ROWAN HOOGERVORST

Improving the Scheduling and Rescheduling of Rolling Stock: Solution Methods and Extensions

Improving the Scheduling and Rescheduling of Rolling Stock: Solution Methods and Extensions

Improving the Scheduling and Rescheduling of Rolling Stock: Solution Methods and Extensions

Het verbeteren van het plannen en bijsturen van materieel: oplosmethoden en extensies

Thesis

to obtain the degree of Doctor from the
Erasmus University Rotterdam
by command of the
Rector Magnificus

Prof.dr. A.L. Bredenoord

and in accordance with the decision of the Doctorate Board.

The public defence shall be held on

Thursday 14 October 2021 at 10:30 hours

by

Rowan Hoogervorst
born in Amstelveen, the Netherlands

Erasmus University Rotterdam

Doctoral committee

Promotor:	Prof.dr. D. Huisman
Other members:	Prof.dr. A.P.M. Wagelmans Prof.dr. D. Pisinger Dr. V. Cacchiani
Copromotor:	Dr. T.A.B. Dollevoet

Erasmus Research Institute of Management - ERIM
The joint research institute of the Rotterdam School of Management (RSM)
and the Erasmus School of Economics (ESE) at the Erasmus University Rotterdam
Internet: www.erim.eur.nl

ERIM Electronic Series Portal: repub.eur.nl

ERIM PhD Series in Research in Management, 534
ERIM reference number: EPS-2021-534-LIS
ISBN 978-90-5892-611-1

Cover image: Romana de Vries
Cover design: PanArt, www.panart.nl
Print: OBT bv, www.obt.eu

This publication (cover and interior) is printed on FSC® paper Magno Satin MC.

FSC www.fsc.org MIX Papier van verantwoorde herkomst FSC® C022586

Acknowledgements

My journey towards a PhD started in the autumn of 2015, when Twan asked me during the award ceremony of the Best Econometric Thesis Award (βETA) if I would be interested in starting a PhD. Already having seen the advertisement for this PhD position before, I was immediately enthusiastic. While this conversation with Twan seemed spontaneous at the time, I later learned that Twan and Dennis had already discussed approaching me beforehand. I am very glad that you did and selected me for the position after I had sent in my application.

When I started in September 2016, joining the group of OR PhDs immediately felt like a warm bath. While my office was far away from most of the other OR PhDs, one floor up that is, I immediately felt welcomed. During my time as a PhD, I have enjoyed all the activities that we have organized as a group. Next to the daily lunches, I remember the game nights, the table tennis tournaments, the artistic food sessions, the Serbian lunch, and the many other get-togethers. I also remember vividly all the fun we had at conferences, where we often shared accommodations. Moreover, we had many fruitful discussions about our research and were always willing to help each other out when someone would become stuck. Bart, Charlie, Evelot, Harwin, Jeroen, Kevin, Lisanne, Mathijs, Mette, Naut, Nemanja, Rick, Rolf, Rutger, Sha, Thomas B., Thomas V., Utku, Weina, Ymro, thank you all for the great time we had as OR PhDs!

I would also like to mention the other ESE PhDs with whom I shared a floor in either the Tinbergen building or the E-building and all the ERIM PhDs that I spoke to during the ERIM events. In particular, I am grateful to Anoek (I could hardly resist the temptation to type Anouk here) for sharing an office with me for all these years. While some others called the interior of our office "cold" (not looking at you Kevin), I greatly enjoyed the many conversations we had about work and life in general. Similarly, I would like to mention my other office mates in the E-building, although I have unfortunately hardly been able to see many of you since the start of the pandemic.

During my PhD, I was also able to regularly visit the NS office, specifically the department of PI (π). It was a great pleasure to be part of π during this time, after already meeting many of the people at π during my time there as a Master thesis intern. Not only would visiting Utrecht often be a good moment to catch up on many personal stories, but the regular "taartmomenten" would also be a good moment to get my daily dose of calories. I also remember gladly all the events at π, varying from

the yearly outing to the monthly Ma-Pi-Do-Mi-Bo drinks. Thank you, Anneke, for being so instrumental in all events at π and for always helping out when there were issues with access to the building or accounts. I would also like to mention Gábor here, who is a co-author on two of the papers in this thesis and acted as my NS-supervisor during my PhD. You were always available to help me with working out research ideas and your corrections on my drafts, often leaving more red than white on the page, have always been immensely helpful. In addition, I look back with great joy on all the (long) conversations that we had on programming, mathematics, TAM, and many other (geeky) topics. Furthermore, I am grateful to Joël and Pieter-Jan, who were always glad to discuss rolling stock scheduling and rescheduling.

Another important group of people to mention is the railway group at Erasmus University, which consists of people from both ESE and RSM. I am very appreciative of all the sessions and discussions that we had, where the paper review sessions were, e.g., always very insightful for improving a paper. Moreover, we spent a lot of time together at conferences and seminars. In particular, I would like to mention Gert-Jaap, Johann, Marie, Paul, and Shadi here.

I am also grateful to the (other) colleagues from the Econometric Institute. In particular, Wilco and Christiaan, with whom I have worked together closely in the courses that I have taught. Wilco, being involved in teaching combinatorial optimization was a great way to get to know many of the organizational details of teaching a course and I really appreciated how you allowed me to take more responsibilities over the years! Moreover, being involved in the MOOC in Econometrics was a nice introduction to the world of online courses and a good way to maintain my rusty Econometrics skills. Your enthusiasm for teaching has always been inspiring, Christiaan!

Another great period during my PhD was my research visit to the Zuse Institute Berlin (ZIB). Not only was it refreshing to work in a different department for some time, but I also really liked the atmosphere at ZIB. Next to that, Berlin was a great city to live in for these months. I would like to thank Prof. Ralf Borndörfer at the Network Optimization department for hosting me at ZIB and for our cooperation on two chapters in this thesis. Moreover, I would also like to mention Boris and Stanley with whom I have worked together both during and after my time at ZIB.

I would also like to take this opportunity to thank those people that are essential for this defense. First of all, all the committee members. Albert, David, Gábor, Marie, Ralf, Valentina, thank you for agreeing to be part of the opposition during my defense. Moreover, I am grateful to my paranymphs Rolf and Marjolein for agreeing to stand next to me today and for helping with organizational tasks around my defense.

Most importantly, I should not forget to thank Dennis and Twan, who have been instrumental in guiding me through my PhD. Twan, I quickly got lost in tracking the number of times that you spotted typos in my drafts and punched holes in my proofs. Moreover, you could probably dream the text of our first paper at some point in time. Dennis, your qualities in keeping track of the larger picture and in linking academic content and practical significance have been very useful in shaping this thesis and in keeping me on track. Without you both, the thesis would certainly

not have looked like it does now. I would also like to mention the support of both of you on a personal level, as you were always available to me for questions or advice. In addition, we have had a great time at conferences, seminars, during your visit to Berlin, and during many other get-togethers.

Lastly, I have to thank my friends and family. During my PhD, I have learned how important it is to be able to create a good balance between academic life and private life. Not only for staying mentally fit but also for staying productive. All the support that I have had from family and friends has been essential in this. Romana, my special thanks go out to you for supporting me through all these years, both in easy and hard times. It feels very appropriate to have your design on the cover of this thesis.

Rotterdam, July 2021,
Rowan Hoogervorst

Table of contents

Chapter 1

Introduction

Public transportation plays an important role within the Dutch mobility system. In 2019, before the COVID-19 crisis, about 13% of all traveled kilometers and about 6% of all trips were made by means of public transportation[1]. This makes public transportation the second mode of transportation, in terms of total kilometers traveled, after the car. A large peak in public transportation usage can especially be seen in the early morning and late afternoon hours, due to the large contribution of people commuting to and from work, schools, and universities.

The railway network forms the backbone of the Dutch public transportation network, covering about 75% of the kilometers traveled by public transportation. The Dutch railway system is split into a so-called core network (Dutch: hoofdrailnet) and regional lines. The core network is operated by Netherlands railways (NS, Dutch: Nederlandse Spoorwegen), while the regional lines are divided over multiple railway operators. The Dutch railway network is one of the most intensively used networks in the world, where NS transports about 1.3 million passengers on an average working day.

To provide its services to passengers, NS operates a large fleet of rolling stock units, i.e., railway vehicles. The majority of the rolling stock at NS is comprised of so-called train units, which are self-propelled rolling stock units that do not require the use of a locomotive. As the cost of rolling stock is generally the largest cost component for a railway operator, it is essential to efficiently use these train units. On the one hand, this is done by efficiently scheduling these rolling stock units, such that there is a good balance between the offered capacity for passengers and the operational costs. On the other hand, efficient rescheduling of rolling stock units is necessary when a disruption impacts the railway system.

In this thesis, we focus on both phases that consider the rolling stock: rolling stock scheduling and rescheduling. The main goal of this thesis is to further integrate these processes within the operational context of a railway operator, where we, e.g., look at reducing passenger delays and at rescheduling those drivers that move train units at the station. Moreover, we compare existing models that have been proposed for

[1]https://www.kimnet.nl/publicaties/rapporten/2020/10/28/kerncijfers-mobiliteit-2020

the scheduling of rolling stock and propose a heuristic for the rescheduling of rolling stock.

1.1 The Railway Planning Process

The planning process of a passenger railway operator generally consists of a number of sequential steps, which are depicted in Figure 1.1. Many years before the day of operation, strategic decisions are made on the rolling stock and crew that should be available to execute future services. For the rolling stock, it has to be decided at this phase how large the rolling stock fleet should be and which types of rolling stock should be part of it. These decisions have to be made well in advance due to the purchasing process of new rolling stock units, which often takes several years from the moment of ordering to the moment of delivery. Similarly, also decisions on the required size of the railway crew have to be made well in advance. These decisions are then reflected in the hiring process of the railway operator.

Long-Term Planning
↓
Line Planning
↓
Timetabling
↓
Rolling Stock Scheduling → Shunting Planning
↓
Crew Planning

Figure 1.1: An overview of the railway planning process.

Considering a slightly shorter planning horizon, a plan is then made on which services, i.e. routes, should be operated on the available infrastructure. These routes are referred to as lines, where a line often indicates both the route taken through the infrastructure as well as the frequency with which the line is operated, i.e., how often per hour the route is operated. The result of this planning step is a so-called line plan, which specifies all lines that will be operated. Generally, the goal is to construct a line plan that provides attractive services to passengers to travel from their origin to destination station and which is also likely to have low operational costs. At NS, a major overhaul of the line plan is made about every 7-10 years, which often leads to significant changes to the journeys made by passengers.

After a line plan has been found, a timetable has to be created. Such a timetable specifies the exact departure time of each train that is part of the line plan at each of the stations that is visited by the train. These departure times in the timetable have to be chosen such that the timetable can be safely operated on the railway infrastructure, meaning that there should, e.g., be a minimum time between any two trains that use the same piece of railway infrastructure. Moreover, one generally wants to find a timetable that allows for easy connections for passengers between those trains that likely have a large number of transferring passengers. At NS, the timetable is cyclic, meaning that the same timetable is operated every hour. Moreover, trains are generally spread evenly over the hour, meaning that most trains leave every ten, fifteen, or thirty minutes. A new timetable is created every year and smaller updates to the timetable are made a couple of times a year.

The timetable then forms the input for the scheduling of rolling stock. In this phase, rolling stock is assigned to the trips in the timetable and it is thus determined how much capacity, in terms of the available number of spots in a train, will be offered on each trip. The output of this phase is often referred to as a rolling stock schedule or rolling stock circulation. We explore the details of this problem in Section 1.3.

Rolling stock scheduling focuses on the assignment of rolling stock at the network level. Based on this assignment, plans then have to be made for the movement of rolling stock within station areas, which are often referred to as railway nodes. For example, when a train unit ends its service at a given station it is generally parked at the shunting yard of that station. In the shunting planning, a path then has to be found from the arrival track of the train unit to the shunting yard. Moreover, a driver needs to be found to drive this train unit from the station to the shunting yard and a location needs to be found at the shunting yard to park the train. During the time that train units are parked at the shunting yard, also the servicing of trains needs to be planned, which includes maintenance to the train units as well as cleaning of the train units. As decisions made in the shunting planning of a given station do not influence those of other stations, shunting planning is done independently for each station at NS.

The rolling stock schedule determines also the required number of crew members, as different rolling stock units generally need a different number of guards. In the crew planning phase, duties are then assigned to the drivers and guards such that all tasks are covered. These duties describe exactly the tasks executed by a single driver or guard during the day, where each duty has to satisfy certain labor conditions. Moreover, it is important to also find duties that are attractive to the crew. Together, the found duties are often referred to as a crew schedule. At NS, the found duties are generally not immediately assigned to individual crew members. Instead, each crew member has a roster that cycles through the duties that are executed by the crew members of his or her depot, i.e., home station. Allocating the duties to the rosters in an attractive and fair way is referred to as crew rostering.

1.2 The Railway Rescheduling Process

While all plans are generally created such that they allow for some robustness to changes, it is impossible to prevent disruptions during the day of operation. These disruptions can have a wide variety of causes. A failing piece of infrastructure could, e.g., lead to the blockage of the railway infrastructure between two stations. Alternatively, a failure on a train unit may, e.g., cause a train to build up a delay. As a result of these disruptions, the original plans often become infeasible. In the rescheduling process, it is then required to adjust the plans based on the faced disruption.

The rescheduling process follows to a large extent the same steps as the planning phase, except for the strategic decisions taken in the planning phase. The steps of the rescheduling process are shown in Figure 1.2. Here, the first step in the rescheduling process is to adjust the timetable according to the disruption. In case of large disruptions, this is done through so-called disruption scenarios, which are predetermined plans that specify exactly which trips have to be canceled or rescheduled. In case of smaller disruptions, it is often up to a timetable dispatcher to make the required changes in the timetable.

$$\begin{tikzcd}
\text{Timetable Rescheduling} \arrow[d] & \\
\text{Rolling Stock Rescheduling} \arrow[r] \arrow[d] & \text{Shunting Rescheduling} \\
\text{Crew Rescheduling} &
\end{tikzcd}$$

Figure 1.2: An overview of the railway rescheduling process.

Based on an updated timetable, new rolling stock plans then have to be created. In particular, it has to be determined how the rolling stock now moves through the trips in the updated timetable. Moreover, the unavailability of rolling stock may sometimes require additional cancellations of trips in the timetable. Any changes to the rolling stock schedule also have an impact again on the shunting that happens at stations, which may require the rescheduling of the local plans at these stations. For example, the locations at which train units are parked at the station yard might have to be rescheduled, as well as the duties of those drivers that move train units between the station and its shunting yard.

The changes made to the rolling stock schedule also have an impact on the duties of the crew. In particular, disruptions often cause drivers and guards to be unable to reach their intended destination on a particular train. As a result, these crew members are not able to execute the remainder of their planned duty and may not reach their crew depot at the end of their duty. In crew rescheduling, it is then needed to reschedule the duties of the crew such that enough crew is available to operate each train and such that the crew can reach their intended destination before the end of their duty.

1.3 Rolling Stock (Re)Scheduling at NS

Let us now take a closer look at the scheduling and rescheduling of rolling stock at NS. Key to the (re)scheduling of rolling stock at NS is that NS operates a fleet consisting of predominantly self-powered train units, instead of cars being pulled by a locomotive. These train units can be of different types, where the type determines, among others, the number of seats available within the train unit for passengers. An example of a train unit of the so-called ICM-III type is shown in Figure 1.3.

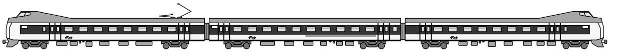


Figure 1.3: Example of a train unit of the ICM-III type.

Train units can be combined into so-called compositions. Doing so increases the number of seats available for passengers on a trip. Moreover, operating a composition consisting of multiple train units generally requires fewer resources than operating the train units independently, for example, in terms of required crew and required infrastructure capacity. An example of a composition is given in Figure 1.4, in this case consisting of a so-called ICM-III and ICM-IV train unit. At NS, one generally looks at only the type of the train units in a composition when scheduling rolling stock. Individual train units of the required type are then scheduled at a later moment, for example taking into account the maintenance that is required for each train unit.

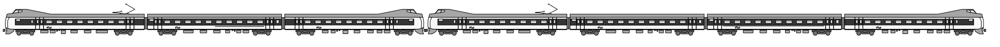


Figure 1.4: Example of a composition consisting of a ICM-III unit and a ICM-IV train unit.

At NS, fixed transitions between trips are defined beforehand which indicate how rolling stock moves from one trip to another. In case no such transition is defined, all rolling stock moves to the shunting yard of a station. The composition of trips can be changed at these transitions, during which a train is at the platform of a station. Train units can either be uncoupled from the composition or coupled to the composition. Train units that are uncoupled are generally moved to the shunting yard of the station and coupled train units are then taken from the shunting yard again. Not all composition changes are generally possible at each station, for example as a result of the shunting yard of a station only being accessible from one side of the station.

The size of the fleet that is available to operate trips is fixed in both the scheduling and rescheduling phase. This means that a fixed number of train units of each type start at each station at the beginning of the planning horizon. Moreover, there is generally a target number of train units of each type that are expected to end at each station. This is done to ensure that enough train units are available to start up the timetable at the start of the next planning period. As rolling stock scheduling is generally done for one day at a time, this then corresponds to having enough train units to start up the timetable of the next day.

The rolling stock scheduling problem can now be formulated as the problem of allocating sequences of tasks, i.e., trips, to each available train unit, such that the compositions that are formed on trips and composition changes made during transitions are feasible. The objective of the problem is in general to find a balance between passenger service on the one hand, and the operational costs on the other. The former is mostly concerned with offering enough seats to passengers on trips, to prevent that passengers have to stand or cannot board the train. The latter considers, for example, the cost of operating the train units on trips, of coupling and uncoupling train units during transitions, and of deviating from the planned number of train units to end at each station.

An example of a rolling stock schedule, or rolling stock circulation, is given in Figure 1.5. The timetable is given here as a time-space diagram, in which the horizontal axis shows the time and the vertical axis shows the different stations. A line between two stations then gives a trip between these two stations. Note that the circulation indicates the location of the individual train units through time, meaning that multiple lines are present for a trip if more than one train unit is part of the composition on that trip. This is, e.g., the case for trips t_1 and t_2, which are operated by a composition consisting of a red (solid) and blue (dashed) train unit. Moreover, also the shunting of train units can be seen in this example, where a red train unit is, e.g., uncoupled from the composition of trip t_2 and is coupled again to another composition before the start of trip t_7.

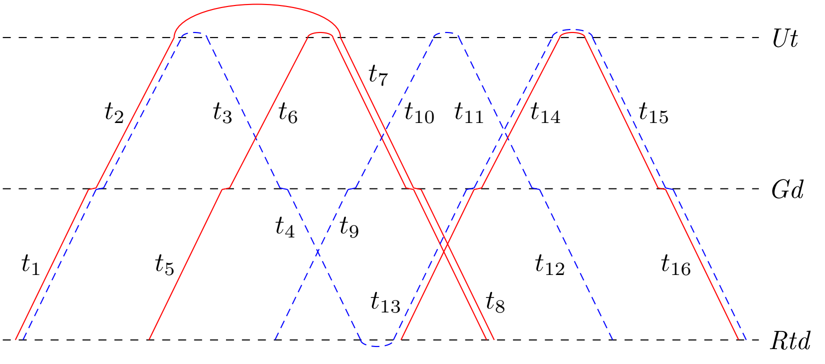


Figure 1.5: Example of a rolling stock circulation.

The rolling stock rescheduling problem is overall very similar to the rolling stock scheduling problem and often contains the same restrictions. The main difference is in the objective, where in rolling stock rescheduling one wants to find a new rolling stock circulation that is not too different from the original circulation. By doing so, the number of changes in the rolling stock assignment that have to be communicated to the crews is limited, as well as the changes that have to be made to the duties of the crew. The complete objective then balances the objectives traditionally used in rolling stock scheduling with this objective of not making too many changes.

1.4 Contributions

The contributions of this thesis are fourfold. First, we compare existing models that have been proposed for rolling stock scheduling. Many of these models are also used within the rescheduling of rolling stock. In Chapter 2, we compare these models by looking at the operational context that is considered in each model. Moreover, we make an analytical and numerical comparison between two of these models: the Composition model proposed by Fioole et al. (2006) and the Hypergraph model proposed by Borndörfer et al. (2016). Our analytical results show that both models can consider the operational setting as considered in Fioole et al. (2006) and that the linear programming relaxation achieved by both models is equal in this setting. However, our numerical results show that the additional modeling power of the Hypergraph model leads to longer running times when considering instances of NS.

Second, we propose a new heuristic for the rolling stock rescheduling problem in Chapter 3. This heuristic is motivated by the observation that it is generally hard to include all of the details that are considered by rolling stock dispatchers during real-time rescheduling into exact solution methods for rolling stock rescheduling. Hence, we propose a Variable Neighborhood Search heuristic for rolling stock rescheduling and introduce several new local search neighborhoods for this problem. Moreover, we show that extensions of the rolling stock rescheduling problem can easily be included in the heuristic by introducing additional neighborhoods. Numerical experiments on instances of NS show that the heuristic can find high-quality solutions quickly and is able to outperform the Composition model of Fioole et al. (2006) when considering an extension of the rolling stock rescheduling problem.

Third, we extend the rolling stock rescheduling problem to consider the delays faced by passenger in Chapter 4 when a delay disturbs the railway system. Our aim is then to reschedule the rolling stock such that the passenger delays are minimized, while also considering the traditional rolling stock rescheduling costs. For this problem, we propose two novel solution methods, one based on a column generation approach and the other being an extension of the Composition model proposed by Fioole et al. (2006). Our results show that rolling stock rescheduling can lead to substantial delay reductions, especially when allowing some freedom in choosing the turnings at terminal stations. Moreover, we show that the model which extends the Composition model performs the best of the two proposed models.

Fourth, we investigate in Chapter 5 how rolling stock rescheduling can be integrated with the rescheduling of shunting drivers who move train units at the stations. This is motivated by the observation that solving these problems sequentially may lead to infeasibilites in case not enough shunting drivers are available to execute the shunting performed in the rolling stock circulation. To solve the integrated problem, we propose two new solution methods, one based on Benders decomposition and the other on a flow-based approach. Our results show that this integrated problem can be solved well for large-scale instances and that doing so prevents a significant number of infeasibilities at the stations. In addition, our results highlight that while both methods perform competitively, the flow-based approach can overall find optimal solutions more quickly.

1.5 Thesis Overview

In this thesis, we focus both on the scheduling and rescheduling of rolling stock. Moreover, we consider two different extensions of the rolling stock rescheduling problem: one focusing on reducing delays by rolling stock rescheduling and the other on rescheduling the rolling stock and shunting drivers in an integrated way. A schematic overview of how these topics are discussed in the chapters of this thesis is given in Figure 1.6.

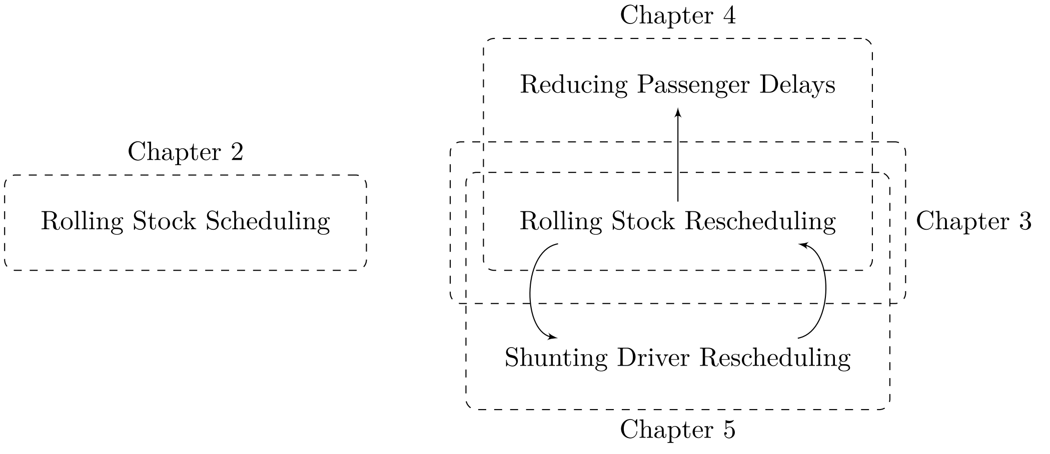


Figure 1.6: An overview of the chapters in this thesis.

Chapter 2 focuses on the rolling stock scheduling problem, where we compare models proposed in the literature for this problem. However, many of the models discussed in this chapter have also been applied to the rescheduling setting discussed in the other chapters. The other chapters all focus on a rolling stock rescheduling setting. In Chapter 3, we discuss a Variable Neighborhood Search (VNS) heuristic for rolling stock rescheduling. In Chapter 4, we then show how rolling stock rescheduling can be used to reduce the delays faced by passengers. In Chapter 5, we look at another extension of the rolling stock rescheduling problem. Here, we consider the integrated problem of rescheduling the rolling stock and the shunting drivers who are responsible for moving trains at the stations.

The chapters in this thesis can be read independently, as each chapter is self-contained. However, as Chapter 2 discusses the rolling stock scheduling problem in more generality and discusses many of the models that have been proposed to solve this problem, readers that are unfamiliar with rolling stock scheduling and rescheduling are advised to start with this chapter. Similarly, Chapter 3 provides a general introduction to the rolling stock rescheduling problem, meaning that this chapter can serve as a good introduction to Chapters 4 and 5. These last two chapters can be read in any particular order, as both consider independent extensions of the rolling stock rescheduling problem.

In the remainder of this section, we give a short summary of Chapters 2 to 5. All of these chapters are based on an academic paper or correspond to a paper that we are currently preparing to submit to a journal. The author of this thesis has been the main author in the work of Chapters 3 to 5, performing the majority of the work for both the numerical experiments and the writing of the paper. The majority of the work in Chapter 2 was done in conjunction with Boris Grimm, where Ralf Borndörfer helped in the conceptualization and proofreading of the paper.

Chapter 2: B. Grimm, R. Hoogervorst, and R. Borndörfer (n.d.). "An Analytical Comparison of Models for Rolling Stock Scheduling". In preparation for journal submission.

A major step in the planning process of passenger railway operators is that of assigning rolling stock, i.e., train units, to the trips in the timetable. A rolling stock assignment should preferably satisfy the passenger demand, but should also be attractive from the perspective of the railway operator. To support railway companies in scheduling their rolling stock, the literature has proposed a wide variety of models. These vary as a result of operational differences, and hence different requirements, between the considered railway companies. In this chapter, we categorize the existing models according to these requirements and show the relations between them. Moreover, we make an analytical comparison between two models that have been proposed for the setting of Netherlands Railways (NS) and DB Fernverkehr AG (DB), respectively. Our analysis shows that these formulations lead to the same linear programming relaxation bound when considering the rolling stock scheduling setting of NS. Moreover, a numerical comparison shows that the model proposed for NS can find optimal solutions in shorter running times in this setting.

Chapter 3: R. Hoogervorst, T. Dollevoet, G. Maróti, and D. Huisman (2021). "A Variable Neighborhood Search heuristic for rolling stock rescheduling". *EURO Journal on Transportation and Logistics* 10, 100032.

We present a Variable Neighborhood Search heuristic for the rolling stock rescheduling problem. Rolling stock rescheduling is needed when a disruption leads to cancellations in the timetable. In rolling stock rescheduling, one must then assign duties, i.e., sequences of trips, to the available train units in such a way that both passenger comfort and operational performance are taken into account. For our heuristic, we introduce three neighborhoods, which focus on swapping duties between train units, on improving the individual duties, and on changing the shunting that occurs between trips, respectively. These neighborhoods are used for both a Variable Neighborhood Descent local search procedure and for perturbing the current solution in order to escape from local optima. Moreover, we show that the heuristic can be extended to the setting of flexible rolling stock turnings at ending stations by introducing a fourth neighborhood. We apply our heuristic to instances of Netherlands Railways (NS). The results show that the heuristic is able to find high-quality solutions within one minute of solving time. This allows rolling stock dispatchers to use our heuristic in real-time rescheduling.

Chapter 4: R. Hoogervorst, T. Dollevoet, G. Maróti, and D. Huisman (2020). "Reducing passenger delays by rolling stock rescheduling". *Transportation Science* 54.3, 762–784.

Delays are a major nuisance to railway passengers. The extent to which a delay propagates, and thus affects the passengers, is influenced by the assignment of rolling stock. We propose to reschedule the rolling stock in such a way that the passenger delay is minimized and such that objectives on passenger comfort and operational efficiency are taken into account. We refer to this problem as the Passenger Delay Reduction Problem (PDRP). We propose two models for this problem, which are based on two dominant streams of literature for the traditional Rolling Stock Rescheduling Problem. The first model is an arc formulation of the problem, while the second model is a path formulation. We test the effectiveness of these models on instances of Netherlands Railways (NS). The results show that the rescheduling of rolling stock can significantly decrease the passenger delays in the system. Especially allowing flexibility in the assignment of rolling stock at terminal stations turns out to be effective in reducing the delays. Moreover, we show that the arc formulation based model performs best in finding high-quality solutions within the limited time that is available in the rescheduling phase.

Chapter 5: R. Hoogervorst, R. Borndörfer, T. Dollevoet, and D. Huisman (n.d.). "Integrated Rolling Stock and Shunting Driver Rescheduling". In preparation for journal submission. Awarded second place in the 2020 RAS Student Paper Competition.

In rolling stock rescheduling, we assign compositions of train units to the trips in the timetable. These compositions can be changed at stations through shunting movements, which are executed by shunting drivers. When changes are made to the rolling stock assignment, and thus to the shunting tasks that need to be performed, the duties of the shunting drivers have to be rescheduled as well. Traditionally, the rescheduling of rolling stock and shunting drivers happens sequentially. This leads to infeasibilities when not all shunting tasks can be performed by the available shunting drivers at one of the stations. To overcome such infeasibilities, we propose to reschedule the rolling stock and the shunting drivers in an integrated way. We propose two solution methods for this integrated problem. The first is a Benders decomposition algorithm, where the Benders master problem and subproblem correspond to the rescheduling of the rolling stock and the shunting drivers, respectively. The second is an arc-based model that is solved through a commercial solver. We test the solution methods on instances of Netherlands Railways (NS), where we find that the integrated problem can be solved quickly and that doing so prevents a significant number of infeasibilities.

Chapter 2

A Comparison of Models for Rolling Stock Scheduling

2.1 Introduction

Decision support tools based on optimization methods have proven their value throughout different stages in the planning process of passenger railway operators. One of the problems that have benefited substantially from mathematical optimization methods is that of rolling stock scheduling, in which we assign rolling stock to the trips in the timetable. Optimization methods have shown to reduce the cost of the found rolling stock schedules for passenger railway operators, while at the same time increasing passenger satisfaction by offering a better match of supply and passenger demand.

The literature offers a diverse set of models for rolling stock scheduling. To a large extent, this diversity of models is due to the differences between the studied railway companies, both in terms of the available rolling stock and the operational constraints. Examples include differences in the planning horizon, being cyclical or non-cyclical, and the extent to which maintenance of train units needs to be taken into account in the planning stage. As a result of these differences in the problem setting, different concepts and ideas have been used throughout the literature to model these problems.

This diversity in models has also led to a wide variety of solution methods to be applied. While some models are solved directly by a commercial mixed-integer linear programming (MILP) solver (Fioole et al. 2006), others are solved by column generation techniques (Borndörfer et al. 2016; Lusby et al. 2017) or by Lagrangian relaxation (Cacchiani et al. 2013). Moreover, heuristics have been proposed, both originating from a planning (Cacchiani et al. 2019) and rescheduling perspective (Chapter 3 of this thesis).

As a result of the diverse models and solution techniques that have been proposed, several streams can be categorized in the literature. Each of these streams focuses on one of the rolling stock scheduling settings and often uses one base model. An example is the stream of papers focusing on the setting of NS, see, e.g., Wagenaar et al. (2017a) and Kroon et al. (2015), which build on the model of Fioole et al. (2006). Similarly, multiple papers focus on the Train Unit Assignment Problem (TUAP) as proposed by Cacchiani et al. (2010) and extend the model proposed by these authors.

While many of the core ideas between these streams of papers are similar, subtle differences in the problem setting often lead to substantial formulation differences and even the need for applying different solution methods. As a result, it is often unclear which of the existing models and solution approaches is the most suitable for a given rolling stock scheduling problem. Moreover, overall little is known about the relative performance between the models. An exception is the comparison made by Haahr et al. (2016) between the models of Fioole et al. (2006) and Lusby et al. (2017). However, this comparison was based solely on numerical results and not on an analytical comparison between the models.

In this chapter, we focus on categorizing the existing papers and on analytically comparing two commonly used rolling stock scheduling models: the Composition model of Fioole et al. (2006) and the Hypergraph model of Borndörfer et al. (2016). Our contributions in this chapter are fourfold. First, we categorize the models that have been proposed in the literature for rolling stock scheduling based on the operational context that they consider. Second, we show that the Hypergraph model generalizes the Composition model. Third, we show that the strength of the formulations of these two models is equal for the rolling stock scheduling setting of NS, meaning that both models provide the same linear programming relaxation value in this setting. Fourth, we compare the two models through numerical experiments for instances of NS that are typically solved by the Composition model.

The remainder of the chapter is organized as follows. In Section 2.2, we define a general version of the rolling stock scheduling problem. In Section 2.3, we categorize the existing rolling stock scheduling models based on several operational requirements. In Section 2.4, we give the formulations of the Composition model and Hypergraph model. In Section 2.5, we compare these models analytically. Moreover, we perform a numerical comparison between these models in Section 2.6. Finally, we conclude the chapter in Section 2.7.

2.2 The Rolling Stock Scheduling Problem

In the Rolling Stock Scheduling Problem (RSSP), we assign rolling stock to the trips in the timetable. Each trip indicates that a train drives from a departure station to an arrival station, with a fixed departure and arrival time. Some of the trips are naturally joined together into timetable services, which define a train service between two terminal stations of a railway line. At these terminal stations rolling stock then moves from one timetable service to another through so-called turnings.

The rolling stock that is available to operate the trips differs per railway operator. In the RSSP, we assume rolling stock that is composed of self-propelled bi-directional train units, so without a separate locomotive, as is the case for many European passenger railway operators. These train units can, generally, be coupled together to form *compositions*, which differentiates rolling stock scheduling from, e.g., vehicle or airplane scheduling. Combining train units allows for more passenger capacity on a trip. Moreover, it generally requires less energy, crew members, and infrastructure capacity to run a coupled multiple unit train than running each of the units individually.

The compositions of train units are generally not fixed throughout the day, but can be changed at some of the stations at which a train stops. This allows adjusting the offered capacity to varying passenger numbers over the day. Composition changes may happen at either some in-between station or a terminal station of the timetable service. Adding train units to a composition is often referred to as *coupling* of train units, while removing train units is referred to as *uncoupling* of train units. Jointly, such coupling and uncoupling actions are referred to as *shunting* of train units.

When determining how a train unit moves from one trip to another, we need to satisfy business rules that are dependent on the operator. In some cases, trips have an assigned successor trip and train units can only move from the predecessor trip to the successor trip, unless the train unit is uncoupled from the composition. This is especially the case when two trips are part of the same timetable service. In other cases, train units can move between any trips as long as this is possible regarding time and location of the trips. In particular, *deadheading*, i.e., running the train without any passengers, may be allowed to move from the arrival station of one trip to the departure station of the other.

Additional conditions may need to be satisfied when determining the trips that a train unit can operate during the planning horizon. For example, maintenance restrictions often need to be satisfied, such as a maximum distance between two consecutive maintenance appointments. In addition, restrictions generally apply to the compositions that are allowed for trips and the ways in which these compositions can be changed. The length of a composition can, for example, be limited by the platform length of the stations that are passed, while coupling and uncoupling can often only happen at one side of the composition due to the station layout.

In the RSSP, we are now asked to find an assignment of the available train units to the trips in the timetable. The path for each train unit that is implied by this assignment should satisfy all the above-mentioned constraints. In addition, the composition that is chosen for each trip in this assignment should be feasible, as well as the ways in which the compositions are changed at stations. The result of the RSSP is referred to as a rotation, or circulation, for the rolling stock units.

An example of a rolling stock circulation is shown in Figure 2.1. In this figure, the rotation is shown on a torus, where each snapshot of the torus shows the location of each train unit at that moment in time. Each blue line then shows a train unit moving from one station to another. Moreover, yellow lines indicate the turnings that are made by the train units. In this case, a cyclic time horizon is considered, which implies that the end of the planning horizon is directly connected to the start of the planning horizon.

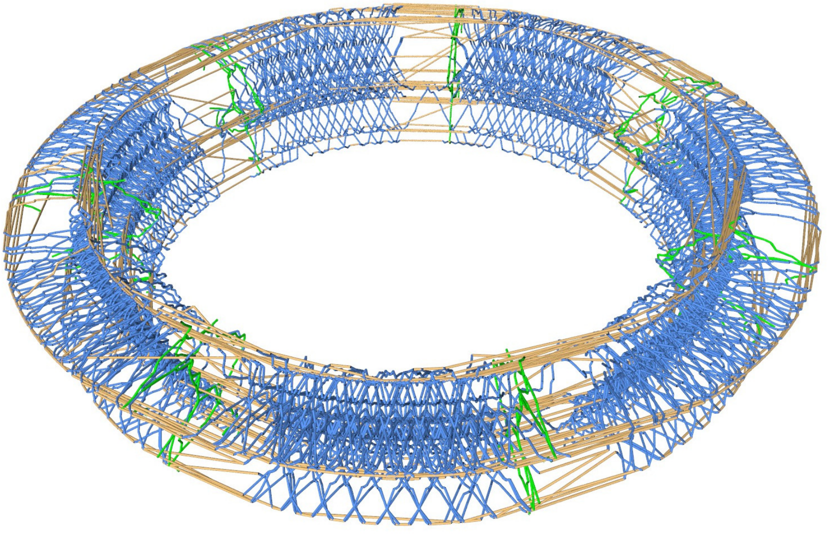


Figure 2.1: Example of a circulation.

2.3 Literature Review

The planning process of a railway passenger operator is generally composed of a few sequential planning steps. In each of these steps, a single operational problem is considered and the solution of this problem is the input to the next planning step. An overview of the typical planning steps is given by, among others, Huisman et al. (2005) and Caprara et al. (2007). In both of these papers, rolling stock scheduling is performed after a timetable has been found and before the crew, i.e., drivers and guards, are scheduled. In this chapter, we focus solely on the phase of rolling stock scheduling.

Over the years, multiple models have been proposed for the rolling stock scheduling problem. To a large extent, the differences between these models are a result of the differences in the operational context of the considered railway passenger operators. These arise, e.g., from differences in the available rolling stock fleet, as well as from differences in the national (infrastructural) context. In the remainder of this section, we categorize the available models according to five main properties of the operational context that these models consider. A schematic overview of this categorization is given in Table 2.1.

2.3.1 Compositions

A first difference between the models concerns the level of detail that is considered in modeling the compositions that train units are in. On the one hand, Cacchiani et al. (2010), Lusby et al. (2017), Lin and Kwan (2016) and Cadarso and Marín (2011) consider only the number of train units of each type in a composition, but not the order of train units within a composition. Such models are mainly used in a setting where there is only a single train unit type or when the compositions are short and where uncoupling and coupling can happen on both sides of the composition.

Models taking into account the exact order of the train units within a composition include those of Fioole et al. (2006) and Peeters and Kroon (2008). Note that these models do not track the individual train units, but do track what type of train unit is in each position of the composition. Taking into account the order of train units is important in these operational contexts, as coupling and uncoupling of train units is often only allowed on one side of the composition due to the station layout. Hence, it is needed to track the order of train units in the composition to determine if a certain train unit can be coupled or uncoupled. A similar restriction is taken into account by Lin and Kwan (2014), who determine the order of the train units in a composition in a second optimization step after determining in a first optimization step how many train units of each type are in a composition.

Finally, Borndörfer et al. (2016) do not only take the order of train units in a composition into account, but also the orientation of the train units themselves, i.e., which side of each train unit is facing in the direction of travel. This is done as infrastructure requirements in the setting of DB prevent some orientations of the train units in a composition and as the orientation of the individual train units is of importance for the seat reservations of the passengers. Note that also the models of Fioole et al. (2006) and Peeters and Kroon (2008) could take these orientations into

Table 2.1: Comparison of the models in the literature based on different characteristics of the rolling stock scheduling problem: the extent to which details of the compositions are considered, the flexibility available in turnings between trips, the way in which maintenance is taken into account, the way in which passenger capacity is considered, and a few other characteristics. Parentheses around a check mark symbol indicate that a paper partially fulfills a certain requirement.

Paper	Compos.			Turnings		Mainten.		Passengers			Other				
	Number	Ordering	Orientation	Fixed	Flexible	Distance	Appointment	Constraint	Objective	Dynamic	Deadheading	Robustness	Core Model	Sol. Method	Company
Fioole et al. (2006)		✓		✓					✓					MILP solver	NS
Peeters and Kroon (2008)		✓		✓					✓					B&P	NS
Cacchiani et al. (2010)	✓				✓	(✓)		✓			✓			Col. Gen. Heur.	Regional
Cadarso and Marín (2011)	✓				✓				✓		✓	(✓)		MILP solver	RENFE
Cacchiani et al. (2012)		✓		✓					✓			✓	Fioole et al. (2006)	Benders Heur.	NS
Nielsen et al. (2012)		✓		✓					✓				Fioole et al. (2006)	MILP solver	NS
Cacchiani et al. (2013)	✓				✓			✓			✓		Cacchiani et al. (2010)	Lagr. heuristic	Regional
Cadarso and Marín (2014)		✓			✓				✓		✓	(✓)	Cadarso and Marín (2011)	Benders Decomp.	RENFE
Giacco et al. (2014)					✓	✓					✓			MILP solver	Trenitalia
Lin and Kwan (2014)		✓			✓			✓			✓			2-stage approach	Southern
Kroon et al. (2015)		✓		✓						✓			Fioole et al. (2006)	MILP solver	NS
Thorlacius et al. (2015)		✓			✓	✓		✓			✓			Heuristic	DSB S-tog
Borndörfer et al. (2016)			✓		✓	✓		✓			✓			B&P	DB
Lin and Kwan (2016)	✓				✓			✓			✓		Lin and Kwan (2014)	B&P	Scotrail
Grimm et al. (2017)			✓		✓	✓					✓		Borndörfer et al. (2016)	B&P	DB
Lusby et al. (2017)	✓			✓		(✓)			✓					B&P	DSB S-tog
Wagenaar et al. (2017b)		✓		✓					✓		✓		Fioole et al. (2006)	MILP solver	NS
Wagenaar et al. (2017a)		✓		✓			✓		✓				Fioole et al. (2006)	MILP solver	NS
Cacchiani et al. (2019)	✓				✓			✓			✓			Heuristic	Regional
Zhong et al. (2019)		✓		✓		✓		✓					Fioole et al. (2006)	Matheuristic	CSHR

account, but in those models it would be needed to create a separate composition for each of the possible orientations of the train units. This strongly inflates the size of these models.

2.3.2 Rolling Stock Turnings

A second distinction between the models is in the way that turnings between trips are determined. On the one hand, these turnings can be determined in advance. In that case, each trip has a fixed follow-up trip, unless all train units on the trip are parked at the station after the trip. All train units that are part of the composition on the predecessor trip then move to the fixed successor trip or are uncoupled from the composition and can only be coupled to another trip again after some fixed reallocation time. Such a setting is considered, e.g., by Fioole et al. (2006) and Lusby et al. (2017). Fixing the turnings is especially common in high-density networks, where there are limited possibilities to execute shunting at the stations.

Alternatively, determining the rolling stock turnings can be part of the decisions that are made in the model. Examples include the models of Cacchiani et al. (2010), Giacco et al. (2014), Lin and Kwan (2014) and Borndörfer et al. (2016). For these models, possible turnings are determined based on business rules that state which turnings are feasible and acceptable for the operators. The model can then determine which of these turnings to take. Note that in these models each train unit in the composition can often follow its own turning, meaning that a composition is split into smaller parts that each turn to a different trip. The advantage of allowing for flexible turning possibilities is that more efficient rolling stock circulations can be found, although often at the expense of more irregular turning patterns at the stations.

2.3.3 Maintenance Requirements

An important restriction when scheduling the rolling stock at many railway operators is that maintenance requirements need to be satisfied. Such requirements ensure that enough maintenance is performed to adhere to safety and quality standards. Some papers, like those of Fioole et al. (2006), Nielsen et al. (2012), Cadarso and Marín (2011) and Cacchiani et al. (2012), do not consider any maintenance requirements. In such models, it is often assumed, either implicitly or explicitly, that maintenance can be scheduled after a circulation has been found. This is, for example, the case when maintenance can be executed during the evening hours when only few trains are operated. Alternatively, maintenance may be scheduled closer to the day of operation during the time that a train unit is parked at the shunting yard of a station.

Other papers have considered distance or time-based maintenance requirements, where train units need to undergo maintenance after a certain amount of use. These include the models of Borndörfer et al. (2016), Lusby et al. (2017) and Cacchiani et al. (2010). The approach taken to include such requirements differs between these models. While Lusby et al. (2017) consider an individual path for each train unit through the network, Borndörfer et al. (2016) consider an additional flow to keep track of the distance traveled by each train unit. An almost identical approach to the latter one is chosen in Giacco et al. (2014), who investigate the benefit of additional deadheading options to reach maintenance facilities in order to decrease the total maintenance hours of a rotation or roster. A different approach is taken by Cacchiani et al. (2010), who require that a certain fraction of the found train unit paths through the network allow for a maintenance activity.

A further type of maintenance requirement is considered by Wagenaar et al. (2017a), where some of the train units have a fixed maintenance appointment. Such an appointment specifies both the location and time at which the maintenance takes place and is usually planned close to the day of operation. Wagenaar et al. (2017a) propose and compare three different models for rolling stock rescheduling to ensure that train units make these appointments.

2.3.4 Passenger Capacity

Another difference between the models is the way in which the passenger demand is taken into account. On the one hand, passenger demand can be enforced through a constraint on the required passenger capacity, i.e., a minimum number of seats per trip. This is done by, e.g., Cacchiani et al. (2010), Borndörfer et al. (2016), Lin and Kwan (2016) and Thorlacius et al. (2015). In these models, the main focus is then on minimizing the costs of the railway operator given these capacity constraints. Note that many of these models can be extended to also include passenger demand as an objective.

Alternatively, some models explicitly take into account the passenger demand by penalizing shortages of capacity. Examples include the models of Fioole et al. (2006), Cadarso and Marín (2011) and Lusby et al. (2017), where any shortages of seats compared to the expected passenger demands are penalized in the objective function. Note that these papers use fixed passenger demands, where the flow of passengers does not change based on the chosen rolling stock assignment. Kroon et al. (2015) instead consider flexible passenger flows for a rescheduling setting in which the capacity of some trains may not be sufficient to accommodate all passengers. As the interaction between rolling stock assignment and passenger flows is hard to include in a single MILP model, these authors consider an iterative framework to solve this problem that alternates between rolling stock rescheduling and the rerouting of passengers.

2.3.5 Other Differences

The characterization above is certainly not complete concerning the differences between the models. In Table 2.1 we state a few other differences. One of them is the inclusion of deadheading trips, which allow rolling stock to be moved from one station to another without any passengers. In some models„ such as those of Fioole et al. (2006) and Lusby et al. (2017), deadheading is not directly considered. Instead, deadheading can only occur in those models in case it has been planned beforehand. Models that do allow to assign deadheading as part of the model include those of Giacco et al. (2014), Cadarso and Marín (2011), Borndörfer et al. (2016) and Wagenaar et al. (2017b). In these models, deadheading can be used either to move between trips in a turning or to move to a maintenance location.

Another difference is the inclusion of some form of robustness, which ensures that a good rolling stock circulation is found even when some of the details regarding the rolling stock scheduling problem are uncertain. For example, Cacchiani et al. (2012) propose a robust two-stage optimization model to better deal with large disruptions. In this way, a rolling stock circulation is found that can be rescheduled relatively well when such a large disruption occurs. Also Cadarso and Marín (2014) look at the robustness of solutions, where they focus on creating a circulation that is likely to be robust in execution. They do this by, e.g, penalizing shunting operations that are likely difficult to execute and by penalizing the expected delay that follows from a certain shunting action.

2.4 The Composition and Hypergraph Model

In this section, we present two of the models discussed in the literature review in more detail: the Composition model as proposed by Fioole et al. (2006) and the Hypergraph model as proposed by Borndörfer et al. (2016). Before we present these models, we first formalize the setting of the RSSP.

Definition 1. *Let $\mathcal{T}$ be the (finite) set of trips that are planned in the timetable and $\mathcal{S}$ the set of stations at which these trips stop.*

Note that each trip has a defined departure location, arrival location, departure time, and arrival time. Although train units are required to operate a trip, it is often sufficient to only consider the specific train unit type instead of the exact physical train unit.

Definition 2. *Let R be the set of available train unit types, where each train unit type has a certain set of characteristics, e.g., the seat capacity and the cost per kilometer of operating a unit of this type.*

Definition 3. *A composition is an n-tuple of train unit types with $n \in \mathbb{N}$.*

For example, assume that there are two train unit types, named red and blue (in shorthand r and b). Then, we would have $R = \{r, b\}$ and the two tuples (r, b, b) and (b, b, r) would form two distinct compositions of two blue units and a red one. In one case the red unit is in front, in the other case it is at the back of the composition.

Definition 4. *Let $\mathcal{P}_t$ denote the (finite) set of compositions that can be operated on trip $t \in \mathcal{T}$. The set $\mathcal{P} = \bigcup_{t \in \mathcal{T}} \mathcal{P}_t$ then denotes the set of all compositions that can be operated in the timetable.*

The necessity to specify the admissible compositions per trip is due to operational requirements, such as the maximum platform length along the route, the faced slope along the route, and the availability of traction along the route.

To define connections between trips of $\mathcal{T}$ we denote by $\mathcal{T}_\emptyset := \mathcal{T} \cup \{t_\emptyset\}$ the set of trips enlarged by a placeholder trip $t_\emptyset$.

Definition 5. *Connections are defined as ordered pairs of trips (or the placeholder item) between which rolling stock can be exchanged. Let* $\mathcal{C} \subseteq \mathcal{T}_\emptyset \times \mathcal{T}_\emptyset$ *be the set of all such connections, where each connection links two trips together. In particular, let* $\mathcal{C}^+(t) = \{(t,t') \in \mathcal{C} \mid t' \in \mathcal{T}_\emptyset\}$ *and* $\mathcal{C}^-(t) = \{(t',t) \in \mathcal{C} \mid t' \in \mathcal{T}_\emptyset\}$ *give the connections succeeding and preceding trip* $t \in \mathcal{T}$*, respectively.*

Connections including the placeholder trip $t_\emptyset$, i.e., $(t, t_\emptyset)$ or $(t_\emptyset, t)$ indicate that no succeeding and preceding trip are chosen, respectively, and that all train units are sent to or pulled from the station's depot. Note that connections can define both rolling stock which moves between two trips of a timetable service, but also turnings between trips at terminal stations. Furthermore, note that the definition also allows connections between trips that require a deadhead trip to go from the arrival station of one trip to the departure station of the other.

Different train units in the composition of an incoming trip can generally use different connections, especially when shunting occurs. However, which incoming and outgoing connection can be used at the same time is limited by turning and shunting restrictions. For a given connection $c = (t,t') \in \mathcal{C}$ and incoming and outgoing compositions $p \in \mathcal{P}_t, p' \in \mathcal{P}_{t'}$, we denote by $m \in \mathbb{N}^{|p|}$, with $m_i \leq |p'|$, a feasible mapping of positions of train units on the predecessor trip to positions on the successor trip. Such a mapping can be interpreted as a matching between the train units on the predecessor and successor composition. Note that for a mapping to be feasible in practice, the shunting required for it should be feasible. There should, e.g., be enough time to execute the shunting actions during the connection and the shunting actions should be possible considering the layout of the station.

The mapping m is interpreted as follows: $m_i > 0$ assigns the incoming train unit on position i to position m_i of the outgoing composition p', whereas $m_i = 0$ indicates that the train unit on position i does not use connection c. Accordingly, if no train unit is mapped to a position j on the outgoing trip, i.e., there is no i such that $m_i = j$, then the train unit on this position of the outgoing trip also does not use connection c. As an example, the mapping $m = (2, 1, 0)$ implies that the first incoming unit is sent to the second position of the outgoing trip, the second incoming unit is sent to the first position of the outgoing trip and the third train unit does not make use of this connection, i.e., is uncoupled.

Definition 6. *For a connection* $c = (t,t') \in \mathcal{C}$ *let* $\mathcal{Q}_c$ *denote the set of composition changes, i.e., triples* (p, p', m) *where* $p \in \mathcal{P}_t, p' \in \mathcal{P}_{t'}$ *and* $m \in \mathbb{N}^{|p|}$ *is an operationally feasible assignment of positions.*

Note that a composition change $q \in \mathcal{Q}_c$ tracks exactly those train units going from trip t to t' in connection c. Here, it gives the successor position on trip t' of train units coming in on trip t and the predecessor position on trip t of units departing on trip t'. For the above definition, we define $\mathcal{P}_{t_\emptyset} = \{\emptyset_p\}$, where $\emptyset_p$ denotes an empty composition. In such connections, all train units are thus either coupled or uncoupled.

Lastly, we need to define the available number of train units at each station at the start of the planning period. Moreover, we define the number of train units that are expected to end at each station.

Definition 7. *Let $i^0_{s,r}$ indicate the number of train units that are available of type $r \in \mathcal{R}$ at station $s \in \mathcal{S}$ at the start of the planning horizon. Similarly, let i^∞_s denote the number of train units of type $r \in \mathcal{R}$ that are supposed to end at station $s \in \mathcal{S}$.*

Note that while the number of starting train units is given, the ending inventories are seen as a target that can be deviated from with a certain penalty.

2.4.1 The Composition Model

The Composition model (Fioole et al. 2006) assumes a setting in which the connections, but not the compositions and composition changes between trips, have been fixed in advance. In particular, let $\mathcal{C}_f \subseteq \mathcal{C}$ give this set of fixed connections. We further assume that $\mathcal{C}^+_f(t) = \mathcal{C}^+(t) \cap \mathcal{C}_f \neq \emptyset$ and $\mathcal{C}^-_f(t) = \mathcal{C}^-(t) \cap \mathcal{C}_f \neq \emptyset$ for all $t \in \mathcal{T}$. In general, there is exactly one predecessor and successor connection, i.e., $|\mathcal{C}^+_f(t)| = 1 = |\mathcal{C}^-_f(t)|$. Train units coming from a predecessor trip then either follow this fixed connection or are moved to the shunting yard and can be used again after some reallocation time without taking into account the composition that these train units were in.

We need to define some additional notation to model the composition changes in the Composition model:

Definition 8. *Let $\upsilon_{q,r}$ and $\gamma_{q,r}$ denote the number of train units of type $r \in \mathcal{R}$ that are uncoupled to the shunting yard and coupled from the shunting yard in composition change $q \in \mathcal{Q}_c$, respectively.*

Definition 9. *Let $\tau^-(c)$ indicate the time that train units that are uncoupled in fixed connection $c \in \mathcal{C}_f$ become available again for coupling. Moreover, let $\tau^+(c)$ be the time at which coupling starts for any train units coupled in fixed connection $c \in \mathcal{C}_f$.*

Definition 10. *Let $s(c) \in \mathcal{S}$ denote the station at which connection $c \in \mathcal{C}$ takes place.*

In the composition model, it is assumed that all train units that do not use the fixed connections have the same reallocation time, at least per station. Hence, it is possible to aggregate the units at the stations into an inventory. In other words, it suffices to track the number of available train units at every station, instead of tracking connections $c \notin \mathcal{C}_f$.

By defining the decision variables

$$X_{t,p} = \begin{cases} 1 & \text{if composition } p \in \mathcal{P}_t \text{ is chosen for trip } t \in \mathcal{T}, \\ 0 & \text{otherwise,} \end{cases}$$

$$Z_{c,q} = \begin{cases} 1 & \text{if composition change } q \in \mathcal{Q}_c \text{ is chosen for connection } c \in \mathcal{C}_f, \\ 0 & \text{otherwise,} \end{cases}$$

$$E_{s,r} = \text{the number of train units of type } r \in \mathcal{R} \text{ ending at station } s \in \mathcal{S},$$

and letting $C_s \subseteq C_f$ be the set of fixed connections at station $s \in \mathcal{S}$, the Composition model is given by

$$\begin{aligned}
\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c_{t,p} X_{t,p} + \sum_{c\in\mathcal{C}_f} \sum_{q\in\mathcal{Q}_c} c_{c,q} Z_{c,q} \\
+ \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c_{s,r} |E_{s,r} - i^{\infty}_{s,r}| && (2.1)
\end{aligned}$$

$$\text{s.t.} \sum_{p\in\mathcal{P}_t} X_{t,p} = 1 \qquad \forall t \in \mathcal{T}, \tag{2.2}$$

$$X_{t,p} = \sum_{c\in\mathcal{C}_f^-(t)} \sum_{q=(p',p,m)\in\mathcal{Q}_c} Z_{c,q} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \tag{2.3}$$

$$X_{t,p} = \sum_{c\in\mathcal{C}_f^+(t)} \sum_{q=(p,p',m)\in\mathcal{Q}_c} Z_{c,q} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \tag{2.4}$$

$$\begin{aligned}
i^0_{s(c),r} + \sum_{c'\in\mathcal{C}_s:\tau^-(c')\leq\tau^+(c)} \sum_{q\in\mathcal{Q}_c} \upsilon_{q,r} Z_{c',q} \\
- \sum_{c'\in\mathcal{C}_s:\tau^+(c')\leq\tau^+(c)} \sum_{q\in\mathcal{Q}_c} \gamma_{q,r} Z_{c',q} \geq 0 && \forall c \in \mathcal{C}_f, r \in \mathcal{R}, && (2.5)
\end{aligned}$$

$$E_{s,r} = i^0_{s,r} + \sum_{c\in\mathcal{C}_s} \sum_{q\in\mathcal{Q}_c} (\upsilon_{q,r} - \gamma_{q,r}) Z_{c,q} \qquad \forall s \in \mathcal{S}, r \in \mathcal{R}, \tag{2.6}$$

$$X_{t,p} \in \{0,1\} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \tag{2.7}$$

$$Z_{c,q} \in \{0,1\} \qquad \forall c \in \mathcal{C}_f, q \in \mathcal{Q}_c, \tag{2.8}$$

$$E_{s,r} \in \mathbb{Z}_+ \qquad \forall s \in \mathcal{S}, r \in \mathcal{R}. \tag{2.9}$$

The objective function (2.1) minimizes the cost that results from the chosen compositions, from the chosen composition changes and from any deviations from the number of train units supposed to end at each station. Note that the absolute value in the last term of the objective function can easily be linearized. Constraints (2.2) ensure that a composition is chosen for each trip. Constraints (2.3) and (2.4) ensure that the compositions chosen for the predecessor and successor trips of a connection match with the chosen composition change for that connection. Constraints (2.5) ensure that the number of train units that are present at a station is never negative and hence that no more train units are used than are available. Constraints (2.6) determine the number of train units of each type that end at a station. The last sets of constraints give the variable domains.

2.4.2 The Hypergraph Model

The Hypergraph model, as proposed by Borndörfer et al. (2016), is based on a hypergraph representation of the problem. A solution to the RSSP corresponds to finding a set of cycles, or paths in an acyclical setting, in the hypergraph that together cover all trips. Let the hypergraph be given by $G := (V, A, H)$, where V is the set of nodes in the graph, A is the set of standard arcs and H is the set of hyperarcs. Note that this hypergraph is a graph-based hypergraph, in which each hyperarc $h \subset A$ is a node disjoint subset of the standard arcs A.

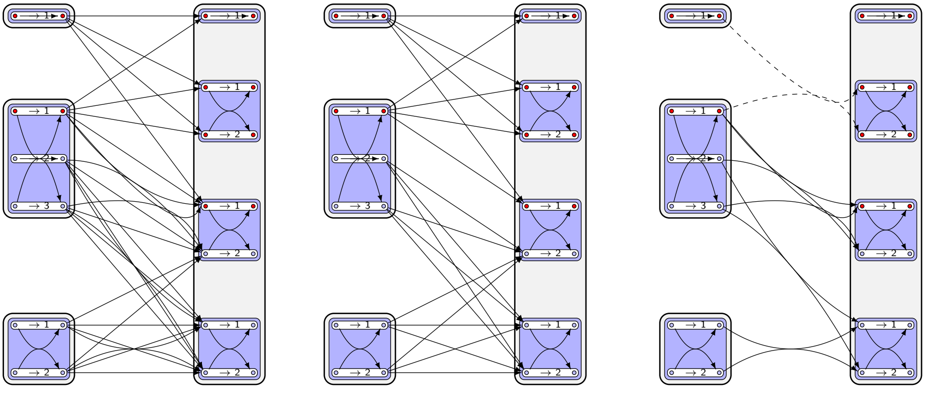
Each node $v \in V$ defines an arrival or departure event of a train unit of a certain type operating a trip in a fixed composition at a fixed position. We thus get two nodes for each train unit in each composition $p \in \mathcal{P}_t$ and trip $t \in \mathcal{T}$, one corresponding to the departure of the train unit and one corresponding to the arrival of the train unit. Note that there might be multiple nodes for the same trip and train unit type since p is a tuple, i.e., there might be more than one physical train unit of the same type in composition p.

The set of standard arcs $A \subseteq V \times V$ now connects arrival and departure nodes. So there is a standard arc $a = (v, w) \in A$ between two nodes $v, w \in V$ if both nodes model the same train unit type and the underlying train unit movement is operationally feasible. In particular, there is a standard arc modeling a trip movement between a departure node v and arrival node w if a train unit that starts at the position of node v ends the trip at the position corresponding to w. Moreover, a standard arc that models a connection is added between an arrival node v for trip t and a departure node w for trip t' if there is a connection $(t, t') \in \mathcal{C}$ and a composition change $q \in \mathcal{Q}_c$ that allows a train unit to move from v to w. In particular, this implies that if v corresponds to a composition p and w to a composition p', then there should be an element $(p, p', m) \in \mathcal{Q}_c$ in which the mapping m connects v and w.

Hyperarcs now define the movements of train units that are coupled into a composition, for example in case of operating a trip or of moving between the trips in a connection. These hyperarcs thus connect equally sized sets of arrival and departure nodes. First, trip-hyperarcs are defined for the train units operating a trip. A trip-hyperarc can be uniquely identified by the trip $t \in \mathcal{T}$ and the composition $p \in \mathcal{P}_t$. Thus, a trip-hyperarc for trip $t \in \mathcal{T}$ and composition $p \in \mathcal{P}_t$ contains all standard arcs that connect a departure node of t in composition p with an arrival node of t in composition p. Secondly, hyperarcs are defined to model the movement of train units in a connection. In the case where a connection $c = (t, t') \in C$ is modeled, the hyperarc is uniquely identified by the in- and outgoing trips $t, t' \in \mathcal{T}$, the in- and outgoing compositions $p \in \mathcal{P}_t, p' \in \mathcal{P}_{t'}$ and the composition in which the connection is made. Thus, a connection-hyperarc for a composition change $q = (p, p', m) \in Q_c$ contains exactly those standard arcs that connect arrival nodes of trip t in composition p with departure nodes of trip t' in composition p' according to the mapping m.

In contrast to the Composition model, the original version of the Hypergraph model of Borndörfer et al. (2016) is set up for a cyclic planning horizon. For a better comparison of the two models, we consider here an adaptation of the acyclic version of the Hypergraph model presented in Grimm et al. (2017). It is a straightforward extension of the cyclic model using a set of origin and destination nodes $O \subset V$, respectively $D \subset V$. There is an origin node $o \in O$ for each train unit at a certain station that is available at the beginning of the planning horizon. Moreover, there is a destination node $d_{s,r} \in D$ for each train unit type $r \in \mathcal{R}$ and each station $s \in \mathcal{S}$. There is a standard arc and a corresponding hyperarc connecting each origin node with a departure node of the same train unit type that could be reached from that station before the associated departure time. Additionally, there are standard arcs and hyperarcs from each arrival node to each destination node of the same train unit type that could be reached from the respective station within the time horizon.

An example of how the hypergraph looks is given in Figure 2.2. On the left-hand side of each of the three figures there are three different trips operated in single, triple, and double composition. On the right-hand side there is a trip shown with the option to operate it in a single unit composition or three different double unit compositions using different train unit types, i.e., two red, two blue, or a blue and a red one. Trip hyperarcs are shown for each of these possible compositions, where there are, e.g., four hyperarcs to model the four possible compositions of the trip on the right. The three figures categorize the connection-hyperarcs, where we separate in the last two figures hyperarcs modeling a single train unit and those modeling two train units. As an example, a connection-hyperarc modeling two train units can be seen in Figure 2.2c that connects the double train unit composition of the bottom-left trip with the composition consisting of two blue train units of the right trip. The dashed hyperarc in Figure 2.2c shows a hyperarc coupling train units from two different incoming trains together to operate a trip in a double configuration. Hyperarcs like this could be added to the Hypergraph model but are explicitly not added in the implementation at DB.



(a) All hyperarcs (b) Single vehicle hyperarcs (c) Double vehicle hyperarcs

Figure 2.2: Example of the hyperarcs contained in H.

Let $H(t) \subseteq H$ now denote those hyperarcs modeling the operation of trip $t \in \mathcal{T}$. Moreover, let the sets $H^-(v) = \{h \in H \mid (v', v) \in h, v' \in V\}$ and $H^+(v) = \{h \in H \mid (v, v') \in h, v' \in V\}$ denote the incoming and outgoing hyperarcs of a node $v \in V$. Taking into account the above hypergraph and defining integer decision variables

$$Y_h = \begin{cases} 1 & \text{if hyperarc } h \in H \text{ is selected,} \\ 0 & \text{otherwise,} \end{cases}$$

$$E_{s,r} = \text{the number of train units of type } r \in \mathcal{R} \text{ ending at station } s \in \mathcal{S},$$

the Hypergraph model is then given by

$$
\begin{aligned}
\min \sum_{h \in H} c_h Y_h + \sum_{s \in \mathcal{S}} \sum_{r \in \mathcal{R}} c_{s,r} |E_{s,r} - i^{\infty}_{s,r}| & & (2.10)\\
\text{s.t.} \sum_{h \in H(t)} Y_h = 1 & \qquad \forall t \in \mathcal{T}, & (2.11)\\
\sum_{h \in H^+(o)} Y_h = 1 & \qquad \forall o \in O, & (2.12)\\
\sum_{h \in H^+(v)} Y_h = \sum_{h \in H^-(v)} Y_h & \qquad \forall v \in V \setminus (O \cup D), & (2.13)\\
\sum_{h \in H^-(d_{s,r})} Y_h = E_{s,r} & \qquad \forall s \in \mathcal{S}, r \in \mathcal{R}, & (2.14)\\
Y_h \in \{0,1\} & \qquad \forall h \in H, & (2.15)\\
E_{s,r} \in \mathbb{Z}_+ & \qquad \forall s \in \mathcal{S}, r \in \mathcal{R}. & (2.16)
\end{aligned}
$$

The objective function (2.10) minimizes the costs of train unit movements associated with the chosen hyperarcs and of any deviations from the number of train units supposed to end at each station. Constraints (2.11) take care that for each trip a hyperarc is chosen that models the operation of that trip by a fixed composition. Constraints (2.12) determine the number of train units available at the start of the planning horizon. The set of flow conservation constraints (2.13) ensures that after a train unit's task, i.e., either a timetable trip or a connection between two trips, a successor task is chosen. Constraints (2.14) determine the number of train units of each train unit type that end at each station. Finally, constraints (2.15) and (2.16) define the variable domains.

Note that, compared to the Composition model, the Hypergraph model tracks all connections $c \in \mathcal{C}$. In particular, this implies that the Hypergraph model can take into account detailed restrictions on the train unit movements, such as deadheading between trips as well as any shunting that occurs between two trips. Moreover, note that the above model does not correspond to the full Hypergraph model as proposed in Borndörfer et al. (2016). Specifically, we have not included the modeling of maintenance here, which is done in Borndörfer et al. (2016) through adding an additional layer of flow that keeps track of the distance driven by each train unit. In addition, more generalized hyperarcs, i.e., hyperarcs composed of other hyperarcs, are defined by Borndörfer et al. (2016) to consider regularity requirements at DB.

2.5 Analytical Comparison

As mentioned in Section 2.3, the Composition model and the Hypergraph model have been designed for a different operational setting. Hence, we have to restrict this analytical comparison to an operational setting that can be included in both models. In this case, we take the operational setting considered for the Composition model in Fioole et al. (2006). This implies that we make the following assumptions about the considered operational setting:

- We consider fixed, i.e., pre-assigned, connections between trips.
- When a train unit is uncoupled from a composition during a connection, it can be coupled to any trip that leaves from the same station after some fixed reallocation time. Note that the composition that the train unit was in while being uncoupled is not taken into account to determine if a train unit can be coupled to another trip.
- We do not consider any maintenance restrictions.
- We do not allow for deadheading, unless such deadheading trips have been added in the timetable beforehand.
- We do not consider the orientation of individual train units.

In this section, we will show that the above operational setting can also be modeled by means of the Hypergraph model. We will thus show that under these assumptions on the operational setting, both models contain the same set of feasible solutions regarding appropriate projections between the two solution spaces. Moreover, we show that the same linear programming relaxation value is obtained by both models for this operational setting.

2.5.1 Linking the Composition and Hypergraph Model

To compare the formulations of the Composition model and the Hypergraph model, we have to make sure that both formulations model the same problem. To achieve this, we first need to make an additional assumption on the composition changes considered in the Composition model. We can then construct an appropriate hypergraph H based on the set of allowed compositions for trips and the set of allowed composition changes for connections as considered in the Composition model.

We first need to make an assumption on the allowed composition changes for fixed connections that do not pull all train units from the shunting yard or send all train units to the shunting yard. Let $\mathcal{C}_f^{'} = \{(t,t') \in \mathcal{C}_f \mid t \neq t_\emptyset, t' \neq t_\emptyset\} \subseteq \mathcal{C}_f$ denote this set of connections. We then make the following assumption:

Assumption 1. *Consider a fixed connection $c = (t,t') \in \mathcal{C}_f^{'}$. We then assume that it holds for each $p \in \mathcal{P}_t$ that there is a position $1 \leq i \leq |p|$ that is not uncoupled in any composition change $q \in \mathcal{Q}_c$, i.e., $m_i \neq 0$ for all $q = (p, p^*, m) \in \mathcal{Q}_c$ with $p^* \in \mathcal{P}_{t'}$. Alternatively, it is also enough if for each $p' \in \mathcal{P}_{t'}$ there is a position $1 \leq i \leq |p'|$ that is never coupled, i.e., $m_j = i$ for some value $1 \leq j \leq |p^*|$ for all $q = (p^*, p', m) \in \mathcal{Q}_c$ with $p^* \in \mathcal{P}_t$.*

Note that the above assumption is often satisfied in practice. This assumption is, e.g., satisfied in case uncoupling or coupling can only take place on one side of the composition and if not all train units can be uncoupled or coupled at once in those cases. Instead, this assumption is also satisfied if the shunting in all composition changes of a fixed connection is limited to only coupling or to only uncoupling.

To construct the hypergraph, we stick to the construction of Section 2.4.2 with one exception concerning the connections. First of all, note that for each trip $t \in \mathcal{T}$ and each composition $p \in \mathcal{P}_t$ we consider a hyperarc $h \in H$ that represents a trip operating with this composition. Translating the composition changes that are allowed by the Composition model to hyperarcs is more challenging. In particular, note that each composition change specifies the predecessor and successor compositions for each trip in a connection. However, each train unit that is uncoupled during such a connection in the Composition model can be coupled to any other trip that leaves at that station after some fixed reallocation time. This, while in the Hypergraph model the trip at which an uncoupled train unit is coupled again is considered explicitly.

To ensure that the same problem is modeled, we add hyperarcs modelling connections in two ways. First, we add a hyperarc for each composition change $q \in \mathcal{Q}_c$ for each connection $c = (t, t') \in \mathcal{C}_f^{'}$. This hyperarc thus models the train units that move from trip t to t'. Further, we mark an arrival or departure node for each position of a composition of a trip as *couplable* if either there is no composition change where the node is involved or there is a composition change in which this particular position gets uncoupled or coupled in the sense of the Composition model. Then single arc hyperarcs are added for each pair of arrival and departure nodes that both are *couplable*, share the same station, and have fitting arrival and departure times to model this particular train unit movement. Note that we can restrict ourselves to linking couplable positions at the same station here, as we assumed that deadheading has to be added explicitly to the timetable as a trip. Moreover, note that couplable nodes are connected to nodes $o \in O$ and nodes $d \in D$ in a similar way as discussed in Section 2.4.2.

We will now show that both the Composition model and Hypergraph model are modeling the same problem, i.e., allow the same set of integer feasible solutions. We do this by first showing that each solution of the Composition model can be translated to a solution of the Hypergraph model in which the same compositions are chosen for each trip. Second, we then show the opposite direction.

Theorem 1. *Consider a solution* $(X, Z, E) \in \mathbb{B}^n \times \mathbb{N}^m$ *of the Composition model that satisfies* (2.2) – (2.9)*. Then there is a solution* $(Y, E) \in \mathbb{B}^{|H|} \times \mathbb{N}^m$ *of the Hypergraph model that satisfies* (2.11) – (2.16)*.*

Proof. We start by looking at the compositions for the trips. Each entry $X_{t,p}$ of X with $X_{t,p} = 1$ represents a trip $t \in \mathcal{T}$ operated in composition $p \in \mathcal{P}_t$. By construction, there are departure and arrival nodes for each train unit in p and a hyperarc h that connects them modeling the operation of trip t in composition p. We set $Y_h = 1$. As X satisfies constraint (2.2), there is no other variable with $X_{t,p} = 1$ for any $p \in \mathcal{P}_t$ and thus (2.11) holds as well for all $t \in \mathcal{T}$.

By constraints (2.3) – (2.4), and the observation that there is a single $p \in \mathcal{P}_t$ such that $X_{t,p} = 1$, there is exactly one $q \in \mathcal{Q}_c$ such that $Z_{c,q} = 1$ for any connection $c = (t, t') \in \mathcal{C}_f$. By construction, there is a hyperarc $h \in H$ that models composition change q, i.e., the movement of train units from trip t to t' in composition change q, in case at least one train unit moves from trip t to trip t' in this composition change. We set $Y_h = 1$. For all nodes $v \in \{u \mid (u,w) \in h \vee (w,u) \in h\}$ constraint (2.13) now holds: on the one hand there is exactly one trip-hyperarc h' that contains v with $Y_{h'} = 1$, while on the other hand h is the unique connection-hyperarc in $H^+(v)$ or $H^-(v)$, with $Y_h = 1$, that contains v.

It remains to show that constraints (2.13) hold for nodes that correspond to uncoupled and coupled train units. For this, let $I(t,p)$ denote the set of positions in composition $p \in \mathcal{P}_t$ of trip $t \in \mathcal{T}$. For position $i \in I(t,p)$ let $t^+(i) \in \mathcal{T}$ and $t^-(i) \in \mathcal{T}$ now indicate the successor and the predecessor trip for the train unit, respectively. Note that for those train units for which no shunting is involved, their predecessor and successor trip are directly defined by c. We now explain how they are defined for other train units.

To determine predecessor and successor trips for coupled and uncoupled train units, we note that constraints (2.5) ensure that the number of train units at a station is always non-negative. For fixed composition changes, these constraints ensure that there is a (non-negative) flow of train units of each rolling stock type in an appropriate network for each station. In particular, each uncoupled train unit defines an inflow and corresponds to a source node in the network, while each coupled train unit defines an outflow and thus corresponds to a sink node. Moreover, also the train units that start at the station at the beginning of the day form an inflow and thus correspond to a source node. Nodes in this network are connected when they are consecutive in time. Remark that this network for each station is acyclic and finite.

It is well known that a flow in such a network allows for a path decomposition, which then specifies at which trip an uncoupled train unit is coupled again. Let p_t now indicate the composition in which trip t is operated, i.e., $X_{t,p_t} = 1$. For any feasible path decomposition, we can then specify $t^-(i)$ and a position $i^- \in I(t^-(i), p_{t^-(i)})$ for each $t \in \mathcal{T}$ and each $i \in I(t,p_t)$ at which a train unit gets coupled. Note that these positions correspond to nodes in the hypergraph which are *couplable*. In case there is no such trip, the respective train unit is used from the station's starting inventory and i^- then corresponds to an origin node of the hypergraph. By construction of the hypergraph there exists a standard arc $a = (v^-, v) \in A$ where nodes v^-, v correspond to the positions i^-, i of the respective trips and compositions. Moreover, there is, again by construction, a single arc hyperarc h' that contains a. Set $Y_{h'} = 1$. In this way, constraints (2.13) are satisfied for v^- and v as we pick hyperarc h' as a unique hyperarc that contains both nodes.

Lastly, note that also constraints (2.14) hold by means of the path decomposition above. In particular, constraints (2.12) ensure that the number of train units starting at each station matches that in constraints (2.6). Moreover, the number of train units that reach a destination node $d_{s,r}$ are exactly those that are not assigned to be coupled at an outgoing composition by the path decomposition. As the inflow and outflow of any source and sink node corresponding to a trip matches the number of coupled and uncoupled train units in (2.6), this implies that the variables $E_{s,r}$ take the same values in both models.

$\square$

Theorem 2. *Consider a solution* $(Y, E) \in \mathbb{B}^{|H|} \times \mathbb{N}^m$ *of the Hypergraph model that satisfies* (2.11) – (2.15). *Then there exists a solution* $(X, Z, E) \in \mathbb{B}^n \times \mathbb{N}^m$ *of the Composition model that satisfies* (2.2) – (2.8).

Proof. As the variables $E \in \mathbb{N}^m$ are considered in both models, we exclude these for now and focus solely on the remaining variables in both models. Let $\mathbb{B}^n, \mathbb{B}^{|H|}$ then be the hyperspaces in which the feasible points of the Composition model, respectively the Hypergraph model are contained and let $x^\intercal = (X_1, \ldots, X_k, Z_1, \ldots, Z_l) \in \mathbb{B}^n$, $y^\intercal = (Y_1, \ldots, Y_{|H|}) \in \mathbb{Q}^{|H|}$ be solution vectors of the respective models.

We then define a linear transformation $M : \mathbb{R}^{|H|} \mapsto \mathbb{R}^n$ of the variables of the Hypergraph model to the variables of the Composition model. The transformation is given by matrix $M \in \mathbb{Q}^{n \times |H|}$ with the following coefficients:

- Let $m_{i\cdot}$ be a row of M associated with a $X_{t,p}$ variable. Row $m_{i\cdot}$ is then a standard unit vector where coefficient $m_{ij} := 1$, with j being the column index for the hyperarc variable Y_h that models the operation of trip t in composition p.

- Furthermore, let $m_{i\cdot}$ be a row of M associated with a $Z_{c,q}$ variable, where $c = (t, t') \in \mathcal{C}_f^{'}$ and $q \in \mathcal{Q}_c$. Row $m_{i\cdot}$ is then a standard unit vector where coefficient $m_{ij} := 1$ with j being the column index for the hyperarc variable Y_h such that the hypergraph h models the movement of all train units from trip t to t' as determined in q. Note that by the construction of the hypergraph this hyperarc h is uniquely defined.

- Finally, let $m_{i\cdot}$ be a row of M associated with a $Z_{c,q}$ variable that assigns for a connection $c = (t, t') \in \mathcal{C}_f$ all train units of an incoming composition $p^- \in \mathcal{P}_t$ to the station's depot. We then set $m_{ij} := 1$ with j being the column index for the hyperarc variable that corresponds to operating trip t in composition p^-. In case of rows of M associated with $Z_{c,q}$ variables that require all train units from the depot, coefficients are set similarly.

It remains to show that My is a feasible solution of the Composition model, which is done by proving that equations (2.2) – (2.8) hold for My. The set of hyperarcs $H(t)$ contains exactly one hyperarc for each $p \in \mathcal{P}_t$. M maps $Y_{h(t,p)}$ to $X_{t,p}$ and it thus follows from (2.11) that (2.2) holds as well for all $t \in \mathcal{T}$. To prove that (2.3) and (2.4) hold, assume first that $X_{t,p} = 0$ and consequently that $Y_{h(t,p)} = 0$. By (2.13) it follows for all hyperarcs $h \in H^+(v) \cup H^-(v)$, with $v \in V(h(t,p))$, that $Y_h = 0$. It therefore holds that $Z_{c,q} = 0$ for all composition changes that include composition p for trip t.

Now focus on Equation (2.3) and instead assume that $X_{t,p} > 0$. First consider the case where t is preceded by a connection $c = (t', t)$ with $t' \neq t_\emptyset$. Let p' be the composition of trip t' of t with $Y_{h(t',p')} = 1$. Summing over all composition changes for connection c in the Composition model translates to summing over all hyperarcs between nodes related to p and p', as these are the only non-zero variables that could be projected via M to variables of the composition changes incoming to (t, p). Let $H_{p,p'}$ denote this set. By Assumption 1, only one such hyperarc h can have $Y_h = 1$, as all share the node that is not couplable. Moreover, by flow conservation constraints (2.13), at least one of these hyperarcs needs to be chosen. Hence, My defines $Z_{c,q} = 1$ for the composition change corresponding to this hyperarc, implying that (2.3) holds. Second, consider the case where we have a preceding connection $c = (t_\emptyset, t)$. The matrix M then ensures that $Z_{c,q} = X_{t,p}$ for $q = (\emptyset_p, p, m) \in \mathcal{Q}_c$, where $\emptyset_p$ denotes again an empty composition. Note that q is also the unique composition change preceding trip t in the Composition model in this case, implying again that (2.3) holds. Similar arguments can be made for (2.4).

To show that also constraints (2.5) are satisfied for this solution My, we note that constraints (2.11) – (2.13) ensure a path through the hypergraph for each train unit. Moreover, constraints (2.12) ensure that no more train units are used than available at the start of the day. Together, these imply that the station inventories are always non-negative. Finally, My satisfies the Equations (2.7) and (2.8) trivially, as each variable $Y_h \in \mathbb{B}$.

It now remains to show that an identity mapping, i.e., a mapping with the identity matrix, works for the $E_{s,r}$ variables. For this, note that the mapping M conserves the number of train units uncoupled and coupled at each station in the sense of the Composition model. As also the number of train units that start at each station is equal by means of constraint (2.12), this implies that also the variables E are equal in both models. □

2.5.2 Strength of the Linear Programming relaxation

We now prove that both formulations produce the same objective function value of the Linear Programming (LP) relaxation. To do so, we first prove that the solution spaces are equal under the mapping M defined above.

Theorem 3. *Let $\mathcal{P}_{COMP}$ and $\mathcal{P}_{HYP}$ be the sets of solutions of the LP-relaxations of respectively the Composition model and Hypergraph model of a fixed problem instance. Then, $M(\mathcal{P}_{HYP}) = \mathcal{P}_{COMP}$.*

Proof. Note that the proof of Theorem 1 remains valid if we consider a feasible fractional solution $(X, Z, E) \in [0,1]^n \times \mathbb{R}^m$. In particular, it is well known that also a fractional network flow can be decomposed into paths between source and sink nodes, i.e., allows for a path decomposition. As for the integer case, this path decomposition gives the values for the hyperarcs that do not follow the fixed connections. Moreover, note that also the proof of Theorem 2 remains valid if we consider a feasible fractional solution $(Y, E) \in [0,1]^{|H|} \times \mathbb{R}^m$. □

We still need to show that if a solution is mapped by means of the mapping M, that then also the same objective value is obtained in both models. To do so, we translate the cost parameters $c_{t,p}$ and $c_{c,q}$ of the Composition model in the following way to the cost parameters c_h of the Hypergraph model. If a hyperarc $h \in H$ corresponds to a trip t being operated in composition $p \in \mathcal{P}_t$, we set $c_h := c_{t,p}$. The exception is when trip t is followed by a connection $c = (t, t_\emptyset)$ or preceded by a connection $c = (t_\emptyset, t)$. In that case, we set $c_h := c_{t,p} + c_{c,q}$ with q the unique composition change that sends all train in composition p to the shunting yard and picks up all train units from the shunting yard, respectively. Now consider a hyperarc $h \in H$ that models the movement of train units from trip $t \neq t_\emptyset$ to $t' \neq t_\emptyset$ in connection $c = (t, t') \in \mathcal{C}_f$. Each such hyperarc corresponds to a composition change $q \in \mathcal{Q}_c$. We set $c_h := c_{c,q}$. For all other hyperarcs $h \in H$, we set $c_h := 0$.

We are now able to prove the main result of our analytical comparison:

Theorem 4. *The objective function value of the Linear Programming relaxation achieved by the Composition model is equal to that of the Hypergraph model.*

Proof. First of all, note that Theorem 3 implies that each solution of the Composition model can be mapped to a solution of the Hypergraph model and vice versa. It remains to show that that the cost of a solution remains equal after being mapped.

First, consider a solution (X, Z, E) of the Composition model. The costs of deviations from the ending inventories are directly transferred to the Hypergraph model. Moreover, by Theorem 1, each composition for a trip is mapped to a corresponding hyperarc, which incorporates the composition cost $c_{t,p}$. In addition, again by Theorem 1, each composition change for a connection can be mapped to a corresponding hyperarc between the trips in the connection in case there is at least one train unit that moves between these trips. The cost of this hyperarc is again equal to that of the composition change by means of the construction of the costs for the hyperarcs above. In case no train unit moves from t to t', then either $t = t_\emptyset$ or $t' = t_\emptyset$ by Assumption 1. In these cases, the cost $c_{c,q}$ of the composition change is incorporated into the cost c_h of the arc $h(t, p)$ that corresponds to the incoming or outgoing composition in connection c.

Second, consider a solution (Y, E) of the Hypergraph model. By the reasoning above, each trip-hyperarc and each hyperarc for a composition change for a connection $c \in \mathcal{C}_f$ corresponds to a composition and composition change variable of equal cost in the Composition model, respectively. The exception includes the cost for hyperarcs modeling a trip that is succeeded by a connection that sends or pulls all train units from the shunting yard. By the mapping M, we then obtain that $Z_{c,q} = X_{t,p}$ and the cost of the hyperarc contains the sum of the costs of the variables $X_{t,p}$ and $Z_{c,q}$. All other hyperarcs have zero cost. □

2.6 Numerical Comparison

For the numerical comparison, we compare the Composition model and Hypergraph model on a set of instances from NS. In particular, we are interested in evaluating the relative performance of both models in terms of the time needed to find (near-)optimal solutions. In the remainder of this section, we first introduce the considered instances and then discuss the numerical results for these instances.

2.6.1 Instances

The instances that we consider are based on the 2018 timetable operated by NS, which is the largest passenger railway operator in the Netherlands. An overview of the network that is operated by NS can be found in Figure 2.3. On this network, NS operates both intercity services and regional (sprinter) services, where the latter generally stop at each encountered station along a railway line. Both intercity and sprinter trains operate in a relatively high frequency, where there are usually 2 to 4 trains per hour for each timetable service.

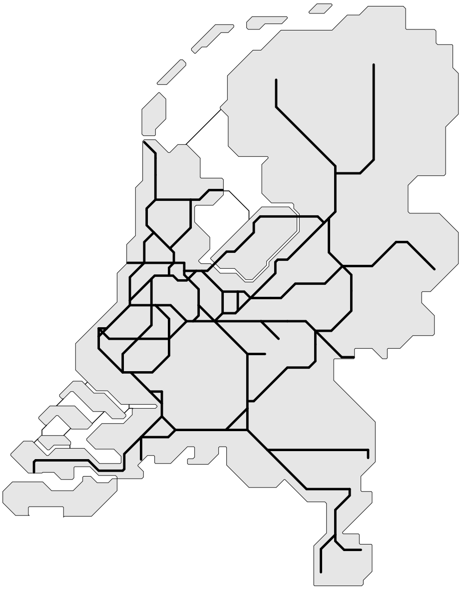


Figure 2.3: Overview of the Dutch railway network

We create rolling stock scheduling instances by considering different rolling stock fleets. At NS, train unit types are categorized into so-called train unit families. Only train unit types that belong to the same family can be combined in a composition. We consider train units of six different families: SLT, SGM, FLIRT, ICM, VIRM, and DDZ. The first three families are used to operate sprinter trains, while the latter three are (mainly) used for intercity services. All train unit families are composed of two different rolling stock types, i.e., contain train units of two types. Instances are then created by only considering those trips in the timetable that should be operated by train units of the considered family. Summary statistics about the chosen instances are shown in Table 2.2, where the number of trips in the instance, the number of considered rolling stock types, and the number of possible compositions are given.

The parameters used for the objective function can be found in Table 2.3. The first two parameters relate to the operated compositions. These define the costs per traveled kilometer of a carriage and per seat that is short compared to the expected number of passengers, respectively. The third parameter relates to the cost of performing a shunting action, which is incurred if either coupling or uncoupling occurs during a connection. The fourth parameter defines the cost of a deviation

Table 2.2: Summary statistics of the considered instances.

Instance	$\|\mathcal{T}\|$	$\|\mathcal{R}\|$	$\|P\|$
NS-SLT	1681	2	14
NS-SGM	758	2	14
NS-FLIRT	584	2	14
NS-ICM	1222	2	30
NS-VIRM	1300	2	10
NS-DDZ	419	2	6

from the number of train units, of a given rolling stock type, that are expected to end at a station.

Table 2.3: Parameters used in the objective function

Parameter	Value
Mileage	0.1
Seat shortage	0.2
Shunting	10
Ending deviation	10000

2.6.2 Results

For the numerical comparison, we implemented both models as pure MILP models and solved them with the CPLEX 20.1.0 general-purpose MILP-solver. We have chosen this approach, over using the exact application of the two models at the two companies, as both are highly specialized for the individual company's needs. The disadvantage of this is that especially the Hypergraph model was designed to be solved via a column generation procedure. Nevertheless, we think that the comparison given here is the most fair to compare the pure MILP models. All computations were performed on an Intel® Xeon(R) Gold 6130 @ 2.10GHz CPU with 16 cores and 32 threads.

Table 2.4 shows the computational results of our experiments. The first column of the table gives the name of the instance. Then, there are two blocks of five columns each, one for the results of the Composition model and one for the Hypergraph model. Within each block, the columns "LP" and "MILP" show the objective function values of the found solutions for the LP-relaxation of the model and the MILP formulation of that approach. The columns "LP-CPU" and "CPU" give the running time in seconds that was taken to solve the linear programming relaxation and the MILP formulation, respectively. Lastly, the column "Nod." denotes the number of nodes processed in the Branch and Bound tree by that solution method.

The results of the objective function values for both the LP-relaxation and the integer solution shown in Table 2.4 confirm our theoretical results, as they are identical

Table 2.4: Computational results for the Composition model and the Hypergraph model applications on the test set. All running times, i.e., the columns LP-CPU and CPU of the table, are given in seconds.

	Composition					Hypergraph				
Instance	LP	LP-CPU	MILP	CPU	Nod.	LP	LP-CPU	MILP	CPU	Nod.
NS-SLT	38531	0.25	38729	1.29	1	38531	0.89	38729	33.95	1
NS-SGM	19630	0.12	19647	0.76	1	19630	0.49	19647	5.09	1
NS-FLIRT	18805	0.04	18992	0.21	1	18805	0.08	18992	0.58	1
NS-ICM	48501	3.39	48615	12.80	1	48501	93.40	48615	1620	43
NS-VIRM	98297	0.40	98733	1.88	1	98297	1.32	98733	15.07	1
NS-DDZ	20110	0.06	20110	0.26	1	20110	0.24	20110	1.86	1

for both models. We note that the integrality gap of the LP-relaxation is relatively small for all instances, with a maximum integrality gap of about 1% and an average integrality gap of about 0.38% for these instances. When comparing the running time needed by both models to solve the MILP problem, it can be seen that the running time of the Composition model is shorter than that of the Hypergraph model for all instances. The largest difference can be seen for the NS-ICM instance, which is also the instance with the largest number of compositions that can be formed. Instead, the difference in computation time is the smallest for the NS-SGM, NS-FLIRT, and NS-DDZ instances. From the summary statistics in Table 2.2, it can be seen that these are the instances containing the fewest trips and these can thus be categorized as the smallest instances to solve.

A particularly interesting result from Table 2.4 is that most instances can be solved within the root node by CPLEX, implying that the addition of cuts by CPLEX is enough to close the integrality gap for these instances. The only exception is the "NS-ICM" instance, for which we see that a larger number of nodes is needed to solve the Hypergraph model. This likely contributed to the large gap in running time for this instance between the Composition model and Hypergraph model. Another contributing factor to the longer running tines of the Hypergraph model can be found in the time needed to solve the linear programming relaxation. For all instances, the time needed to solve the linear programming relaxation of the Hypergraph model is larger than for the Composition model. This is likely related to the larger size of the Hypergraph model, due to a larger number of variables modeling the connections between trips.

The longer computation times for the Hypergraph model can also be expected from our theoretical results, as we have shown that both models obtain the same objective function value for the linear programming relaxation. Then, the ability of the Hypergraph model to track individual train units, comes at the expense of a longer running time. In particular, multiple solutions of the Hypergraph model may correspond to the same solution of the Composition model. The results show clearly that choosing the right model for the right application in railway optimization is highly dependent on the level of detail which is considered, and that choosing the most general model does not always lead to satisfactory results.

2.7 Conclusion

In this chapter, we have made a comparison between models that have been proposed for the scheduling of rolling stock at passenger railway operators. First of all, we have shown through a literature review how models differ due to differences in the operational settings of railway operators, which leads to differences in, e.g., the ways in which compositions and turnings are incorporated into the models. Second, we have analytically compared two well-known models for rolling stock scheduling: the Composition model that has been proposed for the setting of Netherlands Railways (NS) and the Hypergraph model that has been proposed for the setting of DB Fernverkehr AG (DB). This analytical comparison shows that both models can be used to model the setting of NS and that both formulations are equally strong and thus lead to the same LP-relaxation value. Third, we have compared the performance of the Composition model and Hypergraph model numerically. This numerical comparison shows that the ability of the Hypergraph model to include an additional level of operational details comes at the expense of longer running times for the investigated instances.

Chapter 3

A Variable Neighborhood Search Heuristic for Rolling Stock Rescheduling

This chapter is, up to minor modifications, a direct copy of R. Hoogervorst et al. (2021). "A Variable Neighborhood Search heuristic for rolling stock rescheduling". EURO Journal on Transportation and Logistics *10, 100032.*

3.1 Introduction

There is a significant need for decision support for rolling stock rescheduling at train operating companies. On the one hand, rolling stock rescheduling has a large impact on the passenger satisfaction, as passengers have to stand or even wait for the next train if not enough rolling stock is available to operate a trip. On the other hand, having an efficient rescheduling process in place also limits the number of train units that are needed as a buffer to deal with disruptions, which significantly lowers the capital costs of a train operator.

In this chapter, we focus on the real-time rescheduling of rolling stock after a disruption leads to the cancellation of trips in the timetable. This, e.g., happens when a section of railway infrastructure becomes blocked due to a mechanical failure, which leads to the cancellation of all trips using this section of infrastructure for the duration of the disruption. Alternatively, this may happen when a mechanical failure on a train unit causes this train unit to be unable to continue the trips it is currently operating. An average of 16 disruptions was reported per day in 2019 in the passenger information systems for the Dutch railway network.[1]

As a result of the timetable adjustments due to a disruption, the train units that were planned to operate these canceled trips will be unable to reach the station they were supposed to arrive at after these trips. As these train units may be planned to operate other trips starting at this station, this is likely to lead to further cancellations if no rescheduling is performed. It is these knock-on effects of the disruption on the rolling stock assignment that we focus on in this chapter, where we try to minimize any further cancellations and try to ensure that there is a good balance between the number of available seats and the passenger demand. This is done while trying to minimize the changes that are made compared to the original rolling stock assignment.

Existing literature for rolling stock rescheduling has mostly focused on exact solution methods, of which many were originally developed for the rolling stock scheduling setting. State-of-the-art models include those by Fioole et al. (2006), Borndörfer et al. (2016) and Lusby et al. (2017). While these exact models have been extremely successful in solving a large number of variations of the rolling stock rescheduling problem, they also have some fundamental difficulties when used in practice for real-time rolling stock rescheduling.

Most importantly, these exact methods generally consider a simplified version of the rolling stock rescheduling problem that is faced by dispatchers. For example, the model of Fioole et al. (2006) considers all train units that are of the same rolling stock type as fully interchangeable. While this is a realistic assumption in rolling stock scheduling, where the exact train unit is still likely to change between the moment of planning and the day of operation, choosing the correct train unit is often important in real-time rescheduling. This is, e.g., the case when a train unit has a small mechanical defect which limits how the train unit can be used. A broken windshield wiper may, for example, cause one of the two cabins of a train unit to become unusable for a driver, which implies that this cabin cannot be at the front of

[1] `https://www.rijdendetreinen.nl/en/statistics/2019`

the train.

Numerous papers have extended the existing models for issues encountered in rescheduling. Examples include the incorporation of maintenance constraints by Wagenaar et al. (2017a), the incorporation of train delays as considered in Chapter 4 and the incorporation of deadheading by Wagenaar et al. (2017b). Moreover, Nielsen (2011) considers a setting where the turnings between trips at ending stations can be chosen freely, which is referred to as flexible turning. A disadvantage of these approaches is that the time needed to solve these models rises quickly when expanding the model. As a result, exact solution methods often struggle to solve instances when including all, or many, of the details that are considered by rolling stock dispatchers.

Based on the above observation, the main contribution of this chapter is to propose a Variable Neighborhood Search heuristic for the rolling stock rescheduling problem. For this heuristic, we introduce three local search neighborhoods for the rolling stock rescheduling problem, which can be applied to a wide variety of rolling stock rescheduling settings. In addition, to show that the heuristic can easily be extended to more involved rescheduling contexts, especially ones that are hard for exact methods, we show how a fourth neighborhood can be added to deal with the flexible turning setting of Nielsen (2011). We test the heuristic on instances of Netherlands Railways (NS), where we evaluate the quality of the provided solutions by comparing them to an exact solution method.

The remainder of the chapter is organized as follows. In Section 3.2, we define the rolling stock rescheduling problem. In Section 3.3, we discuss the existing literature on rolling stock rescheduling. In Section 3.4, we introduce our Variable Neighborhood Search algorithm and the three neighborhoods that are considered in the algorithm. We extend the heuristic to the rolling stock rescheduling setting with flexible turning in Section 3.5. We then apply the heuristic to instances of NS in Section 3.6. Lastly, we conclude the chapter in Section 3.7.

3.2 The Rolling Stock Rescheduling Problem

In rolling stock rescheduling, we assign rolling stock to the set of trips $\mathcal{T}$ in the timetable. Each trip is characterized by its departure station, arrival station, departure time and arrival time. Let $\mathcal{S}$ be the set of all stations. We will assume that the timetable has already been updated for the initial disruption, implying that all cancellations that follow directly from the disruption have been incorporated into the timetable.

Most train operators use a heterogeneous fleet of train units to operate the trips. Let $\mathcal{R}$ be the set of available train unit types. In this chapter, we focus on self-propelled train units, i.e., no locomotive is needed to pull them. It is often possible to couple train units of compatible types together to form *compositions*, which are sequences of train unit types in $\mathcal{R}$. Let $\mathcal{P}$ indicate the set of all possible compositions. An example of a composition is given in Figure 3.1, where three train units of the ICM family of train units are coupled together to form a composition.

As we consider a rescheduling setting, the size of the rolling stock fleet is fixed. In particular, let $\iota^0_{r,s}$ indicate the number of train units of type $r \in \mathcal{R}$ that start at

Figure 3.1: An example of a composition consisting of two ICM-III train units at the front and back of the composition and one ICM-IV train unit in the middle.

station $s \in \mathcal{S}$. Moreover, let $\iota^{\infty}_{r,s}$ denote the number of train units of type $r \in \mathcal{R}$ that are supposed to end at station $s \in \mathcal{S}$. Such a target number of train units to end at each station is imposed to facilitate the start of the timetable on the next day.

We first assume that fixed rolling stock connections are defined between incoming and outgoing trips at a station. Let $\mathcal{C}$ be the set of these rolling stock *transitions*. Each transition $c \in \mathcal{C}$ is defined by the sets $\mathcal{T}^-_c$ and $\mathcal{T}^+_c$, which indicate the incoming trips and outgoing trips in this transition, respectively. Note that a transition normally links one incoming to one outgoing trip, implying that both sets contain a single trip, and that one of these sets can be empty in case all rolling stock comes from or is moved to the shunting yard. In Section 3.5, we consider the setting with flexible turning, in which the rolling stock connections at terminal stations can be optimized (Nielsen 2011). In that case, determining the set of transitions $\mathcal{C}$ is part of the optimization problem.

The compositions that can be operated on a given trip are limited by infrastructure and operational constraints, such as the maximum platform length at the stations. Let $\mathcal{P}_t \subseteq \mathcal{P}$ denote the compositions that can be used to operate trip $t \in \mathcal{T}$. The way in which compositions can be changed at transitions is also limited. For example, coupling and uncoupling train units is often only possible on one side of the composition due to the station layout. Let $\mathcal{Q}_c$ be the set of possible composition changes for transition $c \in \mathcal{C}$. Each composition change is characterized by the compositions $q(t) \in \mathcal{P}_t$ for each $t \in \mathcal{T}^-_c \cup \mathcal{T}^+_c$, i.e., by the compositions on all incoming and outgoing trips of the transition. Then, let $\mathcal{Q} = \bigcup_{c \in \mathcal{C}} \mathcal{Q}_c$ denote the set of all possible composition changes.

In rolling stock rescheduling our goal is now to assign *duties* to the available train units, where each duty describes the trips that are operated by that unit during the planning horizon and the position it takes in the composition of each of these trips. Let $\mathcal{D}$ be the set of duties, which is often referred to as a rolling stock *circulation*. Note that $\mathcal{D}$ defines both an assignment $\mathcal{T} \to \mathcal{P}$ of compositions to the trips and an assignment $\mathcal{C} \to \mathcal{Q}$ of composition changes to the transitions. The circulation $\mathcal{D}$ is only feasible if the assignment $\mathcal{T} \to \mathcal{P}$ assigns to every trip t a composition in $\mathcal{P}_t$ and if the assignment $\mathcal{C} \to \mathcal{Q}$ assigns to every transition c a composition change in $\mathcal{Q}_c$.

An example of a circulation is given in Figure 3.2. In this example, we consider a timetable that consists of twelve trips ($t_1, \ldots, t_{12}$) between three stations (*Rtd*, *Gd*, *Ut*). The timetable is shown through a time-space diagram, where an arc connects two stations if there is a trip between them. Moreover, the figure shows a circulation consisting of four duties, two for train units of the red (dashed) rolling stock type and two for train units of the blue (solid) rolling stock type. Note that the circulation implies a composition consisting of two train units on trips $t_1, \ldots, t_4$ and compositions of a single train unit on the other trips. Table 3.1 describes three transitions from the example. Transition c_3 is a so-called turning, where a train arrives at a terminal

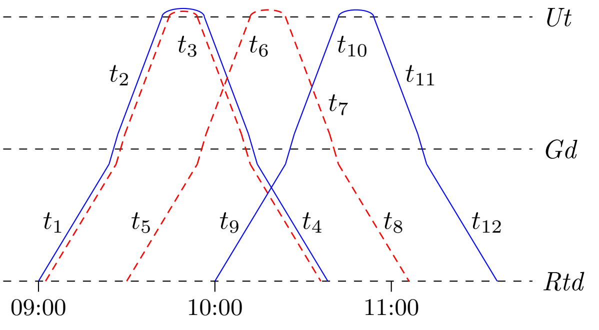


Figure 3.2: An example of a circulation for 4 train units, two of the blue (solid) and two of the red (dashed) type.

stations and returns to the origin station it departed from. During such a transition, no passengers are present in the train and the composition typically switches order.

Table 3.1: Three examples of transitions depicted in Figure 3.2. Here, B and R correspond to a blue (solid) and red (dashed) train unit, respectively.

Transition c	$\mathcal{T}_c^-$	$\mathcal{T}_c^+$	Composition change q
c_1	$\emptyset$	$\{t_1\}$	$q(t_1) = (B, R)$
c_2	$\{t_1\}$	$\{t_2\}$	$q(t_1) = (B, R)$, $q(t_2) = (B, R)$
c_3	$\{t_2\}$	$\{t_3\}$	$q(t_2) = (B, R)$, $q(t_3) = (R, B)$

To judge the quality of a circulation, we assign costs to the compositions and composition changes in the circulation, and to any deviation from the target number of train units to end at each station at the end of the planning horizon. Our most important target is to prevent any cancellations, as cancellations have a large impact on the journey of the passengers. A cancellation is modeled as assigning an empty composition to a trip, which is penalized with a high cost. Moreover, we try to ensure that the chosen compositions offer enough seat capacity while limiting the number of kilometers that the train units make. For the composition changes, we like to avoid any changes in shunting activity. For example, introducing additional coupling of train units may require changes to the duties of the crews at the stations and is thus penalized. Similarly, we would like to minimize any deviations from the target number of train units that are supposed to end at each station at the end of the day, as additional deadheading trips during the night might be necessary to alleviate any off-balances.

The rolling stock rescheduling problem is now formally defined as follows. Given an updated set of trips $\mathcal{T}$ and transitions $\mathcal{C}$, define a set of duties $\mathcal{D}$ with one duty for every train unit, such that the implied composition on trip $t \in \mathcal{T}$ is in $\mathcal{P}_t$ and the implied composition change for transition $c \in \mathcal{C}$ is in $\mathcal{Q}_c$. The aim is to minimize the costs for the assigned compositions and composition changes, and for any deviations from the target number of train units of each type that end at each station.

In Section 3.5, we extend the problem with flexible turning. In this setting, the set of transitions must be determined as part of the rolling stock rescheduling problem.

3.3 Literature Review

The rescheduling of rolling stock is one of the typical steps in the rescheduling process of a passenger railway operator. An overview of the rescheduling process and the steps taken in it is given by Cacchiani et al. (2014). In particular, note that the timetable is normally rescheduled before the rolling stock is rescheduled and that the crew is rescheduled afterward. While some papers have focused on further integrating some of these steps, see, e.g., Veelenturf et al. (2016), it has been shown that this sequential process performs well in practice by Dollevoet et al. (2017).

A large body of literature is available on the rolling stock rescheduling problem, where variations in the problem setting have led to different models. In particular, such model variations are often a result of the different infrastructural and operational contexts encountered by the various railway operators. In this chapter, we specifically focus on passenger railway operators and on rolling stock that consists of train units, is electrically powered and can drive in both directions, implying that no locomotives have to be considered in the rescheduling process.

Numerous exact solution approaches have been proposed for the rolling stock rescheduling problem, where most of these models were originally developed for rolling stock scheduling. A multi-commodity flow based model for rolling stock scheduling has been proposed by Fioole et al. (2006), which was extended by Nielsen et al. (2012) to the rescheduling setting. An important assumption in this model is that all train units of the same type are considered interchangeable, allowing an aggregation of the train units that are stored at the shunting yard of a station at any moment in time.

Alternatively, Borndörfer et al. (2016) solve the rolling stock scheduling problem by finding a flow in an appropriate hypergraph. Recently, a rescheduling approach based on this model has been used by Borndörfer et al. (2019). Other models for rolling stock rescheduling using a hypergraph include those by Borndörfer et al. (2014) and Borndörfer et al. (2017). Unlike Fioole et al. (2006), this model allows to track individual train units, which is used to enforce maintenance constraints based on the distance driven by a train unit. Moreover, while the model of Fioole et al. (2006) can be solved by means of a commercial MIP solver, a column generation approach is needed to solve the hypergraph model due to the large number of hyperarcs. Both the composition model and the hypergraph model are discussed by Reuther (2017).

Instead of making use of a flow based formulation, Cacchiani et al. (2010) and Lusby et al. (2017) generate duties for the individual train units. Due to the large number of possible duties, they, like Borndörfer et al. (2016), also use column generation to solve the model. A different column generation based approach has been considered by Peeters and Kroon (2008). Instead of generating columns that represent duties, they generate columns that correspond to paths in the transition graph, which describes the compositions on the trips and how these compositions can be changed at transitions between trips.

While the above models can all be applied or have been applied to a rolling stock rescheduling context, they are mostly aimed for rescheduling the rolling stock before the day of operation. Numerous papers have extended the above models to include additional details of the real-time rescheduling process. Wagenaar et al. (2017b) consider the possibility of adding deadheading trips and take into account the changes to the passenger demand after a disruption. Similarly, Wagenaar et al. (2017a) consider the effect of maintenance appointments, in which train units need to be present at a station at a specified time for maintenance. The minimization of passenger delays by means of rescheduling the rolling stock assignment is considered in Chapter 4. When considering all of these papers, we see that the inclusion of additional details generally leads to an increase in the computation time.

Concurrently to the successful introduction of exact methods, also heuristics have been proposed to solve the rolling stock scheduling and rescheduling problem. Cacchiani et al. (2019) propose a heuristic for rolling stock scheduling that extends an optimal rolling stock assignment for the peak hours to a rolling stock assignment for the complete day. They apply the heuristic to instances of a railway operator in Northern Italy. Unlike our heuristic, their heuristic does not make use of local search to improve a found solution.

Another matheuristic, i.e., a heuristic based on mathematical programming, for the rolling stock scheduling problem is that of Cacchiani et al. (2013). In this heuristic, the authors apply Lagrangian relaxation based on an arc formulation of the problem to obtain lower bounds. Moreover, a Lagrangian heuristic is used to obtain feasible solutions, in which a constructive procedure transforms an infeasible solution into a feasible one and local search improves the found solution. Like we do in this chapter, also Cacchiani et al. (2013) use neighborhoods to find improved solutions in their Lagrangian heuristic. However, the problem considered is significantly different, as Cacchiani et al. (2013) do not consider the order of train units within a composition and do not consider fixed transitions like we do.

A paper that is closely related to our work in this chapter is the one of Budai et al. (2010), although they focus on a sub-problem of the rolling stock rescheduling problem. In particular, they try to solve any off-balances that are present in the number of train units that end at each station when compared to the planned number of train units to end at each station, a problem they call the Rolling Stock Rebalancing Problem. To solve the off-balances, they exploit the network flow properties of the problem. We will employ a similar idea in two of the presented neighborhoods to find improvements to the current circulation.

3.4 VNS Heuristic for Rolling Stock Rescheduling

We propose a Variable Neighborhood Search (VNS) heuristic to solve the Rolling Stock Rescheduling Problem. VNS is a local search based metaheuristic that has been proposed by Mladenović and Hansen (1997). The main idea in VNS is to iteratively explore multiple neighborhoods, both in terms of local search and in terms of perturbation, where one switches to a larger sized neighborhood if a smaller neighborhood does not yield an improvement to the current solution.

In our VNS heuristic, we consider three different neighborhoods. First, we consider a neighborhood that corresponds to swapping duties between train units. Second, we consider a neighborhood that improves the assignment for a single train unit type. Third, we consider a neighborhood that corresponds to adjusting the composition changes at transitions. In the remainder of this section, we introduce these three neighborhoods and the VNS heuristic that explores these neighborhoods.

3.4.1 Two-Opt Duty Neighborhood

In the Two-Opt Duty neighborhood, we focus on improving the assignment of the different rolling stock types to the trips. We do so by swapping the remaining parts of the duties of two train units, of different rolling stock type, when these train units meet at a station. As a result of this swap, we change the assignment of compositions to trips and that of composition changes to the transitions.

An example of a move in which two duties are exchanged is given in Figure 3.3, where the duties of the highlighted red (dashed) and blue (solid) train units are switched. This swap is possible, as both train units are present at station *Rtd* prior to the departure time of trip t_5. Note that the swap alters the compositions that are used on respectively trips $t_5, \ldots, t_{12}$ and the composition changes that are used on those transitions that precede and succeed these trips. This would, for example, be beneficial if altering these compositions leads to a better matching of capacity with passenger demand on these trips. Specifically, if more passengers are expected on trips $t_5, \ldots, t_8$ than on trips $t_9, \ldots, t_{12}$ and blue (solid) train units offer more seats than red (dashed) units, then the swap improves the objective value.

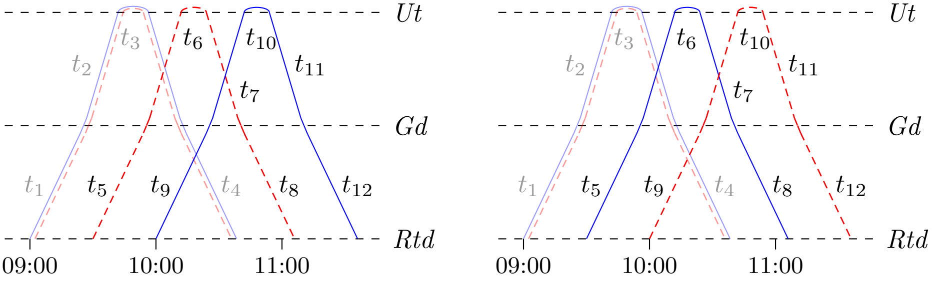


Figure 3.3: A two-opt move on the assigned duties

We can efficiently find the possible moves by noting that the remaining parts of the duties of two train units can only be exchanged if both train units are present at the same station at some moment in time. This follows from the fact that shunting is only executed to couple and uncouple train units. We can then enumerate all rolling stock duties that are available for coupling at some transition, which is the case if the train unit is present at the station at the moment that this coupling action is started. We can then exchange two duties, of different rolling stock types, that are available at the same transition.

The algorithm for finding all two-opt duty exchanges is formalized in Algorithm 1. In this algorithm, `coupled`(d, c) indicates whether duty $d \in \mathcal{D}$ is coupled to a composition at transition $c \in \mathcal{C}$. In particular, this means that the train unit corresponding to duty d is moved from the shunting yard and coupled to an outgoing trip of transition c. Moreover, `findAvailableDuties`() gives a map $f : \mathcal{C} \to 2^{\mathcal{D}}$ that lists for each transition $c \in \mathcal{C}$ the duties that are available for coupling. Note that we only need to consider a swap of duties at a transition if at least one of them is coupled at this transition, as we can consider the same swap at some transition later in time if this is not the case.

Algorithm 1: Two-Opt Duty Neighborhood

```
moves ← ∅;
f ← findAvailableDuties();
foreach c ∈ C do
    foreach {d1, d2} ⊆ f(c) do
        if coupled(d1, c) ∨ coupled(d2, c) then
            moves ← moves ∪ {(d1, d2)};
return moves;
```

Note that Algorithm 1 requires at most $\mathcal{O}(|\mathcal{C}||\mathcal{D}|^2)$ steps, as we can find the available duties by looping over all duties in time $\mathcal{O}(\mathcal{D})$. However, the average number of steps needed by Algorithm 1 is often significantly lower, as normally only a handful of duties are present at the same station at any moment in time. Moreover, the computation time can be further reduced by storing in memory the results of `findAvailableDuties`() and updating the cached results when a new circulation is found.

3.4.2 The Adjusted Path Neighborhood

The main idea behind the Adjusted Path neighborhood is to improve the circulation for one rolling stock type at a time. This implies that we fix the assignment of rolling stock for all-but-one rolling stock types and then improve the assignment for the remaining rolling stock type. We can then exploit the similarity of this simpler problem to a min-cost flow problem to explore this neighborhood efficiently, using an idea inspired by that of Budai et al. (2010).

An example of a move in this neighborhood is shown in Figure 3.4, where we consider a disruption that leads to the cancellation of trips t_5 and t_6 and to no train units being assigned to operate trips t_7 and t_8. Due to the disruption, the transition between trips t_6 and t_7 is updated. In the shown move, we adjust the duty of one blue (solid) train unit, which now operates trips t_7 and t_8 instead of trips t_3 and t_4. In particular, the train unit is uncoupled after trip t_2, moved to the shunting yard, and then coupled to t_7. Note that the move does not alter the duties for the red (dashed) train units. The result of this rescheduling action is that we can prevent two of the four trip cancellations, but also that the number of seats on trip t_3 and t_4 is reduced.

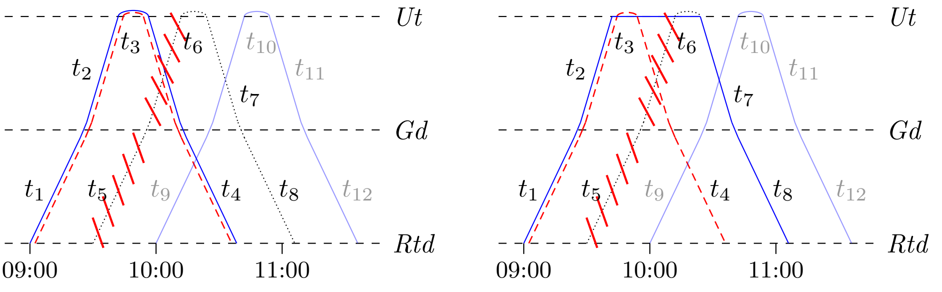


Figure 3.4: An adjustment of the assigned duties for the blue (solid) rolling stock type. The initial disruption is given by the strikethrough line, while a dotted line indicates that no rolling stock is assigned.

By fixing the assignment of all rolling stock types except one, say type a, compositions can only be changed by adding or removing train units of type a from the composition. As we leave the assignment for other types unaltered, we can thus represent each composition by only looking at the number of units of type a that are present at the different positions in the composition. For example, the composition abb, consisting of one train unit of type a in the back and two train units of type b in the front, can only be changed by adding or removing train units at the front of the composition, between the two units of type b and at the end of the composition. Each composition that we consider in this neighborhood for this trip is thus of the form

$$a \cdots a \; b \; a \cdots a \; b \; a \cdots a, \tag{3.1}$$

where we still need to determine the number of units of type a in each group of units of type a. Note that there can also be zero train units of type a in a group, which is, for example, the case for the first two groups in composition abb. This group representation of a composition has been introduced by Budai et al. (2010).

More formally, let g_t be the number of groups of type a in the current composition of trip t. Each composition for trip t can then be represented as a vector $a_t \in \mathbb{Z}_+^{g_t}$, which describes the number of train units of type a present in each group of train units of type a. The problem we consider then becomes to assign feasible vectors a_t to each trip $t \in \mathcal{T}$, such that we use no more train units than available and such that feasible composition changes can be found for each of the transitions. In this way, the assignment of train units for any other type than a remains unaltered.

A Graph Representation

Based on the above group representation, we can consider the duties of the train units of the considered type to represent a flow in a suitable directed acyclic graph $D = (V, A)$. First, we consider the set of nodes V. For each trip $t \in \mathcal{T}$ and each group $g \in \{1, \ldots, g_t\}$ we add nodes $v_{t,g}^{\text{dep}}$ and $v_{t,g}^{\text{arr}}$ representing the departure and arrival of train units in a group of the trip respectively. Moreover, we introduce for each transition $c \in \mathcal{C}$ nodes v_c^{unc} and v_c^{cou}, representing the uncoupling and coupling of train units at this transition respectively.

Second, we consider the set of arcs A. For each trip $t \in \mathcal{T}$ and each group $g \in \{1, \ldots, g_t\}$ we add an arc $(v_{t,g}^{\text{dep}}, v_{t,g}^{\text{arr}})$ to represent that train units are part of this group for the given trip. Furthermore, we add an arc $(v_{t,g}^{\text{arr}}, v_{t',g'}^{\text{dep}})$ for subsequent trips t and t' if there are train units that can operate in group g' of trip t' after operating in group g of trip t. Moreover, we add an arc $(v_{t,g}^{\text{arr}}, v_c^{\text{unc}})$ if train units from group $g \in \{1, \ldots, g_t\}$ can be uncoupled after trip $t \in \mathcal{T}_c^-$ in transition c. Similarly, we add an arc $(v_c^{\text{cou}}, v_{t,g}^{\text{dep}})$ if train units can be coupled to group $g \in \{1, \ldots, g_t\}$ before trip $t \in \mathcal{T}_c^+$ in transition c. Lastly, the coupling and uncoupling nodes are connected by an arc if they occur at the same station and are consecutive in time.

An example of the resulting graph is now given in Figure 3.5 for the blue (solid) train unit type and the circulation as considered in Figure 3.2. The compositions assigned to trips t_1 to t_4 and t_5 to t_8 both contain one red (dashed) train unit. Because of that, there are two blue groups for these trips, as depicted in Figure 3.5. There is only one blue group for trips t_9 to t_{12}, as the composition for these trips consists of a single blue train unit. Moreover, note the arcs that represent the coupling and uncoupling of train units, which connect the nodes at the stations to the trip nodes.

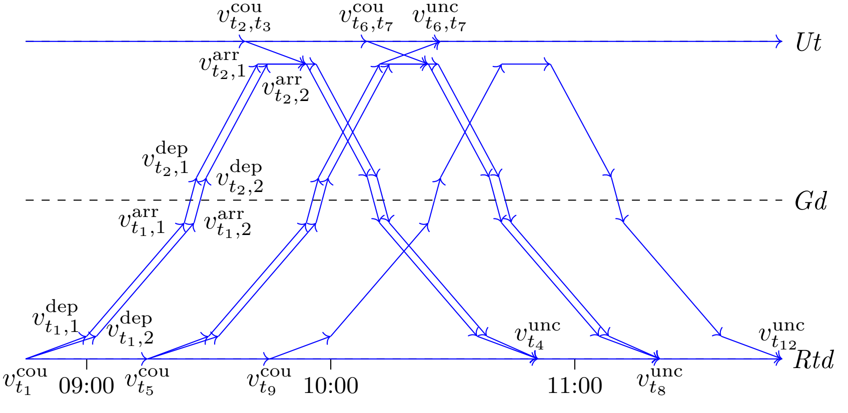


Figure 3.5: Graph representation for the blue (solid) train unit type for the circulation in Figure 3.2. For clarity, only a selection of the node labels is included in the figure.

An Augmenting Path Approach

The current circulation can now be represented as a flow in the graph D. However, not every flow in D will correspond to a feasible flow. For a flow to be feasible, the flow should satisfy the restrictions on the compositions that can be used for the trips and on the composition changes that can be used for the transitions. These restrictions can, in general, not be translated to capacity restrictions on single arcs.

A second complicating factor is that the costs of using an arc are in general not linear in the amount of flow over the arc. Note, for example, that adding a train unit of the considered type a gives a far larger benefit when there is currently a shortage of seats than when there are already enough seats. In the latter case, adding a train unit may even lead to a worse objective value.

However, an interesting observation is that we are still able to determine the cost of adding a train unit to a composition and that of removing a train unit from the composition. Let $f(a_t)$ be the cost of using the composition implied by assignment $a_t \in \mathbb{Z}_+^{g_t}$. Then, the change in cost of adding a train unit to group j is given by

$$f(a_t + e_j) - f(a_t),$$

where e_j is a unit vector with a single one at element j. Similarly, the cost of removing a train unit from group j is given by

$$f(a_t - e_j) - f(a_t).$$

In a similar way, we can determine the change to the costs for changing the number of coupled and uncoupled units at a transition.

Moreover, a second observation is that when adding a train unit or removing a train unit from the composition, we can determine if the new composition is feasible. In essence, we can determine if the compositions given by $a_t + e_j$ and $a_t - e_j$ are feasible. Similarly, we can determine if the new composition change that is is formed by adjusting the flow on one arc in the composition change remains feasible.

The above two observations motivate to look at the residual graph $D' = (V, A')$. In particular, for each arc $(u, v) \in A$, we add the arc (u, v) to A' if we can increase the flow on this arc by a single unit. Furthermore, we add the arc (v, u) to A' if it is possible to decrease the flow on arc (u, v) by a single unit. Let these backward and forward arcs respectively be given by the sets A^* and A^{**}. Moreover, we can assign to each arc a cost that corresponds to the new composition or composition change that is formed.

We argue now that finding an improvement in the assignment of the corresponding rolling stock type corresponds to finding a cycle with negative cost in graph D', where the arc costs are path-dependent. In particular, if a path uses more than one arc corresponding to the same trip or transition, the total arc costs are usually not equal to the sum of the arc costs. For example, it is generally the case that

$$f(a_t + e_i - e_j) \neq f(a_t + e_i) + f(a_t - e_j).$$

Hence, we have to take into account any arcs that are already in the path to determine the cost of a newly encountered arc.

Finding Negative Weight Cycles

As the problem of finding a negative weight cycle with path-dependent arc costs is $\mathcal{NP}$-hard, we use a heuristic approach to find such cycles here. To find negative weight cycles we split the problem into that of freeing an existing duty and finding a new duty for the considered train unit. In particular, this corresponds to first finding a path in D' that only uses backward arcs. For the ending point of this path, a new duty is then found by finding a path from the ending point of this backward path to the starting point of this backward path that only uses forward arcs. This thus restricts the considered cycles to those that can be formed by one backward path and one forward path.

Let $B \subseteq V$ now represent the set of possible starting points of a forward path of a train unit. Such a starting point corresponds to a coupling node v_c^{cou} if a train unit is parked at a station at the start of the planning horizon. Alternatively, it corresponds to a trip node $v_{t,g}^{\text{dep}}$ if a train unit is operating in this group of this trip at the start of the planning horizon. Moreover, let $D^* = (V, A^*)$ and $D^{**} = (V, A^{**})$ be the graphs containing respectively the backward and forward arcs. In addition, let s' be a source and sink node for the backward and forward graph respectively that is connected to the last station node of each station.

The procedure that is used to explore the neighborhood is now formalized in Algorithm 2. In this algorithm, we first fix the train unit type for which we try to find an improvement in the assignment. We then generate for each possible starting node a backward path from the sink node of the graph to the starting point. If such a backward path can be formed, we find a matching forward path from the starting node to the sink node. Together, this backward and forward path form a cycle in the original graph D'. Note that the backward path can be interpreted here as freeing up part of an existing duty for rescheduling, while the forward path can then be seen as finding a new completion of the duty for this train unit for the remainder of the planning horizon.

Algorithm 2: Adjusted Path Neighborhood

```
moves ← ∅;
foreach r ∈ R do
    D* ← createBackwardGraph(r);
    foreach v ∈ B do
        P ← findShortestPath(D*, s', v);
        if P ≠ ∅ then
            D** ← createForwardGraph(r, P);
            P' ← findShortestPath(D**, v, s');
            if P' ≠ ∅ then
                moves ← moves ∪ {(P, P')};
return moves;
```

Note that Algorithm 2 requires at most $\mathcal{O}(|\mathcal{R}||\mathcal{D}||\mathcal{T}|)$ steps. In particular, we execute the algorithm for each rolling stock type $r \in \mathcal{R}$. Moreover, the number of starting points is no more than the number of duties $|\mathcal{D}|$, as each duty is contained in at most one starting point and as we only need to consider those starting points where at least one train unit starts. Lastly, as we consider directed acyclic graphs when finding shortest paths, each shortest path algorithm takes at most $\mathcal{O}(|\mathcal{T}|)$ steps if we assume some fixed maximum length for the compositions. In practice, we can generally speed up the computations. For example, as the backward graph D^* is the same for all backward paths, all single-source shortest paths can be derived at the same time with $\mathcal{O}(|\mathcal{T}|)$ steps. Moreover, as the considered graph is different for each train unit type, we can parallelize the algorithm over the train unit types.

3.4.3 Composition Change Neighborhood

The Composition Change neighborhood adjusts the composition changes at transitions directly. The main motivation for using this neighborhood is that the other neighborhoods tend to only change a few duties at a time. However, as costs are associated with the used composition changes and especially the shunting actions performed in them, these neighborhoods may fail at changing the composition changes in such a way that high costs are avoided.

An example of how a change to a composition change looks is given in Figure 3.6. In this example, originally two units are uncoupled at the transition between trip t_6 and t_7, which are later used to operate trips t_{11} and t_{12}. The composition change for this transition is then changed into one where no shunting occurs and where all train units continue to operate on trips t_7 and t_8. This change might, for example, be beneficial when the composition change before the disruption was also one where no shunting was performed.

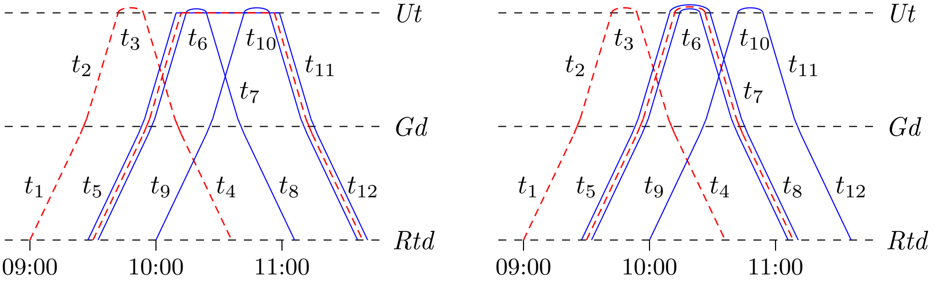


Figure 3.6: Example of changing the composition change for a transition.

If a composition change is altered, the circulation has to be changed accordingly. To do so, we employ an idea similar to the one in the Adjusted Path neighborhood as introduced in the previous section. In particular, we find new duties that match the adjustments that are made to the composition change. These duties then define a new circulation.

Let the current composition change for some transition $c \in \mathcal{C}$ be given by $q \in \mathcal{Q}_c$. Moreover, consider that we want to change q into some new composition change $q' \in \mathcal{Q}_c$. We will require in this neighborhood that all compositions that occur before this transition remain unaltered. This implies that a $q' \in \mathcal{Q}_c$ needs to be found of which the composition $q'(t)$ on the incoming trip $t \in \mathcal{T}_c^-$ is unaltered compared to the current circulation.

The above requirement implies that q' varies from q in the shunting that takes place. We can then identify which train units on the incoming and outgoing trip are affected by this change in shunting. For incoming trips, those are the train units for which the uncoupling is changed, while for outgoing trips those are the train units for which the coupling is changed. We can then create new duty templates for each of the affected train units, which describe the action that is taken in this transition.

To complete these duty templates into actual duties, we use again the group representation graph as introduced in the Adjusted Path neighborhood. In particular, we find for each duty template a starting node in this graph. We then find a backward path from the sink node to this starting node that frees up an existing duty. If such a path can be found, we find a forward path that completes this freed up duty.

The complete algorithm is given in Algorithm 3. In this algorithm, the function `findAdjustedDuties`(c, q) determines the duty templates that are needed for the new composition change. Furthermore, the function `findRelevantNode`(c, d) finds the correct starting node for this duty in the graph representation of this transition. Note that we can only find a move in this algorithm if we find feasible backward and forward paths for each of these duty templates that need to be completed.

Algorithm 3: Composition Change Neighborhood

```
moves ← ∅;
foreach c ∈ C do
    foreach q' ∈ Q_c do
        currentMove ← ∅;
        pathsFound ← true;
        D' = findAdjustedDuties(c, q');
        foreach d ∈ D' do
            D* ← createBackwardGraph(d);
            v ← findRelevantNode(c, d);
            P ← findShortestPath(D*, s', v);
            if P ≠ ∅ then
                D** ← createForwardGraph(d);
                P' ← findShortestPath(D**, v, s');
                currentMove ← currentMove ∪ {(P, P')};
            else
                pathsFound ← false;
        if pathsFound then
            moves ← moves ∪ {currentMove};
return moves;
```

Note that Algorithm 3 requires at most $\mathcal{O}(|\mathcal{C}||\mathcal{Q}||\mathcal{D}||\mathcal{T}|)$ steps, which follows from each shortest path being found with complexity $O(|\mathcal{T}|)$ due to the graphs being directed acyclic graphs. However, the number of needed steps is generally considerably less, as for only a few duties the shunting is altered at a transition. Moreover, only a few composition changes have the required predecessor compositions. As a result, this neighborhood can be explored in a reasonable amount of time even for larger problem instances.

3.4.4 The VNS heuristic

We combine the above neighborhoods into a Variable Neighborhood Search (VNS) heuristic (Mladenović and Hansen 1997). In VNS, we first perturb the current circulation in each iteration, to then apply a local search procedure to improve the perturbed circulation. Perturbation occurs through picking a new circulation randomly from one of the shaking neighborhoods, where we switch to the next neighborhood if no improvement is found in the current one. The general outline of our VNS heuristic is given in Algorithm 4.

Algorithm 4: The VNS heuristic

```
Input: A starting solution c, neighbourhoods N_1, ..., N_{k_max}
while ¬StopConditionSatisfied() do
    k ← 1;
    while k < k_max do
        c' ← Shake(c, N_k);
        c'' ← VariableNeighborhoodDescent(c');
        if f(c'') < f(c) then
            c ← c'';
            k ← 1;
        else
            k ← k + 1;
```

For local search within our VNS heuristic, we employ Variable Neighborhood Descent. In Variable Neighborhood Descent, we explore multiple neighborhoods in a deterministic way. There is thus no diversification and the local search stops when none of the neighborhoods can provide an improvement to the current solution. In our heuristic, we only use the Two-Opt Duty neighborhood N_{duty} and the Adjusted Path neighborhood N_{path} in Variable Neighborhood Descent, where we use the order (N_{duty}, N_{path}) to explore these two neighborhoods. Moreover, we employ a best improvement strategy, where we always explore the complete neighborhood to find the move which results in the largest improvement. Note that computing the improvement made by a move, and checking its feasibility, can be done in $\mathcal{O}(\mathcal{T})$ steps.

As shaking neighborhoods, we then use the Adjusted Path and Composition Change neighborhood in the order (N_{path}, N_{change}), where N_{change} is the Composition Change neighborhood. For both shaking neighborhoods, we draw a move at random from all possible moves that can be made in the neighborhood. By focusing on these two larger sized neighborhoods, we try to ensure that enough variation is present in the found solutions to ensure that local optima are escaped. Moreover, by only sampling from the Composition Change neighborhood, we prevent the relatively large overhead of finding all moves within this neighborhood.

As a stopping criterion, we solely use a maximum elapsed time. This reflects the idea that the heuristic could be stopped at any moment in time by a rolling stock dispatcher to obtain a solution. Hence, if very little time is available, a rolling stock dispatcher can stop the search process and evaluate the solution that is available at that moment in time. Note that this solution is always feasible, as we only consider moves that leave the feasibility of the circulation intact.

3.5 Rolling Stock Rescheduling With Flexible Turning

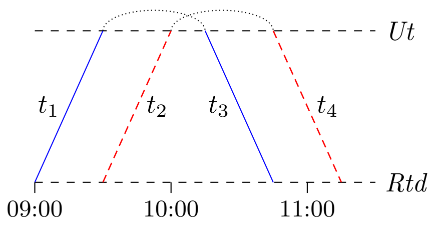
In this section, we extend the heuristic to a setting where the transitions between trips at the ending stations are no longer fixed. Instead, these transitions at ending stations can be changed, implying that incoming trips can be freely reassigned to outgoing trips. This setting for rolling stock rescheduling is often referred to as flexible turning (Nielsen 2011).

3.5.1 Problem Definition

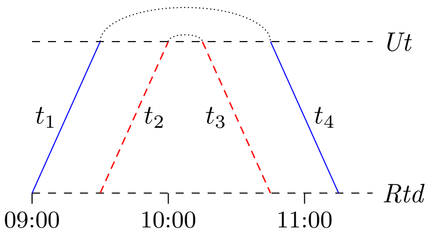
The transitions $\mathcal{C}$ can generally be split into those transitions which occur at in-between stations of a railway line and those that occur at terminal stations. We will refer to the latter as turnings, as trains are often turned around during these transitions (see Table 3.1). Let $\mathcal{C}' \subseteq \mathcal{C}$ give the set of turnings. Unlike transitions at in-between stations, where some passengers often remain in the train to continue their journey towards the terminal station, no passengers remain within the train during turnings. Hence, it is generally possible to change these turnings for a rolling stock dispatcher when faced with a disruption.

Changing these turnings at ending stations essentially corresponds to reassigning incoming trips to other outgoing trips. An example is shown in Figure 3.7. Note how there is originally a turning present between trips t_1 and t_3, and between trips t_2 and t_4. After rescheduling, this is changed to a turning between trips t_1 and t_4, and between trips t_2 and t_3. The impact of this rescheduling action is that the compositions on the outgoing trips are reversed, which impacts the further circulation as well when these train units move on to successor trips later in the planning horizon.

We consider a similar setting as Nielsen (2011) for flexible turning. Here, compositions that arrive on incoming trips can be allocated to any outgoing trip that leaves from the same station after some fixed minimal turning time. Moreover, for each incoming and outgoing trip of a turning, some fixed set of shunting actions can take place after and before the trip, respectively. The composition that arrives from an incoming trip, after shunting has taken place, then remains at the platform of a station until it leaves again, possibly again after shunting takes place, on an outgoing trip.



(a) Original turnings



(b) Alternative turnings

Figure 3.7: Example of changing the turnings at a terminal station

The rolling stock rescheduling problem with flexible turning can then be formulated as the problem of assigning compositions to the trips, composition changes to the transitions and turnings for the ending stations. Next to having to satisfy all requirements of the rolling stock rescheduling problem, we now additionally need to ensure that each incoming trip in a turning is assigned to a feasible outgoing trip. The objective of this problem is similar to that of the rolling stock rescheduling problem, with the difference that we generally want to penalize any changes that are made to the turnings in order to minimize the impact on the station plans and the crew duties.

3.5.2 k-Opt Turning Neighborhood

To incorporate flexible turning into our VNS heuristic, we add a neighborhood. This neighborhood consists of a k-opt procedure for the turnings at the stations, in which we reassign the turnings between k incoming and outgoing trips of a station. Moreover, a repair step is used to adapt the circulation to the changes that are made to the turnings.

Consider the set of turnings $\mathcal{C}'_s \subseteq \mathcal{C}'$ at station $s \in \mathcal{S}$. For a k-opt move, with $k \in \mathbb{Z}_+$ we then consider a set $\mathcal{C}^* \subseteq \mathcal{C}'_s$ with $|\mathcal{C}^*| = k$. Let $\mathcal{T}^-_{\mathcal{C}^*} = \{t \mid t \in \mathcal{T}^-_c, c \in \mathcal{C}^*\}$ and $\mathcal{T}^+_{\mathcal{C}^*} = \{t \mid t \in \mathcal{T}^+_c, c \in \mathcal{C}^*\}$ denote the set of incoming trips and outgoing trips for the turnings in $\mathcal{C}^*$, respectively. Each possible way of reassigning the turnings in $\mathcal{C}^*$ is then given by a perfect matching $g : \mathcal{T}^-_{\mathcal{C}^*} \to \mathcal{T}^+_{\mathcal{C}^*}$, where an incoming trip can only be matched to an outgoing trip when enough time is available to execute the turning.

As an example, a 2-opt move on the turnings in Figure 3.7 has two possible matchings, which are the two cases represented in the figure. Note that one of the possible turning matchings is always the original matching and that the total number of matchings is dependent on the departure and arrival times of the incoming and outgoing trips. Moreover, note that, as in Figure 3.7, each matching can lead to different compositions on the outgoing trips. As these do generally not fit with the remainder of the circulation, we need to repair the circulation based on these new turnings.

In the repair step, we essentially propagate the new compositions on the outgoing trips until the end of the planning horizon. More specifically, we determine for each composition on the outgoing trips the remaining trips it executes during the planning horizon by following the transitions in the timetable. We then assign this composition to all of these trips and pick according composition changes. Moreover, we adjust the compositions of any trips that would originally receive rolling stock from the train units on trips in $\mathcal{T}^+_{\mathcal{C}^*}$. Note that the repair step can fail when we are unable to pick a feasible composition for a trip or a feasible composition change for a transition.

The complete algorithm is shown in Algorithm 5. In this algorithm, the function `possibleAssignments`$(\mathcal{C}^*)$ determines all possible matchings for some set of turnings $\mathcal{C}^*$. Moreover, the function `repair`$(\mathcal{C}^*, g)$ tries to repair the circulation for some set of turnings $\mathcal{C}^*$ and matching g. Note that if the repair step fails, we disregard the current matching.

Algorithm 5: k-Opt Turning Neighborhood

$moves \leftarrow \emptyset$;
foreach $s \in \mathcal{S}$ **do**
  foreach $\mathcal{C}^* \subseteq \mathcal{C}'_s : |\mathcal{C}^*| = k$ **do**
    $G \leftarrow$ `possibleAssignments`$(\mathcal{C}^*)$;
    foreach $g \in G$ **do**
      $s \leftarrow$ `repair`$(\mathcal{C}^*, g)$;
      if $s \neq \emptyset$ **then**
        $moves \leftarrow moves \cup \{(\mathcal{C}^*, s)\}$;
return $moves$;

Algorithm 5 takes at most $\mathcal{O}(|\mathcal{S}||\mathcal{C}'|^k|\mathcal{T}|)$ steps. To see this, note that there are at most $\mathcal{O}(|\mathcal{C}'|^k)$ options of picking a set $\mathcal{C}^*$ for each station and that the number of possible options to reassign the turnings is only dependent on k and is therefore constant. Moreover, note that in the repair step we iterate forward over the trips, thus requiring no more than $\mathcal{O}(|\mathcal{T}|)$ steps.

3.5.3 VNS Heuristic

We consider a similar set-up as before for our VNS heuristic, but adjust the neighborhoods that we use. In particular, we add the 2-opt turning neighborhood both to the existing local search neighborhoods and to the existing shaking neighborhoods. Initial experiments have shown that picking $k = 2$ in both cases leads to a good trade-off between solving time and size of the neighborhood. In both cases, the 2-opt turning neighborhood is the last neighborhood to be explored.

3.6 Computational Results

In this section, we test the heuristic on instances of Netherlands Railways (NS). Our aim is to evaluate the quality of the circulations that are provided by the heuristic, which we do by comparing them to the circulations obtained by the exact solution method of Fioole et al. (2006). Next to looking at the quality of the circulations, we also investigate the performance of the considered neighborhoods. Moreover, we test how the extension of the heuristic to flexible turning performs by comparing the heuristic to the exact method that was proposed by Nielsen (2011) for this setting. Before we look at these numerical results, we introduce the considered rolling stock rescheduling instances and the objective function that is used to evaluate the found rolling stock circulations.

3.6.1 Instances

Our instances are derived from the timetable that was operated by NS in 2018. NS is the largest passenger railway operator in the Netherlands and operates both Intercity and regional (Sprinter) services throughout the country. The network that was operated in the Netherlands by NS in 2018 is shown in Figure 3.8. As the rolling stock planning of NS considers a planning horizon of a day, we consider the timetable as it was on a Tuesday, as Tuesday has on average the highest passenger numbers and is thus seen as the hardest day of the week to plan for.

Instance classes of varying size are created by selecting subsets of the rolling stock types and by including those trips that can be operated by the selected types. An overview of the instance classes is given in Table 3.2, where we describe the included train unit types and give summary statistics about the size of these instance classes. Note that the size of the considered instance classes varies significantly when we add additional rolling stock types.

Table 3.2: Overview of the different instance classes and their properties. Reported are the names of the considered rolling stock types, number of train unit types, number of trips, number of transitions, number of available train units, the average number of allowed compositions per trip and the average number of allowed composition changes per transition.

	Rolling Stock Types	$\|\mathcal{R}\|$	$\|\mathcal{T}\|$	$\|\mathcal{C}\|$	$\|\mathcal{D}\|$	$\bar{\mathcal{P}}_t$	$\bar{\mathcal{Q}}_c$
	ICM	2	1549	1686	124	14.01	23.26
Intercity	ICM-VIRM	4	3764	4031	271	13.15	18.74
	ICM-VIRM-DDZ	6	4241	4560	306	15.23	22.55
	SLT	2	1831	1932	113	7.28	8.30
Sprinter	SLT-SGM	4	2997	3154	182	10.29	11.28
	SLT-SGM-FLIRT	6	3782	3970	229	11.51	12.27

For each of the above instance classes, we create a rolling stock circulation with the exact mixed integer programming (MIP) model as proposed by Fioole et al. (2006), which is also at the heart of rolling stock scheduling at NS. Using an exact approach to determine the circulation for the undisturbed situation allows us to find a circulation according to the same objectives as in rescheduling. This prevents that

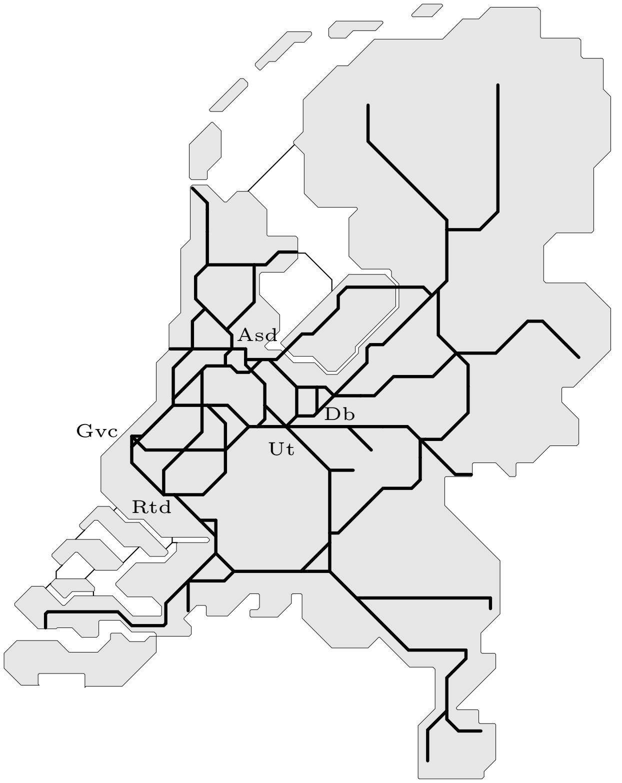


Figure 3.8: The railway network operated by NS in 2018. Included are the four largest stations: Utrecht Centraal (*Ut*), Amsterdam Centraal (*Asd*), Rotterdam Centraal (*Rtd*) and Den Haag Centraal (*Gvc*). Moreover, station Driebergen-Zeist (*Db*) is depicted.

improvements can be made to the original circulation even though no disruptions are faced.

Considered Disruptions

The actual rescheduling instances for a given instance class are now created by incorporating the effect of a disruption on the timetable. We consider two types of disruptions that lead to trip cancellations: small train disruptions that only cause a few cancellations and larger infrastructure failures that lead to many trip cancellations. In this way, we look at the performance of the heuristic for the different use cases in which the heuristic may be employed by dispatchers.

Small cancellations may, e.g., be the result of small technical failures on a train unit or of missing crew to operate a trip. We generate small cancellation instances by picking from all trips in the timetable one trip uniformly at random. We then cancel this trip and all the trips that follow it until the train would arrive at the terminal station for the current passenger service. In particular, note that the train units that operate the canceled trips are unable to get to this terminal station, which impacts any further trips on which these train units were planned to be operated as well.

Secondly, we consider larger disruptions by looking at infrastructure failures in which all tracks between two stations become blocked. Such blockages occur, e.g., when the overhead power lines become damaged on a section of railway infrastructure or when there is a defect in the signaling system for that section of infrastructure. To ease the work for dispatchers, the Dutch infrastructure manager has contingency plans available for such section blockages that specify the adjustments that need to be made in the timetable. We use these contingency plans to create a new timetable for a rescheduling instance.

A starting circulation for our VNS heuristic is created for each rescheduling instance by employing some simple rules to incorporate the effect of the disruption. In essence, we propagate the existing compositions for those trips that have been affected by the disruption, i.e., which are operated by rolling stock of which the duty is affected by the disruption. Moreover, if a trip would originally pick up rolling stock units from the shunting yard and if the duties of those train units are disturbed, then we cancel those trips in the starting circulation. Note that while the cancellations that follow from the disruption can no longer be prevented, it is the aim in the heuristic to prevent these knock-on cancellations.

The Rescheduling Setting

As we consider a real-time rescheduling setting, the information about a disruption only comes in during the day of operation. This implies that a part of the circulation has already been executed and that no changes can be made anymore to that part of the circulation. Moreover, some time is needed to make the decisions and to communicate any changes that are made. As a result, we fix the duties of the rolling stock units up until 30 minutes after the disruption starts. The goal is then to reschedule the circulation for the remainder of the day.

3.6.2 Objective Function

An overview of the considered cost parameters in the objective function is given in Table 3.3. The first three cost components consider the costs that follow as a result of the chosen compositions. First, a large penalty is incurred in the objective function if a trip is canceled, which corresponds to assigning an empty composition to a trip. Second, a penalty is incurred for each passenger that is expected to have to stand on a trip. This penalty is incurred per standing passenger and per kilometer of distance on the trip. Lastly, a penalty is incurred for the use of the train units, which is incurred per kilometer that a train unit is used and is scaled with the length of the train unit in carriages.

Table 3.3: Cost parameters in the objective function.

Element	Objective	Weight
Composition	Cancellation	1000000
	Seat Shortage	0.2
	Mileage	0.1
Shunting	New Shunting	1000
	Changed Shunting	100
	Canceled Shunting	50
Inventories	Ending Inventory Deviation	10000
Transitions	Flexible Turning	1000

The next elements in the objective function relate to the costs that follow from the shunting that occurs at composition changes. Changing the shunting pattern at stations, i.e., at which transitions uncoupling and coupling occurs, may be costly, as it implies that also the local plans at the stations need to be altered. It may, e.g., require additional crew to execute the shunting and additional parking space at the shunting yard to store any uncoupled train units. In the objective function, we differentiate between a new shunting action for a composition change, changing the shunting at a composition change and canceling the shunting at a composition change.

Moreover, any deviations in the ending inventories are penalized, which implies that a penalty is incurred when the number of train units that end at a station deviates from the target number of train units to end there at the end of the planning horizon. This penalty is incurred per unit of deviation for each train unit type and each station. Note that this penalty resembles the costs that a railway passenger operator has to incur to rebalance any deviations overnight.

The last cost parameter relates to the use of flexible turning, which we will consider in Section 3.6.6. Here, we penalize a turning if it is not contained in the original rolling stock circulation. This is done to prevent that too many turnings are changed, which can cause problems in the crew rescheduling phase as crews often stay on the train during turnings. The cost is incurred per changed turning.

3.6.3 Results for Small Disruptions

In this section, we look at the performance of the heuristic on the small disruption instances with fixed rolling stock turnings. To evaluate the performance of the heuristic, we compare the circulations found by the heuristic after one minute of solving time to the optimal circulations obtained by the exact method of Fioole et al. (2006). The experiments have been run on a computer with an Intel Xeon Gold 6130@2.1Ghz processor and 96GB of internal memory. Moreover, CPLEX 12.9 was used to solve the model of Fioole et al. (2006) and the heuristic was programmed in the Java programming language. The obtained results are given in Table 3.4, where 50 instances, each corresponding to a different trip being canceled, have been run for each instance class. Moreover, Figure 3.9 shows the progress of the heuristic over the given computation time.

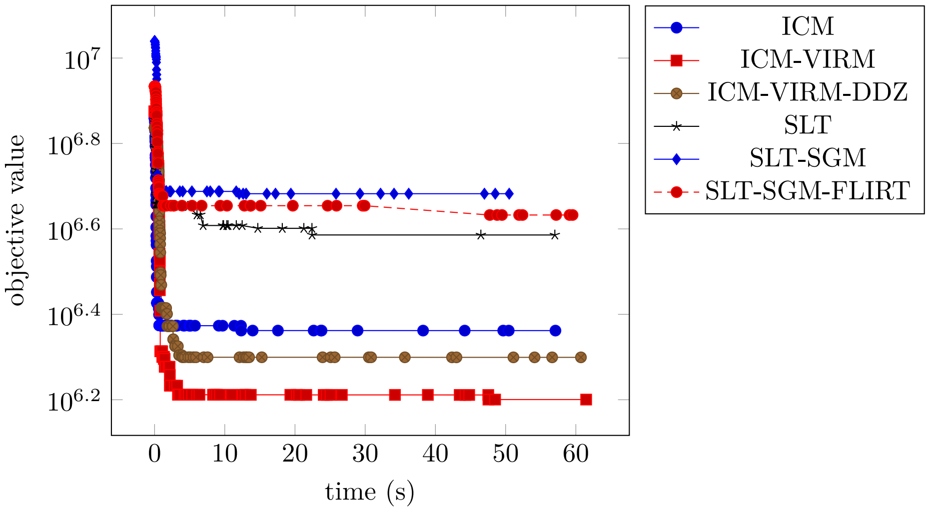


Figure 3.9: Evolution of the objective of the best found solution by the heuristic over time.

The results in Table 3.4 show that the performance of the heuristic varies over the instance classes. For most of the instances, the quality of the circulations found by the heuristic is very close to that of the optimal solutions found by the exact method. This is illustrated by the rather low median objective gap as obtained for the instance classes, especially in relation to the total objective value. In particular, we see that for the SLT and SLT-SGM-FLIRT instance classes we even find the optimal solution for most of the instances within those classes. Moreover, the number of cancellations, which is by far the most dominant factor in our objective function, is for many of the instance classes rather similar for both methods.

At the same time, we see that there are some instances for which the heuristic is unable to match the performance of the exact method. This is illustrated by the rather high maximum gaps that are obtained for the instance classes. This effect

Table 3.4: Results for the small disruptions. Shown for both methods are the obtained objective (*Object.*) and the number of canceled trips (*Canc.*). Moreover, we state the solving time needed by the exact method (*Time*). All these results are averaged over 50 instances. Moreover, we report for the heuristic the absolute objective gap compared to the exact method, where we state the minimum, median, average and maximum gap.

	Exact Method			Heuristic					
						Absolute Gaps			
	Time (s)	Object.	Canc.	Object.	Canc.	min	med.	ave.	max
ICM	1.74	2156767	2.08	2300299	2.22	0	44	143531	3042469
ICM-VIRM	3.46	1420366	1.24	1586121	1.40	0	1060	165755	2027072
ICM-VIRM-DDZ	5.87	1945470	1.74	1991549	1.78	0	222	46079	2027482
SLT	0.45	3630339	3.56	3852331	3.78	0	0	221992	4013357
SLT-SGM	1.16	4166925	4.08	4810782	4.72	0	3	643857	14016779
SLT-SGM-FLIRT	1.54	4266672	4.16	4289680	4.18	0	0	23007	997668

can be contributed to the high penalty assigned to canceling a trip, implying that being unable to prevent a single cancellation, which can be prevented in the exact method, leads to an immediate increase of 1000000 in the obtained objective value. Note that such a difference in the number of cancellations is to be expected for some instances, as the number of changes that need to be made in the circulation to arrive at the minimum number of cancellations might be large, making it significantly more difficult to find such a solution for a heuristic method.

When looking at the performance of the heuristic over the allotted time, as shown in Figure 3.9, we see that by far the largest improvements are made in the first few seconds of the search process. This can be explained by the large number of cancellations that are present at the start of the solving procedure, as numerous trips may no longer have a composition assigned as a result of the initial cancellation. As some of these cancellations can easily be resolved, much progress is made in the first steps of the heuristic. On the other hand, some of the remaining penalty might be hard to reduce, as shown in the tails of the graph for each of the instance classes.

3.6.4 Results for Large Disruptions

In this section, we look at the results for the large disruptions that correspond to section blockages. We again compare the circulations obtained by the heuristic to those obtained by the exact method of Fioole et al. (2006) and use a similar computational setup as in the previous section. In particular, note that we again use a stopping criterion that corresponds to one minute of solving time for the heuristic. The results for the large disruptions are given in Table 3.5, where we consider a section blockage between the stations Utrecht Centraal (*Ut*) and Driebergen-Zeist (*Db*) for the timetable as considered in the ICM-VIRM-DDZ instance class. Note that, opposed to the last section, only one instance is considered for each entry, where each instance corresponds to a different time-frame during which the relevant infrastructure section is blocked.

Table 3.5: Results for the large disruptions. For each instance we state the percentage of fixed trips due to the moment at which the disruption takes place. For both methods we state the obtained objective (*Object.*) and the number of canceled trips (*Canc.*). For the exact method, we also give the solving time (*Time*). For the heuristic, we also state the objective gap compared to the exact method (*Gap*).

		Exact Method			Heuristic		
Instance	Fixed (%)	Time (s)	Object.	Canc.	Object.	Canc.	Gap
Ut-Db: 07-08	11	18.54	6311061	6	8405433	8	2094372
Ut-Db: 09-12	22	12.72	7262961	7	7351712	7	88751
Ut-Db: 12-14	38	7.49	8268163	8	8349301	8	81138
Ut-Db: 16-18	60	4.34	8321181	8	8369584	8	48403
Ut-Db: 21-22	87	2.81	4251822	4	4251841	4	18

The results in Table 3.5 are mostly in accordance with those found in Table 3.4.

We see that for all but one instances the number of cancellations for the heuristic is equal to the number of cancellations for the exact method. As the prevention of cancellations is the most important objective in our problem, this implies that also the objective value is relatively similar between the heuristic and exact method for all these instances. The exception is the 07-08 instance, for which there are 2 additional cancellations in the solution as obtained by the heuristic and for which there is thus a larger difference in the objective value between the heuristic and the exact method.

An interesting observation from the results in Table 3.5 is that the quality of the circulations found by the heuristic tends to improve when the disruption occurs at a later moment in time. This can be explained by the impact that the starting time has on the size of the instance, as the circulation is considered fixed until the start of the disruption. In particular, we see that 87% of the trips can no longer be changed when the disruption occurs from 21:00 - 22:00. Hence, these results show that the heuristic tends to perform better for instances which have a more limited search space.

3.6.5 Performance of the Neighborhoods

To get a better insight into the performance of the heuristic, we look in this section at the contribution of the different neighborhoods to the overall results. To do so, we report summary statistics for both the neighborhoods that we use in local search and for those that we use for shaking. The results that we report correspond to those experiments that were run for the small disruptions. Summary statistics for the local search neighborhoods are given in Table 3.6 and for the shaking neighborhoods in Table 3.7.

Table 3.6: Results for the local search neighborhoods. Reported for each neighborhood are the number of times the neighborhood is explored (*Iter.*), the number of times exploring the neighborhood has led to an improvement (*Impr.*) and the total time spent on exploring this neighborhood (*Time*).

	Two-Opt Duty			Adjusted Path		
	Iter.	Impr.	Time (s)	Iter.	Impr.	Time (s)
ICM	581	15	4	566	177	40
ICM-VIRM	165	6	5	159	62	40
ICM-VIRM-DDZ	161	6	5	154	57	41
SLT	566	28	4	538	206	40
SLT-SGM	252	18	5	234	93	38
SLT-SGM-FLIRT	210	8	5	201	76	39

When looking at the above results, we clearly see the size of the different neighborhoods reflected in the execution time that is spent on these neighborhoods. In particular, we see that while the Two-Opt Duty neighborhood is executed more often than the Adjusted Path neighborhood, far more time is spent on the latter. At the

Table 3.7: Results for the shaking neighborhoods. Reported for each neighborhood are the number of times the neighborhood is explored (*Iter.*) and the total time spent on exploring this neighborhood (*Time*).

	Adjusted Path		Comp. Change	
	Iter.	Time (s)	Iter.	Time (s)
ICM	197	12	111	3
ICM-VIRM	49	12	41	3
ICM-VIRM-DDZ	49	12	30	3
SLT	166	12	165	4
SLT-SGM	71	12	68	4
SLT-SGM-FLIRT	63	12	62	4

same time, we also observe that the Adjusted Path neighborhood is far more often able to find improvements, justifying the larger amount of time spent here. Interestingly, the time spent on the Composition Change neighborhood is limited, which can be explained by the fact that we can efficiently find random moves by iteratively going over the transitions and possible composition changes until we find a potential move.

Another interesting result is that the time spent on the different neighborhoods remains relatively constant over the different instance classes. In particular, we see that we always spent around 40 seconds in local search on the Adjusted Path neighborhood, while we spent around 4 to 5 seconds on the Two-Opt Duty neighborhood. This result can most likely be explained by the fact that both neighborhoods depend in a linear way on the number of trips in the timetable and that the number of trips is the most dominant factor in their running time.

3.6.6 Results for Flexible Turning

To show the performance of the heuristic on rich rolling stock settings, we consider in this section the rolling stock rescheduling problem where flexible turning is allowed at all terminal stations. This implies that we use the version of the heuristic as proposed in Section 3.5, where we use a fourth neighborhood to incorporate flexible turning. We consider again the setting with small disruptions, as in Section 3.6.3. Moreover, the benchmark model we use is now the model for flexible turning as proposed by Nielsen (2011), which extends the model of Fioole et al. (2006). The results of our experiments are given in Table 3.8, where a time limit of 1 minute of solving time is considered for both the heuristic and the exact method. As we found that the exact method is sometimes unable to find a solution within this solving time, and in order to more fairly compare it to the heuristic, we use the same starting solution as in the heuristic to warm start the exact method.

The results in Table 3.8 show that the exact method is unable to solve all instances to optimality within one minute of solving time when including flexible turning. Especially the instances in the ICM-VIRM-DDZ class turn out to be hard to solve,

Table 3.8: Results with flexible turning. For both methods we state the obtained objective (*Object.*) and the number of canceled trips (*Canc.*). Moreover, we state for the exact method the solving time (*Time*) and the number of instances solved to optimality (*Solved*). All results are averaged over 50 instances.

	Exact Method				Heuristic	
	Time (s)	Solved	Object.	Canc.	Object.	Canc.
ICM	6.05	50	2094913	2.02	2199591	2.12
ICM-VIRM	18.41	48	1379904	1.20	1546817	1.36
ICM-VIRM-DDZ	24.14	42	3397814	3.18	1971968	1.76
SLT	6.03	50	1783561	1.72	3150383	3.08
SLT-SGM	20.32	48	1961036	1.88	2828711	2.74
SLT-SGM-FLIRT	18.44	49	3144072	3.04	3809493	3.70

where the exact method is unable to find an optimal solution for 8 out of the 50 instances. This is also reflected in the average objective value and number of cancellations, which turn out to be significantly higher than for the ICM and ICM-VIRM instance classes. However, we also find that instances from instance classes with fewer rolling stock types can be solved relatively quickly, which is the case for the ICM and SLT instance classes.

When we compare the heuristic to the exact method, we see that the heuristic outperforms the exact method for the ICM-VIRM-DDZ instance class. In particular, we see both a lower average objective value and a lower average number of cancellations for the heuristic on this instance class. For the other classes, we again see a similar image as in the setting without flexible turning. For some instances, in this case especially those of the ICM and ICM-VIRM instance class, we see that the solutions found by the heuristic are close to those found by the exact method. For other instances, in this case especially those of the SLT and SLT-SGM instance classes, we see a somewhat larger gap between the heuristic and the exact method.

To further explore the relative performance of the heuristic and the exact solution method on the ICM-VIRM-DDZ instance class, we show in Figure 3.10 for each instance the best found objective value by the exact method over the given solving time. Moreover, we show in Figure 3.11 how the start time of the disruption impacts the solving time of the exact solution method by plotting the start time of the disruption against the solving time for each instance. As we saw that the exact method needs more than one minute of solving time for eight instances in the ICM-VIRM-DDZ instance class, we consider for these figures a longer maximum running time of 30 minutes. Note that such a running time would not be representative for a rolling stock rescheduling setting, but does allow us to analyze if the exact method would perform better with more solving time.

The results in Figure 3.10 show that there are a significant number of instances for which the exact method is only able to find good solutions after a few minutes of solving time. This is illustrated by the jumps in objective value that can be seen around 3 to 5 minutes of solving time. Moreover, it can be seen that for another two instances optimality cannot be proven within 30 minutes of solving time and that for these same instances significant improvements in objective value are still achieved between 5 and 10 minutes of solving time. Overall, the average objective value for the exact method reduces to 1464904, with on average 1.26 cancellations per instance, when given 30 minutes of solving time. Hence, we see that there are on average about 0.5 cancellations more in the solutions of the heuristic, after one minute of solving time, than could be achieved by the exact method when given a longer solving time.

The results for the exact method over 30 minutes of solving time show that there are instances for which the exact method struggles to find solutions within short running times. When looking at Figure 3.11, these hard instances are especially ones that correspond to a cancellation early during the morning. This can be explained by the fact that these are also the instances that contain the largest number of trips. Interestingly, especially instances that occur around the morning peak hours lead to the exact solution method hitting the maximum solving time of 30 minutes. This can likely be explained by the fact that additional trips are operated during the peak hours, increasing the possibilities for flexible turning around the time of the disruption.

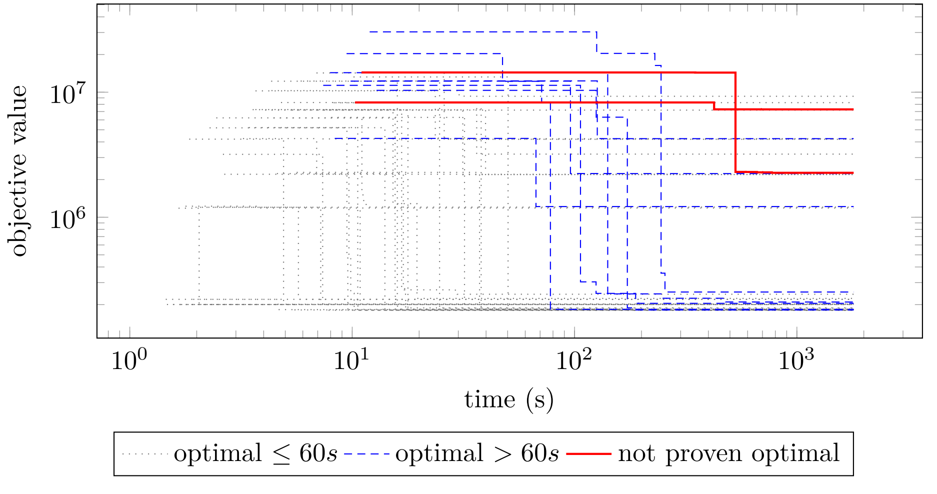


Figure 3.10: Performance of the exact method over 30 minutes of solving time. Both of the axes are shown on a logarithmic scale. The color (style) of a plot indicates if an instance has been solved to optimality within 60 seconds, after 60 seconds, or if no solution could be proven to be optimal within 30 minutes.

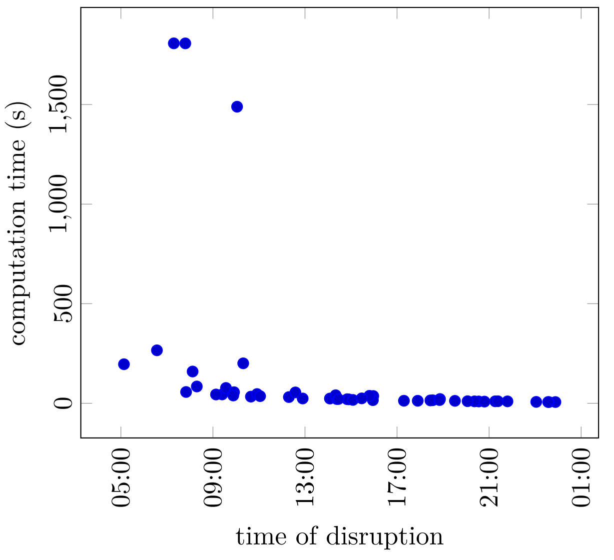


Figure 3.11: Relation between the start time of the disruption and the running time of the exact solution method.

3.7 Conclusion

In this chapter, we have introduced a Variable Neighborhood Search heuristic for the rolling stock rescheduling problem. In this heuristic, we use three new neighborhoods. The first neighborhood considers two-opt swaps on the existing rolling stock duties. The second neighborhood improves the assignment of rolling stock for one rolling stock type at a time by making use of the flow properties of this simpler problem. The third neighborhood improves the assignment of composition changes to the transitions. Moreover, we show that a fourth neighborhood can be used to extend the heuristic to rolling stock rescheduling with flexible turning at the terminal stations.

We have tested our heuristic on instances of Netherlands Railways (NS) for both small and large disruptions. Overall, we find that the heuristic is able to provide circulations that are close to the optimal ones for most of the instances. At the same time, we see that for some instances the number of cancellations is larger than found in the optimal circulation, leading to a higher cost for those instances. While we find that an exact method is able to find optimal solutions quickly for rolling stock rescheduling instances without flexible turning, we show that our heuristic is able to outperform an exact method on some of the instances of the largest instance class when including flexible turning. Moreover, we find that the heuristic is generally able to find good solutions quickly, which allows rolling stock dispatchers to terminate the search early if they feel the current solution is of sufficient quality.

Overall, we believe that our results show the potential of local search based techniques for rolling stock rescheduling. Future research may focus on finding additional neighborhoods, as applications in other fields such as vehicle routing have shown that local search heuristics may benefit significantly from considering a wide variety of neighborhoods. Moreover, the heuristic that we have proposed here might benefit significantly from speeding up the way in which negative weight cycles are found in the Adjusted Path neighborhood.

Chapter 4

Reducing Passenger Delays by Rolling Stock Rescheduling

This chapter is, up to minor modifications, a direct copy of R. Hoogervorst et al. (2020). "Reducing passenger delays by rolling stock rescheduling". Transportation Science *54.3, 762–784.*

4.1 Introduction

Delays are among the largest annoyances experienced by railway passengers. A recent study (KiM 2017) estimates the annual societal costs of train delays and cancellations in the Netherlands to be more than 400 million euro. These costs are not only related to passengers arriving late at their destination, but are, e.g., also a result of the increased uncertainty that passengers feel towards their journey. In this chapter, we focus on reducing the effects of relatively large delays, i.e., delays between 15 and 30 minutes in our application for Netherlands Railways (NS). In particular, we focus on reducing the effects of delays on passengers by means of rolling stock rescheduling.

Traditionally, the planning process of a railway operator consists of three sequential steps: timetabling, rolling stock scheduling and crew scheduling. In the timetabling step, one finds a timetable based on the services that need to be operated. In rolling stock scheduling we then assign train units, i.e., units of rolling stock, to the trips present in this timetable. The aim in rolling stock scheduling is to find a rolling stock assignment that offers enough seat capacity, but which is also not too expensive to operate. Moreover, it should respect the limited number of train units that is available. The found rolling stock schedule, i.e., rolling stock assignment, finally serves as input to the crew scheduling step.

By assigning rolling stock to the trips, the rolling stock schedule creates links between trips which are successively operated by the same train units. These links between trips lead to delay propagation when the assigned rolling stock becomes delayed. This dependence of delay propagation on the rolling stock schedule creates opportunities to reduce the impact of delay on passengers. By changing the rolling stock schedule, we change the links between trips and hence affect the propagation of delay through the railway system. In this way, we may decrease the total delay by making better use of the buffers in the timetable. In addition, we may be able to move the delay to trips with fewer passengers.

We refer to this problem of decreasing the passenger delays by rolling stock rescheduling as the *Passenger Delay Reduction Problem* (PDRP). In this problem, we minimize the delays that are experienced by passengers on the different trips while also taking into account objectives on passenger comfort and operational efficiency. Compared to traditional rolling stock rescheduling, we take into account the propagation of delay through the railway system and actively try to reschedule the rolling stock schedule to decrease the propagation of delay. Moreover, we allow to change the connections between incoming and outgoing trips at some of the terminal stations to minimize this propagation of delay. These connections are generally fixed beforehand in rolling stock rescheduling to lower the computational effort needed to find a rolling stock schedule. Allowing to change such connections is referred to as *flexible turning* (Nielsen 2011).

Our contributions in this chapter are threefold. First, we introduce the PDRP to minimize passenger delays by means of rolling stock rescheduling. Second, we introduce two models to solve the PDRP, which are based on two well-known models for solving the traditional Rolling Stock Rescheduling Problem. Third, we show the applicability of the proposed models to instances of NS. Each of these instances corresponds to a random delay in the timetable operated by NS. Our results show that rolling stock rescheduling can significantly decrease the passenger delays, where especially flexible turning plays a central role in reducing the delays.

The remainder of this chapter is organized as follows. In Section 4.2, we introduce the PDRP. In Section 4.3, we discuss the related literature. In Section 4.4, we discuss two models that are commonly used in rolling stock rescheduling, which are both extended to the PDRP in Section 4.5. In Section 4.6, we introduce our solution methods for the two proposed models. Finally, we test the proposed methods on instances of NS in Section 4.7 and show the extent to which rescheduling for delays changes the original rolling stock schedule. We conclude the chapter in Section 4.8.

4.2 The Passenger Delay Reduction Problem

In this section, we introduce the Passenger Delay Reduction Problem (PDRP) that aims to reduce passenger delays by means of rolling stock rescheduling. We describe the PDRP in the existing rolling stock (re-)scheduling setting of Fioole et al. (2006).

4.2.1 Rolling Stock (Re-)Scheduling

Rolling stock scheduling and rescheduling deal with assigning rolling stock to the trips in the timetable. In this chapter, we restrict ourselves to rolling stock that is composed of self-powered train units, opposed to locomotive-hauled carriages, as is common for European railway operators. A railway operator generally owns train units of different types, which differ in terms of characteristics such as the number of carriages they contain and whether they are single- or double-deck trains. These characteristics affect the number of passengers these train units can carry.

Train units of compatible types may be combined to form *compositions*. A composition is an ordered sequence of the train unit types that are in a train. An example of a composition, consisting of an ICM-III and an ICM-IV train unit, is given in Figure 4.1. The order in a composition matters, meaning that reordering the train unit types within a composition results in a different composition. Moreover, we assume in this chapter that there are no train unit specific constraints, such as maintenance requirements. This implies that train units of the same type are considered interchangeable and that we only have to be concerned with the train unit types in a composition. We note that a similar strategy as developed by Wagenaar et al. (2017a) may be used to consider maintenance restrictions in our problem setting.

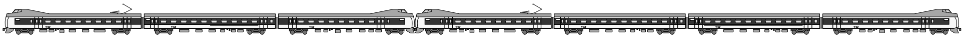


Figure 4.1: Example of a composition with two train units: ICM-IV (4 carriages) in front, ICM-III (3 carriages) in the back

The composition of a train may be changed at *transitions* between trips. A transition links an incoming trip at a station to an outgoing trip and hence links the rolling stock on the predecessor trip to that of the successor trip. Changes to the composition of a train may occur through shunting movements by the *coupling* of additional train units to a composition and by the *uncoupling* of train units from a composition to the shunting yard. Each such possible way of changing the composition at a transition, including that of not changing the composition at all, is captured by a *composition change.*

The total number of train units that is available is limited. The number of train units that is available at a certain station at some moment in time is referred to as the *inventory* of that station. Of particular interest are the starting inventory and ending inventory, which represent the number of train units present at the start and end of the planning horizon. In this chapter, we assume that the starting inventory of train units is fixed, while a target ending inventory is known for each station.

The aim of the *Rolling Stock Scheduling Problem* is to find an assignment of compositions to the trips in the timetable, such that the composition changes implied by these compositions are feasible and such that the starting inventory of train units is respected. All this is done under a large set of mutually conflicting objectives which take into account factors on passenger comfort and operational efficiency. Based on this assignment of compositions to trips we can find an assignment of the individual train units to the trips. This allocation is referred to as the rolling stock *circulation.*

An example of a rolling stock circulation is given in Figure 4.2 for a timetable that includes three stations and eight trips. This rolling stock circulation includes three train units, all of which are part of the starting inventory of station *Rtd.* Moreover, two train units start with trip t_1, while the other train unit starts on trip t_5. Note how shunting takes place at station *Ut*, where a train unit is uncoupled at the transition between trip t_2 and t_3, after which it is coupled at the transition between t_6 and t_7.

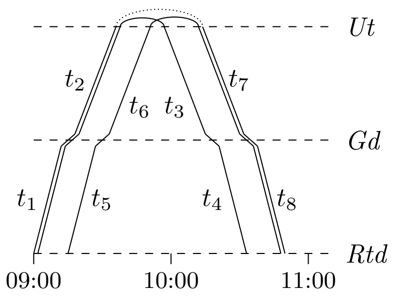


Figure 4.2: Time-space diagram for a timetable that includes three stations (*Rtd*, *Gd*, *Ut*) and eight trips ($t_1, \ldots, t_8$). Moreover, the rolling stock circulation of three train units is shown. Here, the dotted line indicates the shunting of a train unit.

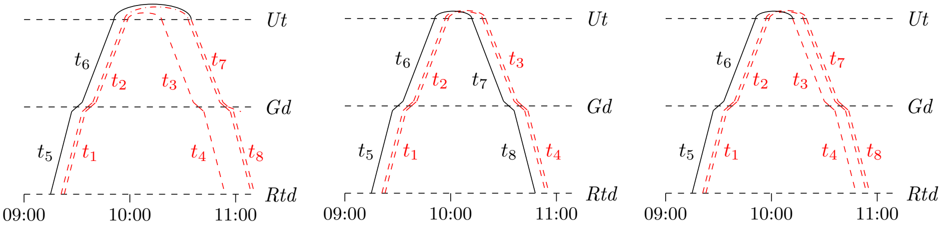
In contrast to the Rolling Stock Scheduling Problem, where we generally find a circulation from scratch, we reschedule an existing circulation in the *Rolling Stock Rescheduling Problem (RSRP).* Rescheduling is performed in the real-time rescheduling phase when a disruption leads to changes in the original timetable. For example, rescheduling may be performed when a blockage of railway infrastructure between two stations leads to the cancellation of trips. Due to the dynamic nature of disruptions, rolling stock rescheduling is generally performed every time new information about disruptions comes in. As each rescheduling step requires communication with the crews, an important objective in rolling stock rescheduling is to minimize the deviation from the original circulation.

4.2.2 The Passenger Delay Reduction Problem

In the PDRP, we again consider the setting of real-time rolling stock rescheduling, but now for a type of disruption that is not considered in the RSRP: a delay that has occurred for a trip, or possibly multiple trips, in the timetable. Such delays occur commonly in practice, e.g., as a result of small technical failures on train units or due to increased dwell times at stations. We focus on delays that cannot quickly be absorbed by slacks in the timetable, which are for NS assumed to be in the order of 15 to 30 minutes.

The rolling stock schedule impacts the propagation of these initial delays in two main ways. First, the transitions that are present between incoming and outgoing trips link the rolling stock that is operated on these trips. These links lead to delay propagation when the rolling stock on the predecessor trip of a transition becomes delayed. Second, the shunting of train units in the rolling stock circulation may lead to further delay propagation. This occurs when a train unit is uncoupled at some transition with a delay and afterwards coupled at some other transition with insufficient time being present between the transitions to absorb the delay.

An illustration of both influences is given in Figure 4.3a, which shows how an initial delay on trip t_1 propagates in the timetable presented in Figure 4.2. First, note how, e.g., the transition between trips t_2 and t_3 causes delay propagation between these trips due to rolling stock that arrives too late at station *Ut*. Second, note how the delay propagates from trip t_2 to t_7 as an effect of the rolling stock circulation. Here, a train unit is uncoupled from trip t_2 with a delay and afterwards coupled to trip t_7. As the time between the moment of uncoupling and coupling is too short to absorb all the delay, this leads to a delay on trip t_7.



(a) No rescheduling (b) No coupling at t_7 (c) Flexible turning

Figure 4.3: Delay propagation based on the rolling stock circulation of Figure 4.2 under different rescheduling actions. The initial delay is present on trip t_1 and a dashed line indicates that a trip is delayed.

The dependence of delay propagation on the rolling stock schedule creates opportunities to reduce the impact that initial delays have on the passengers. First of all, we can change the circulation of train units in order to prevent that a coupled train unit delays a trip. For example, deciding to cancel the coupling of a train unit at trip t_7 prevents that t_7 becomes delayed. This is illustrated in Figure 4.3b. The effect of this rescheduling action is that the passengers on trip t_7 and t_8 no longer face a delay. However, this rescheduling action also reduces the seat capacity on these trips.

A second way to alter the delay propagation is to change the defined transitions between incoming and outgoing trips at terminal stations, which are known as *turnings*. This assignment of rolling stock of incoming trips to outgoing trips by means of turnings is referred to as the *turning pattern* of a station. As no passengers are present in a train during a turning, opposed to transitions at intermediate stations, we can reschedule the turning pattern without impacting the passengers. This is referred to as *flexible turning* (Nielsen 2011). The effect of flexible turning on delay propagation is illustrated in Figure 4.3c. Note how the turnings are changed at the terminal station *Ut*, where trip t_2 is now connected to trip t_7 and trip t_6 to trip t_3. Moreover, note how flexible turning leads to substantial delay reduction compared to performing no rescheduling and retains the same seat capacity on trips t_7 and t_8 as in the undisturbed scenario.

Just like Nielsen (2011), we assume that flexible turning is only allowed at a set of predetermined stations. Limiting flexible turning to these stations reduces the work that is needed in later stages of the rescheduling process. Moreover, we assume that uncoupling and coupling of train units in a flexible turning may occur respectively after the incoming trip and before the outgoing trip. The composition that comes from the incoming trip, after any shunting actions, then remains parked at the platform of a station until it departs on some outgoing trip.

When determining the delay propagation for a given circulation and given turning patterns at the stations, we consider a limited form of delay propagation in which we only consider delay propagation as a result of delayed rolling stock. This implies in particular that we disregard any delay that is created due to headway constraints between trains. Our motivation for doing so stems from the fact that the delays due to headway constraints are likely to be small and can generally be absorbed by running time supplements.

Moreover, we assume that part of the delay may be absorbed by slacks in the timetable. In this chapter, we assume that delay absorption may happen at three different moments in time. First, running time supplements are generally present in the timetable on top of the nominal trip time. Second, supplements may be available for the time that is planned for a transition. Third, some supplement may be available for the time that is needed between uncoupling a train unit at one transition and coupling it at another transition.

The PDRP is now the problem of finding a circulation and corresponding turning patterns for those stations where flexible turning is allowed. This is done under an objective that includes costs for delays and flexible turning next to the traditional objectives in the RSRP on passenger comfort and operational efficiency. To consider the effect that delays have on the passengers, we additionally take into account the number of passengers that are expected to travel on a trip when determining the costs of a delay. In this way, delays on trips with few passengers are preferred over delays on trips with many passengers. Moreover, by penalizing delays in the objective function, opposed to only minimizing the total delay, we follow the idea that disturbing the original circulation to achieve delay reduction can be costly as well. For example, it might lead to trains operating with too few seats for passengers. In addition, by penalizing flexible turning we try to prevent that changes are made to the turning patterns in cases that these do not help to reduce the delays. This is generally disliked, as it breaks up the regularity of the turnings at stations, thus increasing the workload for crews.

Note that it is now required in the PDRP to keep track of the delay that the individual train units have. In particular, the choice of which train unit to use on a trip may influence the amount of delay that is propagated to this trip. In the remainder of this chapter, we present two ways of tracking the delays that train units have, which leads to two models for the PDRP.

4.3 Related Literature

The rescheduling of rolling stock, as considered in the PDRP, is just one of the steps that need to be taken by railway operators in case a disruption occurs. An overview of the problems faced in this setting of real-time rescheduling is given by Cacchiani et al. (2014), Kroon and Huisman (2011) and Jespersen-Groth et al. (2009). These works also show the interrelation between these problems, where rolling stock rescheduling is generally performed after finding an updated timetable and is succeeded by steps such as crew rescheduling and the rescheduling of the shunting plans at the stations. Dollevoet et al. (2017) show that such a sequential approach often performs well for practical instances. Hence, we restrict ourselves to rolling stock rescheduling in this chapter.

4.3.1 Rolling Stock (Re-)Scheduling Without Delays

Solution methods for the RSRP, that is without taking into account delays, are strongly linked to those for rolling stock scheduling. Early works on rolling stock scheduling of train units include those of Schrijver (1993), Ben-Khedher et al. (1998) and Abbink et al. (2005). One of the first papers to consider a similar setting for shunting, i.e., with the ideas of transitions and composition changes as considered here, is that of Alfieri et al. (2006). However, they assign the rolling stock for a single railway line opposed to for a whole network.

The first paper to consider the rolling stock scheduling problem as described in Section 4.2.1 is that of Fioole et al. (2006). They propose a Mixed Integer Programming (MIP) model to solve this problem, which is referred to as the *Composition Model*. It is based on a multi-commodity flow representation of the problem, where additionally the possible composition changes are taken into account. This approach is for this reason often described as a *flow-based approach*. Nielsen (2011) has adapted the Composition Model for the setting of rolling stock rescheduling. Another flow-based approach has been proposed by Borndörfer et al. (2016), who use a hypergraph-based model to solve a rolling stock scheduling problem that includes regularity and maintenance considerations.

A second solution approach that has been considered is a *path-based approach*. This approach has been explored by Peeters and Kroon (2008), Cacchiani et al. (2010) and Lusby et al. (2017). The main difference between these models is the decomposition that is used, where the decomposition is at a train level in the model of Peeters and Kroon (2008), but at the train unit level in the other models. The advantage of the latter is that individual train unit constraints, such as maintenance restrictions, can be taken into account. In all of these papers, column generation is used to solve the model, due to the exponential number of paths that have to be considered.

A comparison of the network flow and path-based approaches is made by Haahr et al. (2016), who compare the models of Fioole et al. (2006) and Lusby et al. (2017) on instances of NS and DSB S-tog. To achieve this, they extend the model of Lusby et al. (2017) to include the order of train units within a composition. In computational experiments, they find that both models are able to obtain solutions in reasonable time, but that the running times of the Composition Model are shorter for most of the instances. However, they argue that this difference in the running times may be the result of the ability to take into account train unit specific constraints in the path-based approach. In this chapter, we will consider both approaches, as it is unclear whether these results also extend to the setting of the PDRP. To do so, we extend the models of Fioole et al. (2006) and Lusby et al. (2017) to the setting of the PDRP.

4.3.2 Rescheduling for Delays

The main focus in railway disruption management when rescheduling for delays has traditionally been on Train Timetable Rescheduling (TTR). Rescheduling the timetable is often necessary as initial delays lead to timetable conflicts in which multiple trains require the same infrastructure at the same moment in time. The main objective in TTR is then to find a conflict-free timetable that minimizes the impact of these initial delays. An overview of the models that have been proposed for this problem can be found in Cacchiani et al. (2014).

Veelenturf et al. (2016) incorporate rolling stock requirements in TTR. In particular, they allow trips to be retimed or canceled in order to find a feasible timetable in case of large disruptions that lead to a (partial) blockage of railway infrastructure. Their objective is to minimize the total delay and the number of canceled trips. Moreover, to ensure that a feasible circulation can be found, they require that rolling stock is available to operate each trip. When compared to this chapter, they do not allow changing the composition of trains during the day and thus take into account only a small part of the RSRP. Similarly, this chapter does not take into account headway constraints between trains and thus takes into account only a small part of TTR.

Retiming is also dealt with by Veelenturf et al. (2012) for crew rescheduling. By retiming the moment of departure of trains slightly, they make it possible to find feasible crew schedules in cases where no feasible crew schedules can be found without retiming. These retiming decisions lead to delays in the system, which they take into account when determining the starting time for subsequent tasks. Note that also in this chapter we implicitly retime trips, by changing the propagation of delay throughout the system. However, while Veelenturf et al. (2012) use retiming to make the crew schedule feasible, retiming in this chapter follows from trying to minimize the impact that delays have on passengers.

Other papers have considered a more complete integration of rolling stock rescheduling and timetabling. Adenso-Díaz et al. (1999) determine the departure time of trips based on the assignment of rolling stock to the trips. Unlike our work in this chapter, they consider locomotive hauled carriages of which the composition cannot be changed throughout the day. Cadarso et al. (2013) consider the recovery of disruptions in rapid transit networks. They allow to insert emergency trips in the timetable and ensure that enough rolling stock is available to operate these trips and the non-canceled original trips. However, unlike our work in this chapter, they do not consider delays for the original trips.

Another cause for passenger delays is considered by Kroon et al. (2015), who deal with passenger delays due to the overcrowding on trains. In particular, they solve a rolling stock rescheduling problem with dynamic passenger flows, where passengers adapt their journey according to the disrupted timetable. Moreover, passengers compete for the capacity on the trains, as determined by the assignment of rolling stock to the trips. To solve this problem, the authors propose an iterative heuristic that iterates between rolling stock rescheduling and the routing of passengers. Unlike Kroon et al. (2015), we assume the passenger routes to be static and do not consider the effect of overcrowding on trains. Moreover, while Kroon et al. (2015) consider only train cancellations, and do not consider train delays, we do include the effect that rolling stock rescheduling has on the propagation of train delays.

4.3.3 Rolling Stock Rescheduling for Delays

To the best knowledge of the authors, this thesis is the first to consider train delays in rolling stock rescheduling in order to minimize delays for passengers. In particular, we are unaware of any papers that take into account the effect that changing the chosen composition changes at transitions and that changing the turning pattern at stations has on delay propagation. In this chapter, we aim to bridge this gap and employ rolling stock rescheduling as a means of minimizing passenger delays.

4.4 The Composition and Path Model for the RSRP

In this section, we describe two state-of-the-art models for solving the RSRP: the *Composition Model* as proposed by Fioole et al. (2006) and the *Path Model* as proposed by Haahr et al. (2016). To describe these models, we first formalize the problem description, where we (mostly) follow the notation of Nielsen (2011).

Let $\mathcal{T}$ be the set of trips in the timetable and let $\mathcal{S}$ be the set of stations that trips arrive at and depart from. Let $\mathcal{C}$ be the set of transitions and let $s(c) \in \mathcal{S}$

be the station at which transition $c \in \mathcal{C}$ takes place. Moreover, let $\mathcal{T}_c^-$ denote the set of incoming trips at this transition and $\mathcal{T}_c^+$ the set of outgoing trips. Note that one of these sets can be empty for a transition that corresponds to the start or end of a train service. Let $\delta^-(t) \in \mathcal{C}$ and $\delta^+(t) \in \mathcal{C}$ indicate respectively the preceding and succeeding transition for trip $t \in \mathcal{T}$. Moreover, we define $\tau^-(c)$ as the moment in time at which a train unit that is uncoupled from the incoming trip of transition $c \in \mathcal{C}$ arrives at the shunting yard. Similarly, we define $\tau^+(c)$ as the moment in time at which a train unit needs to leave the shunting yard in order to be coupled to the outgoing trip for transition $c \in \mathcal{C}$.

Let $\mathcal{R}$ be the set of train unit types. Moreover, let the possible compositions that can be formed by these train unit types be given by $\mathcal{P}$. The set $\mathcal{P}_t \subseteq \mathcal{P}$ indicates the set of allowed compositions for trip $t \in \mathcal{T}$. Similarly, the set $\mathcal{Q}_c$ indicates the allowed composition changes for transition $c \in \mathcal{C}$. To describe the compositions in a composition change $q \in \mathcal{Q}_c$, let $p_{q,t} \in \mathcal{P}_t$ be the incoming composition of trip $t \in \mathcal{T}_c^-$ for composition change q. Similarly, let $p'_{q,t} \in \mathcal{P}_t$ be the outgoing composition of trip $t \in \mathcal{T}_c^+$ for composition change q. Moreover, let $\iota_{s,r}^0$ be the starting inventory of units of type $r \in \mathcal{R}$ that are available at station $s \in \mathcal{S}$.

For the objective function, we associate costs to respectively the chosen compositions, the chosen composition changes and the resulting ending inventories. Let $c_{t,p}^{\mathrm{co}}$ be the cost of assigning composition $p \in \mathcal{P}_t$ to trip $t \in \mathcal{T}$. Similarly, let $c_{c,q}^{\mathrm{ch}}$ be the cost of assigning composition change $q \in \mathcal{Q}_c$ to transition $c \in \mathcal{C}$. Lastly, we penalize deviations from the planned ending inventories at a station. Let $\iota_{s,r}^{\infty}$ be the number of units of type $r \in \mathcal{R}$ that are planned to end at station $s \in \mathcal{S}$. Then, we assign a cost of $c_{s,r}^{\mathrm{id}}$ to each unit of deviation from $\iota_{s,r}^{\infty}$.

We can now formulate the shared components between the Composition Model and the Path Model. We consider the decision variables

$$X_{t,p} := \begin{cases} 1 & \text{if composition } p \in \mathcal{P}_t \text{ is chosen for trip } t \in \mathcal{T}, \\ 0 & \text{otherwise,} \end{cases}$$

$$Z_{c,q} := \begin{cases} 1 & \text{if composition change } q \in \mathcal{Q}_c \text{ is chosen for transition } c \in \mathcal{C}, \\ 0 & \text{otherwise,} \end{cases}$$

$$I_{s,r}^{\infty} \in \mathbb{Z}_+ := \text{the ending inventory of train units of type } r \in \mathcal{R} \text{ at station } s \in \mathcal{S}.$$

The shared constraints are given by

$$\sum_{p \in \mathcal{P}_t} X_{t,p} = 1 \qquad \forall t \in \mathcal{T}, \tag{4.1}$$

$$X_{t,p} = \sum_{q \in \mathcal{Q}_{\delta^+(t)} : p_{q,t} = p} Z_{\delta^+(t),q} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \tag{4.2}$$

$$X_{t,p} = \sum_{q \in \mathcal{Q}_{\delta^-(t)} : p'_{q,t} = p} Z_{\delta^-(t),q} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t. \tag{4.3}$$

Constraints (4.1) ensure that a feasible composition is chosen for each trip $t \in \mathcal{T}$. Constraints (4.2) and (4.3) link the X and Z variables by ensuring that the compositions on the incoming trip and outgoing trip of a transition match with the chosen composition change for this transition. What remains in formulating the RSRP is the connection between the chosen compositions and the availability of train units. Both models do this by means of modeling the natural flow of train units through the timetable, where the Composition Model is an arc formulation for this flow problem and the Path Model is a path formulation. We will describe these two approaches next.

4.4.1 The Composition Model

The Composition Model links the compositions to the available train units by explicitly modeling the inventory of train units at stations. It does so by keeping track of the number of train units that are coupled and uncoupled at each of the transitions and updating the inventory accordingly. Then, taking into account the limited availability of train units corresponds to requiring that the inventory is non-negative for every station and for every moment in time.

To keep track of coupling and uncoupling at each station, we introduce the decision variables

$$\begin{aligned}
C_{c,r} &\in \mathbb{Z}_+ \coloneqq \text{number of units of type } r \in \mathcal{R} \text{ that is coupled at transition } c \in \mathcal{C},\\
U_{c,r} &\in \mathbb{Z}_+ \coloneqq \text{number of units of type } r \in \mathcal{R} \text{ that is uncoupled at transition } c \in \mathcal{C}.
\end{aligned}$$

Moreover, let $\gamma_{q,r}$ and $\upsilon_{q,r}$ indicate the number of train units of type $r \in \mathcal{R}$ that are respectively coupled and uncoupled at composition change q. The Composition Model is then given by

$$\begin{aligned}
\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c^{\text{co}}_{t,p} X_{t,p} &+ \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} c^{\text{ch}}_{c,q} Z_{c,q} && && (4.4)\\
&+ \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c^{\text{id}}_{s,r} |I^\infty_{s,r} - \iota^\infty_{s,r}| \\
\text{s.t. } & (4.1) - (4.3), \\
& C_{c,r} = \sum_{q\in\mathcal{Q}_c} \gamma_{q,r} Z_{c,q} && \forall c \in \mathcal{C}, r \in \mathcal{R}, && (4.5)\\
& U_{c,r} = \sum_{q\in\mathcal{Q}_c} \upsilon_{q,r} Z_{c,q} && \forall c \in \mathcal{C}, r \in \mathcal{R}, && (4.6)\\
& \iota^0_{s(c),r} - \sum_{\substack{c'\in\mathcal{C}:s(c')=s(c),\\ \tau^+(c')\le\tau^+(c)}} C_{c',r} + \sum_{\substack{c'\in\mathcal{C}:s(c')=s(c),\\ \tau^-(c')\le\tau^+(c)}} U_{c',r} \ge 0 && \forall c \in \mathcal{C}, r \in \mathcal{R}, && (4.7)\\
& I^\infty_{s,r} = \iota^0_{s,r} - \sum_{c\in\mathcal{C}:s(c)=s} C_{c,r} + \sum_{c\in\mathcal{C}:s(c)=s} U_{c,r} && \forall s \in \mathcal{S}, r \in \mathcal{R}, && (4.8)\\
& X_{t,p} \in \{0,1\} && \forall t \in \mathcal{T}, p \in \mathcal{P}_t, && (4.9)\\
& Z_{c,q} \in \{0,1\} && \forall c \in \mathcal{C}, q \in \mathcal{Q}_c, && (4.10)\\
& I^\infty_{s,r} \in \mathbb{Z}_+ && \forall s \in \mathcal{S}, r \in \mathcal{R}, && (4.11)\\
& C_{c,r}, U_{c,r} \in \mathbb{Z}_+ && \forall c \in \mathcal{C}, r \in \mathcal{R}. && (4.12)
\end{aligned}$$

The objective function minimizes the sum of the costs associated to the chosen compositions, to the chosen composition changes and to any deviations from the planned ending inventory at the stations. Note that the absolute value in the last term of the objective function can easily be linearized. Constraints (4.1) – (4.3) are shared with the Path Model. Constraints (4.5) and (4.6) determine, for each train unit type, the number of train units that are respectively coupled and uncoupled at a transition. Constraints (4.7) ensure that the inventory is non-negative at the moments of coupling. Note that it is only needed to keep track of the inventory at the coupling moments $\tau^+(c)$, as a non-negative inventory at a coupling moment implies that the inventory was non-negative since the last coupling moment before it. Constraints (4.8) define the ending inventory for each station. The remaining constraints give the variable domains.

4.4.2 The Path Model

The Path Model links the availability of rolling stock to the compositions by considering train unit paths. Such a train unit path describes the trips that are operated by a train unit during the planning period. By assigning a single path to each train unit in the starting inventory, it is ensured that no more train units are used than are available.

To formulate the Path Model, let Π be the set of all feasible train unit paths, i.e., all those sequences of trips that can be operated by a single train unit during the planning horizon. Moreover, let $\Pi_r \subseteq \Pi$ be the set of train unit paths for a train unit of type $r \in \mathcal{R}$ and let $b(\pi)$ and $e(\pi)$ indicate respectively the starting and ending station of path $\pi \in \Pi$. Furthermore, let ω^t_π indicate whether path $\pi \in \Pi$ contains trip $t \in \mathcal{T}$ and let μ^r_p indicate the number of train units of type $r \in \mathcal{R}$ that are present in composition $p \in \mathcal{P}$. If we consider the decision variables

$$L_\pi := \begin{cases} 1 & \text{if path } \pi \in \Pi \text{ is operated by a train unit,} \\ 0 & \text{otherwise,} \end{cases}$$

the Path Model is given by

$$\begin{aligned}
\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c^{\text{co}}_{t,p} X_{t,p} + \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} c^{\text{ch}}_{c,q} Z_{c,q} \qquad\qquad (4.13)\\
+ \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c^{\text{id}}_{s,r} |I^\infty_{s,r} - \iota^\infty_{s,r}|
\end{aligned}$$

$$\begin{aligned}
\text{s.t. } & (4.1) - (4.3), \\
& \sum_{\pi\in\Pi_r} \omega^t_\pi L_\pi = \sum_{p\in\mathcal{P}_t} \mu^r_p X_{t,p} && \forall r \in \mathcal{R}, t \in \mathcal{T}, && (4.14)\\
& \sum_{\pi\in\Pi_r : b(\pi)=s} L_\pi = \iota^0_{s,r} && \forall r \in \mathcal{R}, s \in \mathcal{S}, && (4.15)\\
& \sum_{\pi\in\Pi_r : e(\pi)=s} L_\pi = I^\infty_{s,r} && \forall r \in \mathcal{R}, s \in \mathcal{S}, && (4.16)
\end{aligned}$$

$$L_\pi \in \{0,1\} \qquad \forall \pi \in \Pi, \tag{4.17}$$

$$X_{t,p} \in \{0,1\} \qquad \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \tag{4.18}$$

$$Z_{c,q} \in \{0,1\} \qquad \forall c \in \mathcal{C}, q \in \mathcal{Q}_c, \tag{4.19}$$

$$I^\infty_{s,r} \in \mathbb{Z}_+ \qquad \forall s \in \mathcal{S}, r \in \mathcal{R}. \tag{4.20}$$

The objective function (4.13) and constraints (4.1) – (4.3) are shared with the Composition Model. Constraints (4.14) link the compositions and the paths, by ensuring that enough paths visit a trip to form the composition that is chosen for this trip. Constraints (4.15) ensure that the starting inventory is respected, by assigning a single path to each train unit present in the starting inventory. Similarly, constraints (4.16) determine the ending inventory at each station. The remaining constraints define the domains of the variables.

4.5 Modeling the PDRP

In this section, we propose two models to solve the PDRP: the *Delay Composition Model* and the *Delay Path Model*. These models extend respectively the Composition Model and Path Model, as described in the last section, to the setting of the PDRP. We present the extensions to these models in two steps: we first model the delay propagation that is caused by an initial delay and afterwards model flexible turning and its impact on delay propagation. The complete models can also be found in Appendix 4.A and Appendix 4.B, respectively.

4.5.1 Delay Propagation

We use some additional notation for the delays and delay propagation. Let $\mathcal{D} = \{0, 1, \ldots, d_u\} \subseteq \mathbb{Z}_+$ be the set of delay sizes, with d_u an upper bound on the possible delays. Moreover, let $\mathcal{T}_{\text{init}} \subseteq \mathcal{T}$ be the set of initially delayed trips, which represent the primary delays in the system, with a delay of $d_t \in \mathcal{D}$ for trip $t \in \mathcal{T}_{\text{init}}$. Let $\Delta_t \in \mathbb{Z}_+$ be the running time supplement for trip $t \in \mathcal{T}$. Similarly, let $\Delta_c \in \mathbb{Z}_+$ be the supplement available in the planned dwell time at transition $c \in \mathcal{C}$. Moreover, let $\Delta_s \in \mathbb{Z}_+$ be the supplement to the time that is required at station $s \in \mathcal{S}$ between uncoupling a train unit from one composition and subsequently coupling it to another composition. Lastly, let c^{de}_t be the cost for each unit of delay on trip t.

To keep track of the delay that a trip has, we introduce the decision variables

$$Y_t \in \mathcal{D} := \text{the delay that trip } t \in \mathcal{T} \text{ has at the moment of arrival.}$$

To penalize the delays, the objective function in (4.4) changes to

$$\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c^{\text{co}}_{t,p} X_{t,p} + \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} c^{\text{ch}}_{c,q} Z_{c,q} + \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c^{\text{id}}_{s,r} |I^\infty_{s,r} - \iota^\infty_{s,r}| + \sum_{t\in\mathcal{T}} c^{\text{de}}_t Y_t. \tag{4.21}$$

Here, the last term in the objective function describes the costs due to delays on the trips. Furthermore, we introduce the constraints

$$Y_t = d_t \qquad \forall t \in \mathcal{T}_{\text{init}} \tag{4.22}$$

to model the initial delays.

To model the propagation of these initial delays, we distinguish again between delay that is propagated at transitions and delay that is propagated due to the shunting of train units. We will refer to delay propagated at a transition as *predecessor propagation*, as delay is passed from a predecessor trip to a successor trip. Delay propagation due to a delayed train unit being coupled at a transition is referred to as *inventory propagation*. In this section, we cover both predecessor and inventory propagation for the Delay Composition Model and Delay Path Model.

We include predecessor propagation in both models by means of the constraints

$$Y_t \geq Y_{t'} - \Delta_{\delta^-(t)} - \Delta_t \qquad \forall t \in \mathcal{T}, t' \in \mathcal{T}^-_{\delta^-(t)}. \tag{4.23}$$

In this case, the delay of any successor trip in the transition is at least as large as that of any predecessor trip, minus any of the delay that is absorbed during either the successor trip itself or during the transition that precedes it. It remains to include inventory propagation, which is done separately for the two considered models.

Delay Composition Model

In the Delay Composition Model, we take into account inventory propagation by means of keeping track of the delay that train units have when they enter and leave the inventory, i.e., when being uncoupled from a composition and being coupled to another composition. The delay of a trip, as caused by the coupling of delayed train units, is then determined by looking at the delay with which these units have left the inventory. Moreover, we take the delays at uncoupling and coupling into account when determining the number of train units that are present in the inventory at any moment in time.

To model this formally, we define the decision variables

$C_{c,r,d} \in \mathbb{Z}_+ :=$ the number of units of type r coupled at transition c with delay $d \in \mathcal{D}$,

$U_{c,r,d} \in \mathbb{Z}_+ :=$ the number of units of type r uncoupled at transition c with delay $d \in \mathcal{D}$,

$$D^i_{c,d} := \begin{cases} 1 & \text{if the incoming (uncoupled) units for transition } c \text{ have a} \\ & \text{delay of } d \in \mathcal{D}, \\ 0 & \text{otherwise,} \end{cases}$$

$$D^o_{c,d} := \begin{cases} 1 & \text{if the outgoing (coupled) units for transition } c \text{ have a} \\ & \text{delay of } d \in \mathcal{D}, \\ 0 & \text{otherwise.} \end{cases}$$

We determine the delay that is incurred at uncoupling and the number of units that are uncoupled with this delay by means of the constraints

$$\sum_{d\in\mathcal{D}} dD^i_{c,d} \geq Y_t \qquad \forall c \in \mathcal{C}, t \in \mathcal{T}^-_c, \tag{4.24}$$

$$\sum_{d\in\mathcal{D}} D^i_{c,d} = 1 \qquad \forall c \in \mathcal{C}, \tag{4.25}$$

$$U_{c,r,d} \leq M_1 D^i_{c,d} \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D}, \tag{4.26}$$

$$\sum_{d\in\mathcal{D}} U_{c,r,d} = U_{c,r} \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}. \tag{4.27}$$

Constraints (4.24) and (4.25) link the delay at the moment of uncoupling to the delay of the predecessor trip. Constraints (4.26) link the number of delayed uncoupled units to the delay at uncoupling. The constant M_1 can be chosen here as the maximum number of units that can be uncoupled, which is generally no more than five in practical instances. Lastly, constraints (4.27) ensure that for each uncoupled train unit an appropriate delay is selected.

We introduce similar constraints for the coupling of train units to trips:

$$Y_t \geq dD^o_{\delta^-(t),d} - \Delta_t \qquad \forall t \in \mathcal{T}, d \in \mathcal{D}, \tag{4.28}$$

$$C_{c,r,d} \leq M_2 D^o_{c,d} \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D}, \tag{4.29}$$

$$\sum_{d\in\mathcal{D}} C_{c,r,d} = C_{c,r} \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}. \tag{4.30}$$

Constraints (4.28) determine the delay of an outgoing trip. Constraints (4.29) relate the delay at coupling to the number of units that are coupled with a delay. The constant M_2 can be chosen here as the maximum number of units that can be coupled at this transition, which is again no larger than five for practical instances. Lastly, constraints (4.30) ensure that for each coupled train unit an appropriate delay is selected.

We link the delays that units have at the moments of uncoupling to the delays that units have at the moment of coupling by means of the inventory. To do so, we replace constraints (4.7) by

$$\begin{aligned} \iota^0_{s(c),r} - &\sum_{\substack{c'\in\mathcal{C}:\\ s(c')=s(c)}} \sum_{\substack{d'\in\mathcal{D}:\\ \tau^+(c')+d'\leq\tau^+(c)+d}} C_{c',r,d'} \\ + &\sum_{\substack{c'\in\mathcal{C}:\\ s(c')=s(c)}} \sum_{\substack{d'\in\mathcal{D}:\\ \tau^-(c')+\left(d'-\Delta_{s(c')}\right)^+\leq\tau^+(c)+d}} U_{c',r,d'} \geq 0 \quad \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D}, \end{aligned} \tag{4.31}$$

where $(\cdot)^+ = \max\{\cdot, 0\}$. Constraints (4.31) ensure that the number of available train units at a station for each moment $\tau^+(c) + d$ is non-negative. Taking into account the delays at coupling, these form all the time moments at which a train unit can leave the inventory of a station. Similarly to the definition of the regular inventory, the current inventory equals the number of uncoupled train units minus the number of coupled units on top of the train units that are present at the station at the start of the day. However, we now need to take into account the delays with which these train units are coupled and uncoupled to determine if they are present at the station.

Delay Path Model

To model inventory propagation in the Delay Path Model, we exploit the fact that we have a train unit path for each of the train units. In particular, we extend these paths in such a way that we can record at which trip a train unit is uncoupled before being coupled to another trip. The delay on the trip at which this unit is coupled then becomes related to the delay on the trip at which the train unit was uncoupled.

Keeping track of uncoupling and coupling between trips is only necessary when the time that a train unit stays in the inventory between these trips is short, as the long buffer between the trips will otherwise ensure that no delay is propagated between these trips. For this reason, let $\mathrm{SS}_t \subset \mathcal{T}$ be the set of trips that can send a unit to the inventory which can delay trip $t \in \mathcal{T}$. A trip t' is thus present in SS_t if the time between the moment of uncoupling from t' and the departure of trip t is not large enough to absorb all delay that can be present on trip t'. We refer to such a shunting movement that can propagate delay as a *short shunting*. Let the binary parameter $\zeta_\pi^{t',t}$ now indicate whether a train unit following path $\pi \in \Pi$ is uncoupled at trip $t' \in \mathrm{SS}_t$ and consecutively coupled at trip t.

We introduce the decision variables

$$S_{t',t} := \begin{cases} 1 & \text{if any train unit is uncoupled at } t' \in \mathrm{SS}_t \text{ and consecutively coupled at} \\ & t \in \mathcal{T}, \\ 0 & \text{otherwise.} \end{cases}$$

We then model inventory propagation by means of the constraints

$$\sum_{\pi \in \Pi} \zeta_\pi^{t',t} L_\pi \leq M_3 S_{t',t} \qquad \forall t \in \mathcal{T}, t' \in \mathrm{SS}_t, \tag{4.32}$$

$$Y_t \geq Y_{t'} - d_u(1 - S_{t',t}) + \tau^-(\delta^+(t')) - \tau^+(\delta^-(t)) - \Delta_{s(\delta^-(t))} - \Delta_t \quad \forall t \in \mathcal{T}, t' \in \mathrm{SS}_t. \tag{4.33}$$

Constraints (4.32) determine if there are any train units that are involved in a short shunting between a trip $t \in \mathcal{T}$ and a trip $t' \in \mathrm{SS}_t$. Constraints (4.33) then determine the delay propagation between such trips while taking into account the absorption of the delays, where delay is passed only if short shunting occurs between the trips. Note that the constant M_3 can be chosen relatively small in practical cases, as it can be set equal to the maximum of the number of uncoupled units from t' and the number of coupled units to t. Both of these numbers tend to be smaller than five for practical instances.

4.5.2 Flexible Turning

To allow for flexible turning, we follow the modeling approach of Nielsen (2011), which consists of defining a pool of compositions that are at the platforms of a station. Just like the inventory defines the number of train units present at the shunting yard of a station, this pool of compositions defines the number of compositions of a certain type that are *parked* at the platforms of a station. Incoming trips at this terminal station can add compositions to this pool by *dropping off* compositions at the platform, which can then be *picked up* by outgoing trips.

In the remainder of this section, we present an adaptation of the model that was proposed by Nielsen (2011) for the setting of the Composition Model. Moreover, we extend the approach to the Delay Path Model and determine the delay propagation that is incurred by a certain turning pattern. The biggest difference in the way that we handle flexible turning compared to Nielsen (2011) is that we do not restrict the number of compositions that can be parked at a station. As our numerical experiments will show, the changes made to the turning pattern are generally limited, meaning that for most stations the number of additionally used platforms is limited as well. However, we note that both the Delay Composition Model and Delay Path Model can be extended to restrict the number of parked compositions.

We again need some additional notation. Let $\mathcal{S}_f \subseteq \mathcal{S}$ be the set of stations where flexible turning can occur. Let $\mathcal{T}_i^f \subseteq \mathcal{T}$ be the set of incoming trips that can engage in flexible turning at their arrival station. Similarly, let $\mathcal{T}_o^f \subseteq \mathcal{T}$ be the set of outgoing trips that can engage in flexible turning at their departure station. Let $\widetilde{\mathcal{C}}$ be the set of transitions at which flexible turning can occur. Moreover, let $\mathrm{UN}_{p,t}$ be the set of compositions that can be dropped off at the platform if trip $t \in \mathcal{T}_i^f$ arrives with composition $p \in \mathcal{P}_t$. Similarly, let $\mathrm{CO}_{p,t}$ be the set of compositions that can be picked up from the platform if trip $t \in \mathcal{T}_o^f$ departs with composition $p \in \mathcal{P}_t$. Furthermore, let $\tau_a(c)$ and $\tau_d(c)$ denote the moment that respectively the composition on the incoming trip of transition $c \in \widetilde{\mathcal{C}}$ becomes available at the platform and the moment that the outgoing trip of c departs. Lastly, let c^{ft} be the cost that is associated to dropping off or picking up a composition, opposed to making use of the original transition.

We keep track of the pool of compositions by means of the decision variables

$$Q_{t,p,p'} := \begin{cases} 1 & \text{if trip } t \in \mathcal{T}_i^f \text{ arrives with composition } p \in \mathcal{P}_t \text{ before dropping} \\ & \text{off composition } p' \in \mathrm{UN}_{p,t}, \\ 0 & \text{otherwise,} \end{cases}$$

$$W_{t,p,p'} := \begin{cases} 1 & \text{if trip } t \in \mathcal{T}_o^f \text{ departs with composition } p \in \mathcal{P}_t \text{ after picking up} \\ & \text{composition } p' \in \mathrm{CO}_{p,t}, \\ 0 & \text{otherwise,} \end{cases}$$

$$P_{c,p} \in \mathbb{Z}_+ := \text{the number of compositions of type } p \in \mathcal{P} \text{ that are parked at station } s(c) \text{ at time } \tau_d(c).$$

We now extend the objective function in (4.21) to take into account flexible turning. We then obtain the objective function

$$\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c^{\mathrm{co}}_{t,p} X_{t,p} + \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} c^{\mathrm{ch}}_{c,q} Z_{c,q} + \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c^{\mathrm{id}}_{s,r} |I^{\infty}_{s,r} - \iota^{\infty}_{s,r}| \tag{4.34}$$
$$+ \sum_{t\in\mathcal{T}} c^{\mathrm{de}}_t Y_t + \sum_{t\in\mathcal{T}^f_i} \sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{UN}_{p,t}} c^{\mathrm{ft}} Q_{t,p,p'} + \sum_{t\in\mathcal{T}^f_o} \sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{CO}_{p,t}} c^{\mathrm{ft}} W_{t,p,p'}.$$

The last two terms in the objective function penalize the dropping off and picking up of compositions. In particular, note that each completed flexible turning leads to a cost of $2c^{\mathrm{ft}}$, as a composition needs to be both dropped off and picked up.

Next, we specify the constraints that are shared between the two models. To allow for flexible turning at the terminal stations we introduce the constraints

$$X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^+(t)}: p_{q,t}=p} Z_{\delta^+(t),q} + \sum_{p'\in\mathrm{UN}_{p,t}} Q_{t,p,p'} \qquad \forall t\in\mathcal{T}^f_i, p\in\mathcal{P}_t, \tag{4.35}$$

$$X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^-(t)}: p'_{q,t}=p} Z_{\delta^-(t),q} + \sum_{p'\in\mathrm{CO}_{p,t}} W_{t,p,p'} \qquad \forall t\in\mathcal{T}^f_o, p\in\mathcal{P}_t. \tag{4.36}$$

Constraints (4.35) replace (4.2) for each trip $t \in \mathcal{T}^f_i$. Similarly, constraints (4.36) replace (4.3) for each trip $t \in \mathcal{T}^f_o$. These two sets of constraints extend the original constraints by including the possibility that a flexible turning can occur respectively after or before a trip. Hence, either the normal transition should be maintained or a composition should be respectively dropped off or picked up at the platform.

We determine the number of parked compositions at any moment in time by means of the constraints

$$P_{c,p} = \sum_{\substack{c'\in\widetilde{\mathcal{C}}: s(c')=s(c),\\ \tau_a(c')\le\tau_d(c)}} \sum_{t\in\mathcal{T}^-_{c'}} \sum_{p': p\in\mathrm{UN}_{p',t}} Q_{t,p',p} \tag{4.37}$$
$$- \sum_{\substack{c'\in\widetilde{\mathcal{C}}: s(c')=s(c),\\ \tau_d(c')\le\tau_d(c)}} \sum_{t\in\mathcal{T}^+_{c'}} \sum_{p': p\in\mathrm{CO}_{p',t}} W_{t,p',p} \qquad \forall c\in\widetilde{\mathcal{C}}, p\in\mathcal{P}.$$

The number of parked compositions is determined as the difference between the number of compositions that have been dropped off so far and the number of compositions that have been picked up so far. Note that it is only required to determine the number of available compositions at the moment of departure of an outgoing trip, as a non-negative size of the pool at these moments in time ensures that the size of the pool is non-negative at any moment in time.

Moreover, we need to take into account the use of flexible turning when determining the delay that is passed to a successor trip. This is achieved jointly for both models by replacing constraints (4.23) by the constraints

$$Y_t \ge Y_{t'} - \Delta_{\delta^-(t)} - \Delta_t - d_u \left(\sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{CO}_{p,t}} W_{t,p,p'} \right) \quad \forall t\in\mathcal{T}^f_o, t'\in\mathcal{T}^-_{\delta^-(t)}. \tag{4.38}$$

for each trip $t \in \mathcal{T}_o^f$. Constraints (4.38) state that the delay propagation from the predecessor trip to the successor trip of a transition should be taken into account unless the trip picks up a composition from the platform. In that case, no rolling stock moves from the predecessor to the successor trip of the transition and hence no delay propagation occurs between these trips. It remains to link flexible turning to the available inventory of train units. Moreover, the delay propagated through the chosen turning pattern needs to be taken into account for both models.

Delay Composition Model

In the Delay Composition Model, flexible turning is linked to the available number of train units by means of the inventory. First, we redefine the number of coupled and uncoupled units at a transition to take into account any shunting that is executed during a flexible turning. We replace the sets of constraints (4.5) and (4.6) by

$$C_{c,r} = \sum_{q \in \mathcal{Q}_c} \gamma_{q,r} Z_{c,q} + \sum_{t \in \mathcal{T}_c^+} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} (\mu_p^r - \mu_{p'}^r) W_{t,p,p'} \quad \forall c \in \widetilde{\mathcal{C}}, r \in \mathcal{R}, \tag{4.39}$$

$$U_{c,r} = \sum_{q \in \mathcal{Q}_c} \upsilon_{q,r} Z_{c,q} + \sum_{t \in \mathcal{T}_c^-} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{UN}_{p,t}} (\mu_p^r - \mu_{p'}^r) Q_{t,p,p'} \quad \forall c \in \widetilde{\mathcal{C}}, r \in \mathcal{R}, \tag{4.40}$$

for each $c \in \widetilde{\mathcal{C}}$. Constraints (4.39) and (4.40) record respectively the number of coupled and uncoupled units, where we take into account specifically any shunting that occurs during flexible turning.

Moreover, we need to redefine the ending inventory. Here, we assume that any parked compositions that are not picked up end up in the ending inventory. Hence, we replace the set of constraints (4.8) by

$$\begin{aligned} I_{s,r}^{\infty} = \iota_{s,r}^0 &- \sum_{c \in \mathcal{C}: s(c)=s} C_{c,r} + \sum_{c \in \mathcal{C}: s(c)=s} U_{c,r} \\ &+ \sum_{c \in \widetilde{\mathcal{C}}: s(c)=s} \sum_{t \in \mathcal{T}_c^-} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{UN}_{p,t}} \mu_{p'}^r Q_{t,p,p'} \\ &- \sum_{c \in \widetilde{\mathcal{C}}: s(c)=s} \sum_{t \in \mathcal{T}_c^+} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} \mu_{p'}^r W_{t,p,p'} \qquad \forall s \in \mathcal{S}_f, r \in \mathcal{R}, \end{aligned} \tag{4.41}$$

for each station $s \in \mathcal{S}_f$.

We now consider the delay propagation that occurs when making use of flexible turning. Similar to what we did for inventory propagation, we explicitly keep track of the delays of dropped off and picked up compositions for flexible turning. We introduce the decision variables

$$F_{t,d,p}^i := \begin{cases} 1 & \text{if trip } t \in \mathcal{T}_i^f \text{ drops off composition } p \in \mathcal{P} \text{ at the platform} \\ & \text{with delay } d \in \mathcal{D}, \\ 0 & \text{otherwise,} \end{cases}$$

$$F^o_{t,d,p} := \begin{cases} 1 & \text{if trip } t \in \mathcal{T}^f_o \text{ picks up composition } p \in \mathcal{P} \text{ from the platform} \\ & \text{with delay } d \in \mathcal{D}, \\ 0 & \text{otherwise,} \end{cases}$$

$P_{c,p,d} \in \mathbb{Z}_+ :=$ the number of compositions of type $p \in \mathcal{P}$ that are parked at station $s(c)$ at time $\tau_d(c) + d$.

First, we determine the delay with which compositions are dropped off by means of

$$F^i_{t,d,p} \leq D^i_{\delta^+(t),d} \qquad \forall t \in \mathcal{T}^f_i, p \in \mathcal{P}, d \in \mathcal{D}, \tag{4.42}$$

$$\sum_{d \in \mathcal{D}} F^i_{t,d,p} = \sum_{p' : p \in \mathrm{UN}_{p',t}} Q_{t,p',p} \qquad \forall t \in \mathcal{T}^f_i, p \in \mathcal{P}. \tag{4.43}$$

Constraints (4.42) ensure that the delay of the dropped off composition matches that of the trip that precedes the turning. Constraints (4.43) ensure that if a composition is dropped off at the station, then also a delay is selected with which this composition is dropped off.

Similarly, the picking up of compositions gives rise to

$$F^o_{t,d,p} \leq D^o_{\delta^-(t),d} \qquad \forall t \in \mathcal{T}^f_o, p \in \mathcal{P}, d \in \mathcal{D}, \tag{4.44}$$

$$\sum_{d \in \mathcal{D}} F^o_{t,d,p} = \sum_{p' : p \in \mathrm{CO}_{p',t}} W_{t,p',p} \qquad \forall t \in \mathcal{T}^f_o, p \in \mathcal{P}. \tag{4.45}$$

Constraints (4.44) ensure that the correct delay is propagated to the successor trip of the flexible turning. Constraints (4.45) ensure that if a composition is picked up from the station, then also a delay is selected with which this composition is picked up.

Lastly, we redefine the number of compositions that are parked at the platforms of a station to take into account any delays that are faced at dropping off and picking up compositions. We replace the set of constraints (4.37) by

$$\begin{aligned} P_{c,p,d} = & \sum_{c' \in \widetilde{\mathcal{C}} : s(c') = s(c)} \; \sum_{\substack{d' \in \mathcal{D}: \\ \tau_a(c') + \left(d' - \Delta_{c'}\right)^+ \leq \tau_d(c) + d}} \; \sum_{t \in \mathcal{T}^-_{c'}} F^i_{t,d',p} \\ & - \sum_{c' \in \widetilde{\mathcal{C}} : s(c') = s(c)} \; \sum_{\substack{d' \in \mathcal{D}: \\ \tau_d(c') + d' \leq \tau_d(c) + d}} \; \sum_{t \in \mathcal{T}^+_{c'}} F^o_{t,d',p} \qquad \forall c \in \widetilde{\mathcal{C}}, p \in \mathcal{P}, d \in \mathcal{D}. \end{aligned} \tag{4.46}$$

Constraints (4.46) count the number of parked compositions of some type $p \in \mathcal{P}$ at each time instant $\tau_d(c) + d$. Taking into account each delay $d \in \mathcal{D}$, these are all the possible moments that a composition can leave the pool of compositions. Again, the number of parked compositions of this type is the number of dropped off compositions of this type minus the number of picked up compositions of this type.

Delay Path Model

In the Delay Path Model, it remains to link the train unit paths to the dropped off and picked up compositions. Again, we introduce some additional properties on the paths. Let $\phi^a_{t,\pi}$ indicate whether a train unit that operates path $\pi \in \Pi$ is dropped off at the platform after trip $t \in \mathcal{T}^f_i$. Similarly, let $\phi^d_{t,\pi}$ indicate whether a train unit that operates path $\pi \in \Pi$ is picked up at the platform before trip $t \in \mathcal{T}^f_o$. We enforce that the dropped off and picked up compositions match the selected paths by means of the constraints

$$\sum_{\pi \in \Pi_r} \phi^a_{t,\pi} L_\pi = \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{UN}_{p,t}} \mu^r_{p'} Q_{t,p,p'} \qquad \forall t \in \mathcal{T}^f_i, r \in \mathcal{R}, \tag{4.47}$$

$$\sum_{\pi \in \Pi_r} \phi^d_{t,\pi} L_\pi = \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} \mu^r_{p'} W_{t,p,p'} \qquad \forall t \in \mathcal{T}^f_o, r \in \mathcal{R}. \tag{4.48}$$

Constraints (4.47) and (4.48) ensure that enough train units are respectively dropped off and picked up from the platform to form the composition that is dropped off or picked up. In contrast to the Delay Composition Model, the ending inventory is automatically adjusted for the remaining compositions in case of the Delay Path Model.

It remains to model the delay propagation that is implied by the chosen turning pattern. The way this is modeled is similar to the way inventory propagation has been modeled for the Delay Path Model in Section 4.5.1. Let ST_t be the set of incoming trips that can drop off a composition at a platform which can delay trip t when this composition is picked up at trip t. Let $\xi^{t',t}_\pi$ indicate whether a train unit is dropped off in a composition at trip $t' \in \mathrm{ST}_t$ and picked up from the platform before operating on trip t when this train unit operates path π. We refer to such a situation as a *short turning.*

To model the delay propagation through short turning, we introduce the decision variables

$$B_{t',t,p} = \begin{cases} 1 & \text{if a short turning occurs between trips } t \text{ and } t' \in \mathrm{ST}_t \text{ with} \\ & \text{composition } p, \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, let $\Delta(t, t')$ indicate the delay absorption when a composition is dropped off at trip t' and picked up at trip t. We model short turning and the corresponding delay propagation by means of

$$\sum_{\pi \in \Pi_r} \xi^{t',t}_\pi L_\pi = \sum_{p \in \mathcal{P}_{t'}} \sum_{p' \in \mathrm{UN}_{p,t'}} \mu^r_{p'} B_{t',t,p'} \qquad \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, r \in \mathcal{R}, \tag{4.49}$$

$$\begin{aligned} Y_t \geq{} & Y_{t'} - \Delta_t - \Delta(t, t') \\ & - M_4 \left(1 - \sum_{p \in \mathcal{P}_{t'}} \sum_{p' \in \mathrm{UN}_{p,t'}} B_{t',t,p'} \right) \qquad \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, \end{aligned} \tag{4.50}$$

$$B_{t',t,p'} \leq \sum_{p^* \in \mathcal{P}_{t'}} Q_{t',p^*,p'} \qquad \forall t \in \mathcal{T}_o^f, t' \in \mathrm{ST}_t, p \in \mathcal{P}_{t'}, p' \in \mathrm{UN}_{p,t'}, \tag{4.51}$$

$$B_{t',t,p'} \leq \sum_{p^* \in \mathcal{P}_t} W_{t,p^*,p'} \qquad \forall t \in \mathcal{T}_o^f, t' \in \mathrm{ST}_t, p \in \mathcal{P}_{t'}, p' \in \mathrm{UN}_{p,t'}. \tag{4.52}$$

Constraints (4.49) link the train unit paths to the composition which is used in the short turning. Constraints (4.50) then propagate the delay between the two relevant trips in case a short turning is actually present, taking into account any absorption of delay. Constraints (4.51) link the composition that is involved in the short turning to the composition that is dropped off at the platform. Similarly, constraints (4.52) link the composition that is involved in the short turning to the composition that is picked up from the platform. Moreover, note that the constant M_4 can be chosen as $M_4 = d_u - \Delta_t - \Delta(t, t')$ to make sure that no delay is passed in the case that no short turning occurs.

Lastly, we adjust the number of parked compositions for the delays on the incoming and outgoing compositions. We can do so by replacing constraints (4.37) by

$$\begin{aligned} P_{c,p} = & \sum_{\substack{c' \in \widetilde{\mathcal{C}} : s(c') = s(c), \\ \tau_a(c') \leq \tau_d(c)}} \sum_{t \in \mathcal{T}_{c'}^-} \sum_{p' : p \in \mathrm{UN}_{p',t}} Q_{t,p',p} \\ & - \sum_{\substack{c' \in \widetilde{\mathcal{C}} : s(c') = s(c), \\ \tau_d(c') \leq \tau_d(c)}} \sum_{t \in \mathcal{T}_{c'}^+} \sum_{p' : p \in \mathrm{CO}_{p',t}} W_{t,p',p} \\ & + \sum_{\substack{c' \in \widetilde{\mathcal{C}} : s(c) = s(c') \\ \tau_d(c') \leq \tau_d(c)}} \sum_{t \in \mathcal{T}_{c'}^+} \sum_{\substack{t' \in \mathrm{ST}_t : \\ \tau_a(\delta^+(t')) > \tau_d(c)}} B_{t',t,p} \qquad \forall c \in \widetilde{\mathcal{C}}, p \in \mathcal{P}. \end{aligned} \tag{4.53}$$

Constraints (4.53) correct for the fact that the temporal ordering between trips may be changed when outgoing trips leave with a delay. In particular, it may occur that an outgoing trip uses a composition that would in the original timetable only arrive after the outgoing trip has already departed.

4.6 Solution Approaches

In this section, we propose solution methods for the Delay Composition Model and the Delay Path Model. Both of these models are Mixed Integer Programming (MIP) models, but the dimensions of these models vary considerably. While the complete Delay Composition Model can generally be stored in memory, the number of paths (variables) in the Delay Path Model is generally so large that the model does not fit in memory. As a consequence, we use an off-the-shelf MIP solver for the Delay Composition Model, while we propose a specialized method for the Delay Path Model in the remainder of this section.

Following the approach of Haahr et al. (2016), we solve the Delay Path Model by Branch-and-Price (B&P). Here, column generation is used in each node of a Branch-and-Bound tree to generate promising variables dynamically, alleviating the need of enumerating all variables. To apply column generation, the problem is split in a (restricted) master and pricing problem, where the master problem provides dual information with respect to the current optimal solution, while the pricing problem finds new variables (columns) based on this dual information. These problems are then solved iteratively until no more promising variables can be generated, proving that the current node in the Branch-and-Bound tree is solved to optimality. For a general overview of Branch-and-Price we refer the reader to Barnhart et al. (1998).

In our method, the (restricted) master problem corresponds to solving the linear relaxation of the Delay Path Model proposed in Section 4.5 with a subset of all train unit paths. This set is initialized in the root node with all those paths that are present in the original solution, i.e., the planned circulation for the undisturbed situation, complemented with a set of artificial columns that ensure feasibility. The pricing problem corresponds to generating new train unit paths and decomposes over the different train unit types.

4.6.1 Solving the Pricing Problem

The pricing problem for each train unit type corresponds to a shortest path problem in a directed graph $G = (V, A)$. This graph extends the one considered by Haahr et al. (2016). For the set of nodes V, we introduce for each trip $t \in \mathcal{T}$ a node v_t^d representing the departure of t and a node v_t^a representing the arrival of t. Moreover, we introduce for each trip $t \in \mathcal{T}$ an inventory node v_t^c that indicates the departure of train units from the inventory before being coupled to t. To model flexible turning, we introduce a picking up node v_t^{pi} and a short turning node v_t^{st} for each trip $t \in \mathcal{T}_o^f$. Finally, a source node v_s and sink node v_t are added to the graph to connect the starting and ending inventories of the different stations.

The arcs belonging to A represent actions taken by the train units. We introduce an arc (v_t^d, v_t^a) for all $t \in \mathcal{T}$ to indicate that a train unit is part of the composition on trip t. Moreover, we introduce for each transition $c \in \mathcal{C}$ and between any pair $t_1 \in \mathcal{T}_c^-, t_2 \in \mathcal{T}_c^+$ a connection arc $(v_{t_1}^a, v_{t_2}^d)$ that indicates that a train unit operates on trip t_2 directly after trip t_1. Next, we construct the station arcs. These are added between those inventory nodes v_t^c that occur at the same station and that are consecutive in time. Moreover, the first inventory node for each station is connected to the source v_s and the last inventory node to the sink v_t.

We now represent the coupling and uncoupling of train units. We add an arc (v_t^c, v_t^d) for each trip $t \in \mathcal{T}$ to indicate that a train unit is coupled to t. For uncoupling, we have to take into account short shunting. Consider some trip $t' \in \mathcal{T}$. For each trip $t \in \mathcal{T}$ such that $t' \in \mathrm{SS}_t$, we introduce an arc $(v_{t'}^a, v_t^d)$ that represents that a train unit is involved in a short shunting between these two trips. In Figure 4.4, this is indicated by the dotted arc from $v_{t_1}^a$ to $v_{t_4}^d$. Moreover, we assign an additional arc to the first coupling moment for which no delay can be passed. Specifically, we assign an arc $(v_{t'}^a, v_{t^*}^c)$, such that t^* is the first trip in time such that $\tau^-(\delta^+(t')) < \tau^+(\delta^-(t^*))$,

$s(\delta^+(t')) = s(\delta^-(t^*))$, and $t' \notin \mathrm{SS}_{t^*}$. This arc indicates that the train unit is moved to the inventory and only coupled to a trip at a moment that no delay can be passed. In Figure 4.4, this is indicated by the dotted arc between $v^a_{t_1}$ and $v^c_{t_5}$.

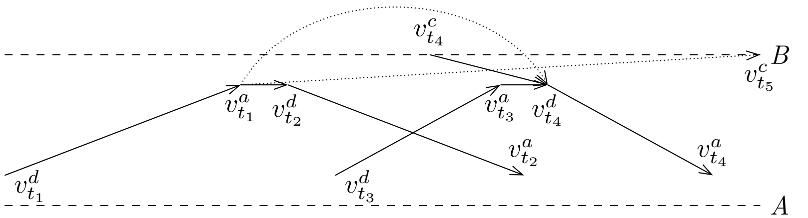


Figure 4.4: Example of the graph found in the pricing problem of the Delay Path Model. Note the dotted short shunting arc between trip t_1 and trip t_4. Moreover, note the dotted uncoupling arc from trip t_1 to trip t_5 that indicates that no delay can be passed.

We now represent the picking up and dropping off of compositions at station $s \in \mathcal{S}_f$. We introduce the arcs $(v^a_{t'}, v^{st}_t)$ and (v^{st}_t, v^d_t) for each pair (t', t) such that $t \in \mathcal{T}^f_o$ and $t' \in \mathrm{ST}_t$. The dotted arcs from $v^a_{t_1}$ to $v^{st}_{t_4}$ and from $v^{st}_{t_4}$ to $v^d_{t_4}$ in Figure 4.5 serve as an example. These arcs represent that a train unit is involved in a short turning between trips t' and t. Moreover, we add an arc $(v^a_{t'}, v^{pi}_{t^*})$ to indicate that some compositions may only be picked up at a trip at which already all delay has been absorbed. Here, t^* indicates the first trip in time such that $\tau_a(\delta^+(t')) < \tau_d(\delta^-(t^*))$, $s(\delta^+(t')) = s(\delta^-(t^*))$ and $t' \notin \mathrm{ST}_{t^*}$. Furthermore, we add an arc (v^{pi}_t, v^d_t) for each $t \in \mathcal{T}^f_o$ to indicate that a train unit is part of a composition that is picked up from the pool of compositions. In addition, we connect those picking up nodes v^{pi}_t that occur at the same station and are consecutive in time. These arcs represent that a train unit is part of a composition that is parked at the platform of a station. Lastly, we connect the last picking up node for each station to the sink v_t.

The pricing problem can now be formulated as that of finding a shortest path from source node v_s to sink node v_t. The costs on the arcs are determined from the duals in the master problem for the considered train unit type. If a shortest path of negative length can be found, it corresponds to a column with negative reduced cost and this column is added to the master problem. Alternatively, if no such path can be found for all train unit types, the current node in the Branch-and-Bound tree is solved to optimality.

4.6.2 Branching Scheme

To obtain an integral solution, we additionally need to impose branching decisions. We adopt all of the branching rules that are used by Haahr et al. (2016). These include the branching on a fractional flow of units that leave the starting inventory of a station, branching on a fractional flow of units that enter the ending inventory of a station and branching on a fractional flow of units that operate on a trip. Moreover,

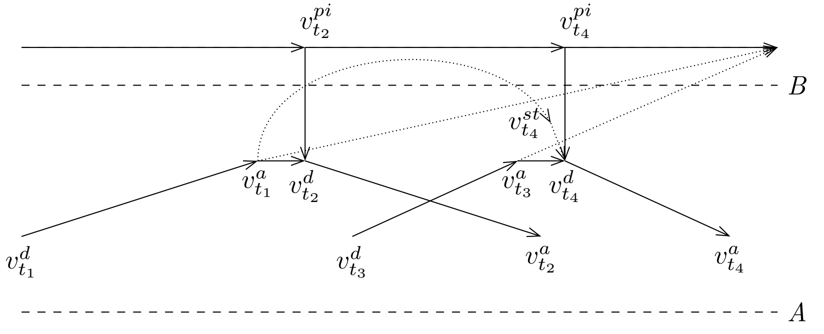


Figure 4.5: Example of the graph found in the pricing problem of the Delay Path Model, where flexible turning is allowed at station B. Note the picking-up arcs for respectively trip t_2 and trip t_4. Moreover, note the dotted short turning arc between trip t_1 and t_4 and the dotted arcs that represent dropping off in case no delay can be passed.

Haahr et al. (2016) branch on subsets of compositions which together have a fractional assignment.

The branching decisions of Haahr et al. (2016) ensure that the composition variables and ending inventory variables are integral, but fractional solutions can still occur in the setting of the PDRP. First, the variables associated to short shunting and short turning may be fractional. Consider the case where some short turning variable $B_{t',t,p}$ is fractional. We then create the branches

$$B_{t',t,p} = 0 \quad \text{and} \quad B_{t',t,p} = 1.$$

We perform branching in a similar way for the short shunting variables $S_{t',t}$.

Moreover, we have to ensure that the composition change variables and the variables that represent the dropping off and picking up of compositions are integral. If no flexible turning can occur at a transition, the integrality of the composition change variables follows from integrality on the composition variables. Let us instead consider a transition where flexible turning can occur, for some trip $t \in \mathcal{T}_i^f$. By constraints (4.35) it holds for each integral solution that there is either some composition change $q \in \mathcal{Q}_{\delta^+(t)}$ such that $Z_{\delta^+(t),q} = 1$ or that there are compositions $p \in \mathcal{P}_t$ and $p' \in \mathrm{UN}_{p,t}$ such that $Q_{t,p,p'} = 1$. If this is not satisfied by a fractional solution, there are specifically a set $B \subseteq \mathcal{Q}_{\delta^+(t)}$ and a set $E \subseteq \{(p,p') \mid p \in \mathcal{P}_t, p' \in \mathrm{UN}_{p,t}\}$ such that

$$0 < \sum_{q \in B} Z_{\delta^+(t),q} + \sum_{(p,p') \in E} Q_{t,p,p'} < 1. \tag{4.54}$$

We then branch by requiring that the above sum is equal to one in one branch and equal to zero in the other. We apply a similar branching decision to enforce that for a trip $t \in \mathcal{T}_o^f$ either a single composition change is chosen that precedes the trip or that a single composition is picked up from the station.

Note that the above branching rules do not necessarily imply that all paths are integral. However, the integrality of the composition variables, composition change variables and the variables related to shunting and turning implies the presence of an integer-valued network flow in the pricing problem graph for each train unit type. It is well known that such a flow can be decomposed into (integral) paths. Moreover, as no costs are directly related to the paths, this implies that we can always find a solution of equal objective value in which all path variables are integral.

4.6.3 Acceleration Strategies

To speed up the Branch-and-Price procedure, we reuse many of the acceleration strategies as proposed by Haahr et al. (2016). First, we apply delayed row generation for the transition constraints (4.2), (4.3), (4.35) and (4.36). In this approach, we first solve the restricted master problem without these sets of constraints and only include these constraints dynamically when we find that a constraint is violated in the optimal solution for this smaller problem. The aim of this approach is to reduce the total time that is spent on solving the restricted master problem, as this time spent on the master problem turned out to be substantial in our computational experiments.

A second strategy to reduce the time spent on solving the master problem is to generate multiple paths in each column generation iteration. In particular, we generate a path for each train unit type and for each possible starting point of the train units. Such a starting point either corresponds to a station or to a trip, depending on the location of train units at the start of the planning horizon. By generating multiple paths in each iteration, we generally have to solve the master problem less often.

As preliminary results showed that solving the complete Branch-and-Price procedure for the Delay Path Model was often prohibitive due to the large number of nodes explored in the Branch-and-Bound tree, we also consider a root node heuristic for this model. In this heuristic, we first solve the root node to optimality by means of column generation. We then solve the master problem with integrality restrictions using only the set of columns that was added in the root node. While this does not necessarily give an optimal solution, this method has been widely applied in literature to obtain high-quality solutions when the available time to solve the model is limited. In our computational experiments, we compare the performance of this method to the full Branch-and-Price procedure.

4.7 Computational Experiments

In this section, we test the two proposed models on instances of NS. Our aim is to compare the performance of the proposed solution methods on real-life instances and to evaluate the extent to which solving the PDRP reduces the passenger delays.

4.7.1 The Problem Instances

The difficulty of a rolling stock instance is influenced by both the size of the timetable, i.e., the number of trips in the timetable, and the choices there are in assigning rolling stock to the trips, i.e., the number of compositions that can be formed by the available train unit types. To test how the proposed models behave under instances of varying difficulty, we consider two timetables that have been operated by NS in 2016: one with Intercity services for the ICM train unit family and one with Regional (Sprinter) services for the SLT train unit family. Both of these timetables concern a Tuesday, which is the day of the week with the highest passenger demand. Basic information about these timetables and the available rescheduling options, in terms of the possible number of compositions and composition changes, is provided in Table 4.1. In particular, note that while both timetables are of roughly equivalent size, the number of compositions that can be formed is larger for the Intercity timetable.

Table 4.1: Summary statistics for the two timetables: the number of trips, number of transitions, average number of allowed compositions per trip and the average number of allowed composition changes per transition.

	Intercity (IC)	Regional (RE)
Trips	1122	1386
Transitions	1207	1614
Compositions	19.39	6.87
Composition Changes	26.65	7.40

Another factor that influences the difficulty of an instance is the moment at which the initial delay occurs, as this determines how long the remaining planning horizon is. In particular, if the disruption occurs at time instant τ, we assume that our remaining planning horizon is from $\tau + 30$ to the end of the day, where the 30 minutes of additional time is needed to find an updated schedule and to communicate the updates to the crews. Hence, a disruption that occurs earlier corresponds to a longer planning horizon and thus to a larger problem. To test the effect of different planning horizons, we split up the problem instances further according to whether the delay occurs during the morning peak (6.30-9.00), off-peak (9.00-16.00) or evening peak (16.00-18.30) hours. Resembling their size, these classes of instances will be referred to as large (L), medium (M) and small (S), giving the six scenarios IC_L, IC_M, IC_S, RE_L, RE_M and RE_S.

For each of the six scenarios we create 25 instances by introducing initial delays into the original timetable. Each instance corresponds to an initial delay on one of the trips in the timetable. This initial delay is drawn uniformly at random between 15 and 30 minutes, while the trip on which this initial delay occurs is chosen randomly from all trips that are operated during the relevant moment of the day.

Based on the generation of delays, we assume an upper bound on the delays d_u of 30 minutes. Moreover, we measure all delays in minutes for all instances, implying that $\mathcal{D} = \{0, \ldots, 30\}$. For the delay buffers, we assume that a buffer of 1 minute is available for each trip on top of the nominal trip time. Moreover, we assume a buffer of 3 minutes for all transitions that correspond to turnings. No buffers are assumed for other transitions or for the time between uncoupling and coupling.

Flexible turning for the generated instances is allowed at the two terminal stations for the line that is affected by the initial delay, i.e., those terminal stations where the affected train departed from and will eventually arrive at. Furthermore, we assume that only trips which arrive or depart within four hours of the initial delay can participate in flexible turning. In Section 4.7.5 we look at the effect of varying the set of flexible turning locations and at the effect of varying the duration during which flexible turning is allowed. In this way, we quantify the costs of limiting flexible turning to a fixed time interval and to a fixed set of stations.

4.7.2 Objective Function

As is common in rolling stock (re)scheduling, our objective consists of a large number of mutually conflicting objectives which are weighed against each other by their respective coefficients in the objective function. In Table 4.2 we give the considered cost components, their coefficient in the objective function and the parameter in the objective function that they are part of. *Seat Shortage* indicates the cost associated with not offering a passenger a seat, which is accounted for per kilometer traveled. *Mileage* refers to the cost of using a train unit, which is accounted for per kilometer of use and per carriage that is part of the train unit. The next three components correspond to costs related to deviating from the originally communicated shunting plan. Here, we distinguish a new shunting movement from a changed or a canceled one, as these are likely to lead to a different level of disruption to the existing shunting plan. *Ending Inventory Deviation* corresponds to the cost of deviating in the ending inventory from the number of units that were planned to end at a station. This cost is incurred per train unit of deviation.

Table 4.2: Coefficients in the objective function.

Parameter	Cost Component	Coefficient
$c^{\text{co}}_{t,p}$	Seat Shortage	0.2
	Mileage	0.1
$c^{\text{ch}}_{c,q}$	New Shunting	1000
	Changed Shunting	100
	Canceled Shunting	50
$c^{\text{id}}_{s,r}$	Ending Inventory Deviation	10000
c^{de}_{t}	Delay	10
c^{ft}	Flexible Turning	1000

Next to these traditional cost objectives, we include two new cost components for the PDRP. To include the impact of delays on passengers, *Delay* gives the cost per minute of delay for a trip and per passenger that is expected to be on a trip. This ensures that a delay of similar size is preferred on a trip that carries few passengers over a trip that carries many passengers. Moreover, we penalize the use of flexible turning, as changes to the turning pattern of a station are likely to lead to platform changes at a station and may be disruptive for crew schedules as well. Here, *Flexible Turning* gives the cost per used flexible turning, i.e., per turning that is changed compared to the original turning pattern.

To make the found objective values comparable for the different rolling stock instances, we will compute the cost of a circulation over the complete day in all our experiments. In particular, this implies that we also include the costs of those trips and transitions which have already become fixed due to falling before the moment at which we allow changes to the rolling stock circulation. Moreover, to ensure that no large improvements can be made to the original circulations, these original circulations have been computed with the Composition Model, up to an optimality gap of 1%, with a similar cost function as the one considered here. In addition, experiments have shown that flexible turning does not lead to any improvements over these original circulations when considering an undisturbed setting without any delays.

4.7.3 Comparing the Proposed Models

In this section, we compare the performance of the two proposed models. The results of the three solution methods that have been proposed for these models are presented in Table 4.3, where *Delay Composition Model* refers to the Delay Composition Model solved by a MIP solver, *Delay Path B&P* refers to the Delay Path Model solved by Branch-and-Price and *Delay Path Root Node* refers to the Delay Path Model solved with a root node heuristic. All results have been obtained on a computer with an Intel Xeon Gold 6130 @ 2.1GHz processor and 96GB of internal memory. Moreover, the CPLEX 12.9.0 solver has been used to solve the MIP model in the Delay Composition Model and the LP models in the Delay Path Model. Furthermore, a time limit of 15 minutes has been imposed for all solution methods to resemble the limited time that is available in rolling stock rescheduling.

Table 4.3 shows that both the *Delay Composition Model* and *Delay Path Root Node* methods are able to consistently find solutions within the set time limit and are able to complete execution on all 150 instances. In contrast, *Delay Path B&P* uses significantly more time for some of the instances and is thus able to find an optimal solution for only 126 out of the 150 instances. This effect is especially apparent for the Regional train instances, where only 51 out of 75 instances can be solved to optimality by *Delay Path B&P*. This weaker performance of *Delay Path B&P* on these instances seems to be caused by the large number of flexible turning opportunities for the Regional trains, increasing the number of nodes that have to be explored in the Branch-and-Bound tree.

Another interesting difference between the *Delay Composition Model* and *Delay Path B&P* is the increase of running time for larger instances. While the average running time of the *Delay Composition Model* increases only moderately when going

Table 4.3: Comparison of the three proposed solution methods. *Time* denotes the computation time in seconds, *Solved* denotes the number of instances for which the solution method completed before the time limit and *Opt. Gap* denotes the obtained optimality gap. The entries for *Time* and *Opt. Gap* are averaged over the 25 instances that have been run for each scenario, including the instances which were not considered solved.

	Delay Composition Model			Delay Path B&P			Delay Path Root Node		
	Time	Solved	Opt. Gap	Time	Solved	Opt. Gap	Time	Solved	Opt. Gap
IC_L	25	25	0.0%	71	25	0.0%	46	25	0.0%
IC_M	20	25	0.0%	31	25	0.0%	21	25	0.1%
IC_S	13	25	0.0%	8	25	0.0%	5	25	0.5%
RE_L	55	25	0.0%	706	9	2.8%	180	25	0.2%
RE_M	32	25	0.0%	399	17	6.1%	55	25	0.3%
RE_S	27	25	0.0%	14	25	0.0%	5	25	0.0%

from the S to the L instances, the running time of *Delay Path B&P* increases more quickly. This behavior seems to be caused by the ability of the CPLEX solver to reduce the problem considerably by means of presolving, eliminating many of the delay variables for trips that cannot incur a delay in the optimal solution. In contrast, the LP models solved by *Delay Path B&P* grow rapidly for larger problem instances.

When comparing the models based on the above observations, the *Delay Composition Model* generally performs best when comparing it to the other exact method *Delay Path B&P*. While it performs comparably to *Delay Path B&P* for the smaller instances, it scales better towards the larger instances. Moreover, we find that *Delay Path Root Node* is able to find solutions that are close to optimality in solution times that are lower than those for *Delay Path B&P*. However, the running times are longer than those of the exact *Delay Composition Model* for larger sized instances. Based on these results, we will use the *Delay Composition Model* throughout the remainder of this section.

4.7.4 Impact on Delays and Original Circulation

In this section, we look at the extent to which solving the PDRP reduces the passenger delays and at the extent to which it alters the original circulation. We do so by comparing the outcome of the PDRP to a baseline scenario in which no changes are made to the original rolling stock schedule. In the baseline scenario, we thus operate the original circulation and leave the original assignment of individual train units to the trips unaltered. The results of our experiments are given in Table 4.4 for the case where we consider delays, but where we do not allow flexible turning. Table 4.5 gives the results when we include flexible turning as well. *Baseline* refers here to the baseline approach, while *Optimal Solution* refers to the optimal solution to the PDRP as found by the *Delay Composition Model*. Note that the original circulation is the same for respectively all IC and RE instances, implying that only the delay costs differ among the different instances when looking at the results for the baseline.

Lastly, Tables 4.6 and 4.7 give a breakdown of the obtained objective values into the different cost components for respectively the IC and RE scenarios.

Table 4.4: Results without flexible turning. All entries in the table are averaged over the 25 instances that have been run for each scenario. *Objective* denotes the average objective value and *Delay* the average delay cost. The shown percentages indicate the change compared to the baseline scenario.

	Baseline		Optimal Solution			
	Objective	Delay	Objective		Delay	
IC_L	395510	278158	395510	(0.0%)	278158	(0.0%)
IC_M	319280	201929	318445	(−0.3%)	201094	(−0.4%)
IC_S	323889	206538	321136	(−0.9%)	203784	(−1.3%)
RE_L	192568	161472	192439	(−0.1%)	161432	(0.0%)
RE_M	118223	87127	116515	(−1.4%)	85482	(−1.9%)
RE_S	139434	108338	134093	(−3.8%)	102998	(−4.9%)

Table 4.5: Results with flexible turning. All entries in the table are averaged over the 25 instances that have been run for each scenario. *Objective* denotes the average objective value, *Delay* the average delay cost and *Turnings* the average number of flexible turnings. The shown percentages indicate the change compared to the baseline scenario.

	Baseline		Optimal Solution				
	Objective	Delay	Objective		Delay		Turnings
IC_L	395510	278158	370755	(−6.3%)	253244	(−9.0%)	0.16
IC_M	319280	201929	290938	(−8.9%)	172361	(−14.6%)	0.48
IC_S	323889	206538	315585	(−2.6%)	197831	(−4.2%)	0.08
RE_L	192568	161472	182390	(−5.3%)	150156	(−7.0%)	1.04
RE_M	118223	87127	104519	(−11.6%)	72680	(−16.6%)	0.68
RE_S	139434	108338	133190	(−4.5%)	101935	(−5.9%)	0.16

The results in Tables 4.4 and 4.5 show that solving the PDRP substantially reduces the delay costs, with average delay reductions of up to 17% when including flexible turning and average delay reductions of up to 5% when excluding flexible turning. However, we do see that the delay reduction differs per scenario. The delay reduction, for example, tends to be larger for the Regional (RE) instances than for the Intercity (IC) instances. Moreover, when including flexible turning, the delay reduction tends to be smaller for the Evening Peak (S) instances than for the other instances.

The results also show that flexible turning plays a central role in delay reduction. For all scenarios the delay reduction is strictly, and often significantly, larger when including flexible turning. Moreover, the use of flexible turning results in delay

Table 4.6: Breakdown of the objective value for the IC scenarios. Here, *Baseline* refers to the baseline, *Fixed* to the situation without flexible turning and *Flex* to the situation with flexible turning. The rows respectively give the delay costs, the seat shortage costs, the mileage costs, the flexible turning costs, the costs of canceling, changing or adding shunting movements and the costs of end inventory deviations.

	IC_L			IC_M			IC_S		
	Baseline	Fixed	Flex	Baseline	Fixed	Flex	Baseline	Fixed	Flex
Delay	278158	278158	253244	201929	201094	172361	206538	203784	197831
Seat Short.	85852	85852	85852	85852	85852	86378	85852	85852	86090
Mileage	31500	31500	31500	31500	31500	31507	31500	31500	31502
Turning	0	0	160	0	0	480	0	0	80
Shunting	0	0	0	0	0	212	0	0	82
Deviation	0	0	0	0	0	0	0	0	0

Table 4.7: Breakdown of the objective value for the RE scenarios. Here, *Baseline* refers to the baseline, *Fixed* to the situation without flexible turning and *Flex* to the situation with flexible turning. The rows respectively give the delay costs, the seat shortage costs, the mileage costs, the flexible turning costs, the costs of canceling, changing or adding shunting movements and the costs of end inventory deviations.

	RE_L			RE_M			RE_S		
	Baseline	Fixed	Flex	Baseline	Fixed	Flex	Baseline	Fixed	Flex
Delay	161472	161432	150156	87127	85482	72680	108338	102998	101935
Seat Short.	8984	8848	8974	8984	8881	9004	8984	8974	8974
Mileage	22112	22158	22175	22112	22153	22155	22112	22121	22121
Turning	0	0	1040	0	0	680	0	0	160
Shunting	0	0	44	0	0	0	0	0	0
Deviation	0	0	0	0	0	0	0	0	0

reduction for each scenario, while there are scenarios in which no delay reduction is achieved without flexible turning. This large impact of flexible turning can be explained by the relatively frequent occurrence of flexible turning opportunities, as trains often reach a terminal station that allows for flexible turning before all delay is absorbed. On the other hand, delay reduction can only be achieved without flexible turning if train units are coupled with a delay at a transition, which turns out to be rare in the baseline scenario.

The results in Tables 4.6 and 4.7 show that delay reduction does generally lead to an increase, albeit often a relatively small one, in other cost components. For example, the inclusion of flexible turning leads to an increase in the seat shortage costs for the IC_M and IC_S scenarios. Similarly, we see that for these two scenarios the inclusion of flexible turning also leads to changes in the shunting plans at some of the stations, leading to larger shunting costs. Interestingly, we also see that we are sometimes able to reduce the costs without increasing any other objectives for the case without flexible turning. In such cases, delay propagation due to short shunting can be prevented by only altering the duties of the individual train units, which were assumed fixed in the baseline.

Another interesting observation from Table 4.5 is that the average number of turnings that are changed is low, where the average number of changed turnings is between 0 and about 1 for all scenarios. This indicates that delay reductions can often be achieved by breaking up only a few existing transitions at terminal stations. This is favorable, as it implies that the effect of flexible turning on the overall shunting and crew plans that are made in later rescheduling phases is limited. Combining this with the observations in Tables 4.6 and 4.7, we see that flexible turning can lead to substantial delay reductions while the impact on the circulation is limited to only changing a few compositions and composition changes. In particular, we find that we can always reschedule the circulation in such a way that even with flexible turning the ending inventories remain unaltered.

Trade-off Between Delay and Circulation Costs

While the above results show that solving the PDRP can lead to substantial delay reductions, they also show that this often leads to increases in other cost components. In this section, we investigate this trade-off between delay costs and circulation costs. We do so by computing trade-off curves between these two cost components, by solving the PDRP for different objective functions that place a different weight on the delay costs and circulation costs. In particular, we consider the objectives

$$\min \lambda \Bigg(\sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c^{\mathrm{co}}_{t,p} X_{t,p} + \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} c^{\mathrm{ch}}_{c,q} Z_{c,q} + \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c^{\mathrm{id}}_{s,r} |I^{\infty}_{s,r} - \iota^{\infty}_{s,r}|$$

$$+ \sum_{t\in\mathcal{T}^f_i} \sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{UN}_{p,t}} c^{\mathrm{ft}} Q_{t,p,p'} + \sum_{t\in\mathcal{T}^f_o} \sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{CO}_{p,t}} c^{\mathrm{ft}} W_{t,p,p'} \Bigg)$$

$$+ (1-\lambda)\left(\sum_{t\in\mathcal{T}} c_t^{\mathrm{de}} Y_t\right)$$

for $\lambda \in \{\frac{1}{20}, \dots, \frac{19}{20}\}$. The corresponding trade-off curves are given in Figure 4.6 for the IC scenarios. Note that each trade-off curve is taken as the average trade-off curve over 25 instances for that scenario.

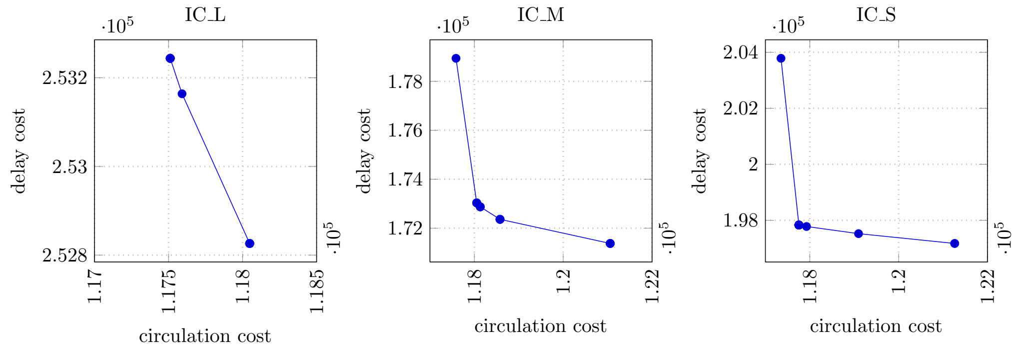


Figure 4.6: Trade-off curves for the IC scenarios.

These trade-off curves show that there is indeed a trade-off between circulation costs and delay costs. In particular, we see for all three scenarios that reductions to the delay costs can be achieved against an increase in circulation costs. Moreover, we see that substantial delay reductions can sometimes already be achieved for only relatively small increases in the circulation cost. This is, for example, the case for the IC_M and IC_S scenarios, where the left part of the trade-off curve is relatively steep and gives multiple Pareto-optimal points which allow for significant delay reductions against relatively modest increases in circulation cost. Alternatively, we see that a higher delay reduction can be achieved for these scenarios for a far larger increase in circulation costs.

A second observation that can be made from these trade-off curves is that the number of breakpoints, i.e., the points representing possible trade-offs between delay costs and circulation costs, in all graphs is limited. In particular, we see that there are only three breakpoints for the IC_L scenario and five for the IC_M and IC_S scenarios. This implies that for many of the considered cost functions, the same solutions are obtained for the instances. This result seems to be caused by the effect of flexible turning, where flexible turning is the main cause of delay reductions. As there are generally only a few good flexible turning options, this implies that many of the obtained solutions are similar as well.

Combining the above results, we see that it might pay off for decision makers to explore the actual trade-off curves. In particular, we see that extreme solutions, which strongly focus on one of the two cost components, can often be improved considerably without large increases in the cost component on which the focus is. At the same time, the limited number of breakpoints also shows that the results are relatively robust against the exact cost parameters that are chosen. Hence, it seems that delay reductions can generally be achieved regardless of the exact chosen cost parameters.

Instance Level Performance

While the results in Tables 4.4 to 4.7 show the average performance of rescheduling, one should bear in mind that the delays can only be reduced for some of the instances. To take a closer look at the performance of the methods on an instance level, Table 4.8 shows the percentage of instances for which the delay can be reduced when including flexible turning. Moreover, this table shows the average delay reduction for those instances in which delay reduction is achieved.

Table 4.8: Results at the instance level for the found optimal solutions. The columns indicate the percentage of instances where the delay cost is reduced (*Reduced*) and the average percentage of reduction in the delay costs for the instances where the delay cost is reduced (*Reduction*).

	Intercity		Regional	
	Reduced	Reduction	Reduced	Reduction
Morning Peak (L)	8%	39%	44%	12%
Between Peak (M)	28%	30%	36%	29%
Evening Peak (S)	8%	44%	16%	29%

The results in Table 4.8 show that delay reduction is only possible for a relatively small subset of all instances, but that the delay reduction for these instances is generally large. These results confirm our intuition, as there are only a limited number of instances for which inventory propagation occurs and for which it is thus possible to reduce the delay propagation by preventing that a delayed train unit is coupled. Moreover, not every terminal station offers good flexible turning opportunities, as some terminal stations only serve a few incoming and outgoing trains per hour. On the other hand, if a flexible turning or changed shunting movement can be found, the delay is often significantly reduced or passed to trips with fewer passengers.

Table 4.9 gives further insight into the relation between the use of flexible turning and delay reduction. In particular, it shows for how many of the instances where delay is reduced, flexible turning is used as well. The results in Table 4.9 show that flexible turning is indeed used for many of the instances where delay can be reduced. This is in line with the results in Tables 4.4 and 4.5, which showed that flexible turning plays a central role in the reduction of delays. Furthermore, it is interesting to note that flexible turning plays especially a large role for the instances which occur earlier during the day. This can be explained by the fact that the number of terminal stations that are passed by a delayed train unit will generally be lower when the delay occurs later in the day.

Table 4.9: The percentage of instances for which flexible turning occurs over all the instances for which delay can be reduced.

	Intercity	Regional
Morning Peak (L)	100%	100%
Between Peak (M)	86%	89%
Evening Peak (S)	50%	50%

4.7.5 The Impact of Flexible Turning Opportunities

An important choice in the PDRP is on which flexible turning opportunities to include in the model. This is determined by the sets $\mathcal{S}_f$, $\mathcal{T}_i^f$ and $\mathcal{T}_o^f$ that define at which stations and at which trips flexible turning can occur. Including more flexible turning possibilities increases the potential of finding rescheduling actions that decrease the delays, but also complicates finding new shunting plans in a later step of the rescheduling process. Moreover, it increases the computation time needed to find an optimal solution to the PDRP. To investigate the impact of the number of flexible turning possibilities on the found results, we vary both the duration during which flexible turning is allowed and the set of locations at which flexible turning is allowed.

The results of our experiments are shown in Figures 4.7 and 4.8. In Figure 4.7 we investigate the effect of varying the duration during which flexible turning is possible. Note that we keep the assumption that flexible turning is only possible at the two terminal stations along the railway line that is affected by the delay. In Figure 4.8 we investigate the effect of changing the set of stations at which flexible turning can occur. In particular, we allow for flexible turning at the ending stations of the affected line (*Ending*), at all stations that are reached by delayed train units (*Reachable*) and at all station in the timetable (*All)*. Moreover, we compare these approaches to a baseline of no flexible turning (*No*). Note that we keep the assumption of a duration of four hours for flexible turning in Figure 4.8.

The results in Figure 4.7 show that, as expected from the increase in flexible turning possibilities, the number of used flexible turnings increases when the duration of flexible turning rises. Interesting to note is that no flexible turning is used for a duration of flexible turning below an hour. This is caused by the fact that it takes some time before a train arrives at a terminal station where flexible turning is available. Moreover, we see in all scenarios that the increase in the use of flexible turning levels off for large durations, as delays have mostly been absorbed at the moment that these turnings occur.

Furthermore, we observe that the delay cost decreases significantly when the flexible turning duration rises. This shows again the role that flexible turning plays in reducing the delays. However, we also see that the computation time steadily rises when the duration increases. In particular, the rate of increase in computation time seems to pick up for larger flexible turning durations. This indicates that one should find a balance between the allowed turning duration and the computational

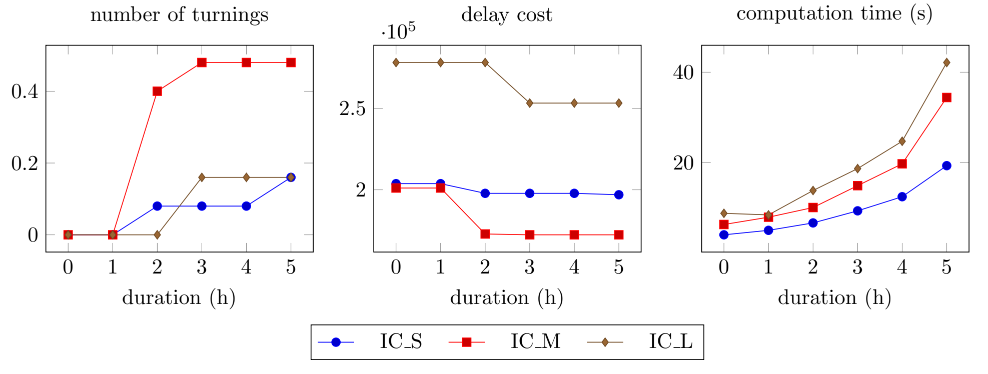


Figure 4.7: Effect of changing the turning duration on the number of flexible turnings, on the delay cost and on the computation time. All results are averaged over 25 instances.

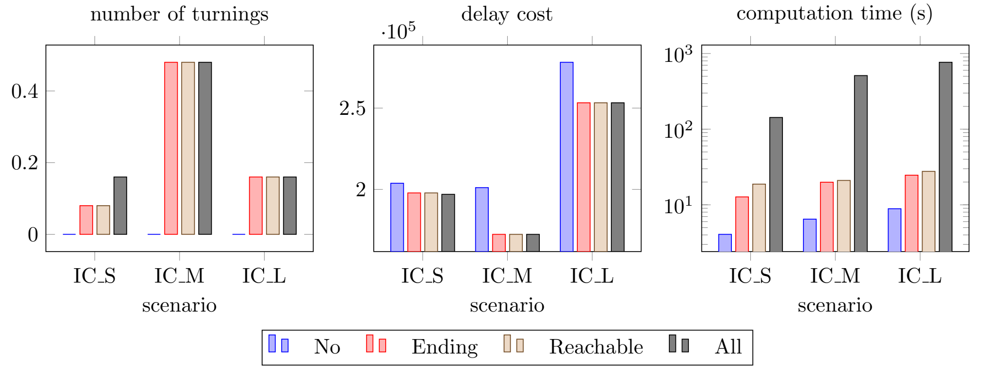


Figure 4.8: Effect of changing the turning locations on the number of flexible turnings, on the delay cost and on the computation time. All results are averaged over 25 instances.

time that is needed to solve the problems.

The results in Figure 4.8 show that the found solutions do not strongly depend on the set of locations where flexible turning is allowed. In particular, we see that adding more flexible turning locations compared to only allowing it at the ending stations of the affected line does not lead to lower delay costs for the IC_M and IC_L scenarios, while only a small improvement is possible for the IC_S scenario. Similarly, we see that only in the IC_S scenario additional flexible turning is used in the setting where flexible turning is allowed at all stations compared to the setting with flexible turning at the ending stations of the affected line. Interestingly, we see that allowing flexible turning at all stations that are reached by delayed train units does not yield an improvement compared to flexible turning at the ending stations of the affected line.

When looking at the obtained computation times, we see that the computation time rises sharply when expanding the set of flexible turning opportunities. In particular, we see a small increase when going from flexible turning at the ending stations of the affected line to flexible turning at the stations that are reached by delayed train units and a large increase in computation time when going to flexible turning at all stations. Moreover, the computation time often reaches the time limit of 15 minutes for especially the IC_L scenario when allowing flexible turning at all stations. This is in line with the previous results of Nielsen (2011), who shows that allowing flexible turning for too many stations can quickly make rolling stock rescheduling models very difficult to solve. Moreover, taking these results together with those on the potential delay reduction, we see that only allowing flexible turning at the ending stations of the affected line gives a good trade-off between delay reduction and computation time.

4.8 Conclusions

In this chapter, we introduced the Passenger Delay Reduction Problem (PDRP). This problem is concerned with finding a rolling stock circulation that minimizes the passenger delay as a result of some initial delays, while at the same time considering objectives on passenger comfort and operational efficiency. Moreover, it allows to change the turning pattern at a set of predefined stations in order to reduce the passenger delays.

We have introduced two models for solving the PDRP: the Delay Composition Model and the Delay Path Model. These are based on two commonly used models for the traditional Rolling Stock Rescheduling Problem. We show that for practical instances the Delay Composition Model performs best in terms of finding high-quality solutions quickly. This model was able to find optimal solutions for all considered instances within a time frame of 15 minutes. In contrast, solving the full Branch-and-Price model for the Delay Path Model turns out to be too expensive for some instances, especially for those that include many flexible turning opportunities.

Computations on practical instances of NS reveal that the impact of solving the PDRP on delay propagation is substantial, even when combining the costs of delays with traditional objectives for rolling stock rescheduling. In addition, we show that such reductions in terms of total delay can only be found for a limited subset of the considered instances. Moreover, an interesting observation from our work is that not much improvement is possible without the use of flexible turning.

Our work provides a first step toward considering the interrelation that exists between the timetable and the rolling stock circulation when rescheduling for delays. An interesting direction for future research is to further integrate these two rescheduling stages. In particular, the problem could benefit from more accurately modeling the delay absorption on trips and at transitions, by considering the delay propagation caused by headway constraints. Moreover, further research may focus on finding efficient (heuristic) solution methods to solve the problem dynamically when new information comes in. These steps would bring the approach closer to the problem that is faced daily by rolling stock dispatchers.

Appendix

4.A Delay Composition Model

$$
\begin{aligned}
\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} c^{\mathrm{co}}_{t,p} X_{t,p} &+ \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} c^{\mathrm{ch}}_{c,q} Z_{c,q} \\
&+ \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} c^{\mathrm{id}}_{s,r} |I^{\infty}_{s,r} - \iota^{\infty}_{s,r}| + \sum_{t\in\mathcal{T}} c^{\mathrm{de}}_t Y_t \\
&+ \sum_{t\in\mathcal{T}^f_i} \sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{UN}_{p,t}} c^{\mathrm{ft}} Q_{t,p,p'} \\
&+ \sum_{t\in\mathcal{T}^f_o} \sum_{p\in\mathcal{P}_t} \sum_{p'\in\mathrm{CO}_{p,t}} c^{\mathrm{ft}} W_{t,p,p'}
\end{aligned}
$$

$$
\begin{aligned}
\text{s.t.} \quad & \sum_{p\in\mathcal{P}_t} X_{t,p} = 1 && \forall t\in\mathcal{T}, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^+(t)}: p_{q,t}=p} Z_{\delta^+(t),q} && \forall t\in\mathcal{T}\setminus\mathcal{T}^f_i, p\in\mathcal{P}_t, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^-(t)}: p'_{q,t}=p} Z_{\delta^-(t),q} && \forall t\in\mathcal{T}\setminus\mathcal{T}^f_o, p\in\mathcal{P}_t, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^+(t)}: p_{q,t}=p} Z_{\delta^+(t),q} + \sum_{p'\in\mathrm{UN}_{p,t}} Q_{t,p,p'} && \forall t\in\mathcal{T}^f_i, p\in\mathcal{P}_t, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^-(t)}: p'_{q,t}=p} Z_{\delta^-(t),q} + \sum_{p'\in\mathrm{CO}_{p,t}} W_{t,p,p'} && \forall t\in\mathcal{T}^f_o, p\in\mathcal{P}_t, \\
& C_{c,r} = \sum_{q\in\mathcal{Q}_c} \gamma_{q,r} Z_{c,q} && \forall c\in\mathcal{C}\setminus\widetilde{\mathcal{C}}, r\in\mathcal{R}, \\
& U_{c,r} = \sum_{q\in\mathcal{Q}_c} \upsilon_{q,r} Z_{c,q} && \forall c\in\mathcal{C}\setminus\widetilde{\mathcal{C}}, r\in\mathcal{R},
\end{aligned}
$$

$$C_{c,r} = \sum_{q \in \mathcal{Q}_c} \gamma_{q,r} Z_{c,q} + \sum_{t \in \mathcal{T}_c^+} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} (\mu_p^r - \mu_{p'}^r) W_{t,p,p'} \qquad \forall c \in \widetilde{\mathcal{C}}, r \in \mathcal{R},$$

$$U_{c,r} = \sum_{q \in \mathcal{Q}_c} \upsilon_{q,r} Z_{c,q} + \sum_{t \in \mathcal{T}_c^-} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{UN}_{p,t}} (\mu_p^r - \mu_{p'}^r) Q_{t,p,p'} \qquad \forall c \in \widetilde{\mathcal{C}}, r \in \mathcal{R},$$

$$Y_t = d_t \qquad \forall t \in \mathcal{T}_{\text{init}},$$

$$Y_t \geq Y_{t'} - \Delta_{\delta^-(t)} - \Delta_t \qquad \forall t \in \mathcal{T} \setminus \mathcal{T}_o^f, t' \in \mathcal{T}_{\delta^-(t)}^-,$$

$$Y_t \geq Y_{t'} - \Delta_{\delta^-(t)} - \Delta_t - d_u \left(\sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} W_{t,p,p'} \right) \qquad \forall t \in \mathcal{T}_o^f, t' \in \mathcal{T}_{\delta^-(t)}^-,$$

$$\sum_{d \in \mathcal{D}} d D_{c,d}^i \geq Y_t \qquad \forall c \in \mathcal{C}, t \in \mathcal{T}_c^-,$$

$$\sum_{d \in \mathcal{D}} D_{c,d}^i = 1 \qquad \forall c \in \mathcal{C},$$

$$U_{c,r,d} \leq M_1 D_{c,d}^i \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D},$$

$$\sum_{d \in \mathcal{D}} U_{c,r,d} = U_{c,r} \qquad \forall c \in \mathcal{C}, r \in \mathcal{R},$$

$$Y_t \geq d D_{\delta^-(t),d}^o - \Delta_t \qquad \forall t \in \mathcal{T}, d \in \mathcal{D},$$

$$C_{c,r,d} \leq M_2 D_{c,d}^o \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D},$$

$$\sum_{d \in \mathcal{D}} C_{c,r,d} = C_{c,r} \qquad \forall c \in \mathcal{C}, r \in \mathcal{R},$$

$$\iota_{s(c),r}^0 - \sum_{\substack{c' \in \mathcal{C}: \\ s(c')=s(c)}} \sum_{\substack{d' \in \mathcal{D}: \\ \tau^+(c')+d' \leq \tau^+(c)+d}} C_{c',r,d'} + \sum_{\substack{c' \in \mathcal{C}: \\ s(c')=s(c)}} \sum_{\substack{d' \in \mathcal{D}: \\ \tau^-(c')+\left(d'-\Delta_{s(c')}\right)^+ \leq \tau^+(c)+d}} U_{c',r,d'} \geq 0 \qquad \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D},$$

$$I_{s,r}^\infty = \iota_{s,r}^0 - \sum_{c \in \mathcal{C}: s(c)=s} C_{c,r} + \sum_{c \in \mathcal{C}: s(c)=s} U_{c,r} \qquad \forall s \in \mathcal{S} \setminus \mathcal{S}_f, r \in \mathcal{R},$$

$$I_{s,r}^\infty = \iota_{s,r}^0 - \sum_{c \in \mathcal{C}: s(c)=s} C_{c,r} + \sum_{c \in \mathcal{C}: s(c)=s} U_{c,r} + \sum_{c \in \widetilde{\mathcal{C}}: s(c)=s} \sum_{t \in \mathcal{T}_c^-} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{UN}_{p,t}} \mu_{p'}^r Q_{t,p,p'} - \sum_{c \in \widetilde{\mathcal{C}}: s(c)=s} \sum_{t \in \mathcal{T}_c^+} \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} \mu_{p'}^r W_{t,p,p'} \qquad \forall s \in \mathcal{S}_f, r \in \mathcal{R},$$

$$
\begin{aligned}
& F^i_{t,d,p} \leq D^i_{\delta^+(t),d} && \forall t \in \mathcal{T}^f_i, p \in \mathcal{P}, d \in \mathcal{D}, \\
& \sum_{d \in \mathcal{D}} F^i_{t,d,p} = \sum_{p' : p \in \mathrm{UN}_{p',t}} Q_{t,p',p} && \forall t \in \mathcal{T}^f_i, p \in \mathcal{P}, \\
& F^o_{t,d,p} \leq D^o_{\delta^-(t),d} && \forall t \in \mathcal{T}^f_o, p \in \mathcal{P}, d \in \mathcal{D}, \\
& \sum_{d \in \mathcal{D}} F^o_{t,d,p} = \sum_{p' : p \in \mathrm{CO}_{p',t}} W_{t,p',p} && \forall t \in \mathcal{T}^f_o, p \in \mathcal{P}, \\
& P_{c,p,d} = \sum_{c' \in \widetilde{\mathcal{C}} : s(c') = s(c)} \sum_{\substack{d' \in \mathcal{D}: \\ \tau_a(c') + \left(d' - \Delta_{c'}\right)^+ \leq \tau_d(c) + d}} \sum_{t \in \mathcal{T}^-_{c'}} F^i_{t,d',p} && \\
& \qquad - \sum_{c' \in \widetilde{\mathcal{C}} : s(c') = s(c)} \sum_{\substack{d' \in \mathcal{D}: \\ \tau_d(c') + d' \leq \tau_d(c) + d}} \sum_{t \in \mathcal{T}^+_{c'}} F^o_{t,d',p} && \forall c \in \widetilde{\mathcal{C}}, p \in \mathcal{P}, d \in \mathcal{D}, \\
& X_{t,p} \in \{0,1\} && \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \\
& Z_{c,q} \in \{0,1\} && \forall c \in \mathcal{C}, q \in \mathcal{Q}_c, \\
& I^\infty_{s,r} \in \mathbb{Z}_+ && \forall s \in \mathcal{S}, r \in \mathcal{R}, \\
& C_{c,r}, U_{c,r} \in \mathbb{Z}_+ && \forall c \in \mathcal{C}, r \in \mathcal{R}, \\
& Y_t \in \mathcal{D} && \forall t \in \mathcal{T}, \\
& C_{c,r,d}, U_{c,r,d} \in \mathbb{Z}_+ && \forall c \in \mathcal{C}, r \in \mathcal{R}, d \in \mathcal{D}, \\
& D^i_{c,d}, D^o_{c,d} \in \{0,1\} && \forall c \in \mathcal{C}, d \in \mathcal{D}, \\
& Q_{t,p,p'} \in \{0,1\} && \forall t \in \mathcal{T}^f_i, p \in \mathcal{P}_t, p' \in \mathrm{UN}_{p,t}, \\
& W_{t,p,p'} \in \{0,1\} && \forall t \in \mathcal{T}^f_o, p \in \mathcal{P}_t, p' \in \mathrm{CO}_{p,t}, \\
& F^i_{t,d,p} \in \{0,1\} && \forall t \in \mathcal{T}^f_i, d \in \mathcal{D}, p \in \mathcal{P}, \\
& F^o_{t,d,p} \in \{0,1\} && \forall t \in \mathcal{T}^f_o, d \in \mathcal{D}, p \in \mathcal{P}, \\
& P_{c,p,d} \in \mathbb{Z}_+ && c \in \widetilde{\mathcal{C}}, p \in \mathcal{P}, d \in \mathcal{D}.
\end{aligned}
$$

4.B Delay Path Model

$$
\begin{aligned}
\min & \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}_t} c^{\mathrm{co}}_{t,p}X_{t,p} + \sum_{c\in\mathcal{C}}\sum_{q\in\mathcal{Q}_c} c^{\mathrm{ch}}_{c,q}Z_{c,q} \\
& + \sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}} c^{\mathrm{id}}_{s,r}|I^{\infty}_{s,r} - \iota^{\infty}_{s,r}| + \sum_{t\in\mathcal{T}} c^{\mathrm{de}}_t Y_t \\
& + \sum_{t\in\mathcal{T}^f_i}\sum_{p\in\mathcal{P}_t}\sum_{p'\in\mathrm{UN}_{p,t}} c^{\mathrm{ft}}Q_{t,p,p'} \\
& + \sum_{t\in\mathcal{T}^f_o}\sum_{p\in\mathcal{P}_t}\sum_{p'\in\mathrm{CO}_{p,t}} c^{\mathrm{ft}}W_{t,p,p'} \\
\text{s.t.} & \sum_{p\in\mathcal{P}_t} X_{t,p} = 1 && \forall t\in\mathcal{T}, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta+(t)}:p_{q,t}=p} Z_{\delta+(t),q} && \forall t\in\mathcal{T}\setminus\mathcal{T}^f_i, p\in\mathcal{P}_t, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta-(t)}:p'_{q,t}=p} Z_{\delta-(t),q} && \forall t\in\mathcal{T}\setminus\mathcal{T}^f_o, p\in\mathcal{P}_t, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta+(t)}:p_{q,t}=p} Z_{\delta+(t),q} + \sum_{p'\in\mathrm{UN}_{p,t}} Q_{t,p,p'} && \forall t\in\mathcal{T}^f_i, p\in\mathcal{P}_t, \\
& X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta-(t)}:p'_{q,t}=p} Z_{\delta-(t),q} + \sum_{p'\in\mathrm{CO}_{p,t}} W_{t,p,p'} && \forall t\in\mathcal{T}^f_o, p\in\mathcal{P}_t, \\
& \sum_{\pi\in\Pi_r} \omega^t_\pi L_\pi = \sum_{p\in\mathcal{P}} \mu^r_p X_{t,p} && \forall r\in\mathcal{R}, t\in\mathcal{T}, \\
& \sum_{\pi\in\Pi_r:b(\pi)=s} L_\pi = \iota^0_{s,r} && \forall r\in\mathcal{R}, s\in\mathcal{S}, \\
& \sum_{\pi\in\Pi_r:e(\pi)=s} L_\pi = I^{\infty}_{s,r} && \forall r\in\mathcal{R}, s\in\mathcal{S}, \\
& Y_t = d_t && \forall t\in\mathcal{T}_{\mathrm{init}}, \\
& Y_t \geq Y_{t'} - \Delta_{\delta^-(t)} - \Delta_t && \forall t\in\mathcal{T}\setminus\mathcal{T}^f_o, t'\in\mathcal{T}^-_{\delta^-(t)}, \\
& Y_t \geq Y_{t'} - \Delta_{\delta^-(t)} - \Delta_t \\
& \qquad - d_u\left(\sum_{p\in\mathcal{P}_t}\sum_{p'\in\mathrm{CO}_{p,t}} W_{t,p,p'}\right) && \forall t\in\mathcal{T}^f_o, t'\in\mathcal{T}^-_{\delta^-(t)}, \\
& \sum_{\pi\in\Pi} \zeta^{t',t}_\pi L_\pi \leq M_3 S_{t',t} && \forall t\in\mathcal{T}, t'\in\mathrm{SS}_t, \\
& Y_t \geq Y_{t'} - d_u(1-S_{t',t}) + \tau^-(\delta^+(t')) - \tau^+(\delta^-(t)) \\
& \qquad - \Delta_{s(\delta^-(t))} - \Delta_t && \forall t\in\mathcal{T}, t'\in\mathrm{SS}_t,
\end{aligned}
$$

$$
\begin{aligned}
P_{c,p} = & \sum_{\substack{c' \in \widetilde{\mathcal{C}}: s(c')=s(c),\\ \tau_a(c') \le \tau_d(c)}} \sum_{t \in \mathcal{T}^-_{c'}} \sum_{p': p \in \mathrm{UN}_{p',t}} Q_{t,p',p} \\
& - \sum_{\substack{c' \in \widetilde{\mathcal{C}}: s(c')=s(c),\\ \tau_d(c') \le \tau_d(c)}} \sum_{t \in \mathcal{T}^+_{c'}} \sum_{p': p \in \mathrm{CO}_{p',t}} W_{t,p',p} \\
& + \sum_{\substack{c' \in \widetilde{\mathcal{C}}: s(c)=s(c')\\ \tau_d(c') \le \tau_d(c)}} \sum_{t \in \mathcal{T}^+_{c'}} \sum_{\substack{t' \in \mathrm{ST_t}:\\ \tau_a(\delta^+(t')) > \tau_d(c)}} B_{t',t,p} && \forall c \in \widetilde{\mathcal{C}}, p \in \mathcal{P}, \\
& \sum_{\pi \in \Pi_r} \phi^a_{t,\pi} L_\pi = \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{UN}_{p,t}} \mu^r_{p'} Q_{t,p,p'} && \forall t \in \mathcal{T}^f_i, r \in \mathcal{R}, \\
& \sum_{\pi \in \Pi_r} \phi^d_{t,\pi} L_\pi = \sum_{p \in \mathcal{P}_t} \sum_{p' \in \mathrm{CO}_{p,t}} \mu^r_{p'} W_{t,p,p'} && \forall t \in \mathcal{T}^f_o, r \in \mathcal{R}, \\
& \sum_{\pi \in \Pi_r} \xi^{t',t}_\pi L_\pi = \sum_{p \in \mathcal{P}_{t'}} \sum_{p' \in \mathrm{UN}_{p,t'}} \mu^r_{p'} B_{t',t,p'} && \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, r \in \mathcal{R}, \\
& Y_t \ge Y_{t'} - \Delta_t - M_4 \left(1 - \sum_{p \in \mathcal{P}_{t'}} \sum_{p' \in \mathrm{UN}_{p,t'}} B_{t',t,p'} \right) \\
& \quad - \Delta(t,t') && \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, \\
& B_{t',t,p'} \le \sum_{p^* \in \mathcal{P}_{t'}} Q_{t',p^*,p'} && \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, p \in \mathcal{P}_{t'}, p' \in \mathrm{UN}_{p,t'}, \\
& B_{t',t,p'} \le \sum_{p^* \in \mathcal{P}_t} W_{t,p^*,p'} && \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, p \in \mathcal{P}_{t'}, p' \in \mathrm{UN}_{p,t'}, \\
& L_\pi \in \{0,1\} && \forall \pi \in \Pi, \\
& X_{t,p} \in \{0,1\} && \forall t \in \mathcal{T}, p \in \mathcal{P}_t, \\
& Z_{c,q} \in \{0,1\} && \forall c \in \mathcal{C}, q \in \mathcal{Q}_c, \\
& I^\infty_{s,r} \in \mathbb{Z}_+ && \forall s \in \mathcal{S}, r \in \mathcal{R}, \\
& Y_t \in \mathcal{D} && \forall t \in \mathcal{T}, \\
& S_{t',t} \in \{0,1\} && \forall t \in \mathcal{T}, t' \in \mathrm{SS}_t, \\
& Q_{t,p,p'} \in \{0,1\} && \forall t \in \mathcal{T}^f_i, p \in \mathcal{P}_t, p' \in \mathrm{UN}_{p,t}, \\
& W_{t,p,p'} \in \{0,1\} && \forall t \in \mathcal{T}^f_o, p \in \mathcal{P}_t, p' \in \mathrm{CO}_{p,t}, \\
& P_{c,p} \in \mathbb{Z}_+ && \forall c \in \widetilde{\mathcal{C}}, p \in \mathcal{P}, \\
& B_{t',t,p'} \in \{0,1\} && \forall t \in \mathcal{T}^f_o, t' \in \mathrm{ST}_t, p \in \mathcal{P}_{t'}, p' \in \mathrm{UN}_{p,t'}.
\end{aligned}
$$

Chapter 5

Integrated Rolling Stock and Shunting Driver Rescheduling

5.1 Introduction

With passenger numbers rising globally, the operating frequency is increased in many public transportation systems. As infrastructure capacity often does not grow accordingly, this leads to reduced space for rescheduling in case of disruptions and hence to an increasing need for decision support tools to support dispatchers. In this chapter, we look at the rescheduling of railway rolling stock, i.e., the assignment of rolling stock to the trips, and its interaction with the rescheduling of shunting drivers at the stations. This is motivated by the operations of Netherlands Railways (NS), for which it turns out that the availability of shunting drivers is often a bottleneck when rescheduling the rolling stock.

Rolling stock rescheduling is performed after a disruption impacts the timetable, requiring to also reschedule the original assignment of rolling stock to the trips. In rolling stock rescheduling, the main goal is, generally, to meet the passenger demand while minimizing operational costs and staying close to the original rolling stock assignment. To better match passenger demands, train units can be combined into compositions. These compositions can then be altered whenever a train stops at a station in between operating trips. Any train units that are uncoupled from a composition during such composition changes are parked at the shunting yard of a station until they are needed again.

Shunting movements, i.e., physical movements of train units, are thus needed to move train units from and to the shunting yards when changing compositions. At NS, these are executed by so-called shunting drivers. These are drivers that are authorized to move train units to the shunting yards of a station and of whom the duty consists of solely executing such shunting tasks. For the chosen shunting tasks to be feasible, we need to be able to find duties for the shunting drivers that cover these shunting tasks. Moreover, when the shunting tasks are changed due to changes in the rolling stock assignment, also the duties of these shunting drivers have to be rescheduled.

Currently, rolling stock rescheduling and the rescheduling of the shunting drivers at the stations are performed sequentially. This implies that rolling stock rescheduling is performed first, where shunting driver capacity is to some extent taken into account by limiting the set of allowed shunting movements and penalizing changes to the shunting movements compared to the original rolling stock assignment. The new rolling stock circulation, i.e., rolling stock schedule, is then communicated to local dispatchers to check if duties can be found for the shunting drivers which cover all shunting tasks. If no such duties can be found, this is communicated back to the rolling stock dispatcher, who solves the rolling stock rescheduling problem with updated inputs. This process then continues until a rolling stock circulation is found for which feasible shunting driver schedules can be found at all stations.

The main disadvantage of the current approach is that it often leads to multiple iterations between rolling stock rescheduling and shunting driver rescheduling. As a result, shunting movements are generally chosen conservatively, leading to suboptimal rolling stock solutions. For example, a rolling stock dispatcher might decide not to add a train unit to a composition on a busy trip, as he is unsure if such a shunting movement can be executed at the station. Moreover, it might also lead to the cancellation of trips when no feasible solutions can be found within the time that is available for rescheduling.

To overcome these issues, we propose to solve the rolling stock rescheduling problem jointly with the shunting driver rescheduling at the stations. We propose two different solution methods for this integrated problem. First, we propose a Benders decomposition approach in which cuts are generated when a rolling stock circulation does not allow for a feasible allocation of shunting tasks to the shunting drivers. The master problem in this approach is based on the Composition model as proposed by Fioole et al. (2006) for rolling stock scheduling, while our subproblem is a set partitioning formulation for rescheduling the shunting drivers. Second, we consider an arc-based model that can be solved by a commercial Mixed Integer Linear Programming (MILP) solver, which is obtained by using an arc-based formulation for shunting driver rescheduling. We test both methods on instances of NS.

Our contributions in this chapter are threefold. First, we propose the integrated rolling stock and shunting driver rescheduling problem. Second, we propose both a Benders decomposition approach and an arc-based model for solving this problem. Third, we show the value of our methods on instances of NS, where we compare the performance of both solution methods and look at the extent to which they can solve the integrated problem.

The remainder of this chapter is organized as follows. In Section 5.2, we define the problem. In Section 5.3, we give an overview of the literature on rolling stock rescheduling and the rescheduling of shunting drivers. A mathematical formulation for the problem is given in Section 5.4. In Section 5.5, we describe the Benders decomposition approach. In Section 5.6, we describe the arc-based model. We apply both methods to instances of NS in Section 5.7. The chapter is concluded in Section 5.8.

5.2 Problem Definition

Rolling stock rescheduling focuses on assigning rolling stock to the trips in the rescheduled timetable, i.e., the timetable that has been adjusted for the disruption. We assume that a fleet of self-powered train units is available to operate these trips, which implies that no locomotives are needed to pull the train units. A large part of the rolling stock at NS, and many other European railway operators, consists of such train units. These train units can be of different types, where, most importantly, the type determines how many passengers can fit in the train unit.

Train units of compatible types can be combined to form compositions in order to better match the passenger demand. Such a composition is thus a sequence, i.e., an ordered multiset, of train unit types. An example of three different compositions as operated at NS is shown in Figure 5.1. In this chapter, we will consider anonymous train units, which implies that we consider train units of the same type to be interchangeable. We will therefore only look at the type of the train units in the composition and not at which physical train units are in the composition.

The compositions in which train units operate are not fixed throughout the day. Instead, they can be changed at transitions between trips to accommodate the varying

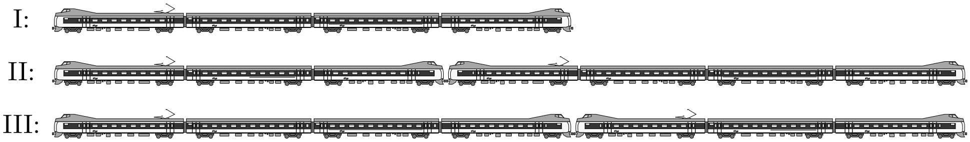


Figure 5.1: Example of three compositions comprised of ICM-III (3 carriages) and ICM-IV (4 carriages) train units. Note that the train units in the last composition are in reversed order compared to the second composition.

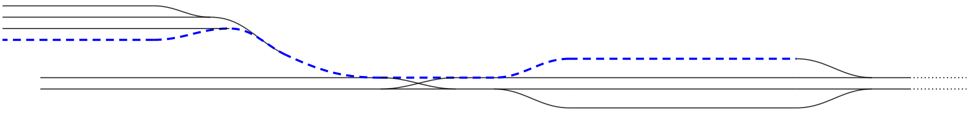


Figure 5.2: Example of a shunting movement, indicated by the blue (dashed) path from a simple station with two platforms (on the right) to a shunting yard with four tracks (on the top left).

passenger numbers that are faced on trips. These transitions define the moments that a train is present at the platform of a station between trips. Changes to the composition of a train can be made through composition changes. These include the coupling of train units, in which train units are added to the composition, and the uncoupling of train units, in which train units are taken out of the composition.

Due to the platform length at stations, it is generally only possible to operate a limited set of compositions for each trip. Moreover, the station layout has an influence on which composition changes can be performed at a transition. For example, when the shunting yard of a station can only be accessed from one side of the station, train units can only be uncoupled from one side of the composition to prevent blocking the continuing train units. Similar restrictions are then in place for the coupling of train units.

As the number of platform tracks at a station is limited, train units that are uncoupled at a transition are often moved to the shunting yard of a station. Train units are then parked at the shunting yard until they are coupled again to an outgoing composition, for which they have to be moved again from the shunting yard to the stations. These movements of train units from and to the shunting yard, or between platform tracks of a station, are referred to as shunting movements. An example of a shunting movement is given in Figure 5.2 in a station area that contains both a station and a shunting yard.

Shunting movements are, generally, executed by shunting drivers. Unlike most other train drivers that arrive at a station, these shunting drivers have a detailed knowledge of the infrastructure at the station and its shunting yard(s). During a day, such shunting drivers execute solely shunting tasks, which implies that every day a dedicated set of shunting drivers is available to execute the shunting movements.

Duties of shunting drivers consist of a set of shunting movements that they execute in sequence. Between such shunting movements, enough time has to be allocated for the shunting driver to walk from the track where one shunting movement ends to the track where the next shunting movement starts. This walking time is limited, i.e., a few minutes, when both tracks are at the same shunting yard or station. A significantly longer time, i.e., in the order of tens of minutes, is needed when a driver needs to walk, or take a taxi, from the shunting yard to the station or vice versa.

Next to duties needing to be feasible in terms of walking time, the duties should also fulfill all labor regulations. In the rescheduling phase, this implies that the planned starting and ending time of a duty should be respected. Moreover, a meal break of the required length should be present for the driver. We will assume that this break needs to occur no later than a certain number of hours from the start of the duty and no earlier than a certain number of hours from the end of the duty.

An example of a duty assignment for shunting drivers is shown in Figure 5.3. In this example, 17 shunting tasks $(m_1, \ldots, m_{17})$ have to be performed and three drivers (d_1, d_2, d_3) are available to operate these tasks. Note how each driver is allocated a duty in which a few shunting movements are executed in sequence and that some time is allocated in between shunting movements to walk from one track to another. Moreover, note that each driver is allocated a break, indicated by br.

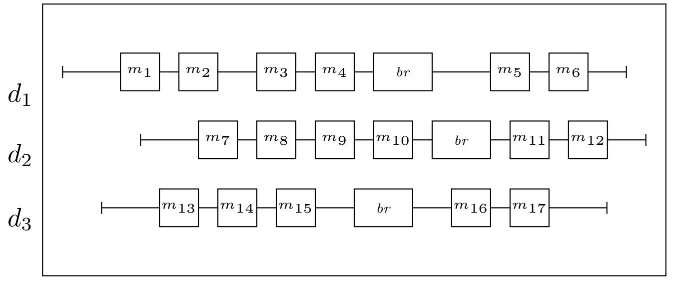


Figure 5.3: Example of a duty assignment for the shunting drivers.

We can now formulate the Rolling Stock and Shunting Driver Rescheduling Problem (RSSDRP). This is the problem of assigning compositions to the trips, in such a way that enough train units are available to form these compositions and such that all restrictions on compositions and composition changes are taken into account. In particular, and as opposed to traditional rolling stock rescheduling, we should assign compositions in such a way that at each station we can find duties for the available shunting drivers that cover all shunting tasks defined by the composition changes. The output of the RSSDRP is then a rolling stock circulation and corresponding duty assignments for the shunting drivers at each station.

The objective of the RSSDRP is to find a balance between meeting the passenger demand and minimizing the operational costs. On the one hand, we would like to offer enough seats on the trains to fulfill the passenger demands for the rescheduled trips. On the other hand, we would like to minimize the deviations from the existing operational plan to minimize the communication needed with crews. Moreover, we would generally like to find shunting driver duties that are attractive for the shunting drivers, e.g., with little walking time between tasks.

5.3 Literature Overview

Rolling stock rescheduling has been covered extensively in the disruption management literature. For an overview of the disruption management process, we refer the reader to Cacchiani et al. (2014), who discuss how disruption management is typically split into a few distinct operational problems and who give common models for each of these problems. Most importantly, we will assume, as in Cacchiani et al. (2014), that the timetable has already been rescheduled before rescheduling the rolling stock and that crew rescheduling for (non-shunting) drivers and guards happens in a later step.

Especially relevant for this chapter is the rolling stock scheduling model of Fioole et al. (2006), which was incorporated into the rescheduling context by Nielsen (2011). This model is based on a multi-commodity flow representation of the rolling stock assignment, which allows the model to be solved by a commercial MILP solver. Moreover, the model is focused on the setting of rolling stock (re)scheduling at NS, as also considered in this chapter. We use this model as part of our formulation for the RSSDRP.

An alternative flow-based model has been proposed by Borndörfer et al. (2016), who introduced a hypergraph model for rolling stock scheduling that can take into account maintenance and regularity constraints. In Chapter 2, we analytically compared this model and the model of Fioole et al. (2006). Path-based formulations have been proposed by, among others, Lusby et al. (2017) and Cacchiani et al. (2010). These approaches are based on finding an individual path, also referred to as duty, for each of the available train units. A comparison of the flow and path-based approaches is made by Haahr et al. (2016) on instances from both NS and DSB S-tog.

Numerous extensions of these rolling stock rescheduling models have been considered to more closely resemble the setting faced by various railway passenger operators. Nielsen et al. (2012) solve the rolling stock rescheduling problem in a rolling horizon framework to include the inherently uncertain nature of the disruption information. Moreover, in Chapter 4 we rescheduled the rolling stock in case of delays, where we tried to find a rolling stock assignment that minimizes the propagation of any primary delays. Alternatively, Wagenaar et al. (2017a) take into account the maintenance appointments that some train units have when rescheduling the rolling stock.

An extension that is particularly relevant for this chapter is by Haahr and Lusby (2017), who integrate rolling stock scheduling with the train unit shunting problem. In the latter problem, it has to be determined where train units are parked at the shunting yard and which incoming train unit at the shunting yard is matched to which outgoing train unit. The motivation for their approach is that many rolling stock assignments for DSB S-tog turned out to be infeasible due to limited shunting yard capacity. Similarly, our work is based on the observation that the number of available shunting drivers often leads to infeasibilities when rescheduling the rolling stock assignment at NS.

Compared to the literature on rolling stock rescheduling, the available literature on shunting driver rescheduling is limited. However, the rescheduling of shunting drivers is closely related to the general crew rescheduling problem as faced by passenger railway operators for (non-shunting) drivers and guards. This problem is discussed by, among others, Huisman (2007), Rezanova and Ryan (2010), Potthoff et al. (2010) and Sato and Fukumura (2011). A common characteristic of these papers is that they use a set covering or set partitioning formulation that is solved through column generation. We use a similar approach for the rescheduling of shunting drivers in our Benders decomposition approach, for which we use a set partitioning formulation that is also solved using column generation.

The only paper, as far as we are aware, that discusses the scheduling of shunting drivers is by van Wezel and Riezebos (2011), who propose a prototype of a planning tool for shunting yards at NS. In this prototype, they reschedule the shunting drivers through a shortest augmenting path approach that minimizes the walking time of the shunting drivers. Note that, unlike our work in this chapter, van Wezel and Riezebos (2011) do not explicitly consider the breaks of shunting drivers in their approach. Moreover, they do not look at the interaction of shunting drivers with the rolling stock circulation and instead look at scheduling the shunting yards for a fixed circulation.

Instead, the paper of Thorlacius et al. (2015) is the only paper, as far as we are aware, that discusses the interaction of the rolling stock assignment and shunting drivers. In this paper, the authors take into account the number of available shunting drivers as part of an integrated rolling stock planning problem of DSB S-tog. In particular, they check that no more shunting movements are performed at a time than there are shunting drivers available at a station. Unlike our approach, they do not consider the actual scheduling of duties for shunting drivers, meaning that they, e,g., do not take into account the required walking time between different shunting movements. Moreover, while we focus on exact solution approaches for the RSSDRP, the solution approach of Thorlacius et al. (2015) is based on a heuristic.

5.4 Mathematical Formulation for the RSSDRP

In this section, we introduce a MILP formulation for the RSSDRP. Before we do so, we introduce all necessary notation, where most of our notation follows that of Nielsen (2011).

Let $\mathcal{T}$ be the set of trips in the rescheduled timetable, which has been adjusted for any disruptions. Moreover, let $\mathcal{C}$ be the set of transitions, where $\delta^-(t) \in \mathcal{C}$ and $\delta^+(t) \in \mathcal{C}$ are the transition preceding and succeeding trip $t \in \mathcal{T}$, respectively. Similarly, let $\mathcal{T}_c^- \subseteq \mathcal{T}$ and $\mathcal{T}_c^+ \subseteq \mathcal{T}$ denote, respectively, the incoming and outgoing trips for transition $c \in \mathcal{C}$. Note that these sets usually contain only a single trip, but can also be empty when all train units start or end at this station. Let $\mathcal{S}$ be the set of stations where these transitions take place, where $s(c) \in \mathcal{S}$ denotes the station at which transition $c \in \mathcal{C}$ occurs. Furthermore, let $\tau^-(c)$ and $\tau^+(c)$ indicate the time at which uncoupled and coupled train units in transition $c \in \mathcal{C}$ arrive at and depart from the shunting yard, respectively.

Let $\mathcal{R}$ be the set of available train unit types. The set $\mathcal{P}$ denotes all compositions that can be formed by train units of these train unit types. Then, let $\mathcal{P}_t \subseteq \mathcal{P}$ denote the set of compositions that can be used on trip $t \in \mathcal{T}$. Moreover, let $\mathcal{Q}$ be the set of possible composition changes that can occur at the transitions. We then denote by $\mathcal{Q}_c \subseteq \mathcal{Q}$ all composition changes that can occur at transition $c \in \mathcal{C}$. For each composition change $q \in \mathcal{Q}$, $p_{q,t} \in \mathcal{P}$ and $p'_{q,t} \in \mathcal{P}$ denote the composition of some incoming trip $t \in \mathcal{T}_c^-$ and outgoing trip $t \in \mathcal{T}_c^+$ in this composition change, respectively.

Composition changes create tasks that need to be executed by the shunting drivers. Let $\mathcal{M}_q$ be the set of shunting tasks that need to be executed for composition change $q \in \mathcal{Q}$, where $\mathcal{M}_q = \emptyset$ if no shunting takes place in this composition change. Note that multiple composition changes for a transition can lead to a similar task for the shunting drivers. That is to say, in general we can find $q_1, q_2 \in \mathcal{Q}_c$ such that $\mathcal{M}_{q_1} \cap \mathcal{M}_{q_2} \neq \emptyset$.

Each station at which uncoupling and coupling may take place has available shunting drivers to execute these shunting tasks. Let $\mathcal{D}$ be the set of available shunting drivers and let $\mathcal{D}_s \subseteq \mathcal{D}$ be the set of shunting drivers available at station $s \in \mathcal{S}$. Note that we assume that each station has its own set of shunting drivers, i.e., $\mathcal{D}_{s_1} \cap \mathcal{D}_{s_2} = \emptyset$ for $s_1 \neq s_2$.

Let $\mathcal{K}_d$ be the set of feasible completions of the duty for shunting driver $d \in \mathcal{D}$. Similar to Huisman (2007), each completion determines the tasks that are executed by a shunting driver during the remainder of his or her duty. We only consider those completions that take into account all labor regulations, i.e., that respect the starting and ending time of the shunting driver's duty and all restrictions on the meal break. Let $\kappa^d_{m,k}$ indicate whether completion $k \in \mathcal{K}_d$ contains task m.

The number of available train units is limited. Let $\iota^0_{s,r}$ be the number of train units of type $r \in \mathcal{R}$ that are available at station $s \in \mathcal{S}$. Moreover, let $\iota^\infty_{s,r}$ be the number of train units of type $r \in \mathcal{R}$ that are planned to end at station $s \in \mathcal{S}$. Furthermore, let $\gamma_{q,r}$ and $\upsilon_{q,r}$ denote the number of train units of type $r \in \mathcal{R}$ that are coupled and uncoupled in composition change $q \in \mathcal{Q}$, respectively.

For the objective function, let $\omega_{t,p}$ be the cost of assigning composition $p \in \mathcal{P}_t$ to trip $t \in \mathcal{T}$. Similarly, let $\omega_{c,q}$ be the cost of assigning composition change $q \in \mathcal{Q}_c$ to transition $c \in \mathcal{C}$. Each train unit of deviation from the targeted ending inventory of type $r \in \mathcal{R}$ at station $s \in \mathcal{S}$ is penalized with cost $\omega_{s,r}$. Lastly, let $\omega_{d,k}$ be the cost of assigning completion $k \in \mathcal{K}_d$ to driver $d \in \mathcal{D}$.

We can now introduce the decision variables

$$X_{t,p} = \begin{cases} 1 & \text{if composition } p \in \mathcal{P}_t \text{ is assigned to trip } t \in \mathcal{T}, \\ 0 & \text{otherwise,} \end{cases}$$

$$Z_{c,q} = \begin{cases} 1 & \text{if composition change } q \in \mathcal{Q}_c \text{ is assigned to transition } c \in \mathcal{C}, \\ 0 & \text{otherwise,} \end{cases}$$

$$Y_{d,k} = \begin{cases} 1 & \text{if shunting driver } d \in \mathcal{D} \text{ performs completion } k \in \mathcal{K}_d, \\ 0 & \text{otherwise,} \end{cases}$$

$$I^\infty_{s,r} = \text{the number of train units of type } r \in \mathcal{R} \text{ that end at station } s \in \mathcal{S}.$$

A formulation for the RSSDRP is then given by:

$$\min \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}_t}\omega_{t,p}X_{t,p}+\sum_{c\in\mathcal{C}}\sum_{q\in\mathcal{Q}_c}\omega_{c,q}Z_{c,q} \tag{5.1}$$
$$+\sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}}\omega_{s,r}|I^{\infty}_{s,r}-\iota^{\infty}_{s,r}|+\sum_{d\in\mathcal{D}}\sum_{k\in\mathcal{K}_d}\omega_{d,k}Y_{d,k}$$
$$\text{s.t.} \sum_{p\in\mathcal{P}_t}X_{t,p}=1 \qquad \forall t\in\mathcal{T}, \tag{5.2}$$
$$X_{t,p}=\sum_{q\in\mathcal{Q}_{\delta^+(t)}:p_{q,t}=p}Z_{\delta^+(t),q} \qquad \forall t\in\mathcal{T},p\in\mathcal{P}_t, \tag{5.3}$$
$$X_{t,p}=\sum_{q\in\mathcal{Q}_{\delta^-(t)}:p'_{q,t}=p}Z_{\delta^-(t),q} \qquad \forall t\in\mathcal{T},p\in\mathcal{P}_t, \tag{5.4}$$
$$\iota^0_{s(c),r}-\sum_{\substack{c'\in\mathcal{C}:s(c')=s(c),\\ \tau^+(c')\le\tau^+(c)}}\sum_{q\in\mathcal{Q}_{c'}}\gamma_{q,r}Z_{c,q} \tag{5.5}$$
$$+\sum_{\substack{c'\in\mathcal{C}:s(c')=s(c),\\ \tau^-(c')\le\tau^+(c)}}\sum_{q\in\mathcal{Q}_{c'}}\upsilon_{q,r}Z_{c,q}\ge 0 \qquad \forall c\in\mathcal{C},r\in\mathcal{R},$$
$$I^{\infty}_{s,r}=\iota^0_{s,r}-\sum_{c\in\mathcal{C}:s(c)=s}\sum_{q\in\mathcal{Q}_c}\gamma_{q,r}Z_{c,q} \tag{5.6}$$
$$+\sum_{c\in\mathcal{C}:s(c)=s}\sum_{q\in\mathcal{Q}_c}\upsilon_{q,r}Z_{c,q} \qquad \forall s\in\mathcal{S},r\in\mathcal{R},$$
$$\sum_{d\in\mathcal{D}_{s(c)}}\sum_{k\in\mathcal{K}_d}\kappa^d_{m,k}Y_{d,k}=\sum_{q\in\mathcal{Q}_c:m\in\mathcal{M}_q}Z_{c,q} \qquad \forall c\in\mathcal{C},m\in\bigcup_{q\in\mathcal{Q}_c}\mathcal{M}_q, \tag{5.7}$$
$$\sum_{k\in\mathcal{K}_d}Y_{d,k}=1 \qquad \forall d\in\mathcal{D}, \tag{5.8}$$
$$X_{t,p}\in\{0,1\} \qquad \forall t\in\mathcal{T},p\in\mathcal{P}_t, \tag{5.9}$$
$$Z_{c,q}\in\{0,1\} \qquad \forall c\in\mathcal{C},q\in\mathcal{Q}_c, \tag{5.10}$$
$$I^{\infty}_{s,r}\in\mathbb{Z}_+ \qquad \forall s\in\mathcal{S},r\in\mathcal{R}, \tag{5.11}$$
$$Y_{d,k}\in\{0,1\} \qquad \forall d\in\mathcal{D},k\in\mathcal{K}_d. \tag{5.12}$$

The objective (5.1) considers the costs of the assigned compositions, of the assigned compositions changes, of any deviations from the planned ending inventory, and of the assigned duties for the shunting drivers. Note that the chosen cost parameters balance the relative importance of these different cost components and that the absolute value for the ending inventory deviations can easily be linearized. Constraints (5.2) ensure that a composition is assigned to each trip. Constraints (5.3) and (5.4) link the compositions and composition changes, by ensuring that a composition change is chosen for a transition that matches the compositions on respectively the succeeding and preceding trips of that transition. Constraints (5.5) ensure that there is always a non-negative number of train units available in the inventory of a station directly after each coupling moment $\tau^+(c)$ with $c \in \mathcal{C}$. These constraints thus ensure that no more train units are used than are available at the start of the planning horizon. Constraints (5.6) determine the number of train units of each train unit type that end at each station. Constraints (5.7) ensure that each task, as generated by the chosen composition changes, is covered by a shunting driver. Constraints (5.8) ensure that each shunting driver is assigned a completion of his or her duty. The remaining constraints state the domains for all decision variables.

5.5 Benders Decomposition for the RSSDRP

We solve formulation (5.1) – (5.12) by means of Benders decomposition (Benders 1962). In Benders decomposition, the problem is decomposed into a Benders master problem and a Benders subproblem. The Benders subproblem is solved for fixed solutions of the Benders master problem and used to generate cuts. These cuts are then added to the Benders master problem. These steps are repeated until no cuts can be found anymore or until some other stopping criterion is met.

In our case, the Benders master problem corresponds to rescheduling the rolling stock, which implies that solving the Benders master problem gives a circulation. Based on this circulation, we know which shunting tasks need to be executed at each station. For each station, we then consider a Benders subproblem in which we reschedule the shunting drivers at that station for the found shunting tasks. As both the Benders master problem and the Benders subproblem are integral problems, we solve, as in Cordeau et al. (2001), the linear relaxation of the RSSDRP by Benders decomposition and add branching decisions to ensure integrality.

More formally, for a fixed solution $(X, Z, I^\infty) \in \mathbb{R}^p_+$, with $p \in \mathbb{N}$, that satisfies constraints (5.2) – (5.6), the linear relaxation of the Benders subproblem is given by

$$\min \sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}_d} \omega_{d,k} Y_{d,k} \tag{5.13}$$

$$\text{s.t.} \quad \sum_{d \in \mathcal{D}_{s(c)}} \sum_{k \in \mathcal{K}_d} \kappa^d_{m,k} Y_{d,k} = \sum_{q \in \mathcal{Q}_c : m \in \mathcal{M}_q} Z_{c,q} \qquad \forall c \in \mathcal{C}, m \in \bigcup_{q \in \mathcal{Q}_c} \mathcal{M}_q, \tag{5.14}$$

$$\sum_{k \in \mathcal{K}_d} Y_{d,k} = 1 \qquad \forall d \in \mathcal{D}, \tag{5.15}$$

$$Y_{d,k} \geq 0 \qquad \forall d \in \mathcal{D}, k \in \mathcal{K}_d. \tag{5.16}$$

Note that this formulation is the linear programming relaxation of a set partitioning formulation in case of integral variables $Z_{c,q}$, as the right-hand side of constraints (5.14) is now fixed. Moreover, as exactly one composition change is chosen for each transition in an integral solution, each task appears either once or not at all, meaning that the right-hand side of these constraints always equals 0 or 1. This formulation is thus very similar to the linear relaxation of a set partitioning model for the crew rescheduling problem.

The dual of this formulation is given by

$$\max \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} \sum_{m\in\mathcal{M}_q} Z_{c,q}\lambda_{c,m} + \sum_{d\in\mathcal{D}} \mu_d \tag{5.17}$$

$$\text{s.t.} \quad \sum_{c:d\in\mathcal{D}_{s(c)}} \sum_{q\in\mathcal{Q}_c} \sum_{m\in\mathcal{M}_q:\kappa^d_{m,k}=1} \lambda_{c,m} + \mu_d \leq \omega_{d,k} \qquad \forall d\in\mathcal{D}, k\in\mathcal{K}_d, \tag{5.18}$$

$$\lambda_{c,m} \in \mathbb{R} \qquad \forall c\in\mathcal{C}, m\in\bigcup_{q\in\mathcal{Q}_c}\mathcal{M}_q, \tag{5.19}$$

$$\mu_d \in \mathbb{R} \qquad \forall d\in\mathcal{D}, \tag{5.20}$$

where λ and μ are the dual vectors corresponding to constraints (5.14) and (5.15), respectively. As picking the 0 vector for (λ,μ) is a feasible solution for the above problem, we see that this dual problem either is feasible and bounded or that it is unbounded. In the first case, we can determine an optimal solution for the problem, while in the second case we are able to find an extreme ray. Let P be the set of all extreme points and R the set of all extreme rays of the considered polyhedron.

The relaxation of the Benders master problem that takes into account all extreme points and extreme rays is then given by

$$\begin{aligned}\min \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}_t} \omega_{t,p}X_{t,p} + \sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} \omega_{c,q}Z_{c,q} \\ + \sum_{s\in\mathcal{S}} \sum_{r\in\mathcal{R}} \omega_{s,r}|I^\infty_{s,r} - \iota^\infty_{s,r}| + z\end{aligned} \tag{5.21}$$

$$\text{s.t.} \quad \sum_{p\in\mathcal{P}_t} X_{t,p} = 1 \qquad \forall t\in\mathcal{T}, \tag{5.22}$$

$$X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^+(t)}:p_{q,t}=p} Z_{\delta^+(t),q} \qquad \forall t\in\mathcal{T}, p\in\mathcal{P}_t, \tag{5.23}$$

$$X_{t,p} = \sum_{q\in\mathcal{Q}_{\delta^-(t)}:p'_{q,t}=p} Z_{\delta^-(t),q} \qquad \forall t\in\mathcal{T}, p\in\mathcal{P}_t, \tag{5.24}$$

$$\begin{aligned}\iota^0_{s(c),r} - \sum_{\substack{c'\in\mathcal{C}:s(c')=s(c),\\ \tau^+(c')\leq\tau^+(c)}} \sum_{q\in\mathcal{Q}_{c'}} \gamma_{q,r}Z_{c,q} \\ + \sum_{\substack{c'\in\mathcal{C}:s(c')=s(c),\\ \tau^-(c')\leq\tau^+(c)}} \sum_{q\in\mathcal{Q}_{c'}} \upsilon_{q,r}Z_{c,q} \geq 0 \qquad \forall c\in\mathcal{C}, r\in\mathcal{R},\end{aligned} \tag{5.25}$$

$$\begin{aligned}I^\infty_{s,r} = \iota^0_{s,r} - \sum_{c\in\mathcal{C}:s(c)=s} \sum_{q\in\mathcal{Q}_c} \gamma_{q,r}Z_{c,q} \\ + \sum_{c\in\mathcal{C}:s(c)=s} \sum_{q\in\mathcal{Q}_c} \upsilon_{q,r}Z_{c,q} \qquad \forall s\in\mathcal{S}, r\in\mathcal{R},\end{aligned} \tag{5.26}$$

$$\sum_{c\in\mathcal{C}} \sum_{q\in\mathcal{Q}_c} \sum_{m\in\mathcal{M}_q} \lambda_{c,m}Z_{c,q} + \sum_{d\in\mathcal{D}} \mu_d \leq z \qquad \forall(\lambda,\mu)\in P, \tag{5.27}$$

$$\sum_{c\in\mathcal{C}}\sum_{q\in\mathcal{Q}_c}\sum_{m\in\mathcal{M}_q}\lambda_{c,m}Z_{c,q}+\sum_{d\in\mathcal{D}}\mu_d\leq 0 \qquad \forall(\lambda,\mu)\in R, \tag{5.28}$$

$$0\leq X_{t,p}\leq 1 \qquad \forall t\in\mathcal{T},p\in\mathcal{P}_t, \tag{5.29}$$

$$0\leq Z_{c,q}\leq 1 \qquad \forall c\in\mathcal{C},q\in\mathcal{Q}_c, \tag{5.30}$$

$$I^{\infty}_{s,r}\geq 0 \qquad \forall s\in\mathcal{S},r\in\mathcal{R}, \tag{5.31}$$

$$z\in\mathbb{R}. \tag{5.32}$$

Note that all constraints, except for constraints (5.27) – (5.28) and constraint (5.32), of this model are part of the formulation of the RSSDRP defined in Section 5.4. Constraints (5.27) are then the Benders cuts that correspond to the extreme points. Similarly, constraints (5.28) are the Benders cuts that correspond to the extreme rays.

While one could solve the above problem that contains all Benders cuts, the number of Benders cuts is far too large in practice for this problem to be solved to optimality or even to be loaded into computer memory. Instead, we add the cuts iteratively, implying that we only consider some subset $P^{'}\subseteq P$ of the extreme points and some subset $R^{'}\subseteq R$ of the extreme rays each time when the Benders master problem is solved. Based on the found solution, we then solve the Benders subproblem again to generate new extreme points or extreme rays that are added to the sets P' and R', respectively.

5.5.1 Solving the Relaxed Benders Master Problem

Note that the Benders master problem, except for the addition of Benders cuts, is similar to the model proposed by Fioole et al. (2006) for rolling stock scheduling. As we only solve the linear relaxation of this problem, and as the number of variables of this model is relatively modest, we use a commercial linear programming (LP) solver for this. Note that the integral version of this model can also be solved effectively by a MILP solver, as illustrated by Fioole et al. (2006) and Nielsen (2011).

5.5.2 Solving the Relaxed Benders Subproblem

Due to the large number of possible completions for a duty, and hence the large number of variables for the subproblem, we apply column generation to solve the subproblem. In column generation, we iteratively solve a restricted master and pricing problem to generate promising variables. This is then repeated until no variable can be found which could improve the current solution. For a general introduction to column generation, we refer to Lübbecke and Desrosiers (2005).

In our case, the restricted master problem corresponds to solving model (5.13) – (5.16) with a subset of the generated completions. The pricing problem then corresponds to generating completions for the available shunting drivers. Note that the pricing problem decomposes over the different drivers, in addition to the decomposition of the subproblems over the stations that is implied by the Benders decomposition.

Completions can be generated for a driver $d \in \mathcal{D}$ based on a directed acyclic graph $D^d = (V^d, A^d)$. Each node in this graph corresponds to a shunting task that occurs at the considered station. Arcs are added between tasks that can be executed in succession. In particular, we add an arc from task m_1 to m_2 if the time between $\tau^e_{m_1}$ and $\tau^s_{m_2}$ is large enough to cover the time it takes to go from the ending location of task m_1 to the starting location of task m_2. Here, τ^s_m and τ^e_m give the starting and ending time for task m, respectively.

We can now find completions by finding paths through the graph D^d. In particular, we add a node v^d_s to D^d that indicates the start of the duty and a node v^d_e that indicates the end of the duty. We ensure that the start node is only connected to tasks that start after the starting time of the duty, taking into account any tasks that have been executed already by the driver before the start of the planning horizon. Moreover, an end node is only connected to nodes that end before the end of the duty. A completion is then equal to a path from v^d_s to v^d_e.

To additionally fulfill the conditions of a meal break, we record for each arc if it accommodates a break for the driver. The pricing problem then reduces to finding a path from v_s to v_e that includes an arc that allows for a break. This can be modeled as finding a resource constrained shortest path. It is well known that this problem is, in general, $\mathcal{NP}$-hard. However, efficient algorithms have been developed to solve the problem. We use a labeling algorithm to solve the resource constrained shortest path problem, see, e.g., Irnich and Desaulniers (2005).

5.5.3 Solving the Integral Problem

The methods described so far solve the linear relaxation of the model (5.1) – (5.12) using Benders decomposition. It remains to find the integer optimum of the model. We do so by integrating the Benders decomposition procedure into a Branch and Bound approach, as proposed by Cordeau et al. (2001). In this approach, we solve in each node of the Branch and Bound tree a linear relaxation of the RSSDRP by means of Benders decomposition and additionally take into account all branching decisions relevant to this node. New branching decisions, and thus nodes in the tree, are added when the solution is fractional. This process is repeated until the optimal solution is found or when the best-found solution is within a given tolerance of the optimal solution.

The branching scheme we use is as follows. With regards to the master problem, we use a SOS1 based branching rule on constraints (5.2) that has been proposed by Haahr et al. (2016) for rolling stock (re)scheduling. In particular, if some composition is chosen with a fractional value for some trip $t' \in \mathcal{T}$, we can identify a set of compositions $P' \subseteq \mathcal{P}_{t'}$ such that $0 < \sum_{p \in P'} X_{t',p} < 1$. We can then branch by imposing the above sum to be equal to one in one of the branches and zero in the other. Note that by the reasoning of Fioole et al. (2006), the above branching decisions ensure the integrality of the Benders master problem.

For the Benders subproblems, we apply Ryan-Foster branching (Ryan and Foster 1981). In particular, we branch on the assignment of a task to a shunting driver. If this assignment is fractional, we enforce that the task is fixed to the driver in one branch while in the other branch we forbid that the driver performs the task. Note that the above decisions lead to an integral solution in set partitioning problems and thus also in our Benders subproblem. Moreover, note that these branching decisions can be taken into account in the pricing problem of the Benders subproblem by deleting appropriate arcs from the graph.

Combining the above branching decisions leads to an integral solution to model (5.1) – (5.12). First of all, the Benders decomposition approach ensures that each solution of the relaxation is feasible with respect to the shunting driver rescheduling. Moreover, branching decisions on the master problem ensure that an integral rolling stock circulation is found. Furthermore, the branching decisions on the subproblems ensure that integral solutions to the subproblems are found. Last, we note that these branching decisions for the subproblems have an impact on the master problem through the generated Benders cuts.

5.6 Arc-Based Model for the RSSDRP

As an alternative to the Benders decomposition approach, we consider a large, but compact, MILP formulation for the RSSDRP that can be solved by a commercial MILP solver. This is achieved by considering an arc-based formulation for the rescheduling of shunting drivers instead of the set partitioning formulation that we consider in our Benders decomposition approach. This arc-based formulation can then be linked to the formulation of Fioole et al. (2006) for rolling stock rescheduling in a similar way as for model (5.1) – (5.12).

For this arc-based formulation of shunting driver rescheduling, we consider for each driver $d \in \mathcal{D}$ again a directed graph $D^d = (V^d, A^d)$. This graph is the same as the one considered in the pricing problem of our Benders subproblem, where arcs are present in the graph between two task nodes when these tasks can be operated in sequence by this driver. However, we now consider all the tasks that may need to be executed in this graph, i.e., all tasks that could be generated by a composition change at the considered station. Each path between a start-of-duty node v_s^d and an end-of-duty node v_e^d then corresponds again to a feasible completion of the duty for this driver if it contains a meal break.

In addition to the variables we saw before for rolling stock rescheduling, we now consider variables for the shunting driver rescheduling that determine the sequencing of tasks for each driver. We consider the decision variables

$$S_{a,d} = \begin{cases} 1 & \text{if driver } d \text{ uses arc } a \in A^d, \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, let $V_m \subseteq \bigcup_{d \in \mathcal{D}} V^d$ be the set of nodes corresponding to task $m \in \mathcal{M}$. Let $b_{a,d}$ indicate whether arc a allows driver d to have his or her break during the time between the two tasks that are implied by arc a. Furthermore, let $\delta^-(v)$ and $\delta^+(v)$ give the set of incoming and outgoing arcs in A^d for a node $v \in V^d$, respectively. Lastly, let $c_{a,d}$ give the cost of using arc $a \in A^d$.

The arc-based formulation for the RSSDRP is then given by

$$\min \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}_t} \omega_{t,p}X_{t,p} + \sum_{c\in\mathcal{C}}\sum_{q\in\mathcal{Q}_c} \omega_{c,q}Z_{c,q} \tag{5.33}$$

$$+ \sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}} \omega_{s,r}|I^{\infty}_{s,r} - \iota^{\infty}_{s,r}| + \sum_{d\in\mathcal{D}}\sum_{a\in A^d} c_{a,d}S_{a,d}$$

s.t. $(5.2) - (5.6), (5.9) - (5.11),$

$$\sum_{a\in\delta^-(v)} S_{a,d} = \sum_{a\in\delta^+(v)} S_{a,d} \qquad \forall d\in\mathcal{D}, v\in V^d\setminus\{v^d_s, v^d_e\}, \tag{5.34}$$

$$\sum_{a\in\delta^+(v^d_s)} S_{a,d} = 1 \qquad \forall d\in\mathcal{D}, \tag{5.35}$$

$$\sum_{a\in\delta^-(v^d_e)} S_{a,d} = 1 \qquad \forall d\in\mathcal{D}, \tag{5.36}$$

$$\sum_{v\in V_m}\sum_{a\in\delta^-(v)} S_{a,d} = \sum_{q\in\mathcal{Q}_c:m\in\mathcal{M}_q} Z_{c,q} \qquad \forall c\in\mathcal{C}, m\in\bigcup_{q\in\mathcal{Q}_c}\mathcal{M}_q, \tag{5.37}$$

$$\sum_{a\in A^d} b_{a,d}S_{a,d} \geq 1 \qquad \forall d\in\mathcal{D}, \tag{5.38}$$

$$S_{a,d}\in\{0,1\} \qquad \forall a\in A, d\in\mathcal{D}. \tag{5.39}$$

Note that we consider the same objective as in formulation (5.1) – (5.12) for rolling stock rescheduling and now determine the costs for shunting driver rescheduling based on the chosen arcs. Moreover, all constraints related to rolling stock rescheduling are the same as in (5.1) – (5.12) as well. For the shunting driver rescheduling, constraints (5.34) ensure flow conservation in the graph of each driver. Moreover, (5.35) and (5.36) ensure that the driver starts and ends his or her duty, respectively. Together, constraints (5.34) – (5.36) ensure that each driver follows a path through the graph defining the tasks that he or she executes. Constraints (5.37) ensure that each task is performed by a driver. Constraints (5.38) ensure that each driver can take a break during his or her duty. The remaining constraints define the domain of the decision variables.

Note that while the number of possible duties for the shunting drivers, and hence the number of variables of model (5.1) – (5.12), is generally exponential in the number of tasks, the number of arcs in each graph D^d is polynomial in the number of tasks. Hence, we can solve model (5.33) – (5.39) using a commercial MILP solver. However, it is well known that the linear programming relaxation of the set partitioning formulation is in general stronger than the linear programming relaxation of the arc-based formulation above.

5.7 Computational Experiments

In this section, we perform computational experiments to show the performance of both proposed solution methods on instances of NS. Our goal is to evaluate the relative performance of the methods and to evaluate the extent to which the RSSDRP can be solved for these instances. Moreover, we would like to determine to what extent the solutions obtained for the RSSDRP differ from the solutions obtained in a sequential approach. Before we discuss the computational experiments, we introduce the considered instances and the used objective parameters.

5.7.1 Problem Instances

We consider rolling stock instances from the 2018 timetable operated by NS. All instances consider a planning horizon of a single day and correspond to the Tuesday timetable, as this is the day with the highest average passenger numbers at NS. Moreover, we focus in our experiments on the Intercity services operated by NS and thus reschedule only the rolling stock types that are used to operate these Intercity services.

Disruptions correspond in this case to blockages of the railway infrastructure on a part of the network as operated by NS. The locations of the considered disruptions, and thus of the blocked section of infrastructure, are shown in the NS network in Figure 5.4. To adjust the timetable for such disruptions, the Dutch infrastructure manager has made contingency plans that describe which modifications need to be made to the timetable. These contingency plans also define how the rolling stock transitions between trips should be updated at the stations affected by the disruption. We then obtain, for each of the disruptions that we consider, a new timetable by applying the corresponding contingency plan to the original timetable.

Next to considering multiple disruptions, we also create additional instances by considering different time frames during which the disruption takes place. As we consider a real-time rescheduling setting, any decisions which have been made before the disruption becomes known cannot be changed. Hence, the later during the day the disruption occurs, the more of the rolling stock circulation and the shunting plans becomes fixed. A disruption that occurs earlier during the day is thus generally harder to solve than one at the end of the day.

An overview of all instances is given in Table 5.1, which also gives some summary statistics about each instance. Three instance classes are formed by the location of the disruption. Moreover, for each instance class we consider three instances based on the time during which the disruption occurs. Note that these three disruption times are picked such that the disruption is either during the morning peak hours, between the peak hours, or during the evening peak hours.

5.7.2 Objective Parameters

Our objective function balances passenger demand with operational costs. The parameters that we consider in the objective function are given in Table 5.2. For the compositions, we penalize the cancellation of a trip, which corresponds to assigning

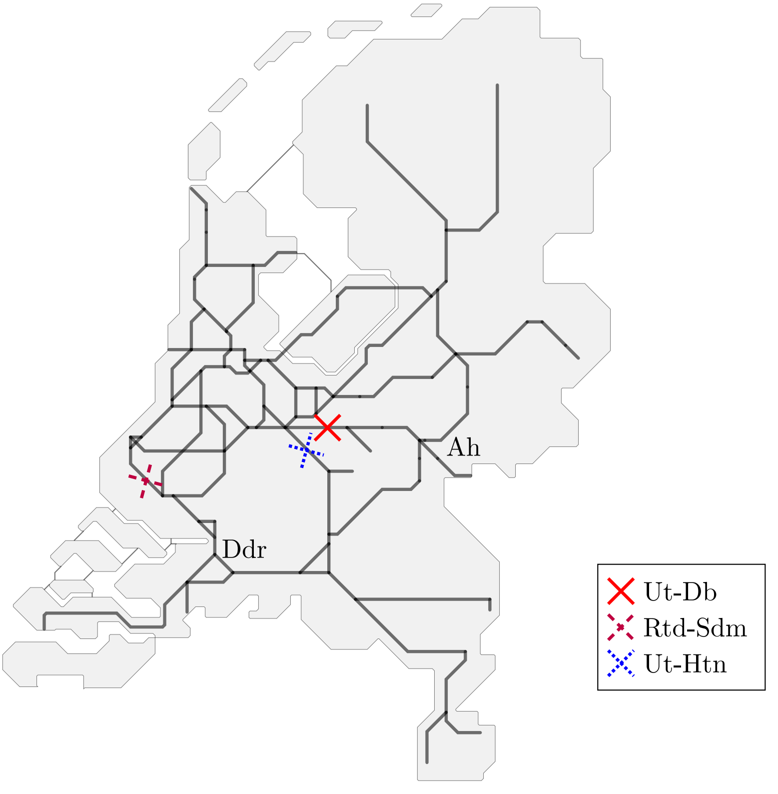


Figure 5.4: Disruptions in the NS network. Moreover, the locations of stations Dordrecht (Ddr) and Arnhem (Ah) are shown.

Table 5.1: Overview of the considered instances. Reported are the instance name (*Instance*), the blocked section of infrastructure (*Section*), the time-frame of the disruption (*Time*), the number of (remaining) trips in the instance ($|\mathcal{T}|$), the number of transitions in the instance ($|\mathcal{C}|$) and the total number of available shunting drivers in the instance ($|\mathcal{D}|$).

Instance	Section	Time	$\lvert\mathcal{T}\rvert$	$\lvert\mathcal{C}\rvert$	$\lvert\mathcal{D}\rvert$
Db-L	Ut–Db	7–10	4059	4245	276
Db-M	Ut–Db	10–12	3301	3467	276
Db-S	Ut–Db	14–18	2305	2449	224
Rtd-L	Rtd–Sdm	9–11	3547	3725	276
Rtd-M	Rtd–Sdm	12–14	2802	2958	247
Rtd-S	Rtd–Sdm	14–18	2288	2439	224
Htn-L	Ut–Htn	8–10	3780	3957	276
Htn-M	Ut–Htn	12–14	2781	2930	247
Htn-S	Ut–Htn	15–17	2044	2179	214

a composition without a train unit to that trip. Moreover, we penalize any shortage of seats, which is incurred per seat that is short compared to the expected passenger demand for the trip and per kilometer that is traveled on the trip. In addition, there is a fixed cost per kilometer for using a composition, which is incurred per carriage in the composition.

Table 5.2: Objective Parameters

Parameter	Cost Component	Value
$\omega_{t,p}$	Cancellation	1000000
	Seat Shortage	0.2
	Mileage	0.1
$\omega_{c,q}$	New Shunting	1000
	Changed Shunting	100
	Canceled Shunting	50
$\omega_{s,r}$	End Inventory Deviation	10000

To limit changes to the shunting plans, we penalize changes in the composition changes compared to the original circulation. So, e.g., when coupling would occur in the original composition change but not in the new one. The cost parameter depends on how the shunting movement is changed, where a distinction is made between adding a new shunting movement, changing an existing one, or canceling an existing shunting movement.

Lastly, we penalize any deviations in the ending inventory, which represent the costs that need to be made to rebalance inventories overnight and to start with a potential shortage of train units the next morning. These costs are incurred per train unit of deviation at each station and for each rolling stock type. Note that we do not consider a cost parameter for the shunting drivers here, which implies that we consider shunting driver rescheduling as a feasibility problem. Hence, we only generate feasibility cuts in our Benders decomposition approach.

5.7.3 Comparison of Solution Approaches

In this section, we explore the performance of the two proposed methods on instances of NS. The results of our experiments are summarized in Table 5.3. These experiments were performed on a computer with an Intel Xeon Gold 6130@2.1GHz processor. Moreover, both the arc-based model as well as the LP models in our Benders decomposition method have been solved using the CPLEX 12.10 solver. Furthermore, all methods have been programmed in the Java programming language. The solvers were stopped after a maximum solving time of 15 minutes or after finding a solution that is proven to be less than 0.001% from the optimal solution.

Table 5.3: Comparison of Benders decomposition approach to the arc-based model. Reported for each method is the reached optimality gap (*Gap*), the solving time (*Time*) and the number of nodes explored (*Nodes*). Moreover, we report for the Benders decomposition approach the number of added Benders cuts (*Cuts*), the time spent in the Benders master problem (*Mast.*) and the time spent in the Benders subproblem (*Sub.*)

Instance	Arc-Based Model			Benders Decomposition Approach					
	Gap (%)	Time (s)	Nodes	Gap (%)	Time (s)	Nodes	Cuts	Mast. (s)	Sub. (s)
Db-L	0.00	334	110	0.00	381	605	47	269	67
Db-M	0.00	84	1	0.01	909	1988	211	607	183
Db-S	0.00	27	21	0.00	54	245	7	13	17
Rtd-L	0.00	129	167	0.00	128	200	21	85	23
Rtd-M	0.00	29	1	0.00	67	278	12	20	23
Rtd-S	0.00	18	1	0.00	52	235	4	15	16
Htn-L	0.00	136	1	0.01	909	1632	438	592	189
Htn-M	0.00	28	1	0.00	31	121	8	4	12
Htn-S	0.00	12	1	0.00	53	307	4	10	18

The results in Table 5.3 show that the integrated problem can generally be solved well within the time limit, where all but two instances are solved to optimality within 15 minutes by both methods. Moreover, the average solving time is significantly below 15 minutes for both approaches, where the arc-based model takes an average of 89 seconds and the Benders decomposition approach an average of 287 seconds. A clear trend among the instances is that those that correspond to a disruption earlier during the day are harder to solve than those that correspond to a later disruption, which agrees with the larger number of trips and transitions present in this first category of instances.

When zooming in on the relative performance of both methods, we see that for most instances the arc-based model shows better performance than the Benders decomposition approach. In particular, we see that the Benders decomposition approach is unable to prove optimality for the Db-M and Htn-L instances, although the solution found is in both cases only 0.01% away from optimality. This while the arc-based model can find an optimal solution within 15 minutes of solving time for both instances. For the other instances, the differences between the methods are smaller, where we, e.g., see that for the Htn-M instance the Benders decomposition approach is able to find an optimal solution in about the same amount of time as the arc-based model.

A clear difference can be seen between both methods when looking at the number of explored nodes, where the arc-based model always shows a significantly lower number of nodes than the Benders decomposition approach. Surprisingly, for 6 of the instances the optimal solution is found in the root node for the arc-based model. This result can be mainly attributed to the successful addition of general MILP cuts by CPLEX in the root node, significantly reducing the integrality gap that is seen after exploring the root node. Moreover, we found that especially finding integral shunting driver schedules was difficult in the Benders decomposition approach, explaining the larger number of nodes that are explored in this method.

Another interesting result is in the number of Benders cuts that are added in the Benders decomposition approach. In particular, we see that for all instances between 4 and 450 cuts are generated during the solving process. First of all, this indicates that the first LP solution in Benders decomposition, which corresponds to the LP of the rolling stock rescheduling model without taking into account shunting driver rescheduling, is never feasible. This is a clear sign that using a sequential process leads to infeasible solutions. Moreover, the relatively minor number of Benders cuts also shows that this approach does not suffer from a very significant tail-off effect, as is often encountered in other applications of Benders decomposition.

When looking at the time spent in the Benders master problem and Benders subproblem in the Benders decomposition approach, we see that the time spent in the master problem is larger for most of the instances. The difference between these problems is especially significant for the harder instances, in this case Db-L, Db-M, Rtd-L, and Htn-L. The cause for this larger time spent in the master problem seems to be the degeneracy of the LP formulation, which results in a larger number of LP iterations being executed by CPLEX. This effect has been noted before for the model of Fioole et al. (2006) in Chapter 4. Moreover, due to the decomposition of the subproblem over the stations, the shunting driver rescheduling problems are generally relatively small. This reduces the solving time needed for the Benders subproblems.

5.7.4 Comparison to Sequential Approach

In this section, we compare the solutions obtained for the RSSDRP to the solutions of a sequential approach. In this sequential approach, the rolling stock is rescheduled first without taking into account the availability of shunting drivers. Next, the shunting drivers are then rescheduled for each station. Note that we do not perform any feedback loop here, implying that it is likely that no feasible duty assignments can be found for some of the stations. Our aim is then to quantify the infeasibility of the solutions found by this sequential approach and to compare the rolling stock circulations found by both approaches.

In Table 5.4, the results for the sequential approach are shown alongside those of the integrated approach. In particular, this table shows the objective value and number of canceled trips obtained for each instance by the sequential approach when not taking into account the shunting drivers. Moreover, it shows for how many of the stations no feasible duty assignment for the shunting drivers can be found in the sequential approach. As a comparison, it also gives the objective value and number of canceled trips obtained for each instance by solving the RSSDRP, in this case by using the arc-based model. Furthermore, Figure 5.5 plots for each instance the stations for which no feasible duty assignment can be found in the sequential approach on a map.

Table 5.4: Comparison of the results obtained for the RSSDRP and those of a sequential approach. For both approaches, the obtained objective value (*Cost*) and the number of canceled trips (*Canc.*) in the obtained solution is given. Moreover, it is shown for how many stations the results of the sequential approach are infeasible (*Inf. Stat.*)

	Sequential			Integrated	
Instance	Cost ($\times 10^6$)	Canc.	Inf. Stat.	Cost ($\times 10^6$)	Canc.
Db-L	10.21	10	7	11.21	11
Db-M	10.20	10	8	13.20	13
Db-S	10.20	10	4	11.21	11
Rtd-L	20.23	20	8	24.24	24
Rtd-M	13.21	13	6	17.21	17
Rtd-S	6.22	6	4	7.22	7
Htn-L	18.23	18	8	23.24	23
Htn-M	2.19	2	6	3.19	3
Htn-S	2.20	2	4	4.20	4

The results in Table 5.4 show that there are significant differences between the rolling stock circulations found by the sequential approach and those found by solving the RSSDRP. In particular, it can be seen that not taking into account the shunting drivers leads to infeasibilities at a significant number of stations. Looking at the objective values obtained by both approaches, we also see that not taking into account the shunting drivers leads to an underestimation of the actual costs. This is mostly because the limited availability of shunting drivers also limits the number of shunting actions that can be performed. This can again lead to the necessity of canceling trips in the rolling stock circulation, which incurs a large objective function penalty. When looking at the number of canceled trips in the solutions obtained by both approaches, we see that taking into account the availability of shunting drivers indeed leads to additional trips being canceled for each instance. Combining the above observations, the circulations obtained by the sequential approach are likely far away from those

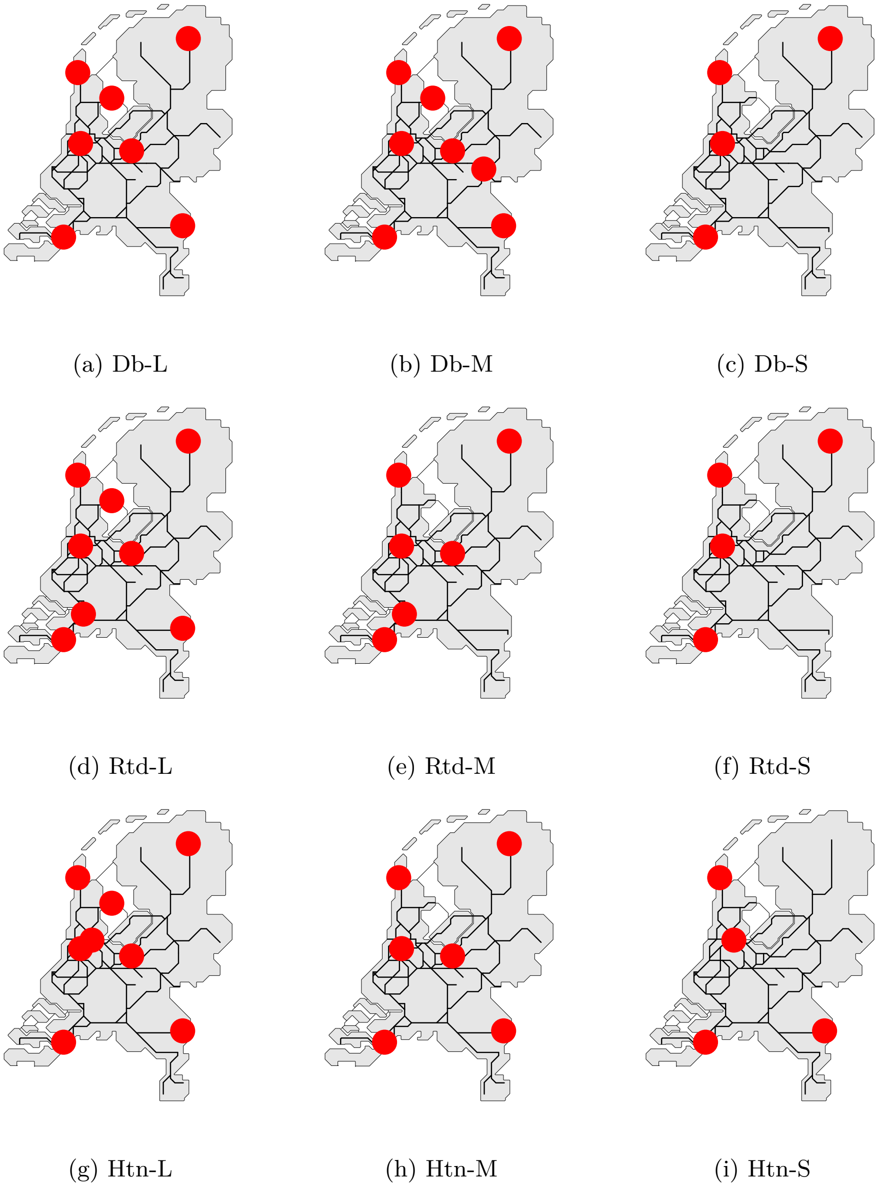


Figure 5.5: Overview of the stations for which no feasible solution can be found for each instance.

circulations that can be operated in practice.

When looking at Figure 5.5, we see that the stations at which infeasibilities occur vary between the instances. Some stations appear in most or even all of the instances. This can most likely be explained by the fact that, when the shunting drivers are not considered, shunting movements may be planned which would lead to a rolling stock circulation of lower cost. In particular, such movements can even be planned when no disruption is present in case the cost reduction achieved by including the movement is larger than the cost assigned to changing the shunting movement. However, we also see that some stations appear less often. This is, e.g., the case for station Arnhem (see *Ah* in Figure 5.4) in the east of the country, which appears only in instance Db-M, and station Dordrecht (see *Ddr* in Figure 5.4) in the south-west of the country, which only appears in instances Rtd-L and Rtd-M. In these cases, the infeasibilities are likely a direct effect of the faced disruptions. Here, station Arnhem is, e.g., likely impacted by a blockage of the tracks at *Db* since some train services to Arnhem pass this piece of the infrastructure. Station Arnhem is not directly impacted by any of the other disruptions.

5.8 Conclusion

In this chapter, we looked at the problem of integrated rolling stock and shunting driver rescheduling, where the shunting tasks that follow from rolling stock rescheduling have to be assigned to the shunting drivers. By integrating these problems, infeasibilities are prevented that can occur in a sequential approach when the generated shunting tasks cannot be covered by the available shunting drivers. We proposed a Benders decomposition approach for this integrated problem, where the Benders master problem corresponds to a rolling stock rescheduling problem with additional cuts and the subproblem to a set partitioning formulation for rescheduling the shunting drivers. In addition, we have proposed an arc-based model that is solved directly by a MILP solver. This model was obtained by considering a flow-based approach for shunting driver rescheduling instead of the set partitioning approach used in Benders decomposition.

Numerical experiments on instances of NS showed that the arc-based model shows overall better performance than the Benders decomposition approach, although the relative performance of the models differs per instance. Moreover, we showed that the integrated problem can generally be solved quickly, even to optimality, by both solution methods. A comparison to a sequential approach furthermore showed that an integrated approach can prevent a significant number of infeasibilities, which occur in multiple stations throughout the considered network when not taking into account the availability of shunting drivers.

With our work, we show the benefit that an integrated approach for solving the rolling stock rescheduling and shunting driver rescheduling problem can bring. This aligns with the earlier work of Haahr and Lusby (2017), who show the benefit of integrating rolling stock scheduling with the parking of trains at the shunting yard. Further research could focus on further exploring the integration of rolling stock rescheduling with the shunting processes at the stations, e.g., looking at the integration of both shunting driver rescheduling and the parking of trains with rolling stock rescheduling. We believe that our Benders decomposition approach could be an interesting starting point for such further research.

Chapter 6

Summary and Conclusions

In this thesis, we have looked at the scheduling and rescheduling of rolling stock at passenger railway operators. The main aim of these problems is to assign the available train units to the trips in the timetable such that both passenger comfort and operational costs are considered. In the chapters of this thesis, we have looked both at these traditional problems, as well as at extensions of the rolling stock rescheduling problem. In our work, we specifically evaluate the existing optimization models and propose new optimization models for these problem settings.

In Chapter 2, we have looked at the models that are used for rolling stock scheduling. Many of these same models are also used in the rolling stock rescheduling context. First, we compared these models from the literature based on the characteristics of the problem setting that they consider. Second, we made an analytical comparison between two models proposed in the literature: the Composition model proposed by Fioole et al. (2006) and the Hypergraph model proposed by Borndörfer et al. (2016). Our analysis shows that the best model to use strongly depends on the faced problem setting. While the Composition model and Hypergraph model lead to the same linear programming relaxation value for the rolling stock rescheduling setting of Netherlands Railways (NS), our numerical results show better performance for the former model based on instances of NS.

In Chapter 3, we then looked at the rolling stock rescheduling problem. Here, we specifically proposed a heuristic for solving this problem, based on the observation that including all of the details faced by rolling stock dispatchers in exact solution methods is often challenging and leads to relatively long running times. The Variable Neighborhood Search (VNS) heuristic that we propose makes use of three newly introduced neighborhoods for the rolling stock rescheduling problem. Moreover, we show how an additional neighborhood can be used to extend this heuristic to a richer rolling stock rescheduling setting in which the turnings at terminal stations are to be decided in the optimization problem. Our results show that the heuristic can provide high-quality solutions quickly, allowing it to be used by rolling stock dispatchers. Moreover, our results show that the heuristic can outperform an exact method for some instances in a rolling stock rescheduling setting with flexible turning.

In Chapter 4, we looked at how rolling stock rescheduling can be used to reduce passenger delays. The chosen rolling stock assignment has an impact on these delays as a trip will leave with a delay when one of the train units assigned to this trip arrives with a delay. Changing the rolling stock assignment can thus lead to a different distribution of delays on the trips in the timetable. In this chapter, we propose two optimization models to solve this problem of minimizing the delays faced by passengers, one of which is based on a flow-based approach and one on a path-based approach. Our results show that this problem can be solved relatively quickly and that significant delay reductions can be achieved by solving it. Moreover, our results highlight that especially the use of flexible turnings at terminal stations turns out to be beneficial for achieving such delay reductions.

In Chapter 5, we looked at another extension of the rolling stock rescheduling problem. This extension was motivated by the interaction of the rolling stock assignment with the duties executed by shunting drivers at the stations. Here, shunting drivers are responsible for moving the train units during shunting movements at the station. Traditionally, these problems are solved sequentially, but this may lead to infeasibilities and a loss of solution quality. In this chapter, we proposed two solution methods for solving these two problems in an integrated way: a Benders decomposition approach and an arc-based approach. Our experiments show that this problem can be solved well for rescheduling instances of NS and that a significant number of infeasibilities is prevented compared to a sequential approach.

6.1 Practical Implications

Our results offer several interesting insights for railway passenger operators. First of all, our work shows that further integrating rolling stock rescheduling into the overall rescheduling process can provide significant benefits. Here, we find that rolling stock rescheduling can efficiently be used to reduce passenger delays, especially when allowing some more freedom in the transitions chosen at terminal stations. Similarly, integrated rescheduling of rolling stock and shunting drivers can prevent infeasibilities occurring at the railway node level. Based on the results of these papers, we would advise railway operators to look at further integrating these extensions as part of their decision support systems.

Secondly, we have presented a heuristic for rolling stock rescheduling that we believe is very suitable for everyday usage at railway operators. It offers several benefits over exact approaches for railway operators. First, it can be easily extended by introducing additional local search neighborhoods. As an example, we show that flexible turning can be included by introducing a neighborhood that swaps turnings at terminal stations. Second, the steps taken in the local search neighborhoods are easier to understand for rolling stock dispatchers. In this way, dispatchers can evaluate the moves made in the heuristic and judge if these are indeed beneficial changes in their opinion. Third, the heuristic offers potential for especially those settings which are hard to solve by exact solution methods. As an example, we show that the heuristic can outperform an exact solution method for some instances when including flexible turning opportunities at the terminal stations. In this way, the heuristic offers the potential to railway companies to take further steps in incorporating more real-life details into rolling stock rescheduling.

Third, our work in Chapter 2 can act as a guide for railway companies in selecting a model for rolling stock scheduling or rolling stock rescheduling. The literature review that we perform in this chapter highlights the operational context considered by models proposed in the literature and can thus act as a guide for selecting a model that fits with the operational context of the operator. Moreover, our analytical and numerical results in this chapter show the relative performance of two commonly used models in the literature. In particular, our results provide an interesting trade-off between these models: while the Hypergraph model allows to model a richer rolling stock scheduling setting, the more compact Composition model provides better performance for the simpler rolling stock scheduling setting of NS.

Fourth, our work also shows the benefit that more flexible turnings at terminal stations can provide. For example, we show in Chapter 4 that flexible turnings can help to find significant delay reductions. Moreover, we also show that rolling stock rescheduling instances with flexible turning can be solved relatively well, where we suggest a local search neighborhood for flexible turning in Chapter 3. Our results strengthen those found by Nielsen (2011), but also show that modern solvers allow solving instances in which more flexible turning is considered. We therefore advise railway operators, like NS, who currently mostly fix the turnings in their optimization models, to further evaluate the use of flexible turnings and to identify in which situations these could be used to find better rolling stock circulations.

6.2 Future Research

This thesis offers a number of interesting directions for future research. First of all, our results in Chapter 5 show the benefits of integrating the rescheduling of shunting drivers with the rescheduling of rolling stock. Similarly, Haahr and Lusby (2017) have shown the potential of integrating the decisions on the parking of train units at the shunting yards with the scheduling of rolling stock. We believe that further benefits could be obtained by looking also at the other shunting processes that occur at the stations, such as finding suitable paths in the timetable to go from the station to the shunting yard. In particular, this would require the development of a framework that combines the planning and rescheduling of these shunting processes with the scheduling and rescheduling of rolling stock. As integrating all these steps in an exact method may be challenging, also the development of a heuristic to do this could be beneficial in our opinion.

Second, a further step could be made in representing the journeys of passengers while minimizing delays through rolling stock rescheduling. In Chapter 4, we have represented the passenger delay by considering the number of delay minutes obtained by passengers on the respective trips. This corresponds to assuming that passengers do not change their journey considering the faced delays. A next step could be made by considering a situation in which passengers can adapt their journey according to the delays and where the delay is measured over the full journey of the passengers. To incorporate those journeys of passengers completely in the rolling stock rescheduling phase, a two-step approach is likely needed, as for example done by Kroon et al. (2015).

A third direction of further research would lie in improving and extending heuristic approaches for rolling stock scheduling and rescheduling. Our work in Chapter 3 and that of Cacchiani et al. (2019) already show the benefits that a heuristic can provide. However, compared to exact approaches, little attention has been given to heuristics for rolling stock scheduling and rescheduling so far. This while we believe that heuristics could be instrumental to achieve further integration of planning steps within railway planning and rescheduling. On the one hand, deriving new local search neighborhoods and improving the ones we have proposed could be beneficial in building more efficient heuristics for rolling stock scheduling and rescheduling. On the other hand, new neighborhoods can be derived for extensions of the rolling stock scheduling and rescheduling problems. For example, such new neighborhoods could tackle the above-mentioned problem of further integrating the planning and rescheduling of shunting processes.

References

Abbink, E., B. van den Berg, L. Kroon, and M. Salomon (2005). “Allocation of Railway Rolling Stock for Passenger Trains”. *Transportation Science* 38.1, 33–41.

Adenso-Díaz, B., M.O. González, and P. González-Torre (1999). “On-line timetable re-scheduling in regional train services”. *Transportation Research Part B: Methodological* 33.6, 387–398.

Alfieri, A., R. Groot, L. Kroon, and A. Schrijver (2006). “Efficient Circulation of Railway Rolling Stock”. *Transportation Science* 40.3, 378–391.

Barnhart, C., E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, and P.H. Vance (1998). “Branch-and-Price: Column Generation for Solving Huge Integer Programs”. *Operations Research* 46.3, 316–329.

Ben-Khedher, N., J. Kintanar, C. Queille, and W. Stripling (1998). “Schedule Optimization at SNCF: From Conception to Day of Departure”. *Interfaces* 28.1, 6–23.

Benders, J. (1962). “Partitioning procedures for solving mixed-variables programming problems”. *Numerische Mathematik* 4.1, 238–252.

Borndörfer, R., B. Grimm, M. Reuther, and T. Schlechte (2017). “Template-based re-optimization of rolling stock rotations”. *Public Transport* 9 (1), 365–383.

Borndörfer, R., B. Grimm, and T. Schlechte (2019). “Re-optimizing ICE Rotations after a Tunnel Breakdown near Rastatt”. *RailNorrköping 2019. 8th International Conference on Railway Operations Modelling and Analysis (ICROMA), Norrköping, Sweden, June 17th – 20th, 2019.* Linköping Electronic Conference Proceedings 69. Linköping University Electronic Press, Linköpings universitet, 160–168.

Borndörfer, R., J. Mehrgardt, M. Reuther, T. Schlechte, and K. Waas (2014). “Re-Optimization of Rolling Stock Rotations”. *Operations Research Proceedings 2013.* Ed. by D. Huisman, I. Louwerse, and A.P. Wagelmans. Cham: Springer International Publishing, 49–55.

Borndörfer, R., M. Reuther, T. Schlechte, K. Waas, and S. Weider (2016). “Integrated Optimization of Rolling Stock Rotations for Intercity Railways”. *Transportation Science* 50.3, 863–877.

Budai, G., G. Maróti, R. Dekker, D. Huisman, and L. Kroon (2010). “Rescheduling in passenger railways: the rolling stock rebalancing problem”. *Journal of Scheduling* 13.3, 281–297.

Cacchiani, V., A. Caprara, L. Galli, L. Kroon, G. Maróti, and P. Toth (2012). “Railway Rolling Stock Planning: Robustness Against Large Disruptions”. *Transportation Science* 46.2, 217–232.

Cacchiani, V., A. Caprara, and P. Toth (2010). “Solving a real-world train-unit assignment problem”. *Mathematical Programming* 124.1-2, 207–231.

Cacchiani, V., A. Caprara, and P. Toth (2013). “A Lagrangian heuristic for a train-unit assignment problem”. *Discrete Applied Mathematics* 161.12, 1707–1718.

Cacchiani, V., A. Caprara, and P. Toth (2019). “An Effective Peak Period Heuristic for Railway Rolling Stock Planning”. *Transportation Science* 53.3, 746–762.

Cacchiani, V., D. Huisman, M. Kidd, L. Kroon, P. Toth, L. Veelenturf, and J. Wagenaar (2014). “An overview of recovery models and algorithms for real-time railway rescheduling”. *Transportation Research Part B: Methodological* 63, 15–37.

Cadarso, L. and Á. Marín (2011). “Robust rolling stock in rapid transit networks”. *Computers & Operations Research* 38.8, 1131–1142.

Cadarso, L. and Á. Marín (2014). “Improving robustness of rolling stock circulations in rapid transit networks”. *Computers & Operations Research* 51, 146–159.

Cadarso, L., Á. Marín, and G. Maróti (2013). “Recovery of disruptions in rapid transit networks”. *Transportation Research Part E: Logistics and Transportation Review* 53, 15–33.

Caprara, A., L. Kroon, M. Monaci, M. Peeters, and P. Toth (2007). “Passenger Railway Optimization”. *Transportation*. Ed. by C. Barnhart and G. Laporte. Vol. 14. Handbooks in Operations Research and Management Science. Elsevier, 129–187.

Cordeau, J.-F., G. Stojković, F. Soumis, and J. Desrosiers (2001). “Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling”. *Transportation Science* 35.4, 375–388.

Dollevoet, T., D. Huisman, L.G. Kroon, L.P. Veelenturf, and J.C. Wagenaar (2017). “Application of an iterative framework for real-time railway rescheduling”. *Computers & Operations Research* 78, 203–217.

Fioole, P.-J., L. Kroon, G. Maróti, and A. Schrijver (2006). “A rolling stock circulation model for combining and splitting of passenger trains”. *European Journal of Operational Research* 174.2, 1281–1297.

Giacco, G.L., A. D’Ariano, and D. Pacciarelli (2014). “Rolling stock rostering optimization under maintenance constraints”. *Journal of Intelligent Transportation Systems* 18.1, 95–105.

Grimm, B., R. Borndörfer, M. Reuther, S. Schade, and T. Schlechte (2017). “A Propagation Approach to Acyclic Rolling Stock Rotation Optimization”. *7th International Conference on railway operations modelling and Analysis (RailLille 2017)*, 688–698.

Haahr, J.T. and R.M. Lusby (2017). “Integrating rolling stock scheduling with train unit shunting”. *European Journal of Operational Research* 259.2, 452–468.

Haahr, J.T., J.C. Wagenaar, L.P. Veelenturf, and L.G. Kroon (2016). “A comparison of two exact methods for passenger railway rolling stock (re)scheduling”. *Transportation Research Part E: Logistics and Transportation Review* 91, 15–32.

Hoogervorst, R., T. Dollevoet, G. Maróti, and D. Huisman (2020). "Reducing passenger delays by rolling stock rescheduling". *Transportation Science* 54.3, 762–784.

Hoogervorst, R., T. Dollevoet, G. Maróti, and D. Huisman (2021). "A Variable Neighborhood Search heuristic for rolling stock rescheduling". *EURO Journal on Transportation and Logistics* 10, 100032.

Huisman, D. (2007). "A column generation approach for the rail crew re-scheduling problem". *European Journal of Operational Research* 180.1, 163–173.

Huisman, D., L.G. Kroon, R.M. Lentink, and M.J.C.M. Vromans (2005). "Operations Research in passenger railway transportation". *Statistica Neerlandica* 59.4, 467–497.

Irnich, S. and G. Desaulniers (2005). "Shortest Path Problems with Resource Constraints". *Column Generation.* Ed. by G. Desaulniers, J. Desrosiers, and M.M. Solomon. Boston, MA: Springer US, 33–65.

Jespersen-Groth, J., D. Potthoff, J. Clausen, D. Huisman, L. Kroon, G. Maróti, and M.N. Nielsen (2009). "Disruption Management in Passenger Railway Transportation". *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems.* Ed. by R.K. Ahuja, R.H. Möhring, and C.D. Zaroliagis. Berlin, Heidelberg: Springer Berlin Heidelberg, 399–421.

Kennisinstituut voor Mobiliteitsbeleid (2017). *Mobiliteitsbeeld 2017.* Tech. rep.

Kroon, L. and D. Huisman (2011). "Algorithmic Support for Railway Disruption Management". *Transitions Towards Sustainable Mobility: New Solutions and Approaches for Sustainable Transport Systems.* Ed. by J.A. van Nunen, P. Huijbregts, and P. Rietveld. Berlin, Heidelberg: Springer Berlin Heidelberg, 193–210.

Kroon, L., G. Maróti, and L. Nielsen (2015). "Rescheduling of Railway Rolling Stock with Dynamic Passenger Flows". *Transportation Science* 49.2, 165–184.

Lin, Z. and R.S.K. Kwan (2014). "A two-phase approach for real-world train unit scheduling". *Public Transport* 6.1, 35–65.

Lin, Z. and R.S.K. Kwan (2016). "A branch-and-price approach for solving the train unit scheduling problem". *Transportation Research Part B: Methodological* 94, 97–120.

Lübbecke, M.E. and J. Desrosiers (2005). "Selected Topics in Column Generation". *Operations Research* 53.6, 1007–1023.

Lusby, R.M., J.T. Haahr, J. Larsen, and D. Pisinger (2017). "A Branch-and-Price algorithm for railway rolling stock rescheduling". *Transportation Research Part B: Methodological* 99, 228–250.

Mladenović, N. and P. Hansen (1997). "Variable neighborhood search". *Computers & Operations Research* 24.11, 1097–1100.

Nielsen, L.K. (2011). "Rolling Stock Rescheduling in Passenger Railways: Applications in short-term planning and in disruption management". PhD thesis. Erasmus Research Institute of Management.

Nielsen, L.K., L. Kroon, and G. Maróti (2012). "A rolling horizon approach for disruption management of railway rolling stock". *European Journal of Operational Research* 220.2, 496–509.

Peeters, M. and L. Kroon (2008). “Circulation of railway rolling stock: a branch-and-price approach”. *Computers & Operations Research* 35.2, 538–556.

Potthoff, D., D. Huisman, and G. Desaulniers (2010). “Column Generation with Dynamic Duty Selection for Railway Crew Rescheduling”. *Transportation Science* 44.4, 493–505.

Reuther, M. (2017). “Mathematical optimization of rolling stock rotations”. Doctoral Thesis. Berlin: Technische Universität Berlin.

Rezanova, N.J. and D.M. Ryan (2010). “The train driver recovery problem - A set partitioning based model and solution method”. *Computers & Operations Research* 37.5. Disruption Management, 845–856.

Ryan, D.M. and B. Foster (1981). “An Integer Programming Approach to Scheduling”. *Computer Scheduling of Public Transport.* Ed. by A. Wren. Amsterdam, North-Holland: Elsevier, 269–280.

Sato, K. and N. Fukumura (2011). “Real-Time Freight Train Driver Rescheduling during Disruption”. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* E94.A.6, 1222–1229.

Schrijver, A. (1993). “Minimum circulation of railway stock”. *CWI Quarterly* 6.3, 205–217.

Thorlacius, P., J. Larsen, and M. Laumanns (2015). “An integrated rolling stock planning model for the Copenhagen suburban passenger railway”. *Journal of Rail Transport Planning & Management* 5.4, 240–262.

Van Wezel, W. and J. Riezebos (2011). “Case Study: Advanced Decision Support for Train Shunting Scheduling”. *Behavioral Operations in Planning and Scheduling.* Ed. by J.C. Fransoo, T. Waefler, and J.R. Wilson. Berlin, Heidelberg: Springer Berlin Heidelberg, 413–430.

Veelenturf, L.P., M.P. Kidd, V. Cacchiani, L.G. Kroon, and P. Toth (2016). “A Railway Timetable Rescheduling Approach for Handling Large-Scale Disruptions”. *Transportation Science* 50.3, 841–862.

Veelenturf, L.P., D. Potthoff, D. Huisman, and L.G. Kroon (2012). “Railway crew rescheduling with retiming”. *Transportation Research Part C: Emerging Technologies* 20.1, 95–110.

Wagenaar, J.C., L.G. Kroon, and M. Schmidt (2017a). “Maintenance Appointments in Railway Rolling Stock Rescheduling”. *Transportation Science* 51.4, 1138–1160.

Wagenaar, J.C., L. Kroon, and I. Fragkos (2017b). “Rolling stock rescheduling in passenger railway transportation using dead-heading trips and adjusted passenger demand”. *Transportation Research Part B: Methodological* 101, 140–161.

Zhong, Q., R.M. Lusby, J. Larsen, Y. Zhang, and Q. Peng (2019). “Rolling stock scheduling with maintenance requirements at the Chinese High-Speed Railway”. *Transportation Research Part B: Methodological* 126, 24–44.

Nederlandse Samenvatting (Summary in Dutch)

Openbaar vervoer speelt een belangrijke rol binnen het Nederlandse mobiliteitssysteem, waarbij het een tweede plaats inneemt, na de auto, wat betreft het gemiddeld aantal afgelegde kilometers per persoon. Van de kilometers in het openbaar vervoer wordt ongeveer driekwart per trein afgelegd. De grootste treinvervoerder is Nederlandse Spoorwegen (NS), die verantwoordelijk is voor het rijden van treinen op het zogenaamde hoofdrailnet. Om deze diensten uit te kunnen voeren bezit NS een vloot van treinstellen. Aangezien de aankoop en het bezit van treinstellen—ook wel materieel genoemd—gewoonlijk de grootste kostenpost is voor een treinvervoerder, is het noodzakelijk om deze treinstellen efficiënt te gebruiken. Aan de ene kant betekent dit dat de treinstellen goed ingepland moeten worden om te zorgen dat er een goede balans wordt gevonden tussen servicekwaliteit en operationele kosten. Aan de andere kant is ook het bijsturen van treinstellen belangrijk, om zo verstoringen binnen het systeem goed op te kunnen vangen met het beschikbare aantal treinstellen.

De planning en bijsturing van materieel vinden plaats als onderdeel van respectievelijk een plannings- en bijsturingsproces bij treinvervoerders. In beide processen wordt gewoonlijk een stapsgewijze aanpak gevolgd, waarbij de dienstregeling wordt vastgelegd voordat wordt begonnen met de planning of bijsturing van materieel. In materieelplanning of -bijsturing focussen we ons dan op het toewijzen van materieel aan de ritten in de dienstregeling. Belangrijk is dat treinstellen gekoppeld kunnen worden tot materieelsamenstellingen om zo meer plek te kunnen bieden aan passagiers. Deze samenstellingen kunnen gewijzigd worden tussen de ritten, bijvoorbeeld wanneer de vraag op de volgende ritten die deze trein uitvoert groter is dan op de voorgaande ritten. In de toewijzing van materieel willen we ervoor zorgen dat we toegelaten samenstellingen vinden voor alle ritten en de samenstellingen ook op toegelaten manieren veranderen tussen ritten. Hierbij willen we er aan de ene kant voor zorgen dat de samenstellingen lang genoeg zijn om de passagiersvraag op te kunnen vangen, maar willen we aan de andere kant, bijvoorbeeld, ook voorkomen dat we treinstellen onnodig gebruiken of dat we te vaak de samenstellingen veranderen. De output van dit proces wordt een materieelomloop genoemd. Deze materieelomloop dient dan weer als input voor de personeelsplanning of -bijsturing.

In dit proefschrift kijken we naar het verbeteren van zowel de planning als bijsturing van materieel. Aan de ene kant focussen we ons hierbij op het analyseren en verbeteren van de oplosmethoden voor deze problemen. Aan de andere kant kijken we ook naar het verder integreren van materieelbijsturing binnen het bijsturingsproces van een treinvervoerder. Zo heeft de gevonden materieelomloop bijvoorbeeld een impact op de processen die plaatsvinden op de stations, specifiek op de rangeeracties die nodig zijn om wisselingen in de samenstellingen van treinen mogelijk te maken. Verdere integratie tussen materieelbijsturing en de andere stappen in het bijsturingsproces kan zo leiden tot een betere oplossing voor het complete probleem en uiteindelijk tot een betere service voor passagiers bij verstoringen.

In hoofdstuk 2 focussen we ons op de materieelplanning. Hier categoriseren we allereerst de modellen die voor dit probleem zijn voorgesteld in de literatuur op basis van de details van het materieelplanningsprobleem dat ze bekijken. Zo verschillen de modellen vaak door verschillen in de manier waarop de bestudeerde treinvervoerders opereren, bijvoorbeeld in hoeverre ze materieelsamenstellingen meenemen in de planning van materieel en hoeveel vrijheid ze toelaten in het bepalen van keringen tussen ritten. Hierna zoomen we in op twee veelgebruikte modellen: het compositiemodel van Fioole e.a. (2006) dat is ontworpen voor de setting van NS en het hypergraafmodel van Borndörfer e.a. (2016) dat zich focust op de setting van DB Fernverkehr. Een analytische vergelijking laat zien dat beide wiskundige formuleringen even sterk zijn voor de setting van NS, wat betekent dat ze tot dezelfde lineaire-programmeringsrelaxatie leiden voor het materieelplanningsprobleem van NS. Een numerieke vergelijking op instanties van NS laat echter zien dat het hypergraafmodel, wat toelaat om meer details mee te nemen in de modellering, op dit simpelere probleem tot langere rekentijden leidt.

We kijken naar een heuristiek voor materieelbijsturing in hoofdstuk 3. Deze heuristiek is gemotiveerd door de observatie dat het meenemen van alle praktische details, zoals meegenomen door materieelbijstuurders, vaak lastig is in exacte modellen voor materieelbijsturing en tot lange rekentijden leidt. In dit hoofdstuk introduceren we daarom een Variable Neighborhood Search (VNS) heuristiek voor dit probleem. Onderdeel van deze heuristiek zijn drie nieuwe lokale zoekomgevingen voor materieelbijsturing die het mogelijk maken om een bestaande oplossing te verbeteren. We laten ook zien dat de heuristiek uitgebreid kan worden naar omvangrijkere materieelbijsturingsproblemen door het introduceren van een extra zoekomgeving voor een situatie waarin de keringen tussen treinen op eindstations flexibel gekozen kunnen worden. Onze resultaten laten zien dat deze methode over het algemeen snel goede oplossingen kan vinden, wat deze methode goed bruikbaar maakt voor materieelbijstuurders. Daarnaast laten we ook zien dat we voor een aantal instanties met flexibele keringen betere oplossingen kunnen vinden dan een exacte oplosmethode gegeven een maximale rekentijd van één minuut.

In hoofdstukken 4 en 5 kijken we naar extensies van het materieelbijsturingsprobleem. Hierbij kijken we in hoofdstuk 4 hoe materieelbijsturing kan leiden tot een vermindering van de vertragingen die passagiers ervaren wanneer een bronvertraging zich voordoet in het spoornetwerk. Vertragingen volgen vaak de treinstellen, aangezien een trein die te laat aankomt op een station ook weer te laat vertrekt wanneer er niet genoeg buffer tussen aankomst en vertrek zit. Het bijsturen van materieel leidt tot een andere toewijzing van de treinstellen aan de ritten en op die manier ook tot andere vertragingen op de ritten. We stellen twee modellen voor om het materieel zo bij te sturen dat de vertragingen worden geminimaliseerd, maar tegelijkertijd ook de andere doelstellingen voor materieelbijsturing worden meegenomen. Een van deze modellen is gebaseerd op een stroom in een netwerk (network-flow), terwijl de andere een pad-gebaseerde aanpak is. Onze resultaten laten zien dat de network-flow aanpak gemiddeld sneller opgelost kan worden dan de pad-gebaseerde aanpak. Daarnaast tonen we aan dat significante verminderingen in vertragingen gerealiseerd kunnen worden, zeker wanneer we meer vrijheid toelaten in de keringen tussen ritten op eindstations.

In hoofdstuk 5 kijken we naar het integreren van materieelbijsturing met het bijsturen van de RET-machinisten op stations. Deze RET-machinisten zijn verantwoordelijk voor het verplaatsen van treinstellen tussen het station en het opstelterrein van een station, waar treinstellen worden geparkeerd tot ze weer opnieuw worden gebruikt. De taken die door deze RET-machinisten worden uitgevoerd worden voor een groot deel bepaald door de rangeeracties die voortkomen uit het veranderen van materieelsamenstellingen. Dit betekent dat het bijsturen van materieel, en dus van de rangeeracties die uitgevoerd moeten worden, ook tot veranderingen in de taken van RET-machinisten leidt. Traditioneel worden materieel en RET-machinisten op een sequentiële manier bijgestuurd. Dit kan tot problemen leiden wanneer er niet genoeg RET-machinisten aanwezig zijn om de rangeeracties in het materieelplan uit te voeren. Daarom stellen we voor om beide problemen integraal op te lossen, waarvoor we twee oplosmethoden voorstellen: een aanpak gebaseerd op Benders-decompositie en een network-flow aanpak. Onze resultaten tonen aan dat dit geïntegreerde probleem goed opgelost kan worden en dat er vergeleken met een sequentiële aanpak een significant aantal conflicten worden voorkomen. Daarnaast zien we dat de network-flow aanpak over het algemeen iets beter presteert, maar dat de relatieve prestatie van de aanpakken verschilt per instantie.

Het onderzoek in dit proefschrift kan op meerdere manieren waarde hebben voor treinvervoerders. Aan de ene kant laat ons werk zien hoe verdere integratie van materieelbijsturing in het bijsturingsproces voordelen kan opleveren, bijvoorbeeld in het voorkomen van vertragingen voor passagiers en in het effectief bijsturen van RET-machinisten. De methodes die we voor deze problemen hebben ontwikkeld kunnen hierbij als basis dienen voor beslissingsondersteuning bij treinvervoerders. Daarnaast kan ons werk ook handvatten bieden voor het selecteren van oplosmethoden voor materieelplanning en -bijsturing. Hierbij geloven we dat de heuristiek die we voorstellen zeer geschikt is voor gebruik door treinvervoerders, aangezien deze makkelijk uitbreidbaar is en stappen neemt die relatief gemakkelijk uit te leggen zijn aan materieelbijstuurders. De vergelijking van modellen voor materieelplanning kan bovendien door treinvervoerders gebruikt worden om een model te selecteren dat aansluit bij het materieelplanningsprobleem dat ze willen oplossen en geeft inzicht in de relatieve prestatie van twee veelgebruikte modellen voor materieelplanning.

Concluderend laat ons werk zien dat er nog steeds stappen kunnen worden gezet in het verbeteren van materieelplanning en -bijsturing. Hierbij sluit ons werk aan bij een algemene trend van integratie van planningsstappen binnen het openbaar vervoer en het steeds beter worden van oplosmethoden voor deze problemen. Alles bij elkaar, maakt dit het mogelijk om een steeds aantrekkelijkere service te kunnen bieden aan passagiers in het openbaar vervoer en draagt het zo bij om meer mensen te laten kiezen voor deze milieuvriendelijke vorm van vervoer.

About the author

In his research, Rowan mainly focuses on problems in transportation and logistics, but his research interests also extend to other applied problems with a clear societal relevance. A characteristic of Rowan's work is that he tries to bridge the gap between theory and practice, where he uses a range of tools from mathematical programming to efficiently solve problems coming from practice. Rowan's work was published in journals such as Transportation Science and Information Sciences. Moreover, he was awarded second place in the 2020 INFORMS RAS student paper award at the INFORMS annual meeting.

Rowan holds a master's degree in Econometrics and Management Science from Erasmus University in Rotterdam. He worked as a PhD candidate at the Econometric Institute, Erasmus University Rotterdam, and was part of the PhD program of the Erasmus Research Institute of Management (ERIM). During his PhD, he regularly visited the department of Process quality and Innovation (π) of Netherlands Railways. Moreover, Rowan spent three-and-a-half months at the Zuse Institute Berlin (ZIB) as part of a research visit to Prof. Ralf Borndörfer. He is currently working as a Post-Doctoral researcher at the Technical University of Denmark (DTU) within the Management Science department.

Portfolio

Publications in International Journals

Hoogervorst, R., T. Dollevoet, G. Maróti, and D. Huisman (2020). “Reducing passenger delays by rolling stock rescheduling”. *Transportation Science* 54.3, 762–784.

Hoogervorst, R., T. Dollevoet, G. Maróti, and D. Huisman (2021). “A Variable Neighborhood Search heuristic for rolling stock rescheduling”. *EURO Journal on Transportation and Logistics* 10, 100032.

Hoogervorst, R., Y. Zhang, G. Tillem, Z. Erkin, and S. Verwer (2019). “Solving bin-packing problems under privacy preservation: Possibilities and trade-offs”. *Information Sciences* 500, 203–216.

Peer-reviewed Conference Proceedings

Hoogervorst, R., E. Essink, W. Jansen, M. van den Helder, K. Schouten, F. Frasincar, and M. Taboada (2016). “Aspect-Based Sentiment Analysis on the Web Using Rhetorical Structure Theory”. *Web Engineering*. Ed. by A. Bozzon, P. Cudre-Maroux, and C. Pautasso. Cham: Springer International Publishing, 317–334.

Teaching

Introduction to Programming (Minor and Pre-Master), Erasmus School of Economics, 2020, Lecturer and Course Coordinator

MOOC Econometrics: Methods and Applications, Erasmus School of Economics, 2017-2020, Assistant

Combinatorial Optimization, Erasmus School of Economics, 2016, 2017, 2018, 2019, Tutorial Lecturer and Assistant to the Lecturer

Bachelor’s Thesis Econometrics and Operational Research, 2017, 2018, 2020, Thesis Supervisor

PhD Courses
Convex Analysis for Optimization
Networks and Polyhedra
Networks and Semidefinite Programming
Randomized Algorithms
Algorithms and Complexity
Interior Point Methods
Integer Programming Methods
Robust Optimization (Attendance only)
2018 School on Column Generation
Reading Group Constraint Programming
Scientific Integrity
Publishing Strategy
English (CPE certificate)

Conferences Attended
Lunteren Conference 2021, Virtual
INFORMS Annual Meeting 2020, Virtual
OR 2019, Dresden, Germany
EURO 2019, Dublin, Ireland
CASPT 2018, Brisbane, Australia
IFORS 2017, Quebec City, Canada
Lunteren Conference 2017, 2018, 2019, 2020, Lunteren, The Netherlands
EURO 2016, Poznań, Poland

The ERIM PhD Series

The ERIM PhD Series contains PhD dissertations in the field of Research in Management defended at Erasmus University Rotterdam and supervised by senior researchers affiliated to the Erasmus Research Institute of Management (ERIM). All dissertations in the ERIM PhD Series are available in full text through the ERIM Electronic Series Portal: repub.eur.nl/pub. ERIM is the joint research institute of the Rotterdam School of Management (RSM) and the Erasmus School of Economics (ESE) at the Erasmus University Rotterdam (EUR).

Dissertations in the last four years

Ahmadi, S., *A motivational perspective to decision-making and behavior in organizations*, Promotors: Prof. J.J.P. Jansen & Dr. T.J.M. Mom, EPS-2019-477-S&E, https://repub.eur.nl/pub/116727

Akemu, O., *Corporate Responses to Social Issues: Essays in Social Entrepreneurship and Corporate Social Responsibility*, Promotors: Prof. G.M. Whiteman & Dr. S.P. Kennedy, EPS-2017-392-ORG, https://repub.eur.nl/pub/95768

Albuquerque de Sousa, J.A., *International stock markets: Essays on the determinants and consequences of financial market development*, Promotors: Prof. M.A. van Dijk & Prof. P.A.G. van Bergeijk, EPS-2019-465-F&A, https://repub.eur.nl/pub/115988

Alserda, G.A.G., *Choices in Pension Management*, Promotors: Prof. S.G. van der Lecq & Dr. O.W. Steenbeek, EPS-2017-432-F&A, https://repub.eur.nl/pub/103496

Anantavrasilp, S., *Essays on Ownership Structures, Corporate Finance Policies and Financial Reporting Decisions*, Promotors: Prof. A. de Jong & Prof. P.G.J. Roosenboom, EPS-2021-516-F&E, https://repub.eur.nl/pub/134947

Arampatzi, E., *Subjective Well-Being in Times of Crises: Evidence on the Wider Impact of Economic Crises and Turmoil on Subjective Well-Being*, Promotors: Prof. H.R. Commandeur, Prof. F. van Oort & Dr. M.J. Burger, EPS-2018-459-S&E, https://repub.eur.nl/pub/111830

Arslan, A.M., *Operational Strategies for On-demand Delivery Services*, Promotors: Prof. R.A. Zuidwijk & Dr. N.A.H. Agatz, EPS-2019-481-LIS, https://repub.eur.nl/pub/126463

Aydin, Z., *Mobile Consumers and Applications: Essays on Mobile Marketing*, Promotors: Prof. G.H. van Bruggen & Dr. B. Ataman, EPS-2021-519-MKT, https://repub.eur.nl/pub/135352

Azadeh, K., *Robotized Warehouses: Design and Performance Analysis*, Promotors: Prof. dr. ir M.B.M. de Koster & Prof. D. Roy, EPS-2021-515-LIS, https://repub.eur.nl/pub/135208

Avci, E., *Surveillance of Complex Auction Markets: a Market Policy Analytics Approach*, Promotors: Prof. W. Ketter, Prof. H.W.G.M. van Heck & Prof. D.W. Bunn, EPS-2018-426-LIS, https://repub.eur.nl/pub/106286

Balen, T.H. van, *Challenges of Early Stage Entrepreneurs: the Roles of Vision Communication and Team Membership Change*, Promotors: Prof. J.C.M. van den Ende & Dr. M. Tarakci, EPS-2019-468-LIS, https://repub.eur.nl/pub/115654

Bansraj, S.C., *The Principles of Private Equity: Ownership and Acquisitions*, Promotors: Prof. J.T.J Smit & Dr. V. Volosovych, EPS-2020-507-F&A, https://repub.eur.nl/pub/132329

Bavato, D., *With New Eyes: The recognition of novelty and novel ideas*, Promotors: Prof. D.A. Stam & Dr. S. Tasselli, EPS-2020-500-LIS, https://repub.eur.nl/pub/134264

Bernoster, I., *Essays at the Intersection of Psychology, Biology, and Entrepreneurship*, Promotors: Prof. A.R. Thurik, Prof. I.H.A. Franken & Prof. P.J.F. Groenen, EPS-2018-463-S&E, https://repub.eur.nl/pub/113907

Blagoeva, R.R., *The Hard Power Of Soft Power: A behavioral strategy perspective on how power, reputation, and status affect firms*, Promotors: Prof. J.J.P. Jansen & Prof. T.J.M. Mom, EPS-2020-495-S&E, https://repub.eur.nl/pub/127681

Bouman, P., *Passengers, Crowding and Complexity: Models for Passenger Oriented Public Transport*, Promotors: Prof. L.G. Kroon, Prof. A. Schöbel & Prof. P.H.M. Vervest, EPS-2017-420-LIS, https://repub.eur.nl/pub/100767

Breugem, T., *Crew Planning at Netherlands Railways: Improving Fairness, Attractiveness, and Efficiency*, Promotors: Prof. D. Huisman & Dr. T.A.B. Dollevoet, EPS-2020-494-LIS, https://repub.eur.nl/pub/124016

Bunderen, L. van, *Tug-of-War: Why and when teams get embroiled in power struggles*, Promotors: Prof. D.L. van Knippenberg & Dr. L. Greer, EPS-2018-446-ORG, https://repub.eur.nl/pub/105346

Burg, G.J.J. van den, *Algorithms for Multiclass Classification and Regularized Regression*, Promotors: Prof. P.J.F. Groenen & Dr. A. Alfons, EPS-2018-442-MKT, https://repub.eur.nl/pub/103929

Chammas, G., *Portfolio concentration*, Promotor: Prof. J. Spronk, EPS-2017-410-F&E, https://repub.eur.nl/pub/94975

Chan, H.Y., *Decoding the consumer's brain: Neural representations of consumer experience*, Promotors: Prof. A. Smidts & Dr. M.A.S. Boksem, EPS-2019-493-MKT, https://repub.eur.nl/pub/124931

Couwenberg, L., *Context dependent valuation: A neuroscientific perspective on consumer decision-making*, Promotors: Prof. A. Smit, Prof. A.G. Sanfrey & Dr. M.A.S. Boksem, EPS-2020-505-MKT, https://repub.eur.nl/pub/129601

Dalmeijer, K., *Time Window Assignment in Distribution Networks*, Promotors: Prof. A.P.M. Wagelmans & Dr. R. Spliet, EPS-2019-486-LIS, https://repub.eur.nl/pub/120773

Dennerlein, T., *Empowering Leadership and Employees' Achievement Motivations: the Role of Self-Efficacy and Goal Orientations in the Empowering Leadership Process*, Promotors: Prof. D.L. van Knippenberg & Dr. J. Dietz, EPS-2017-414-ORG, https://repub.eur.nl/pub/98438

Dolgova, E., *On Getting Along and Getting Ahead: How Personality Affects Social Network Dynamics*, Promotors: Prof. P.P.M.A.R. Heugens & Prof. M.C. Schippers, EPS-2019-455-S&E, https://repub.eur.nl/pub/119150

Duijzer, L.E., *Mathematical Optimization in Vaccine Allocation*, Promotors: Prof. R. Dekker & Dr. W.L. van Jaarsveld, EPS-2017-430-LIS, https://repub.eur.nl/pub/101487

Fasaei, H., *Changing the Narrative: The Behavioral Effects of Social Evaluations on the Decision Making of Organizations*, Promotors: Prof. J.J.P. Jansen, Prof. T.J.M. Mom & Dr. M.P. Tempelaar, EPS-2020-492-S&E, https://repub.eur.nl/pub/129598

Eijlers, E., *Emotional Experience and Advertising Effectiveness: on the use of EEG in marketing*, Promotors: Prof. A. Smidts & Prof. M.A.S. Boksem, EPS-2019-487-MKT, https://repub.eur.nl/pub/124053

El Nayal, O.S.A.N., *Firms and the State: An Examination of Corporate Political Activity and the Business-Government Interface*, Promotors: Prof. J. van Oosterhout & Dr. M. van Essen, EPS-2018-469-S&E, https://repub.eur.nl/pub/114683

Feng, Y., *The Effectiveness of Corporate Governance Mechanisms and Leadership Structure: Impacts on strategic change and firm performance*, Promotors: Prof. F.A.J. van den Bosch, Prof. H.W. Volberda & Dr. J.S. Sidhu, EPS-2017-389-S&E, https://repub.eur.nl/pub/98470

Frick, T.W., *The Implications of Advertising Personalization for Firms, Consumer, and Ad Platfroms*, Promotors: Prof. T. Li & Prof. H.W.G.M. van Heck, EPS-2018-452-LIS, https://repub.eur.nl/pub/110314

Fytraki, A.T., *Behavioral Effects in Consumer Evaluations of Recommendation Systems*, Promotors: Prof. B.G.C. Dellaert & Prof. T. Li, EPS-2018-427-MKT, https://repub.eur.nl/pub/110457

Gai, J., *Contextualized Consumers: Theories and Evidence on Consumer Ethics, Product Recommendations, and Self-Control*, Promotors: Prof. S. Puntoni & Prof. S.T.L. Sweldens, EPS-2020-498-MKT, https://repub.eur.nl/pub/127680

Ghazizadeh, P., *Empirical Studies on the Role of Financial Information in Asset and Capital Markets*, Promotors: Prof. A. de Jong & Prof. E. Peek, EPS-2019-470-F&A, https://repub.eur.nl/pub/114023

Giurge, L., *A Test of Time; A temporal and dynamic approach to power and ethics*, Promotors: Prof. M.H. van Dijke & Prof. D. De Cremer, EPS-2017-412-ORG, https://repub.eur.nl/pub/98451

Gobena, L., *Towards Integrating Antecedents of Voluntary Tax Compliance*, Promotors: Prof. M.H. van Dijke & Dr. P. Verboon, EPS-2017-436-ORG, https://repub.eur.nl/pub/103276

Groot, W.A., *Assessing Asset Pricing Anomalies*, Promotors: Prof. M.J.C.M. Verbeek & Prof. J.H. van Binsbergen, EPS-2017-437-F&A, https://repub.eur.nl/pub/103490

Hanselaar, R.M., *Raising Capital: On pricing, liquidity and incentives*, Promotors: Prof. M.A. van Dijk & Prof. P.G.J. Roosenboom, EPS-2018-429-F&A, https://repub.eur.nl/pub/113274

Harms, J.A., *Essays on the Behavioral Economics of Social Preferences and Bounded Rationality*, Promotors: Prof. H.R. Commandeur & Dr. K.E.H. Maas, EPS-2018-457-S&E, https://repub.eur.nl/pub/108831

Hendriks, G., *Multinational Enterprises and Limits to International Growth: Links between Domestic and Foreign Activities in a Firm's Portfolio*, Promotors: Prof. P.P.M.A.R. Heugens & Dr. A.H.L. Slangen, EPS-2019-464-S&E, https://repub.eur.nl/pub/114981

Hengelaar, G.A., *The Proactive Incumbent: Holy grail or hidden gem? Investigating whether the Dutch electricity sector can overcome the incumbent's curse and lead the sustainability transition*, Promotors: Prof. R.J.M. van Tulder & Dr. K. Dittrich, EPS-2018-438-ORG, https://repub.eur.nl/pub/102953

Jacobs, B.J.D., *Marketing Analytics for High-Dimensional Assortments*, Promotors: Prof. A.C.D. Donkers & Prof. D. Fok, EPS-2017-445-MKT, https://repub.eur.nl/pub/103497

Jia, F., *The Value of Happiness in Entrepreneurship*, Promotors: Prof. D.L. van Knippenberg & Dr. Y. Zhang, EPS-2019-479-ORG, https://repub.eur.nl/pub/115990

Kahlen, M.T., *Virtual Power Plants of Electric Vehicles in Sustainable Smart Electricity Markets*, Promotors: Prof. W. Ketter & Prof. A. Gupta, EPS-2017-431-LIS, https://repub.eur.nl/pub/100844

Kampen, S. van, *The Cross-sectional and Time-series Dynamics of Corporate Finance: Empirical evidence from financially constrained firms*, Promotors: Prof. L. Norden & Prof. P.G.J. Roosenboom, EPS-2018-440-F&A, https://repub.eur.nl/pub/105245

Karali, E., *Investigating Routines and Dynamic Capabilities for Change and Innovation*, Promotors: Prof. H.W. Volberda, Prof. H.R. Commandeur & Dr. J.S. Sidhu, EPS-2018-454-S&E, https://repub.eur.nl/pub/106274

Keko, E., *Essays on Innovation Generation in Incumbent Firms*, Promotors: Prof. S. Stremersch & Dr. N.M.A. Camacho, EPS-2017-419-MKT, https://repub.eur.nl/pub/100841

Kerkkamp, R.B.O., *Optimisation Models for Supply Chain Coordination under Information Asymmetry*, Promotors: Prof. A.P.M. Wagelmans & Dr. W. van den Heuvel, EPS-2018-462-LIS, https://repub.eur.nl/pub/109770

Khattab, J., *Make Minorities Great Again: a contribution to workplace equity by identifying and addressing constraints and privileges*, Promotors: Prof. D.L. van Knippenberg & Dr. A. Nederveen Pieterse, EPS-2017-421-ORG, https://repub.eur.nl/pub/99311

Kim, T.Y., *Data-driven Warehouse Management in Global Supply Chains*, Promotors: Prof. R. Dekker & Dr. C. Heij, EPS-2018-449-LIS, https://repub.eur.nl/pub/109103

Klitsie, E.J., *Strategic Renewal in Institutional Contexts: The paradox of embedded agency*, Promotors: Prof. H.W. Volberda & Dr. S. Ansari, EPS-2018-444-S&E, https://repub.eur.nl/pub/106275

Koolen, D., *Market Risks and Strategies in Power Systems Integrating Renewable Energy*, Promotors: Prof. W. Ketter & Prof. R. Huisman, EPS-2019-467-LIS, https://repub.eur.nl/pub/115655

Kong, L., *Essays on Financial Coordination*, Promotor: Prof. M.J.C.M. Verbeek, Dr. D.G.J. Bongaerts & Dr. M.A. van Achter. EPS-2019-433-F&A, https://repub.eur.nl/pub/114516

Korman, B., *Leader-Subordinate Relations: The Good, the Bad and the Paradoxical*, Promotors: S.R. Giessner & Prof. C. Tröster, EPS-2021-511-ORG, https://repub.eur.nl/pub/135365

Kyosev, G.S., *Essays on Factor Investing*, Promotors: Prof. M.J.C.M. Verbeek & Dr. J.J. Huij, EPS-2019-474-F&A, https://repub.eur.nl/pub/116463

Lamballais Tessensohn, T., *Optimizing the Performance of Robotic Mobile Fulfillment Systems*, Promotors: Prof. M.B.M de Koster, Prof. R. Dekker & Dr. D. Roy, EPS-2019-411-LIS, https://repub.eur.nl/pub/116477

Leung, W.L., *How Technology Shapes Consumption: Implications for Identity and Judgement*, Promotors: Prof. S. Puntoni & Dr. G. Paolacci, EPS-2019-485-MKT, https://repub.eur.nl/pub/117432

Li, W., *Competition in the Retail Market of Consumer Packaged Goods*, Promotors: Prof. D.Fok & Prof. Ph.H.B.F. Franses, EPS-2021-503-MKT, https://repub.eur.nl/pub/134873

Li, X., *Dynamic Decision Making under Supply Chain Competition*, Promotors: Prof. M.B.M de Koster, Prof. R. Dekker & Prof. R. Zuidwijk, EPS-2018-466-LIS, https://repub.eur.nl/pub/114028

Liu, N., *Behavioral Biases in Interpersonal Contexts*, Promotors: Prof. A. Baillon & Prof. H. Bleichrodt, EPS-2017-408-MKT, https://repub.eur.nl/pub/95487

Maas, A.J.J., *Organizations and their external context: Impressions across time and space*, Promotors: Prof. P.P.M.A.R. Heugens & Prof. T.H. Reus, EPS-2019-478-S&E, https://repub.eur.nl/pub/116480

Maira, E., *Consumers and Producers*, Promotors: Prof. S. Puntoni & Prof. C. Fuchs, EPS-2018-439-MKT, https://repub.eur.nl/pub/104387

Manouchehrabadi, B., *Information, Communication and Organizational Behavior*, Promotors: Prof. G.W.J. Hendrikse & Dr. O.H. Swank, EPS-2020-502-ORG, https://repub.eur.nl/pub/132185

Matawlie, N., *Through Mind and Behaviour to Financial Decisions*, Promotors: Prof. J.T.J. Smit & Prof. P. Verwijmeren, EPS-2020-501-F&A, https://repub.eur.nl/pub/134265

Mirzaei, M., *Advanced Storage and Retrieval Policies in Automated Warehouses*, Promotors: Prof. M.B.M. de Koster & Dr. N. Zaerpour, EPS-2020-490-LIS, https://repub.eur.nl/pub/125975

Nair, K.P., *Strengthening Corporate Leadership Research: The relevance of biological explanations*, Promotors: Prof. J. van Oosterhout & Prof. P.P.M.A.R. Heugens, EPS-2019-480-S&E, https://repub.eur.nl/pub/120023

Nullmeier, F.M.E., *Effective contracting of uncertain performance outcomes: Allocating responsibility for performance outcomes to align goals across supply chain actors*, Promotors: Prof. J.Y.F. Wynstra & Prof. E.M. van Raaij, EPS-2019-484-LIS, `https://repub.eur.nl/pub/118723`

Okbay, A., *Essays on Genetics and the Social Sciences*, Promotors: Prof. A.R. Thurik, Prof. Ph.D. Koellinger & Prof. P.J.F. Groenen, EPS-2017-413-S&E, `https://repub.eur.nl/pub/95489`

Peng, X., *Innovation, Member Sorting, and Evaluation of Agricultural Cooperatives*, Promotor: Prof. G.W.J. Hendriks, EPS-2017-409-ORG, `https://repub.eur.nl/pub/94976`

Petruchenya, A., *Essays on Cooperatives: Emergence, Retained Earnings, and Market Shares*, Promotors: Prof. G.W.J. Hendriks & Dr. Y. Zhang, EPS-2018-447-ORG, `https://repub.eur.nl/pub/105243`

Plessis, C. du, *Influencers: The Role of Social Influence in Marketing*, Promotors: Prof. S. Puntoni & Prof. S.T.L.R. Sweldens, EPS-2017-425-MKT, `https://repub.eur.nl/pub/103265`

Pocock, M., *Status Inequalities in Business Exchange Relations in Luxury Markets*, Promotors: Prof. C.B.M. van Riel & Dr. G.A.J.M. Berens, EPS-2017-346-ORG, `https://repub.eur.nl/pub/98647`

Polinder, G.J., *New Models and Applications for Railway Timetabling*, Promotors: Prof. D. Huisman & Dr. M.E. Schmidt, EPS-2020-514-LIS, `https://repub.eur.nl/pub/134600`

Pozharliev, R., *Social Neuromarketing: The role of social context in measuring advertising effectiveness*, Promotors: Prof. W.J.M.I. Verbeke & Prof. J.W. van Strien, EPS-2017-402-MKT, `https://repub.eur.nl/pub/95528`

Qian, Z., *Time-Varying Integration and Portfolio Choices in the European Capital Markets*, Promotors: Prof. W.F.C. Verschoor, Prof. R.C.J. Zwinkels & Prof. M.A. Pieterse-Bloem, EPS-2020-488-F&A, `https://repub.eur.nl/pub/124984`

Reh, S.G., *A Temporal Perspective on Social Comparisons in Organizations*, Promotors: Prof. S.R. Giessner, Prof. N. van Quaquebeke & Dr. C. Troster, EPS-2018-471-ORG, `https://repub.eur.nl/pub/114522`

Riessen, B. van, *Optimal Transportation Plans and Portfolios for Synchromodal Container Networks*, Promotors: Prof. R. Dekker & Prof. R.R. Negenborn, EPS-2018-448-LIS, `https://repub.eur.nl/pub/105248`

Romochkina, I.V., *When Interests Collide: Understanding and modeling interests alignment using fair pricing in the context of interorganizational information systems*, Promotors: Prof. R.A. Zuidwijk & Prof. P.J. van Baalen, EPS-2020-451-LIS, `https://repub.eur.nl/pub/127244`

Schie, R.J.G. van, *Planning for Retirement: Save More or Retire Later?*, Promotors: Prof. B.G.C. Dellaert & Prof. A.C.D. Donkers, EOS-2017-415-MKT, `https://repub.eur.nl/pub/100846`

Schneidmüller, T., *Engaging with Emerging Technologies: Socio-cognitive foundations of incumbent response*, Promotors: Prof. H. Volberda & Dr. S.M. Ansari, EPS-2020-509-S&E, `https://repub.eur.nl/pub/131124`

Schouten, K.I.M., *Semantics-driven Aspect-based Sentiment Analysis*, Promotors: Prof. F.M.G. de Jong, Prof. R. Dekker & Dr. F. Frasincar, EPS-2018-453-LIS, `https://repub.eur.nl/pub/112161`

Sihag, V., *The Effectiveness of Organizational Controls: A meta-analytic review and an investigation in NPD outsourcing*, Promotors: Prof. J.C.M. van den Ende & Dr. S.A. Rijsdijk, EPS-2019-476-LIS, `https://repub.eur.nl/pub/115931`

Slob, E., *Integrating Genetics into Economics*, Promotors: Prof. A.R. Thurik, Prof. P.J.F. Groenen & Dr. C.A. Rietveld, EPS-2021-517-S&E, `https://repub.eur.nl/pub/135159`

Smolka, K.M., *Essays on Entrepreneurial Cognition, Institution Building and Industry Emergence*, Promotors: P.P.M.A.R. Heugens & Prof. J.P. Cornelissen, EPS-2019-483-S&E, `https://repub.eur.nl/pub/118760`

Straeter, L.M., *Interpersonal Consumer Decision Making*, Promotors: Prof. S.M.J. van Osselaer & Dr. I.E. de Hooge, EPS-2017-423-MKT, `https://repub.eur.nl/pub/100819`

Stuppy, A., *Essays on Product Quality*, Promotors: Prof. S.M.J. van Osselaer & Dr. N.L. Mead, EPS-2018-461-MKT, `https://repub.eur.nl/pub/111375`

Subaşi, B., *Demographic Dissimilarity, Information Access and Individual Performance*, Promotors: Prof. D.L. van Knippenberg & Dr. W.P. van Ginkel, EPS-2017-422-ORG, `https://repub.eur.nl/pub/103495`

Suurmond, R., *In Pursuit of Supplier Knowledge: Leveraging capabilities and dividing responsibilities in product and service contexts*, Promotors: Prof. J.Y.F Wynstra & Prof. J. Dul, EPS-2018-475-LIS, `https://repub.eur.nl/pub/115138`

Toxopeus, H.S., *Financing sustainable innovation: From a principal-agent to a collective action perspective*, Promotors: Prof. H.R. Commandeur & Dr. K.E.H. Maas, EPS-2019-458-S&E, `https://repub.eur.nl/pub/114018`

Turturea, R., *Overcoming Resource Constraints: The Role of Creative Resourcing and Equity Crowdfunding in Financing Entrepreneurial Ventures*, Promotors: Prof. P.P.M.A.R. Heugens, Prof. J.J.P. Jansen & Dr. I. Verheuil, EPS-2019-472-S&E, https://repub.eur.nl/pub/112859

Valboni, R., *Building Organizational (Dis-)Abilities: The impact of learning on the performance of mergers and acquisitions*, Promotors: Prof. T.H. Reus & Dr. A.H.L. Slangen, EPS-2020-407-S&E, https://repub.eur.nl/pub/125226

Vandic, D., *Intelligent Information Systems for Web Product Search*, Promotors: Prof. U. Kaymak & Dr. Frasincar, EPS-2017-405-LIS, https://repub.eur.nl/pub/95490

Verbeek, R.W.M., *Essays on Empirical Asset Pricing*, Promotors: Prof. M.A. van Dijk & Dr. M. Szymanowska, EPS-2017-441-F&A, https://repub.eur.nl/pub/102977

Visser, T.R., *Vehicle Routing and Time Slot Management in Online Retailing*, Promotors: Prof. A.P.M. Wagelmans & Dr. R. Spliet, EPS-2019-482-LIS, https://repub.eur.nl/pub/120772

Vlaming, R. de., *Linear Mixed Models in Statistical Genetics*, Promotors: Prof. A.R. Thurik, Prof. P.J.F. Groenen & Prof. Ph.D. Koellinger, EPS-2017-416-S&E, https://repub.eur.nl/pub/100428

Vongswasdi, P., *Accelerating Leadership Development: An evidence-based perspective*, Promotors: Prof. D. van Dierendonck & Dr. H.L. Leroy, EPS-2020-512-ORG, https://repub.eur.nl/pub/134079

Vries, H. de, *Evidence-Based Optimization in Humanitarian Logistics*, Promotors: Prof. A.P.M. Wagelmans & Prof. J.J. van de Klundert, EPS-2017-435-LIS, https://repub.eur.nl/pub/102771

Wang, R., *Corporate Environmentalism in China*, Promotors: Prof. P.P.M.A.R. Heugens & Dr. F. Wijen, EPS-2017-417-S&E, https://repub.eur.nl/pub/99987

Wang, R., *Those Who Move Stock Prices*, Promotors: Prof. P. Verwijmeren & Prof. S. van Bekkum, EPS-2019-491-F&A, https://repub.eur.nl/pub/129057

Wasesa, M., *Agent-based inter-organizational systems in advanced logistics operations*, Promotors: Prof. H.W.G.M. van Heck, Prof. R.A. Zuidwijk & Dr. A.W. Stam, EPS-2017-LIS-424, https://repub.eur.nl/pub/100527

Wessels, C., *Flexible Working Practices: How Employees Can Reap the Benefits for Engagement and Performance*, Promotors: Prof. H.W.G.M. van Heck, Prof. P.J. van Baalen & Prof. M.C. Schippers, EPS-2017-418-LIS, https://repub.eur.nl/

Wiegmann, P.M., *Setting the Stage for Innovation: Balancing Diverse Interests through Standardisation*, Promotors: Prof. H.J. de Vries & Prof. K. Blind, EPS-2019-473-LIS, https://repub.eur.nl/pub/114519

Wijaya, H.R., *Praise the Lord!: Infusing Values and Emotions into Neo-Institutional Theory*, Promotors: Prof. P.P.M.A.R. Heugens & Prof. J.P. Cornelissen, EPS-2019-450-S&E, https://repub.eur.nl/pub/115973

Williams, A.N., *Make Our Planet Great Again: A Systems Perspective of Corporate Sustainability*, Promotors: Prof. G.M. Whiteman & Dr. S. Kennedy, EPS-2018-456-ORG, https://repub.eur.nl/pub/111032

Witte, C.T., *Bloody Business: Multinational investment in an increasingly conflict-afflicted world*, Promotors: Prof. H.P.G. Pennings, Prof. H.R. Commandeur & Dr. M.J. Burger, EPS-2018-443-S&E, https://repub.eur.nl/pub/104027

Wu, J., *A Configural Approach to Understanding Voice Behavior in Teams*, Promotors: Prof. D.L. van Knippenberg & Prof. S.R. Giessner, EPS-2020-510-ORG, https://repub.eur.nl/pub/132184

Ye, Q.C., *Multi-objective Optimization Methods for Allocation and Prediction*, Promotors: Prof. R. Dekker & Dr. Y. Zhang, EPS-2019-460-LIS, https://repub.eur.nl/pub/116462

Yuan, Y., *The Emergence of Team Creativity: a social network perspective*, Promotors: Prof. D.L. van Knippenberg & Dr. D.A. Stam, EPS-2017-434-ORG, https://repub.eur.nl/pub/100847

Zhang, Q., *Financing and Regulatory Frictions in Mergers and Acquisitions*, Promotors: Prof. P.G.J. Roosenboom & Prof. A. de Jong, EPS-2018-428-F&A, https://repub.eur.nl/pub/103871

Railway transportation plays an important role in the Dutch mobility system. To make train travel both comfortable and affordable, it is essential for operators to efficiently use their train units. This means that these train units have to be scheduled such that a good balance between passenger service and operational costs is found. Moreover, efficient rescheduling of these train units is necessary to maintain a good service in case a disruption occurs in the railway system. In this thesis, we focus on improving both the scheduling and rescheduling of rolling stock, where we are specifically interested in further integrating these problems into the operational process of a railway operator.

First, we look at the solution methods that are used for solving both rolling stock problems. Here, we compare existing models for rolling stock scheduling and propose a new heuristic for the rolling stock rescheduling problem that can be easily extended for rich rolling stock settings. Second, we look at extensions of the traditional rolling stock rescheduling problem. Here, we consider the reduction of passenger delays through rolling stock rescheduling, where we find that rolling stock rescheduling can significantly reduce the delays experienced by passengers. Furthermore, we consider rescheduling the rolling stock together with those drivers that move train units at the stations.

ERIM

ERIM PhD Series
Research in Management

Erasmus University Rotterdam (EUR)
Erasmus Research Institute of Management
Mandeville (T) Building
Burgemeester Oudlaan 50
3062 PA Rotterdam, The Netherlands

P.O. Box 1738
3000 DR Rotterdam, The Netherlands
T +31 10 408 1182
E info@erim.eur.nl
W www.erim.eur.nl