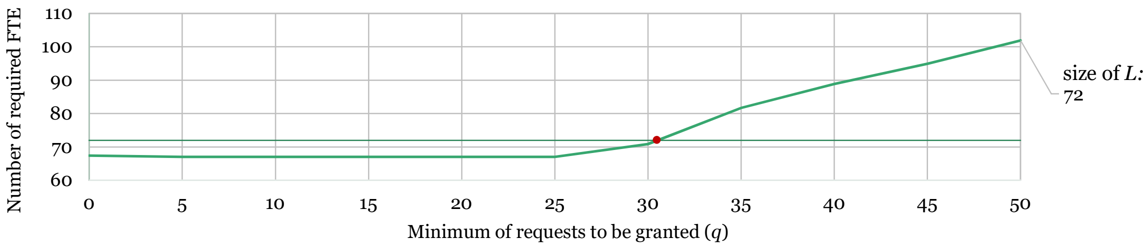


Figure 8.2: Results of the isolated sensitivity analysis of the provided set of crew members to the static rostering model where size of $L = 72$ (step size of varying $q = 5$, $n = 10$)

For clarification, Figure 8.2 shows the isolated sensitivity analysis where the size of $L = 72$. The development of n_{FTE} with increasing q (i.e., the solid line) as well as the level of 72 FTEs (i.e., the dashed line) is plotted. In the figure, it is clear that with low numbers of minimum granted pairing requests, around $\tilde{6}7$ FTEs are required. However, 70 FTEs were required according to the historical airline crew demand data for the pairing schedule. The reason why the pairings can be assigned to 67 FTEs in the model, rather than the 70 FTEs crew members required according to airline data, is due to the assumptions on activity length. The length of the pairing activities in the model have been rounded to full days, while in fact some activities may take fractions of days. For the model, this increases the number of possibilities for some pairings to succeed each other, leading to a bigger solution space. The same assumption causes carry-in pre-assignments of all crew members to be rounded to full days, which also allows for more possibilities for some pairings to succeed each other.

Another remark on Figure 8.2 is that with interpolation, the horizontal line that bounds the number of 72 required FTEs is crossed at a minimum of pairing requests to be granted (i.e., q) is just over 30. This means that with this size of the workforce, given that a sufficient amount of pairings are requested, a maximum of around 42% preferred pairings of the total 71 pairings can be guaranteed. By analyzing Figure 8.1 it can be stated that with increasing size of L, steeper increases of required FTEs exist after this approximate tipping point of $q \approx 32$. Therefore, for higher sizes of L this cross-over point exists at a higher level of q since more flexibility exists for assigning preferred pairings. An answer to the second question in this sensitivity analysis is, that the provided set of 72 FTEs is exceeded when constraining the model to a minimum of around 30 pairing requests.

8.1.2. Sensitivity of costs for slack crew rosters

Recall that the objective function of the rostering optimization consisted of costs for regular crew rosters (i.e., c_r^l) and costs for slack crew rosters (i.e., c_r^s). This objective function is repeated in Equation 8.1. The slack crew rosters take in the pairings cannot be operated by the regular crew rosters. However, the situation in which all the pairings are covered by the regular crew rosters is desirable as slack crew members represent additional required FTEs. The cost parameter for the slack crew roster c_r^s can regulate this. With sufficiently high costs for slack crew rosters, all possibilities for assigning a pairing to a regular crew roster will be considered first before having to resort to slack crew rosters. By exploring the effects of varying c_r^s, the key objective of this analysis is to find an appropriate value for c_r^s in the model.

$$\text{Minimize} \sum_{l\in L}\sum_{r\in R_l} c_r^l \cdot x_r^l + \sum_{s\in S}\sum_{r\in R_s} + c_r^s \cdot x_r^s \tag{8.1}$$

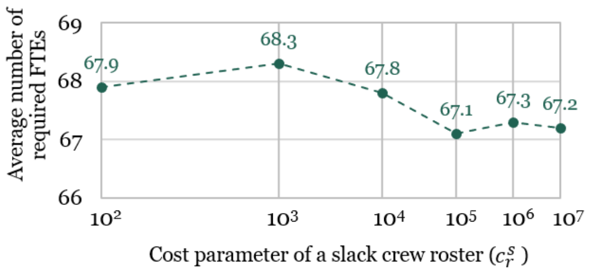


Figure 8.3: Results of the sensitivity analysis of the provided costs for assigning a roster to a slack crew member (i.e., c_r^s) in the static rostering model where size of $L = 72$, size of $P = 71$ ($n = 10$)

Figure 8.3 shows the results of this sensitivity analysis for 10 experiments for each model setting. Along the horizontal axis, the cost parameter for the slack crew roster c_r^s has been varied with a factor of 10 and the resulting average number of required FTEs n_{FTE} has been presented. Based on these results, the decision was made to select 10^6 as a value for c_r^s in the experiments. The objective value of the optimization problem averages at around $4.8 \cdot 10^5$. Therefore, selecting 10^6 as a value for c_r^s ensures an order of magnitude that is sufficiently high for serving the purpose of driving the solution of the optimization problem to choosing for regular crew rosters.

8.1.3. Sensitivity of bonus costs for pairing request assignment

Recall that the cost function of regular crew roster costs (i.e., c_r^l) contained a bonus cost parameter for assigning pairing requests in a crew roster (i.e. c_b). This cost function is repeated in Equation 8.1. Bonus costs are used in current literature for incorporating crew preferences in an objective function. However, in this research, the void costs are the other driver of costs in the cost function of Equation 8.2. In addition to the minimum pairing request constraint, the bonus costs can be used as a means to prefer a roster that holds requested pairings over a roster that holds no requested pairings. The degree to which to enforce this preference can be expressed through c_b. By exploring the effects of varying c_b, the key objective of this analysis is to find an appropriate value for c_b in the model.

$$c_r^l = \left(\sum_{i=1}^{j} (n_v^i)_r^l \cdot c_v^i\right) + \left(q_r^l \cdot c_b\right) \tag{8.2}$$

Figure 8.4 shows the results of this sensitivity analysis for 10 experiments for each model setting. Along the horizontal axis, the bonus costs for assigning requested pairing in a roster c_b has been varied with a factor

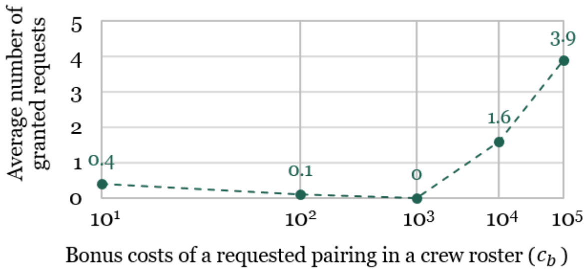


Figure 8.4: Results of the sensitivity analysis of the provided bonus costs for assigning a requested pairing in a roster (i.e., c_b) in the static rostering model where size of $L = 72$, size of $P = 71$ ($n = 10$)

of 10 and the resulting average number of granted requests n_q has been presented. Because the model for this sensitivity analysis is the static rostering model, it is assumed that no pairing requests have already been granted in an earlier planning stage. Another important assumption in this sensitivity analysis is that the minimum has been set to a value of 0. Therefore, the granted pairing requests are driven by the bonus costs c_b, only. Based on these results, the decision was made to select 10^1 as a value for c_b in the experiments. With this value, a slight preference is given to rosters that hold pairing requests in the search for feasible solutions in the search process of the optimization problem.

8.2. Sensitivity of dynamic rostering model parameters with pairing request evaluation algorithms

The parameters that have been tested in this section relate to the dynamic rostering model with integration of pairing request evaluation algorithms. For integration of the algorithms, it is important to test the sensitivity of parameters on which the working principles of the algorithms are based. Therefore, the following sensitivity analysis have been performed and discussed below; sensitivity to the provided set of crew members and granted requests and sensitivity to cost parameters in the roster.

8.2.1. Sensitivity to probability for granting pairing requests

An important parameter in the evaluation of pairing requests with the random-based pairing request evaluation algorithm is the probability p_{grant} with which a pairing request is granted after it has endured the feasibility check. Specifics on this evaluation step have been discussed in Section 6.2, in which the parameter p_{grant} was introduced. By analyzing the effects of varying p_{grant}, the key objective of this analysis is to find an appropriate value for this granting probability in the case of incorporation of the random-based pairing request evaluation algorithm in the dynamic rostering model. Figure 8.5 shows the results of this sensitivity analysis for 10 experiments for each value of p_{grant}. Along the horizontal axis, the probability for granting a feasible pairing request is varied. The full simulation scenario has been executed and the resulting values for n_{FTE} of the final eight iterations in the scenarios has been presented along the vertical axis. Above the graph, the average values for each boxplot are presented that are marked in the figure with an x-mark. One value along the horizontal axis is the specific $p_{grant} = 0.57$. This value is based on airline data in the case study, which has been explained in Section 7.3. With increasing p_{grant}, there is an increasing trend in the average number of required FTEs, which is expected. It is important to note that the effects of increasing p_{grant} still result in a reasonable increase of FTEs (i.e., 5.12 extra FTEs required if all feasible pairing requests would be granted). This is possible due to the fact that only *feasible* pairing requests that *fit* in the existing roster are evaluated. As can be seen in the figure, the average number of required FTEs surpasses the 72 provided FTEs when p_{grant} equals 0.1 where this value is 72.64 FTEs. Therefore, lower probabilities for p_{grant} have been tested as well.

Figure 8.5 shows the results of the sensitivity analysis for lower values of p_{grant} where 10 experiments are performed for each value. The layout of Figure 8.6 is similar to Figure 8.5. As can be seen in Figure 8.6, an average number of required for a value of p_{grant} that is lower than 0.04. Therefore, a value of $p_{grant} = 0.04$ has been selected for experiments with the random-based pairing request evaluation algorithm.

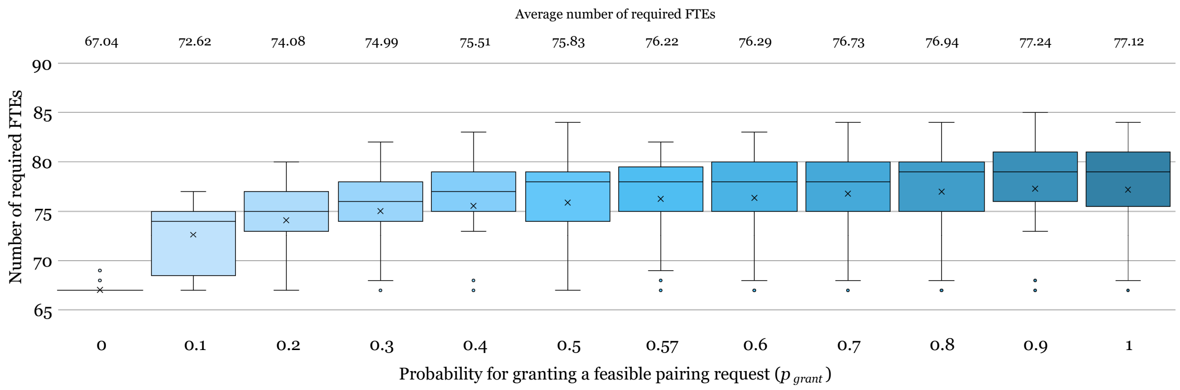


Figure 8.5: Results of the sensitivity analysis of the probability for granting a feasible pairing request (i.e., p_{grant}) in the dynamic rostering model with random-based pairing request evaluation, size of $L = 72$, size of $P = 71$ ($n = 10$)

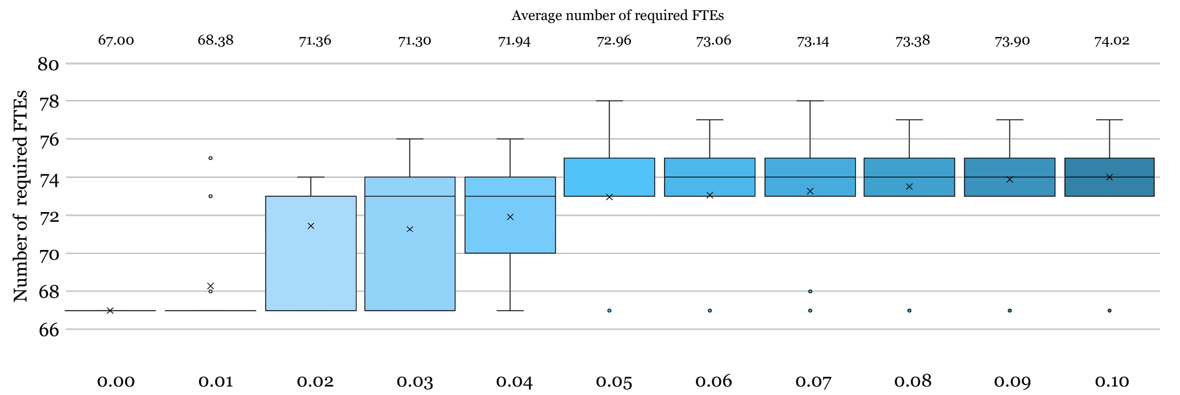


Figure 8.6: Results of the sensitivity analysis of the lower probability for granting a feasible pairing request (i.e., p_{grant}) in the dynamic rostering model with random-based pairing request evaluation, size of $L = 72$, size of $P = 71$ ($n = 10$)

8.2.2. Sensitivity to number of submitted requests

In airline practices, the number of submitted pairing requests can vary over time. In this research, however, a number of average submitted pairing request per FTE per week has been used as an input to the model (i.e., n_y). It is important to note that this value is inseparably connected with p_{grant}. For a higher quantity of submitted requests, a similar value will result in higher absolute amount of granted requests. The difference, from a practical point of view, is that p_{grant} can be adjusted in the scheduling process while n_y is characterized by its more external nature; the submitting behaviour of pairing requests by crew members. By analyzing the effects of varying n_y, the key objective of this analysis is to gain insights in the effects of this variation in the case of incorporation of the random-based pairing request evaluation algorithm in the dynamic rostering model with constant p_{grant}.

Figure 8.7 shows the results of this sensitivity analysis for 10 experiments for each value of n_y. Along the horizontal axis, the number of submitted pairing requests per FTE per week is varied. The full simulation scenario has been executed and the resulting values for n_{FTE} of the final eight iterations in the scenarios has been presented along the vertical axis. Above the graph, the average values for each boxplot are presented that are marked in the figure with an x-mark. One value along the horizontal axis is the specific $n_y = 0.457$. This value is based on airline data in the case study, which has been explained in Section 7.3.

From the average numbers of required FTEs in Figure 8.7, it can be concluded that the variation in n_y results in values of n_{FTE} behaves as expected as it is proportional to the effects of increasing p_{grant} with similar rates. This relations is important when using a random-based pairing request evaluation algorithm and p_{grant} should be determined with n_y either provided or predicted.

8.2.3. Sensitivity of rules in rule-based pairing request evaluation algorithm

The rule-based pairing request evaluation algorithm was inspired by current airline practices. Recall that the assessment for granting or rejecting a pairing request relied on assessing the length of the induced void in

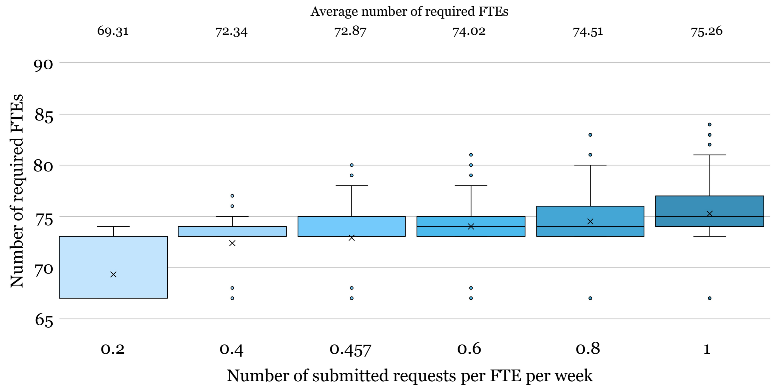


Figure 8.7: Results of the sensitivity analysis of the number of submitted requests by crew members (i.e., n_y) in the dynamic rostering model with random-based pairing request evaluation, size of $L = 72$, size of $P = 71$ ($n = 10$)

the roster in case of granting the request. The median pairing length in the roster is selected and multiples of this median pairing length comprise the set of acceptable void lengths induced by granting a request. The objective of this sensitivity analysis is to test the effects of deviating from this standard rule. The set of acceptable void lengths can be subject to deviations that add more possibilities to this set. This is illustrated with an example in Table 8.1. In the table, the set of acceptable void lengths is presented for a 0% and 1% deviation from multiples of the median pairing length (i.e., 7). The results of the sensitivity analysis of varying these deviations are presented in Figure 8.8. From the figure, it is clear that a 0% deviation from the standard rule already results in a requirement of more FTEs (i.e., 73.73 on average) than the 72 FTEs that were provided to the model. This indicates that the rule-based pairing request algorithm could further be explored to make a better rule-based assessment on granting or rejecting a pairing request. An example possibility would be to hybridize this approach with a random-based approach. For the experiments with the rule-based pairing request algorithm, the rule with a deviation of 1% was selected since early results showed a lower average on number of required FTEs. In retrospect, a deviation of 0% would have been more appropriate.

Table 8.1: Example of acceptable void lengths for two deviation percentages

Deviation from rule	**Acceptable void lengths in days induced by granting a request**																					
0%	7	14	21	28	35	42	49	56	62	69	76	83										
1%	7	14	21	27	28	34	35	41	42	48	49	55	56	62	63	69	70	76	77	82	83	84

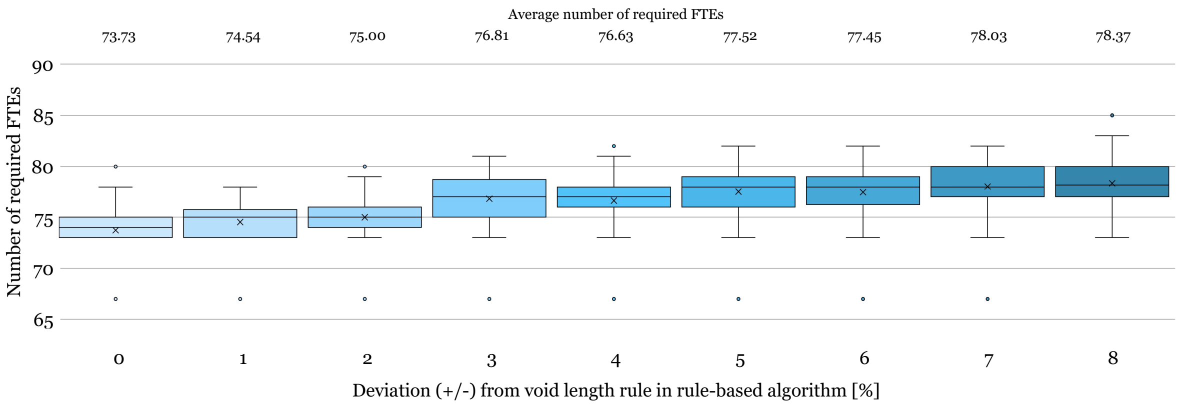


Figure 8.8: Results of the sensitivity analysis of the deviation percentage from the standard rule in the dynamic rostering model with rule-based pairing request evaluation, size of $L = 72$, size of $P = 71$ ($n = 10$)

8.2.4. Sensitivity of training data in classification-based pairing request evaluation algorithm

The classification-based pairing request evaluation algorithm has been designed to learn from a feedback mechanism on historic pairing request evaluation decisions. Multiple sets of training data have been provided to the model and the objective of this sensitivity analysis is to compare the effects of providing training data with different sets of features to the classification-based pairing request algorithm. Table 8.2 presents the five sets of training data that have been tested to which is referred to in the figures that follow.

The results of this sensitivity analysis are presented in Figure 8.9 and Figure 8.10. Along the horizontal axis of both figures, the iteration number i in the simulation is indicated. Figure 8.9 shows the results for average n_{FTE} in the course of the dynamic rostering model simulation for each of the feature sets. In Figure 8.9, the level of the provided set of crew members of 72 FTEs is indicated as a benchmark. Figure 8.10 shows the results for average number of granted requests. Correlation of n_{FTE} and n_q in the two figures can be identified for similar feature sets. This is expected as n_{FTE} increases with an increase in granted pairing requests, which was clear from Section 8.1.1. Feature set 1 shows a clear bias towards granting pairing requests in a reduced set of iterations. It is the only feature set in which the iteration number and batch number have been taken into account. This result indicates that these features are not suitable for the classification-based evaluation algorithm as pairing requests are evaluated in a rolling roster in practice. The results of feature set 2 show a low level of granted requests and a required set of crew members of $\tilde{6}7$. This indicates that this set of features is conservative in granting pairing requests. In turn, too many requests are granted for provided set of crew members of 72 FTEs to be sufficient when using feature set 4 and 5. These sets, show very similar performance in both n_{FTE} and n_q. Both sets hold the feature for number of granted requests at the moment of requesting which does not appear to be a good indicator for the decision on granting a requests. Feature set 3 shows the best performance in which the provided level of crew members is not exceeded. This sensitivity analysis shows that the training data provided to the classification-based pairing request algorithm is key for a desired performance of a classification-based pairing request algorithm. This has been discussed in more detail in Section 7.2.

Table 8.2: Overview of chosen features in the sensitivity analysis of the classification-based pairing request evaluation algorithm (* number of granted requests counted at the decision moment of the request evaluation)

Feature set	**Iteration number**	**Batch number**	**Pairing length**	**Pairing departure day**	**Number of granted requests***
1	X	X	X	X	X
2			X	X	X
3			X		
4					X
5			X		X

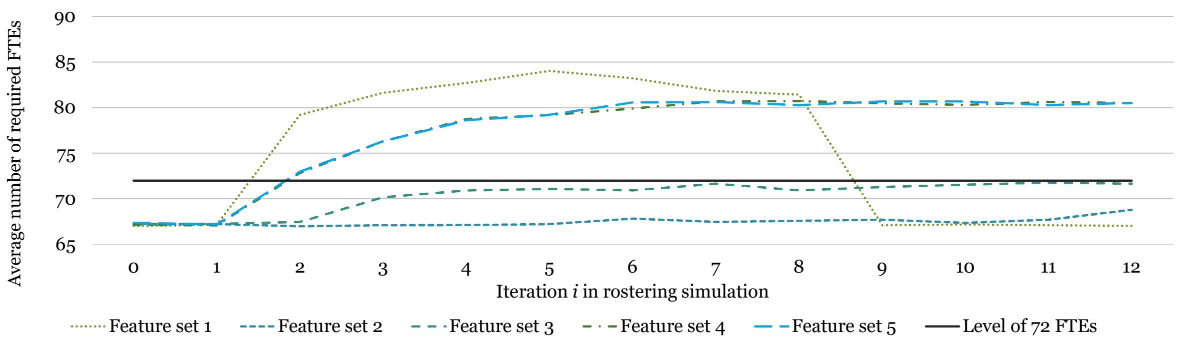


Figure 8.9: Results of the sensitivity analysis of average number of required FTEs when using different feature sets in the training data for the dynamic rostering model with classification-based pairing request evaluation, size of $L = 72$, size of $P = 71$ $(n = 50)$

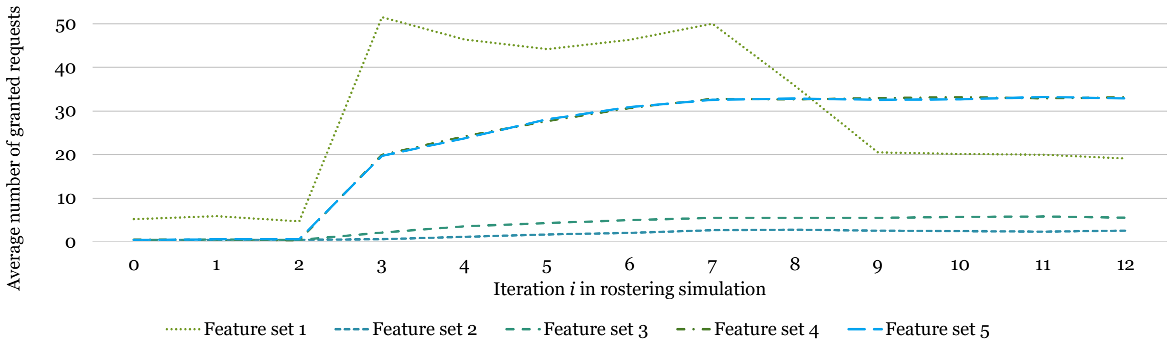


Figure 8.10: Results of the sensitivity analysis of average number granted requests when using different feature sets in the training data for the dynamic rostering model with classification-based pairing request evaluation, size of $L = 72$, size of $P = 71$ ($n = 50$)

9
Conclusions and recommendations

This chapter finalizes the thesis report with conclusions on the performed research and reflections on the contributions of this research. The conclusions on the research have been discussed in Section 9.1. Moreover, contributions of this research to the current practice in airline crew rostering has been discussed in Section 9.2. Finally, recommendations for further research in pairing request evaluation have been presented in Section 9.3.

9.1. Conclusions

Requests by crew members for operating specific pairings need to be evaluated effectively to avoid inefficient schedules. This is especially challenging in a rostering problem in which pairing requests are submitted while other pairing requests have already been granted. Despite this being a dynamic problem, literature addresses it as a static problem. Furthermore, it was found that current state of the art does not provide methods for evaluating pairing requests. The research that has been presented in this thesis report has focused on the evaluation of pairing requests in dynamic airline crew rostering. Central to this research has been the research question that is presented below:

How could pairing requests in the airline crew rostering problem be evaluated?

The research in this thesis challenges the static approach to airline crew rostering and provides the first integrated means for evaluating pairing requests in a broader planning horizon. The objective is to make recommendations on methods to evaluate pairing requests while capturing the dynamic nature of the crew rostering problem. It can be concluded that this has been achieved by addressing four key aspects that are discussed below.

Firstly, a method for capturing the dynamic nature of the crew rostering problem has been established. This dynamic model is required to effectively evaluate pairing requests in an earlier scheduling stage than is possible with current optimization approaches. Since a construction-based method is used, the effects of granted pairing requests over time can be captured. In a case study with a major European airline, the need for a dynamic rostering model over a static model was confirmed. The dynamic model has been used throughout this research for simulations of crew rostering problems with incoming pairing requests that need to be evaluated.

Secondly, the effects of pairing requests to the number of required FTEs to operate all the pairings in a schedule have been investigated. The number of required FTEs represents the crew resources that an airline needs for the feasible operation of the pairings within a crew division. These crew resources can be translated into financial resources required to operate a schedule feasibly. The effects of pairing requests on the solution process have been tested in experiments. In these experiments, the linear programming rostering model was constrained to a minimum of to be granted pairing requests. Varying this minimum of pairing requests resulted in insights on the (financial) effects of pairing requests. A level of granted pairing requests can be determined at which the available set of crew members is not sufficient to cover all the pairings in the schedule. For a workforce of 72 FTEs, the level of granted pairing requests at which this workforce was not sufficient to operate the pairings was determined at ≥ 31 granted pairing requests out of the 71 pairings in the schedule. This approach can be used by airline scheduling departments to explore the allowance of pairing requests in their crew preference management practices.

Thirdly, historical crew scheduling data and pairing request data has been leveraged to calibrate the models and pairing request evaluation algorithms. Airline schedule data has been used as a method for pricing voids in rosters with a pricing method that is based on the expected loss days of different types of voids in the schedule. Moreover, pairing request data that was available in a case study with a major European airline served as a benchmark for comparison of the methods for evaluating pairing requests that have been developed in this research. For the case study airline, 72 FTEs are required for feasible operation of the pairing schedule in this research while 15.51 pairing requests can be granted each week, on average.

Finally, three algorithms have been developed and tested as methods for the automated evaluation of pairing requests. These algorithms have been integrated into the dynamic rostering model. With the implementation of the random-based pairing request evaluation algorithm, 72.57 FTEs are required for feasible operation of the pairings while an average of 7.31 pairing requests can be granted each week. This indicates that even with random evaluation, a reasonable level of granted pairing requests can be reached. With the implementation of the rule-based pairing request evaluation algorithm, 74.67 FTEs are required for an average of as much as 16.02 granted pairing requests each week. This shows its potential for practical implementation, as the rule-based approach is inspired by current scheduling practices. Lastly, with the implementation of the classification-based pairing request evaluation algorithm, 71.48 FTEs are required for an average of 5.59 granted pairing requests each week. Although this number of granted pairing requests is relatively low compared to the other methods, the classification-based algorithm is the only method for which the provided set of crew members of 72 FTEs is sufficient. Training the classification-based algorithm with simulation-based rostering data has proven to be an effective approach for a pairing request evaluation method. It can be concluded that the pairing request evaluation algorithms are viable methods for the evaluation of pairing requests. Selection of an appropriate method depends on the incentives for pairing request evaluation within an airline practice. In airline practices, the algorithms can be used as a decision mechanism on top of the crew rostering software that is used by an airline. This implementation step has the potential to mitigate hours of manual trade-off making for pairing request evaluation by scheduling personnel.

9.2. Research contributions

It can be concluded that the commitments that were made concerning the scope of this research have been successfully completed. The methods that have been developed have been designed to address a gap in the scientific body. Below, the contributions that were made in this research are discussed in more detail.

Definition of a dynamic airline crew rostering problem

To model the dynamic nature of the airline crew rostering problem, a rolling rostering approach has been formulated and a dynamic airline crew rostering problem has been defined. This dynamic rostering model with integration of pairing request evaluation is a novel construction-based crew rostering method. It allows for evaluating crew preferences in an earlier scheduling stage than is possible with current optimization approaches. The dynamic airline crew rostering model is useful in multiple ways. Firstly, like was illustrated in this research, simulation scenarios can be formulated that resemble a crew rostering airline practice with a rolling rostering approach. Secondly, a crew preference management strategy for crew pairing requests can be incorporated into a dynamic model as the input of pairing requests has a stochastic and dynamic nature. While crew preferences in the form of bids could already be integrated in models in available literature, this was not the case for crew preferences in the form of regularly incoming pairings requests. Thirdly, because of its relatively small crew rostering planning horizon, the dynamic rostering model is less complex in terms of solution space. Its approach to viewing the rostering problem can serve as starting point for a construction based rostering method. Such an approach could be useful for the personnel that creates the schedules at an airline that could potentially make trade-offs in construction-based decisions rather than having to wait hours for large optimization problems to be solved.

Development of a data-driven method for roster pricing in a rostering solution process

A new data-driven method for roster pricing ensures that rosters are valued based on expected loss days of roster inefficiencies. In order to distinguish between the effect of different types of voids in the roster, a novel method of approaching the costs of a roster was introduced. This method is a way of translating the decisions that are made in the scheduling phase of the airline planning problem to the eventual operational outcome of the roster. It can be regarded as measure of feedback for a better performing rostering process which can be updated by up-to-date of the current situation in terms of, for instance, the absenteeism rate and the expected manpower.

Capturing the (financial) effect of crew preference management in a crew rostering problem
By expressing the number of required FTEs for ensuring productivity of the pairings as a function of the minimum of requests in a roster, the financial effect of crew preference management has been captured in the form of crew resources. Expressing this financial effect of crew preference management explicitly, can serve as a guidance for crew preference management policy to determine thresholds on crew pairing requests for given scheduling situations. With this information at hand, a trade-off can be made by airline scheduling departments too which degree the granting of pairing requests is desired at what cost.

Development of method to generate training data for machine learning algorithms in airline crew rostering
The convenience of the dynamic crew rostering model is also evident from a data collection point of view. Since decisions that are made in an earlier scheduling stage can be tracked throughout the simulation iterations, data can be collected on how decisions on, in this case, pre-assigned activities turn out when the roster is finalized. The dynamic crew rostering model can, therefore, serve as a means to collect training data for machine learning algorithms in crew rostering. This is a novel way of approaching scheduling decisions that are traditionally made by large-scale optimization problems.

Development of algorithms for pairing request evaluation in a wide planning horizon
As a means to evaluate pairing requests in a wider planning horizon, pairing request evaluation algorithms have been developed that have been proven to be valid methods for evaluating pairing requests in dynamic airline crew rostering. Similar methodology than was introduced for the pairing request evaluation algorithms could be explored for the pre-assignment of other types of activities such as training sessions or leave days. Another important advantage of this way of evaluating pairing requests is that it is integrated into a rostering model rather than having to be done manually.

9.3. Recommendations

This research is suitable for pursuing both methodological research as well as purely practical implementation. Since the inspiration for the research came partly from the challenges of current practices in an existing crew scheduling process, this research can serve as a starting point for revising current pairing request practices from an implementation point of view. The following three main recommendations have been identified as interesting pursuits.

Extension of activity types and part-time crew members
The assumptions that have been made to properly scope this research limit the type of activities to pairings only. To more accurately resemble a real airline practice, other types of activities could be included as well. It depends on the implementation purpose of the user what level of complexity and resemblance is required for this modeling approach to yield what is desired. Moreover, the model has been designed to suit full-time crew members only. However, for the model to be valid in airline practices, part-time crew members needs to be incorporated as well in the model and the evaluation algorithms.

Dynamic pricing of crew preferences in airline crew rostering
A method that could be explored further is to steer the pairing requests submitting behaviour of crew members towards pairings requests that are more likely to be granted. This yields an open-market environment for the number of pairing requests submitted by crew members that can more proactively be managed by airlines.

Feature engineering for classification-based decision making in airline crew rostering
A more methodological research could be formulated in which the collection of training data for pairing request evaluation algorithms or other types of machine learning algorithms in a crew rostering environment are explored further. As this research has not investigated the effects or the potential redundancy of certain features, more research could be done to investigate the potential of applying classification-based algorithms into a construction-based rostering problem.

A
Pairing schedule

This appendix presents the data of the weekly pairing schedule used as input to the rostering model in Table A.1, the geographical indicators of the destination IDs in Table A.2 and a visualization of the weekly pairing schedule in Figure A.1.

Table A.1: Data of the weekly pairing schedule used as input to the rostering model

Pairing ID	**Destination ID**	**Departure time**	**Flight duty days**	**Rest period days**	**Request probability**
PA_0001	ME_003	1-01-18 0:00	5	2	0.00197
PA_0002	ME_002	1-01-18 0:00	3	3	0.00346
PA_0003	NA_005	1-01-18 0:00	4	3	0.00385
PA_0004	CA_004	1-01-18 0:00	3	4	0.02412
PA_0005	ME_001	1-01-18 0:00	3	3	0.04703
PA_0006	CA_001	1-01-18 0:00	4	3	0.02737
PA_0007	NA_004	1-01-18 0:00	4	3	0.00264
PA_0008	AF_003	1-01-18 0:00	3	2	0.00255
PA_0009	NA_002	1-01-18 0:00	5	2	0.02990
PA_0010	AF_001	1-01-18 0:00	4	3	0.00197
PA_0011	ME_002	2-01-18 0:00	3	3	0.00346
PA_0012	CA_003	2-01-18 0:00	3	4	0.00385
PA_0013	ME_003	2-01-18 0:00	6	3	0.02412
PA_0014	CA_004	2-01-18 0:00	3	4	0.02038
PA_0015	ME_001	2-01-18 0:00	3	3	0.00264
PA_0016	CA_001	2-01-18 0:00	4	3	0.00255
PA_0017	AF_002	2-01-18 0:00	7	4	0.00279
PA_0018	AF_003	2-01-18 0:00	3	2	0.00197
PA_0019	AF_001	2-01-18 0:00	4	3	0.00385
PA_0020	NA_005	3-01-18 0:00	5	3	0.02412
PA_0021	ME_001	3-01-18 0:00	3	3	0.04703
PA_0022	NA_004	3-01-18 0:00	4	3	0.02737
PA_0023	AF_003	3-01-18 0:00	3	2	0.00264
PA_0024	AF_004	3-01-18 0:00	4	2	0.00255
PA_0025	ME_002	3-01-18 0:00	3	3	0.00058
PA_0026	CA_004	3-01-18 0:00	3	4	0.02990
PA_0027	CA_002	3-01-18 0:00	4	4	0.00197
PA_0028	AF_001	3-01-18 0:00	4	3	0.00346
PA_0029	CA_003	3-01-18 0:00	3	4	0.00385
PA_0030	ME_002	4-01-18 0:00	3	3	0.02412
PA_0031	NA_003	4-01-18 0:00	5	4	0.02665
PA_0032	ME_001	4-01-18 0:00	3	3	0.00264
PA_0033	AF_003	4-01-18 0:00	3	2	0.00255
PA_0034	CA_002	4-01-18 0:00	4	3	0.01672
PA_0035	AF_001	4-01-18 0:00	4	3	0.00197
PA_0036	CA_003	4-01-18 0:00	3	4	0.00385
PA_0037	NA_002	4-01-18 0:00	4	3	0.02412
PA_0038	ME_003	4-01-18 0:00	5	2	0.04703
PA_0039	CA_004	4-01-18 0:00	3	4	0.00264
PA_0040	CA_004	5-01-18 0:00	3	4	0.00255

Table A.1 continued from previous page

Pairing ID	Destination ID	Departure time	Flight duty days	Rest period days	Request probability
PA_0041	CA_003	5-01-18 0:00	4	3	0.00058
PA_0042	AF_004	5-01-18 0:00	4	2	0.02990
PA_0043	AF_003	5-01-18 0:00	3	2	0.00279
PA_0044	ME_001	5-01-18 0:00	5	3	0.00197
PA_0045	AF_002	5-01-18 0:00	6	3	0.02412
PA_0046	AF_001	5-01-18 0:00	4	3	0.04703
PA_0047	ME_002	5-01-18 0:00	3	3	0.06577
PA_0048	CA_001	5-01-18 0:00	4	3	0.00264
PA_0049	CA_003	5-01-18 0:00	3	4	0.00255
PA_0050	NA_004	5-01-18 0:00	4	3	0.00197
PA_0051	ME_002	6-01-18 0:00	4	4	0.02412
PA_0052	AF_001	6-01-18 0:00	4	3	0.03840
PA_0053	AF_003	6-01-18 0:00	3	2	0.01672
PA_0054	NA_005	6-01-18 0:00	4	3	0.00058
PA_0055	NA_001	6-01-18 0:00	3	4	0.02990
PA_0056	NA_002	6-01-18 0:00	4	3	0.00279
PA_0057	CA_003	6-01-18 0:00	5	3	0.00197
PA_0058	CA_002	6-01-18 0:00	4	3	0.00346
PA_0059	CA_004	6-01-18 0:00	3	4	0.00385
PA_0060	ME_003	6-01-18 0:00	4	2	0.02412
PA_0061	CA_003	7-01-18 0:00	5	3	0.04703
PA_0062	NA_004	7-01-18 0:00	3	4	0.02737
PA_0063	AF_003	7-01-18 0:00	3	2	0.00264
PA_0064	AF_002	7-01-18 0:00	7	4	0.00255
PA_0065	NA_003	7-01-18 0:00	6	4	0.02990
PA_0066	CA_004	7-01-18 0:00	3	4	0.00197
PA_0067	AF_001	7-01-18 0:00	4	3	0.00346
PA_0068	AF_004	7-01-18 0:00	5	2	0.00385
PA_0069	CA_001	7-01-18 0:00	4	3	0.02412
PA_0070	NA_001	7-01-18 0:00	4	3	0.02038
PA_0071	ME_003	7-01-18 0:00	4	2	0.00264

Table A.2: Geographical indicators of each of the destination IDs in the pairing schedule of the weekly pairing schedule that is used as input to the rostering model

Destination ID	Geographic indicator
AF_001	Africa 1
AF_002	Africa 2
AF_003	Africa 3
AF_004	Africa 4
CA_001	Caribbean 1
CA_002	Caribbean 2
CA_003	Caribbean 3
CA_004	Caribbean 4
ME_001	Middle East 1
ME_002	Middle East 2
ME_003	Middle East 3
NA_001	North America 1
NA_002	North America 2
NA_003	North America 3
NA_004	North America 4
NA_005	North America 5

	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED
PA_0001	PA_0001	ME_003	ME_003	ME_003	ME_003	RP	RP										
PA_0002	PA_0002	ME_002	ME_002	RP	RP	RP											
PA_0003	PA_0003	NA_005	NA_005	NA_005	RP	RP	RP										
PA_0004	PA_0004	CA_004	CA_004	RP	RP	RP	RP										
PA_0005	PA_0005	ME_001	ME_001	RP	RP	RP											
PA_0006	PA_0006	CA_001	CA_001	CA_001	RP	RP	RP										
PA_0007	PA_0007	NA_004	NA_004	NA_004	RP	RP	RP										
PA_0008	PA_0008	AF_003	AF_003	RP	RP												
PA_0009	PA_0009	NA_002	NA_002	NA_002	NA_002	RP	RP										
PA_0010	PA_0010	AF_001	AF_001	AF_001	RP	RP	RP										
PA_0011		PA_0011	ME_002	ME_002	RP	RP	RP										
PA_0012		PA_0012	CA_003	CA_003	RP	RP	RP	RP									
PA_0013		PA_0013	ME_003	ME_003	ME_003	ME_003	ME_003	RP	RP	RP							
PA_0014		PA_0014	CA_004	CA_004	RP	RP	RP	RP									
PA_0015		PA_0015	ME_001	ME_001	RP	RP	RP										
PA_0016		PA_0016	CA_001	CA_001	CA_001	RP	RP	RP									
PA_0017		PA_0017	AF_002	AF_002	AF_002	AF_002	AF_002	AF_002	RP	RP	RP	RP					
PA_0018		PA_0018	AF_003	AF_003	RP	RP											
PA_0019		PA_0019	AF_001	AF_001	AF_001	RP	RP	RP									
PA_0020			PA_0020	NA_005	NA_005	NA_005	NA_005	RP	RP	RP							
PA_0021			PA_0021	ME_001	ME_001	RP	RP	RP									
PA_0022			PA_0022	NA_004	NA_004	NA_004	RP	RP	RP								
PA_0023			PA_0023	AF_003	AF_003	RP	RP										
PA_0024			PA_0024	AF_004	AF_004	AF_004	RP	RP									
PA_0025			PA_0025	ME_002	ME_002	RP	RP	RP									
PA_0026			PA_0026	CA_004	CA_004	RP	RP	RP	RP								
PA_0027			PA_0027	CA_002	CA_002	CA_002	RP	RP	RP	RP							
PA_0028			PA_0028	AF_001	AF_001	AF_001	RP	RP	RP								
PA_0029			PA_0029	CA_003	CA_003	RP	RP	RP	RP								
PA_0030				PA_0030	ME_002	ME_002	RP	RP	RP								
PA_0031				PA_0031	NA_003	NA_003	NA_003	NA_003	RP	RP	RP	RP					
PA_0032				PA_0032	ME_001	ME_001	RP	RP	RP								
PA_0033				PA_0033	AF_003	AF_003	RP	RP									
PA_0034				PA_0034	CA_002	CA_002	CA_002	RP	RP	RP							
PA_0035				PA_0035	AF_001	AF_001	AF_001	RP	RP	RP							
PA_0036				PA_0036	CA_003	CA_003	RP	RP	RP	RP							
PA_0037				PA_0037	NA_002	NA_002	NA_002	RP	RP	RP							
PA_0038				PA_0038	ME_003	ME_003	ME_003	ME_003	RP	RP							
PA_0039				PA_0039	CA_004	CA_004	RP	RP	RP	RP							
PA_0040					PA_0040	CA_004	CA_004	RP	RP	RP	RP						
PA_0041					PA_0041	CA_003	CA_003	CA_003	RP	RP	RP						
PA_0042					PA_0042	AF_004	AF_004	AF_004	RP	RP							
PA_0043					PA_0043	AF_003	AF_003	RP	RP								
PA_0044					PA_0044	ME_001	ME_001	ME_001	ME_001	RP	RP	RP					
PA_0045					PA_0045	AF_002	AF_002	AF_002	AF_002	AF_002	RP	RP	RP				
PA_0046					PA_0046	AF_001	AF_001	AF_001	RP	RP	RP						
PA_0047					PA_0047	ME_002	ME_002	RP	RP	RP							
PA_0048					PA_0048	CA_001	CA_001	CA_001	RP	RP	RP						
PA_0049					PA_0049	CA_003	CA_003	RP	RP	RP	RP						
PA_0050					PA_0050	NA_004	NA_004	NA_004	RP	RP	RP						
PA_0051						PA_0051	ME_002	ME_002	ME_002	RP	RP	RP	RP				
PA_0052						PA_0052	AF_001	AF_001	AF_001	RP	RP	RP					
PA_0053						PA_0053	AF_003	AF_003	RP	RP							
PA_0054						PA_0054	NA_005	NA_005	NA_005	RP	RP	RP					
PA_0055						PA_0055	NA_001	NA_001	RP	RP	RP	RP					
PA_0056						PA_0056	NA_002	NA_002	NA_002	RP	RP	RP					
PA_0057						PA_0057	CA_003	CA_003	CA_003	CA_003	RP	RP	RP				
PA_0058						PA_0058	CA_002	CA_002	CA_002	RP	RP	RP					
PA_0059						PA_0059	CA_004	CA_004	RP	RP	RP	RP					
PA_0060						PA_0060	ME_003	ME_003	ME_003	RP	RP						
PA_0061							PA_0061	CA_003	CA_003	CA_003	CA_003	RP	RP	RP			
PA_0062							PA_0062	NA_004	NA_004	RP	RP	RP	RP				
PA_0063							PA_0063	AF_003	AF_003	RP	RP						
PA_0064							PA_0064	AF_002	AF_002	AF_002	AF_002	AF_002	AF_002	RP	RP	RP	RP
PA_0065							PA_0065	NA_003	NA_003	NA_003	NA_003	NA_003	RP	RP	RP	RP	
PA_0066							PA_0066	CA_004	CA_004	RP	RP	RP	RP				
PA_0067							PA_0067	AF_001	AF_001	AF_001	RP	RP	RP				
PA_0068							PA_0068	AF_004	AF_004	AF_004	AF_004	RP	RP				
PA_0069							PA_0069	CA_001	CA_001	CA_001	RP	RP	RP				
PA_0070							PA_0070	NA_001	NA_001	NA_001	RP	RP	RP				
PA_0071							PA_0071	ME_003	ME_003	ME_003	RP	RP					

Figure A.1: Visualization of the weekly pairing schedule used as input to the rostering model

B

Example rostering model solution

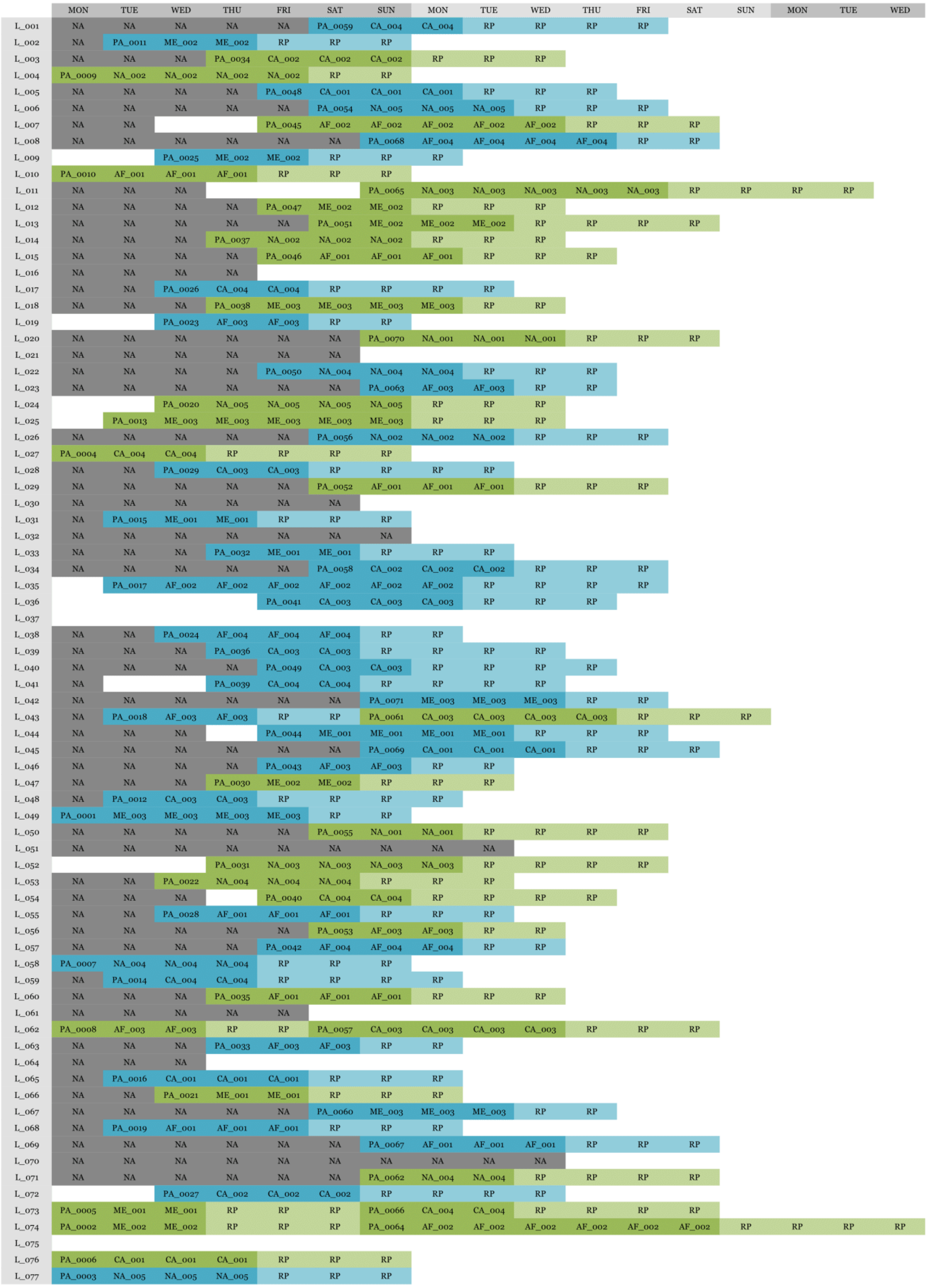


Figure B.1: Visualization of an example solution to the static rostering model

C

Experiment model settings

Table C.1: Model parameter settings for experiment A

Cost parameter	**Cost explanation**	**Initial value**
c_v	Unindexed costs for a void in the roster	100
c_b	Bonus costs for an assigned requested pairing	0
c_r^s	Cost for assigning a slack roster	1000000
c_p	Penalty costs for a covered pairing	50
c_{SSA}	Costs for a schedule start arc	1
c_{PSA}	Costs for a pairing start arc	1
c_{PA}	Costs for a pairing arc	1
c_{PEA}	Costs for a pairing end arc	1
c_{VSA}	Costs for a pre-assignment start arc	1
c_{VA}	Costs for a pre-assignment arc	100
c_{VEA}	Costs for a pre-assignment end arc	1
c_{BSA}	Costs for a base arc	100
c_{SEA}	Costs for a schedule end arc	1

Table C.2: Model parameter settings for experiment B

Parameter	**Parameter explanation**	**Initial value**
$n_{l\,request}$	Number of pairing requests per crew member per batch	5
q	Minimum desired number of requested and assigned pairings in the schedule	0
c_v	Unindexed costs for a void in the roster	100
c_b	Bonus costs for an assigned requested pairing	0
c_r^s	Cost for assigning a slack roster	1000000
c_p	Penalty costs for a covered pairing	50
c_{SSA}	Costs for a schedule start arc	1
c_{PSA}	Costs for a pairing start arc	1
c_{PA}	Costs for a pairing arc	1
c_{PEA}	Costs for a pairing end arc	1
c_{VSA}	Costs for a pre-assignment start arc	1
c_{VA}	Costs for a pre-assignment arc	100
c_{VEA}	Costs for a pre-assignment end arc	1
c_{BSA}	Costs for a base arc	100
c_{SEA}	Costs for a schedule end arc	1

Table C.3: Additional model settings in sensitivity analyses

Sensitivity analysis	**Size of** L	**Size of** P	q_{FTE}^{week}	n_y	c_r^s	c_b	p_{grant}
Sensitivity to the provided set of crew members and requests to be granted (presented in Section 8.1.1)	tested	71	5	tested	10^6	10	-
Sensitivity of costs for slack crew rosters (presented in Section 8.1.2)	72	71	0.426	0	tested	10	-
Sensitivity of bonus costs for pairing request assignment(presented in Section 8.1.3)	72	71	0.426	0	10^6	tested	-
Sensitivity to probability for granting pairing requests (presented in Section 8.2.1)	72	71	0.426	0	10^6	10	tested
Sensitivity to number of submitted requests (presented in Section 8.2.2)	72	71	tested	0	10^6	10	0.04
Sensitivity of rules in rule-based pairing request evaluation algorithm (presented in Section 8.2.3)	72	71	0.426	0	10^6	10	-
Sensitivity of training data in classification-based pairing request evaluation algorithm (presented in Section 8.2.4)	72	71	0.426	0	10^6	10	-

Bibliography

Abbink, E., Fischetti, M., Kroon, L., Timmer, G., and Vromans, M. (2005). Reinventing crew scheduling at netherlands railways. *Interfaces*, 35(5):393–401.

Achour, H., Gamache, M., Soumis, F., and Desaulniers, G. (2007). An Exact Solution Approach for the Preferential Bidding System Problem in the Airline Industry. *Transportation Science*, 41(3):354–365.

Arabeyre, J. P., Fearnley, J., Steiger, F. C., and Teather, W. (1969). The Airline Crew Scheduling Problem: A Survey. *Transportation Science*, 3(2):140–163.

Asta, S., Özcan, E., and Curtois, T. (2016). A tensor based hyper-heuristic for nurse rostering. *Knowledge-Based Systems*, 98:185–199.

Awadallah, M. A., Bolaji, A. L. A., and Al-Betar, M. A. (2015). A hybrid artificial bee colony for a nurse rostering problem. *Applied Soft Computing Journal*, 35:726–739.

Azadeh, A., Farahani, M. H., Eivazy, H., Nazari-Shirkouhi, S., and Asadipour, G. (2013). A hybrid meta-heuristic algorithm for optimization of crew scheduling. *Applied Soft Computing Journal*, 13(1):158–164.

Bard, J. F. and Purnomo, H. W. (2004). Preference scheduling for nurses using column generation. *European Journal of Operational Research*, 164:510–534.

Barnhart, C., Belobaba, P., and Odoni, A. R. (2003a). Applications of Operations Research in the Air Transport Industry. *Transportation Science*, 37(4):368–391.

Barnhart, C., Cohn, A. M., Johnson, E. L., Klabjan, D., Nemhauser, G. L., and Vance, P. H. (2003b). Airline Crew Scheduling. In *Handbook of Transportation Science*, pages 517–560.

Belobaba, P., Odoni, A., and Barnhart, C. (2015). *The Global Airline Industry*.

Biskup, D. and Simons, D. (2004). Common due date scheduling with autonomous and induced learning. *European Journal of Operational Research*, 159(3):606–616.

Boubaker, K., Desaulniers, G., and Elhallaoui, I. (2010). Bidline scheduling with equity by heuristic dynamic constraint aggregation. *Transportation Research Part B: Methodological*, 44(1):50–61.

Burke, E., Kendall, G., and Soubeiga, E. (2003). A Tabu-Search Hyperheuristic for Timetabling and Rostering. *Journal of Heuristics*, 9(6):451–470.

Butchers, E. R., Day, P. R., Goldie, A. P., Miller, S., Meyer, J. A., Ryan, D. M., Scott, A. C., and Wallace, C. A. (2001). Optimized New Zealand Crew Scheduling at Air. *Interfaces*, 31(1):30–56.

Campbell, K. W., Durfee, R. B., and Hines, G. S. (1997). FedEx Generates Bid Lines Using Simulated Annealing. *Interfaces*, 27(2):1–16.

Cappanera, P. and Gallo, G. (2004). A Multicommodity Flow Approach to the Crew Rostering Problem. *Operations Research*, 52(4):583–596.

Caprara, A., Toth, P., Vigo, D., and Fischetti, M. (1998). Modeling and Solving the Crew Rostering Problem. *Operations Research*, 46(6):820–830.

Castillo, I., Joro, T., and Li, Y. Y. (2009). Workforce scheduling with multiple objectives. *European Journal of Operational Research*, 196(1):162–170.

Christou, I. T., Zakarian, A., Liu, J.-M., and Carter, H. (1999). A Two-Phase Genetic Algorithm for Large-Scale Bidline- Generation Problems at Delta Air Lines. *Interfaces*, 29(5):51–65.

Dawid, H., König, J., and Strauss, C. (2001). An enhanced rostering model for airline crews. *Computers & Operations Research*, 28(7):671–688.

Day, P. R. and Ryan, D. M. (1997). Flight Attendant Rostering for Short-haul Airline Operations. *Operations Research*, 45(5):649–661.

Deng, G. F. and Lin, W. T. (2011). Ant colony optimization-based algorithm for airline crew scheduling problem. *Expert Systems with Applications*, 38(5):5787–5793.

Doi, T., Nishi, T., and Voß, S. (2017). Two-level Decomposition based Matheuristic for Airline Crew Rostering Problems with Fair Working Time. *European Journal of Operational Research*, 0:1–11.

Ehrgott, M. (2006). A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operations Research*, 147:343–360.

Eltoukhy, A. E. E., Chan, F. T. S., and Chung, S. H. (2016). Airline schedule planning: a review and future directions. *Industrial Management & Data Systems*, 117(6):1201–1243.

Fahle, T., Junker, U., Karisch, S. E., Kohl, N., Sellmann, M., and Vaaben, B. (2002). Constraint programming based column generation for crew assignment. *Journal of Heuristics*, 8(1):59–81.

Gamache, M. (1998). The Preferential Bidding System at Air Canada. *Transportation Science*, 32(3):246–255.

Gamache, M. (1999). A Column Generation Approach for Large-Scale Aircrew Rostering Problems. *Operations Research*, 47(2):247–263.

Gamache, M., Hertz, A., and Ouellet, J. O. (2007). A graph coloring model for a feasibility problem in monthly crew scheduling with preferential bidding. *Computers & Operations Research*, 34(8):2384–2395.

Guo, Y., Mellouli, T., Suhl, L., and Thiel, M. P. (2006). A partially integrated airline crew scheduling approach with time-dependent crew capacities and multiple home bases. *European Journal of Operational Research*, 171(3):1169–1181.

Gurobi Optimization, L. L. C. (2018). Gurobi Optimizer Reference Manual.

Hanne, T., Dornberger, R., and Frey, L. (2009). Multiobjective and preference-based decision support for rail crew rostering. *IEEE Congress on Evolutionary Computation*, pages 990–996.

Hoffman, K. L. and Padberg, M. (1993). Solving Airline Crew Scheduling Problems by Branch-and- Cut. *Management Science*, 39(6):657–682.

Hojati, M. (2010). Near-optimal solution to an employee assignment problem with seniority. *Annals of Operations Research*, 181:539–557.

Ingels, J. and Maenhout, B. (2017). Optimised buffer allocation to construct stable personnel shift rosters. *Omega*, 0:1–16.

Jarrah, A. I. and Diamond, J. T. (1997). The problem of generating crew bidlines. *Interfaces*, 27(4):49–64.

Jütte, S., Müller, D., and Thonemann, U. W. (2017). Optimizing railway crew schedules with fairness preferences. *Journal of Scheduling*, 20:43–55.

Kasirzadeh, A., Saddoune, M., and Soumis, F. (2017). Airline crew scheduling: models, algorithms, and data sets. *EURO Journal on Transportation and Logistics*, 6(2):111–137.

Kohl, N. and Karisch, S. E. (2004). Airline Crew Rostering: Problem Types, Modeling, and Optimization. *Annals of Operations Research*, 127:223–257.

Konak, A., Coit, D. W., and Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering and System Safety*, 91:992–1007.

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, 31:249–268.

Lee, L. H., Lee, C. U., and Tan, Y. P. (2007). A multi-objective genetic algorithm for robust flight scheduling using simulation. *European Journal of Operational Research*, 177(3):1948–1968.

Lettovský, L., Johnson, E. L., Nemhauser, G. L., and Lettovsk, L. (2000). Airline Crew Recovery. *Transportation Science*, 34(4):337–348.

Lourenço, H. R., Paixão, J. P., and Portugal, R. (2001). Multiobjective Metaheuristics for the Bus Driver

Scheduling Problem. *Transportation Science*, 35(3):331–343.

Lučić, P. and Teodorović, D. (1999). Simulated annealing for the multi-objective aircrew rostering problem. *Transportation Research Part A: Policy and Practice*, 33(1):19–45.

Lučić, P. and Teodorović, D. (2007). Metaheuristics approach to the aircrew rostering problem. *Annals of Operations Research*, 155(1):311–338.

Maenhout, B. and Vanhoucke, M. (2010). A hybrid scatter search heuristic for personalized crew rostering in the airline industry. *European Journal of Operational Research*, 206(1):155–167.

Medard, C. P. and Sawhney, N. (2007). Airline crew scheduling from planning to operations. *European Journal of Operational Research*, 183:1013–1027.

Mohan, S. (2008). Scheduling part-time personnel with availability restrictions and preferences to maximize employee satisfaction. *Mathematical and Computer Modelling*, 48(11-12):1806–1813.

Pedregosa, F., Weiss, R., and Brucher, M. (2011). Scikit-learn : Machine Learning in Python. 12:2825–2830.

Rasmussen, M. S., Justesen, T., Dohn, A., and Larsen, J. (2012). The Home Care Crew Scheduling Problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, 219:598–610.

Rosenberger, J. M., Schaefer, A. J., Goldsman, D., Johnson, E. L., Kleywegt, A. J., and Nemhauser, G. L. (2002). A Stochastic Model of Airline Operations. *Transportation Science*, 36(4):357–377.

Saddoune, M., Desaulniers, G., Elhallaoui, I., and Soumis, F. (2011). Integrated airline crew scheduling: A bi-dynamic constraint aggregation method using neighborhoods. *European Journal of Operational Research*, 212(3):445–454.

Saddoune, M., Desaulniers, G., Elhallaoui, I., and Soumis, F. (2012). Integrated Airline Crew Pairing and Crew Assignment by Dynamic Constraint Aggregation. *Transportation Science*, 46(1):39–55.

Schaefer, A. J., Johnson, E. L., Kleywegt, A. J., and Nemhauser, G. L. (2005). Airline Crew Scheduling Under Uncertainty. *Transportation Science*, 39(3):340–348.

Sherali, H. D., Bae, K.-H., and Haouari, M. (2013). A benders decomposition approach for an integrated airline schedule design and fleet assignment problem with flight retiming, schedule balance, and demand recapture. *Annals of Operations Research*, 210(210).

Smet, P., Bilgin, B., De Causmaecker, P., and Berghe, G. (2014). Modelling and evaluation issues in nurse rostering. *Annals of Operations Research*, 218:303–326.

Souai, N. and Teghem, J. (2009). Genetic algorithm based approach for the integrated airline crew-pairing and rostering problem. *European Journal of Operational Research*, 199(3):674–683.

Stojković, M., Soumis, F., and Desrosiers, J. (2009). The Operational Airline Crew Scheduling Problem. *Transportation Science*, 39(January):2014–2014.

Strevell, M. W. and Chong, P. S. (1985). Gambling on Vacation. *Interfaces*, 15(2):63–67.

Suraweera, P., Webb, G. I., Evans, I., and Wallace, M. (2013). Learning crew scheduling constraints from historical schedules. *Transportation Research Part C: Emerging Technologies*, 26:214–232.

Tanino, T., Tanaka, T., and Inuiguchi, M. (2003). *Multi-Objective Programming and Goal Programming*.

Turner, C., Tiwari, A., Starr, A., and Blacktop, K. (2016). A review of key planning and scheduling in the rail industry in Europe and UK. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 230(3):984–998.

Yen, J. W. and Birge, J. R. (2006). A Stochastic Programming Approach to the Airline Crew Scheduling Problem. *Transportation Science*, 40(1):3–14.

Zeghal, F. M. and Minoux, M. (2006). Modeling and solving a Crew Assignment Problem in air transportation. *European Journal of Operational Research*, 175:187–209.