ROLF NELSON VAN LIESHOUT

Integration, Decentralization and Self-Organization

Towards Better Public Transport

Integration, Decentralization and Self-Organization

Towards Better Public Transport

Integration, Decentralization and Self-Organization Towards Better Public Transport

Integratie, decentralisatie en zelf-organisatie
op naar beter openbaar vervoer

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

Friday 13 May 2022 at 10:30 hours

by

Rolf Nelson van Lieshout
born in Rotterdam, The Netherlands

Erasmus University Rotterdam

Doctoral committee

Promotor: Prof. dr. D. Huisman

Other members: Prof. dr. R.M.P. Goverde
Prof. dr. A. Schöbel
Prof. dr. A.P.M. Wagelmans

Co-promotors: Dr. P.C. Bouman
Dr. ir. J.M. van den Akker

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, 547
ERIM reference number: EPS-2022-547-LIS
ISBN 978-90-5892-631-9

Cover image: Eva Jansen www.evaontwerp.nl
Cover design: PanArt, www.panart.nl
Print: OBT bv, www.obt.eu

Acknowledgements

Even though only my name is written on the cover, this dissertation was not a solo effort. On the contrary, I have been supported by an incredible collective of supervisors, colleagues, friends and family. It would not have been possible to write this thesis without you, and *acknowledging* hardly does any justice to the extent of gratitude that I feel towards all of you. Nonetheless, I gladly take the opportunity to fill these pages to thank the people that contributed to the amazing and rewarding journey that I experienced in the past (little over) four years.

First of all, I would like to thank my supervisors. Dennis, choosing vehicle rescheduling as the topic of my bachelor thesis almost six years ago must have been one of the best decisions I ever made, as this led to our collaboration. Reflecting on the last years, what strikes me most is that despite our large differences in experience and seniority, you always treated me as an equal, valued my opinion and were outspoken regarding your confidence in my abilities. This made for a truly enjoyable and fruitful experience. Paul, your flexible supervision style fitted my needs and productivity waves perfectly. Whenever I needed something, I could always jump into your office. Often, my questions would trigger long, pleasant discussions about research, conferences, teaching or programming. Writing the ATMOS paper together within four weeks was also tremendous fun. Marjan, aside from offering the necessary computer science perspective, your efforts in the simulation project were invaluable. Moreover, I am thankful for how you would often step in to cover managerial or administrative tasks, allowing me to focus on research.

Next, I want to thank prof. dr. Anita Schöbel, prof. dr. Rob Goverde and prof. dr. Albert Wagelmans for being part of my inner doctoral committee, and prof. dr. Iris Vis and prof. dr. Martin Savelsbergh for completing the opposition. In the past years, I had the pleasure to witness all of you present or teach in person, and these were, without exception, inspiring experiences that contributed to my development as a researcher. I am therefore truly honored that you have taken the time to serve on my doctoral committee.

I am also indebted to NWO for funding my PhD and to everyone involved in the NWO project *Improving the resilience of railway systems*. Quite literally, my research would not have been possible without you. I would like to thank Deb for leading the project and all other project members for their efforts and contributions during the project meetings.

Furthermore, I am grateful to NS and ProRail for their involvement in my PhD project. The inside look into the world of rail and passenger transport was extremely motivating and inspiring. I especially want to thank Pieter-Jan, Gábor, Joël, Maaike, Ramon, Anneke, and everyone alse at the PI department of NS. Ever since I stepped into 3.314 to write my master thesis, I have always felt most welcome. Aside from the many insightful train-related discussion, I look back with great pleasure at all the taartmomenten, the PI-uitjes, the cycling trips, winning trophies at the Midwintermarathon, and on spending countless hours on the AIVD Kerstpuzzel.

In addition, it was a privilege to be part of such an awesome group of PhD students. It was simply wonderful to be able to share research ideas, teaching experiences, and the sweets and sours of (academic) life in such a friendly and supportive environment. Rutger, Thomas (x2), Kevin, Gert-Jaap, Johann, Rowan, Nemanja, Weina, Mathijs, Naut, Ymro, Lisanne, Utku, Karel, Rick, Mette, Bart, Pedro, I had never expected to have so much pure fun during my PhD, and this was largely thanks to all of you. It was an absolute pity that, due to the pandemic, we missed opportunities to attend conferences together and had to discontinue the PhriDays and game nights. I still hope we can catch up on some of these missed experiences in the future. Rowan, it was a great pleasure to be your paranimf and I am very glad that you are returning the favor.

I also would like to thank my other (past) colleagues from the Econometric Institute, and others I have worked with during the last years. Twan, Marie, Wilco, Remy, Robin, Christiaan, Ilker, Emö, Evelot, Rommert, Olga, Riley, Marieke, Shadi, Noa, if I could name one characteristic that you share it would be that you are all remarkably approachable. Despite lack of time, I have always felt completely at ease to ask your help or advice, and I cannot overstate how much this stimulated me. I definitely plan to pass on this admirable trait to others during my future career.

Finally, I would like to thank my family. Pap en mam, het was vast vermoeiend om mijn onophoudelijke vragenstroom van jongs af aan in gang te houden, maar dit heeft nu toch tot een mooi tastbaar resultaat geleid. Dank voor jullie onvoorwaardelijke steun en interesse in wat ik doe. Lucia, jij zet me altijd met beide voeten weer terug op de grond wanneer ik te ver in de treinenwereld dreig af te dwalen. Ik ben dan ook erg dankbaar om je naast me te hebben staan als paranimf.

I would like to end with some words that may provide you some comfort when you find yourself stranded in an out-of-control situation, as (spoiler alert) we have not been able to solve them all (yet).

Verbinding verstoord?

Verstoring verbindt.

Verlangen vertraagd?

Vertraging verlangt.

Verlichting gestrand?

Stranden verlicht.

Rolf Nelson van Lieshout
Rotterdam, February 2022

Table of contents

Chapter 1

Introduction

The history of public transport can be traced back to 1662, when the French mathematician, inventor and philosopher Blaise Pascal came up with the brilliant idea to efficiently channel the transportation demand of 17th century Parisians by proposing a system that we now refer to as public transport. In Pascal's *carosses à cinq sols* (five-penny coaches), horse-drawn buses were used to operate a network with five routes according to a fixed schedule, with frequencies up to eight buses per hour (Monmerqué, 1828). Despite a positive initial reception by the public, the system was short-lived, which has been attributed to a ruling of the Parlement de Paris that only the nobility can use the carosses.

Pascal's idea was ahead of its time, but as cities, economic activity and transportation demand kept growing, the potential of public transport did as well. In today's society, public transport plays an essential role that benefits *everyone*. Users of public transport are getting around from point A to point B in an affordable, comfortable, and quick manner, stimulating economic mobility and economic activity in general. Moreover, by temporal and spatial pooling of passengers in high-capacity vehicles, public transport drastically reduces congestion, allowing users of private transport to enjoy relatively uncongested, and therefore fast, roads. An advantage of public transport that has become increasingly important in recent years is its low climate impact. On a local scale, public transport helps to reduce noise and air pollution within cities. On a global scale, the lower emission per passenger-kilometer compared to road transport can help to battle global warming.

Given its undisputed benefits, there is a clear need for making public transport systems as attractive, cost-efficient and sustainable as possible. Lines and frequencies, the timetable and resource schedules should be designed in order to maximize passenger demand, while simultaneously limiting operator costs and environmental impact. In the spirit of the mastermind behind the carosses à cinq sols, the main tool for achieving these objectives is *mathematics*. By framing these problems in the language of mathematics, we can optimize public transport. This is the overarching theme of this thesis: improving public transport using mathematical models.

Within the field of public transport optimization, this thesis explores two topics. The first topic is the integrated planning of public transport, which is introduced in Section 1.1. The second topic concerns decentralized control in public transport, which is introduced in Section 1.2. Next, Section 1.3 provides an overview of the contents of this thesis, and its contributions. Finally, Section 1.4 briefly discusses the context in which this research has been conducted.

1.1 Integrated Public Transport Planning

The public transport planning process consists of several steps, visualized in Figure 1.1. Traditionally, these steps are performed sequentially. As the first step, the network is designed, which involves planning the infrastructure, the stops (stations) and determining between which stops services can be operated. This network design serves as input for the line planning step, where one decides which lines to operate and their frequencies (or equivalently, their *headways*, the time between consecutive services). In the timetabling step, the departure and arrival times are determined at all stops for the installed lines. Next, the vehicles (referred to as *rolling stock* in railways) are scheduled, such that all trips in the timetable are performed. The final crew planning step is typically also decomposed into scheduling and rostering: crew scheduling involves constructing the days of work or duties such that all tasks are covered, and crew rostering involves assigning the duties to the crew members.

The first planning steps – network design and line planning – correspond to strategic, long-term decisions and are usually only performed once every few years. Timetabling is seen as a tactical problem, with operators commonly updating their timetable every year. By adjusting the vehicle and crew schedules, operators can respond to short-term trends, hence these steps are usually performed on a more regular basis.

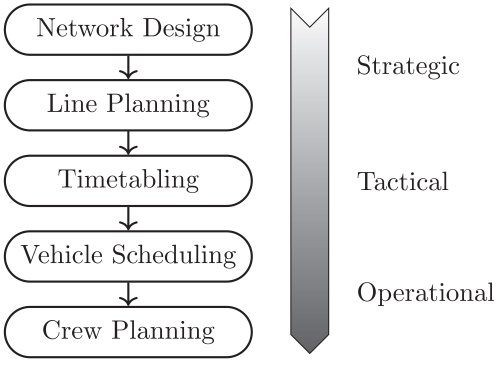


Figure 1.1: Overview of the steps in the planning process of public transport.

In public transport planning, the objective is typically threefold: to maximize the service quality for the passengers, to minimize the costs for the operator (thereby also minimizing environmental impact), and to maximize the robustness of the system against disturbances. To support planners in realizing these objectives, many mathematical models and methods have been developed. The development and application of mathematical techniques to support decision-makers is called Operations Research. For an overview of such approaches in the field of public transport planning, see Borndörfer et al. (2018b), Caprara et al. (2007) and Huisman et al. (2005).

In Part I of this thesis, we focus on improving Operations Research models for public transport by (partially) integrating the steps in the planning process. This has large potential benefits, because one cannot evaluate the exact quality of decisions taken in the earlier planning stages until the outcomes of the later stages are known. As a consequence, the sequential approach asks for making estimations and approximations, likely resulting in suboptimal solutions. For example, service quality highly depends on the paths taken by passengers and the resulting travel times, which are determined by the timetable. Hence, in the line planning step planners need to use estimates for driving, dwelling and transfer times, resulting in an approximate measure of service quality that may not reflect the actual service quality. Integrated planning can also be motivated from a mathematical point of view: the sequential approach is a greedy approach, which is generally suboptimal.

Complete integration leads to an immensely large and therefore unsolvable problem. Furthermore, complete integration neglects the different time scales associated with the planning problems: the later-stage problems depend on many operational details that are simply unknown when the early-stage problems need to be solved. Therefore, a more tractable and practical approach is to consider partial integration and improve the quality of the approximations of later-stage outcomes in the earlier steps. This is the approach studied in Chapter 2 and 3 of this thesis.

1.2 Decentralized Control in Public Transport

The overview of the planning steps described in the previous section neglects one important feature of public transport systems: disruptions. Events such as infrastructure malfunctions, vehicle breakdowns or heavy traffic cause deviations from the planned schedules. This leads to the need for real-time rescheduling, or *disruption management*. Below, we first discuss traditional, schedule-based, disruption management. Thereafter, we discuss decentralized control, and when this could be the preferable alternative.

1.2.1 Traditional Disruption Management

As disruptions render the regular schedule infeasible, disruption management involves finding a new feasible schedule by modifying the timetable and resource schedules. Analogously to the respective planning problems, rescheduling is usually performed in sequence. First, a new timetable is constructed by canceling, retiming and rerouting services. Then, the vehicles are rescheduled, matching the new timetable. Finally, the crew schedule is updated to account for the changes to the timetable and vehicle schedule. As infeasibilities may be encountered in the rescheduling of the vehicle and crew schedules, it may be necessary to perform further updates in a feedback loop. For an overview of Operations Research techniques for disruption management in public transport, we refer to Cacchiani et al. (2014), Ghaemi et al. (2017) and Visentini et al. (2014).

In addition to prescriptive Operations Research models, a more descriptive model relevant in disruption management is the *bathtub* model. This model identifies three phases in the process of handling a disruption, see Figure 1.2. The first phase starts directly after the initial disruption and involves gathering all information about the disruption and its expected duration, determining the disruption timetable and transitioning towards it. In the second phase, the disruption timetable is operated, often based on predefined contingency plans. In the third phase, the disruption is over and there is a transition towards the regular timetable. The name of the bathtub model refers to the traffic level during the three phases, (loosely) resembling a bathtub: it decreases during the first phase, reaches a constant lower-than-regular level during the second phases, and increases back to the regular level in the third phase.

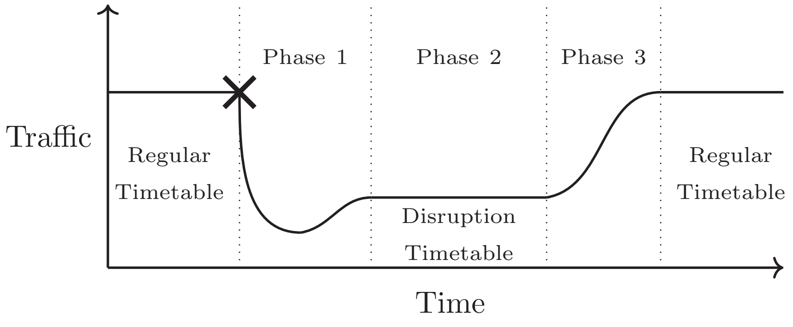


Figure 1.2: Bathtub model for disruption management.

1.2.2 Decentralized Strategies

Conventional disruption management is entirely schedule-based: timetables and resource schedules are updated in real-time, and it is assumed that these schedules can be communicated to all actors involved in operating the system. This requires there to be one or multiple control centers, where dispatchers are responsible for performing and communicating rescheduling actions.

Decentralized control refers to a different paradigm that can be used for operating public transport. The concepts of a timetable and resource schedules are abandoned, and since there is no schedule, there is also no need to reschedule. Consequently, there is no control center. Instead, local agents are responsible for dispatching vehicles and crew, based on local information. This approach is the focus of Part II of this thesis.

In this thesis, decentralized strategies always specify what to do when a vehicle has finished a trip. At in-between stations of lines, vehicles should of course always continue on their current line, but once a vehicle arrives at the terminal station, it should be decided *which line* the vehicle performs next and *when* it should depart. If crew is also considered, the same should be determined for the crew. We now give two examples of decentralized dispatching strategies that are proposed and analyzed in this thesis. Both are explained using an example of a station that serves as a terminal for two (bidirectional) lines, A and B, with a headway of 10 minutes.

Example 1 (ASAP-STAT)**.** *Arriving vehicles return on the same line (in the reverse direction) and are instructed to depart as soon as possible. For example, a vehicle arriving on line A at 10:00 will depart again on line A as soon as possible.*

Example 2 (SYNC-DYN)**.** *Arriving vehicles are assigned to a line based on the most recent departure times and next departure times are determined in order to minimize deviations from the target headway. For example, suppose the most recent departure times are 9:50 for line A and 9:52 for line B. Then, a vehicle arriving at 9:55 is assigned to line A and instructed to depart at 10:00 to meet the 10-minute target headway. The next arriving vehicle will always be assigned to line B. If it arrives before 10:02, it is instructed to depart at 10:02, otherwise it should depart as soon as possible*

The ASAP-STAT and SYNC-DYN strategies are examples of decentralized control, as they can be applied independently at the terminal stations of the network, without any form of communication between stations. Moreover, both are easy to implement, requiring little information and computation.

In case of functional information systems, smooth rescheduling and seamless communication, centralized control is clearly preferred over decentralized control: one has a global overview of the system and can continuously reschedule based on complete information. However, if one of these conditions is not met, the merits of decentralized control become clear: these strategies are fast to apply, robust against incomplete information and, provided that the line plan is known to all actors, only require communication between actors within the same station. Below, we describe multiple practical situations where decentralized control can prove valuable.

Out-of-Control Situations

Our main motivation for studying decentralized control of public transport are so-called *out-of-control situations*. These are situations in railway systems where dispatchers cease to have an overview of the system and consequently decide to terminate all railway traffic in the affected region, even though the required resources (infrastructure, rolling stock and crew) might be available. In the Dutch railway system, out-of-control situations occurred about ten times in the period 2009-2012 due to extreme weather. From then on, it was decided to operate a reduced timetable in case of extreme weather forecasts. However, out-of-control situations still occur occasionally because of, for example, (short-lasting) power outages or bomb alerts.

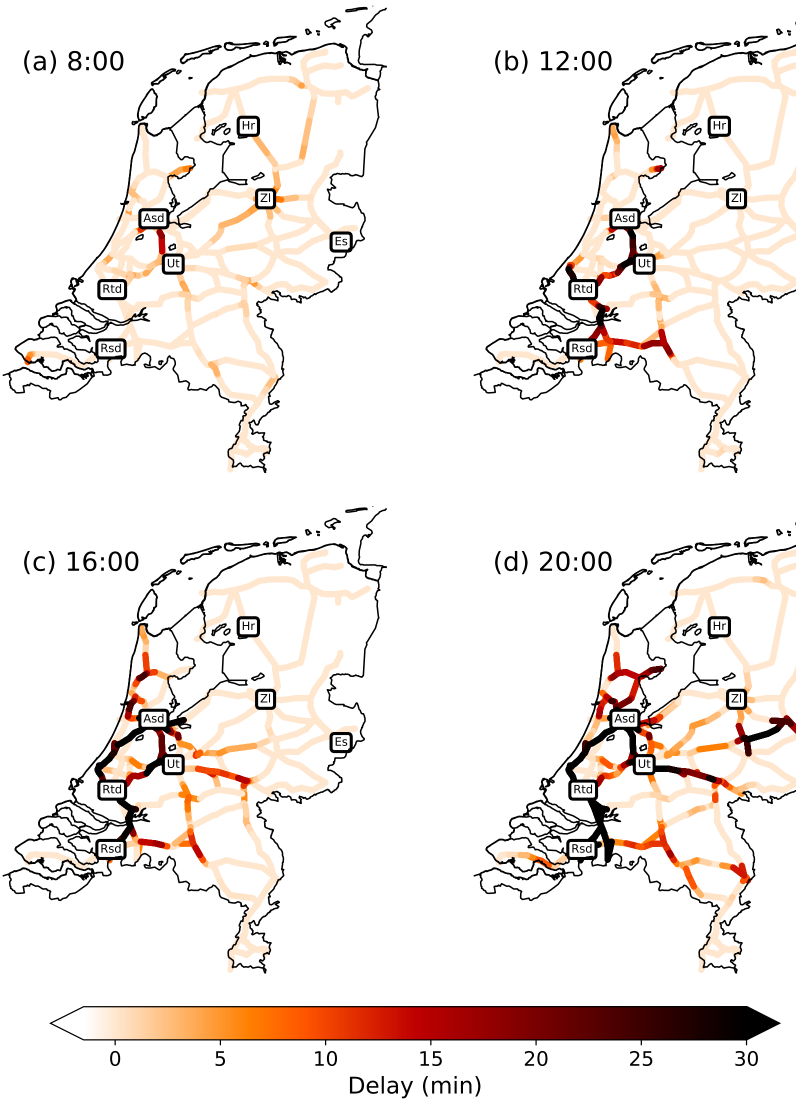


Figure 1.3: Average delay on the Dutch railway network on February 3, 2012. Abbreviations: Amsterdam (Asd), Rotterdam (Rtd), Roosendaal (Rsd), Utrecht (Ut), Enschede (Es), Zwolle (Zl) and Heerenveen (Hr). Taken from Dekker et al. (2021).

The size and impact of out-of-control situations can be illustrated using Figure 1.3, which visualizes the evolution of delay in the Dutch railway network on February 3, 2012. As a result of extreme weather, throughout this day there were 305 infrastructure malfunctions and 250 issues with rolling stock (including six completely broken trains). The number of trains that suffered delays due to missing crew was twice as high as on a regular day. Dispatchers could not keep up with the rapid succession of disruptions, resulting in an out-of-control situation. As can be seen in the figure, initially most delays occurred between Amsterdam and Utrecht. During the day, the delay spread towards Rotterdam and Roosendaal, reaching even the far east of the Netherlands by the beginning of the evening.

Out-of-control situations have a number of characteristics that yield traditional disruption management strategies ineffective. Primarily, there is no isolated, well-defined disruption with a known duration (such as a track blockage that lasts 2 hours). Rather, out-of-control situations are fuzzy, large-scale and characterized by incomplete information. One or multiple source events strongly disrupt a considerable portion of the timetable and resource schedules, causing information systems for the timetable, rolling stock and crew to lag behind. Dispatchers are faced with uncertainty, both regarding the duration of the disruption and the actual whereabouts of rolling stock and crew. Another typical assumption in disruption management is that all stakeholders in the operations act as expected. In out-of-control situations, this assumption is not met. In fact, there have been cases of train drivers and conductors not being aware or even ignoring rescheduling decisions made by dispatchers.

A decentralized dispatching approach is robust against the discussed features of out-of-control situations. As long as infrastructure, rolling stock and crew are available, simple local rule-based strategies can be applied, reducing the dependence on central dispatchers and restoring traffic to a reasonable level. A decentralized approach could also be used as a preventive measure, if there is a high risk for going out-of-control. For a further discussion of out-of-control situations, and how decentralized control can help to reduce their impact, we refer to Dekker et al. (2021).

Other Applications

In addition to out-of-control situations, there are other scenarios where the application of decentralized control might be fruitful. Firstly, consider the case of operators in developing countries or remote areas, operators of small networks or services that are run on an incidental basis, for example bus services replacing disrupted rail lines. Centralized control can be undesirable in these situations because it demands considerable effort and comes with high monetary costs: it requires constant tracking of vehicles and crew, monitoring of schedule adherence and rescheduling in case of schedule deviations or disruptions, possibly using a costly decisions support system. Conversely, by applying simple decentralized strategies, the system essentially runs itself, without any information or decision support system.

Secondly, decentralized strategies can be the preferable alternative for high-frequency systems. If frequencies get high enough, passengers stop consulting timetables and arriving at the station shortly before the scheduled departure time of their trip. Instead, passengers start arriving uniformly. The result is twofold: exact schedule adherence becomes less important, while maintaining constant headways becomes increasingly important. For example, suppose that services of a certain line are scheduled every 4 minutes, at 10:00, 10:04, 10:08 et cetera. If the 10:00 service is delayed by 3 minutes, running a service at 10:04 is inefficient, as it will run close to empty (potentially causing further bunching). Instead, it might be preferred to delay all services by 2 minutes, to maintain constant headways. Of course, the latter can also be achieved through centralized control, but decentralized strategies are potentially faster because they cut out the role of the control center.

Thirdly, decentralized control could be be advantageous in the first phase of a disruption, referring back to the bathtub model. Decentralized strategies could be helpful in the first phase because it shares features with out-of-control situations: often all resources are there, but the right contingency plan still has to be determined, resulting in the cancellation of many services. As it can take quite some time for the control center to gather all information and determine the disposition timetable, decentralized strategies could provide a means to maintain a base service level during the first phase.

Despite the lack of present-day relevance, we end this section with a final reference to the carosses à cinq sols. We are not aware of any sources about the real-time management of Pascal's system. However, it is a fascinating thought that the decentralized strategies explored in this thesis do not at all depend on modern technology, such that they *could* have been applied to the carosses.

1.3 Thesis Outline and Contributions

This thesis is structured around two parts and consists of seven chapters. An overview of the thesis is provided in Figure 1.4. All chapters are self-contained, so they can be read independently.

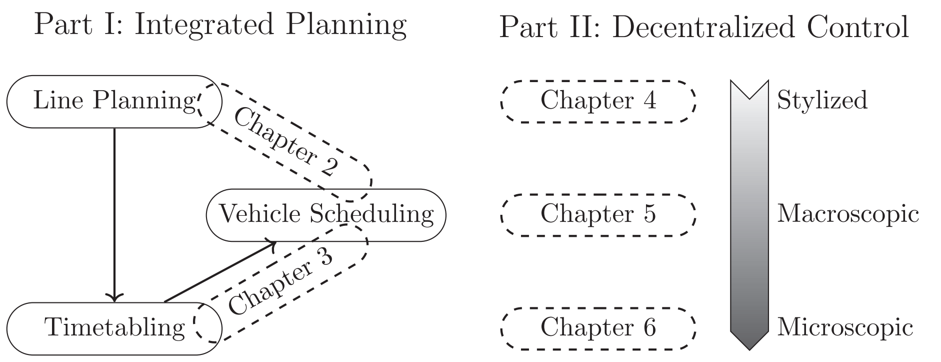


Figure 1.4: Schematic overview of the thesis.

Part I considers the integrated planning of public transport. Chapter 2 combines line planning and vehicle scheduling, and presents a new approach to determine the number of vehicles that are required to operate a line plan, without the intermediate step of computing a timetable. Chapter 3 combines timetabling and vehicle scheduling, and develops a novel optimization model for jointly optimizing a periodic timetable and vehicle circulation schedule.

Part II deals with decentralized control in public transport and railway systems in particular. Chapter 4 presents a theoretical analysis of a simple, decentralized strategy for vehicle dispatching. Chapter 5 considers the application of decentralized control to out-of-control situations in railways. This chapter develops a novel algorithm to find line plans that are suited for these circumstances and evaluates the performance of decentralized strategies on these line plans in a macroscopic railway simulation. To ensure that the line plan is feasible with respect to the available rolling stock, the line plan model partially integrates rolling stock scheduling, using ideas developed in Chapter 2. Chapter 6 again focuses on railways and tests decentralized dispatching of both vehicles and crew in a microscopic railway simulation.

Chapter 2: "Vehicle Scheduling Based on a Line Plan". Published in the *ATMOS* conference proceedings as Van Lieshout and Bouman (2018).

Chapter 2 analyzes approaches to estimate the number of vehicles required to operate a line plan, without having to first compute a timetable. Formally, it studies the following problem: given a set of lines in a public transportation network with their round trip times and frequencies, a maximum number of vehicles and a maximum number of lines that can be combined into a vehicle circulation, does there exist a set of vehicle circulations that covers all lines given the constraints. We show that this problem is NP-hard if the number of lines that can be combined into a circulation is equal to or greater than three. We pay special attention to the case where at most two lines can be combined into a circulation, which is NP-hard if a single line can be covered by multiple circulations. If this is not allowed, a matching algorithm can be used to find an optimal solution. We also provide an exact algorithm that is able to exploit low treewidth of the so-called circulation graph.

Chapter 3: "Integrated Periodic Timetabling and Vehicle Circulation Scheduling". Published in *Transportation Science* as Van Lieshout (2021).

The impact timetabling has on the number of required vehicles, which directly translates to operator costs, is rarely taken into account in timetabling models. Therefore, in Chapter 3, we consider the problem of jointly optimizing the (periodic) timetable and the vehicle circulation schedule, which specifies the cyclic sequences of trips vehicles perform. In order to obtain high-quality solutions to realistic instances, we derive new theoretical results on vehicle circulation scheduling, which are then used to enhance the formulation. Ultimately, this allows the operator to make an informed trade-off between costs and passenger service. A computational study demonstrates the effectiveness of the improved formulation. Moreover, using this approach we are able to find timetables requiring substantially fewer vehicles at the cost of minimal increases of the average travel time of passengers.

Chapter 4: "A Self-Organizing Policy for Vehicle Dispatching in Public Transit Systems with Multiple Lines". Published in *Transportation Research Part B: Methodological* as Van Lieshout et al. (2021).

In Chapter 4, we propose and analyze an online, decentralized policy for dispatching vehicles in a multi-line public transit system. In the policy, vehicles arriving at a terminal station are assigned to the lines starting at the station in a round-robin fashion. Departure times are selected to minimize deviations from the target headway. We prove that this policy is self-organizing: given that there is a sufficient number of vehicles, a timetable spontaneously emerges that meets the target headway of every line. We present both theoretical and numerical results on the time until a stable state is reached and on how quickly the system recovers after the breakdown of a vehicle. Experiments on three real-world transit systems show that our policy performs well, even if not all assumptions required for the theoretical analysis are met: if there are enough vehicles, the realized headways are close to the target headways.

Chapter 5: "Determining and Evaluating Alternative Line Plans in Out-of-Control Situations". Published in *Transportation Science* as Van Lieshout et al. (2020).

Chapter 5 develops and tests disruption management strategies for out-of-control situations. First, we propose an algorithm that finds an alternative line plan that can be operated in the affected part of the railway network. As the line plan should be feasible with respect to infrastructural and rolling stock restrictions, we integrate these aspects in the algorithm in a Benders'-like fashion. Second, to operate the railway system within the disrupted region, we propose several local train dispatching strategies requiring varying degrees of flexibility and coordination. Computational experiments based on disruptions in the Dutch railway network indicate that the algorithm performs well, finding workable and passenger oriented line plans within a couple of minutes. Moreover, we also demonstrate in a simulation study that the produced line plans can be operated smoothly without depending on central coordination.

Chapter 6: "Microscopic Simulation of Decentralized Dispatching Strategies in Railways". Submitted.

Chapter 6 analyzes the effectiveness of decentralized strategies for dispatching rolling stock and train drivers in a railway system. We test the performance of four rolling stock and two driver dispatching strategies in a microscopic simulation. Our test case is a part of the Dutch railway network, containing eleven stations linked by four train lines. We find that with the decentralized dispatching strategies, target frequencies of the lines are approximately met and train services are highly regular without large delays. Especially strategies that allow rolling stock to switch between lines result in a good performance.

Contributions

The main contribution of Chapters 2 to 6 is fourfold.

Firstly, we provide new theoretical insights in the structure of certain problems and systems arising in public transport. In Chapter 2, we analyze the problem of estimating the number of vehicles required to operate a line plan, without computing a timetable, and present numerous complexity results on different variants of the problem, including a heuristic with an approximation guarantee. In Chapter 3, we prove, among other results, that given a timetable, a greedy algorithm is optimal in order to determine the number of required vehicles. In Chapter 4, we prove that a simple decentralized strategy induces a self-organizing public transport system and also derive bounds on the time until the system stabilizes.

Secondly, we use the derived theoretical results to develop effective optimization approaches. The theoretical results in Chapter 3 are the key ingredients for an improved formulation for the integrated periodic timetabling and vehicle circulation scheduling problem that is developed later in that chapter. In Chapter 5, we present a real-time line planning algorithm with a rolling stock component that is based on the ideas developed in Chapter 2.

Thirdly, we test and assess the performance of the proposed approaches using numerical experiments on both real-world and artificial public transport networks. The integrated timetabling and vehicle scheduling model developed in Chapter 3 is applied to the entire Dutch intercity network operated by Netherlands Railways (NS), as well as a partial network containing both intercity and regional services. This partial network also serves as the test bed for the line planning algorithm developed in Chapter 5 and the microscopic simulation in Chapter 6. The practical performance of the decentralized strategy of Chapter 4 is assessed using the bus networks of The Hague, Amersfoort and Göttingen, as well as artificially created networks.

Fourthly, our analyses lead to valuable insights for public transport operators. In Chapters 2, we show that despite the problem being hard in theory, it is possible to estimate the number of vehicles using a fast heuristic, bypassing the time-consuming and tedious process of planning a timetable. In Chapter 3, we illustrate that integrated planning can lead to large benefits: compared to the sequential approach, the developed integrated model is able to considerably reduce the number of vehicles at the cost of minimal increases in passenger travel time. Chapters 4, 5 and 6 all suggest that decentralized control is an alternative worth considering in certain situations.

1.4 Research Statement

This thesis is a result of research conducted within the interdisciplinary project "Improving the resilience of railway systems", funded by the Dutch Research Council (NWO), as part of the research programme Complexity in Transport & Logistics. This project is a collaboration between Erasmus University Rotterdam, Utrecht University, Delft University of Technology, Netherlands Railways and ProRail. The main goal of this project is to investigate approaches to avoid or reduce the impact of out-of-control situations. The chapters in Part II of this thesis are directly related to this topic. This research also triggered two spin-off projects, which led to the chapters in Part I.

The research in all chapters except Chapter 6 was primarily conducted by the author of this thesis. For Chapter 6, Rafael Mendes Borges, Teun Druijf and a team of bachelor students Computer Science in the faculty of Science, Utrecht University, were responsible for the majority of the coding, while the author of this thesis was responsible for research design, testing and fine-tuning the developed software, analyzing the results and the write-up. For all chapters in this thesis, it holds that their quality has greatly been improved by frequent discussions with doctoral advisors Paul Bouman, Marjan van den Akker en Dennis Huisman, and with other partners within the NWO project.

Part I

Integrated Planning in Public Transport

Chapter 2

Vehicle Scheduling Based on a Line Plan

This chapter is based on Van Lieshout, R.N. and Bouman, P.C. (2018). Vehicle Scheduling Based on a Line Plan. 18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018). Schloss Dagstuhl-Leibniz-Zentrum für Informatik (OASIcs), 15:1–15:14.

2.1 Introduction

Traditionally, the planning of public transport services occurs in a number of steps. First, a *line plan* is constructed where service routes, usually referred to as lines, are selected such that high quality service is provided to the customers (Borndörfer et al., 2007; Schöbel, 2012). In the second step, a *timetable* is constructed that specifies the departure and arrival times along the stops of all lines (Caimi et al., 2017; Ibarra-Rojas et al., 2015). In the final step, *vehicles* and possibly *human resources* are planned as they are necessary resources to execute the services (Abbink et al., 2011; Fioole et al., 2006; Kliewer et al., 2012). As the individual scheduling steps are already quite challenging, the sequential planning approach is traditionally applied because an integrated approach is computationally not tractable. The disadvantage of the sequential approach is that the objectives of the subsequent steps are not taken into account when the prior steps are solved. In particular, the line plan and timetable are usually optimized based on passengers' convenience, while the vehicle schedule is optimized based on the operator costs. Therefore, the optimal solution for the combined problem is likely to be missed.

Recently, a number of authors have proposed ideas to integrate the separate planning steps. One example, the *eigenmodel* (Schöbel, 2017), replaces a fixed order with an iterative approach that takes a different route through the separate steps, controlling both the passengers' convenience and the operator costs during the process. Pätzold et al. (2017) takes a different approach and incorporates penalties during the line planning phase for lines which can not be covered efficiently by a vehicle in a periodic timetable. That is, assuming the cycle time is 60 minutes, a line with frequency one for which a round trip takes 54 minutes (a downtime of 6 minutes) is given a low penalty, while a line with frequency one for which a round trip takes 65 minutes (a downtime of 55 minutes) is given a very high penalty.

In this chapter, we consider the construction of vehicle schedules based on the line plan without the intermediate step of constructing a timetable. One goal of this is to quickly assess whether a line plan can be operated using a small number of vehicles. This allows public transport operators to detect potential inefficiencies early in the planning process, without having to compute a timetable first. The novel aspect of our approach is that we explicitly consider the possibility to combine lines into larger vehicle circulations. To illustrate, while a line that takes 65 minutes with a period of 60 minutes may seem inefficient by itself, it may be a good option if we can combine it with a line of 55 minutes. Although combinations of lines can help to reduce the number of vehicles required to operate a line plan, large and complex combinations of lines may result in greater dependencies between the operations of the different lines. Therefore, we provide a detailed examination of cases where at most two lines can be combined in a vehicle schedule.

The remainder of this chapter is organized as follows. In Section 2.2 we formally introduce the vehicle circulation scheduling on a line plan. In Section 2.3 we study the computational complexity of the general case. In Section 2.4 we study the special case where only two lines can be combined in a circulation. We conclude and discuss ideas for future research in Section 2.5.

2.2 Problem Formulation

In this chapter, we assume a *line plan* is already given and want to determine the minimum number of vehicles that are required to operate the line plan *without* the intermediate step of constructing a timetable. For the line plan, we have a fixed time period denoted by T and a set of lines $\mathcal{L}$ where a line $\{v, u\} \in \mathcal{L}$ is an unordered pair of terminal stations of the line. The *line graph* $L = (V, \mathcal{L})$ has terminal stations V as vertices and the lines as edges. In the line plan each line $l \in \mathcal{L}$ has a round trip time t_l and an integer frequency f_l assigned to it. The round trip times specified by the line plan should at least include the minimum driving and dwelling times required to execute the line, but can also include some slack to make operations more robust against disruptions. The frequency defines the number of times the line service must be executed by a vehicle within each time period of length T.

If vehicles are only allowed to operate a single line, the number of vehicles required for line l equals $\lceil \frac{t_l}{T} f_l \rceil$. In order to reduce the number of vehicles required to operate the line plan, public transport operators may consider *circulations*, which are a combination of lines that can be executed by a single vehicle. Formally, a circulation $c \subseteq \mathcal{L}$ is a set of lines that can be operated by one or more vehicles. We allow lines to be contained more than once in the set, since this can be relevant if the line has a frequency greater than 1. We call the number of times a line l is contained in a circulation the *multiplicity* of l in c. Furthermore, we assume that a summation over the lines in the circulation includes a line multiple times if it has a multiplicity greater than 1. An example of a situation where we want to assign the same line to a circulation multiple times, is a line with a round trip time of $\frac{T}{2}$ and a frequency of 2. In that case, we want to consider a circulation where we execute the line twice during each period.

For this chapter, we only consider circulations c such that the lines it contains form a connected subgraph of L. As a consequence of this, the time t_c needed to perform a single round trip of a circulation c can be expressed as $t_c = \sum_{l \in c} t_l$. Furthermore, a circulation c corresponds to a directed Eulerian tour in $\overleftrightarrow{L}$, where $\overleftrightarrow{L}$ is the *directed line graph*, which is a a symmetric directed graph derived from L where each edge of L is replaced by two arcs, one for each direction. Let us now consider the correspondence between a connected subset of L and the directed cycle in $\overleftrightarrow{L}$.

Lemma 2.1. *A connected subset $c \subseteq \mathcal{L}$ of lines in the line graph L corresponds to a directed Eulerian sub-graph in the directed line graph $\overleftrightarrow{L}$. Thus, there always exists a directed cycle in $\overleftrightarrow{L}$ that visits all arcs that correspond to both directions of the lines in c a number of times that is exactly equal to the multiplicity of the lines in the circulation.*

Proof. A directed graph contains a directed Eulerian cycle if two conditions hold: (1) for every vertex the in-degree is equal to the out-degree, and (2) the graph is strongly connected. Since the graph $\overleftrightarrow{L}$ is symmetric, each vertex must have one outgoing arc for each incoming arc, and thus condition (1) always holds. As c is a connected subset of lines, the corresponding lines in $\overleftrightarrow{L}$ must be connected as well. Since the graph is symmetric, this implies that it is also strongly connected. □

Although more general concepts of circulations that do not enforce connectivity can be considered, these would require dead-heading of vehicles as part of a line plan. While public transport operators have to use dead-heading when operations start up, or frequencies of lines are changed throughout the day, it is usually avoided as much as possible by public transport operators during regular operations. As we focus on regular operations, we consider those generalized concepts of circulations beyond the scope of this chapter.

In the problem we consider we do not only need to decide which circulations should be used, but we also have to decide how many vehicles must be assigned to each circulation to cover the constraints of the line plan. Since a circulation c can have a round trip time that is larger than T, we may need to assign multiple vehicles to it in order to enforce that every line in the circulation is covered during every period. We define the number of vehicles required to perform the circulation once in every period as $k_c = \lceil \frac{t_c}{T} \rceil$. If we would assign fewer vehicles than k_c to a circulation, the vehicle is not finished with its circulation when a new period starts and as a consequence no vehicle executes the circulation during some periods. Furthermore, an upper bound on the number of vehicles that can be assigned to a circulation c is given by the expression $\min_{l \in c} f_l k_c$, as assigning more vehicles to the circulation would imply that the line where the minimum was attained is executed with greater frequency than the line plan prescribes. In the special case that all frequencies are 1, k_c itself is an upper bound on the number of vehicles that can be assigned to c.

Note that we do not enforce that each line is covered by a single circulation. For example, suppose we have two lines a and b with round trip times $t_a = t_b = 30$, but different frequencies $f_a = 3$ and $f_b = 1$, with a period time of $T = 60$. The most efficient way to cover these lines is to have one vehicle circulation that executes line a two times each period, while a second vehicle alternates between line a and line b, executing both lines once each period. We investigate a strict version of the problem where a line can only be covered by a single circulation in Section 2.4.2.

The goal of the problem studied in this chapter is to find a set of circulations and the number of vehicles assigned to each circulation such that all lines are covered. From a practical point of view it is desirable that the selected circulations do not contain too many lines, as this creates significant dependencies in the operations that make the operations extremely sensitive to minor disruptions. When a disruption occurs somewhere in a circulation, all subsequent lines in the circulation are affected. Furthermore, very large circulations can only be operated in practice if the timetables of all the lines in the circulation are synchronized. This may lead to problems in the timetabling phase, especially if you want to offer good transfer possibilities to passengers who want to follow different paths than the vehicles.

To avoid large circulations, we consider a restriction on the maximum number of lines than can be included in a circulation. We call a circulation c an α-circulation if the number of unique lines in c is α. A 1-circulation is also referred to as a *fixed circulation* and a 2-circulation is also referred to as a *combined circulation.* We introduce an input parameter κ that restricts the number of unique lines that can be combined in a single circulation. With this concept clearly defined, we can introduce the decision variant of our problem in a formal way:

VEHICLE CIRCULATION SCHEDULING ON A LINE PLAN (VCS-LP)
INSTANCE: A line graph $L = (V, \mathcal{L})$, a maximum number of of unique lines that are allowed to exist in a single circulation κ and a maximum number of vehicles z
QUESTION: Does there exist a set of circulations C, with for each circulation $c \in C$ a value assigned to the integer decision variable $\theta_c \in \mathbb{N}$ that indicates how many vehicles are assigned to circulation c, such that:
(1) the circulations cover all lines in every period, i.e. $\forall l \in \mathcal{L} : f_l = \sum_{c \in C : l \in c} \frac{\theta_c}{k_c}$,
(2) there are no α-circulations in C with $\alpha > \kappa$, and
(3) at most z vehicles are required to execute all circulations, i.e. $\sum_{c \in C} \theta_c \leq z$.

The optimization version of this problem seeks to find the smallest z for a given line graph and a given parameter κ, such that there exists a set of circulations C with integer vehicle assignments θ that satisfy the conditions.

2.3 Computational Complexity

First, we consider a lower bound on the number of vehicles that is needed in a given line graph. Since we are not allowed to use dead-heading, we must consider the connected components of a line graph separately, as no vehicle will be able to move from one component to another. Thus without loss of generality we assume that a line graph is connected, since if it is not we can decompose the problem into independent sub-problems. A lower bound on the number of vehicles required can be computed by dividing the total running time by the cycle time. As no fractional vehicles can be used, we can round the number of vehicles up. This gives the following necessary condition to check if the instance can possibly be a YES-instance:

$$z \geq \left\lceil \frac{\sum_{l \in \mathcal{L}} t_l \cdot f_l}{T} \right\rceil \tag{2.1}$$

If we have an instance of the problem where $|\mathcal{L}|$-circulations are allowed, i.e. $\kappa \geq |\mathcal{L}|$, this lower bound can be obtained by a single circulation that contains all lines as many times as their frequency dictates. Thus all such instances are YES-instances.

If we are only allowed to used fixed circulations, i.e. $\kappa = 1$, the only way to cover all lines is to use a fixed circulation for each line. In this case an instance is a

YES-instance if and only if z is sufficient for the sum over all fixed circulations:

$$z \geq \sum_{l \in \mathcal{L}} \left\lceil \frac{t_l \cdot f_l}{T} \right\rceil \tag{2.2}$$

Theorem 2.2. *Any instance with $\kappa \geq |\mathcal{L}|$ for which the condition of Equation 2.1 holds is a YES-instance. Any instance with $\kappa = 1$ is a YES-instance if and only if the condition of Equation 2.2 holds.*

As we noted earlier, circulations that do not contain too many lines are preferred as they have many advantages. However, the number of vehicles required with fixed circulations can be significantly greater than when any circulation is allowed. We can see this via application of the following general identity for sums over ceiling functions. Suppose we have a sequence $a_1, \dots a_n$ of n numbers with $n \geq 2$, then it is straightforward to show that

$$\sum_{i=1}^{n} \lceil a_i \rceil - \left\lceil \sum_{i=1}^{n} a_i \right\rceil \leq n - 1 \tag{2.3}$$

Thus, we can see that the difference between the lower bound of Equation 2.2 and Equation 2.1 can become as large as $|\mathcal{L}| - 1$. If we allow 2-circulations, we are able to halve the number of terms in Equation 2.2 which halves the worst case gap. It thus can be very beneficial to consider restrictions on the circulation κ that are small but strictly greater than one. Unfortunately, for any case with a fixed $\kappa \geq 3$, we can show that the resulting problem becomes NP-hard.

Theorem 2.3. *For any fixed $\kappa \geq 3$, the VCS-LP is NP-hard.*

Proof. We show this by reduction from 3-partition. Let $S = \{s_1, s_2, \dots, s_m\}$ be a set of integers such that $\sum_{i=1}^{m} s_i = \frac{m}{3}B$ and $\forall 1 \leq i \leq m : \frac{B}{4} < s_i < \frac{B}{2}$. A 3-partition instance is a YES-instance if it is possible to partition S into $n = \frac{m}{3}$ triplets $S_1, S_2, \dots S_n$ such that each triplet sums to B.

For the case where $\kappa = 3$, we reduce the 3-partition instance to a line graph $L = (V, \mathcal{L})$ where we have a central hub station $v_0 \in V$ and m external stations $v_1, \dots, v_m \in V$. This line plan must be periodically operated in a period of $T = B$ time units. Furthermore we have a set of m lines where $l_i \in \mathcal{L} = \{v_0, v_i\}$, with frequency $f_i = 1$ and a round trip and total time equal to s_i. As the sum of the round trip times is exactly mB, the only way to execute this line plan with m vehicles is to have every circulation take B time. Otherwise, at least one circulation will require two vehicles. Thus we set $z = m$. If the 3-partition instance is a YES-instance, we can use the triplets to create 3-circulations with this precise property. If the 3-partition instance is a NO-instance, we can not nicely divide the lines over 3-circulations and thus need at least one additional vehicle.

For cases where $\kappa > 3$, we scale the 3-partition instance by setting $s'_i = 4 \cdot \kappa \cdot s_i$ and $B' = (\kappa - 3) + 4 \cdot \kappa \cdot B$. This way we make sure that $\frac{B'}{4} > \kappa - 3$. We now introduce a set of $\kappa \cdot n$ lines where lines $l_1, \ldots, l_m$ have round trip times $s'_1, \ldots, s'_m$ and lines $l_{m+1}, \ldots, l_{\kappa \cdot n}$ all have round trip time of 1. We define $1 + \kappa$ terminal stations and let lines i connect terminal stations $\{v_0, v_i\}$ in the same structure as the $\kappa = 3$ case. By construction, only circulations that consist of $\kappa - 3$ lines with round trip time 1 and three other lines can sum up to B'. This way it is enforced that it is a YES-instance if and only if the 3-partition instance is a YES-instance.

□

2.4 Fixed and Combined Circulations

Of special interest is the case of $\kappa = 2$ where we are only allowed to have fixed and combined circulations, as this gives us some flexibility to decrease the number of vehicles required to operate the line plan, while we still keep the number of lines in a circulation low. Since all finite cases with $\kappa > 2$ are NP-hard, the $\kappa = 2$ case is also of particular interest from a theoretical perspective.

For the remainder of this section, we restrict ourselves to instances of VCS-LP where the frequency of each line is 1. Although this restriction may seem unrealistic, we can approximate instances with higher frequencies either by splitting up a line l with a frequency higher than 1 into f_l lines with frequency 1 and the same characteristics or by increasing the round trip times as a function of the frequencies. The straightforward approach increases the round trip times based on the frequency, i.e. the new round trip time becomes $f_l \cdot t_l$. More sophisticated approaches can add slack to model the periodicity in more detail in order to increase the probability that a regular timetable exists, possibly at the cost of requiring more vehicles to execute the line plan.

In case $\kappa = 2$ and $f_l = 1$ for all lines, we can represent an instance of VCS-LP by a *circulation graph* $G = (\mathcal{L}, E)$. In this graph, the lines are the vertices and the edges are the circulations. The set of edges consists of edges for the fixed circulations E_1 and of the 2-circulations E_2, thus $E = E_1 \cup E_2$. The set E_1 contains a self loop for every line $l_i \in \mathcal{L}$. The set E_2 contains an edge between two lines l_i and l_j if they have a common terminal station. To ease notation, we denote a circulation $\{l_i\} \in E_1$ simply as l_i.

An example of a VCS-LP instance represented by a circulation graph is given in Figure 2.1. On the circulation graph, we also depict the k_c value of each circulation c. For the self loops, these values are depicted inside the nodes to make the graph more clear. For example, line 3 has a round trip time of 70 minutes, so $k_3 = \lceil \frac{70}{60} \rceil = 2$. Line 4 also needs 2 vehicles if it is performed in a fixed circulation, but if lines 3 and 4 are combined only 3 vehicles are required to operate both lines since $k_{34} = \lceil \frac{70+100}{60} \rceil = 3$.

Using the circulation graph, we can formulate the following optimization problem for VCS-LP:

$$\nu(\mathcal{L}) = \min_{\theta} \quad \sum_{c \in E} \theta_c \tag{2.4}$$

$$\text{s.t.} \quad \frac{1}{k_l}\theta_l + \sum_{c \in E_2 | l \in c} \frac{1}{k_c}\theta_c = 1 \qquad \forall l \in \mathcal{L}, \tag{2.5}$$

$$\theta_c \geq 0 \text{ and integer} \qquad \forall c \in E, \tag{2.6}$$

The objective is to minimize the number of used vehicles. We now consider how to rewrite this problem to a maximization problem where the goal is to maximize the number of *saving* circulations. First note that we can rewrite the summation in the

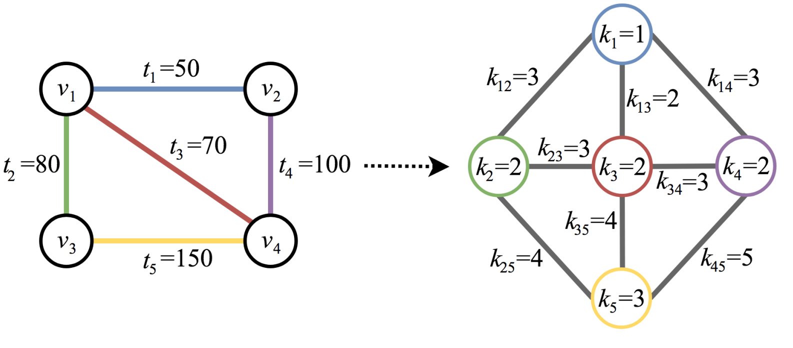


Figure 2.1: Example of a line graph with round trip times and the corresponding circulation graph. The cycle time T is 60.

objective of Equation 2.4 by splitting the summation over E into summations over E_1 and E_2, and that by definition a summation over E_1 is equal to a summation over $\mathcal{L}$. We rewrite the objective as follows:

$$\min_{\theta} \sum_{l \in \mathcal{L}} \theta_l + \sum_{c \in E_2} \theta_c \tag{2.7}$$

In the next step we make use of Constraints (2.5), which state that a line is included in a sufficient number of circulations. As these constraints imply that $\theta_l = k_l - \sum_{c \in E_2 | l \in c} \frac{k_l}{k_c} \theta_c$, we can substitute the left term to obtain

$$\min_{\theta} \sum_{l \in \mathcal{L}} \left(k_l - \sum_{c \in E_2 | l \in c} \frac{k_l}{k_c} \theta_c \right) + \sum_{c \in E_2} \theta_c \tag{2.8}$$

Note that the double summation over $l \in \mathcal{L}$ and $c \in E_2 | l \in c$ can also be written as a double summation over $c \in E_2$ and $l \in c$. Rewriting the double summation and reshuffling terms gives:

$$\sum_{l \in \mathcal{L}} k_l + \min_{\theta} \sum_{c \in E_2} \left(\theta_c - \sum_{l \in c} \frac{k_l}{k_c} \theta_c \right) \tag{2.9}$$

In the last step we factor out $\frac{\theta_c}{k_c}$ and Equations 2.4 – 2.6 are written as:

$$\nu(\mathcal{L}) = \sum_{l \in \mathcal{L}} k_l + \min \left\{ \sum_{c \in E_2} \left[k_c - \sum_{l \in c} k_l \right] \frac{\theta_c}{k_c}, \text{ s.t. } (2.5) - (2.6) \right\} \tag{2.10}$$

If we now apply Equation 2.3, it can be seen that $\left[k_c - \sum_{l \in c} k_l\right]$ equals either -1 or 0. If the term is -1, we call the circulation *saving*, otherwise we call it *non-saving*. For example, in Figure 2.1, circulation $\{3, 4\}$ is saving, while circulation $\{1, 2\}$ is non-saving. If we let the set of all saving circulations be denoted as E_2^S, we have that $\nu(\mathcal{L}) = \sum_{l \in \mathcal{L}} k_l - \sigma(\mathcal{L})$ where $\sigma(\mathcal{L})$ is the *savings problem* defined as follows:

$$\sigma(\mathcal{L}) = \max_{\theta} \left\{ \sum_{c \in E_2^S} \frac{\theta_c}{k_c} \text{ s.t. } (2.5) - (2.6) \right\} \tag{2.11}$$

As such, it can be observed that minimizing the number of vehicles is equivalent with maximizing the savings over a vehicle schedule that only uses fixed circulations. In the remainder of this section, we use this observation to give a proof of the NP-hardness of VCS-LP with $\kappa = 2$ and $f_l = 1$ for all lines, and to develop an exact algorithm and an $\frac{16}{15}$-approximation algorithm.

2.4.1 NP-hardness

Our proof depends on the fact that we can construct an arbitrary circulation graph from the line graph if it is accompanied by auxiliary restrictions on which 2-circulations are allowed, which are not part of the formal input to VCS-LP. This can be seen from the fact that a star-shaped line graph translates to a complete circulation graph, since we can combine all pairs of lines. If we can provide auxiliary restrictions on which combinations can be combined into circulations and which not, we have complete control over the structure of the circulation graph. In practice, such restrictions are realistic, since the possibility to combine lines into circulations does not only depend on the lines having a shared terminal station, but also on the precise layout of the infrastructure at the terminal station, and whether there exists a type of rolling stock that is able to operate both lines.

Theorem 2.4. *VCS-LP with $\kappa = 2$ and auxiliary restrictions on which 2-circulations are allowed is NP-hard.*

Proof. Our proof is based on a reduction from the NP-complete NUMERICAL 3-DIMENSIONAL MATCHING problem (Garey and Johnson, 1979) to an instance of VCS-LP expressed by means of a circulation graph. Since each circulation graph can be generated based on a line graph with auxiliary restrictions, this is sufficient to prove the theorem.

The inputs to the N3DM are three multisets of integers X, Y, Z, each containing k elements, and a bound b. An N3DM instance is a YES-instance if there exist k disjoint triples (x, y, z) such that $x + y + z = b$ holds for every triple.

We transform this to the following instance of VCS-LP. For every element in X, Y and Z we create three lines in $\mathcal{L}$, all with $k_l = 2$. The three lines can be combined in saving circulations ($k_c = 3$) such that they form a triangle. One of the three lines serves as the *connect line*, the other lines are referred to as *dummy lines*. For every triple (x, y, z) that sums up to b (all such triples can be found in polynomial time), a *triple line* is created with $k_l = 1$, which can be combined in non-saving circulations ($k_c = 3$) with the connect-lines corresponding to x, y and z. Letting μ denote the number of triples that sum up to b, the resulting VCS-LP instance has $9k + \mu$ lines.

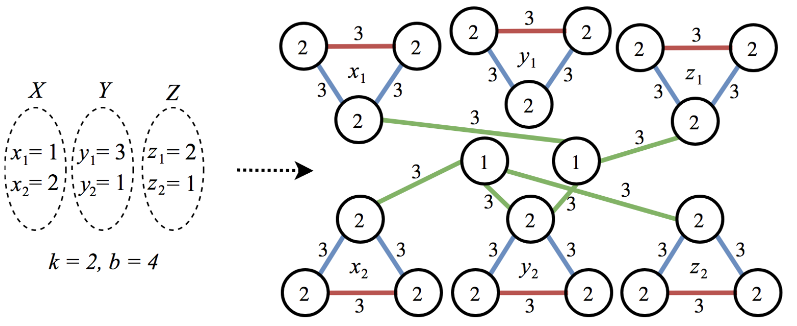


Figure 2.2: Transformation of a N3DM instance to a VCS-LP instance represented by a circulation graph.

We call the generated instance a YES-instance, if the number of required vehicles is at most $14k + \mu$, equivalent to a saving $\sigma(\mathcal{L})$ of at least $4k$. In Figure 2.2, we visualized this transformation for a small N3DM instance.

We claim that the constructed VCS-LP instance is a YES-instance if and only if the N3DM instance is a YES-instance.

(if) Each disjoint triple (x, y, z) can be used to generate a saving of 4 by assigning 1 vehicle to each of the 3 circulations combining the triple line with the connect lines (green in Figure 2.2, 1 vehicle to each of the 6 circulations combining the triples lines with the dummy lines (blue) and 2 vehicles to each of the circulations combining dummy lines (red). In every triangle, this gives a saving of $\frac{4}{3}$, so the total saving generated by every disjoint triple equals 4. As such, if there are k disjoint triples, the total saving is $4k$ and VCS-LP instance is indeed a YES-instance.

(only if) Since only the combined circulations in the triangle are saving, if the instance is a YES-instance, every triangle must generate a saving of $\frac{4}{3}$. This implies that in every triangle, the circulations combining the triple lines and dummy lines are assigned 1 vehicle (blue in Figure 2.2) and the circulations combining dummy lines are assigned 2 vehicles (red). As a consequence, for every connect line, one of the circulations connecting the line with a triple line must be assigned 1 vehicle (otherwise the connect line would not be covered entirely). Next, note that a triple line cannot be partially performed by combined circulations. Hence, if one of the circulations combining a certain triple line and a connect-line is assigned a vehicle, all three such circulations must be assigned a vehicle. So, as every connect line is included in exactly one circulation with a triple line, and as every triple line is included in either zero or three combined circulations, there must be k triple lines that are connected with 3 connect lines. Clearly, the associated triples in the N3DM instance must be disjoint, hence the N3DM instance must also be a YES-instance. □

2.4.2 The Strict Variant

We define the *strict* version of VCS-LP to state that each circulation c is either not executed at all, or executed by exactly k_c vehicles. We will now show that the strict version with $\kappa = 2$ and all frequencies equal to 1 can be solved exactly using an approach based on matching.

As in the non strict version, we have the relation that the minimum number of vehicles required under the strictness assumption, denoted as $\bar{\nu}(\mathcal{L})$, equals $\sum_{l \in \mathcal{L}} k_l - \bar{\sigma}(\mathcal{L})$, where $\bar{\sigma}(\mathcal{L})$ denotes the strict savings problem, obtained by replacing $\frac{\theta_c}{k_c}$ with the binary variable γ_c in the regular savings problem $\sigma(\mathcal{L})$:

$$\bar{\sigma}(\mathcal{L}) = \max_{\gamma} \quad \sum_{c \in E_2^S} \gamma_c \tag{2.12}$$

$$\text{s.t.} \quad \gamma_l + \sum_{c \in E_2 | l \in c} \gamma_c = 1 \quad \forall l \in \mathcal{L}, \tag{2.13}$$

$$\gamma_c \in \{0, 1\} \qquad \forall c \in E, \tag{2.14}$$

Constraints (2.13) now state that a line is either operated with a fixed circulation, or using one of the combined circulations. Since the objective does not contain the γ_c variables for the fixed circulations anymore, the γ-variables for these circulations can be viewed as slack variables for the Constraints (2.13). Furthermore, since the non-saving circulations have zero contribution to the objective, there always exists an optimal solution that does not contain any non-saving circulations. As a consequence, Constraints (2.13) can be rewritten as:

$$\sum_{c \in E_2^S | l \in c} \gamma_c \leq 1 \quad \forall l \in \mathcal{L} \tag{2.15}$$

Since the circulations in E_2^S contain precisely two lines, the resulting formulation is equivalent to a matching problem where we have to maximize the number of selected saving circulations. Thus, we can compute $\bar{\nu}(\mathcal{L})$ by computing the maximum matching in the graph that only contains the edges from E_2^S.

Theorem 2.5. *The strict VCS-LP with $\kappa = 2$, $f_l = 1$ for each line $l \in \mathcal{L}$ is solvable in polynomial time.*

2.4.3 The Matching Approximation

Since the strict VCS-LP provides solutions that are also feasible for the regular problem, it can be applied as a heuristic. In this section we derive an approximation guarantee for this heuristic. Our approximation results are based on the observation that the savings problem is a maximization problem and that the linear programming relaxations of the savings problem and its strict version, denoted as $\sigma^{\mathrm{LP}}(\mathcal{L})$ and $\bar{\sigma}^{\mathrm{LP}}(\mathcal{L})$ respectively, are equal. This easily follows from the fact that the strict version is obtained from the non strict version by performing a linear variable substitution, which does not influence the value of the linear relaxation.

Theorem 2.6. *For any line plan it holds that $\bar{\nu}(\mathcal{L}) - \nu(\mathcal{L}) \leq \left\lfloor \frac{|\mathcal{L}|}{6} \right\rfloor$. Furthermore, there exists an instance that attains this bound.*

Proof. Note that we can consider the difference between the savings instead of the difference between the number of vehicles, as $\bar{\nu}(\mathcal{L}) - \nu(\mathcal{L}) = \sum_{l \in \mathcal{L}} k_l - \bar{\sigma}(\mathcal{L}) - \sum_{l \in \mathcal{L}} k_l + \sigma(\mathcal{L}) = \sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L})$. For any graph it holds that the difference between the value of the maximum fractional matching and the value of the maximum matching is at most $\frac{n}{6}$, with n the number of nodes (Choi et al., 2016). This implies that $\bar{\sigma}^{\mathrm{LP}}(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) \leq \frac{|\mathcal{L}|}{6}$. Since the linear programming relaxations of the savings problem and its strict version are equal, it follows that $\sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) \leq \sigma^{\mathrm{LP}}(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) = \bar{\sigma}^{\mathrm{LP}}(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) \leq \frac{|\mathcal{L}|}{6}$. Furthermore, the right hand side of this equation can be rounded down since the difference between savings must be integral.

To show that this bound is tight, consider the circulation graph depicted in Figure 2.3. The example contains $2k + 1$ triangles, where k is a positive integer, and one central node connected to all triangles. The circulations between the lines in the triangles are saving. The value $\bar{\sigma}(\mathcal{L})$ is equal to the size of the maximum matching in the graph induced by all saving circulations, i.e. the graph with only the $2k + 1$ triangles. Since we can pick only one circulation in every triangle, we have that $\bar{\sigma}(\mathcal{L}) = 2k + 1$. The optimal unrestricted solution is as follows. We can assign 1 vehicle to all green circulations, k vehicles to all blue circulations and $k + 1$ vehicles to all red circulations. The objective attained with this solution equals $\sigma(\mathcal{L}) = \sum_{c \in E_2^S} \frac{\theta_c}{k_c} = (2k + 1)(\frac{k+k+k+1}{2k+1}) = 3k + 1$.

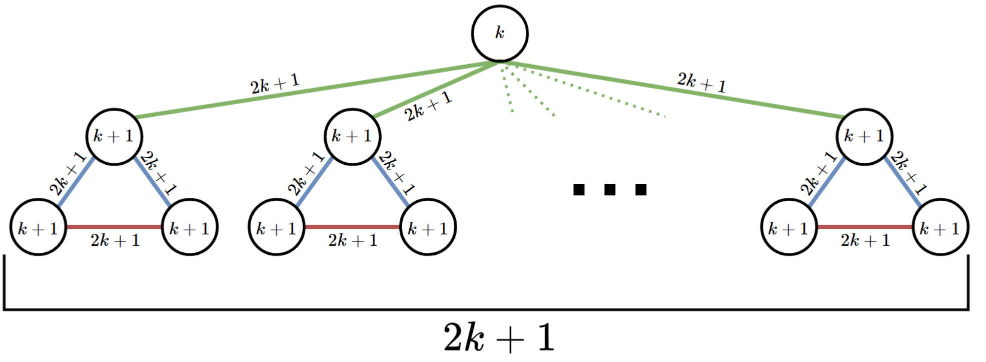


Figure 2.3: Circulation graph of the instance used to show that the bound of Theorem 2.6 is tight.

Comparing the two objectives, we have that $\sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) = k$. As the bound equals $\left\lfloor \frac{|\mathcal{L}|}{6} \right\rfloor = \left\lfloor \frac{3(2k+1)+1}{6} \right\rfloor = \lfloor k + \frac{4}{6} \rfloor = k$, this circulation graph attains the bound for every k. $\square$

Lemma 2.7. *If $\sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) = k$, the circulation graph contains at least $2k+1$ disjoint odd cycles of saving circulations.*

Proof. Every vertex x of the fractional matching polytope is half-integral, i.e. $x_e \in \{0, \frac{1}{2}, 1\}$ (Balinski, 1965). Moreover, the edges with $x_e = 1$ form a matching and the set of edges with $x_e = \frac{1}{2}$ form a set of disjoint odd cycles. If the optimal solution to the fractional matching problem contains ω such odd cycles, the difference between the size of the fractional matching and the size of the matching equals $\frac{\omega}{2}$, as a fractional matching in a single odd cycle can improve the objective only by $\frac{1}{2}$ compared to an integer matching in the same cycle. As such, if $\sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) = k$, we certainly have that $\bar{\sigma}^{\mathrm{LP}}(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) \geq k$, implying that the circulation graph contains at least $2k$ disjoint odd cycles of saving circulations.

We prove that the number of odd cycles of saving circulations should be one more than $2k$ by contradiction. As we have already established that the number of odd cycles is at least $2k$, we assume that $\sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L}) = k$ while there are exactly $2k$ odd cycles. Note that an odd cycle cannot contribute strictly more than $\frac{1}{2}$ to the difference between $\sigma(\mathcal{L})$ and $\bar{\sigma}(\mathcal{L})$, as this violates the fact that fractional matching is the relaxation of the savings problem. Hence, it must hold that every odd cycle contributes exactly $\frac{1}{2}$ to the difference between savings.

However, we will now show that for VCS-LP, every odd cycle can increase the difference between $\sigma(\mathcal{L})$ and $\bar{\sigma}(\mathcal{L})$ with strictly less than $\frac{1}{2}$. This is the case since it is not possible to select all circulations in an odd cycle of saving circulations in the circulation graph with value $\frac{1}{2}$. To see this, note that if circulation $c = \{l, m\}$ contributes $\frac{1}{2}$ to the objective of VCS-LP, this implies that k_c is even (e.g. $k_c = 4$ and $\theta_c = 2$). Furthermore, since $k_c = k_l + k_m - 1$ (the circulation is saving), it must hold that k_l and k_m have a different parity (one of them is odd, the other even). As such, if we do have an odd cycle in which every circulation is selected with value 1/2, there must exists a 2-coloring of the vertices of the cycle. Since this is clearly not possible for an odd cycle, we reach a contradiction. □

Theorem 2.8. *For any line plan it holds that* $\frac{\bar{\nu}(\mathcal{L})-\nu(\mathcal{L})}{\nu(\mathcal{L})} \leq \frac{1}{15}$. *Furthermore, there exists an instance where this bound is attained. This implies that* $\bar{\nu}(\mathcal{L})$ *is a* $\frac{16}{15}$*-approximation algorithm for VCS-LP with* $\kappa = 2$ *and all frequencies* 1.

Proof. First note that

$$\max \frac{\bar{\nu}(\mathcal{L}) - \nu(\mathcal{L})}{\nu(\mathcal{L})} = \max \frac{\sigma(\mathcal{L}) - \bar{\sigma}(\mathcal{L})}{\sum_{l \in \mathcal{L}} k_l - \sigma(\mathcal{L})}. \tag{2.16}$$

It follows from Lemma 2.7 that for a given value of $\bar{\nu}(\mathcal{L}) - \nu(\mathcal{L}) = k$, the worst case ratio must be attained by using $2k+1$ cycles of 3 vertices (more or larger cycles only lead to larger values in the denominator). Next to that, for a fixed numerator it is easily seen that the denominator of the ratio is minimized by letting a single node connect all the cycles. This implies that for a given value of $\bar{\nu}(\mathcal{L}) - \nu(\mathcal{L}) = k$, the instance in Figure 2.3 gives the worst case ration. Maximizing over k gives

$$\max_{k \in \mathbb{N}} \frac{k}{(2k+1)(3k+3) + k - (3k+1)} = \frac{1}{15}, \tag{2.17}$$

with the maximum being attained at $k = 1$. □

2.4.4 An Exact Algorithm for Bounded Treewidth

In this section we consider how to solve VCS-LP exactly with $\kappa = 2$ where the circulation graph has a low *treewidth*. Treewidth is a graph property that was introduced by Robertson and Seymour (1986) that, informally, indicates how "similar to a tree" the graph is. Many problems that are NP-hard on general graphs, such as independent and dominating set, are solvable in polynomial time if the treewidth of the input graph is bounded by a constant.

Formally, the treewidth of a graph G is the smallest *width* for which there a exists *tree-decomposition* of G with that width. A tree-decomposition of an undirected graph $G = (V, E)$ is a tree $\mathcal{T}$, where each node $n \in \mathcal{T}$ is associated with a bag $X_n \subseteq V$ and these two properties hold: (1) the endpoints of each edge should occur simultaneously in at least one bag, i.e. for each edge $\{v, w\} \in E$ there is a node $n \in \mathcal{T}$ such that both $v \in X_n$ and $w \in X_n$, and (2) for each vertex $v \in V$, all nodes n for which the associated bag contains v, i.e. $v \in X_n$, are a connected subtree of $\mathcal{T}$. The width of a tree decomposition $\mathcal{T}$ is equal to $\max_{n \in \mathcal{T}} |X_n| - 1$. Although finding a tree-decomposition of the treewidth of a graph is NP-hard, there is a linear time algorithm for any fixed width (Bodlaender, 1996). Furthermore, there are algorithms that are able to efficiently find good tree decompositions in practice, e.g. Tamaki (2017).

One versatile approach in the design of algorithms that exploit bounded treewidth is to perform *dynamic programming* on the tree-decomposition. Central to this idea is the interpretation of every bag X_n in the tree-decomposition as a *graph separator*, which means that if we remove the nodes in the bag from the graph, the graph splits up in different parts. By moving up the the tree of the tree-decomposition, the algorithm looks at the current bag of vertices of the graph, which separates the part of the graph that is already processed by the algorithm from the part still needs to be processed, with the invariant that all connections between the processed and unprocessed parts of the graph must go through the current bag. In each state of the algorithm a state table is constructed for (combinations of) vertices in the bag associated with the current node in the tree, under the assumption that optimal decisions were made for the processed part of the graph.

A helpful way to design a dynamic programming algorithm based on the tree-decomposition is to assume it is a *nice* tree-decomposition (Bodlaender and Kloks, 1996). Such a tree-decomposition has a root and as a consequence the order in which the dynamic programming algorithm visits the nodes of the tree is fixed: we start at the leaves and moves up to the root. In our description of the algorithm, we say that the algorithm moves from *parent* nodes to *child* nodes. A nice tree-decomposition distinguishes four types of nodes: *create* nodes which corresponds to leaves in the tree that only have a single vertex in their bag, *introduce* nodes which introduce a single new vertex into the bag of their parent, *forget* nodes which remove a single vertex from the bag of their parents and *join* nodes which have the same bag as their two parents. Thus in a nice tree-decomposition join nodes have two parents, leaf nodes have no parents and the other nodes have a single parent. There exists a linear time algorithm that converts any tree-decomposition into a nice tree-decomposition with $O(|V|)$ nodes and the same width (Bodlaender and Kloks, 1996).

Our algorithm adopts this approach. For each bag a state table is constructed for all partial covers of the lines in the current bag, based on the possible combinations of values of the left hand sides of Constraints (2.5). In every step we are only allowed to increase the θ_c values and thus the coverage of each line. Since each circulation $c \in E$ can only be selected an integer number of times, there is a finite number of fractions $\sum_{c\in\delta(l)} \frac{\theta_c}{k_c}$ that lie in the range $[0, 1]$ for each line l. The total number of combinations of values of the left hand sides of Constraints (2.5) for a particular set of lines is at most the product of the possible number of values for the individual constraints. This gives us an upper bound on the number of states we need to maintain in a state table when we enumerate the optimal partial covers for that bag. An upper bound on the number of possible fractional values for the left hand side of the constraint of a line l is denoted by ρ_l. One (crude) upper bound for this can be computed as $1 + \prod_{c\in\delta(l)} k_c$. Note that if the circulations are short enough compared to T, which they often are in practice, this number will typically be small.

A single state in the state table for a bag X_n assigns a fraction q_l to each line $l \in X_n$ where we have $q_l \in \{0, \frac{1}{\rho_l}, \ldots, \frac{\rho_l - 1}{\rho_l}, 1\}$. The state table for a node $n \in \mathcal{T}$ maps each state to the minimum number of vehicles required to reach this state. If the algorithm generates the same state multiple times, it is sufficient to store only the state for which the minimum number of vehicles was required to reach the state. If we introduce ρ as the maximum ρ_l for all lines $l \in \mathcal{L}$, the size of the table is for a single bag is $O(\rho^{w+1})$.

To conclude the description of the algorithm, we describe how the state table for each of the four types of nodes in the tree-decomposition can be computed based on the state tables of the parent(s). In a *start* node, only a single line l is introduced and thus only a fixed circulation is considered, modeled by a self loop. This loop can be used at most ρ_l times and may not be used at all. These state tables can be generated in $O(\rho)$ time. In an *introduce* node, a new line l is introduced to the state table of the parent. Due to this introduction, we have to expand all states in the parent table with all possible multiplicities of the circulations in $\delta(l)$, including multiplicities of zero. As the state table of the parent contains $O(\rho^w)$ states and there are $O(w)$ circulations that connect the new line l to lines in the current bag, each of which can be used at most ρ times in the expansion, we can construct the state table for an introduce node in $O(w \cdot \rho^{w+1})$ time. In a *forget* node, a line l is removed from the state table of parent. This means that from this step onward, that particular line is in the set of lines that have been processed by the algorithm and thus we must make sure that it is fully covered. This can be achieved by removing all states from the parent table where the q_l of this line is not equal to 1, as the removal of line l implies that these states are infeasible. This can be done in $O(\rho^{w+1})$ time as we only have to filter the state table of the parent. Finally, in a *join* node we have two parent nodes with the same bag, but potentially different states. We construct the new state table by either taking the state and its associated number of vehicles from one of the two parents' state table or by taking a state from one table and combining it with a state from the second table, by adding up the q_l's of both states and adding up the number of vehicles of the two states. These combinations are only worth considering if none of the resulting q_l's exceeds 1. As both parent tables can be of size ρ^{w+1}, there are ρ^{2w+2}' combinations that can be explored in this step. This means the table for a join node can be computed in $O(\rho^{2w+2})$ time.

When the algorithm is done, we can find the optimal solution to VCS-LP in the root node at the state where all q_l's are equal to one. Recall that the size of the tree-decomposition is $O(|\mathcal{L}|)$ and each node in this decomposition can be processed in $O(\rho^{2w+2})$ time.

Theorem 2.9. *VCS-LP with $\kappa = 2$ and auxiliary constraints on the allowed circulations can be solved in $O(n\rho^{2w+2})$ time where $n = |\mathcal{L}|$, w is the tree-width of the circulation graph and $\rho = \max_{l \in \mathcal{L}} \prod_{x \in \{k_c | c \in \delta(l)\}} x$.*

2.5 Conclusion

We have shown that Vehicle Circulation Scheduling on a Line Plan is NP-hard for any finite restriction on the number of lines that can be included in a circulation (κ) greater than two. For the $\kappa = 2$ case we need to make the (realistic) additional assumption that we have auxiliary restrictions on which lines can be combined in order to prove NP-hardness. For the $\kappa = 2$ case we show that if we can cover each line by at most one unique circulation, a matching algorithm yields the optimal solution. This solution provides a $\frac{16}{15}$-approximation in case multiple circulations can be used. We also provide an exact algorithm that can exploit low treewidth of the circulation graph, and a low number of vehicles required per circulation. For future research, it makes sense to combine these algorithms with the line planning process to see if it can help to make line plans that allow better vehicle schedules. Furthermore, it is interesting to consider whether algorithms exist that are useful for cases where κ is small, but greater than two. Finally, it is still an open question whether the $\kappa = 2$ case without auxiliary restrictions is NP-hard.

Chapter 3

Integrated Periodic Timetabling and Vehicle Circulation Scheduling

This chapter is based on Van Lieshout, R.N. (2021). Integrated Periodic Timetabling and Vehicle Circulation Scheduling. Transportation Science 55(3), 768–790.

3.1 Introduction

The timetable is the foundation of any public transportation system. It influences the travel times of passengers and the paths they take to travel from their origins to their destinations. Furthermore, the extent to which disturbances such as delays propagate through the system also highly depends on the timetable. All in all, it is widely accepted that the timetable determines the attractiveness of the public transportation system from the passengers' point of view. Because of the importance of a good timetable, and the fact that finding such a timetable is a non-trivial exercise, the timetabling problem has triggered the interest of many researchers in the field of public transport optimization. However, the impact the timetable has on the costs of the operator is rarely considered in timetabling models.

In this chapter, we consider *periodic timetabling*. In a periodic timetable, the schedule of a single period (e.g. one hour) is repeated throughout the day. Periodic timetables are favoured by passengers as they can easily memorize the departure times of services (e.g. a traveller can always take the train at xx:06). Furthermore, operators that use a periodic timetable only need to plan services for one period, after which they can repeat the timetable during the entire day.

Most periodic timetabling models are based on the Periodic Event Scheduling Problem (PESP). Serafini and Ukovich (1989) introduce the PESP and prove it is NP-complete. In its original form, the PESP is a feasibility problem. Later, researchers have started to consider optimization variants of the PESP, where the objective typically is to minimize passenger travel time or to maximize the robustness of the timetable. The PESP can be solved using constraint programming or SAT solvers (Gattermann et al., 2016; Großmann et al., 2012; Kroon et al., 2009; Schrijver and Steenbeek, 1993), the modulo network simplex heuristic (Borndörfer et al., 2016; Goerigk and Schöbel, 2013; Nachtigall and Opitz, 2008) or mixed-integer programming based approaches (Goerigk and Liebchen, 2017; Liebchen, 2008; Maróti, 2017; Polinder et al., 2019).

The goal of this chapter is to find timetables that are both attractive for the passengers and for the operator. Specifically, we analyze the trade-off between the travel time of passengers and *the number of vehicles required to operate the timetable*. To illustrate the impact a timetable has on the required number of vehicles, consider the two timetables in Figure 3.1. The timetable on the left induces two vehicle circulations, which are cyclic sequences of trips performed by the same vehicles, both

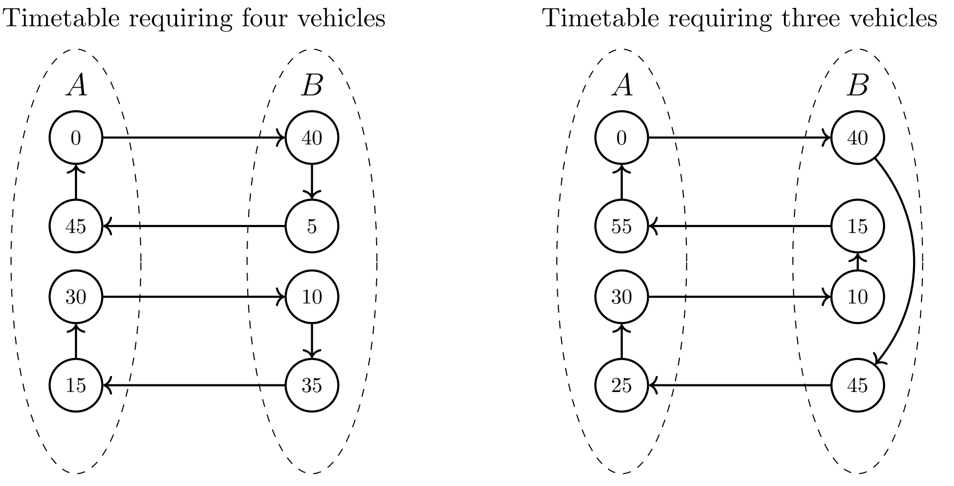


Figure 3.1: Two timetables for a line (A, B) with a frequency of 2 per hour and a driving time of 40 minutes.

requiring two vehicles. In contrast, the timetable on the right induces one vehicle circulation requiring three vehicles. Moreover, both timetables yield the same travel times for passengers. Therefore, as vehicles constitute a large proportion of the operator costs, the second timetable is clearly preferred over the first one.

Traditionally, timetabling and vehicle scheduling are performed sequentially: first the timetable is optimized to minimize travel times of passengers and subsequently the vehicles are scheduled to minimize operator costs. However, this is a greedy approach that leads to globally sub-optimal solutions. More recently, integrated public transport planning, where the goal is to find a good overall line plan, timetable and vehicle schedule, has seen increasing attention (Burggraeve et al., 2017; Kaspi and Raviv, 2013; Pätzold et al., 2017; Schöbel, 2017; Van Lieshout and Bouman, 2018). The potential of jointly optimizing the timetable and vehicle schedule has already been demonstrated for non-periodic timetables by Desfontaines and Desaulniers (2018), Fonseca et al. (2018), Ibarra-Rojas et al. (2014) and Schmid and Ehmke (2015). Even when only limited modifications to an initial timetable are allowed, large savings in operational costs can be obtained without dramatic increases in travel times.

In contrast, only a few contributions consider the integration of periodic timetabling and vehicle scheduling. Peeters (2003) notes that if the operated vehicle circulations are fixed a priori, the number of required vehicles can easily be counted within the PESP. Furthermore, the author shows that choosing the circulation can be incorporated in the PESP (without introducing auxiliary variables) for the case that at most two services terminate at each station. However, this only works under the *sufficient spread* assumption, which states that there is never more than one vehicle idle at a station. Liebchen and Möhring (2007) show that this assumption can be avoided when there are at most two services terminating at each station, at the cost of introducing an auxiliary binary variable. Kroon et al. (2013) are able to choose vehicle turnarounds within the PESP between an arbitrary number of services without leaving the PESP framework, but again require the sufficient spread assumption. Nührenberg (2016) proposes a general explicit matching approach that is also valid when the sufficient spread assumption does not hold, which requires extending the PESP with matching variables and constraints. In this model, both the timetable and the vehicle circulations are entirely determined. We refer to this completely integrated problem as the Vehicle Circulation Periodic Event Scheduling Problem (VC-PESP). However, the proposed formulation contains many big-M constraints, causing it to not perform well, even on medium-sized instances. Moreover, the author purely focuses on minimizing the number of required vehicles, without regard to the passengers' perspective.

In this chapter, we improve the formulation of Nührenberg (2016), in order to be able to tackle realistic instances of the VC-PESP and gain insight into the trade-off between the average travel time a timetable offers and the number of vehicles it requires. To do so, we first formalize and analyze the vehicle circulation scheduling problem (given a timetable), which, in contrast to traditional non-periodic vehicle scheduling (see e.g. Bunte and Kliewer (2009)), has not received much attention in the literature. Next, we use the properties of the vehicle circulation scheduling problem to develop a stronger formulation of the VC-PESP. In particular, to reduce the number of big-M constraints, we develop a modeling technique that computes the minimum turnaround time at stations in a contracted graph. In addition, we present new valid inequalities that bound the minimum turnaround time of vehicles, strengthening the linear programming relaxation. Finally, we prove that a greedy algorithm is optimal for the vehicle circulation scheduling problem and use this observation to develop symmetry-breaking inequalities.

Besides strengthening the formulation of the VC-PESP, we also extend the problem by considering restrictions on the number of lines that may occur in a single vehicle circulation. In practice, public transport operators prefer short circulations where vehicles perform only a single line or alternate between two lines, as such circulations are more easy to operate and avoid dependencies between different parts of the network. We show how such practical requirements can be incorporated.

Computational experiments with different instances based on the Dutch railway network demonstrate the effectiveness of the proposed improved formulation. When minimizing the number of required vehicles, the improved formulation finds timetables requiring fewer vehicles compared to the original formulation, reducing the average optimality gap from 5.0 to 2.6 percent. An analysis of the trade-off between the number of vehicles and the perceived travel time of timetables illustrates the benefit of integrating circulation scheduling within the timetabling problem: minimizing average travel time without taking vehicles into account results in relatively costly timetables. In more than half of the considered instances, our approach finds timetables that require fewer vehicles without any increase in the travel time. Moreover, if one permits a travel time increase of 0.1 percent, we are typically able to reduce the number of vehicles by about 10 percent. Furthermore, the experiments show that in most instances vehicle circulations consisting of at most two lines suffice for realizing very efficient timetables and only small decreases in travel time can be attained if longer vehicle circulations are allowed.

Summarizing, the main contributions of this chapter are (1) the analysis of the vehicle circulation scheduling problem, (2) the improvement of a mathematical formulation to efficiently solve periodic timetabling problems while controlling the number of required vehicles, (3) showing that neglecting vehicle circulations in the periodic timetabling problem will often lead to inefficient timetables; considerable savings in the number of vehicles can generally be achieved without compromising the passengers' perspective and (4), demonstrating that a large proportion of these savings can already be obtained with vehicle circulations consisting of at most two lines, which are attractive for operators.

The remainder of this chapter is structured as follows. In Section 3.2, we discuss the periodic timetabling problem and in Section 3.3, the vehicle circulation scheduling problem. In Section 3.4, we describe the mathematical formulation for the VC-PESP that is proposed in Nührenberg (2016). In Section 3.5, we analyze the computational complexity of the integrated problem. In Section 3.6, we present the improved formulation. In Section 3.7, we describe how restrictions on the number of lines in a circulation can be included in the formulation. In Section 3.8, we discuss the results of the conducted computational experiments. Finally, in Section 3.9, we conclude and describe future research directions.

3.2 Periodic Timetabling

In periodic timetabling, the goal is to determine departure and arrival times for a given set of services, such that all operational and service requirements are met. As the schedule repeats every period (e.g. one hour), it suffices to determine departure times and arrival times for a single period.

3.2.1 The Periodic Event Scheduling Problem

The most widely applied method to formulate periodic timetabling problems is through the Periodic Event Scheduling Problem (PESP), introduced in Serafini and Ukovich (1989). The input to the PESP is a directed network $N = (E, A)$ referred to as an *event-activity network*, a period or cycle time T and lower and upper bounds l_a and u_a for all $a = (e, f) \in A$. Every node $e \in E$ represents a repeating event that needs to be scheduled at some time $\pi_e \in [0, T)$ and every arc $a \in A$ represents an activity for which the duration should be between the specified bounds:

Definition 3.1 (PESP)**.** Given a period T, an event-activity network $N = (E, A)$ and lower and upper bounds l_a and u_a for all $a = (e, f) \in A$, the Periodic Event Scheduling Problem is to determine event times $\pi_e \in [0, T)$ for all $e \in E$, satisfying

$$(\pi_f - \pi_e - l_a) \bmod T \leq u_a - l_a \text{ for all } a = (e, f) \in A$$

or to conclude that no such schedule exists.

Serafini and Ukovich (1989) prove the NP-completeness of the PESP. Odijk (1997) shows the problem is NP-complete even for fixed $T \geq 3$ by a reduction from vertex coloring.

The PESP can be used to formulate a wide range of timetabling constraints. The events that need to be scheduled are the arrivals and departures of services at all stations. Activities are used to model various operational and service requirements of the timetable, covering e.g. driving and dwell times, minimum and maximum transfer times and minimum separation times. For a more comprehensive overview, we refer to Peeters (2003) and Liebchen and Möhring (2007).

3.2.2 Mixed Integer Programming Formulations

By introducing an integer variable p_a for all $a \in A$, the PESP can be formulated as the following mixed integer program:

$$
\begin{aligned}
\text{find} \quad & (\boldsymbol{\pi}, \mathbf{p}) && && (3.1)\\
\text{s.t.} \quad & l_a \leq \pi_f - \pi_e + T p_a \leq u_a && \forall a = (e,f) \in A, && (3.2)\\
& 0 \leq \pi_e \leq T && \forall e \in E, && (3.3)\\
& p_a \in \mathbb{Z} && \forall a \in A. && (3.4)
\end{aligned}
$$

In this formulation, the p_a variables take the role of the modulo operator. For example, consider an activity $a = (e, f)$ with $l_a = 5$ and $u_a = 15$ and let $T = 30$. If $\pi_f = 5$ and $\pi_e = 25$, p_a can take the value 1, such that $\pi_f - \pi_e + T p_a = 5 - 25 + 30 = 10 \in [5, 15]$.

The standard formulation of the PESP (3.1)-(3.4) has a very weak linear programming (LP) relaxation and does not perform well on reasonably sized instances. A formulation with fewer integer variables can be obtained by formulating the PESP in terms of the activity durations or *tensions* $x_a = \pi_f - \pi_e + T p_a$ and exploiting the *cycle periodicity* property:

Lemma 3.2 (Cycle Periodicity (Odijk, 1996))**.** *Consider a feasible PESP solution* $(\boldsymbol{\pi}, \mathbf{p})$ *and let* $x_a = \pi_f - \pi_e + T p_a$ *for* $a = (e, f) \in A$. *Let* C *denote a cycle in the event-activity network, with forward arcs* C^+ *and backward arcs* C^-. *It holds that*

$$\sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = q_C T \ \text{ for some integer } q_C.$$

Moreover, $a_C \leq q_C \leq b_C$*, with*

$$a_C = \left\lceil \frac{\sum_{a \in C^+} l_a - \sum_{a \in C^-} u_a}{T} \right\rceil \text{ and } b_C = \left\lfloor \frac{\sum_{a \in C^+} u_a - \sum_{a \in C^-} l_a}{T} \right\rfloor$$

Lemma 3.2 shows that the cycle periodicity property is a necessary condition for a feasible solution to the PESP. Odijk (1996) proves that it is in fact both necessary and sufficient, so if the property holds for all cycles, the solution is feasible. The final ingredient for the cycle periodicity formulation is asserted by Nachtigall (1994), who shows that if the periodicity property holds for the cycles in a carefully chosen subset of the set of all cycles, it actually holds for all cycles in the event-activity network.

Specifically, if the cycle periodicity property holds for all cycles in an *integral cycle basis* of the event-activity network, it holds for any cycle (Liebchen and Peeters, 2009).

From these results, it follows that the PESP can be formulated in terms of tension variables x_a for all arcs and periodicity variables q_C for all cycles in an integral cycle basis $\mathcal{B}$. The entire cycle periodicity formulation reads as follows:

$$
\begin{aligned}
\text{find} \quad & (\mathbf{x}, \mathbf{q}) && && (3.5)\\
\text{s.t.} \quad & l_a \leq x_a \leq u_a && \forall a \in A, && (3.6)\\
& \sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = q_C T && \forall C \in \mathcal{B}, && (3.7)\\
& a_C \leq q_C \leq b_C && \forall C \in \mathcal{B}, && (3.8)\\
& x_a \in \mathbb{R} && \forall a \in A, && (3.9)\\
& q_C \in \mathbb{Z} && \forall C \in \mathcal{B}. && (3.10)
\end{aligned}
$$

Constraints (3.6) impose the lower and upper bounds for the tensions. Constraints (3.7) model the periodicity of the timetable through the cycle periodicity property. Constraints (3.8) are not required for the correctness of the formulation but are included to strengthen the LP relaxation.

Assuming the network N is connected, the cycle periodicity formulation contains only $|\mathcal{B}| = |A| - |E| + 1$ integer variables, compared to $|A|$ in the standard formulation. The cycle periodicity formulation also performed better than the standard formulation in empirical studies, see e.g. Liebchen et al. (2008).

3.3 Vehicle Circulation Scheduling

Once the timetable is known, operators need to schedule the vehicles in order to cover all trips. This includes determining sequences of trips performed by vehicles, and in e.g. railway contexts also coupling and decoupling decisions in order to better meet passenger demand. In this chapter, we do not consider (de)coupling, but limit ourselves to *vehicle circulation scheduling*, which involves determining which trips are operated consecutively *within the periodic pattern*, under the assumption that a single vehicle suffices to cover a trip. As the timetable is periodic, every trip should have exactly one successor and one predecessor, which results in cycles of trips, referred to as circulations, that are performed by one or multiple vehicles. For example, if T is 60 minutes and a certain circulation takes 180 minutes, three vehicles are needed to operate this circulation. The vehicle circulation scheduling problem (VCSP) is to find the set of circulations that cover all trips and minimize the number of required vehicles. Note that if (de)coupling is allowed, the optimal solution to the VCSP minimizes the number of *vehicle compositions*, which is strongly related to, but not necessarily the same as the number of vehicles.

To model vehicle transitions between trips, we let $E_{\text{end}}(s) \subseteq E$ and $E_{\text{start}}(s) \subseteq E$ denote the ends and starts of trips at station s, and expand the event-activity network with *turnaround activities* $A_{\text{turn}}(s)$ linking end events $e \in E_{\text{end}}(s)$ with start events $f \in E_{\text{start}}(s)$. That is, if these events and activities were not present in the original network, we add them. An example is depicted in Figure 3.2.

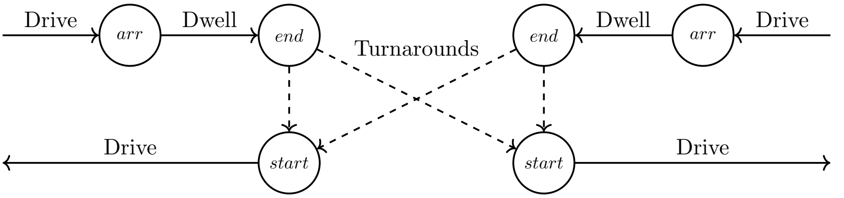


Figure 3.2: Event-activity network with turnaround activities (dashed arcs).

In general, turnaround activities could also involve a vehicle driving from one station to another without passengers (a deadhead trip). However, in this chapter we assume that there are only turnarounds between ends and starts of trips at the same station. Furthermore, we assume that all transitions between trip ends and trip starts at the same station are possible, such that the *turnaround graph* $G_{\text{turn}}(s) = (E_{\text{end}}(s), E_{\text{start}}(s), A_{\text{turn}}(s))$ is a complete bipartite graph for every station s. We let $E_{\text{end}}, E_{\text{start}}$ and A_{turn} denote the union of the sets $E_{\text{end}}(s), E_{\text{start}}(s)$ and $A_{\text{turn}}(s)$ over all stations, respectively. Finally, we incorporate the minimum turnaround time in the final dwell activity before the end event. This allows us to set $l_a = 0$ and $u_a = T$ for all $a \in A_{\text{turn}}$, which will simplify the analysis of the VCSP.

Let A_{veh} denote all drive and dwell activities in the event-activity network. Then, a vehicle circulation schedule is a set of directed cycles $\mathcal{C}$ in the event-activity network, consisting of vehicle activities and turnaround activities, such that all vehicle activities are covered exactly once. The cycles represent the sequences of trips performed by vehicles in the periodic pattern. As trips are connected by turnaround activities, there is a direct correspondence between the vehicle circulation schedule and the selected turnaround activities, denoted as $\mathcal{T}(\mathcal{C})$. Given a feasible timetable, the sum of the tensions of a vehicle circulation are an integer multiple of T by Lemma 3.2. Hence, for a given timetable specified in terms of the tensions $\mathbf{x}$, the number of vehicles required to operate the timetable using vehicle circulation schedule $\mathcal{C}$, denoted as $n(\mathbf{x}, \mathcal{C})$, equals

$$n(\mathbf{x}, \mathcal{C}) = \frac{1}{T} \sum_{c \in \mathcal{C}} \sum_{a \in c} x_a = \frac{1}{T} \left(\sum_{a \in A_{\text{veh}}} x_a + \sum_{a \in \mathcal{T}(\mathcal{C})} x_a \right).$$

For a given timetable, only the selected turnaround activities still need to be determined. To model the VCSP, let $\delta^+(e)$ denote the set of turnaround activities leaving $e \in E_{\text{end}}$ and let $\delta^-(e)$ denote the set of turnaround activities entering $e \in E_{\text{start}}$. Next, we introduce the binary decision variable y_a for $a \in A_{\text{turn}}$ which is equal to 1 if a is selected and 0 otherwise. Then, for a given timetable $\mathbf{x}$ we can find a vehicle circulation schedule that minimizes the number of required vehicles by solving the following problem:

$$\min \sum_{a \in A_{\text{turn}}} x_a y_a \tag{3.11}$$

$$\text{s.t.} \sum_{a \in \delta^+(e)} y_a = 1 \quad \forall e \in E_{\text{end}}, \tag{3.12}$$

$$\sum_{a \in \delta^-(e)} y_a = 1 \quad \forall e \in E_{\text{start}}, \tag{3.13}$$

$$y_a \in \{0, 1\} \quad \forall a \in A_{\text{turn}}. \tag{3.14}$$

The objective (3.11) is to minimize the total turnaround time. Constraints (3.12) and (3.13) ensure that every end event has a successor and every start event has a predecessor. As already observed by Orlin (1982) and Serafini and Ukovich (1989), it follows that finding a vehicle circulation schedule requiring the fewest number of vehicles, comes down to solving a weighted perfect matching problem in a bipartite graph (the graph induced by all turnaround arcs). Moreover, as we assume that deadheading is not allowed, the bipartite graph decomposes into a complete bipartite graph per station, such that a global optimal solution can be found by solving a weighted perfect matching problem for each terminal station independently. Borndörfer et al. (2018a) prove that solving this formulation is actually equivalent with solving a traditional non-periodic vehicle scheduling problem (see e.g. Bunte and Kliewer (2009)) on a roll out of the periodic timetable over the day.

3.3.1 Properties

We now present properties of the VCSP. Later, we use these properties to develop a strong formulation for the VC-PESP.

Our analysis is based on the observation that the arc weights x_a of the weighted perfect matching problem are generated from a special metric. Specifically, there exist end times $\pi_e \in [0,T)$ for all $e \in E_{\text{end}}$ and start times $\pi_f \in [0,T)$ for all $f \in E_{\text{start}}$ such that $x_a = \pi_f - \pi_e \bmod T$ for all $a = (e,f)$[1]. Hence, the end events and start events can be represented by points on the circle with circumference T, such that we can view the VCSP as *clockwise bipartite perfect matching on a circle*, which needs to be performed per station. To illustrate, Figure 3.3 depicts an example of such a matching problem and a possible solution. The following proposition provides a first result of this structure:

Proposition 3.3. *Let y^1 and y^2 denote feasible solutions to* (3.11)-(3.14)*. For a feasible timetable* $\mathbf{x}$*, it holds that* $\sum_{a\in A_{\text{turn}}} x_a y_a^1 - \sum_{a\in A_{\text{turn}}} x_a y_a^2 = zT$ *for some integer* z*.*

Proof. Intuitively, this statement must be true since the number of vehicles required to operate a timetable must always be integer. For a formal proof, let $M^1 = \{a \in A_{\text{turn}} : y_a^1 = 1\}$ and $M^2 = \{a \in A_{\text{turn}} : y_a^2 = 1\}$. The symmetric difference of two perfect matchings $M^1 \oplus M^2 = \left(M^1 \setminus M^2\right) \cup \left(M^2 \setminus M^1\right)$ is the union U of vertex-disjoint cycles. Within these cycles, arcs from M^1 and M^2 alternate. It follows that

$$\begin{aligned}
\sum_{a\in A_{\text{turn}}} x_a y_a^1 - \sum_{a\in A_{\text{turn}}} x_a y_a^2 &= \sum_{C\in U}\left(\sum_{a\in C: a\in M^1} x_a - \sum_{a\in C:a\in M^2} x_a\right) \\
&= \sum_{C\in U} q_C T \qquad \text{(Lemma 3.2)} \\
&= \left(\sum_{C\in U} q_C\right) T
\end{aligned}$$

for integers q_C. □

To further analyze the problem, we define the *inventory function* of a matching

[1]We disregard the case where $x_a = T$, as this is never optimal for minimizing the number of vehicles.

$I_s(t, M)$, which denotes the number of idle vehicles at station s under matching M at time $t \in [0, T)$. Figure 3.3 depicts an example of the inventory function. Using this concept, the following lemma characterizes optimal solutions to the VCSP.

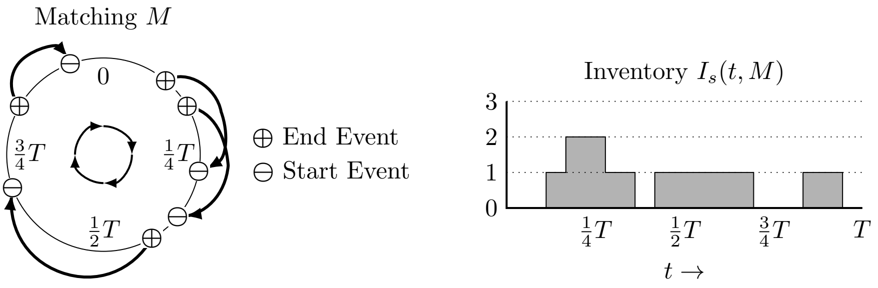


Figure 3.3: Example of a matching at a station and the associated inventory function.

Lemma 3.4. *Consider station s with turnaround arcs $A_{\text{turn}}(s)$ and tensions x_a for all $a \in A_{\text{turn}}(s)$. Let $M \subseteq A_{\text{turn}}(s)$ and $M' \subseteq A_{\text{turn}}(s)$ denote perfect matchings from the associated end events to start events. The following statements are true:*
(i) $\sum_{a \in M} x_a = \int_0^T I_s(t, M)dt$.
(ii) $I_s(t, M) - I_s(t, M') = z$ for all $t \in [0, T)$ for some constant integer z.
(iii) M is a minimum cost perfect matching if and only if $I_s(t, M) = 0$ for some $t \in [0, T)$.

Proof. (i) Consider an arc $a = (e, f)$. Let $I_s(t, a)$ be equal to 1 if a "covers" time instant t and 0 otherwise. Formally, if $\pi_e \leq \pi_f$, a covers t if $t \in [\pi_e, \pi_f]$, otherwise a covers t if $t \in [0, \pi_f] \cup [\pi_e, T]$. It holds that $I_s(t, M) = \sum_{a \in M} I_s(t, a)$. Hence, $\int_0^T I_s(t, M)dt = \int_0^T \sum_{a \in M} I_s(t, a)dt = \sum_{a \in M} \int_0^T I_s(t, a)dt = \sum_{a \in M} x_a$.
(ii) Let $\text{end}(a, b)$ and $\text{start}(a, b)$ denote the number of end and start events in the interval $[a, b)$, respectively. It holds that $I_s(t, M) = I_s(0, M) + \text{end}(0, t) - \text{start}(0, t)$. Therefore, $I_s(t, M) - I_s(t, M') = I_s(0, M) - I_s(0, M')$, which is integer.
(iii) First assume that $I_s(t', M) = 0$ for some $t' \in [0, T)$. Let M^{opt} denote the optimal matching. Clearly, $I_s(t', M^{\text{opt}}) \geq 0$, such that $I_s(t', M) - I_s(t', M^{\text{opt}}) \leq 0$. From (ii), it follows that $I_s(t, M) - I_s(t, M^{\text{opt}}) \leq 0$ for all t. From (i), we then find that

$$\sum_{a \in M} x_a - \sum_{a \in M^{\text{opt}}} x_a = \int_0^T I_s(t, M) - I_s(t, M^{\text{opt}})dt \leq 0,$$

hence M is optimal. To prove the reverse direction, assume that M is optimal but $I_s(t, M) > 0$ for all $t \in [0, T)$. Then, there exist arcs $(e_1, f_1), ...(e_k, f_k) \in M$ that cover all $t \in [0, T)$ such that

$$0 \leq \pi_{e_1} < \pi_{f_k} \leq \pi_{e_2} < \pi_{f_1} \leq ... \leq \pi_{e_k} < \pi_{f_{k-1}} < T.$$

Note that the alternating structure is due to the non-empty inventory at any t. The strict inequalities come from the observation that if $\pi_e = \pi_f$ for some end event e and start event f, this pair does not affect the inventory function and can therefore be left out of consideration. Clearly, the cost of the matching can be decreased by selecting the arcs $(e_1, f_k), (e_2, f_1), ..., (e_k, f_{k-1})$, contradicting the optimality of M. To illustrate, an example of a matching for which $I_s(t, M) > 0$ for all $t \in [0, T)$ is presented in Figure 3.4. $\square$

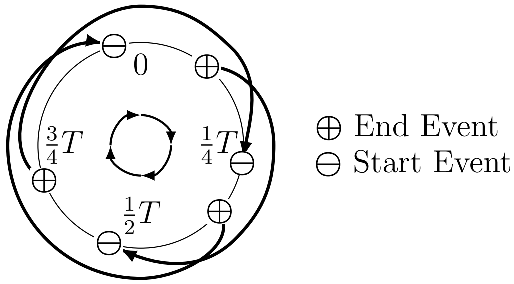


Figure 3.4: Example of a matching for which the inventory is non-empty at any t.

Next, we present a greedy algorithm for vehicle circulation scheduling in Algorithm 1. The algorithm simply iteratively matches unmatched end events with the unmatched start event that gives the shortest turnaround time.

Algorithm 1 Greedy algorithm for vehicle circulation scheduling

Input: Turnaround graph $G_{\text{turn}}(s) = (E_{\text{end}}(s), E_{\text{start}}(s), A_{\text{turn}}(s))$ and turnaround times $x_{e,f}$ $\forall (e, f) \in A_{\text{turn}}(s)$
Output: Minimum cost perfect matching M in $G_{\text{turn}}(s)$

```
M = ∅, U_end = E_end(s), U_start = E_start(s)
while M is not a perfect matching do
    Pick any event e ∈ U_end
    Find f = argmin_{g∈U_start} x_{e,g}
    M ← M ∪ (e, f)
    U_end ← U_end \ e, U_start ← U_start \ f
Return M
```

Theorem 3.5. *Algorithm 1 is optimal for the VCSP.*

Proof. Assume M^{greedy} is not optimal. Then, by Theorem 3.4, $I_s(t, M^{\text{greedy}}) > 0$ for all $t \in [0, T)$. As in the proof of part (iii) of Lemma 3.4, this implies that there exist arcs $(e_1, f_1), ...(e_k, f_k) \in M$ that cover all $t \in [0, T)$, such that

$$0 \leq \pi_{e_1} < \pi_{f_k} \leq \pi_{e_2} < \pi_{f_1} \leq ... \leq \pi_{e_k} < \pi_{f_{k-1}} < T.$$

We can further assume that e_1 is the end event that is matched first in the greedy algorithm (of the events $e_1, ..., e_k$), as otherwise the events can be relabeled and the times can all be shifted by a constant. The event e_1 is matched with f_1. However, it holds that $x_{e_1,f_k} < x_{e_1,f_1}$. Therefore, the greedy algorithm would select the arc (e_1, f_k) instead of (e_1, f_1), resulting in a contradiction. It follows that M^{greedy} is optimal. □

As a corollary, we find that the vehicle circulation scheduling problem can be solved in time $O(m^2 \log m)$, where $m = |E_{\text{end}}| = |E_{\text{start}}|$.

Corollary 3.6. *The VCSP can be solved in $O(m^2 \log m)$, where $m = |E_{end}|$.*

Proof. Algorithm 1 is correct for any order in which the end events are matched. Hence, a valid approach is to first sort all turnaround activities in the entire network according to their tension, and then choose the first m (starting with the shortest tension) non-conflicting activities in this list as the selected turnarounds. Since there are at most m^2 activities in the turnaround graph, this results in a complexity of $O(m^2 \log m^2) = O(m^2 \log m)$. □

We conclude this section by proving two additional properties of the VCSP. In Proposition 3.7 we show that in case the events at a station repeat with a shorter period than the global period (e.g. every 30 minutes instead of 60 minutes), the inventory function at this station will also repeat with the shorter period, for any matching. In Theorem 3.8, we provide a lower bound on the minimum turnaround time at a station based on a weighted sum of the durations of all turnaround activities.

Proposition 3.7. *Consider station s with turnaround arcs $A_{\text{turn}}(s)$ and suppose the events $E_{\text{end}}(s)$ and $E_{\text{start}}(s)$ repeat periodically every T_s time units, with $T_s < T$. Then, for any perfect matching $M \subseteq A_{\text{turn}}(s)$, it holds that $I_s(t, M) = I_s(t + T_s \bmod T, M)$.*

Proof. Since the end and start events repeat every T_s time units, it holds that $\text{end}(0, T_s) = \text{start}(0, T_s)$. Moreover, it holds that $\text{start}(0, t) = \text{start}(T_s, T_s + t)$ and $\text{end}(0, t) = \text{end}(T_s, T_s + t)$. First, consider the case where $t + T_s < T$. Then,

$$\begin{aligned}
I_s(t + T_s \bmod T, M) &= I_s(t + T_s, M) \\
&= I_s(0, M) + \text{end}(0, t + T_s) - \text{start}(0, t + T_s) \\
&= I_s(0, M) + \text{end}(T_s, T_s + t) - \text{start}(T_s, T_s + t) \\
&= I_s(0, M) + \text{end}(0, t) - \text{start}(0, t) \\
&= I_s(t, M).
\end{aligned}$$

As T must be a multiple of T_s, it follows that

$$I_s(t, M) = I_s(t + iT_s, M) \text{ for } t \in [0, T_s) \text{ and } i \in \{1, ..., T/T_s - 1\} \tag{3.15}$$

Next, consider the case where $t+T_s \geq T$. It holds that $t+T_s \bmod T = t+T_s-iT$ for some $i \in \mathbb{Z}$. Because T must be a multiple of T_s. it follows that $t+T_s \bmod T = t-jT_s$ for some $j \in \mathbb{Z}$. Equation (3.15) implies that $I_s(t+T_s \bmod T, M) = I_s(t-jT_s, M) = I_s(t, M)$. □

Theorem 3.8. *Consider station s with end events $E_{\text{end}}(s)$, turnaround arcs $A_{\text{turn}}(s)$ and tensions x_a for all $a \in A_{\text{turn}}(s)$ and let $M(s) \subseteq A_{\text{turn}}$ denote any feasible perfect matching. It holds that*

$$\sum_{a \in M(s)} x_a \geq \frac{1}{|E_{\text{end}}(s)|} \sum_{a \in A_{\text{turn}}(s)} x_a - T\frac{|E_{\text{end}}(s)| - 1}{2}. \tag{3.16}$$

Proof. We first consider the case where end events and start events alternate in the periodic pattern. Let $k = |E_{\text{end}}(s)|$. Then, we can partition the arcs $A_{\text{turn}}(s) = A_1(s) \cup A_2(s) \cup ... \cup A_k(s)$, where $A_1(s)$ contains all activities from end events to the next start, $A_2(s)$ all the activities from end events to start events that 'skip' one start event and in general $A_i(s)$ contains the activities from end events to start events that 'skip' i departures (see Figure 3.5 for an illustration). It can be observed that $\sum_{a \in A_i(s)} x_a = \sum_{a \in A_1(s)} x_a + (i-1)T$. Therefore, we have that

$$\sum_{a \in A_{\text{turn}}(s)} x_a = \sum_{i=1}^{k} \sum_{a \in A_i(s)} x_a = k \sum_{a \in A_1(s)} x_a + T\frac{k(k-1)}{2},$$

or equivalently,

$$\sum_{a \in A_1(s)} x_a = \frac{1}{k} \sum_{a \in A_{\text{turn}}(s)} x_a - T\frac{k-1}{2}.$$

The theorem follows from the observation that $A_1(s)$ minimizes the total turnaround time. For the case where end and start events do not alternate, we first relate the

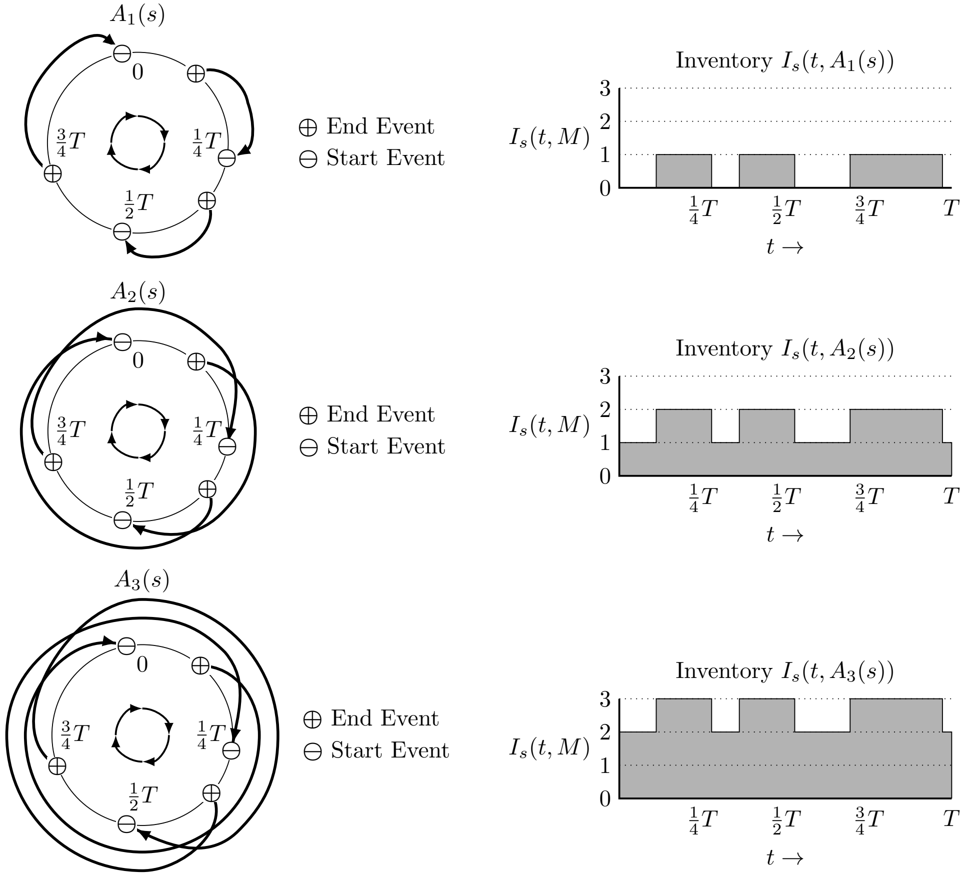


Figure 3.5: Illustration of the decomposition of all turnaround arcs into $A_1(s)$, $A_2(s)$ and $A_3(s)$, along with the corresponding inventory functions.

theorem to the difference between the minimum turnaround time and the average turnaround time. Let $\mathcal{M}(s)$ denote the set of all perfect matchings at station s and let $x(M) = \sum_{a \in M} x_a$ denote the turnaround time of matching M. Since the turnaround graph is a complete bipartite graph, there are $|E_{\text{end}}(s)|!$ distinct perfect matchings and every arc is contained in $|E_{\text{end}}(s) - 1|!$ matchings, so that

$$\begin{aligned}\frac{1}{|\mathcal{M}(s)|}\sum_{M\in\mathcal{M}(s)}x(M) &= \frac{1}{|E_{\text{end}}(s)|!}\sum_{a\in A_{\text{turn}}(s)}\sum_{M\in\mathcal{M}(s):a\in M}x_a\\ &= \frac{1}{|E_{\text{end}}(s)|!}\sum_{a\in A_{\text{turn}}(s)}|E_{\text{end}}(s)-1|!\,x_a\\ &= \frac{1}{|E_{\text{end}}(s)|}\sum_{a\in A_{\text{turn}}(s)}x_a.\end{aligned}$$

In other words, the theorem gives a lower bound on the turnaround of any matching based on the average turnaround time over all possible matchings. Hence, what remains to prove is that the difference between the minimum turnaround time and the average turnaround time is maximized when end events and start events alternate.

Let M^{opt} denote the matching attaining the minimum turnaround time and let M' denote any other matching. Let $\pi_e \in [0,T)$ denote the event time of event e. As ends and starts do not alternate, there are end events e_1 and e_2 and start events f_1 and f_2 such that $(e_1,f_1)\in M^{\text{opt}}$, $(e_2,f_2)\in M^{\text{opt}}$ and

$$0=\pi_{e_1}\le\pi_{e_2}<\pi_{f_1}\le\pi_{f_2}.$$

Now, consider shifting f_1 with δ such that $\pi'_{f_1}=\pi_{f_1}-\delta$ and $\pi_{e_1}\le\pi'_{f_1}\le\pi_{e_2}$, i.e. after the shift the end events e_1, e_2 and start events f_1, f_2 do alternate. Let $x'(M)$ denote the turnaround time of matching M after shifting f_1. Clearly, $x'(M^{\text{opt}}) = x(M^{\text{opt}})-\delta$. Let e_{alt} denote the end event that turns on f_1 in matching M'. Then,

$$x'(M')=\begin{cases}x(M')+T-\delta & \text{if } \pi_{f_1}-\delta<\pi_{e_{\text{alt}}}<\pi_{f_1}\\ x(M')-\delta & \text{else.}\end{cases}$$

It follows that $x'(M')-x'(M^{\text{opt}})\ge x(M')-x(M^{\text{opt}})$. In other words, by disentangling non-alternating end events and start events, the difference between the minimum turnaround time and the turnaround time of an arbitrary matching cannot decrease. Therefore, the difference is maximized when all ends and starts alternate.

Putting the ingredients together, we find that

$$\begin{aligned}
\frac{1}{|E_{\text{end}}(s)|}\sum_{a\in A_{\text{turn}}(s)} x_a - \sum_{a\in M(s)} x_a &\le \frac{1}{|E_{\text{end}}(s)|}\sum_{a\in A_{\text{turn}}(s)} x_a - \sum_{a\in M^{\text{opt}}(s)} x_a \\
&= \frac{1}{|\mathcal{M}(s)|}\sum_{M\in\mathcal{M}(s)} x(M) - \sum_{a\in M^{\text{opt}}(s)} x_a \\
&\le \frac{1}{|\mathcal{M}(s)|}\sum_{M\in\mathcal{M}(s)} x'(M) - \sum_{a\in M^{\text{opt}}(s)} x'_a \\
&\le T\frac{|E_{\text{end}}(s)|-1}{2}.
\end{aligned}$$

□

Note that for stations with only a single terminating line, Equation (3.16) directly gives the minimum turnaround time, since the end events and start events of one line always alternate. Hence, for such stations, the minimum turnaround time can be computed without explicitly solving a matching problem.

3.4 Mathematical Formulation for the VC-PESP

We are now ready to present a mathematical formulation for the integrated timetabling and vehicle circulation scheduling problem. To cover the timetabling part, we introduce the same decision variables as in the cycle periodicity formulation of the PESP: continuous tension variables x_a for all $a \in A$ and integer valued periodicity variables q_C for all $C \in \mathcal{B}$, where $\mathcal{B}$ is an integral cycle basis. For the vehicle circulation scheduling part, we introduce binary matching variables y_a for all $a \in A_{\text{turn}}$ which are equal to 1 if turnaround activity a is selected in the matching and 0 otherwise. Furthermore, we introduce an integer variable n representing the total number of vehicles required to operate all circulations. The VC-PESP can then be formulated as the following bi-objective problem:

$$
\begin{aligned}
\min \quad & \sum_{a \in A} \lambda_a x_a && && (3.17)\\
\min \quad & n && && (3.18)\\
\text{s.t.} \quad & l_a \leq x_a \leq u_a && \forall a \in A, && (3.19)\\
& \sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = q_C T && \forall C \in \mathcal{B}, && (3.20)\\
& a_C \leq q_C \leq b_C && \forall C \in \mathcal{B}, && (3.21)\\
& \sum_{a \in \delta^+(e)} y_a = 1 && \forall e \in E_{\text{end}}, && (3.22)\\
& \sum_{a \in \delta^-(e)} y_a = 1 && \forall e \in E_{\text{start}}, && (3.23)\\
& n \geq \frac{1}{T} \left(\sum_{a \in A_{\text{veh}}} x_a + \sum_{a \in A_{\text{turn}}} x_a y_a \right) && && (3.24)\\
& x_a \in \mathbb{R} && \forall a \in A, && (3.25)\\
& q_C \in \mathbb{Z} && \forall C \in \mathcal{B}, && (3.26)\\
& y_a \in \{0, 1\} && \forall a \in A_{\text{turn}}, && (3.27)\\
& n \in \mathbb{Z}. && && (3.28)
\end{aligned}
$$

The objectives are to minimize the average (perceived) travel times of all passengers (which can be achieved by setting appropriate weights λ_a) and the operator costs (captured by the number of required vehicles to operate the timetable). Constraints (3.19)-(3.21) cover the timetabling part of the formulation. Constraints (3.22) and (3.23) handle the matching part of the formulation, making sure that every trip has exactly one successor and one predecessor. Constraint (3.24) provides the link between the tension variables, the matching variables and the vehicle variable. The right hand side represents the total driving, dwelling and turnaround time of the chosen timetable in combination with the selected turnaround, divided by the period length. In a feasible integer solution, the sum of tensions of each cycle will necessarily be an integer multiple of the period, such that the division by the period gives an integer result. The remainder of the constraints state the domains of the variables.

Formulation (3.17)-(3.28) is nonlinear, as constraint (3.24) contains the product of binary and continuous variables. However, it can be linearized by defining auxiliary continuous variables w_e for all $e \in E_{\text{end}}$, representing the turnaround time between end event e and the start event it connects to, and replacing constraint (3.24) by

$$n \geq \frac{1}{T} \left(\sum_{a \in A_{\text{veh}}} x_a + \sum_{e \in E_{\text{end}}} w_e \right). \tag{3.29}$$

To ensure that the w-variables attain the correct values, we add the following constraints:

$$w_e \geq x_a - M_1(1 - y_a) \quad \forall e \in E_{\text{end}} \quad \forall a \in \delta^+(e), \tag{3.30}$$

$$w_e \leq x_a + M_1(1 - y_a) \quad \forall e \in E_{\text{end}} \quad \forall a \in \delta^+(e), \tag{3.31}$$

$$w_e \geq 0 \quad \forall e \in E_{\text{end}}. \tag{3.32}$$

It suffices to set $M_1 = T$, since x_a is smaller than T for turnaround arcs. The resulting formulation is the one studied by Nührenberg (2016), except that we include both the average travel time and the number of vehicles as objectives, instead of only the number of vehicles.

3.5 Complexity

As the PESP is NP-complete, it follows that the VC-PESP is NP-complete as well. However, it turns out that even if all timetabling complexity is removed from the problem (such that any timetable is feasible), it is still a hard problem to decide whether there exists a timetable that can be operated with a certain number of vehicles.

Specifically, let us define the SIMPLE-VC-PESP as follows. Consider a line plan $\mathcal{L}$, where each line $l = (s_1, s_2) \in \mathcal{L}$ has a minimum unidirectional trip time t_l and a frequency f_l. From this line plan, we derive an event-activity network with events for the departures and arrivals of the lines at the terminals, trip activities linking the departure of a line at one terminal to the arrival of the line at the other terminal, synchronizing activities linking different events of the same line and turnaround activities linking arrivals with departures at the same terminal. The synchronizing activities are added for lines with frequencies larger than 1 and impose that the departures of a line with frequency f are separated by exactly T/f time units. In this problem, there are no safety or transfer activities and all drive and dwell activities are contracted in a single trip activity with a fixed duration t_l. This implies that the arrival times of lines directly follow from the departure times. Therefore, the SIMPLE-VC-PESP contains only two timetabling decision variables $\pi_l^{s_1}$ and $\pi_l^{s_2}$ per line $l = (s_1, s_2)$, representing the departure times at the terminals.

Definition 3.9 (SIMPLE-VC-PESP)**.** Given a period T, a line plan $\mathcal{L}$ specifying the lines and frequencies and given a maximum number of vehicles n, the Simple Vehicle Circulation Periodic Event Scheduling Problem is to determine departure times at the terminals $\pi_l^{s_1}$ and $\pi_l^{s_2}$ for all lines $l = (s_1, s_2) \in \mathcal{L}$ such that at most n vehicles are needed to operate the periodic timetable, or to conclude that no such timetable exists.

Note that in the SIMPLE-VC-PESP, all timetables $\boldsymbol{\pi}$ are feasible. Hence, the only remaining complexity lies in making sure that all arrivals have short turnarounds to departures, such that only a small number of vehicles is required.

Theorem 3.10. *The SIMPLE-VC-PESP is NP-complete.*

Proof. The SIMPLE-VC-PESP clearly is in NP. We prove NP-completeness by reduction from the strongly NP-complete 3-PARTITION problem (Garey and Johnson, 1979). Let $S = \{s_1, s_2, \ldots, s_m\}$ be a set of integers and let B be an integer such that $\sum_{i=1}^{m} s_i = \frac{m}{3}B$ and for all i it holds that $\frac{B}{4} < s_i < \frac{B}{2}$. A 3-partition instance is a YES-instance if it is possible to partition S into $k = \frac{m}{3}$ triplets $S_1, S_2, \ldots S_k$ such that each triplet sums to B.

We reduce the 3-PARTITION instance to a star network where we have a central hub station $v_0 \in V$ and $m+1$ outer stations $v_1, \ldots, v_{m+1} \in V$. We put $T = k + \sum_{i=1}^{m} s_i$ time units. For every element $s_i \in S$, we create a line $l_i \in \mathcal{L} = \{v_0, v_i\}$ with frequency $f_i = 1$ and a trip time $t_i = \frac{1}{2}s_i$. Furthermore, we create a line $l_{m+1} = \{v_0, v_{m+1}\}$ with frequency $f_{m+1} = k$ and a trip time $t_{m+1} = \frac{1}{2k}$.

Note that the total driving time of the created SIMPLE-VC-PESP instance equals $\sum_{i=1}^{m+1} 2f_i t_i = k + \sum_{i=1}^{m} s_i = T$. Therefore, at least 1 vehicle is required to operate the line plan. We put $n = 1$. Furthermore, w.l.o.g. we can assume that the vehicles have a turnaround time of 0 time units at the outer stations (i.e. $\pi_{l_i}^{v_i} = \pi_{l_i}^{v_0} + t_i \bmod T$). Therefore, only the departure times at the central station need to be determined. We claim that the 3-PARTITION instance is a YES-instance if and only if the SIMPLE-VC-PESP instance is a YES-instance.

If the 3-PARTITION instance is a YES-instance, we can create a solution using only a single vehicle. Let $S_1, ..., S_k$ denote the triplets. We set $\pi_{l_{m+1}} = 0$, so line $m+1$

has departures at $0, \frac{T}{k}, ..., \frac{(k-1)T}{k}$. For triplet $S_j = (s_p, s_q, s_r)$, we set $\pi_{l_p} = \frac{(j-1)T}{k} + \frac{1}{k}, \pi_{l_q} = \pi_{l_p} + 2t_p$ and $\pi_{l_r} = \pi_{l_q} + 2t_q$. Then, it is possible to create a circulation that performs the first service of line $m+1$, then the lines corresponding to S_1, then the second service of line $m+1$, then all lines corresponding to S_2 et cetera. As this circulation takes exactly one period, only a single vehicle is needed to operate the timetable, such that the SIMPLE-VC-PESP instance is a YES-instance.

If the SIMPLE-VC-PESP instance is a YES-instance, it must be that there is a single circulation covering all lines with a duration of T. W.l.o.g. we can set $\pi_{l_{m+1}} = 0$. We can therefore write this circulation as

$$l^1_{m+1}, \underbrace{l'_1, ..., l'_p}_{\mathcal{L}_1}, l^2_{m+1}, \underbrace{l''_1, ..., l''_q}_{\mathcal{L}_2} l^3_{m+1} ...... l^k_{m+1} \underbrace{l'''_1, ..., l'''_r}_{\mathcal{L}_k}$$

As the services of line $m+1$ should be spread exactly $\frac{1}{k}T$ time units and the total circulation time equals T, it holds that

$$\sum_{l \in \mathcal{L}_1} t_l = \sum_{l \in \mathcal{L}_2} t_l = ... = \sum_{l \in \mathcal{L}_k} t_l = B,$$

such that $\bigcup_{i=1}^k \mathcal{L}_i = \mathcal{L} \setminus \{l_{m+1}\}$. The corresponding elements in S partition S into k subsets with a sum of B. Moreover, since $\frac{B}{4} < s_i < \frac{B}{2}$, all these subsets must contain exactly three elements. We conclude that that the 3-PARTITION instance also is a YES-instance. □

3.6 A Stronger Formulation

The mathematical formulation of the VC-PESP given in Section 3.4 has been investigated by Nührenberg (2016). However, the author finds that the formulation is weak and results in large optimality gaps even for medium-sized instances. Besides the large optimality gaps that tend to come with the PESP model itself (see e.g. Borndörfer et al. (2019)), this is likely caused by the big-M constraints (3.30)-(3.31) that are necessary to linearize the formulation. Furthermore, the formulation also allows for many symmetric solutions, as there can be multiple matchings that attain the same turnaround time.

In this section, we present several ways to enhance the formulation of the VC-PESP by exploiting the special matching structure of the vehicle circulation scheduling problem. First, we discuss how the number of binary matching variables can be reduced. Next, we give multiple valid inequalities that strengthen the linear relaxation. Finally, we propose symmetry-breaking constraints, that make sure only a single matching that attains the minimum turnaround time is feasible.

3.6.1 Computing the Matching in a Contracted Network

Even though the period of the entire timetable might be equal to T, often the period is smaller from a local perspective. We can use this property to reduce the number of matching variables necessary to compute the total turnaround time at a station. In particular, if the events at station s repeat every T_s time units, we say that the *local period* at station s is T_s. In case the local period is strictly smaller than the global period, it turns out to be possible to reduce the number of matching variables at s with a factor of $\left(\frac{T}{T_s}\right)^2$.

For instance, consider a station that serves as a terminal for two lines, line A that runs every 15 minutes and line B that runs every 30 minutes. Regardless of the global period, the local period T_s at this station is 30 minutes, since all events repeat at least every 30 minutes. By Proposition 3.7, this implies that the inventory function also repeats every 30 minutes, for any matching. Hence, we can simply optimize the matching for all events occurring in 30 minutes and repeat this pattern in the subsequent periods. Figure 3.6 visualizes the contraction for this station for the case that the global period is 60 minutes. The events e_1^A and e_3^A are end events of line A that are separated with exactly 30 minutes, so they are combined in a single node in the contracted graph. The same holds for other pairs of events in the original graph. This results in a reduction of the number of turnaround arcs from 36 to 9.

To formally model the total turnaround time in the contracted graphs, we define for each line l that terminates at s and has global frequency f_l, the local line frequency $f_l^s := \frac{f_l T_s}{T}$. Next, we create a contracted graph in which every line is represented with f_l^s contracted end events and f_l^s contracted start events. Each contracted event corresponds to $\frac{T}{T_s}$ events in the regular network that are all separated by a multiple of T_s minutes. Let $E_{\text{end}}^{\text{con}}(s)$ and $E_{\text{start}}^{\text{con}}(s)$ denote all end events and start events in the contracted graph at s. The set $A_{\text{turn}}^{\text{con}}(s)$ denotes the set of all turnaround arcs in the contracted networks, and contains an arc for each pair of contracted end and start events, such that the contracted network is complete. It follows that each contracted arc a corresponds to $\left(\frac{T}{T_s}\right)^2$ arcs in the regular network. We let $A_{\text{turn}}(a)$ denote the set of arcs in the regular network that correspond to the contracted arc a. Finally, we let S denote the set of all terminal stations.

To perform the matching in the contracted network, we introduce a binary matching variable γ_a and a continuous tension variable χ_a for all $a \in A^{\text{con}}_{\text{turn}}(s)$ and a turnaround time variable ω_e for all $e \in E^{\text{con}}_{\text{end}}(s)$. The following constraints then model the matching in this contracted networks:

$$\sum_{a \in \delta^+(e)} \gamma_a = 1 \qquad \forall e \in E^{\text{con}}_{\text{end}}, \tag{3.33}$$

$$\sum_{a \in \delta^-(e)} \gamma_a = 1 \qquad \forall e \in E^{\text{con}}_{\text{start}}, \tag{3.34}$$

$$\chi_a \geq x_{a'} - T + T_s \qquad \forall s \in S, \quad \forall a \in A^{\text{con}}_{\text{turn}}(s), \quad \forall a' \in A_{\text{turn}}(a), \tag{3.35}$$

$$\chi_a \leq x_{a'} \qquad \forall s \in S, \quad \forall a \in A^{\text{con}}_{\text{turn}}(s), \quad \forall a' \in A_{\text{turn}}(a), \tag{3.36}$$

$$\omega_e \geq \chi_a - M_2(1-\gamma_a) \qquad \forall e \in E^{\text{con}}_{\text{end}}, \quad \forall a \in \delta^+(e), \tag{3.37}$$

$$\omega_e \leq \chi_a + M_2(1-\gamma_a) \qquad \forall e \in E^{\text{con}}_{\text{end}}, \quad \forall a \in \delta^+(e), \tag{3.38}$$

$$n \geq \frac{1}{T}\left(\sum_{a \in A_{\text{veh}}} x_a + \sum_{s \in S} \frac{T}{T_s} \sum_{e \in E^{\text{con}}_{\text{end}}(s)} \omega_e\right) \tag{3.39}$$

$$\omega_e \geq 0 \qquad \forall e \in E^{\text{con}}_{\text{end}}, \tag{3.40}$$

$$\chi_a \in \mathbb{R} \qquad \forall a \in A^{\text{con}}_{\text{turn}}, \tag{3.41}$$

$$\gamma_a \in \{0,1\} \qquad \forall a \in A^{\text{con}}_{\text{turn}}. \tag{3.42}$$

Constraints (3.33)-(3.34) are the matching constraints for the contracted network. Constraints (3.35)-(3.36) link the tensions of the activities of the contracted network with those of the original network. Here, we use that the tensions of the arcs in the original network associated to an arc in the contracted network are $x, x+T_s, x+2T_s, ..., x+T-T_s$ for some $x \in [0, T_s)$. For example, if the local period is 15 minutes and the contracted arc a corresponds to tensions of 12, 27, 42 and 57 minutes in the original network, these constraints enforce that $\chi_a = \min\{12, 27, 42, 57\} = 12$.

Constraints (3.37)-(3.38) make sure that the turnaround time of every contracted event is equal to the tension of the arc that is selected to be in the matching. Since it holds that $\chi_a \in [0, T_s)$, it now suffices to set $M_2 = T_s$ instead of T, which also strengthens the formulation. Constraint (3.39) counts the number of vehicles, and

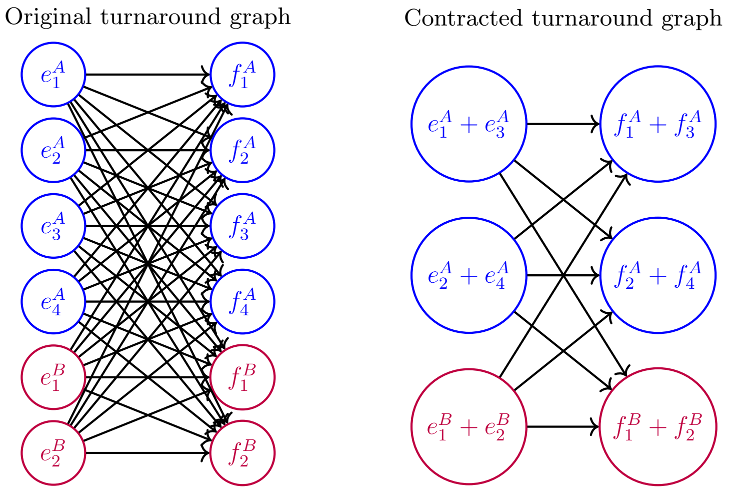


Figure 3.6: Contraction of the turnaround graph at a station with lines A and B with frequencies 4 and 2, respectively.

now takes into account that turnaround times of contracted events should be counted according to their multiplicity.

For the sake of notation, in the remainder of this section, we omit the superscript "con" to denote the contracted turnaround graph and write x_a instead of χ_a, y_a instead of γ_a and w_e instead of ω_e.

3.6.2 Strengthening the LP relaxation

A potential weakness of the original formulation is that its linear programming (LP) relaxation is weak due to the linearization of the quadratic terms $x_a y_a$. When inspecting solutions to the LP relaxation, we noticed that even when the periodicity variables q_C are integer for all $C \in \mathcal{B}$ (so the timetable is feasible) and only the integrality of the matching variables y_a is relaxed, the turnaround time variables w_e often still equal 0 for all end events. To illustrate this, consider the timetable displayed in Figure 3.7, with two vehicles turning at a certain station. We assume the event times are fixed and that the end events take place at time instants $-\varepsilon$ and ε, and the start events at $\frac{1}{2}T_s - \varepsilon$ and $\frac{1}{2}T_s + \varepsilon$, for some small ε. Clearly, the minimum turnaround time, which is actually obtained by both possible matchings, is T_s. However, if the matching variables y_a are equal to $\frac{1}{2}$, the turnaround time variables w_e can take the value ε, such that the total turnaround time is smaller than 2ε. In order to decrease the gap between the actual minimum turnaround time and the turnaround time in the relaxation, we develop three classes of valid inequalities.

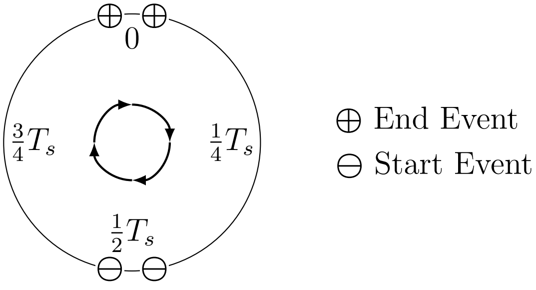


Figure 3.7: Example illustrating the weak linear programming relaxation of the formulation.

Min-Mean Cuts

The first class of valid inequalities is based on the lower bound on the minimum turnaround time at a station given by Theorem 3.8. This theorem directly implies that the following inequalities are valid for the VC-PESP:

$$\sum_{e \in E_{\text{end}}(s)} w_e \geq \frac{1}{|E_{\text{end}}(s)|} \sum_{a \in A_{\text{turn}}(s)} x_a - T_s \frac{|E_{\text{end}}(s)| - 1}{2} \quad \forall s \in S. \tag{3.43}$$

We refer to these inequalities as the *min-mean* cuts, as Theorem 3.8 relates the minimum turnaround time with the mean turnaround time over all matchings. Recall that for stations with only a single terminating line, these cuts directly give the minimum turnaround time, since the end events and start events of one line always alternate, so for these stations it is no longer necessary to add any binary variables.

For the example in Figure 3.7, the min-mean cut states that $\sum_{e \in E_{\text{end}}(s)} w_e \geq \frac{1}{2}T_s$. Therefore, the original LP solution where both turnaround time variables have value ε is cut off.

Reference Matching Cuts

To improve the effectiveness of the min-mean cuts, we also include the following constraints. Let M_s denote a reference matching for station s and introduce the integer variable z_s for all $s \in S$. By Proposition 3.3, the difference in turnaround time between two matchings must be an integer multiple of T_s, hence the following equality is valid:

$$\sum_{e \in E_{\text{end}}(s)} w_e = \sum_{a \in M_s} x_a - T_s z_s \quad \forall s \in S, \tag{3.44}$$

$$z_s \in \mathbb{Z} \quad \forall s \in S. \tag{3.45}$$

To illustrate the usefulness of these cuts, consider again the example in Figure 3.7. In case the reference matching variable z_s is integer, this implies that $\sum_{e \in E_{\text{end}}(s)} w_e$ is either 0 or T_s. Next, the min-mean cut states that $\sum_{e \in E_{\text{end}}(s)} w_e \geq \frac{1}{2} T_s$. Hence, provided that z_s is integer, $\sum_{e \in E_{\text{end}}(s)} w_e$ must be equal to T_s. The z_s variables are therefore useful variables to branch on in a branch-and-bound context.

Headway Cuts

Headway cuts can be used if all departures at a station should be separated by a headway time $h_s > 0$. In this case, if a start event f is scheduled at time π_f and an end event e does *not* connect to f, it follows that e connects to a departure outside the interval $[\pi_f - h_s, \pi_f + h_s]_{T_s}$. This implies that the following inequalities are valid:

$$w_e \geq x_a + h_s - T_s \quad \forall s \in S, \quad \forall e \in E_{\text{end}}(s), \quad \forall a \in \delta^+(e), \tag{3.46}$$

In case the end event e connects to f, the inequality clearly holds as $h_s < T_s$. Otherwise, e connects to a different start event, which must be separated at least h_s time units from f. Taking periodicity into account, the right hand side of the inequality represents a lower bound on the turnaround time. For example, if $T_s = 60$ minutes, $h_s = 3$ minutes, $x_{e,f} = 59$ and e does not connect to f, it should hold that $w_e \geq 2$.

These inequalities cut off a relatively small portion of the feasible region, since they only have an effect if $x_{e,f} + h_s > T_s$ and h_s usually is relatively small. By preprocessing the event-activity network, the minimum separation between departures can usually be strengthened. Furthermore, these constraints can easily be incorporated by setting $M_2 = T_s - h_s$ in constraints (3.37)-(3.38).

3.6.3 Symmetry Breaking

In the VC-PESP, there can be a very large number of matchings that lead to the same number of required vehicles. This even holds for a fixed timetable. As an example, consider the timetable at a station depicted in Figure 3.8. All 24 possible matchings at this station attain the same (and therefore minimum) turnaround time. To understand why this symmetry occurs, recall that the greedy algorithm described in Section 3.3 minimizes the total turnaround time, regardless of the order in which the events are matched. In other words, any order leads to an optimal matching with the same total turnaround time, but can lead to a different solution. Different orders do lead to different solutions when the end events and start evens do not alternate. As a result, branching on the matching variables y_a leads to many similar nodes being created, increasing the computational burden.

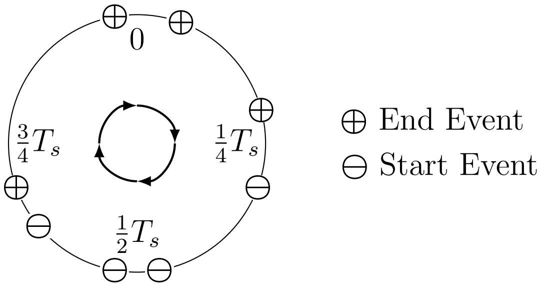


Figure 3.8: Example illustrating the existence of many symmetric solutions.

The symmetries can be broken if we specify an order in which the end events nodes are matched in the greedy algorithm a priori, and incorporate this order in the mathematical formulation. Let $E_{\text{end}}^{<e}$ ($E_{\text{end}}^{>e}$) denote the set of end events that come earlier (later) in the ordering than end event e and consider a turnaround activity (e, f) at some station s with tension $x_{e,f}$. Recall that h_s denotes the minimum separation between departures at station s. In an integer solution, the following cases can occur:

$$w_e \leq \begin{cases} x_{e,f} & \text{if } e \text{ connects to } f \text{ (so } y_{e,f} = 1\text{)}, \\ x_{e,f} + M_3 & \text{if } e' \text{ connects to } f \text{ with } e' \in E_{\text{end}}^{<e}, \\ x_{e,f} - h_s & \text{if } e' \text{ connects to } f \text{ with } e' \in E_{\text{end}}^{>e}. \end{cases} \tag{3.47}$$

Here M_3 is a sufficiently large constant. Intuitively, if e connects to f, the turnaround time will be exactly $x_{e,f}$ by constraints (3.30). In the second case, some e' that comes earlier in the ordering than e turns on f, such that we cannot restrict w_e. In the final case, some e' that comes later in the ordering than e connects to f, which implies that the turnaround time w_e should be at most $x_{e,f} - h_s$, as e has a higher priority and will hence connect to a start event that departs earlier than f. The above stated relations can be captured in the following constraints:

$$w_e \leq x_{e,f} + M_3 \sum_{e' \in E_{\text{end}}^{<e}} y_{e',f} - h_s \sum_{e' \in E_{\text{end}}^{>e}} y_{e',f} \quad \forall s \in S, \quad \forall e \in E_{\text{end}}(s), \quad \forall f \in E_{\text{start}}(s). \tag{3.48}$$

These constraints ensure that for every possible timetable, only one matching that attains the minimum turnaround time is feasible, namely the one induced by the specified order. A sufficiently large value for M_3 is $T_s - h_s$.

3.7 Short Circulations

In practice, operators typically prefer shorter vehicle circulations over longer ones, since short circulations are more robust and easier to manage. If a disturbance occurs in a short circulation, it affects only a small part of the network. In contrast, if a disturbance occurs in a long circulation, the disturbance may propagate through all lines in the circulation, and therefore may have a large impact on the operations.

In this chapter, we consider three types of vehicle circulations. *Fixed* circulations consist of services corresponding to a single line. Hence, if the entire network is operated using fixed circulations, there are dedicated vehicles for all lines. *Combined* circulations consist of services associated with two lines, so vehicles can alternate between lines. *Flexible* circulations consist of services belonging to three or more lines.

Fixed circulations are easily imposed within the VC-PESP by removing all turnaround activities between different lines. For the case where both fixed and combined circulations are allowed, we consider the slightly more restrictive case that every line can be combined with at most one other line. Then, let $\mathcal{L}$ denote the set of lines, let $A_{\text{turn}}^{\text{comb}}$ denote the set of all turnaround activities that connect different lines and let lines(a) denote the pair of lines a combined turnaround connects. If we introduce binary decision variables v_{lm} for all pairs of lines $l, m \in \mathcal{L}$ that are equal to one if the lines are combined and zero else, the following constraints model combined circulations:

$$\sum_{m \in \mathcal{L}, l \neq m} v_{lm} \leq 1 \quad \forall l \in \mathcal{L}, \tag{3.49}$$

$$y_a \leq v_{\text{lines}(a)} \quad \forall a \in A_{\text{turn}}^{\text{comb}}, \tag{3.50}$$

$$v_{lm} \in \{0,1\} \quad \forall l, m \in \mathcal{L}, \quad l \neq m. \tag{3.51}$$

Constraints (3.49) ensure that every line is combined with at most one other line. Constraints (3.50) link the matching variables with the line combination variables.

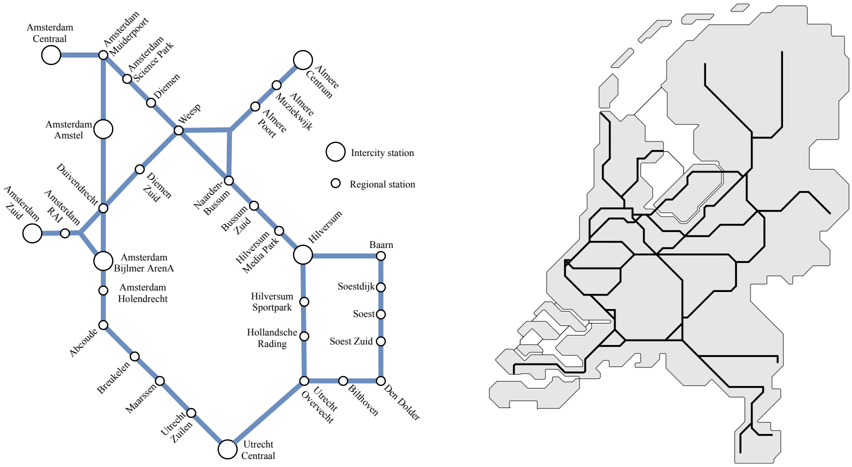
Finally, note that when restrictions are imposed on the number of lines in vehicle circulations, it is not necessary to include all improvements developed in the previous section. If all circulations are fixed, the min-mean cuts (3.43) suffice for directly modeling the turnaround times, so none of the other techniques are needed. In fact, the matching variables and constraints can be omitted entirely. In case combined circulations are imposed, the matching problems at the different stations are linked, implying that they cannot be solved for each station independently. As such, the symmetry-breaking constraints (3.48) are no longer included, as they potentially might cut off the optimal solution and symmetries are already broken.

3.8 Computational Experiments

In order to demonstrate the effectiveness of the proposed formulation, we present the results of a series of experiments. First, we focus on minimizing the number of vehicles to analyze the impact of the contraction techniques, valid inequalities and symmetry-breaking constraints. Thereafter, we consider the trade-off between the number of required vehicles and the quality of the timetable from the passengers' perspective (measured in terms of average perceived travel time).

3.8.1 Instances

The instances used for the experiments are derived from the Dutch railway network. In the Dutch network, two types of lines are operated: *regional* lines that have stops at all intermediate stations and *intercity* lines that only have stops at intercity stations. Both line types have completely dedicated rolling stock types and we account for this in the mathematical model by decomposing the turnaround graphs at each station (that is, only turnaround activities between trips of the same type are considered).



(a) Network for the two-type instances (b) Complete Dutch intercity network

Figure 3.9: Networks used in the computational study.

We consider ten instances that are based on the part of the Dutch railway network depicted in Figure 3.9a. These instances contain both regional and intercity lines and are referred to as two-type instances. To construct these instances, we create ten different line plans for this network from which the event-activity networks are derived. Safety constraints are generated with a headway of 3 minutes based on the railway infrastructure. Besides the two-type instances, we also consider a larger instance with the complete Dutch intercity network with the 2019 line plan, depicted in Figure 3.9b. This instance is referred to as "complete-ic".

Table 3.1 gives for all instances the number of lines $|\mathcal{L}|$, the period in minutes T, the number of events $|E|$, the number of activities $|A|$, the number of turnaround activities $|A_{\text{turn}}|$ and a lower bound on the number of required vehicles $\underline{n}$ (computed using minimum driving and dwell times). In order to optimize the perceived travel time, we employ an origin destination matrix with passenger counts. The origin destination pairs are routed over the network based on the line plan, giving the number of passengers per activity. We then define the perceived travel time of a route as the sum of expected waiting time for the first service, the in-vehicle time and an additional transfer penalty of 10 minutes plus twice the transfer time. The activity weights λ_a are set accordingly.

The experiments are performed on a machine with an Intel Xeon E5-2650 v2 2.60Ghz processor with 16 cores and 64 GB RAM. CPLEX 12.8.0 with default settings is used to solve the mixed-integer programs, using 16 parallel threads. As the number of valid inequalities is relatively small, we do not separate violated valid inequalities dynamically but add them directly to the model.

Table 3.1: Characteristics of the instances used in the computational study.

Instance	$\|\mathcal{L}\|$	T	$\|E\|$	$\|A\|$	$\|A_{\text{turn}}\|$	$\underline{n}$
two-type1	10	60	522	732	90	15
two-type2	15	60	684	1000	122	20
two-type3	14	60	752	1105	138	21
two-type4	9	60	908	1405	200	26
two-type5	10	60	1030	1643	290	28
two-type6	10	30	522	734	90	29
two-type7	12	60	1122	1899	318	31
two-type8	15	60	1158	1932	380	33
two-type9	14	30	714	1028	124	40
two-type10	11	20	550	747	78	47
complete-ic	23	60	3932	6683	280	140

3.8.2 Comparing the Formulations

To examine the effectiveness of the proposed techniques, we define the following formulations:

- *Orig*: the original formulation presented in Section 3.4.
- *Con*: the formulation with the contraction technique described in Section 3.6.1.
- *Con+Val*: the formulation with contraction and with the valid inequalities described in Section 3.6.2.
- *Con+Sym*: the formulation with contraction and with the symmetry-breaking constraints described in Section 3.6.3.
- *Con+Val+Sym*: the formulation with contraction, valid inequalities and the symmetry-breaking constraints.

Size of the Formulations

Before we test the actual performance of the formulations, we first examine the impact the contraction technique and the valid inequalities have on the size of the formulation. In Table 3.2, we present the number of binary and continuous variables and the number of constraints that are included in the formulations. Only the variables and constraints related to the matching part of the formulation are included here. It can be seen that the contraction on average reduces the number of binaries by more than a factor of two. Despite the auxiliary variables and constraints that are introduced, the contraction of the turnaround graphs also results in an overall decrease of the number of continuous variables and constraints. The reduction in the number of binaries varies quite strongly over the instances, as it is really dependent on the properties of the line plan. This reduction is the largest for the complete-ic instance, from 280 to 61. This can be explained by the fact that the local period in the densely populated parts of the Netherlands is often 30 or 15 minutes, whereas the global period of the line plan is 60 minutes. It is exactly in these situations that the contraction technique is useful.

Table 3.2: The number of binary variables, continuous variables and constraints required by the different formulations in the matching part of the model.

	Orig			Con			Con+Val			Con+Sym			Con+Val+Sym		
Instance	Bin.	Cont.	Const.	Bin.	Cont.	Const.	Bin.	Cont.	Const.	Bin.	Cont.	Const.	Bin.	Cont.	Const.
two-type1	90	113	220	78	99	208	75	96	222	78	99	262	75	96	276
two-type2	122	152	300	122	152	300	120	150	317	122	152	390	120	150	407
two-type3	138	171	336	126	157	324	123	154	342	126	157	416	123	154	434
two-type4	200	239	476	68	89	336	65	86	340	68	89	380	65	86	384
two-type5	290	340	680	83	108	476	81	106	481	83	108	532	81	106	537
two-type6	90	113	220	78	99	208	75	96	222	78	99	262	75	96	276
two-type7	318	376	752	81	109	436	80	108	448	81	109	488	80	108	500
two-type8	380	434	868	161	196	680	159	194	691	161	196	804	159	194	815
two-type9	124	155	304	112	141	292	109	138	309	112	141	372	109	138	389
two-type10	78	101	196	78	101	196	75	98	211	78	101	248	75	98	263
complete-ic	280	356	712	61	73	344	38	50	266	61	73	370	38	50	292
Average	192	232	460	95	120	345	91	116	350	95	120	411	91	116	416

The valid inequalities are also able to slightly decrease the number of variables. This can be explained by the fact that if only a single line terminates at a station, the min-mean cuts provide an exact description of the minimum turnaround time, such that the matching variables and constraints for that station can be omitted. For some of the instances, this implies that even the total number of constraints is reduced when the valid inequalities are added. Including the symmetry-breaking constraints does lead to an increase of the number of constraints, on average with 66 constraints.

Tightness of the Formulations

To gain more insight into the effect the contraction technique and the valid inequalities have on the strength of the formulation, we perform the following experiment. First, we find a timetable $\mathbf{x}$ by minimizing, to optimality, the number of vehicles using the original formulation with all variables relaxed, except the periodicity variables. As the periodicity variables still are required to be integer, the resulting solution corresponds to a feasible timetable. We let $n(\mathbf{x})$ denote the number of vehicles this timetable requires. Next, we fix the timetabling part of the model entirely and solve the linear relaxation of the remainder of the model with the different formulations. We let $n^{LP}(\mathbf{x})$ denote the solution value of this relaxed problem. In a tight formulation, the integrality gap, which we define as $\frac{n(\mathbf{x})}{\lceil n^{LP}(\mathbf{x}) \rceil}$, should be small (we take the ceiling as the number of vehicles is integer).

Table 3.3: Results of solving the linear relaxations of the formulations with a fixed timetable.

		Orig		Con		Con+Val	
Instance	$n(\mathbf{x})$	$n^{LP}(\mathbf{x})$	Int. Gap	$n^{LP}(\mathbf{x})$	Int. Gap	$n^{LP}(\mathbf{x})$	Int. Gap
two-type1	21	14.3	1.40	14.7	1.40	16.7	1.24
two-type2	26	19.0	1.37	19.0	1.37	21.5	1.18
two-type3	25	20.2	1.19	20.5	1.19	23.1	1.04
two-type4	31	24.4	1.24	25.5	1.19	28.1	1.07
two-type5	34	27.5	1.21	28.1	1.17	31.2	1.06
two-type6	34	28.7	1.17	29.0	1.13	31.5	1.06
two-type7	38	30.0	1.27	31.3	1.19	35.5	1.06
two-type8	41	32.7	1.24	32.9	1.24	36.9	1.11
two-type9	47	38.5	1.21	38.7	1.21	41.7	1.12
two-type10	52	46.0	1.13	46.0	1.13	48.3	1.06
complete-ic	151	140.6	1.07	146.9	1.03	149.7	1.01
Average	45.5	38.3	1.23	39.3	1.20	42.2	1.09

In Table 3.3, the results of solving the relaxations are presented for the formulations *Orig*, *Con* and *Con+Val*. The other formulations are not included since the symmetry-breaking constraints do not have an impact on the strength of the formulation. We can observe that the contraction technique and valid inequalities are able to reduce the integrality gap considerably. For example, for instance two-type1, the timetable that is found in the first step requires 21 vehicles. In the linear relaxation however, the original formulation finds a solution with 14.3 vehicles, giving a gap of 1.4. The formulation with contraction finds a slightly better bound of 14.7. An even stronger bound of 16.7 is obtained by the formulation that also includes the valid inequalities, driving down the integrality gap to 1.24. In the other instances, the same pattern is visible. On average, the integrality gap decreases from 1.23 to 1.09 when the improvements are included in the formulation. All in all, this experiment illustrates that the linear relaxation becomes much more informative about the number of required vehicles by a timetable when the contraction technique and especially the valid inequalities are included in the model.

Performance of the Formulations

We next perform two experiments to assess the practical performance of the formulations. In both experiments, we solve the VC-PESP with the objective to minimize the number of required vehicles, with flexible circulations, using all five formulations. In the first experiment, we solve each instance with five different seeds to take into account that the performance of CPLEX can vary with the seed that is used. For this experiment, we use a time limit of 15 minutes and consider the average performance over the five runs. In the second experiment, we use a longer time limit of 1 hour and perform a single run.

Table 3.4: The average number of required vehicles, the average relative optimality gap (as a percentage, given by CPLEX) and the number of runs out of five that are solved to optimality, obtained by the different formulations with a time limit of 15 minutes.

	Orig			Con			Con+Val			Con+Sym			Con+Val+Sym		
Instance	Veh.	Gap	Opt.	Veh.	Gap	Opt.	Veh.	Gap	Opt.	Veh.	Gap	Opt.	Veh.	Gap	Opt.
two-type1	16.0	1.2	4	16.0	0.0	5	16.0	0.0	5	16.0	0.0	5	16.0	1.2	4
two-type2	20.0	0.0	5	20.0	0.0	5	20.2	1.0	4	20.2	1.0	4	20.0	0.0	5
two-type3	22.8	7.8	0	22.4	6.1	1	21.6	2.6	3	21.6	2.6	3	21.4	1.8	3
two-type4	26.4	1.5	3	26.0	0.0	5	26.0	0.0	5	26.0	0.0	5	26.0	0.0	5
two-type5	31.2	10.2	0	30.2	7.3	0	30.2	7.3	0	30.0	6.7	0	30.0	6.7	0
two-type6	30.0	0.0	5	30.0	0.0	5	30.0	0.0	5	30.0	0.0	5	30.0	0.0	5
two-type7	35.2	11.8	0	33.4	7.2	0	33.2	6.6	0	33.2	6.6	0	33.0	6.1	0
two-type8	38.4	14.0	0	37.0	10.8	0	37.2	11.3	0	37.2	11.3	0	36.8	10.3	0
two-type9	41.4	3.4	0	40.8	1.9	2	42.0	4.7	0	40.6	1.4	3	40.2	0.5	4
two-type10	48.0	0.0	5	48.0	0.0	5	48.0	0.0	5	48.0	0.0	5	48.0	0.0	5
complete-ic	150.8	5.2	0	148.8	3.4	0	148.2	3.1	0	149.6	4.0	0	148.0	2.6	0
Average	41.8	5.0	2.0	41.1	3.3	2.5	41.1	3.3	2.5	41.1	3.1	2.7	40.9	2.6	2.8

In Table 3.4, we present the results of the first experiment, stating the average number of vehicles, the relative optimality gap and the number of times (out of five runs with different seeds) a provably optimal solution is obtained with the different formulations. On average, the original formulation clearly is the worst performing formulation, resulting in the largest number of vehicles, the largest optimality gap and the fewest number of optimal solutions. However, there is some variation across the instances. For instances two-type1, two-type2, two-type6, two-type10, all formulations perform (roughly) equally well. This can be explained by the fact these are precisely the instances with the smallest number of turnaround arcs (see Table 4.2), implying that the vehicle circulations scheduling part of the formulations is relatively small. Therefore, it is likely that for these instances, the impact of the enhancements is limited. For the other seven instances, where the number of turnaround arcs is larger, the original formulation does result in strictly the largest number of vehicles and optimality gap, except for instance two-type9. Introducing the contraction technique considerably improves the performance, most notably on two-type5, two-type7, two-type8 and the complete intercity instance. Adding the valid inequalities on top of the contraction does not improve the average performance. The formulation that includes symmetry-breaking constraints is not able to find solutions that on average require fewer timetables, but is able to find better lower bounds, resulting in a smaller average gap. Moreover, it turns out that when both the valid inequalities and the symmetry-breaking constraints are included in the formulation, the average performance does further improve: the *Con+Val+Sym* formulation attains both the lowest average number of vehicles and the lowest average gap. In addition, on all instances this formulation finds the same or a lower number of vehicles than the other formulations. Compared to the original formulation, *Con+Val+Sym* reduces the average number of required vehicles by 0.9 (2.2 percent) and almost halves the average optimality gap. *Con+Val+Sym* also finds the optimal solution most often.

Table 3.5: The number of required vehicles, the relative optimality gap (as a percentage, given by CPLEX) and the computation time, obtained by the different formulations with a time limit of 1 hour.

	Orig			Con			Con+Val			Con+Sym			Con+Val+Sym		
Instance	Veh.	Gap	Time (s)	Veh.	Gap (%)	Time (s)	Veh.	Gap	Time (s)	Veh.	Gap	Time (s)	Veh.	Gap	Time (s)
two-type1	16	0.0	45	16	0.0	27	16	0.0	32	16	0.0	52	16	0.0	273
two-type2	20	0.0	3265	20	0.0	34	20	0.0	197	20	0.0	486	20	0.0	34
two-type3	22	4.8	3600	22	4.8	3600	23	8.7	3600	22	4.8	3600	22	4.8	3600
two-type4	27	9.3	OOM	26	4.8	334	26	0.0	97	26	0.0	248	26	0.0	44
two-type5	31	9.7	OOM	30	6.7	3600	31	9.7	3600	30	6.7	3600	30	6.7	3600
two-type6	30	0.0	15	30	0.0	11	30	0.0	40	30	0.0	6	30	0.0	21
two-type7	34	8.8	3600	33	6.1	3600	33	6.1	3600	33	6.1	3605	33	6.1	3600
two-type8	37	10.8	3600	36	8.3	3600	36	8.3	3600	34	2.9	3600	34	2.9	3600
two-type9	42	4.8	3600	41	2.4	3600	42	4.8	3600	40	0.0	1291	40	0.0	964
two-type10	48	0.0	33	48	0.0	12	48	0.0	41	48	0.0	17	48	0.0	23
complete-ic	151	5.3	3600	149	3.4	OOM	147	2.0	3600	147	2.0	3600	147	2.0	3600
Average	41.6	4.9	2596	41.0	3.3	2002	41.1	3.6	2001	40.5	2.0	1828	40.5	2.0	1760

Note. OOM, out-of-memory. The OOM computation times are set to 3600s for computing the average computation time.

The results of the second experiment are presented in Table 3.5, stating the number of vehicles, optimality gap and computation time obtained with the different formulations. The results are in line with those of the previous experiment: the original formulation attains an average optimality gap of 4.9 percent compared to 2.0 percent using the formulation with all three enhancements. The contraction technique and symmetry-breaking constraints appear to lead to the largest improvement in performance. The formulations *Con+Sym* and *Con+Val+Sym* achieve the same exact number of vehicles and optimality gap for all instances. In terms of computation time, *Con+Val+Sym* performs better, but the difference is small. Combined with the results of Table 3.4, we can conclude that the *Con+Val+Sym* formulation is the most effective formulation for solving the VC-PESP, yielding considerable improvements over the original formulation. Finally, it is important to note that these results also illustrate the limitations of the proposed enhancements. Despite that the improved formulations perform much better than the original formulation, *Con+Val+Sym* is still unable to solve five out of the eleven instances to optimality. Moreover, the gap improvement compared to shorter running time is only due to finding better feasible solutions. The lower bounds are identical to those found in the previous experiment, indicating that the relaxation is still relatively weak.

3.8.3 Trade-Off Between Number of Vehicles and Travel Time

To analyze the trade-off between the number of vehicles and the average travel time, we use the concept of a Pareto-efficient timetable, where it is impossible to reduce the number of vehicles without increasing the average travel time, and vice-versa. To approximate the set of Pareto-efficient timetables, we first separately minimize the average travel time and the number of vehicles, resulting in an upper and lower bound on the minimum number of required vehicles. These first two problems are solved with a time limit of 30 minutes. Next, for every integer m between the lower bound and the upper bound, the VC-PESP is solved with the average travel time as the objective and the constraint that the number of vehicles is at most m. For these problems, the time limit is set to 15 minutes. As we do not solve the problems to optimality, this results only in an approximation of Pareto-efficient solutions. The best performing formulation found in the previous section, *Con+Val+Sym*, is used in these experiments.

We approximate the Pareto-efficient solutions for the cases with fixed circulations, combined circulations and flexible circulations. To speed up the computations, we use the obtained solutions with fixed circulations as starting solutions for the combined circulations, and the obtained solutions with combined circulations as starting solutions for the flexible circulations.

Two-Type Instances

The obtained solutions for the two-type instances are visualized in Figure 3.10. The figure also depicts the sequential solutions obtained by first optimizing the timetable to minimize the travel time and subsequently optimizing the vehicle circulations for the found timetable. It can be observed that sequential optimization leads to very inefficient timetables. In many cases, the number of vehicles of these solutions can be reduced without any increases in travel time. Furthermore, if an increase in travel time is required to reduce the number of vehicles, this increase is typically very limited. In general, it can be observed that reductions in the number of vehicles only require small increases in average travel time, but the required increases progressively become larger when the number of vehicles approaches its minimum.

A second observation from Figure 3.10 is that, as expected, the flexible solutions dominate the combined solutions, which in turn dominate the fixed solutions. Especially the difference between the combined solutions and the fixed solutions is considerable. The benefit of flexible circulations over combined circulations is less consistent over the instances, but still clearly noticeable for two-type5, two-type9 and two-type10. For seven of the instances, flexible circulations do allow finding solutions with fewer vehicles than with combined circulations, but these timetables on the far end of the curve are often much less attractive from the passengers' side.

To further analyze the benefit of integrating timetabling and circulation scheduling, Figure 3.11 depicts the increase in travel time and decrease in number of vehicles for all Pareto-efficient solutions, relative to the sequential solution. The key insight is that substantial decreases in the number of required vehicles can be achieved at the cost of very limited increases in travel time. Moreover, for combined circulations it is possible to decrease the number of vehicles without any increases in travel time for six out of the ten instances. For fixed and flexible circulations, this is possible in five of the instances. If a 0.01 percent travel time increase is permitted, significant savings in vehicles can be obtained for the majority of the instances. This clearly shows that the timetable obtained by sequentially optimizing the travel time and the number of vehicles is strongly inferior to timetables obtained by jointly optimizing the travel time and the number of vehicles. All in all, the results indicate that the number of vehicles can typically be reduced considerably without doing any serious harm to the passengers' perspective.

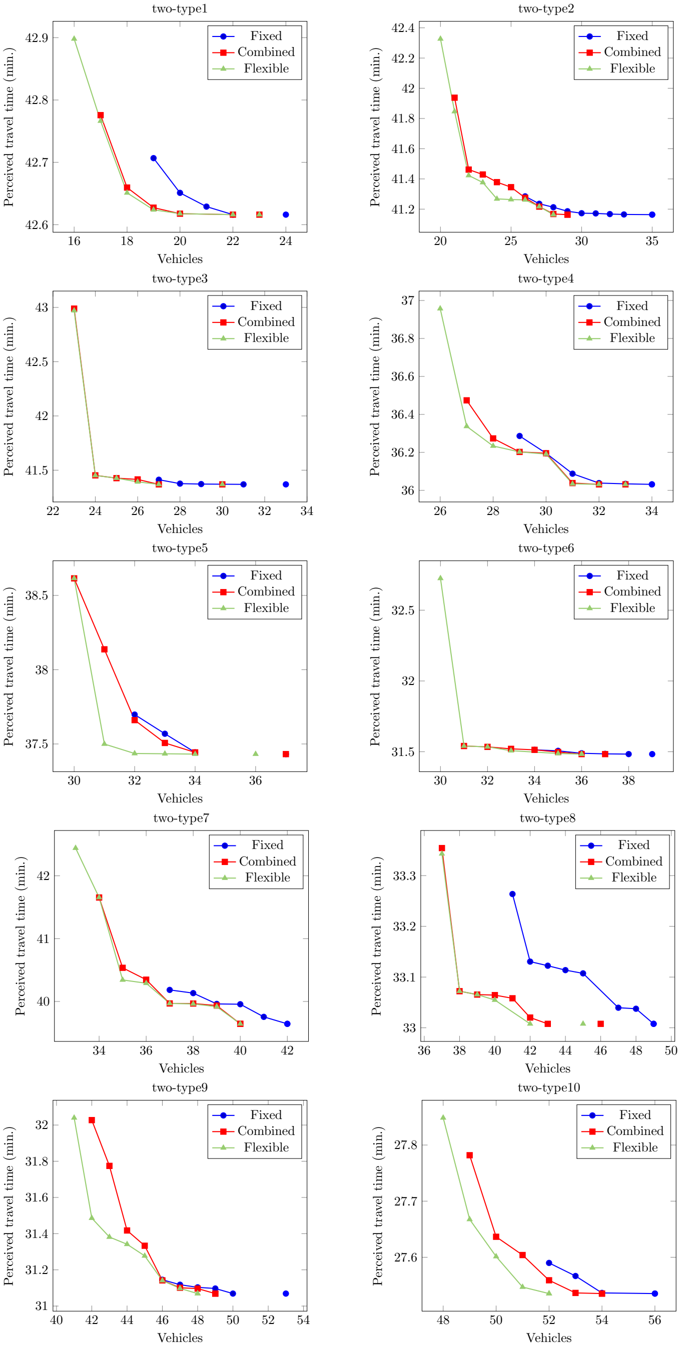


Figure 3.10: The approximated Pareto-efficient solutions and the sequential solutions. In some cases, the sequential solutions are not Pareto-efficient, these points are disconnected from the curve.

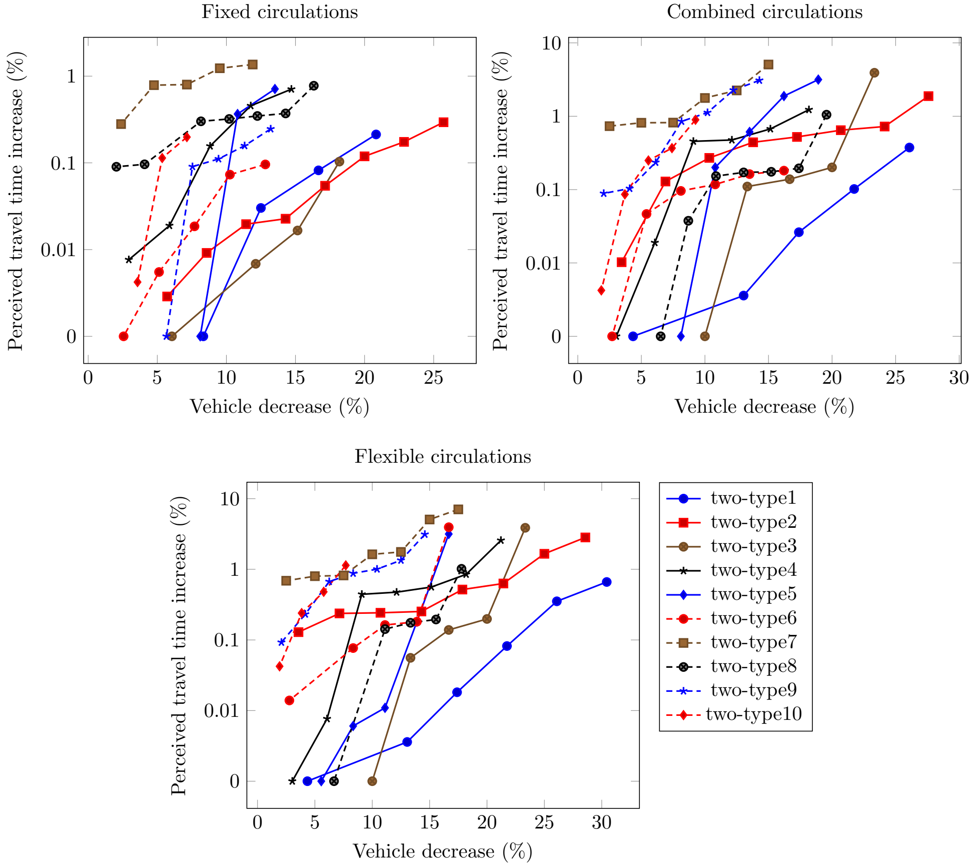
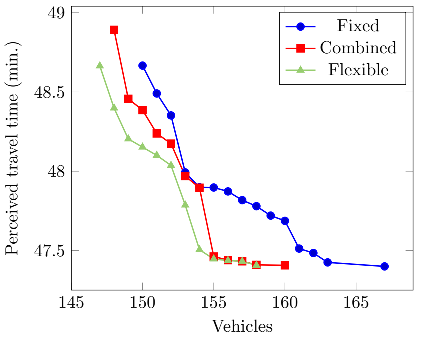


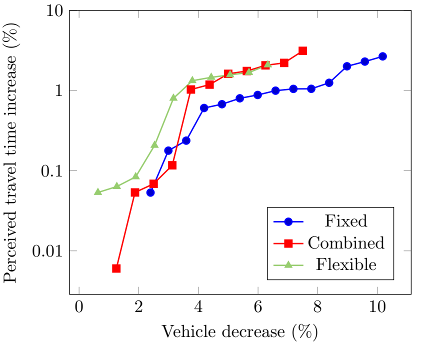
Figure 3.11: Increase in travel time (on a logarithmic scale) plotted against the decrease in the number of vehicles, both relative to the sequential solution.

Complete Intercity Network

The results for the complete Dutch intercity network are visualized in Figure 3.12. The plot on the left depicts the approximated Pareto curve and the plot on the right depicts the increase in perceived travel time and decrease in number of vehicles for the obtained solutions, relative to the sequential solution. As in the two-type instances, we can observe that it is possible to realize a decrease in the number of vehicles compared to the sequential solution at the expense of a relatively small increase in the average perceived travel time. On the other hand, a decrease in the number of vehicles of five percent or more is only possible if the travel time is increased by



Approximated Pareto-efficient solutions



Increase in travel time against decrease in vehicles, relative to sequential solution

Figure 3.12: Results of the complete intercity instance.

slightly less than one percent for fixed circulations, and more than one percent for combined and flexible circulations, whereas this required much smaller travel time increases for most two-type instances. Moreover, the maximum attainable relative vehicle decrease for the two-type instances is much larger. This difference can likely be attributed to the larger size of the complete-ic instance, as the longer driving times of the lines imply that the number of vehicles is to a smaller extent determined by the turnaround times. Hence, optimizing the turnarounds has a relatively smaller effect on the number of vehicles.

When we compare the results of the different types of vehicle circulations, we observe the same patterns as in the results of the two-type instances. The benefit of allowing circulations with multiple lines is clear, especially when the number of vehicles is 155 or more. For example, if there are 155 available vehicles, the average perceived travel time under fixed circulations is almost 30 seconds longer than under combined or flexible circulations. The additional benefit of allowing circulations to contain more than two lines is smaller. When the number of vehicles is 155 or more, the solutions with combined and flexible circulations even coincide.

Next, we analyze the timetables obtained under flexible circulations in more detail, to get better insight into the impact that decreasing the number of vehicles has on the journeys of passengers. Figure 3.13 presents the average perceived travel times of these timetables, separately for direct passengers. As can be seen, the travel time for these passengers hardly differs over the different solutions. This can be explained by the fact that both the travel times of direct passengers and the number of vehicles benefit from keeping driving and dwell times as small as possible. It follows that the increase in average travel time that is necessary to realize a decrease in the number of vehicles is almost entirely passed on to the transfer times of transferring passengers.

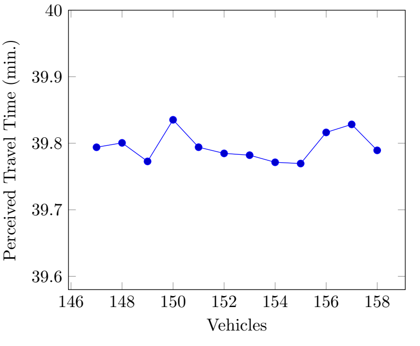


Figure 3.13: Perceived travel times of direct passengers with timetables requiring different number of vehicles.

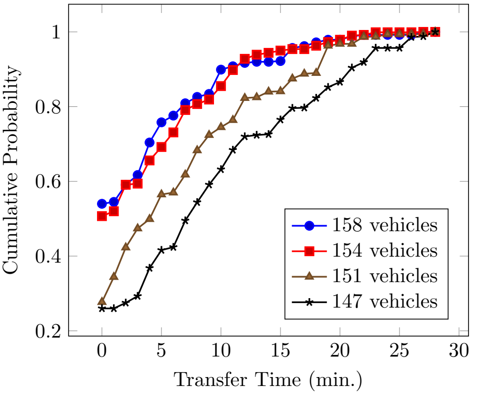


Figure 3.14: Empirical cumulative distribution function of the transfer time of transferring passengers with timetables requiring a different number of vehicles.

To illustrate how the transfer times change when the number of vehicles is decreased, Figure 3.14 displays the empirical cumulative distribution function of the transfer times of passengers with four different timetables. As expected, the timetable requiring the most (158) vehicles offers the shortest transfer times to passengers. For over half of the transfers, the arrival and departure of the connecting trains are perfectly synchronized, with a transfer time of 0 minutes. The timetable requiring 154 vehicles realizes approximately the same transfer times. This is also consistent with

Figure 3.12, where we observed that compared to the sequential solution, it is possible to achieve a small reduction in the number of vehicles without a large increase in average travel time. When the number of vehicles is reduced to 151, there is a clear increase in transfer time. With this timetable, still over half of the transfers are shorter than 5 minutes. Transfer times are increased further with the timetable requiring the smallest number of vehicles. Here, a quarter of the transferring passengers has a transfer longer than 15 minutes. As long transfers are very much disliked by passengers, these results illustrate that when considering the trade-off between the number of vehicles and travel times of passengers, it is important to not only assess the average travel time, but also the travel times of transferring passengers separately.

3.9 Conclusion

We presented a new periodic timetabling approach that allows to explicitly consider the trade-off between passengers' travel time and the number of required vehicles. We introduced contraction techniques, valid inequalities and symmetry-breaking constraints to cope with the complexities of this integrated problem.

Computational results based on the Dutch railway network illustrate the value of our approach. The improved formulation performs better when it comes to finding timetables requiring few vehicles compared to an existing formulation, approximately halving the average optimality gap. Furthermore, compared to sequential optimization of the travel time and the number of vehicles, we are able to find timetables requiring considerably fewer vehicles with only minimal increases of average travel times. As such, our approach gives operators the opportunity to save costs without strongly decreasing the level of service, or conversely, increasing the level of service by introducing new lines or raising frequencies without strongly increasing the costs.

For further research, the presented approach could be embedded in an (iterative) scheme for integrated public transport planning, where the goal is to find a line plan, timetable and vehicle schedule that are both inexpensive to operate and attractive to passengers. Furthermore, our methods could be applied in different periodic scheduling problems, such as aircraft scheduling or job-shop scheduling. Finally, it would be interesting to consider a further integration between timetabling and vehicle scheduling in railway contexts, where vehicles can be coupled to meet passenger demand.

Part II

Decentralized Control in Public Transport

Chapter 4

A Self-Organizing Policy for Vehicle Dispatching in Public Transit Systems with Multiple Lines

This chapter is based on Van Lieshout, R.N., Bouman, P.C., Van den Akker, J.M. and Huisman, D. (2021). A Self-Organizing Policy for Vehicle Dispatching in Public Transit Systems with Multiple Lines. Transportation Research Part B: Methodological 152, 46–64.

4.1 Introduction

Self-organizing strategies are a promising concept to increase the resilience of urban public transit systems. In such a strategy, the concept of a schedule or timetable is abandoned and instead, departure times and/or destinations of vehicles are determined locally at stations according to an easy-to-implement policy. In the absence of perturbations, an adequate self-organizing policy causes the system to converge to some preferable state, typically a periodic repetition of services with constant headways (the time between consecutive services). As a result, the impact of disruptions always dies out spontaneously, without intervention by a central control authority. In this chapter, we propose and analyze an easy-to-implement decentralized policy for dispatching vehicles in a multi-line public transit system and prove the resulting system exhibits self-organizing behavior.

There are multiple advantages of self-organizing strategies over centralized approaches to dispatch vehicles. First of all, a self-organizing strategy completely eliminates any need for the public transit operator to constantly track the vehicles, monitor schedule adherence and adjust schedules in case of disruptions. Given the cost and effort that comes with applying centralized control through a decision support system, avoiding the need for such a system altogether can be an attractive alternative, especially for smaller operators, operators in developing countries or for services that are only operated on an incidental basis, for example to replace disrupted train lines. Secondly, centralized control requires communication between the vehicles or local dispatchers and a central control center, where dispatching decisions are made. This process, involving communication, determining a new schedule and coordination, can be time consuming and prone to errors. Especially if rescheduling needs to be performed frequently, for example because congestion causes travel times to be highly volatile, this can be cumbersome. Moreover, public transit systems and railway systems in particular may suffer from *out-of-control* situations, where extreme events such as power outages or blizzards result in largely disrupted operations (Dekker et al., 2021; Van Lieshout et al., 2020). In such situations, centralized rescheduling approaches are ineffective due to the sheer size of the disruptions and a lack of complete information available at the central control center. In contrast, a self-organizing strategy is typically easy and very fast to apply in all situations, without requiring communication or even the use of a computer. A self-organizing strategy could also serve as a back-up plan: operators may want to apply centralized control as a default and switch to the self-organizing approach when disruptions make it too complex to manage the system centrally. Of course a disadvantage of the self-organizing approach is that passengers cannot rely on a schedule to plan their trips. Therefore, this approach is most suited for urban high-frequency networks, as passengers tend to arrive approximately uniformly if the headway is smaller than 10 minutes, despite a schedule being available (Fan and Machemehl, 2009).

The self-organizing approach has already been introduced in the context of public transit by Bartholdi and Eisenstein (2012), who developed a simple rule for holding buses at a control point to reduce headway variation (variation in the time between consecutive services) and prevent bus bunching. The authors show analytically that for the case with a single circular line, as long as there are no perturbations, under their policy any starting position will converge to some fixed point where the headways between all vehicles are equal. Note that this directly implies that whenever there is a perturbation, the headways will automatically self-equalize after some time. Even when one of the vehicles breaks down, a new system headway will naturally emerge. This approach was later extended by Liang et al. (2016) and Zhang and Lo (2018), who consider both the backward headway and the forward headway when deciding how long a vehicle should wait at a control point.

In this chapter, we extend the literature on self-organizing dispatching strategies by considering more complex public transit networks. Specifically, we consider networks consisting of multiple lines, with the condition that all lines have the same target headway. In such a system, when a vehicle reaches a terminal station of a line one not only needs to decide *when* to depart again, but also *which line* to perform. For this problem, we propose and theoretically analyze an easy-to-implement decentralized dispatching policy. In our policy, every terminal station maintains a fixed cyclic ordering of its outgoing lines and keeps track of the most recent departure times of these lines. Vehicles arriving at the station are assigned to the outgoing lines in round-robin fashion, that is, a vehicle is assigned to each line in turn according to the fixed ordering. The departure times of vehicles are chosen such that deviations from the target headway are minimized.

Our policy can be viewed as a generalization of the *rotor-router model*, originally introduced by Priezzhev et al. (1996). In the rotor-router model, one or more agents move over a graph at discrete and synchronous steps. Nodes propagate agents over the network in round-robin fashion, similar to our policy. The main difference between our policy and the rotor-router model is that our policy uses the concept of a target headway, sometimes instructing vehicles to wait in order to meet the target headway. Furthermore, we also allow lines to have different travel times, while in the rotor-router model it always takes one time unit until to traverse an edge. These two differences completely change the dynamics of the system, such that the properties of our policy cannot be derived from known results in the rotor-router literature.

Our main theoretical contribution is that we prove that our policy is self-organizing, leading to emergent behavior. Once converged, the decentralized policy matches the performance that can be achieved under centralized control. As long as the number of vehicles is large enough to perform a schedule meeting the target headways, our policy guarantees convergence to such a schedule. This result holds regardless of the initial locations of the vehicles. As a consequence, even when one of the vehicles breaks down or a bus returns to the depot at the end of the driver's shift, the remaining vehicles spontaneously redistribute over the network to again meet the target headway of all lines. In case the number of vehicles is not sufficient to meet the target headways using a centralized approach, we prove that our policy keeps the headways, on average, as small as possible given the number of available vehicles. We also derive upper bounds on the largest headway that can occur and the stabilization time. Finally, we also show that the assumption of the common target headway is crucial to our analysis. If target headways are different, matching the performance of centralized control is equivalent to solving an NP-complete problem. Therefore, there does not exist a policy that requires no more than the optimal number of vehicles and is guaranteed to stabilize within polynomial time (unless P=NP).

Besides a theoretical analysis, we also assess the practical performance of our policy in numerical experiments. We show that after a vehicle breakdown, the target headway is restored quickly, especially when there is more than one spare vehicle available. In other experiments, we analyze the impact of relaxing the assumptions that are required for the theoretical results. We find that the performance of the policy degrades if the target headways are different or time varying or the travel times are stochastic, but this can be compensated by having some buffer in the system in terms of the number of available vehicles. Experiments on the transit systems of the cities of Göttingen, Amersfoort and The Hague support the conclusion that a relatively small buffer is sufficient for attaining good performance.

The remainder of this chapter is structured as follows. In Section 4.2, we describe the problem setting and explain the policy. In Section 4.3, we discuss related literature. In Section 4.4, we theoretically analyze the performance of the policy. In Section 4.5, we discuss the results of a series of experiments that illustrate the practical performance of the policy. Finally, we conclude the chapter in Section 4.6.

4.2 The Policy

4.2.1 Problem Setting and Notation

We represent the public transit system by a directed network $\mathcal{G} = (\mathcal{S}, \mathcal{L})$, where $\mathcal{S}$ is the set of terminal stations and $\mathcal{L}$ the set of lines. The intermediate stations are not relevant for our policy and therefore not included in the network. We assume that the network is symmetric, such that for every line $(s \to s') \in \mathcal{L}$, the reverse line $(s' \to s)$ is also an element of $\mathcal{L}$. Furthermore, we assume that $\mathcal{G}$ is connected (otherwise the connected components can be considered separately). For line $l = (s, s') \in \mathcal{L}$, the time between a departure of a vehicle at s and its arrival at the other terminal station s' is referred to as the travel time of line l and denoted by t_l. We allow for asymmetric travel times, so the travel time of a line and its reverse line are not required to be equal. Every line has the same target headway, which we denote as H. In other words, the goal is to operate each line every H time units. We assume that all travel times and H are integer. Fractional inputs can be converted into integers by scaling. We let $\delta^+(s)$ and $\delta^-(s)$ and denote the set of lines originating and terminating at $s \in \mathcal{S}$, respectively.

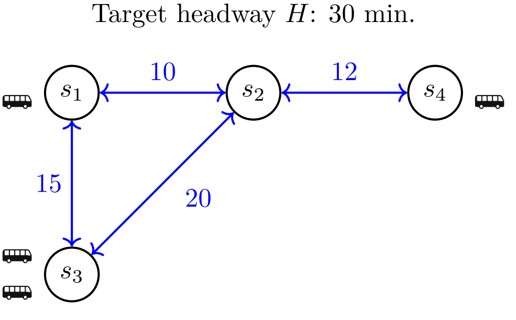


Figure 4.1: Illustration of the problem setting. Travel times are symmetric and given in minutes.

We assume there is a fixed number of vehicles available in the system, which we denote as n. At the moment of initialization, all vehicles are at stations. Vehicles are allowed to switch between lines at the terminal stations, but are not allowed to deadhead (drive without passengers). Therefore, after a vehicle performs line $(s \to s')$, the next line the vehicle is assigned to must be an element of $\delta^+(s')$. To meet all the target headways, one needs at least n^* vehicles, with

$$n^* = \frac{\sum_{l \in \mathcal{L}} t_l}{H}.$$

In general, n^* may be fractional, so it can be rounded up to the next integer to obtain a stronger bound. Furthermore, this bound on the number of required vehicles does not depend on whether the system is operated using a centralized or a decentralized approach. Although operators will naturally choose a target headway that is feasible given their fleet size, we consider both the case where $n \geq n^*$ and where $n < n^*$, as the latter may be relevant when there is a breakdown of a vehicle or travel times are longer than anticipated.

A visual illustration of the problem setting is provided in Figure 4.1, depicting a network of four stations, four lines and four vehicles. In this example, $n^* = 3.8$, so at least four vehicles are necessary to meet the target headways.

4.2.2 Policy Definition

We now propose a policy for dispatching vehicles at a terminal station. The policy determines the next line and the next departure time of a vehicle arriving at a terminal station. In the policy, the lines starting at a terminal station are selected in round-robin fashion, according to a fixed (but arbitrary) cyclic order. Departure times are based on the previous departure times of the lines, which are assumed to be known at the station. The departure time is taken to be the maximum of the target departure time, which is equal to the sum of the previous departure time and the target headway, and the current time (as it is not possible to depart in the past). Note that any minimum required time between services can be incorporated in the definition of the travel times, so we assume without loss of generality that an arriving vehicle can depart immediately.

As an example, suppose a vehicle arrives at station s_2 from Figure 4.1 at 9:10. Table 4.1a displays the relevant information at s_2 at this time, indicating which line should be performed next, the previous departure times of the lines starting at s_2 and the target departure times. Our policy assigns the arriving vehicle to line $(s_2 \to s_4)$, as it is indicated that this line should be performed next. Naturally, the

Table 4.1: The state at station s_2 at two different time instants.

(a) Current time: t = 9:10

Line	**Next**	**Prev. Dep.**	**Target Dep.**
$s_2 \to s_1$		8:50	9:20
$s_2 \to s_3$		9:00	9:30
$s_2 \to s_4$	●	8:35	9:05

(b) Current time: t = 9:15

Line	**Next**	**Prev. Dep.**	**Target Dep.**
$s_2 \to s_1$	●	8:50	9:20
$s_2 \to s_3$		9:00	9:30
$s_2 \to s_4$		9:10	9:40

previous departure time of this line is also the longest ago. As the target departure time has already passed, the departure time is set at 9:10. Table 4.1b shows the updated information after the departure. Note that the target departure time of line $(s_2 \to s_4)$ is now equal to 9:40, as it is only based on the most recent departure time. Suppose that the next arrival occurs at time 9:15. The policy assigns the arriving vehicle to line $(s_2 \to s_1)$. As the target departure time of line $(s_2 \to s_1)$ is 9:20, the policy instructs the vehicle to wait for 5 minutes and depart exactly at 9:20.

For a formal definition of the policy, let us (arbitrarily) order the lines starting at station $s \in \mathcal{S}$ as $l_1^s, l_2^s, ..., l_{|\delta^+(s)|}^s$, representing the cyclic order in which the lines from this station are performed. Let $l_{\text{next}}^s \in \delta^+(s)$ denote the next line to be performed from station s and let τ_l denote the current target departure time of line l (at initialization, all target departure times are 0 and $l_{\text{next}}^s = l_1^s$). Suppose at time t_{now}, a vehicle arrives at station s and $l_{\text{next}}^s = l_i^s$. Our policy assigns the arriving vehicle to line l_i^s and schedules it at time $t' = \max\{\tau_{l_i^s}, t_{\text{now}}\}$. Next, the policy updates the target departure time of the selected line:

$$\tau_{l_i^s} \leftarrow t' + H.$$

Finally, the policy updates l_{next}^s according to the order of the lines:

$$l_{\text{next}}^s \leftarrow l_{(i \bmod |\delta^+(s)|)+1}^s.$$

4.3 Related Literature

4.3.1 Self-Organizing Approaches in Public Transit

Bartholdi and Eisenstein (2012) were the first to introduce the concept of self-organization or self-coordination in the field of public transit scheduling. In their approach a vehicle is delayed at a control point by a time proportional to the headway to the trailing vehicle. The authors prove that for the case with a single circular line, this policy ensures that all headways self-equalize over time, regardless of the initial locations of the vehicles. This approach has been extended by Liang et al. (2016) and Zhang and Lo (2018), who consider both the backward headway and the forward headway when computing how long a vehicle should be delayed, resulting in a faster convergence rate. Zhang and Lo (2018) also provide theoretical evidence that the headway variation remains limited under stochastic travel times. However, only single-line systems are considered in these papers.

4.3.2 Multi-Line Control

For multi-line systems, the approach of Argote-Cabanero et al. (2015) is closest to our work. In this study, the authors propose an adaptive control rule for holding, accelerating and decelerating vehicles with the aim to adhere to the schedule as well as possible. However, the possibility to dynamically switch lines after a vehicle reaches a terminal station is not considered. Furthermore, this approach requires the specification of a target schedule and a number of functions and parameters. In contrast, our approach does explicitly allow vehicles to change lines in order to better spread vehicles over the network and only requires the specification of a target headway, making the policy easier to implement. Other papers focusing on multi-line systems, such as Hernández et al. (2015) and Petit et al. (2019), consider centralized optimization based approaches to reduce bus bunching, as opposed to applying a local decision rule.

4.3.3 Rotor-Router Systems

Our policy can be viewed as a generalization of the *rotor-router model*, which was originally introduced in Priezzhev et al. (1996) as the deterministic counterpart of a random walk on a graph. In the random walk on a graph, one or more agents move over a graph at discrete and synchronous steps. The next edge to be traversed by an agent is selected randomly from the set of incident edges of the current node where the agent is located (Lovász, 1993). In the rotor-router model, a node does not send agents visiting it to a random neighbour, but instead selects the incident edges in round-robin fashion. That is, every node in the graph maintains a cyclic ordering of its incident edges and has a pointer indicating the next edge to be traversed by an entering agent. Whenever an agent enters a node, the pointer is advanced to the next edge in the cyclic ordering.

As our policy assigns arriving vehicles at a station to lines in a round-robin fashion, it is similar to the the rotor-router system. On the other hand, in the rotor-router model it takes one time step to traverse an edge, whereas in our case a line can have any positive integer valued travel time. Moreover, our policy sometimes instructs to hold a vehicle at a station to meet the target headway, where agents in the rotor-router model move in every time step. However, we will see that some of the results for the rotor-router model also hold for our policy.

For the rotor-router system, a number of relevant results have been established. When there is only one agent, Priezzhev et al. (1996) proves for the rotor-router mechanism that after a sufficiently long time, the agent gets locked-in in a cycle where every edge is traversed exactly once in both directions. Yanovski et al. (2003) and Bampas et al. (2009) show that the lock-in time is bounded by $2mD$, where m is the number of edges in the graph and D the diameter of the graph. The ability of the system to recover from, for example, edge deletions (corresponding to the removal of lines in a public transit network) is investigated in Bampas et al. (2017). For the case with multiple agents, Wagner et al. (1999) prove that the difference in the number of traversals of two edges cannot grow unbounded. Yanovski et al. (2003) present a stronger bound for the maximum difference between the number of traversals of two edges and also prove that a rotor-router system with multiple agents converges to a periodic motion. Chalopin et al. (2015) provide a further analysis of the limit behavior of the multi-agent rotor router system, and show that unlike the case with one agent the duration of the periodic motion (so the time until the system returns to the same state) can be superpolynomial in the number of edges. Finally, Dereniowski et al. (2016) prove that the time it takes until all edges are traversed with k agents is at least $\log(k)$ times shorter than with one agent.

4.4 Theoretical Analysis

In this section, we analyze the emerging behavior of the system in case all vehicles are scheduled according to the proposed policy. First, we investigate whether the policy serves all lines in a fair or balanced manner, as preferably each line should have approximately the same number of departures. Secondly, we analyze the long run behavior of the system and investigate whether the target headway of every line is met. Thirdly, we provide worst case results on the maximum headway that can occur under the policy and the time it can take before the system reaches a stable state. We conclude this section with a discussion regarding the performance of the policy in case there is no common target headway of all lines.

4.4.1 Balanced Services

We first analyze the extent to which our policy leads to a balanced service of all lines. Ideally, at any point in time, every line should have approximately the same number of departures. Formally, let $f(s \to s')$ denote the number of departures of line $(s \to s') \in \mathcal{L}$ up to time t. The lemmas and theorems that we prove hold for any t. Hence, for readability, we omit the index t. In this section, the goal is to show that the difference between $f(s_1 \to s_1')$ and $f(s_2 \to s_2')$ is bounded for two arbitrary lines $(s_1 \to s_1'), (s_2 \to s_2') \in \mathcal{L}$

As the policy serves lines in a round-robin fashion, for two lines $(s \to s')$ and $(s \to s'')$ originating at the same station, it holds that $|f(s \to s') - f(s \to s'')| \leq 1$. The first part of our analysis only depends on this property of our policy. Because this property is shared with the rotor-router system, we apply a similar analysis as presented by Yanovski et al. (2003).

Let $\mathcal{S}_1, \mathcal{S}_2$ be a partition of the set of all stations. Then, we define $f(\mathcal{S}_1 \to \mathcal{S}_2)$ as the number of times lines starting in $\mathcal{S}_1$ and ending in $\mathcal{S}_2$ have been performed up to some time t (again we omit the index t). We also refer to $f(\mathcal{S}_1 \to \mathcal{S}_2)$ as the flow from $\mathcal{S}_1$ to $\mathcal{S}_2$. As it holds for every of the n vehicles that it is impossible to cross from $\mathcal{S}_1$ to $\mathcal{S}_2$ twice, without crossing back from $\mathcal{S}_2$ to $\mathcal{S}_1$, we can make the following observation (first made by Wagner et al. (1999)).

Observation 4.1. *For a partition $\mathcal{S}_1, \mathcal{S}_2$ of the set of stations, it holds that $f(\mathcal{S}_1 \to \mathcal{S}_2) - f(\mathcal{S}_2 \to \mathcal{S}_1) \leq n$.*

Using Observation 4.1, it is possible to prove Lemma 4.2, which gives an upper bound on the difference between the number of times a line and its reverse line are performed.

Lemma 4.2. *For every line* $(s \to s') \in \mathcal{L}$*, it holds that* $|f(s \to s') - f(s' \to s)| \leq n$.

Proof. We define $f(s) := \min_{(s \to s') \in \delta^+(s)} f(s \to s')$ for a station $s \in \mathcal{S}$, denoting the minimum number of departures for any line leaving s. As the lines are always served in a round-robin fashion, we have that $0 \leq f(s \to s') - f(s) \leq 1$.

Now, suppose that the lemma is not true, such that there exists a pair of opposite lines $(s \to s')$ and $(s' \to s)$ with $f(s \to s') = j$ and $f(s' \to s) \leq j - n - 1$. By definition, it holds that $f(s) \geq j - 1$ and $f(s') \leq j - n - 1$. Consider the partition $\mathcal{S}_1, \mathcal{S}_2$ where $\mathcal{S}_1 = \{s \in \mathcal{S} : f(s) \geq j - n\}$ and $\mathcal{S}_2 = \{s \in \mathcal{S} : f(s) \leq j - n - 1\}$. We have that $s \in \mathcal{S}_1$ and $s' \in \mathcal{S}_2$. Let the number of lines crossing from $\mathcal{S}_1$ to $\mathcal{S}_2$ be m. As the flow from s to s' is j and the flow over all other lines from $\mathcal{S}_1$ to $\mathcal{S}_2$ is at least $j - n$, we find that

$$f(\mathcal{S}_1 \to \mathcal{S}_2) \geq (m - 1)(j - n) + j.$$

Similarly, the flow from $\mathcal{S}_2$ to $\mathcal{S}_1$ is at most

$$f(\mathcal{S}_2 \to \mathcal{S}_1) \leq (m - 1)(j - n) + j - n - 1.$$

Therefore, it follows that

$$f(\mathcal{S}_1 \to \mathcal{S}_2) - f(\mathcal{S}_2 \to \mathcal{S}_1) \geq n + 1.$$

As this contradicts Observation 4.1, the assumption that the lemma is not true must be wrong. □

We now present a theorem that bounds the difference between $f(s_1 \to s_1')$ and $f(s_2 \to s_2')$ for two arbitrary lines, which implies that the number of times two lines are performed cannot differ too much, at any point in time. To do so, let $\text{lines}(s_1, s_2)$ denote the minimum number of lines that has to be traversed on a path from s_1 to s_2. For example, if s_1 and s_2 are connected by a line it holds that $\text{lines}(s_1, s_2) = 1$ and if s_1 and s_2 are not directly connected, but both are connected to some other station s_3, it holds that $\text{lines}(s_1, s_2) = 2$.

Theorem 4.3. *For two lines* $(s_1 \to s_1'), (s_2 \to s_2') \in \mathcal{L}$*, it holds at any time that* $|f(s_1 \to s_1') - f(s_2 \to s_2')| \leq (\text{lines}(s_1', s_2) + 1)(n + 1)$.

Proof. First, we prove a bound on the difference between the number of times two consecutive lines are performed. Thereafter, we consider a shortest path between s_1' and s_2 and iteratively apply this bound to prove the theorem.

By Lemma 4.2 it holds that $f(s \to s') \leq f(s' \to s) + n$ and by definition of the policy it holds that $f(s' \to s) \leq f(s' \to s'') + 1$. Hence, for two consecutive lines we find that $f(s \to s') \leq f(s' \to s'') + n + 1$.

Next, let $s_1' = s_a, s_b, ..., s_p = s_2$ denote a shortest path from s_1' to s_2. It holds that

$$\begin{aligned} f(s_1 \to s_1') &\leq f(s_a \to s_b) + n + 1 \\ &\leq f(s_b \to s_c) + 2(n + 1) \\ &\vdots \\ &\leq f(s_0 \to s_p) + \text{lines}(s_1', s_2)(n + 1) \\ &\leq f(s_2 \to s_2') + (\text{lines}(s_1', s_2) + 1)(n + 1). \end{aligned}$$

Hence, $f(s_1 \to s_1') - f(s_2 \to s_2') \leq (\text{lines}(s_1', s_2) + 1)(n + 1)$. The theorem follows by symmetry. □

The desirable property of the bound proven in Theorem 4.3 is that it does not depend on t. Therefore, it holds that the difference between the number of times two lines are performed cannot grow unbounded. In what follows, we use this observation to characterize the long run behavior of the system.

4.4.2 Limit Behavior

In this section, we analyze the emerging properties of the system in the long run. The first result states that after some time it is guaranteed that the system enters a periodic motion (i.e. starts to cycle) and that every line is performed the same number of times in one cycle. The proof is an adaption of the proof by Yanovski et al. (2003) of the same property for the rotor-router system.

Lemma 4.4. *After a certain finite time, the system enters a periodic motion. In every cycle, every line is performed the same number of times.*

Proof. To prove the first part of the lemma, it suffices that the system can only be in a finite number of states. Since the policy is deterministic, it then directly follows that the system must eventually return to the same state, at which point the system enters a periodic motion. To see why the system can only be in a finite number of states, note that by the integrality of the travel times and the target headway H, vehicles always depart at integer time points. Hence, it suffices to consider the system only at integer time points. The state of the system at an integer time point can be represented by all locations of the vehicles and the time since the latest departure of all lines. According to Theorem 4.3, the number of times two lines are performed cannot differ too much. This implies that the time since the latest departure of a line cannot be unbounded. It then follows that the number of possible states is finite.

To prove the second part of the lemma, note that if the number of times two lines are performed during a cycle would be different, over time the difference would grow without bound, contradicting Theorem 4.3. □

It follows from the proof of Lemma 4.4 that we can represent the state of the system at time t using some state vector $\mathcal{V}_t$. Following Chalopin et al. (2015), we call a state $\mathcal{V}_t$ *stable* if there exists $t' > t$ such that $\mathcal{V}_{t'} = \mathcal{V}_t$. Equivalently, we say that the system has stabilized once it has entered the periodic motion. By Lemma 4.4, $\mathcal{V}_t$ will always be stable for large enough t. The *stabilization time*, denoted as T_{stable}, is the smallest value such that $\mathcal{V}_{T_{\text{stable}}}$ is stable. Furthermore, the periodicity, denoted as T_{period}, is the smallest value such that $\mathcal{V}_{T_{\text{stable}}+T_{\text{period}}} = \mathcal{V}_{T_{\text{stable}}}$.

At this point, the role of the target headway in the policy also comes into play, which is not present in the rotor-router model. Hence, all theoretical results that follow require novel analysis. In the remainder of this subsection, we analyze the properties of the system once it reaches a stable state in more detail. In Section 4.4.3, we analyze how large the stabilization time can be in the worst case.

To provide more insight into the emerging behavior of the system, we first consider the number of idle vehicles at stations. Let $\text{arr}_s(a, b)$ and $\text{dep}_s(a, b)$ denote the number of arrivals and departures at s in the half-open interval $[a, b)$ respectively. Then, the number of idle vehicles at station s at time t, denoted by $i_s(t)$, satisfies

$$i_s(t) = i_s(0) + \text{arr}_s(0, t) - \text{dep}_s(0, t).$$

This brings us to the following lemma:

Lemma 4.5. *For the number of idle vehicles at a station, it holds that* $i_s(t) \geq i_s(t+H)$*.*

Proof. Assume $i_s(t) < i_s(t+H)$. Then, it must hold that $\text{dep}_s(t, t+H) < \text{arr}_s(t, t+H)$. As every line has at most one departure per H time units, the number of arrivals at s during H time units, is at most $|\delta^-(s)|$ (the number of lines that terminates at s). By the symmetry of the network $|\delta^-(s)| = |\delta^+(s)|$, so we find that $\text{dep}_s(t, t+H) < |\delta^+(s)|$. As such, there exists a line l without a departure in the interval $[t, t+H)$. This implies that $i_s(t+H) = 0$, as otherwise line l would have had a departure in this interval. As $0 = i_s(t+H) > i_s(t) \geq 0$, we reach a contradiction. The conclusion is that $i_s(t) \geq i_s(t+H)$. □

Next, we analyze a global performance indicator, the *utilization*. Let $\gamma_s(a, b)$ denote the total idle time of vehicles at station s in the interval $[a, b)$. The utilization, denoted by $\text{util}(a, b)$, represents the average proportion of time that the vehicles are driving in the interval $[a, b)$:

$$\text{util}(a, b) := 1 - \frac{\sum_{s \in \mathcal{S}} \gamma_s(a, b)}{(b-a)n}.$$

By Lemma 4.5, the number of idle vehicles cannot increase over time. Therefore, it holds that the utilization cannot decrease over time. The next lemma formalizes this statement.

Lemma 4.6. *For the utilization, it holds that* $\text{util}(t, t+H) \leq \text{util}(t+H, t+2H)$*. Moreover, for* $t' \geq T_{\text{stable}}$ *it holds that*

$$\text{util}(t', t'+H) = \text{util}(t'+iH, t'+(i+1)H) \textit{ for any } i \in \mathbb{Z}^+.$$

Proof. Observe that $\gamma_s(a, b) = \int_a^b i_s(t)dt$. By applying Lemma 4.5 we find

$$\begin{aligned}
\gamma_s(t, t+H) &= \int_t^{t+H} i_s(x)dx \\
&\geq \int_t^{t+H} i_s(x+H)dx \\
&= \int_{t+H}^{t+2H} i_s(x)dx \\
&= \gamma_s(t+H, t+2H).
\end{aligned}$$

Therefore, it holds that

$$\begin{aligned}\text{util}(t, t+H) &= 1 - \frac{\sum_{s\in\mathcal{S}} \gamma_s(t, t+H)}{Hn}\\ &\leq 1 - \frac{\sum_{s\in\mathcal{S}} \gamma_s(t+H, t+2H)}{Hn}\\ &= \text{util}(t+H, t+2H).\end{aligned}$$

For the second part of the lemma, note that by the periodicity of the system, we have that for $t' > T_{\text{stable}}$ $\text{util}(t', t'+H) = \text{util}(t' + iT_{\text{period}}, t' + iT_{\text{period}} + H)$ for any $i \in \mathbb{Z}^+$. Since $\text{util}(t, t+H) \leq \text{util}(t+H, t+2H)$, it follows for $t' > T_{\text{stable}}$ that $\text{util}(t', t'+H) = \text{util}(t'+iH, t'+(i+1)H)$ for any $i \in \mathbb{Z}^+$.

□

From the above lemma, it follows that once the system reaches a stable state, the utilization is constant over consecutive intervals of duration H, even though T_{period}, the duration of the periodic motion, can in general be strictly larger than H. We formally define this limit value of the utilization as $u := \text{util}(T_{\text{stable}}, T_{\text{stable}} + H)$, to which we refer as the *stable utilization*. Note that the policy ensures that at all lines are performed at least once in one cycle of the periodic motion, such that $u > 0$.

Before we state the main theorem, recall that n^* is a lower bound on the number of vehicles required to meet the target headways. Theorem 4.7 shows that the behavior of the system depends on whether $n < n^*$ or $n \geq n^*$.

Theorem 4.7. *If $n < n^*$, then the stable utilization u equals 1 and the average headway of all lines during the periodic motion equals $\frac{n^*}{n}H$. Otherwise, the headways of all lines during the periodic motion equal H for all lines and the stable utilization equals $\frac{n^*}{n}$.*

Proof. As we have shown that the utilization converges to a certain stable utilization $0 < u \leq 1$, we can distinguish the following two cases:

Case I: $0 < u < 1$. This implies that there exists at least one station where there is strictly positive idle time during the periodic motion. It must be that the lines originating from this station are performed once per H time units, as a vehicle only waits if it is in time to meet the target headway of its next line. As all lines are performed the same number of times in the periodic motion by Lemma 4.4, it follows that $T_{\text{period}} = H$ and every line is performed exactly once per H time units in both directions. By definition, this implies that $n \geq n^*$. As every line is operated once per H time units, the utilization converges to

$$u = \frac{\sum_{l \in \mathcal{L}} t_l}{nH} = \frac{n^*}{n}.$$

Case II: $u = 1$. As every line can be operated at most once every H time units, this implies that $n \leq n^*$. Moreover, it follows from Lemma 4.4 that the system enters a periodic motion in which all vehicles have no idle time and all lines are performed the same number of times. Let g denote the number of times each line is performed during a cycle. As all vehicles are running all the time, it holds that

$$nT_{\text{period}} = g \sum_{l \in \mathcal{L}} t_l = gn^* H.$$

Consequently, the average headway of all lines, which we denote as $\bar{H}$, equals

$$\bar{H} = \frac{T_{\text{period}}}{g} = \frac{n*}{n} H.$$

□

The above theorem provides a concise characterization of the behavior of the system under the proposed policy. The result can be seen to be optimal in some sense. If the number of vehicles is large enough to meet the target headways using centralized control, our decentralized policy is also able to meet the target headways. In case there are not sufficient vehicles to meet the target headways, every vehicle is used all the time and every line has the same average headway, equal to the smallest headway possible under centralized scheduling. Moreover, if $n \geq n^* + 1$, there is some slack in the system, such that if a vehicle breaks down, the headways of all lines again converge to H. In the other case, there is no slack in the system and every breakdown of a vehicle leads to an increase in headways, and therefore to a reduction in passenger service.

4.4.3 Worst Case Analysis

In this part, we provide worst case results of the headway deviation in case $n < n^*$ and on the time it takes to reach a stable state.

Worst Case Headway Deviation

In contrast with the results Bartholdi and Eisenstein (2012) and Zhang and Lo (2018) obtained for the single-line case, in case $n < n^*$ (so there are not enough vehicles available), our policy leads to convergence of the *average headways* of all lines, but not necessarily to convergence of the headways themselves. A natural question to ask is how large the maximum headway can become in the worst case. Theorem 4.8 shows that the headway cannot be larger than $H + (n^* - n)H$, such that the excess headway is never larger than $(n^* - n)H$.

Theorem 4.8. *If $n < n^*$, once the system is in a stable state, all headways are at most $H + (n^* - n)H$.*

Proof. For this proof it is convenient to think of every line l as having length t_l and think of every vehicle as a snake having length H and moving 1 unit distance per unit time (such that it takes t_l time units to traverse a line). As $n < n^*$, it holds according to Theorem 4.7 that the system converges to a periodic motion where the snakes are constantly moving. Furthermore, the policy ensures that consecutive departures of the same line are always separated by at least H time units, such that two snakes, despite having length H, cannot occupy the same part of a line. Therefore, once the system has stabilized, the snakes cover a part of the network of length nH. The part of the network that is not covered by any of the snakes then has length, $\sum_{l \in \mathcal{L}} t_l - nH = n^*H - nH = (n^* - n)H$. Thus, whenever a snake starts traversing a line, the distance between the front of the snake and the tail of the preceding snake on the line is at most $(n^* - n)H$. As the length of every snake is H, it follows that the distance between the fronts of two vehicles is, at any time, at most $H + (n^* - n)H$. Hence, the time between two consecutive departures of the same line is at most $H + (n^* - n)H$. □

This theorem has a nice interpretation, as it shows that if there is only a small shortage of vehicles, the headways cannot become very large. As long as the discrepancy between n and n^* is not too big, the target headways are met reasonably well. For example, if $n = n^* - 1$, the maximum headway that can occur is $2H$.

Stabilization Time

In this part, we derive worst case bounds on the stabilization time T_{stable}. We investigate the time until stabilization for two special cases. In both cases, the stabilization time depends on the unweighted diameter of the network, a graph parameter which only depends on the topology of the network. The diameter is denoted as D and is defined as follows (recall that $\text{lines}(s_1, s_2)$ equals the minimum number of lines that has to be traversed on a path from s_1 to s_2)

$$D = \max_{s_1, s_2} \text{lines}(s_1, s_2), \tag{4.1}$$

so D can be seen as the longest shortest path in the network. First, we analyze the case where $n = n^* = 1$, so a single vehicle suffices for meeting the target headway.

Theorem 4.9. *If $n = n^* = 1$, it holds that $T_{\text{stable}} \leq DH$.*

Proof. As there is only one vehicle, the system is stabilized if the vehicle continuously performs an Euler tour every H time units. Bampas et al. (2009) shows that for the rotor-router model with a single agent, an Euler tour is established in "phases" and that in the worst case, D phases are required. This result directly extends to our setting. In every phase, the vehicle performs a tour starting and ending at s_0, the initial location of the vehicle. A phase ends when the vehicle returns to s_0 and all the lines originating at s_0 have been traversed during that phase. Furthermore, in the worst case, the cyclic order of the outgoing lines at every s is such that in phase i, station s is visited if and only if $\text{lines}(s_0, s) \leq i$. Therefore, after round D the vehicle will have entered the periodic motion and continuously perform an Euler tour. Furthermore, since $n^* = 1$ every closed tour over the network takes at most H time units, which implies that the duration of every round is H. It follows that the system stabilizes in the worst case at time DH. □

Next, we analyze the case where $t_l = H$ for all $l \in \mathcal{L}$ and $n = n^* = |\mathcal{L}|$. Since all travel times are equal to the target headway, we can analyze the system in iterations of duration H and define $i_s(m)$ as the number of vehicles located at station s at the end of iteration m. The system has stabilized if and only if $i_s(m) = \deg(s)$ for every $s \in \mathcal{S}$, where $\deg(s)$ denotes the degree of station s in the network (the number of lines in the set $\delta^+(s)$). This motivates the following definition:

Definition 4.10. We define $C_s(m) = i_s(m) - \deg(s)$ as the *charge* of station s after iteration m. The station s is called *positively charged* if $C_s(m) > 0$ and *negatively charged* if $C_s(m) < 0$. Otherwise, the station is called *neutral.*

We can observe that the total charge over the network equals zero:

$$\sum_{s \in \mathcal{S}} C_s(m) = \sum_{s \in \mathcal{S}} (i_s(m) - \deg(s)) = n - \sum_{s \in \mathcal{S}} \deg(s) = n - n^* = 0.$$

According to the policy, the number of vehicles leaving station s in iteration $m+1$ equals $\min\{i_s(m), \deg(s)\}$. As the number of vehicles entering station s in an iteration is at most $\deg(s)$, it follows that the charge of a neutral or positively charged station can never increase. Hence, a station can change from being positively charged to neutral, but not vice versa. On the other hand, until the system stabilizes, it is possible that negatively charged stations become neutral and vice versa.

We define the potential function $\Phi(m) = \sum_{s \in \mathcal{S}: C_s(m) > 0} C_s(m)$, equal to the sum of the positive charges. As the charge of positively stations can only decrease and neutral stations cannot become positively charged, it follows directly $\Phi(m) \geq \Phi(m+1)$. If $m \geq T_{\text{stable}}/H$, it holds that $\Phi(m) = 0$. In order to bound T_{stable}, we use the following result from the rotor-router system

Lemma 4.11. *For a rotor-router system with $k > 1$ agents, the cover time (the time until all edges have been visited at least once) on a graph with diameter D and m edges is at most $O\left(\frac{mD}{\log k}\right)$. If there is only 1 agent, the cover time is $O(mD)$.*

Proof. See Dereniowski et al. (2016). □

Theorem 4.12. *If $t_l = H$ for all $l \in \mathcal{L}$ and $n = n^* = |\mathcal{L}|$, it holds that $T_{stable} = O\left(|\mathcal{L}|^2 DH\right)$.*

Proof. Clearly, $\Phi(m)$ is integer and $0 \leq \Phi(m) \leq n-1$. To bound the number of iterations the potential function can stay constant, we use the concept of *anti-vehicles*: whenever a line is **not** traversed by a vehicle in some iteration, it is traversed by an anti-vehicle. Thus, in case station s is negatively charged after iteration m, there are $i_s(m)$ regular vehicles and $-C_s(m)$ anti-vehicles leaving s in iteration $m+1$. Moreover, if $\Phi(m) = \Phi(m+1)$, it holds that in case s is negatively charged after iteration $m+1$, the absolute value of the charge equals the number of anti-vehicles entering s in iteration m. Hence, the anti-vehicles can be seen as carriers of the negative charge over the network.

Suppose that $\Phi(m) = f > 0$. Then, there are f anti-vehicles in the network after iteration m. We are interested in how long it takes until one of the anti-vehicles arrives at a positively charged station, as such an event reduces the potential function. As in any iteration the anti-vehicles traverse the lines that are not traversed by regular vehicles, it can be seen that that the anti-vehicles move according to the same policy as the regular vehicles, *but with the cyclic order of the lines reversed*. Moreover, since anti-vehicles move in every iteration (otherwise the number of anti-vehicles at a station would be larger than the degree), this system is equivalent to a rotor-router system. Therefore, the number of iterations until one of the anti-vehicles hits a positively charged station is at most the cover time of a rotor-router system with f agents. Applying Lemma 4.11 and using that the potential can decrease at most $n - 1 = |\mathcal{L}| - 1$ times and that every iteration takes H time units, it follows that

$$T_{\text{stable}} = O(|\mathcal{L}|DH) + \sum_{f=2}^{|\mathcal{L}|-1} O\left(\frac{|\mathcal{L}|D}{\log f}H\right) = O\left(|\mathcal{L}|^2 DH\right).$$

□

The results in Theorem 4.9 and 4.12 illustrate that the stabilization time depends on the diameter and number of lines of the network and the target headway. For high-frequency networks that are highly connected, for example urban transit systems, stabilization occurs rapidly. For large elongated networks operated at lower frequencies, for example inter-regional transit systems, stabilization is established more slowly.

4.4.4 Different Target Headways Among Lines

In order to apply our policy to instances with different target headways, it needs to be slightly altered. Instead of maintaining a fixed cyclical order of the lines, an incoming vehicle should be assigned to the line whose target departure time is minimal. Intuitively, this line needs a departure most urgently. Moreover, whenever there is a departure of line l' at time t', the target departure time should now be updated according to the formula $\tau_{l'} \leftarrow t' + h_{l'}$, where h_l denotes the target headway of line l.

The presented theoretical results cannot be extended to settings where there is no common target headway. There are instances with different headways where the system fails to stabilize under our policy, despite enough vehicles being available. For such instances, stabilization can only be attained if the lines are performed in a specific order. However, if our policy initially performs the lines in the wrong order, it is unable to modify the order, such that the system remains in an unstable state indefinitely.

Furthermore, it holds in general that, unless P=NP, there does not exist a policy that requires no more than the optimal number of vehicles and is guaranteed to stabilize within polynomial time. In Chapter 3, we proved that the problem of deciding whether a line plan with arbitrary frequencies can be operated with a certain number of vehicles (i.e. whether there exists a timetable that meets all target headways) is NP-complete in the strong sense. Hence, if a policy would have the property that the headways of all lines converge to the target headways if the number of available vehicles is at least the minimum required number under centralized scheduling, an NP-complete problem would be solved by simulating this policy.

Despite these insights, it is still possible to apply our policy to systems with different target headways. A first possibility is to decompose the network into sub-networks where there is a common target headway. Secondly, one could still choose to apply the policy and accept that there are no theoretical performance guarantees. We assess the performance of the latter approach numerically in the next section.

4.5 Numerical Experiments

In this section, we describe the results of a series of experiments that illustrate the practical performance of the proposed policy. First, we analyze how the time it takes to reach a stable state grows if the size of the network increases. Next, we investigate how long it takes to re-stabilize after one of the vehicles breaks down. Then, we we test the performance of the policy in situations where the assumptions of the theoretical analysis of the previous section are not met. Specifically, we analyze the behavior of the system in case the lines do not have a common target headway and in case the travel times are not fixed but stochastic. Finally, we assess the performance of the policy on three real-world transit systems.

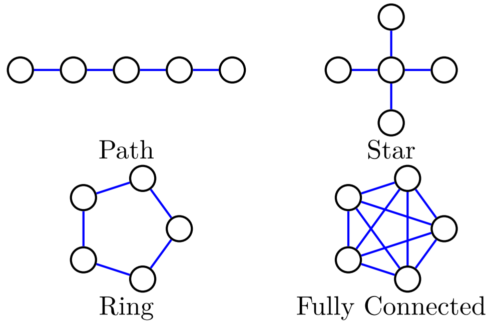


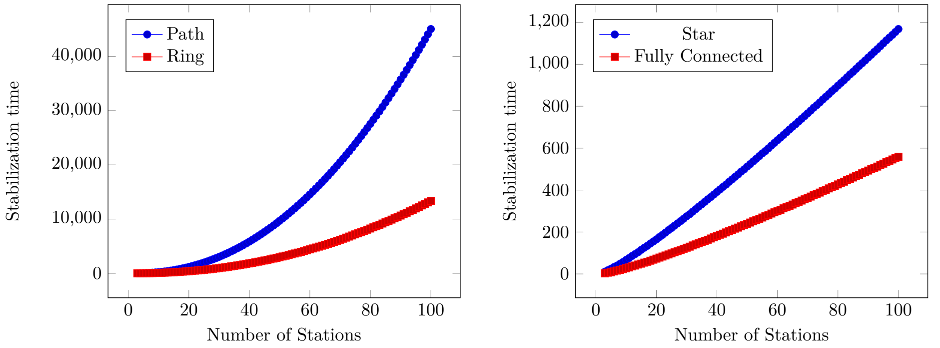
Figure 4.2: Different network topologies used in the numerical experiments.

4.5.1 Stabilization Time

We assess the time to reach a stable state for four types of network topologies: path, ring, star and fully connected. The differences between these types of networks are illustrated in Figure 4.2. In the first two experiments, we set the travel time of each edge equal to one time unit and set $H = 1$ and $n = n^*$.

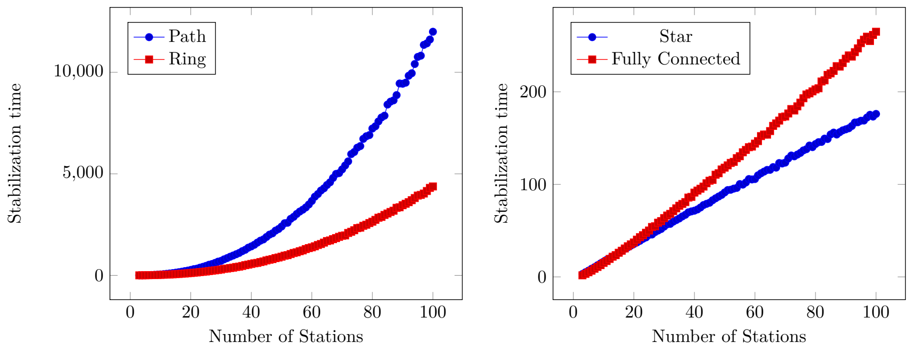
In Figures 4.3a-4.3b, it is shown how the stabilization time grows with the size of the network, if all vehicles start from an unbalanced starting position. That is, we start with all vehicles at a single station. For the path network and the star network, we start with all vehicles at one of the outer stations. To minimize the rate at which the vehicles are spread out over the networks, for each station s, l^s_{next} is initialized such that the first time a vehicle enters s, the vehicle is sent back over the reverse line it came from. When comparing the stabilization time for a fixed number of stations, we find that the network topologies with the largest diameter also have the largest stabilization time, which we also expected based on the worst case results in Section 4.4.3. Moreover, the stabilization time grows at a faster rate for the path and ring network compared to the star and fully connected network, which is likely caused by the fact that the diameters of the latter two networks is constant in the number of stations, whereas the diameter of the former two networks increases linearly in the number of stations.

In Figures 4.4a-4.4b, it is shown how the time until stabilization grows with the size of the network, if the stating configuration is randomly generated. Here, we take the average over 2,500 runs. The required time is much shorter compared to the unbalanced starting situation. Interestingly, the fully connected network takes more time on average to stabilize than the star network, whereas in the previous experiment the star network required more time. Most likely, this is caused by the fact that if all vehicles often attend the same station, their departure times are more quickly coordinated. However, in the previous experiment all vehicles started at the same outer station of the star network, such that it took a long time before all vehicles had departed from the starting station.



(a) Path and Ring (b) Star and Fully Connected

Figure 4.3: Stabilization time, starting from an unbalanced starting configuration.



(a) Path and Ring (b) Star and Fully Connected

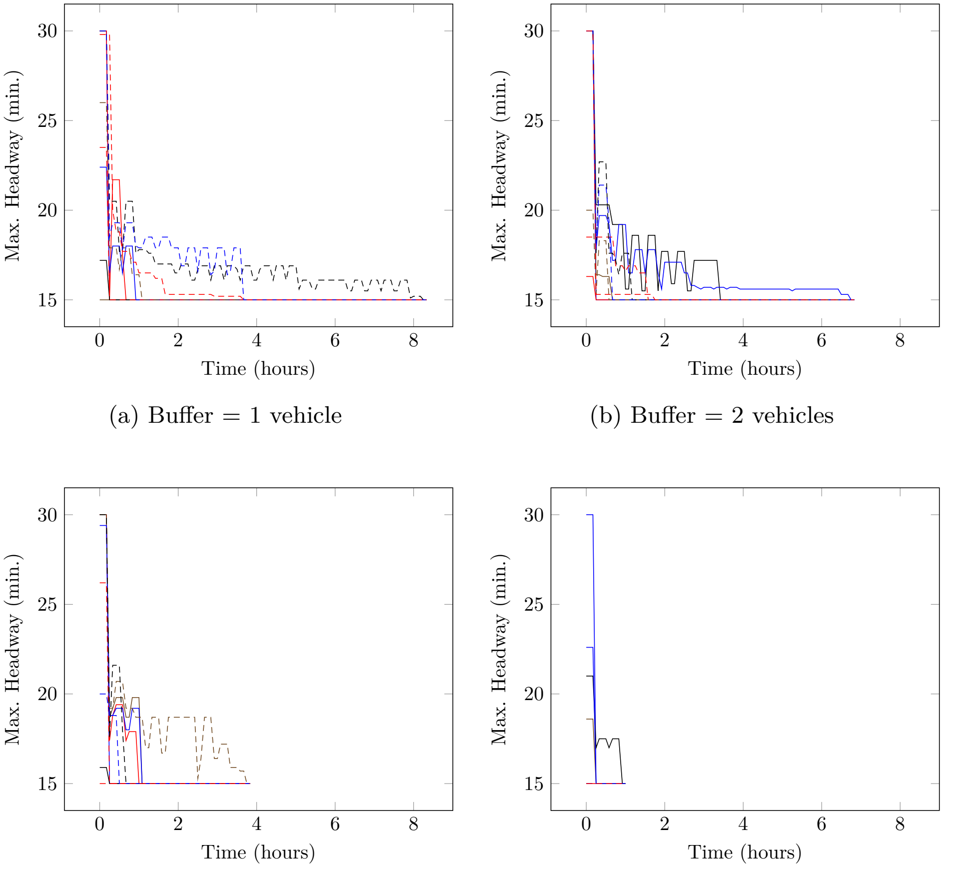
Figure 4.4: Stabilization time, starting from a random configuration, averaged over 2500 samples.

4.5.2 Re-Stabilizing after a Vehicle Breakdown

To get a better sense of the performance in practice, we perform a third experiment, where we start in a stable state (i.e. a feasible timetable). Then, we let one of the vehicles break down and analyze how long it takes to re-stabilize. We refer to the number of vehicles in the system above the minimum number of required vehicles to reach a stable state as the *buffer*. Provided that there is a buffer of at least one vehicle, we know that the system will always bounce back to a stable state after the breakdown. Note that having a buffer in the number of vehicles can be seen as the 'self-organizing' analog of the commonly used method in schedule-based approaches to include time supplements in a timetable: it increases the robustness, at the cost of a decrease in efficiency.

We perform this experiment on a star network with five lines with a target headway of 15 minutes and travel times uniformly drawn between 10 and 30 minutes. We start this experiment from a random stable state, which is achieved by having the system converge to a stable state from a random starting configuration.

In Figures 4.5a-d, the results of this experiment are visualized, for different sizes of the buffer and with ten randomly generated networks for each buffer size. The horizontal axis depicts the time since the breakdown and the vertical axis the current maximum headway in the network. As expected, the maximum headway in the system can be quite large right after the vehicle breakdown. However, the impact of the breakdown dies out rather quickly. Even with only a single vehicle as a buffer, the maximum headway in the system reduces to less than 20 minutes within the first hour. On the other hand, there can be a tailing off effect, as for some of the scenarios we observe it takes a long time before all headways really have converged to 15 minutes. However, in practice the difference between a headway of 15 minutes and a headway that is up to 45 seconds longer is not very noticeable to passengers. Moreover, regardless of whether one uses a self-organizing approach or a schedule-based approach, the breakdown of a vehicle causes a decrease in capacity that needs to be absorbed in one way or another. Spreading out the "pain" caused by the disruption over a long time can be a preferable solution in some scenarios. We also observe that the maximum headway converges to 15 minutes much faster when there is a larger buffer in the number of vehicles.



(a) Buffer = 1 vehicle (b) Buffer = 2 vehicles

(c) Buffer = 3 vehicles (d) Buffer = 5 vehicles

Figure 4.5: Impact of vehicle breakdown at $t = 0$ given vehicle availability. Each figure shows how the maximum headway in the network evolves over time for ten scenarios, with each line representing a scenario. The target headway equals 15 minutes.

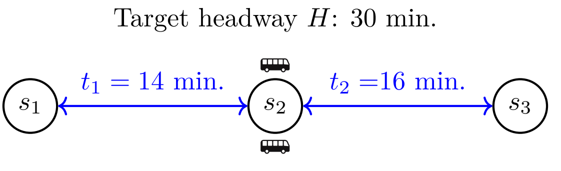


Figure 4.6: Example with a long stabilization time. Travel times are symmetric.

The cause of the tailing off effect visible in the previous experiment can be understood using the simple example depicted in Figure 4.6. There are two lines and two buses available at the station connection the lines. It holds that $n^* = (2 \cdot 14 + 2 \cdot 16)/30 = 60/30 = 2$, so it is guaranteed that with two buses the system will eventually stabilize.

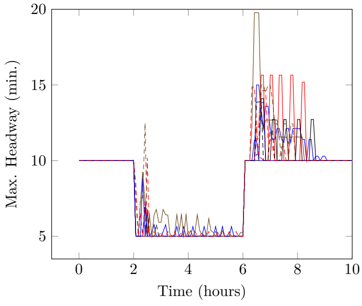
Let us suppose that at the initialization ($t = 0$), the first bus at s_2 is sent towards s_1 and the second bus towards s_3. After 28 minutes ($t = 28$), the first bus will have returned to s_2, where it is instructed depart towards s_1 again at $t = 30$. The second bus arrives at s_2 at $t = 32$ and is instructed to depart immediately towards s_3 since the current headway of line $(s_2 \to s_3)$ is already 32 minutes. Continuing to apply the policy, the first departures for line $(s_2 \to s_1)$ occur at 0, 30, 60, 90 et cetera, while the first departure times for line $(s_2 \to s_3)$ occur at 0, 32, 64 and 96. The headway of the line $(s_2 \to s_3)$ remains 32 until $t = 448$. From that point on, the headways of both lines are always 30 minutes, so the system has stabilized after 448 minutes.

Before the system has stabilized, one bus only performs the line between s_1 and s_2, with a waiting time of 2 minutes before every departure at s_2, while the other bus only performs the line between s_2 and s_3. At $t = 448$, the bus arriving at s_2 from s_1 is assigned to line $(s_2 \to s_3)$, because that line has been performed 13 times while line $(s_2 \to s_1)$ 14 times. After this time, the buses alternate between the two lines, which is more efficient as there is no waiting, causing the system to stabilize.

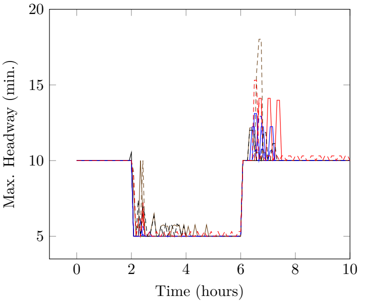
This example illustrates that if the system is not yet in a stable state, there must be at least one vehicle running an inefficient circulation, so a circulation with strictly positive waiting time. Furthermore, we can observe that there is a direct relation between the waiting time before every departure of line $(s_2 \to s_1)$, the deviation from the target headway for line $(s_2 \to s_3)$ and the stabilization time. By working out the policy for different values of t_2, we find that the stabilization time is approximately inversely proportional with $2t_2 - H$, the headway deviation of line $(s_2 \to s_3)$ before stabilization: if the headway deviation is large, the stabilization time is small, but if the headway deviation is small, the stabilization time can blow up, leading to a long tailing off effect. For example, if the travel times of the lines were 13 and 17 respectively, instead of 14 and 16, the waiting time of the vehicle performing line $(s_2 \to s_1)$ would be 4 minutes instead of 2 minutes, the headway deviation of line $(s_2 \to s_3)$ would be 4 minutes instead of 2 minutes and the system would stabilize after 236 minutes instead of 448 minutes. In other words, large inefficiencies and headway deviations are quickly flushed out, while small inefficiencies and headway deviations can be more persistent.

4.5.3 Time-Varying Target Headways

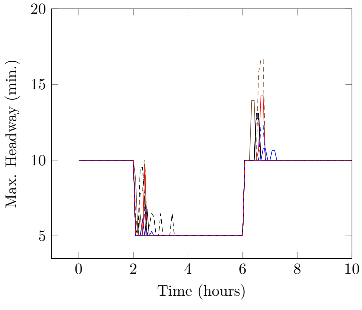
We next test the policy's abilities to accommodate for time-varying headways. To do so, we perform an experiment that resembles a peak period that requires smaller headways to transport all passengers. In the first 2 hours of the simulation, the target headway equals 10 minutes. The next 4 hours represent the peak period with a target headway of 5 minutes. The final 4 hours are again off-peak hours with a target headway of 10 minutes. During the peak-hour, also an additional number of vehicles is available, keeping the buffer constant. These vehicles are again removed from the network when the peak hour ends. We perform this experiment on a star network with three lines and travel times uniformly drawn between 10 and 30 minutes. As in the previous experiment, we start from a random stable state.



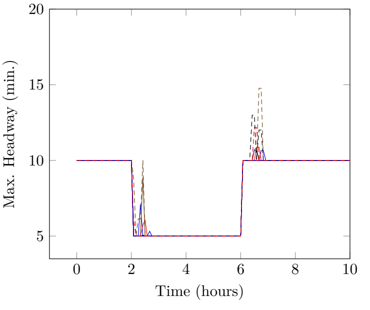
(a) Buffer = 0 vehicles



(b) Buffer = 1 vehicle



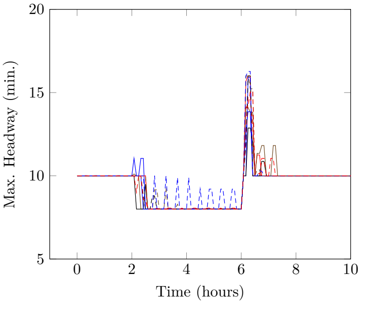
(c) Buffer = 3 vehicles



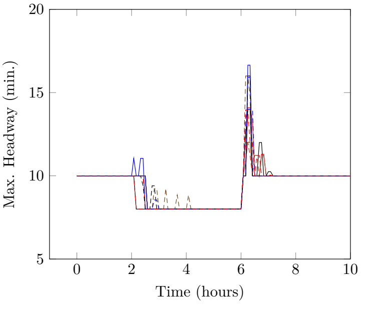
(d) Buffer = 5 vehicles

Figure 4.7: Impact of time-varying headways given vehicle availability. Each figure shows how the maximum headway in the network evolves over time for ten scenarios, with each line representing a scenario. The target headway equals 10 minutes in the first 2 and last 4 hours and equals 5 minutes in hours 2-6.

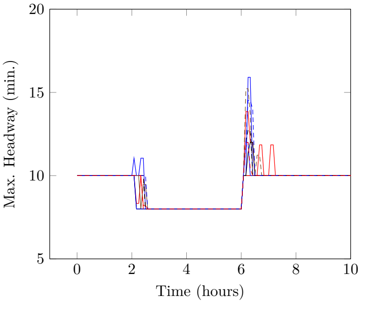
The results of the experiment are presented in Figures 4.7 for different sizes of the buffer and with ten randomly generated travel time instances for each buffer size. The horizontal axis depicts the time since the start of the experiment and the vertical axis the maximum headway across the lines in the network at that particular time. The maximum headway is 10 minutes during the first 2 hours of the experiment as we initialize the system from a stable state. We observe that decreasing the target headway after 2 hours and increasing the target headway again after 6 hours acts as a shock that is eventually absorbed by the system. For all buffer sizes, it is apparent that the increase in the target headway after 6 hours causes larger deviations from the target headway than the decrease in target headway after 2 hours. If there is no buffer in the system, in some of the scenarios the headway can become 15 up to 20 minutes. This indicates that for small buffer sizes, reducing the number of vehicles in an uncoordinated manner may lead to large gaps between vehicles. On the other hand, if a larger number of vehicles is available, both increasing and decreasing the headway is far less problematic, as the shocks can be seen to quickly die out.



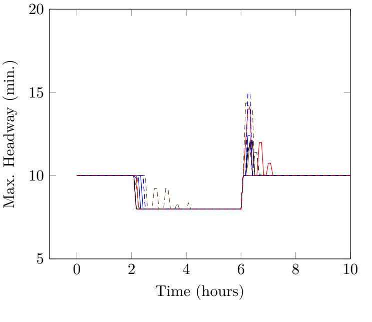
(a) Buffer = 0 vehicles



(b) Buffer = 1 vehicle



(c) Buffer = 3 vehicles



(d) Buffer = 5 vehicles

Figure 4.8: Impact of time-varying headways given vehicle availability. The target headway equals 10 minutes in the first 2 and last 4 hours and equals 8 minutes in hours 2-6.

In Figure 4.8, the results of the same experiment are shown, but with a peak target headway of 8 minutes instead of 5 minutes. Comparing with Figure 4.7, we observe that the smaller difference between the peak and off-peak target headway lessens the impact of the change in target headway. However, there are still deviations from the target headway present when the target headway changes.

4.5.4 Different Target Headways Among Lines

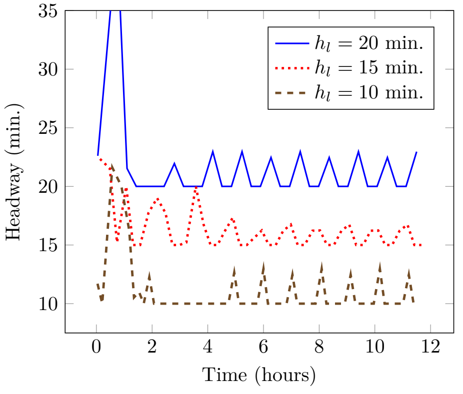
The performance under different target headways is the subject of the next experiment. We conduct this experiment on a star network with three lines, with target headways of 10, 15 and 20 minutes, respectively. The travel time for each line is uniformly drawn between 10 and 30 minutes.

In Figures 4.9a-4.9b, the headways of the three lines are plotted over time for a randomly generated instance. Only the headways of the lines from the central station of the star network to the outer stations are included here. If the number of vehicles is equal to the minimum number required to meet the target headways, we observe that the system does not converge to a stable state where the target headways are always met. Instead, the system converges to a periodic motion where there are deviations from the target headways. In this periodic motion, the headways of the line with a target headway of 10 minutes vary between 10 and 13 minutes and for the line with a target headway of 20 minutes they vary between 20 and 23 minutes. This is in line with our expectations, since, as discussed in Section 4.4.4, there are no guarantees that the system stabilizes if there is no common headway. Assuming uniform passenger arrivals, the expected waiting times without a buffer are 5.2 minutes for the line with a target headway of 10 minutes, 8.0 minutes for the line with a target headway of 15 minutes and 10.4 minutes for the line with a target headway of 20 minutes.[1] These values are marginally larger than the theoretical minimum values of 5 minutes, 7.5 minutes and 10 minutes, respectively.

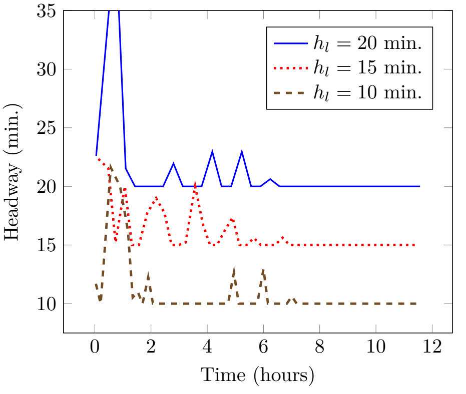
The persistent headway deviations without any buffer indicate that there is a constant shortage of vehicles. If the number of vehicles in the system is increased by one, we can observe that the headways do all converge to the target headways. Hence, this indicates that despite the absence of theoretical guarantees, the policy still performs well, but that a larger number of vehicles may be required to ensure that the target

[1] These values are computed using the formula $\mathrm{E}(W)=\frac{\mathrm{E}(h)}{2}(1+\mathrm{CV}^2))$, with W the waiting time, h the headway and CV the coefficient of variation of the headway.

headways are met at all times.



(a) Buffer = 0 vehicles



(b) Buffer = 1 vehicle

Figure 4.9: Headways of lines with different target headways, plotted over time, with and without a buffer.

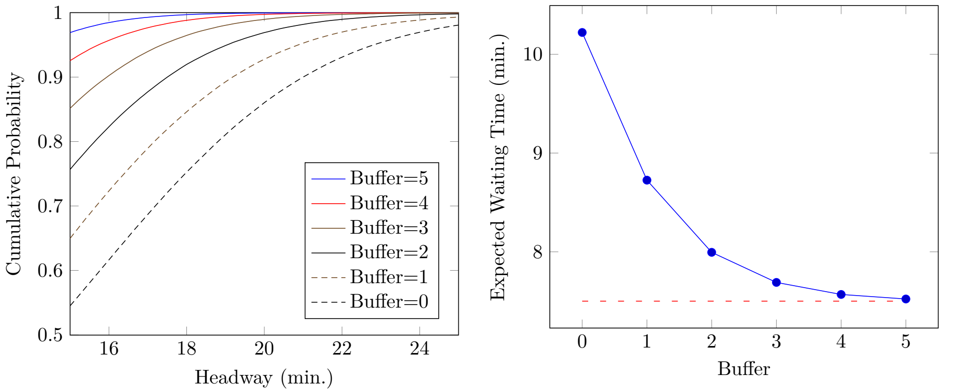
4.5.5 Stochastic Travel Times

In this experiment, we test the performance of the policy in case the travel times are not fixed, as we assumed in the theoretical analysis, but stochastic. We perform this experiment on a star network with five lines with a target headway of 15 minutes. The nominal travel time for each line is uniformly drawn between 10 and 30 minutes. The realized travel time is equal to the sum of the nominal travel time and a disturbance term. As it is likely that there is correlation in the duration of subsequent trips, we generate the disturbances ε^l for each line according to an autoregressive model:

$$\varepsilon_i^l = \rho\varepsilon_{i-1}^l + \eta_i, \quad \eta_i^l \sim N(0, \sigma^l). \tag{4.2}$$

We set $\rho = 0.8$ and $\sigma^l = \frac{1}{4}t_l$. To get a complete image of the variation of the headway in this scenario, we simulate 100 hours with ten different seeds, during which we collect all headways.

The results are presented in Figure 4.10a and Figure 4.10b. Figure 4.10a visualizes the empirical cumulative distribution function (ECDF) of the headway. Figure 4.10 shows the expected waiting time assuming uniform passenger arrivals. The ECDF and waiting times presented for different number of buffers, which are computed based on the nominal travel times. The ECDF shows for every size h of the headway, what proportion of the observed headways is smaller than h. For example, we can observe that with a buffer of 0, about 75% of the headways are smaller than 18 minutes. As expected, the headways are not always equal to the target headway of 15 minutes as there are constant disturbances keeping the system away from a stable state. However, it can be observed that the headways are reasonably close to the target headway. Even without any buffer, over 50 percent of the headways are equal to the target headway and 80 percent of the headways are shorter than 20 minutes. The expected waiting time without a buffer is more than 10 minutes, substantially larger than 7.5 minutes. When the buffer is larger, the headway distribution gradually shifts towards the left. With a buffer of 1, the expected waiting time already decreased to about 8.7 minutes. Therefore, this suggests that the policy performs reasonably well if the travel times are stochastic, but that a larger number of vehicles is required to obtain a (very) high service level.



(a) Empirical cumulative distribution function (ECDF) of the headway. For every size h of the headway on the horizontal axis, the ECDF shows what proportion of the observed headways is smaller than h on the vertical axis.

(b) Expected waiting time assuming uniform passenger arrivals for all lines. The dotted line is the expected waiting time under evenly spaced headways (7.5 minutes).

Figure 4.10: Results with stochastic travel times for a star network with five lines. The target headway equals 15 minutes.

4.5.6 Real-World Transit Systems

In a final experiment, we evaluate the performance of the policy on the transit systems of The Hague, Amersfoort and Göttingen. For the bus networks of The Hague and

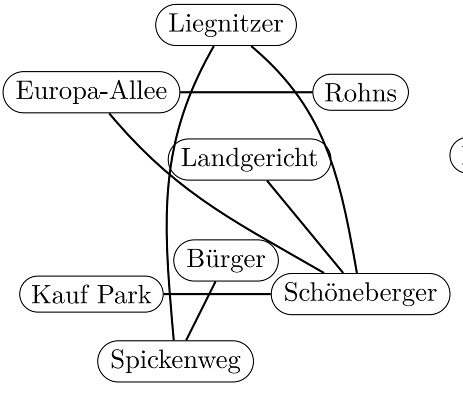
Amersfoort, we retrieve the lines, their headways and travel times from the published line plans and timetables. For the Göttingen bus network, we use a data set from the open source software library LinTim (Schiewe et al., 2020). As this line plan contains lines that are unidirectional, we only use a smaller part of the network, containing seven connected lines that are operated in both directions. The data for the three networks can be found at `http://www.ecopt.nl`.

The three real-world transit systems are visualized in Figure 4.11. As can be observed, the networks have different sizes and topologies. The Amersfoort network is star-shaped with all lines having the central station as a terminal. The network in The Hague has a different structure, with most lines terminating in neighborhoods outside the city center. The Göttingen network is a bit smaller and has a topology somewhat in between the networks of Amersfoort and The Hague. Table 4.2 presents for each transit system the number of stations and lines, the minimum number of vehicles required to meet the target headways, the minimum, maximum and average target headway across all lines, and the minimum, maximum and average travel time across all lines. For none of the transit systems, all lines have the same target headway. The lines in The Hague relatively have the smallest headways and those in Göttingen the largest.

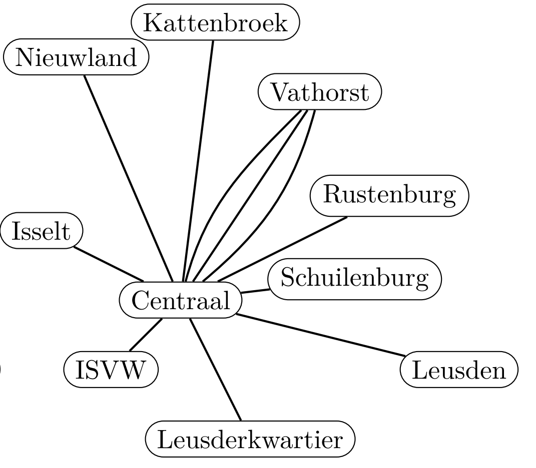
In the experiment we perform with the real-world transit systems, we let the realized travel times be stochastic. The travel times disturbances generated according to Equation (4.2). In accordance with the customary practice of public transport operators, we let the nominal (or average) travel times be such that that the realized travel time is smaller or equal to the timetabled travel time with 85% probability. We do so to take into account that public transport timetables include time supplements in order to absorb the majority of small delays.

Table 4.2: Characteristics of the considered transit systems.

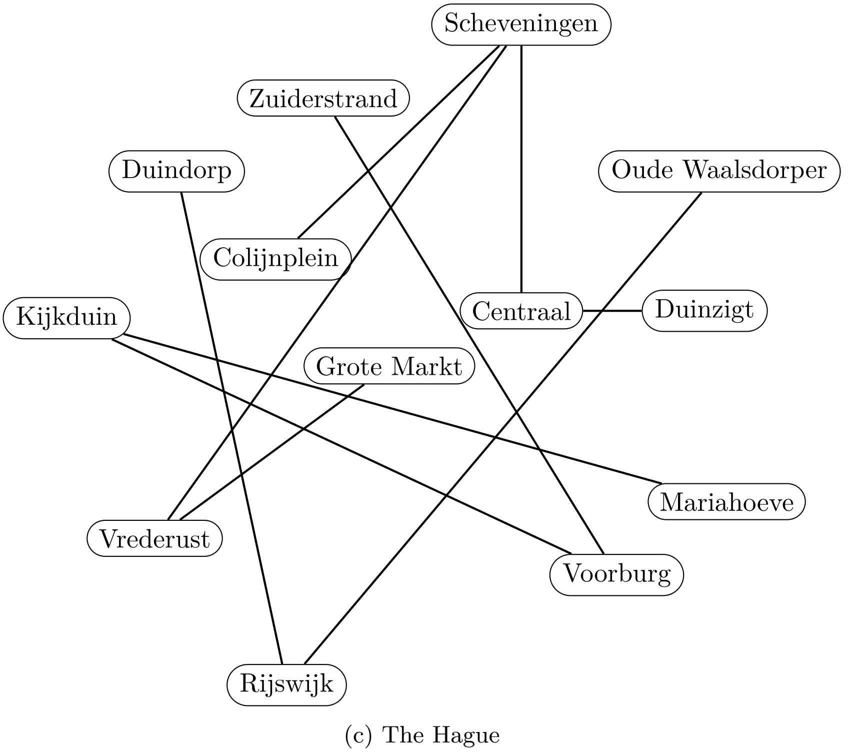
				Headway (min.)			Travel Time (min.)		
Network	Stations	Lines	Min. Veh.	Min.	Max.	Ave.	Min.	Max.	Ave.
Göttingen	8	7	16	15	60	36.4	19	43	34.6
Amersfoort	10	11	25	15	30	21.8	11	35	20.8
The Hague	13	10	65	7.5	30	14.0	17	68	40.3



(a) Göttingen



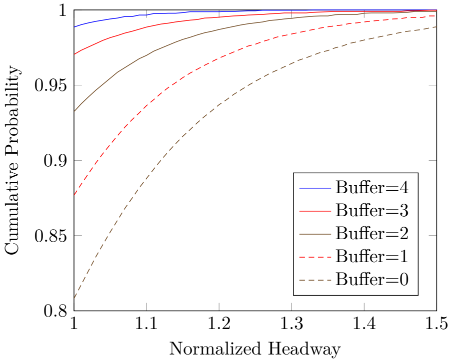
(b) Amersfoort



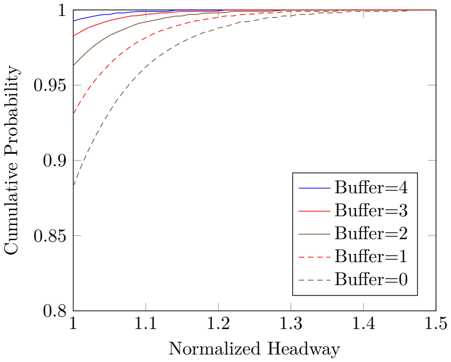
(c) The Hague

Figure 4.11: The three considered real-world transit systems networks.

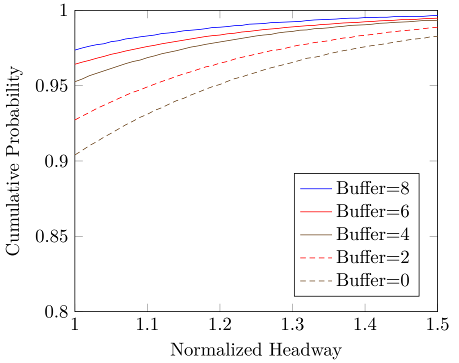
As in the previous experiment, we simulate 100 hours with ten different seeds, during which we collect all headways. As the considered networks contain lines whose target headways differ, we consider the *normalized headway*, which is simply the realized headway divided by the target headway. For example, if a line has a target headway of 10 minutes, a headway of 12 minutes corresponds to a normalized headway of 1.2 and a headway of 15 minutes to a normalized headway of 1.5. As such, the normalized headway should ideally be equal to 1.0.



(a) Göttingen



(b) Amersfoort



(c) The Hague

Figure 4.12: Empirical cumulative distribution function (ECDF) of the normalized headway for the real-world transit systems, under stochastic travel times and with different number of vehicles on the network.

In Figure 4.12, the empirical cumulative distribution function (ECDF) of the normalized headway is presented for the three real-world transit systems, with different sizes of the buffer. We find that without any buffer, the target headway is met around 80 percent of the time in Göttingen and around 90 percent of the time in Amersfoort and The Hague. Again without any buffer, a headway deviation of more than 20 percent (corresponding to a normalized headway of 1.2), occurs in around 5 percent of the cases in Göttingen and The Hague and only in around 1 percent of the cases in Amersfoort. For Göttingen and Amersfoort, a buffer of three vehicles is required to ensure that more than 95 percent of the headways meet the target headway. For the larger network of The Hague, a buffer of four vehicles is required to attain this service level. Overall, we can conclude that our policy performs well and leads to small deviations from the target headway with only a small number of additional vehicles.

4.6 Conclusion

We proposed a self-organizing policy for dispatching vehicles in multi-line public transit systems. Theoretical and numerical analyses illustrate that our policy performs well. In idealized conditions and provided that a sufficient number of vehicles is available, it is guaranteed that the system converges to a stable state where the target headway of each line is met. Experiments based on three real-world transit systems show that the policy also attains good performance in case travel times are not fixed but stochastic, or lines have different target headways: the deviations from the target headways are small, especially if there is some reserve capacity in the number of vehicles. Furthermore, our policy causes the system to quickly recover after disruptions, such as the breakdown of a vehicle.

Our promising theoretical and numerical results show that the potential of self-organizing strategies extends to multi-line public transit networks. Specifically urban high-frequency networks seem suited for our approach, as convergence is more rapidly established if the target headway and size of the network are small. However, for networks with heterogeneous headways, theoretical performance guarantees can only be obtained by decomposing the network into sub-networks with homogeneous headways. Compared to schedule-based approaches, the self-organizing approach is much easier to implement, as it does not require constructing a schedule, monitoring adherence to the schedule and rescheduling after disruptions. Only the lines and target headway need to be determined, which should be set in order to provide good service to all passengers, of course taking the number of available vehicles into account.

For further research, it would be interesting to investigate if the policy can be generalized to ensure that headways are always self-equalizing, even if $n < n^*$. This requires the target headway to no longer be exogenous, but emerge spontaneously due to the dynamics of the system. It is an open question whether this can be achieved by a simple decentralized policy, without coordination or communication between different parts of the network. Another potential direction of future research is to extend the policy to also resist bunching effects, for example by integrating our work with that of Bartholdi and Eisenstein (2012). Finally, in this chapter the passengers have been taken into account implicitly, as maintaining constant headways minimizes passenger waiting time if passengers arrive uniformly. It would be interesting to model the costs of passengers more explicitly, revealing the interactions between different dispatching approaches and the experience of passengers.

Chapter 5

Determining and Evaluating Alternative Line Plans in Out-of-Control Situations

This chapter is based on Van Lieshout, R.N., Bouman, P.C. and Huisman, D. (2020). Determining and Evaluating Alternative Line Plans in Out-of-Control Situations. Transportation Science 54(3), 740–761.

5.1 Introduction

Every once in a while, railway systems suffer from very large disruptions as the result of power outages, extreme weather conditions or other severe incidents. Intensive use of the railway infrastructure and strong interdependencies between timetable, rolling stock and crew schedules cause these disruptions to easily propagate and accumulate. Since the affected number of resources can be very large and durations of disruptions are uncertain, this leads to an immensely complex problem for dispatchers. As a result, decisions made by dispatchers take a long time, and are often based on information that is already out-of-date, rendering the decision unworkable. In the end, dispatchers are confronted with a lack of accurate and up-to-date information on the current state of the system, preventing them from making viable rescheduling decisions. This can ultimately result in an *out-of-control situation*, meaning that all traffic in the affected region is terminated, even though the required resources (infrastructure, rolling stock and crew) might be available.

On the Dutch railway network, out-of-control situations happened about ten times during the period 2009-2012 because of extreme weather conditions. For this reason, Netherlands Railways (NS), the largest railway operator in the Netherlands, and ProRail, the Dutch infrastructure manager, decided to anticipate on such events by operating a reduced timetable if bad weather is expected. This reduces the probability of losing control, but the downside is that a worse service is provided to the passengers. Moreover, this does not prevent all out-of-control situations, since they also occur with completely unexpected causes, such as the power outages in Amsterdam in 2015 and 2017 and the terrorist attack in Amsterdam Central Station in 2018. In these cases, control was lost during the process of restarting operations after the forced shutdowns. As such, there is a clear need for effective countermeasures in out-of-control situations.

In spite of the many recent advancements in disruption management, currently existing disruption management techniques cannot be applied in out-of-control situations due to the absence of complete information and the large number of affected trips and resources. Therefore, Dekker et al. (2021) propose a new strategy for avoiding or escaping out-of-control situations. The core idea of this strategy is to completely decouple the disrupted region from the rest of the railway network. Rolling stock nor crew is allowed to change over between the two regions, isolating the disruption and preventing it from spreading. The traffic outside the disrupted region can be managed using conventional disruption management techniques, because in that region complete information is available. Inside the disrupted region, the lack of accurate and up-to-date information makes it very difficult to dispatch trains according to a centrally agreed upon timetable. Therefore, it is suggested to adjust the line plan within the disrupted region, such that it becomes possible to dispatch trains in the region using local and self-organizing mechanisms, which can be applied even when the information about the resources in the system degrades. All in all, this strategy could lead to stable operations both inside and outside the disrupted region, where the current practice simply terminates all traffic within the affected region.

In this chapter, we examine whether the strategy proposed by Dekker et al. (2021) indeed provides a viable alternative in out-of-control situations. In particular, we investigate how the line plan within the disrupted region should be modified. An alteration of the line plan is necessary, as the regular line plan most likely becomes infeasible due to limited turning capacity at the boundary stations and a limited availability of rolling stock in the disrupted region. In addition, we investigate how trains within this region can be dispatched without relying on central coordination. Note that by addressing these two research questions, we increase the scope of railway disruption management, which traditionally involves rescheduling the timetable, rolling stock schedule and crew schedule, by also modifying the line plan and the way the system is operated.

The contribution of this chapter is threefold. The first contribution is a novel line planning algorithm that provides as many passengers with as many travel options as possible, while taking into account timetabling and rolling stock constraints in a Benders'-like approach. In our application, integration of multiple planning steps is required as we alter the line plan on the day of operations, hence it needs to be guaranteed that the line plan is feasible with respect to the available infrastructure capacity and rolling stock. However, the same decomposition approach can also be applied in strategic planning contexts, where it avoids having to optimize a complete timetable and/or rolling stock schedule before finding out that a line plan is in fact infeasible. The second contribution is that we propose several local dispatching strategies that can be applied in out-of-control situations. The proposed strategies differ in the amount of required information and flexibility, offering a range of options to dispatchers. The final contribution is the evaluation of the produced line plans and suggested dispatching strategies by simulating them on the Dutch railway network. In particular, we illustrate that by applying the proper dispatching strategies, the modified line plans indeed lead to feasible operations, as intended.

The structure of the chapter is as follows. In Section 5.2, the problem is described in detail. In Section 5.3, relevant literature is discussed. Section 5.4 addresses the line planning algorithm. Section 5.5 describes the local dispatching strategies and discusses how to test the performance of line plans and dispatching strategies by means of simulation. In Section 5.6, the results of both the line planning algorithm and the simulations are presented. We conclude in Section 5.7.

5.2 Problem Description

The schedule of a railway operator typically consists of the following main components. The *line plan* specifies between which stations direct trains are operated, their frequencies and the stopping patterns. The *timetable* specifies the exact departure and arrival times of train services. The *rolling stock schedule* specifies the train units that are used for each trip. Finally, the *crew schedule* specifies which train drivers and conductors perform which tasks. In an out-of-control situation, all components are heavily disrupted and are required to be modified. Dekker et al. (2021) describe a new approach for dealing with these situations, taking the size of the disruption and the lack of complete information into account. Their framework is visualized in Figure 5.1.

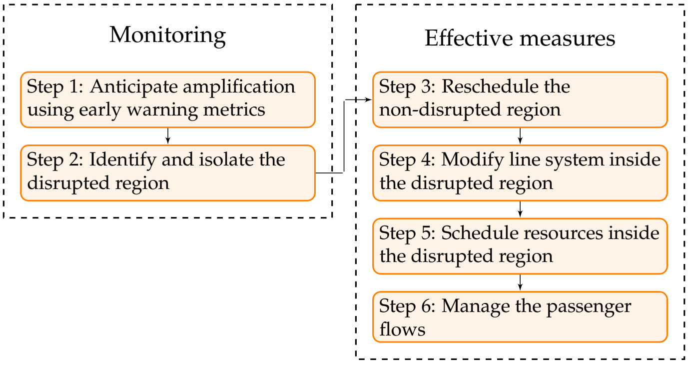


Figure 5.1: The framework for dealing with out-of-control situations proposed by Dekker et al. (2021).

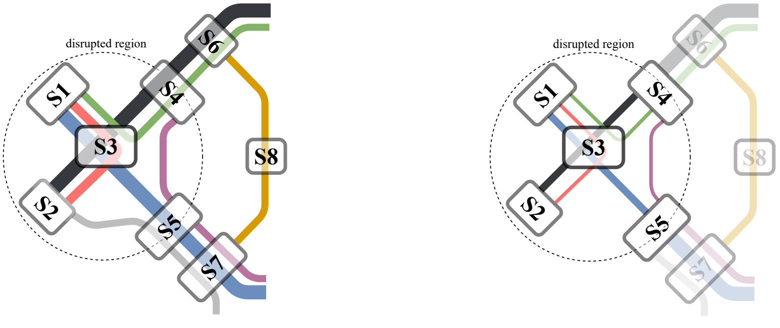
In the first step, it is detected that the railway system is in a state of (near) out-of-control and the rest of the framework is set in motion to prevent or escape the out-of-control situation. In the second step, the *disrupted region* is identified. Once this region has been determined, no rolling stock or crew is allowed to transfer between the disrupted and the non-disrupted region, in order to encapsulate the disruption. In practice, the disrupted region often directly follows from the cause of the out-of-control situation. For example, if a temporary power outage in Amsterdam has resulted in an out-of-control situation around Amsterdam, the disrupted region will most likely consist of Amsterdam Central Station and a number of surrounding stations. The strict separation between the operations in the disrupted and non-disrupted region is maintained for the remainder of the day, as re-coupling the regions is very complex and comes with the risk of new disruptions. Over night, all resources can be set up again in order to start the regular timetable on the next day.

Besides preventing the disruption from spreading, decoupling the disrupted region makes it possible to develop tailored disruption management for both regions, which are steps 3 to 6 of the framework. Most importantly, it can be assumed that outside the disrupted region complete information is available, while inside there is not. For this reason, it is argued that the non-disrupted region can be rescheduled using conventional disruption management techniques, see Cacchiani et al. (2014) for a recent review. To operate the disrupted region, Dekker et al. (2021) propose to adjust the line plan and to schedule the resources using local, self-organizing strategies. Such strategies strongly reduce the dependence on central dispatchers and are robust against the lack of complete and up-to-date information that characterizes out-of-control situations.

In this chapter, we focus on managing the operations inside the disrupted region, steps 4 and 5 of the framework. That is, we assume that a disrupted region has been identified and decoupled from the rest of the network, and investigate how the line plan within this region should be modified. In addition, we propose and evaluate self-organizing, local dispatching strategies that can be used to operate the railway traffic in the disrupted region. To limit the scope of this research, we only consider the available infrastructure and rolling stock in the dispatching strategies and when determining the line plan, so train drivers and conductors are not taken into account. As such, we assume that the crew operating a train at the moment the disrupted region is decoupled continues to operate that train for the rest of the day.

5.2.1 Modifying the Line Plan

A modification of the line plan within the disrupted region is necessary, as the original line plan most likely becomes infeasible when the region is decoupled from the rest of the network. To illustrate this, consider the example in Figure 5.2a. When the disrupted region is decoupled, trains from both sides have to turn at stations S4 and S5. As such, only a part of the platforms at these stations can be used to serve the disrupted region, which is probably not sufficient to operate as many train services as in the original line plan. Moreover, there is only limited rolling stock available in the disrupted region, while the average proportion of time that trains are running decreases when all trains must be turned at the boundary (turning a train takes more time than a regular stop). Taking these two factors into account, one might have to settle for the alternative line plan in Figure 5.2b.



(a) Original line plan (b) Alternative line plan

Figure 5.2: Modification of a line plan. Relative frequencies are indicated by the line thickness.

As the above presented example illustrates, the challenge is to find a line plan that offers sufficient transport capacity, while ensuring that the line plan is feasible with respect to the available infrastructure and available rolling stock. As such, even though we will operate the disrupted region using local dispatching strategies instead of using a timetable and rolling stock schedule, the line plan should admit a feasible timetable and rolling stock schedule. Therefore, by considering line planning as an operational rather than a strategic problem, we are forced to integrate line planning with the subsequent planning stages.

During operations, trains using the same piece of infrastructure must be separated by a *headway time* of a certain number of minutes. This implies that every line consumes a certain amount of capacity. In strategic planning, these restrictions are taken into account during the timetabling phase. As such, we can avoid finding line plans that require more capacity than available by considering timetabling restrictions when redesigning the line plan. However, in an out-of-control situation it suffices to consider a relaxed version of the timetabling problem as trains cannot be dispatched according to a centrally agreed upon timetable due to the lack of complete information. Hence, requiring that a line plan exactly fits in a period of e.g. 60 minutes is too strong of a restriction (a line plan that fits in a period of slightly over 60 minutes likely performs equally well under the local, self-organizing dispatching strategies). For this reason, we only consider certain necessary conditions for the existence of a timetable.

As every line requires a certain number of trains and the available number of trains in the disrupted region is limited, it is essential to take rolling stock into account when modifying the line plan. The set of rolling stock compositions available in the disrupted region at the time the region is decoupled serves as input. Given the nature of out-of-control situations, we assume that it is not possible to couple or decouple units to and from a composition. Therefore, we can ignore the rolling stock compositions of the operating trains and simply refer to a rolling stock composition as a train. A complicating factor is that the exact number of trains required to operate a line plan depends on the timetable, which is unavailable. However, it is possible to find lower bounds based on the minimum running, dwell and turning times.

Although our methodology extends to the general case, we consider only two types of lines, *intercity* lines that only have stops at major stations and *regional* lines that have stops at every station. Both line types have dedicated rolling stock types. Furthermore, we assume that it is given which stations are *decoupling stations*, the stations that may serve as terminals for a line. Regional lines can be introduced between regional as well as intercity decoupling points, intercity lines can only be introduced between intercity decoupling points.

The main objective when modifying the line plan is to minimize passenger inconvenience. In (the aftermath of) out-of-control situations that occurred over the last few years, travelers often had to seek alternative modes of transport as there were either no trains running or not all travelers would fit in the few trains that were running. Therefore, maintaining a large transport capacity is crucial, which is why we measure the inconvenience of a passenger through the decrease in the number of travel options per period the passenger has in the modified line plan compared with the original line plan. For example, if a passenger has four travel options per hour in an undisrupted situation, ideally the modified line plan should offer this passenger four travel options per hour. However, given the restrictions it is inevitable to reduce the number of travel options for some groups of passengers.

5.2.2 Operating the Disrupted Region

Once the new line plan for the disrupted region has been determined, we can start executing the line plan by dispatching trains. In regular operations, trains are dispatched according to the timetable and adjustments are made to the timetable in case of disturbances or disruptions. Conversely, when operating trains in the disrupted region during an out-of-control situation, a timetable is not available. Therefore, radically different train dispatching strategies are required to operate the modified line plan. These strategies should specify simple rules that determine when trains depart. Moreover, as out-of-control situations are characterized by a lack of complete and accurate information, the strategies should be *local*, meaning that only information of the directly surrounding part of the railway network is required to decide what to do next, such that dispatching decisions can be made locally. We limit ourselves to determining departure times for trains, and scheduling the rolling stock. Train drivers and conductors are not taken into account.

5.3 Literature Review

Traditionally, railway disruption management involves finding a new timetable by rerouting, retiming (delaying) and canceling train services and rescheduling the rolling stock and crew such that the adapted timetable is compatible with the adapted resource schedules (Jespersen-Groth et al., 2009). Given that complete integration of these steps is computationally intractable, contributions focus on one of the rescheduling steps. Recently, researchers have started to incorporate aspects of multiple rescheduling phases in their models. For example, Veelenturf et al. (2015) consider the timetabling rescheduling problem, but also guarantee that every trip in the adapted timetable can be assigned a rolling stock composition. A second interesting development is the application of robust optimization methods. For example, Veelenturf et al. (2014) take the uncertain disruption duration into account when solving the crew rescheduling problem. For recent surveys on railway disruption management we refer to Cacchiani et al. (2014) and Ghaemi et al. (2017).

To the best of our knowledge, we are the first to consider redesigning the line plan after disruptions. In strategic planning on the other hand, line planning has received a considerable amount of attention, see Schöbel (2012) for a general survey. However, existing methods are not directly applicable to our problem, since in strategic contexts computation time is of less importance and it is usually assumed that infrastructural and rolling stock restrictions can be dealt with later. In this chapter however, the goal is to find a modified line plan that is feasible with respect to the infrastructure and actual available rolling stock, in real time. Therefore, we limit ourselves to discussing papers that (partially) integrate line planning with timetabling or vehicle/rolling stock scheduling.

Schöbel (2017) argues that the conventional approach of solving line planning, timetabling and vehicle scheduling problems in the traditional sequential manner can be seen as a greedy and therefore suboptimal algorithm. By considering the problem in an integrated way, one should be able to find solutions that are better for both operators and passengers. As no efficient algorithm for the complete integrated problem currently exists, the author presents a way to design new heuristics by iterating between line planning, timetabling and vehicle scheduling.

Kaspi and Raviv (2013) develop a metaheuristic to solve the integrated line planning and timetabling problem. In every iteration of their heuristic, trains are randomly scheduled based on certain distribution parameters. The resulting operator costs and passenger inconvenience is used to change the parameters for the next iteration. Burggraeve et al. (2017) also iterate between the line planning and timetable phase, but use feedback from the timetable to make deterministic changes to the line plan. In the line planning problem, constraints are included that increase the likelihood a timetable exists with large enough buffer times between trains. With this approach they are able to construct line plans that allow for timetables with larger minimum buffer times, thereby increasing the robustness. In this chapter, we use a similar feedback loop connecting line planning and timetabling, but we compute partial timetables instead of a complete timetable as a quick check (albeit more crude) that the line plan does not exhaust all infrastructure capacity suffices in our application.

Some cost-oriented line planning approaches consider aspects from rolling stock scheduling in the line planning problem. Claessens et al. (1998) and Goossens et al. (2004) decide which lines to operate as well as how many carriages should be assigned to each line. Based on the driving, dwell and turnaround times, this can be used to compute a lower bound on the number of train units that is necessary to operate the line plan (the exact number depends on the timetable). A similar approach is taken by Pätzold et al. (2017), who construct line plans that can likely be operated with a small number of vehicles by only considering lines that can be operated efficiently with a fixed circulation. Specifically, a line can only be selected if a vehicle continuously performing round trips of the line has a short downtime between round trips in order to maintain periodicity. Chapter 2 extends these ideas and compute the minimum number of vehicles required to operate a line plan, while allowing vehicles to perform circulations consisting of multiple lines. In this chapter, we also consider combining circulations as it results in more efficient use of the available rolling stock, and integrate these decisions into the line planning problem.

There are also studies on integrating line planning and rolling stock considerations into the timetabling problem. Liebchen and Möhring (2007) partially incorporate line planning by assuming that line segments are fixed and need to be matched at a certain station to constitute lines. In addition, the authors show that for certain cases the required amount of train compositions for a line can be taken into account in the timetabling problem. Kroon et al. (2013) are able to generalize this idea. Under the assumption that arriving and departing trains are sufficiently spread, they are able to find timetables requiring the smallest possible number of rolling stock compositions.

Local dispatching strategies, which we use to operate the modified line plans in the disrupted region, have to the best of our knowledge not yet been applied in railway settings. In bus networks on the other hand, there is some work on this topic. Dessouky et al. (1999) simulate different strategies for holding buses at transfer stations in order to find a good compromise between missed connections and passenger delays. Other studies investigate holding strategies to maintain stable headways and prevent bus bunching. Daganzo (2009) proposes and analytically analyzes a dynamic control scheme for a single line where every bus is accelerated or decelerated based on the real-time headway with the next bus. Bartholdi and Eisenstein (2012) consider a simpler setting with a single control point on a circular line, and propose a simple holding strategy that leads to constant headways in the long run, regardless of initial headways. Argote-Cabanero et al. (2015) address the bus bunching problem for more complex network structures and define a strategy where the holding time of a bus at a station is determined by real-time headway deviations of all lines that visit the same station. Besides holding control approaches based on headway deviation, researchers have proposed (stochastic) optimization models to determine optimal dispatching decisions (Berrebi et al., 2015; Delgado et al., 2012; Sánchez-Martínez et al., 2016). However, such models require a lot of information and depend on real-time communication between the vehicles and a central control center, which is why we do not consider this approach viable for our problem. Instead, we define a number of strategies based on headway deviations and use simulation to assess their performance.

5.4 Line Planning Algorithm

Our line planning algorithm rests on a decomposition in a *master* problem and a *slave* problem, which can be described as the integer or combinatorial variant of Benders' decomposition (Codato and Fischetti, 2006; Vanderbeck and Wolsey, 2010). The master problem amounts to finding the optimal line plan subject to certain timetabling and rolling stock restrictions, such that both the available infrastructure and the available rolling stock is taken into account. The line plan produced by the master serves as input for the subproblem, which performs an additional capacity check by evaluating whether the line plan admits feasible *partial* timetables, which we define as timetables for each station independently. If the line plan is feasible, the algorithm terminates. If not, we identify a combinatorial cut in terms of the variables of the master problem. The cut is then added to the master, after which the process iterates.

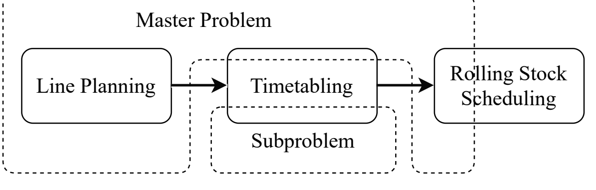


Figure 5.3: Decomposition of the planning problems into a master problem and a subproblem.

Figure 5.3 visualizes the decomposition. The master problem contains the entire line planning problem, but, as we include timetabling and rolling stock constraints, also includes parts of the timetabling and rolling stock scheduling problems. The master problem is discussed in Section 5.4.2. In the subproblem, the timetable feasibility of the line plan is further evaluated, but since we only compute partial timetables and not a complete timetable for the whole disrupted region, not the entire timetabling problem is integrated in the algorithm. Section 5.4.3 explains this in detail.

In both the master problem and the subproblem, the timetabling constraints are based on necessary conditions for the existence of a feasible periodic timetable, in which the schedule exactly repeats each period. Such a timetable fits most naturally to the concept of a line plan, since a line plan specifies frequencies per period. Furthermore, the disrupted region will be operated according to local dispatching rules and not according to a timetable. As a result, it is likely that travelers arrive at stations approximately uniformly, in which case regular interdeparture intervals (e.g. every 30 minutes) minimize traveler waiting time. Therefore, we base the timetabling constraints on periodic timetables, even though the realized timetable will most likely deviate.

5.4.1 Definitions and Concepts

We represent the railway network using a connection network $G = (S, E)$. The set S contains the stations in the disrupted region and the set E contains an edge for every pair of stations between which a train can run without dwelling at an in-between station. In particular, we have *basic edges* connecting all pairs of stations that have no other station in-between, and *intercity edges* connecting all pairs of adjacent intercity stations, see Figure 5.4 for an example. We let T denote the period of the original line plan and let g_e denote the original edge-frequency of edge e (the frequency at which trains are originally operated on the edge).

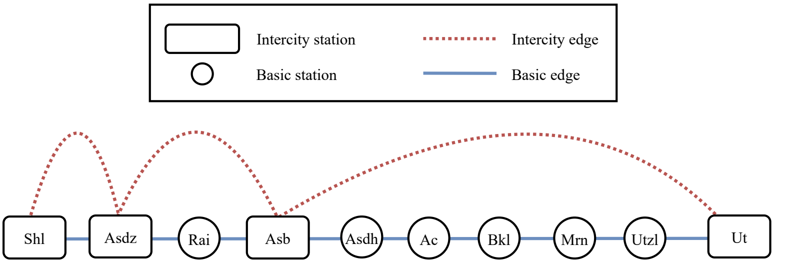


Figure 5.4: Connection graph of a part of the Dutch railway network.

A line l is defined as a tuple (π_l, f_l), with $\pi_l = (e_1, e_2, ..., e_m)$ a path in G and f_l an integer representing the intended frequency per period of T time units. If l is a regional line, all edges in the path must be basic edges and if l is an intercity line, all edges must be intercity edges. A line l dwells at all stations in the path π_l. The terminal stations of a line must be decoupling stations compatible with the type of the line. We let S_l denote the set of stations that line l passes. If l is an intercity line, S_l also contains the stations where l passes but does not dwell. We assume lines are always operated in both directions. The set Π contains all paths that can be used to form lines and the line pool $\mathcal{L}$ contains all lines that may be selected. As is most line planning models, we assume these sets are defined a priori (Schöbel, 2012). We use subscripts to denote certain subsets of the line pool: the set $\mathcal{L}_e$ contains all lines covering edge $e \in E$, the set $\mathcal{L}_s$ contains all lines attending station $s \in S$ and the set $\mathcal{L}_\pi$ contains all lines with path $\pi \in \Pi$.

We let $\mathcal{OD} \subseteq S \times S$ denote the set of all origin-destination (OD) pairs and let $n_{o,d}$ denote the number of passengers traveling from station o to station d. We let $\rho_{o,d} \subseteq E$ denote the edges that appear in the shortest path from o to d. The shortest paths are computed using the running times on all edges. We let $\mathcal{OD}_{\text{dir}}$ denote the set of all OD pairs that have a direct connection in the regular line plan and let $\mathcal{L}_{o,d}$ denote the set of lines offering a direct connection between stations o and d. Finally, we let $g_{o,d}$ and $g^{\text{dir}}_{o,d}$ denote the total number of travel options per hour and the number of direct travel options between o and d in the original line plan, respectively. We also refer to $g_{o,d}$ and $g^{\text{dir}}_{o,d}$ as the original *OD-frequency* and the original *direct OD-frequency*.

5.4.2 Solving the Master Problem

We start by explaining the canonical form of the master problem. Thereafter, we describe the timetabling and rolling stock constraints that are included.

The binary decision variables x_l indicate whether line l is selected. Next, the integer decision variables $z_{o,d}$ represent the reduction in OD-frequency between o and d. Similarly, the integer decision variables $z^{\text{dir}}_{o,d}$ represent the reduction in the direct OD-frequency between o and d. The canonical form of the master problem is then given by:

$$
\begin{aligned}
\text{minimize}\quad & w_1 \sum_{(o,d)\in\mathcal{OD}} n_{o,d}\left(\frac{z_{o,d}}{g_{o,d}}\right)^2 + w_2 \sum_{(o,d)\in\mathcal{OD}_{\text{dir}}} n_{o,d}\left(\frac{z^{\text{dir}}_{o,d}}{g^{\text{dir}}_{o,d}}\right)^2 + w_3 \sum_{l\in\mathcal{L}} x_l && (5.1)\\
\text{s.t.}\quad & z_{o,d} + \sum_{l\in\mathcal{L}_e} f_l x_l \ge g_{o,d} && \forall (o,d)\in\mathcal{OD},\quad \forall e \in \rho_{o,d}, \quad (5.2)\\
& z^{\text{dir}}_{o,d} + \sum_{l\in\mathcal{L}_{o,d}} f_l x_l \ge g^{\text{dir}}_{o,d} && \forall (o,d)\in\mathcal{OD}_{\text{dir}}, \quad (5.3)\\
& \sum_{l\in\mathcal{L}_e} f_l x_l \le g_e && \forall e\in E, \quad (5.4)\\
& \sum_{l\in\mathcal{L}_\pi} x_l \le 1 && \forall \pi\in\Pi, \quad (5.5)\\
& \text{(timetabling constraints)}, && (5.6)\\
& \text{(rolling stock constraints)}, && (5.7)\\
& x_l \ \in \{0,1\} && \forall l\in\mathcal{L}, \quad (5.8)\\
& z_{o,d}, z^{\text{dir}}_{o,d} \ge 0 \text{ and integer} && \forall (o,d)\in\mathcal{OD}. \quad (5.9)
\end{aligned}
$$

The objective is to minimize the weighted sum of the decrease in OD-frequency, the decrease in direct OD-frequency and the number of lines. The two latter terms are included with relatively smaller weights to favor line plans that offer more direct connections and have fewer lines. A quadratic objective is used for the first two terms to distribute the inconvenience fairly over all passengers. That is, if two OD pairs have the same original OD-frequency, we prefer to reduce both frequencies by one instead of reducing one of the frequencies by two. Moreover, if two OD pairs have a different original OD-frequency we prefer to reduce the OD-frequency of the OD-pair with the higher frequency, as this results in a lower relative decrease. As the $z-$variables take on a limited number of values, the quadratic formulation can easily be transformed into an equivalent linear formulation by defining some auxiliary variables. We provide this linearization in Appendix 5.A.

As for the remainder of the formulation, constraints (5.2) and (5.3) make sure that the decreases in OD-frequency and direct OD-frequency are measured correctly. Constraints (5.4) guarantee that for every edge, the total frequency of the selected lines covering that edge is at most the original frequency of the edge (we do not allow operating more trains than in the original line plan). Constraints (5.5) impose that for every path, at most one line can be selected (so we cannot operate two lines with the same path but a different frequency). In the following sections, we provide constraints (5.6) and two variants for constraints (5.7).

Infrastructure Capacity

In this section, we derive necessary conditions for the existence of a feasible timetable in terms of the line planning variables. These constraints ensure that we do not find line plans that cannot be operated on the available infrastructure. The idea behind these conditions is that every line consumes a certain amount of the capacity at stations and a timetable can only exist if not all capacity is exhausted. The novelty of our approach is that when determining how much capacity a line requires, we take into account that in periodic timetables trains often have longer stops and turnaround times to enforce the periodic pattern. To illustrate this, consider a line between stations A and B with a frequency of 2 and a travel time between A and B of 31 minutes. Assume all lines turn on themselves, meaning that when a train arrives at its terminal station, the train turns and starts performing the reverse trip of the trip the train just finished. In a periodic timetable with a period of 60 minutes, without loss of generality trains depart at station A at minutes 0, 30, 60 and so on. Figure 5.5 depicts an example of a timetable for this line. The train leaving A at 0 arrives at B at minute 31. Assuming a minimal turning time of 5 minutes, the soonest the train can again arrive at A is at minute 67. Likewise, the soonest the train is ready to depart again from A is at minute 72. However the earliest next trip it can perform starts at minute 90. This implies that on top of the minimum dwell times, we are certain the train has to dwell an additional 18 minutes. In the displayed timetable, the additional dwell time or *downtime* is spent entirely at station A, but it is of course possible to spread this time over the stations where the train dwells.

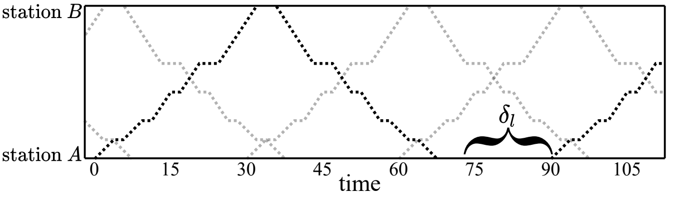


Figure 5.5: Time space diagram depicting the timetable of a line between A and B with frequency of 2 per hour. The downtime is denoted by δ_l.

Clearly, a timetable can only exist if there is sufficient platform capacity to serve all selected lines, taking into account that the total dwell times must be long enough to enforce the periodic pattern. To formalize this, let P^s denote the set of platforms at station s and let $P^s_{ld} \subseteq P^s$ denote the set of platforms that can be used by a train of line l arriving from direction d, taking into account infrastructural restrictions. The set D^s_l contains the directions of line l at station s. If s is a terminal station of line l, this set only contains one element, otherwise two. Next, the parameter b^s_l denotes the minimum time a platform at station s is blocked when a train from line l attends station $s \in S_l$. This time includes the headway time that needs to separate two trains using the same resource, and the minimum dwell time if l stops at s. Furthermore, we let t_l denote the minimum time it takes a train to perform a complete circulation of line l, i.e. twice the sum of all driving and minimum dwell times. Then, the downtime δ_l of line l is the smallest positive number satisfying $t_l + \delta_l = 0 \mod \frac{T}{f_l}$. That is, the downtime is the additional dwell time needed to enforce periodicity. The parameter $\delta^{\max}_{ls}$ indicates the maximum allowed additional dwell time of line l at station s.

To incorporate these conditions in the master problem, we declare variables and constraints that describe both the platform assignment of the selected lines and how the downtime is divided over the stations. We introduce the variables w_{lsd} representing the downtime of line l at station s in direction d. In a periodic timetable, this value is the same for every train of the line in the same direction. We also introduce the binary decision variables $y_{l_i sdp}$, which are equal to 1 if the i'th train service of line l at station s in direction d is assigned to platform p, where $i = 1, 2, ..., f_l$. Then, the following set of constraints are necessary conditions for the existence of a periodic timetable:

$$\sum_{s \in S_l} \sum_{d \in D^s_l} w_{lsd} = \delta_l x_l \qquad \forall l \in \mathcal{L}, \tag{5.10}$$

$$\sum_{p \in P^s_{ld}} y_{l_i sdp} = x_l \qquad \forall l \in \mathcal{L}, \quad i = 1, ..., f_l, \quad \forall s \in S_l, \quad \forall d \in D^s_l, \tag{5.11}$$

$$\sum_{l \in \mathcal{L}_s} \sum_{d \in D^s_l} \sum_{i=1}^{f_l} (b^s_l + w_{lsd}) y_{l_i sdp} \leq T \quad \forall s \in S, \quad \forall p \in P^s, \tag{5.12}$$

$$0 \leq w_{lsd} \leq \delta^{\max}_{ls} \qquad \forall l \in \mathcal{L}, \quad \forall s \in S_l, \quad \forall d \in D^s_l, \tag{5.13}$$

$$y_{l_i sdp} \in \{0, 1\} \qquad \forall l \in \mathcal{L}, \quad i = 1, ..., f_l, \quad \forall s \in S_l, \quad \forall d \in D^s_l, \quad \forall p \in P^s_{ld}. \tag{5.14}$$

Constraints (5.10) ensure that if a line is selected, the downtime is divided over the stations. Constraints (5.11) ascertain that the train services of selected lines are assigned to platforms. Constraints (5.12) impose that the total time a platform is blocked per period is less than the total available time. Constraints (5.13) guarantee that the downtimes are nonnegative and less than the specified upper bounds. Constraints (5.12) can be linearized by introducing the decision variables $w_{l_i sdp} = w_{lsd} y_{l_i sdp}$. This relation is enforced by adding the following linear constraints:

$$w_{l_i sdp} \geq w_{lsd} - \delta_{ls}^{\max}(1 - y_{l_i spd}) \quad \forall l \in \mathcal{L}, i = 1, ..., f_l, \forall s \in S_l, \forall d \in D_l^s, \forall p \in P_{ld}^s. \tag{5.15}$$

Rolling Stock Capacity

We take rolling stock scheduling into account when redesigning the line plan by imposing that for every selected line, there must be a sufficient number of trains assigned to circulations covering to the line. We distinguish two cases. In the first case we assume that all rolling stock circulations must be *fixed*, such that all trains turn on themselves. In the second case we relax this assumption and allow for *flexible* rolling stock circulations, in which trains can switch lines when they reach their terminal station. This leads to shorter turning times, which allows more trains to be operated as (i) trains spend less time dwelling and more time running and (ii) pressure is released at the turning stations, such that there is capacity for additional trains. Moreover, flexible circulations increase flexibility during operations. As a timetable is unavailable, the number of trains required to operate circulations is determined based on the minimum trip times in both cases.

Note that the downtime constraints (5.10) defined in the previous section are valid under the assumption of fixed circulations. It is possible to generalize these constraints, such that the additional dwell time is not distributed over the stations of all selected lines, but over the stations of the selected circulations. However, this constraint is rarely violated when flexible circulations are allowed. The reason for this is that in order to use the available rolling stock as efficiently as possible, the model selects circulations with short downtimes. Moreover, in the rare case that the constraint is violated, this likely has a small impact on the overall performance, as the line plan will be operated according to local dispatching rules and crude timetabling constraints suffice. Therefore, we ignore constraints (5.10) if we allow for flexible circulations. Note that constraints (5.11-5.12) and (5.14) then still restrict the allowed capacity consumption of the line plan.

Fixed Rolling Stock Circulations

We let n_l denote how many trains are at least necessary to operate line l when operated solely with fixed circulations. This value can be computed as follows:

$$n_l = \left\lceil \frac{t_l}{T} f_l \right\rceil, \text{ or alternatively } n_l = \frac{t_l + \delta_l}{T} f_l. \tag{5.16}$$

Next, we let R denote the set of available trains located in the disrupted region. The parameter a_{rl} indicates whether train r can be assigned to line l, based on the location and type of the train. We introduce decision variables v_{rl} indicating whether train r is assigned to line l. The allocation of rolling stock can be included in the master problem by adding the following constraints:

$$\sum_{r \in R} a_{rl} v_{rl} = n_l x_l \quad \forall l \in \mathcal{L}, \tag{5.17}$$

$$\sum_{l \in \mathcal{L}} v_{rl} \leq 1 \quad \forall r \in R, \tag{5.18}$$

$$v_{rl} \in \{0, 1\} \quad \forall l \in \mathcal{L}, \quad \forall r \in R. \tag{5.19}$$

These constraints state that if a line is selected, n_l trains should be assigned to the line and that every train can only be assigned to a single line.

Flexible Rolling Stock Circulations

The advantage of flexible circulations is that multiple lines with long downtimes can be combined into a circulation with a short downtime. Consider for example two lines l and m with $f_l = f_m = 1$ and $t_l = t_m = 85$ minutes such that $\delta_l = \delta_m = 35$ minutes. Combining these lines in a circulation gives a travel time of 170 minutes and a downtime of only 10 minutes, which is clearly more efficient.

We formally define a rolling stock circulation c as a sequence of lines $c = l_1, l_2, ..., l_{|c|}$, such that all consecutive lines, and the first and last line, have a shared terminal station and all lines are either all regional or all intercity lines. A train performing this circulation continuously traverses the sequence from left to right: first the train performs a round trip of l_1, then a round trip of l_2 and so on. We let f_c denote the frequency of circulation c, which equals the minimum frequency of the lines in the circulation. Analogously to lines, we let t_c denote the minimum travel time of circulation c, which equals $\sum_{l\in c} t_l$, and we let δ_c denote the downtime of a circulation, which equals the smallest positive number satisfying $t_c + \delta_c = 0 \mod \frac{T}{f_c}$.

To provide a mathematical formulation of flexible rolling stock circulations, we let $\mathcal{C}$ denote the set of allowed circulations and introduce the decision variables γ_c, indicating whether circulation c is selected, and θ_c, representing how many trains perform circulation c. We also have the assignment variables v_{rc}, indicating whether train $r \in R$ is assigned to circulation c. We let the parameter a_{rc} indicate whether such an assignment is possible. The formulation now reads as follows:

$$\sum_{c\in\mathcal{C}|l\in c} \frac{T}{t_c+\delta_c}\theta_c = f_l x_l \quad \forall l \in \mathcal{L}, \tag{5.20}$$

$$l_c\gamma_c \le \theta_c \le u_c\gamma_c \quad \forall c \in \mathcal{C} \tag{5.21}$$

$$\gamma_c \le x_l \quad \forall c \in C, \quad \forall l \in c, \tag{5.22}$$

$$\sum_{r\in R} a_{rc}v_{rc} = \theta_c \quad \forall c \in \mathcal{C}, \tag{5.23}$$

$$\sum_{c\in\mathcal{C}} v_{rc} \le 1 \quad \forall r \in R, \tag{5.24}$$

$$\theta_c \ge 0 \text{ and integer} \quad \forall c \in \mathcal{C}, \tag{5.25}$$

$$\gamma_c, v_{rc} \in \{0,1\} \quad \forall c \in \mathcal{C}, \quad \forall r \in R. \tag{5.26}$$

Constraints (5.20) guarantee that if a line is selected, a sufficient number of trains is assigned to circulations covering the line. The crucial observation is that every period, a train assigned to circulation c performs $\frac{T}{t_c+\delta_c}$ trips of each line in c. Constraints (5.21) assure that if a circulation is selected, the number of trains assigned to the circulation is between a certain upper and lower bound. These bounds are given by

$$l_c = \left\lceil \frac{t_c}{T} \right\rceil \quad \text{and} \quad u_c = \frac{t_c}{T} f_c. \tag{5.27}$$

The lower bound ensures that every selected circulation accounts for at least one train service of each line in every period. The upper bound is derived from constraints (5.20). Constraints (5.22) impose that a circulation can only be selected if all lines in the circulation are selected. These constraints are not necessary for a valid formulation, but slightly strengthen the linear programming relaxation. The assignment part of the formulation is covered by constraints (5.23) and (5.24). These constraints make sure that the number of trains assigned to a circulation equals the number of times the circulation is selected and that every train is assigned to at most one circulation.

To keep the number of circulations limited, we only allow fixed circulations and circulations with two lines that satisfy $\delta_c < \sum_{l \in c} \delta_l$, i.e. the downtime of the combined circulation is strictly smaller than the sum of the downtimes of the individual lines. Doing so ensures that the size of the model does not increase too much, while still including the most promising circulations.

5.4.3 Solving the Subproblem

The goal of the subproblem is to evaluate whether the line plan produced by the master problem admits a feasible timetable and to identify one or more violated inequalities or cuts if this is not the case. As explained in Section 5.2, it is not required to consider the complete timetabling problem since trains will be dispatched using self-organizing strategies rather than according to a timetable. Therefore, we do not try to compute a timetable for the entire network, but rather compute partial timetables for every station independently. Additional advantages of this approach are that it gives timetabling instances that can be solved quickly and that it allows us to identify small sets of inconsistent lines (and therefore strong cuts that can be added to the master problem). Conversely, computing a timetable for the entire network is very time consuming and has the disadvantage that only a single line plan is ruled out by the generated cut if the line plan is found to be infeasible.

We formalize the timetabling problem using the Periodic Event Scheduling Problem (PESP). The PESP is originally introduced by Serafini and Ukovich (1989) and can be used to model a wide variety of timetabling constraints. In the PESP we include safety, dwell, synchronizing and capacity constraints. The capacity constraints lie outside the standard formulation of the PESP but are necessary given the limited capacity of the boundary stations. We refer to the extended model as the C-PESP. A formulation of the C-PESP is provided in Appendix5.B. For a more extensive discussion of the (C-)PESP we refer to Peeters (2003).

In our implementation of the C-PESP, we increase the minimum dwell times by the headway time to strengthen the capacity constraints (if both the dwell time and headway time are 2 minutes, every dwelling train effectively blocks the platform for 4 minutes). In addition, we apply constraint propagation techniques before we split up the timetabling problem into the timetabling problems for all stations. This way, the local constraints at the stations are strengthened using constraints outside the station. Furthermore, junctions between stations and points in the network where the number of tracks changes are considered as dummy stations.

When we detect an infeasibility at a station, it is first checked whether the set of lines attending the station is a minimal inconsistent set by iteratively removing the lines in the set and checking whether the C-PESP now has become feasible. If removing one of the lines still results in an infeasible C-PESP, the line is removed from the set. This process is repeated until the set of lines corresponds to a minimal inconsistency.

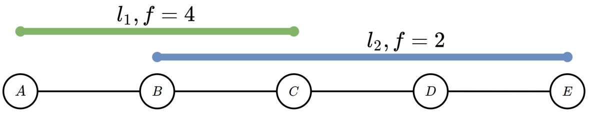
To give a mathematical formulation of the cut, let $\mathscr{L}^s$ denote a minimal inconsistent set of lines attending station s in the current solution of the master problem. Then, the following cut rules out this line combination in the next iterations of the master problem:

$$\sum_{l \in \mathscr{L}^s} x_l \leq |\mathscr{L}^s| - 1. \tag{5.28}$$

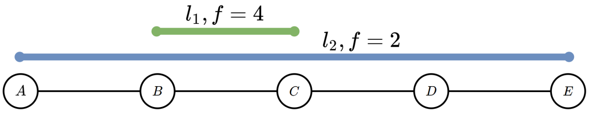
However, this is not the only cut we can derive from the discovered infeasibility, since we know all line plans that generate the same C-PESP at s, or a C-PESP with a smaller feasible region, must also result in an infeasibility. To illustrate this, consider the line plan visualized in Figure 5.6a, containing the lines l_1 and l_2 with frequencies 4 and 2, respectively. Now assume this line plan leads to an infeasible C-PESP at station C. After adding the corresponding cut to the master and resolving, we might find the line plan in Figure 5.6b in the next iteration. However, this line plan must also result in an infeasibility at station C, as from the perspective of this station, the solution has not changed (assuming that the minimum dwell time only depends on whether trains turn or not). In other words, the C-PESPs generated for station C are equivalent. Therefore, we could have excluded both line plans upon finding the infeasibility at C.

More generally, let $\mathcal{M}_i^s$ denote the set of lines that have the same frequency, the same in- and outbound edge and the same minimum dwell time at station s as the i'th line in $\mathscr{L}^s$. Then, from a station infeasibility at station s, we can derive the following cuts:

$$\sum_{l \in M} x_l \leq |\mathscr{L}^s| - 1, \quad \forall M \in \mathcal{M}_1^s \times \mathcal{M}_2^s \times ... \times \mathcal{M}_{|\mathscr{L}^s|}^s. \tag{5.29}$$



(a) Solution of the master problem.



(b) Alternative line plan.

Figure 5.6: Two line plans that generate the same C-PESP at station C.

The advantage of adding multiple cuts per iteration is that it is likely to reduce the total number of iterations before the C-PESPs at all stations are feasible. On the other hand, the time spent solving the master problem might increase, as adding all cuts (5.29) increases the size of the master problem, especially when the disrupted region and/or the set $\mathscr{L}^s$ is large. In preliminary experiments, we observed that the benefits of adding multiple cuts greatly outweigh the disadvantages. Therefore, we always add all cuts (5.29) when running the algorithm.

5.5 Operating the Disrupted Region

In this section, we propose dispatching strategies that can be used to operate the line plans that are produced by the model discussed in Section 5.4. In addition, we describe the simulation framework and evaluation criteria used to assess the performance of the line plans and dispatching strategies.

5.5.1 Train Dispatching Strategies

The train dispatching strategies that we develop specify what to do next when a train arrives at a station. Specifically, the strategies state (i) *when* the arrived train will depart and (ii) *where to* the train will depart. The information that is allowed to be used to make these decisions are previous departure times at the station and information from trains directly surrounding the station.

As for the *when* aspect of strategies, we consider three *timing principles*, referred to as ASAP, SYNC and SYNC + COOR. When trains are operated using the ASAP (as soon as possible) principle, arriving trains always leave as soon as possible. This may be reasonable, as the station capacity is limited, hence trains should not occupy platforms longer than necessary.

When trains are operated using the SYNC (synchronize) principle, trains do not depart as soon as possible at terminal stations. Instead, the departure time at these stations is decided based on the previous departure time of the line in order to promote the regularity of the departure times. For example, if a train from a certain line with a frequency of 4 per hour arrives at the terminal station and the previous train of the line departed 7 minutes ago, the train will depart in 8 minutes. However, we do impose that if a train is unable to enter a terminal station because of a waiting train, the waiting train will depart immediately and not wait until its desired departure time, to free a platform for the entering train.

The SYNC + COOR (synchronize and coordinate) principle extends the SYNC principle by coordinating the departure times of different types of trains. Firstly, the principle imposes that if a regional train has departed from a station on a part of the network that has one track per direction, intercity trains can only depart when enough time has passed to make sure the faster intercity train does not have to wait for the slower regional train. Secondly, at stations where overtaking is possible, regional trains wait at the station if an intercity train is coming within 3 minutes and the regional train would have otherwise blocked the incoming intercity train. Note that it is also possible to take a different value than 3 minutes or let the maximum waiting time depend on the decrease in travel time of the intercity train.

As for the *where to* aspect of dispatching strategies, we consider two *turning principles*, STAT and DYN. In the STAT (static) principle, trains reaching their terminal station start performing the same line in the reverse direction. In the DYN (dynamic) principle, trains can be reassigned to a different line when they reach their terminal station. Trains are reassigned based on the type of the train and the previous departure times. The line that needs a departure the earliest gets assigned the first compatible train. The advantage of the DYN principle is that it results in shorter turning times at the terminal stations. Even more, this principle leads to more efficient use of the trains, such that it is possible to operate more trains per hour. Clearly, the STAT and DYN principles are related to the respectively fixed and flexible rolling stock constraints discussed in Section 5.4.2. If the line plan is optimized using flexible rolling stock constraints, it is expected that the STAT principle is too restrictive and will not lead to satisfactory operations. Conversely, if fixed rolling stock circulations are imposed in the line planning algorithm, it might still be beneficial to use the DYN principle, as it is more flexible.

The three timing principles and two turning principles give rise to six different strategies, with ranging degrees of coordination. For instances without intercity trains we have only four strategies, as the SYNC+COOR principle is only different from the SYNC principle in the way it deals with intercity trains.

5.5.2 Simulation Framework

We evaluate the line plans and dispatching strategies by simulating the railway traffic in the disrupted region. The simulation takes as input a disrupted region, the modified line plan, the regular timetable, the dispatching strategy that should be applied and the time instant the disrupted region is decoupled from the rest of the network. Using these inputs we can retrieve the current position of the trains in the disrupted region from the regular timetable. The simulation is initialized by assigning the trains to lines. We use a simple model to perform this initialization, assigning the trains in such a way that trains assigned to the same line are roughly spread out over the network. In the case that a train not assigned to any line, it is assumed that this train can be routed to a shunt yard without interfering with other trains.

For the simulation we use a macroscopic representation of the railway network where nodes are stations and edges are tracks. Junctions are modeled as dummy stations with the number of tracks as the number of platforms. In the actual operations, tracks are subdivided into block sections and a train is allowed to enter a section if the previous train is no longer occupying the section. Otherwise, the train needs to wait before the red signal placed at the beginning of the block. In the simulation, the block sections are not taken into account and trains therefore only wait upon arriving at or departing from a (dummy) station. Between stations, trains run at a constant speed. Dummy stations are also introduced between stations to prevent faster trains from overtaking slower trains on the same track.

Trains can enter a station if there is a platform available that is compatible with the in- and outgoing track of the train. A platform is available if the predefined headway time has passed since the previous train used the platform. If multiple platforms are available, the simulation picks one at random. If there is no available platform, the train is added to the arrival queue at the station. When a train arrives at a station, the departure is scheduled according to the dispatching strategy that is being used. When a train wants to depart, the simulation checks whether the outgoing track is available for the type of the train. The outgoing track is available if the headway time has passed since the last train of the same type has departed. If the outgoing track is still blocked, the train is added to the departure queue of the track. After a train has departed, the simulation checks whether the newly available platform can be used by one of the trains in the arrival queue and an arrival is scheduled if that is the case.

5.5.3 Evaluation Criteria

As the main objective when optimizing the line plan is to offer sufficient travel options to the passengers, it is expected that a good execution of the line plan also leads to a good service for the passengers. Therefore, we introduce three *operational measures* that quantify how close the realized timetable is to a perfect execution of the line plan. The considered aspects in these measures are the realized frequencies of lines, the regularity of the inter-departure times and the train delays. Next to the operational measures, we also directly assess the impact of the different line plans and strategies on the journeys of passengers with two *travel measures*, taking into account the realized number of travel options per hour offered to passengers and travel times. The exact definitions of the measures can be found in Appendix 5.C. Below, we describe their interpretation.

Operational measures. The operational measures are defined for the operation of a line in a certain direction. Later we describe how the measures can be evaluated for an entire line plan. The measures are defined in such a way that if and only if a train line is operated perfectly (i.e. trip times are at their minimum and the trains depart from the terminal stations according to a perfect synchronized pattern), the train line scores exactly 1 for all measures at all times. This allows us to clearly observe deviations from the ideal scenario.

The *frequency* measure relates the number of realized departures to the number of departures there should have been according to the line plan. For example, if the measure for a certain line is 1.1, there have been 10 percent more departures than indicated by the line plan and if the measure is 0.9, there have been 10 percent fewer departures. The *regularity measure* relates the cumulated relative deviation from the ideal interdeparture time to the number of departures. For example, for a line with a frequency of 4 per hour, if the measure equals 0.8, the average absolute deviation from the ideal time between departures equals $0.2 \times 15 = 3$ minutes. If the measure equals 0.6, the average absolute deviation is 6 minutes. Relative deviations are considered such that a deviation of 10 minutes on a line with a period of 30 minutes is penalized less than a deviation of 10 minutes on a line with a period of 15 minutes. Finally, the *delay measure* relates the realized trip time to the minimum trip time of the line. For example, if the minimum trip time of a line is 30 minutes, and the measure equals 1.1, the average realized trip time is $30 \times 1.1 = 33$ minutes. If the measure is 1.2, the average realized trip time is 36 minutes.

The frequency, regularity and delay measure of an entire line plan at any given t can be computed by taking the average of the measures over all lines and directions. Note that averaging over lines can result in a frequency measure of 1.0, even though the performance of the individual lines may be very bad, for example one line has frequency 1.5 and another line 0.5. However, in such a case we are still able to detect that the line plan has poor performance as both lines will have very bad scores on the regularity measure, which averages absolute deviations from the intended interdeparture times.

Travel measures. The first travel measure corresponds to an empirical analog of the objective used in the line planning algorithm and hence referred to as the realized objective. To enable a proper comparison to the objective attained by the line plans, we assume travelers take the same path as the path that is used in optimizing the line plan. For the realized objective at time t, we assume that at that time $n_{o,d}$ passengers want to travel from o to d, and find the number of travel options over path $\rho_{o,d}$ in the next hour according to the simulation. To evaluate the long run average performance, we compute the measure for every whole minute during the simulation and take the average. By comparing this measure to the theoretical objective of the line plan, we can assess to what extent dispatching strategies result in the same number of travel options per period as intended in the line plan.

The second travel measure equals the (weighted) average travel time of travelers, including waiting, in-vehicle and transfer times. For this measure, we assume passengers travel according to a shortest path that is computed using the modified line plan. To evaluate the long run average performance, we again compute the measure for every whole minute during the simulation and take the average. Besides using this measure to compare the performance of different line plans and dispatching strategies, it also gives an indication of the overall performance of our approach by comparing it with the average travel time in an undisrupted situation.

5.6 Computational Results

To test our approach, we conduct experiments on multiple parts of the network operated by NS. Disrupted regions of different sizes are considered, such that we can examine the practicability of the algorithm in different scenarios. Furthermore, we also investigate multiple variants of the rolling stock and timetabling constraints and analyze the effect these have on the overall performance.

5.6.1 Problem Instances

We test the developed algorithms on two disrupted regions, using the 2017 line plan and timetable. The original line plan of the part of the Dutch railway network that we consider is presented in Figure 5.7. NS operates a dense line system in this part of the Netherlands, with six intercity lines and nine regional lines. Figure 5.7 also indicates which stations serve as decoupling stations, as specified by NS and ProRail.

The largest part of the considered railway network is double-tracked. To accommodate for higher frequencies of intercity and regional trains, there are four tracks between Utrecht Centraal and Amsterdam-Zuid and between Utrecht Centraal and Utrecht Overvecht. Furthermore, intercity trains can overtake regional trains at Amsterdam Muiderpoort, Weesp and Naarden-Bussum. The part between Baarn and Den Dolder is single-tracked, with a passing possibility at Soest.

For the passenger data we use an origin-destination matrix with the average daily number of passengers between stations provided by NS. As the network that we consider is only a part of the Dutch railway network, there are many passengers that travel through the considered network without having both the origin and destination in this network. This is taken into account by including the passenger counts from and to major intercity stations outside this part of the network. If possible, we derive the number of available platforms at boundary stations for serving the disrupted region from available contingency plans used by NS and ProRail. If no applicable contingency plan is available, we estimate the number of available platforms.

The first disrupted region we consider is the region bounded by Utrecht Centraal,

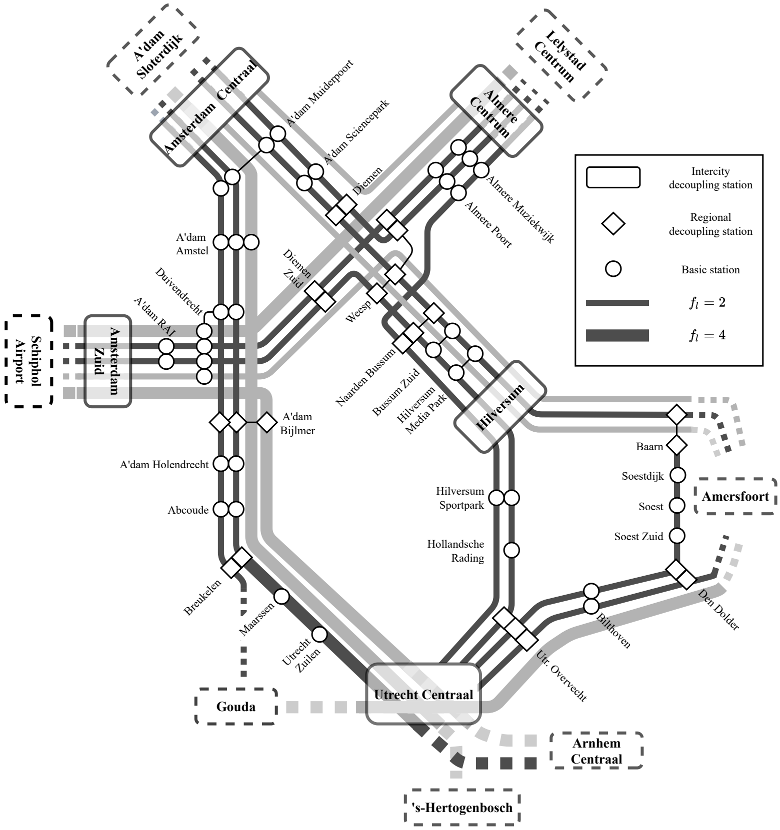


Figure 5.7: The regular line plan in a part of the Dutch railway network. Line frequencies are indicated by the thickness of the lines. Dark lines are regional lines, bright lines are intercity lines.

Den Dolder, Baarn and Hilversum. An out-of-control situation in this region could occur due to a power outage at Amersfoort, directly impacting five of the eight lines in this region. We assume buses are transporting passengers from Baarn and Den Dolder to Amersfoort, and vice versa. An interesting aspect of this disrupted region is that while the most important lines (in terms of the number of passengers) in this region are the intercity lines between Hilversum and Amersfoort and between Utrecht Centraal and Amersfoort, it is not possible to operate any intercity trains in the disrupted region. The reason for this is that Baarn and Den Dolder are regional decoupling stations, such that intercity trains cannot turn at these stations. Hence, the challenge is to optimally use the available regional trains in order to also serve the passengers that normally take intercity trains. The large instance that is considered is the region bounded by Amsterdam Centraal, Amsterdam Zuid, Almere Centrum, Utrecht Centraal, Den Dolder, Baarn and Hilversum. An out-of-control situation in such a large part of the Dutch railway network could occur after a combination of major disruptions at Amsterdam and Utrecht. In this instance it is possible to plan both regional and intercity lines. Besides its size, an interesting aspect of this instance is that there is limited capacity for turning trains at Amsterdam Zuid, Utrecht Centraal and Almere Centrum. Given that a large number of trains attend these stations in the regular line plan, the limited station capacity is most likely one of the bottlenecks in the large instance.

5.6.2 Parameter Settings and Experimental Setup

In our experiments, we generate the line pool $\mathcal{L}$ by considering all paths that either constitute a shortest path between the terminal stations (shortest in terms of the running times) or are a path or sub-path of a line in the regular line plan. For a path π_l we consider all lines (π_l, f_l) as long as the frequency f_l is a positive integer smaller or equal to $\min_{e \in \pi_l} g_e$, since we do not allow higher frequencies on an edge than the original edge-frequencies. This way, the size of the master problem remains sufficiently small, such that we can find good solutions in a short time.

The running times between stations are derived from the regular timetable of NS. For the dwell times, we use 5 minutes if the train turns and 2 minutes otherwise. We use a headway time of 2 minutes. Hence, the blocking time of a line at a station is 7 minutes if the line has a turn at the station and 4 minutes otherwise. The parameter T, the period of the timetable is set to 60 minutes, the cycle time used by NS.

We impose the timetabling constraints (5.10-5.14) in their most stringent form. That is, we set the maximum additional dwell time $\delta_{ls}^{\max}$ of line l at station s to 0 minutes unless s is a terminal of line l, in which case we set $\delta_{ls}^{\max}$ to 60 minutes. In other words, we impose that the downtime of every line must be entirely spent at the terminal stations. This way we avoid finding line plans that exhaust all capacity at terminal stations, causing trains to queue up in front of these stations when operating the line plan without a timetable.

After initial experiments we decided to set the weights in the objective function as follows. The first term, representing frequency decrease of travel options is given weight $w_1 = \frac{1}{\sum_{(o,d) \in \mathcal{OD}} n_{o,d}}$, i.e. we normalize by dividing by the number of travelers.

The second term, representing the decrease of direct travel options is given weight $w_2 = \frac{1}{2\sum_{(o,d)\in\mathcal{OD}_{\text{dir}}} n_{o,d}}$, i.e. we divide by two times the number of travelers with a direct travel option. The last term, the line cost, is given weight $w_3 = 0.001$. These settings ensure that the most emphasis is put on providing all passengers with sufficient travel options ("getting everyone home"), and that direct connections are maintained unless this strongly harms the first objective. Ties are broken based on the number of lines.

For a thorough analysis of the performance of the line planning algorithm, we perform tests with three variants of the rolling stock constraints. In the first setting, no rolling stock constraints are included. This setting serves as a reference and is used to illustrate which line plan would be possible if ample rolling stock would be present. In the two other settings, *fixed* and *flexible* rolling stock constraints are included, respectively.

The algorithm takes as input the trains that are present in the disrupted region when the region is decoupled. Given this time instant, the positions of trains can be derived from the timetable. Since this of course varies over the hour, we run the algorithm for five equidistant time instants (minutes 1, 13, 25, 37 and 49). As NS operates most lines every 15 or 30 minutes, this approach ensures that the majority of the variation in the number of available trains and their positions is captured. When determining the performance of the different dispatching strategies on the line plans, we use a simulated time of 4 hours and average the measures over the five time instants.

The line planning algorithm and the simulation are implemented in Java 8 on a Dell Precision 7520 running Windows 10 with an Intel Core i7-7820HQ processor at 2.9 GHz and 16 GB of RAM. CPLEX 12.8.0 is used to solve the mixed-integer programs. In every iteration of the line planning algorithm, we let the master problem terminate either when the optimal solution is found, or when a time limit of 120 seconds is reached.

5.6.3 Results Small Disrupted Region

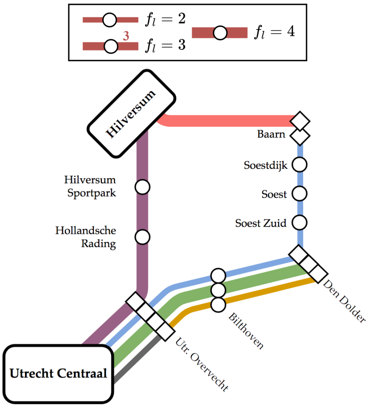
In Table 5.1, the objective, number of lines, number of iterations, number of cuts and computation times are presented for the three settings. The results of the *fixed* setting and *flexible* setting depend on the time instant, hence for these settings we present the averages over the 5 time instants. It can be seen that without any rolling stock constraints we find a line plan with an objective very close to 0, indicating that hardly any travelers have a reduction in their indirect or direct OD-frequency. However, as rolling stock constraints are neglected, it turns out that this line plan requires 14 trains whereas only 6 or 7 trains (depending on the exact time the disrupted region is decoupled) are available. The line plans obtained with the *fixed* and *flexible* rolling stock constraints can be operated with the available rolling stock, but this of course results in a worse objective. The *flexible* line plans achieve a much better objective than the *fixed* line plans.

With regards to the computational performance, it can be seen that all solutions are found within a few seconds. The solution with the *basic* setting is found after 2 iterations of the line planning algorithm, in which 32 cuts are added to the master problem. For the *fixed* setting and *flexible* setting, the number of iterations varies between 1 and 2, and fewer violated inequalities are encountered. This shows that in this disrupted region, the amount of rolling stock available is the main limiting factor for designing an attractive line plan.

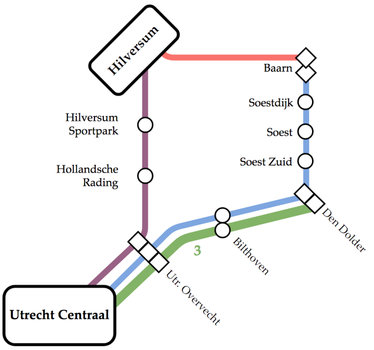
Table 5.1: Results of the line planning algorithm on the small disrupted region using the three types of rolling stock (RS) constraints.

RS constraints	Obj.	Lines	Iterations	Cuts	Master CPU (s)	Total CPU (s)	Max Total CPU (s)
None	0.033	6	4	32	0.5	2.7	2.7
Fixed	0.447	3.8	1.2	0.4	0.1	0.4	1.1
Flexible	0.351	4	1.6	0.8	0.7	1.1	2.4

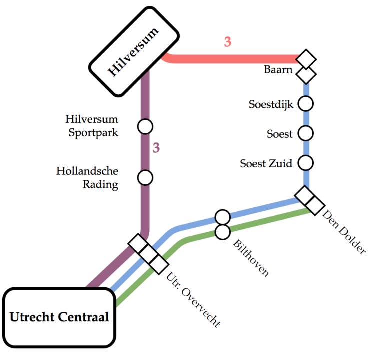
To illustrate the impact of the different settings on the solution, the line plans for one of the time instants are visualized in Figure 5.8. As can be seen, the line plan obtained without rolling stock constraints is much denser than the *fixed* and *flexible* line plans, once again showing that not taking into account rolling stock restrictions results in line plans that cannot be operated with the available rolling stock. The *fixed* and *flexible* line plans have the same lines, but with different frequencies. By combining lines into circulations, the frequency between Utrecht Centraal and Hilversum and between Baarn and Hilversum can be increased from 2 to 3. The frequency of the line between Utrecht Centraal and Den Dolder however is decreased from 3 to 2. Overall, this results in a theoretically better solution.



(a) No RS constraints

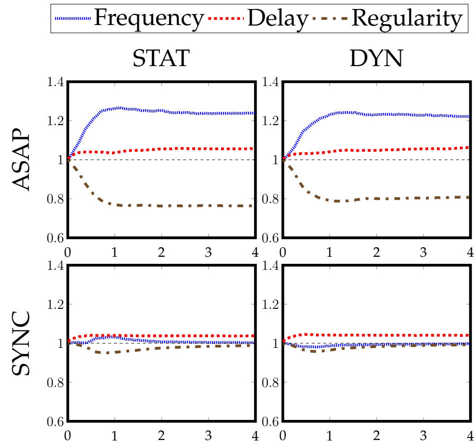


(b) Fixed RS constraints

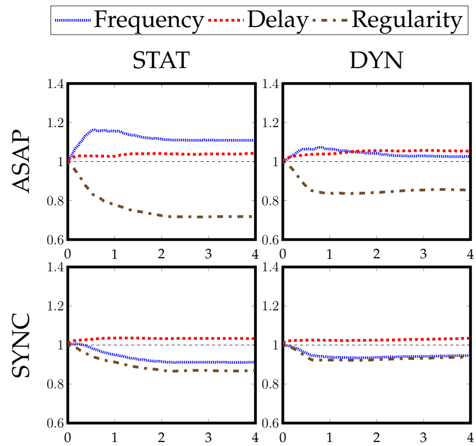


(c) Flexible RS constraints

Figure 5.8: Line plans for the small disrupted region obtained with different rolling stock (RS) constraints included. All lines are regional lines.



(a) Fixed RS constraints



(b) Flexible RS constraints

Figure 5.9: Operational measures in the small disrupted region obtained with (a) fixed and (b) flexible rolling stock (RS) constraints. The four strategies are structured in a 2x2 matrix. In the figures, the horizontal axis denotes the time in hours and the vertical axis the score on the three measures. The closer a measure is to the horizontal dashed line, the better the performance.

The simulation results are presented in Table 5.2 and Figure 5.9. Table 5.2 contains the travel measures, averaged over the entire simulation and Figure 5.9 shows the values for the operational measures plotted over time. As this instance only contains regional lines, the SYNC+COOR principle is not considered here. The *fixed* setting, which theoretically attains an objective of 0.447, results in realized objectives of about 0.39 and 0.48 for strategies with the ASAP and SYNC timing principle, respectively. This can be explained using Figure 5.9(a), where we observe that the ASAP strategies have normalized frequencies that are much larger than 1, whereas the SYNC strategies have normalized frequencies very close to 1. This means that the ASAP strategies lead to 'more trains than promised' and consequently, a better objective than promised. The ASAP strategies also result in shorter travel times. The turning principle has a much smaller effect on the performance of the *fixed* line plans. The DYN strategies have slightly shorter travel times.

When the *flexible* setting is used, the turning principle now plays a more important role in the performance. The DYN principle clearly outperforms the STAT principle when the SYNC timing principle is used, resulting in both a lower objective and shorter travel times. This was expected, since the *flexible* setting generates line plans that cannot be operated if only fixed turnings are allowed. As with the *fixed* setting, the strategies with the ASAP principle attain a better realized objective, although the theoretical objective of 0.351 is not reached.

Table 5.2: Values of the travel measures for the different strategies and rolling stock (RS) constraints in the small disrupted region, averaged over the five time instants. In the undisrupted situation, the average travel time (without waiting time) is about 18 minutes.

	Fixed RS constraints		Flexible RS constraints	
	Real. Obj.	Travel Time (min.)	Real. Obj.	Travel Time (min.)
ASAP-STAT	0.389	27.6	0.355	27.5
SYNC-STAT	0.478	30.6	0.434	29.6
ASAP-DYN	0.386	27.4	0.363	27.7
SYNC-DYN	0.486	30.3	0.396	28.5

Comparing the results for the line planning settings, we can observe that for any dispatching strategy, the *flexible* setting has a better realized objective, and that for both line planning settings, the objective and travel times are lowest using the ASAP strategies. On the other hand, Figure 5.9 shows that the SYNC strategies generate a more stable and predictable service, which can also be deemed important when dealing with an out-of-control situation.

5.6.4 Large Disrupted Region

In Table 5.3, the objective, number of lines, number of required trains under fixed circulations, number of iterations, number of cuts and computation times are presented for the three settings. The results of the *fixed* and *flexible* constraints are again averages over the 5 time instants. As in the small disrupted region, neglecting rolling stock constraints leads to a line plan that achieves a very low objective, but is actually infeasible. However, the difference is smaller than in the small region, showing that next to the number of available trains, the available infrastructure also poses a large restriction on the line plans that can be realized in the large disrupted region.

The computation time of the algorithm strongly depends on the setting. Using the *fixed* setting, the optimal solution is found on average within 10 seconds. The *flexible* setting takes considerably more time, on average about 40 seconds. The *basic* setting even takes more than 13 minutes, but this setting is unlikely to be used in real-time applications as it produces overly optimistic line plans. Both the *fixed* and *flexible* setting require on average 2 iterations during which 160 cuts are added. In fact, it turns out that the same 160 cuts are added for both settings.

The line plans for one of the time instants are visualized in Figure 5.10. The line plan without rolling stock constraints is similar to the regular line plan that is operated. The frequency of regional trains is increased between Hilversum and Baarn and between Utrecht Centraal and Den Dolder to compensate for the intercity trains that cannot run between these stations. As for the intercity lines, the limited turning capacity at Amsterdam Zuid, Utrecht and Almere Centrum leads to the cancellation of the line between Amsterdam Zuid and Utrecht Centraal and the reduction of the frequency of the line between Amsterdam Zuid and Almere Centrum from 4 to 2 per hour. In the *fixed* line plan, many regional lines are canceled or have their frequency reduced, e.g. between Amsterdam Zuid and Almere Centrum and between Amsterdam Centraal and Utrecht Centraal. The frequency of the intercity between Amsterdam Zuid and Hilversum is also reduced. On the other hand, the intercity between Amsterdam Zuid and Almere Centrum has frequency 4 compared to 2 in the basic line plan, and the intercity between Amsterdam Zuid and Utrecht Centraal, which was canceled entirely in the *basic* line plan, is included in the *fixed* line plan, albeit with frequency 1. In the *flexible* line plan, many improvements are visible over the *fixed* line plan. More and longer regional lines are operated, restoring direct connections between Utrecht Centraal and Baarn and Amsterdam Zuid and Almere.

Furthermore, the intercity between Amsterdam Zuid and Utrecht Centraal is included with frequency 2, compared to 1 in the *fixed* line plan.

Table 5.3: Results of the line planning algorithm on the large disrupted region using the three types of rolling stock (RS) constraints.

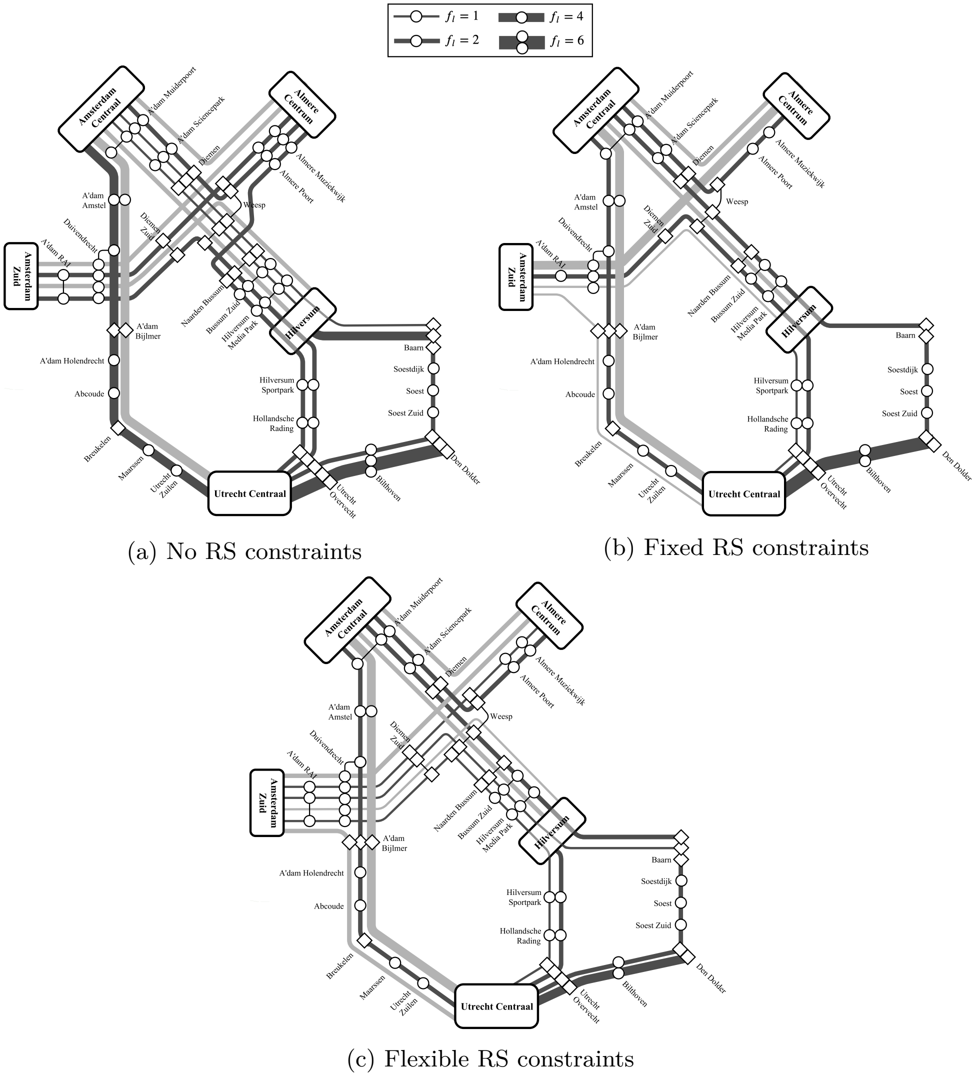
RS constraints	Obj.	Lines	Iterations	Cuts	Master CPU (s)	Total CPU (s)	Max Total CPU (s)
None	0.074	15	6	408	802.6	822.1	822.1
Fixed	0.162	13.0	2.0	160.0	5.2	9.2	15.2
Flexible	0.133	17.0	2.0	160.0	36.7	40.5	44.9

Table 5.4: Values of the travel measures for the different strategies and rolling stock (RS) constraints in the large disrupted region, averaged over the five time instants. In the undisrupted situation, the average travel time (without waiting time) is about 25 minutes.

	Fixed RS constraints		Flexible RS constraints	
	Real. Obj.	Travel Time (min.)	Real. Obj.	Travel Time (min.)
ASAP-STAT	0.226	37.5	0.217	37.4
SYNC-STAT	0.224	36.1	0.244	37.9
SYNC+COOR-STAT	0.219	35.6	0.248	37.5
ASAP-DYN	0.197	35.8	0.195	36.5
SYNC-DYN	0.220	36.0	0.207	36.2
SYNC+COOR-DYN	0.219	35.9	0.217	36.3

The simulation results of the large disrupted region are presented in Table 5.4 and Figure 5.11. Table 5.4 contains the travel measures, averaged over the entire simulation and Figure 5.11 shows the values for the operational measures plotted over time. The *flexible* rolling stock constraints in combination with the ASAP-DYN strategy results in the best realized objective. However, note that in contrast to the small instance, the realized objectives are worse than the theoretical objectives that are attained. The figures with the operational measures provide an explanation. It is visible that the delay measure is over 1.1 for all line planning settings and dispatching strategies, which is simply caused by the larger size of the network, with more potential places for conflicts between trains. This implies that the actual travel times are about 10 percent larger than travel times used in optimizing the line plans. As a consequence, the realized frequencies are below the 'promised' frequencies, causing the realized objective to be worse. These results suggest that for larger disrupted regions, it might be worthwhile to use more conservative trip time estimates in the line planning algorithm.

In addition, it is visible that the large difference in the attained theoretical objective between the *fixed* and *flexible* line plans does not translate in the same difference in realized objective. On the other hand, the dispatching strategies that apply the DYN principle, which allows flexible turnings during operations, do achieve significantly better objectives, travel times and also score better on the operational measures. This shows that flexible turnings partially compensate for the longer than anticipated travel times.

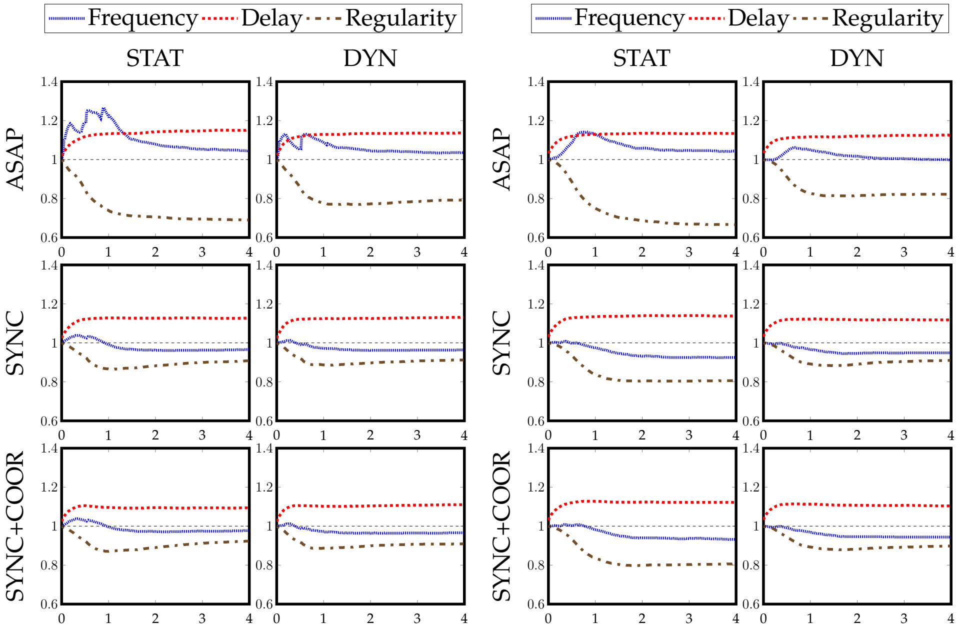


(a) No RS constraints

(b) Fixed RS constraints

(c) Flexible RS constraints

Figure 5.10: Line plans for the large disrupted region obtained with the different rolling stock (RS) constraints. Dark lines are regional lines, bright lines are intercity lines.



(a) Fixed RS constraints

(b) Flexible RS constraints

Figure 5.11: Operational measures in the large disrupted region obtained with (a) fixed and (b) flexible rolling stock (RS) constraints. The six strategies are structured in a 3x2 matrix. In the figures, the horizontal axis denotes the time in hours and the vertical axis the score on the three measures. The closer a measure is to the horizontal dashed line, the better the performance.

A last observation regarding Table 5.4 and Figure 5.11 concerns the SYNC+COOR principle. When considering the operational measures, the SYNC+COOR principle leads to an average delay of about 2 percent lower compared to the SYNC principle, which also results in a lower travel time for passengers in all cases except one. In fact, if combined with the *fixed* setting, the SYNC+COOR-STAT strategy results in the lowest travel time over all strategies, illustrating that adding small coordination mechanisms can greatly improve the experience of passengers. However, this does increase the complexity of the operations, potentially harming the controllability of the system, which conflicts with the original goal of preventing out-of-control situations.

5.6.5 Impact of the Timetabling Constraints

As a final experiment, we examine the timetabling constraints have on the line plans and resulting performance of the dispatching strategies. To do so, we define three variants of the model. In the *no timetabling constraints* variant, we disregard the capacity entirely, so we omit the subproblem and constraints (5.10-5.14). In the *basic* timetabling constraints variant, we still omit the subproblem but include the basic timetabling constraints (5.11-5.14) in the master problem. The *advanced* timetabling constraints variant represents the original model, including the subproblem and all timetabling constraints in the master problem. As we are also interested in the impact of the timetabling constraints in case the rolling stock is not a limiting factor, we also run the analysis for the case where there are ample trains in the disrupted region. We only consider the large disrupted region in these experiments.

Table 5.5: Values of the performance measures in the large disrupted region for the three variants of the timetabling constraints. With ample rolling stock and no timetabling constraints, the original line plan is returned by the line planning algorithm.

(a) Average performance of dispatching strategies with the ASAP principle

Rolling stock setting	Timetabling constraints	Frequency	Regularity	Delay	Real. obj.	Travel time (min.)
Fixed	None	0.98	0.66	1.35	0.217	37.7
	Basic	1.04	0.74	1.15	0.197	36.5
	Advanced	1.04	0.74	1.14	0.205	36.1
Flexible	None	0.94	0.64	1.34	0.227	39.2
	Basic	1.00	0.75	1.14	0.207	37.0
	Advanced	1.02	0.74	1.13	0.198	36.4
Ample	None	0.89	0.58	1.54	0.179	41.5
	Basic	1.03	0.77	1.18	0.119	36.0
	Advanced	1.02	0.75	1.16	0.118	34.3

(b) Average performance of dispatching strategies with the SYNC principle

Rolling stock setting	Timetabling constraints	Frequency	Regularity	Delay	Real. obj.	Travel time (min.)
Fixed	None	0.93	0.85	1.26	0.223	37.5
	Basic	0.98	0.90	1.12	0.211	35.9
	Advanced	0.98	0.90	1.11	0.220	35.9
Flexible	None	0.91	0.81	1.27	0.240	38.4
	Basic	0.95	0.87	1.12	0.224	37.2
	Advanced	0.96	0.87	1.12	0.229	37.0
Ample	None	0.88	0.76	1.42	0.150	39.4
	Basic	0.97	0.89	1.16	0.122	34.3
	Advanced	0.97	0.91	1.13	0.121	33.7

In Table 5.5, the values of the performance measures obtained with the different timetabling constraints are presented. As can be seen, disregarding timetabling altogether leads to very poor performance. This setting generates line plans that cannot be operated on the available infrastructure, leading to delays of around 30 percent with the actual number of trains and fixed or flexible rolling stock constraints, and around 50 percent for the case where there are ample trains. As expected, this also leads to noticeably longer average travel times for passengers. With basic timetabling constraints, the average delays are already strongly reduced to between 10 and 20 percent, translating in lower travel times. Adding the advanced timetabling constraints on top of the basic capacity constraints has a smaller effect, but still tends to decrease the average delays with 1 up to 3 percentage points. Especially when there are ample trains, this results in considerable decreases in average travel time. In some cases, the advanced timetabling constraints do worsen the realized objective. However, the realized objective is only based on the number of possible travel options passenger have and not on the trip duration. Hence, in these cases passengers have slightly fewer travel options per hour, but do on average still arrive earlier at their destination due to shorter delays of the trains.

5.7 Conclusion

In this chapter, we addressed new disruption management strategies for out-of-control situations occurring in railway systems. We developed a novel algorithm for redesigning the line plan, such that the railway system within a disrupted region can be operated using self-organizing principles. In order to ensure that the resulting line plan is feasible with respect to the available railway infrastructure and rolling stock, the proposed algorithm partially integrates line planning with timetabling and rolling stock scheduling using the integer version of Benders' decomposition. Besides investigating which lines should be operated when the system gets out-of-control, we also analyzed how the adapted line system should be operated. To this end, we developed several dispatching strategies that only require local coordination.

Computational experiments on the Dutch railway network indicate that the algorithm performs well. Optimal alternative line plans are provided in short amounts of time, making the approach applicable for use in practice. Using simulation, we also demonstrated that by applying the appropriate dispatching strategies, the produced line plans can be operated smoothly without relying on central coordination. The results show that allowing flexible turnings leads both to better line plans and to a better performance during operations, resulting in more travel options for passengers and shorter travel times. In addition, we observed that adding simple coordination mechanisms between slower and faster trains also significantly improves the experience of passengers, at the cost of increased complexity of the operations. All in all, our work certainly highlights the opportunity to offer limited service in out-of-control situations, where the current practice leads to termination of all traffic within the affected region.

As a next step, it is interesting to also consider train drivers and conductors when operating the disrupted region using self-organizing principles. In addition, we see an application of the proposed decomposition approach to integrated models for strategic railway planning.

Appendix

5.A Linearizing a Quadratic Program

Here we describe how to linearize a quadratic program of the type

$$\text{minimize} \quad \sum_i \left(\frac{z_i}{g_i}\right)^2 \tag{5.30}$$

$$\text{s.t.} \quad \sum_i a_{ij} z_i \leq b_j, \qquad \forall j, \tag{5.31}$$

$$0 \leq z_i \leq g_i, \quad z_i \text{ integer}, \quad \forall i. \tag{5.32}$$

To this end, we introduce the auxiliary binary variables $u_{i1}, u_{i2}, ..., u_{ig_i}$, add the constraints

$$z_i = \sum_{k=1}^{g_i} u_{ik}, \quad \forall i, \tag{5.33}$$

$$u_{ik} \in \{0, 1\}, \quad \forall i, \forall k, \tag{5.34}$$

and replace the objective by

$$\text{minimize} \quad \sum_i \sum_{k=1}^{g_i} c_{ik} u_{ik} \tag{5.35}$$

where the cost coefficients c_{ik} are defined as follows:

$$c_{ik} = \left(\frac{k}{g_i}\right)^2 - \left(\frac{k-1}{g_i}\right)^2. \tag{5.36}$$

Clearly the original formulation has a feasible solution if and only if the linearized formulation has a feasible solution. Furthermore, note that the cost coefficients are increasing in k. Hence, if $z_i = m$, it is optimal to set $u_{e1} = u_{e2} = ... = u_{em} = 1$ and $u_{e(m+1)} = u_{e(m+2)} = ... = u_{eg_e} = 0$. This results in an objective contribution of $\sum_{k=1}^m c_{ik} = \sum_{k=1}^m \left(\frac{k}{g_i}\right)^2 - \left(\frac{k-1}{g_i}\right)^2 = \left(\frac{m}{g_i}\right)^2 - \left(\frac{m-1}{g_i}\right)^2 + \left(\frac{m-1}{g_i}\right)^2 - ... - \left(\frac{1}{g_i}\right)^2 + \left(\frac{1}{g_i}\right)^2 = \left(\frac{m}{g_i}\right)^2$. In terms of the original objective, the contribution equals $\left(\frac{z_i}{g_i}\right)^2 = \left(\frac{m}{g_i}\right)^2$. As the contribution to the objective value is the same for both formulations, they are equivalent.

5.B Formulation of the C-PESP

The PESP can be formulated concisely using an event-activity network $\mathcal{N} = (\mathcal{E}, \mathcal{A})$. The arrival of the i'th train of line l at station s is represented by an arrival node $(l, s, \text{arr}, i) \in \mathcal{E}^{\text{arr}}$. Similarly, departures are represented by nodes $(l, s, \text{dep}, i) \in \mathcal{E}^{\text{dep}}$. In the basic form, there are four types of activities (arcs) linking two nodes:

- Dwelling activities link arrival nodes (l, s, arr, i) to departure nodes (l, s, dep, i).
- Driving activities link the departure node (l, s_1, dep, i) at station s_1 to the arrival node (l, s_2, arr, i) at the next station s_2.
- Safety activities link departure nodes (l, s, dep, i) or arrival nodes (l, s, arr, i) with departure nodes (l', s, dep, j) or arrival nodes (l', s, arr, j).
- Synchronizing activities link departure nodes (l, s, dep, i) with departure nodes (l, s, dep, j), with $i \neq j$.

Every activity corresponds to a constraint stating that the duration of the activity should be in a certain interval. For example, the safety activities correspond to constraints stating that the time between two trains using the same piece of infrastructure should be at least the headway time.

We let $\pi_i \in [0, T-1]$ denote the decision variable representing the time instant assigned to node i. The periodic constraint for an activity $(i, j) \in \mathcal{A}$ is then given by

$$l_{ij} \leq \pi_j - \pi_i + Tp_{ij} \leq u_{ij}, \quad p_{ij} \in \{0, 1\}, \tag{5.37}$$

where l_{ij} is the lower bound of the duration of activity (i, j) and u_{ij} the upper bound. The decision variables p_{ij} are introduced to compute the duration of activities correctly when $\pi_j < \pi_i$ and referred to as the *modulo* parameters.

Station Capacity in the PESP
A station capacity constraint can be formulated using the following interpretation of the modulo parameters:

$$p_{ij} = \begin{cases} 1, & \text{if event } j \text{ takes place before event } i, \\ 0, & \text{if event } j \text{ takes place after or at the same time as event } i. \end{cases}$$

Here, *before* or *after* refers to the sequence of the events on the linear axis $[0, T-1]$. Now assume the event-activity network of a station $\mathcal{N}_s = (\mathcal{E}_s, \mathcal{A}_s)$ contains all arcs

$$\begin{aligned}&(i,j), \text{ where } i \in \mathcal{E}_s^{\text{arr}} \text{ and } j \in \mathcal{E}_s^{\text{dep}},\\&(i,j), \text{ where } i \in \mathcal{E}_s^{\text{arr}} \text{ and } j \in \mathcal{E}_s^{\text{arr}} \text{ and } i < j.\end{aligned}$$

Let $\mathcal{A}_s^{\text{dwell}}$ denote the set of dwell activities at station s, let $\mathcal{A}_i^a$ denote the set of outgoing activities to arrivals from event i and let $\mathcal{A}_i^d$ denote the set of outgoing activities to departures from event i. Then, we can enforce that the capacity of station s is never violated by adding the following constraints:

$$1 + \sum_{(k,l)\in\mathcal{A}_s^{\text{dwell}}} p_{kl} + \sum_{(j,i)\in\mathcal{A}_j^a} (1-p_{ji}) + \sum_{(i,j)\in\mathcal{A}_i^a} p_{ij} - \sum_{(i,j)\in\mathcal{A}_i^d} p_{ij} \le |P^s| \quad \forall i \in \mathcal{A}_s^{\text{arr}}. \tag{5.38}$$

Usually, the platforms at a station are subdivided into groups that are assigned to the different lines and directions. In such cases, the capacity constraint can be included for every group of platforms.

A slight inaccuracy in the constraints is that when events occur concurrently, this is not dealt with consistently. This issue can be resolved for by introducing a second modulo parameter for every pair of events. We refer to Peeters (2003) for details.

5.C Definitions of the Performance Measures

Operational measures. The measures are first defined for the operation of a line in a certain direction and can be computed at every departure at the associated terminal station. Later we describe how the measures can be evaluated for an entire line plan at any given time.

For the sake of notation, we start counting the departures from zero. The i'th realized departure time of line l in direction d is denoted as $\text{dep}_l^d(i)$. Furthermore, we let p_l equal T/f_l, the period of a line. At the time of the i'th departure of line l in direction d, the frequency measure is given by:

$$\text{Frequency}_l^d(i) = \frac{i \times p_l}{\text{dep}_l^d(i) - \text{dep}_l^d(0)}. \tag{5.39}$$

The regularity measure is defined as follows:

$$\text{Regularity}_l^d(i) = 1 - \frac{\sum_{j=1}^{i} |\text{dep}_l^d(j) - \text{dep}_l^d(j-1) - p_l|/p_l}{i}. \tag{5.40}$$

For the delay measure, we let d_i denote the relative delay of the i'th trip. Normally, delays are computed with respect to a timetable, but since a timetable is not available, we compute delays relative to the minimum trip times, the sum of minimum running and dwell times from terminal to terminal. For example, if the minimum trip time of line l is 20 minutes, and the i'th trip takes 25 minutes, d_i equals 0.25. The delay measure is given by:

$$\text{Delay}_l^d(i) = \frac{\sum_{j=0}^{i}(1+d_j)}{(i+1)}. \tag{5.41}$$

To compute the performance of a line in a direction at any given time t, we compute the measures at the last departure before t and the first departure after t and interpolate. Next, the frequency, regularity and delay measure of an entire line plan at any given t can be computed by taking the average of the measures over all lines and directions.

Travel measures. The first travel measure corresponds to an empirical analog of the objective used in the line planning algorithm and hence referred to as the realized objective. We let $\mathscr{L}$ denote the set of selected lines in the line plan and let $q_{o,d}^{(\text{dir})}(t)$ denote the number of travel options (direct travel options) on $\rho_{o,d}$ in the interval $(t, t+T)$. Then, the realized objective at time t is simply obtained by replacing $\sum_{l\in\mathcal{L}} x_l$ with $|\mathscr{L}|$ and by substituting the theoretical reduction in (direct) OD-frequency $z_{o,d}^{(\text{dir})}$ in the objective with $g_{o,d} - q_{o,d}^{(\text{dir})}(t)$, the experienced reduction in the next T minutes:

$$\text{Real. Obj.}(t) = w_1 \sum_{(o,d)\in\mathcal{OD}} n_{o,d} \left(\frac{g_{o,d} - q_{o,d}(t)}{g_{o,d}}\right)^2 + w_2 \sum_{(o,d)\in\mathcal{OD}_{\text{dir}}} n_{o,d} \left(\frac{g_{o,d}^{\text{dir}} - q_{o,d}^{\text{dir}}(t)}{g_{o,d}^{\text{dir}}}\right)^2 + w_3|\mathscr{L}|. \tag{5.42}$$

The second travel measure equals the average travel time of travelers entering the system at time t. For this measure, the passengers are routed according to the modified line plan. If we let $\text{time}_{o,d}(t)$ denote the travel time from o to d for a passenger entering station o at time t according to this routing, the measure is defined as follows:

$$\text{Travel Time}(t) = \frac{\sum_{(o,d)\in\mathcal{OD}} n_{o,d}\text{time}_{o,d}(t)}{\sum_{(o,d)\in\mathcal{OD}} n_{o,d}}. \tag{5.43}$$

Chapter 6

Microscopic Simulation of Decentralized Dispatching Strategies in Railways

6.1 Introduction

When railway operations become heavily disrupted, central traffic control may lose the overview of the system and terminate all traffic. In such extreme events, decentralized dispatching strategies could provide a robust back-up plan, offering passengers a service that may not be as good as in regular circumstances, but is much preferred over the alternative of no service at all. In this chapter, we analyze whether decentralized dispatching can indeed serve as a contingency plan when centralized dispatching is impossible. To do so, we develop an integrated platform that simulates decentralized dispatching strategies in a microscopic representation of the railway system.

The relevance of decentralized control in railway systems can be motivated by so-called *out-of-control* situations, which we define as situations 'where dispatchers cease to have an overview of the system and consequently decide to terminate all railway traffic in the affected region, even though the required resources (infrastructure, rolling stock and crew) might be available' (Dekker et al., 2021). Out-of-control situations are often caused by extreme weather events, (possibly short-lasting) power outages, or malfunctioning of telecommunication systems. These situations are typically characterized by a large number of affected resources and incomplete information, yielding traditional rescheduling approaches ineffective. Instead, decentralized decision-making is more robust and better suited for out-of-control situations.

Theoretical justification for the good performance of decentralized dispatching was provided in Chapter 4, where we proved analytically that, under several assumptions, an easy-to-implement dispatching policy matches the performance of centralized dispatching in the long run. A first attempt in applying decentralized dispatching in railways was made in Chapter 5, where we proposed and tested decentralized strategies for dispatching rolling stock. A macroscopic simulation of a part of the Dutch railway network showed that the decentralized approach can quickly restore services to a reasonable level. In this chapter, we take the next step in validating the adequacy of decentralized control by testing the rolling stock strategies proposed in Chapter 5 in a full scope simulation of the railway network, including train drivers and infrastructure at a microscopic level of detail. As decentralized dispatching of train drivers has, to the best of our knowledge, not been considered before, we propose and assess two new driver dispatching strategies that can be applied in conjunction with the rolling stock strategies.

A microscopic railway simulation provides an accurate description of railway traffic by representing all details about the infrastructure (e.g. track gradients, curvature radii), the vehicles (e.g. mass, tractive-effort speed curves of the traction unit, braking rates), signaling (e.g. position and aspect of signals) and the interlocking (position of switches, interlocking rules to prevent conflicts at junctions) (Hansen and Pachl, 2008). Microscopic simulation allows for in-depth analysis of important timetable performance indicators, such as infrastructure occupation, feasibility, robustness and energy efficiency (Goverde and Hansen, 2013). Therefore, such simulations have primarily been used for assessing the quality of timetables or timetable rescheduling approaches, see e.g. Quaglietta et al. (2013), Schlechte et al. (2011) and Solinen et al. (2017). Our simulation framework makes use of the flexible microscopic railway traffic simulator EGTRAIN (Quaglietta, 2014), which features an API module that allows customization of built-in train control functionalities and the interface with external algorithms for real-time dispatching, as the one assessed in this research.

Experiments on a part of the Dutch railway network showcase the potential of decentralized dispatching approaches. Despite the lack of central control, it is possible to approximately meet the target frequencies of the lines in the network with a large degree of regularity and with only small delays. This also holds when crew is added into the mix, as long as we assume that all drivers are willing to work up to 2 hours longer than planned.

Summarizing, the contribution of this chapter is twofold. First, we propose two strategies for decentrally dispatching drivers along with the rolling stock. Secondly, we assess the performance of decentralized dispatching strategies using a microscopic simulation of a part of the Dutch railway network.

The remainder of this chapter is structured as follows. In Section 6.2, we discuss the problem setting and the rolling stock and driver dispatching strategies. In Section 6.3, we discuss the simulation platform. In Section 6.4, we discuss the different performance measures. In Section 6.5, we discuss the results of a series of experiments. Finally, we conclude the chapter in Section 6.6.

6.2 Problem Description and Dispatching Strategies

In this section, we describe the problem we consider in this chapter and discuss the rolling stock and driver dispatching strategies.

6.2.1 Problem Description

In this chapter, we consider the problem of decentrally operating a railway system. As timetables, rolling stock and crew schedules all require centralized control, this implies that we aim to operate the system without a centrally planned timetable, rolling stock and crew schedule. Instead, we use local policies that determine the next task incoming rolling stock and crew should perform. We assume that there is a line plan, specifying the lines and frequencies, that is known by all local dispatchers. Every line is operated in both directions. The objective of the local policies then is to execute this line plan as well as possible, i.e. the frequencies in the line plan should be met, delays should be avoided and the service should be regular.

We assume that the rolling stock is composed of self-powered train units and that there are no restrictions to the use of the rolling stock, so every piece of rolling stock can be used on every line. For the drivers, we assume that there are three constraints that should be taken into account: a break constraint, a planned end-of-duty constraint and a duty length constraint. The break constraint and the end-of-duty constraint are soft constraints, meaning that although it is undesirable to have drivers skip their breaks or work past their planned end-of-duty time, this is not strictly forbidden. We assume that drivers can take breaks at all stations. The duty length constraint is a hard constraint: it is strictly not allowed for a driver to operate any new trips after the driver has worked longer than a specified amount of time. Note that the planned end-of-duty time is earlier than the time at which the maximum duty length is reached. The duty length constraint is evaluated at the beginning of a trip using the minimum trip time. Hence, it may occur that a driver surpasses the maximum duty length while operating a trip because of delays. In such a case, the driver is allowed to finish the trip.

We also need to make assumptions with regards to the safety system to prevent decentralized strategies from causing deadlocks. To see why this is necessary, note that since train routes are not coordinated, such strategies could potentially lead to deadlock situations on single tracks or in station areas. On single tracks, this can simply be prevented by using an (electronic) token system. For station areas (or junctions), it is necessary that local traffic controllers are able to set a route for a train through the station to and/or from the platform, such that no other trains can cross the route until the train has either arrived at the platform or left the station.

6.2.2 Rolling Stock Dispatching

In this chapter, we study rolling stock dispatching strategies that are proposed in Chapter 5. For completeness, we explain them here as well.

The rolling stock dispatching strategies are used to determine the next service of a train when a train finishes a service. Hence, these strategies are only applied at the terminal stations of lines. At other stations, trains always continue their service with a dwell time that is as short as possible. The strategies comprise of two components: the *timing* component and the *turning* component. The timing component of a strategy determines the departure time of the train and the turning component determines the next line of the train.

There are two options for the turning component: *STAT* and *DYN*. STAT stands for static turning. When a strategy uses the STAT component, a train finishing a service of line l is instructed to perform a return trip of line l. In other words, if STAT is used, lines have dedicated vehicles. Conversely, when a strategy uses the DYN component, trains can be exchanged between lines. Then, a train finishing a service is assigned to the line with the earliest desired departure time, which is defined as the sum of the most recent departure time of the line and the desired interdeparture time of the line (e.g., 30 minutes for a line with frequency 2/h). Note that if all lines have the same frequency, an incoming train is always assigned to the line whose latest departure is the longest time ago.

There are also two options for the timing component: *ASAP* and *SYNC*. When a strategy uses the ASAP component, a train finishing its service is always instructed to depart as soon as possible. When a strategy uses the SYNC component, the departure time is determined based on the most recent departure time of the selected line. For example, if a train finishes its service at 09:15 and is assigned to a line with frequency 2 per hour and a most recent departure time of 9:05, the train is instructed to depart at 9:35, to meet the desired interdeparture time of 30 minutes. If instead, the most recent departure time would be before 8:45, the train is instructed to depart as soon as possible.

6.2.3 Driver Dispatching

In this chapter, we propose two strategies that can be used to dispatch drivers in a decentralized manner. Similar to the rolling stock strategies, these strategies determine the next service to be performed by a driver, whenever a driver finishes a service. Moreover, as the rolling stock strategies, the driver strategies also require little information and computation. The driver strategies use the concept of *availability score*. This score indicates to what extent a driver can perform the service within the labor regulations. The availability score is computed based on the characteristics of the service (departure time and destination) and also takes into account for example break and duty length constraints. The lower the availability score, the more labor constraints are (likely) violated. An availability score of 0 indicates that a driver cannot perform a service. The idea of the driver dispatching strategies is to swap drivers at terminals whenever someone with a higher availability score is available.

We next describe how the driver dispatching, in conjunction with the rolling stock dispatching, works in more detail. When a train finishes a service, the rolling stock dispatching strategy proposes a tentative next service. If either the driver currently on the train or any of the drivers that are present at the corresponding station is able to perform this service (i.e. has an availability score strictly larger than 0), the line and departure time are fixed. If none of the drivers is able to perform the service, the departure time and line are adjusted, until either a driver is available or all options are exhausted. Once the line and departure time of the next service are fixed, the crew dispatching strategy determines which driver should operate the service. By default, this is the driver that is currently on the train. If there is a driver at the station with a higher availability score, this driver is assigned to the service and the driver on the train stays at the station.

The availability score can be computed for any combination of a service and a driver. The score is based on a driver's last break time, planned end-of-duty time and crew base, and of the departure time and destination of the service. A lower availability score corresponds to a violation of a more important labor constraint. The exact interpretation of the availability score is stated in Table 6.1. Since drivers are dispatched decentrally and dynamically, it is not possible to predict with certainty whether performing a service will later lead to the violation of a constraint. Therefore, we propose two strategies that have different ways of performing this prediction. In the first strategy, OneStepAhead, only violations during the service are considered. For example, if a driver's planned end-of-duty time is 16:30, the OneStepAhead strategy will only give an availibility score of at most 1 to services that end later than 16:30, regardless of the destination of a service. In contrast, the

TwoStepAhead strategy also takes into account the time required to travel from the destination of a service to the driver's crew base. Only, if a driver is able to return to his/her crew base before the planned end-of-duty time, this strategy will give an availability score larger than 1. This works similarly for the other constraints.

Table 6.1: Interpretation of the availability score

Availability Score	Performing the service...
0	... causes a violation of the duty length constraint
1	... causes the driver to work past his/her planned end-of-duty time
2	... causes a violation of the break constraint
3	... does not violate any constraints

We also need to specify when drivers are sent on to having a break or can sign off completely. In both strategies, any driver that is idle at a break station is assumed to be having a break. Furthermore, whenever a driver is present at his/her crew base and it is past the planned end-of-duty time, the workday of the driver is ended. If a driver is idling at a station other than his/her crew base and is no longer able to perform any services without violating the planned end-of-duty time (i.e. the availability score is always 0 or 1), we assume that the driver travels to the crew base as a passenger. The workday is only ended upon arrival at the crew base.

In principle, any rolling stock dispatching strategy can be combined with any driver dispatching strategy. However, the added flexibility of the DYN strategies is especially useful when drivers are also considered, as they allow a driver at the end of a shift to operate a train towards his/her crew base. On top of that, if there are lines that connect two stations that are not crew bases, a STAT strategy may be ineffective, since a driver operating a train on this line will at some point have to abandon the train and travel to his/her crew base and it is unlikely that there is a driver available who can start operating the train on this line.

6.3 Simulation Platform

We simulate the dispatching strategies using a platform that integrates a dispatching tool and a simulator and manages the continuous communication between these entities. The dispatching tool determines dispatching decisions based on incoming messages that specify departure and arrival times of trains and communicate these

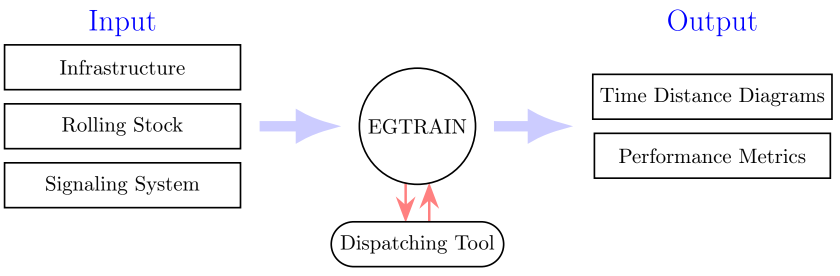


Figure 6.1: Visualization of the integrated platform.

decisions back to the simulator. The simulator simulates the railway traffic that follows from the dispatching decisions and communicates all departure and arrival times back to the dispatching tool, which is in turn processed by the dispatching tool to make new dispatching decisions. Besides managing the continuous communication between the dispatching tool and the simulator, the platform also has an editor for setting up experiments. Such an experiment is sent to a server, which processes it by running the Simulator and dispatching tool. Once an experiment is finished, the realized time distance diagrams of all trains can be visualized, and a variety of metrics can be chosen to display for the assessment of one, or multiple simulation runs.

The simulation platform has a modular design and is, therefore, able to simulate any type of strategy using any type of simulator. In this chapter, the dispatching tool determines dispatching decisions based on the decentralized rolling stock and crew strategies explained in the previous sections. Moreover, as a simulator, we use the microscopic simulator EGTRAIN. The platform is visualized in Figure 6.1. In the remainder of this section, we discuss this simulator in more detail.

EGTRAIN

EGTRAIN (Environment for the desiGn and simulaTion of RAIlway Networks) is a C++ object-oriented model for simulating railway operations at a microscopic level of detail by relying on time-driven processing of traffic events (Quaglietta, 2014). Input data are grouped within four main interacting modules, namely:

- The infrastructure module, which builds on a weighted directed-graph representation of the network where nodes represent physical infrastructure elements like switches, signals, balizes and station platforms while links are rail tracks connecting those elements. Node weights describe geographical coordinates of corresponding infrastructure elements, while link weights depict physical track characteristics such as gradients, speed limits and curvature radii.

- The rolling stock module, that collects physical and mechanical features of trains, including train masses lengths and car composition, as well as braking rates, tractive effort-speed curves and motion resistance coefficients.
- The signaling system module, which stores data about operational principles and rules of both the signaling and the interlocking system. Dependencies between signal aspects, speed codes of the Automatic Train Protection (ATP) system are modeled. Several signaling systems can be simulated ranging from traditional multi-aspect fixed-block signaling (e.g. the Dutch ATB/NS '54, the Italian BACC) to the three levels of the interoperable European signaling standard ERTMS/ETCS (Theeg and Vlasenko, 2009) for which the communication of Movement Authority and train position updates between trains and the RBC are specifically modelled. Additional functionality has been recently added to the signaling module to describe train operations under the next-generation signaling concept of Virtual Coupling (Quaglietta et al., 2020).
- The timetabling module, which contains data about the train schedule such as planned departure/arrival times and minimum dwell times at stations. This module also takes as input stochastic distributions of entrance delays and station dwell times to assess the impact of disturbances on planned operations. In this chapter, trains are dispatched according to decentralized strategies instead of a timetable, so this module is not used.

The core of EGTRAIN simulates train movements by integrating Newton's motion formula over time. At each time step, the speed and position of trains are calculated based on track and vehicle characteristics and the status of the signaling system is updated accordingly to respect safety constraints. Output from the simulation consists of train diagrams (e.g. time-distance, speed-time), delay statistics, mechanical energy consumption and blocking time diagrams. EGTRAIN features an API module for customizing functions, modifying model parameters and interface simulated railway operations with external applications such as sensitivity analysis toolboxes or traffic rescheduling algorithms. The decentralized dispatching strategies presented in this research have hence been interfaced with the EGTRAIN API module to reschedule train services, in real-time during the simulation. The impact of different decentralized dispatching strategies are assessed in simulation in terms of relevant performance measures pertaining to both train services (i.e. frequency, regularity, and delay) as well as crew duty planning (e.g. number of violations to planned lengths of break and duty times of the train crew).

In the remainder of this section, we discuss relevant implementation details.

Signaling system

The signaling of the simulation consists of a three-aspect fixed-block signaling system that resembles the Dutch railway signaling. The control of single tracks is based on a token system, such that a single track can only be occupied by a train at a time. This implies that even trains traveling in the same direction are not allowed to cross the single track simultaneously. This approach also prevents the single track to be used by consecutive trains in the same direction, which would lead to the single track being used for a long time in the same direction. Such a strategy is applied because train services are not known in advance, due to the characteristics of the dispatching strategies.

An additional train route management algorithm has been implemented in EGTRAIN to prevent deadlock between incoming and outgoing trains within a station area. Specifically, the route management algorithm coordinates the entrance and the exit of trains from the same station by allowing trains to enter a station only when there are no trains leaving it in the opposite direction. In this way train deadlocks over bidirectional tracks in approach to the station area are avoided.

Maneuvers at terminal stations

At terminal stations, trains always depart from the platform where they have arrived. An minimum time of 5 minutes is required before a train can depart in the opposite direction.

Dynamic platform allocation

A main characteristics of decentralized dispatching strategies is that trains should preferably have a dynamic platform allocation when entering stations. This means that rather than being pre-scheduled to stop at a specific station platform they can stop at any suitable free platform which is dynamically assigned as they approach the station area. A dynamic platform assignment can hence prevent trains from getting delayed for waiting outside of station areas if a specific platform is occupied by giving the chance to use instead any suitable available platform. The developed dynamic platform allocation algorithm hence automatically assigns trains approaching a station to a suitable free platform in case the pre-scheduled platform is occupied at that moment.

Initial position of trains

For every experiment, trains always start at terminal stations. In some cases, the number of trains starting at a given station exceeds the number of platforms available. When that is the case, the second train assigned to a given platform only enters the station after the first train departs. A maximum of two trains can be assigned to a platform when defining the initial position of trains.

Timestep

For all experiments, a timestep of one second is used, which provides a high level of detail when computing train motion dynamics.

6.4 Performance Measures

To measure the performance of the rolling stock and crew dispatching strategies, we use a set of performance measures. We use the three operational measures proposed in Chapter 5 to assess whether the line plan is executed satisfactorily. These measures are *frequency, regularity* and *delay* and consider the realized frequencies of the lines, the regularity of interdeparture times of lines and the delays, respectively. Besides the operational measures, we also discuss how we assess the performance of our strategies with respect to the constraints for train drivers.

6.4.1 Operational Measures

The metrics are defined for the operation of a line in one direction. Let h denote the target interdeparture time of the line in minutes (so the hourly frequency is $60/h$) and τ the minimum trip time from one terminal to the other in minutes. Let the departures be labeled as 1, 2, ..., n, with departure times (in minutes from the start of the simulation) $d_1 \leq d_2 \leq \cdots \leq d_n$ and realized trip times $t_1, t_2, ..., t_n$. We first define the average realized interdeparture time, which we denote as $\mathcal{H}$:

$$\mathcal{H} = \frac{1}{n-1} \sum_{i=1}^{n-1} (d_{i+1} - d_i)$$

Frequency: The frequency metric, denoted as $\mathcal{F}$, measures the realized frequency, relative to the target frequency (or equivalently, the average realized interdeparture time relative to the target interdeparture time):

$$\mathcal{F} = \frac{h}{\mathcal{H}}.$$

Regularity: The regularity metric, denoted as $\mathcal{R}$, measures to which extent the interdeparture times vary with respect to the average realized interdeparture time:[1]

$$\mathcal{R} = 1 - \frac{1}{(n-1)\mathcal{H}} \sum_{i=1}^{n-1} |d_{i+1} - d_i - \mathcal{H}|.$$

Delay: The delay metric, denoted as $\mathcal{D}$, captures the average delay of the line, measured relative to the theoretical minimum trip time:

$$\mathcal{D} = \frac{\sum_{i=1}^{n} t_i}{n\tau}.$$

Note that all metrics are normalized, such that a value of 1.00 for all metrics indicates the scenario where the target frequency is exactly met, all interdeparture intervals are constant and there are no delays. To assess the performance of the decentralized strategies on the complete line plan, we take an unweighted average over all lines in both directions.

[1] This definition sligtly deviates from Chapter 5. In this chapter, the deviations are measures with respect to the realized average interdeparture time, whereas Chapter 5 measure the deviations with respect to the target interdeparture time.

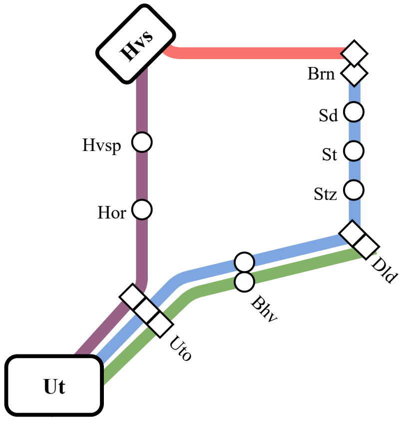


Figure 6.2: The network considered in the experiments.

6.4.2 Crew Measures

To analyze the performance of the crew dispatching strategies, we simply count how many times the break and duty length constraints are violated, and how often drivers need to work past their planned end-of-duty time.

6.5 Results

6.5.1 Instances

Network and line plan

For the experiments, we use a part of the Dutch railway network. This network is depicted in Figure 6.2. The network contains four lines, all of which should be operated with a frequency of 2 per hour. The largest station in this network is Ut (Utrecht Centraal), which also serves as the crew base in our experiments. The part between Dld (Den Dolder) and Brn (Baarn) is single track, with a passing loop at St (Soest).

The infrastructure input data for the microscopic simulation is built from a very detailed database of the Dutch railway network, provided by ProRail, the Dutch infrastructure manager. It was necessary to convert the original data from the database into the specific format of input data used by EGTRAIN. After the conversion, the model of the network is still detailed but with some approximations. These include, for example, the assumption of a fixed block section length and approximated track gradients and speed limits.

To compute the motion dynamics, we use the characteristics of a 6-wagon SLT train, which is an often-used rolling stock type in the considered area. The input data of rolling stock includes the tractive effort-speed curves and other characteristics, e.g. maximum speed, length and mass.

Finally, the default values of dwell times are based on the real timetable of train services running across the area.

Crew data and simulation duration

We use crew data that is based on crew schedules used by NS. The majority of duties of train drivers at NS can roughly be subdivided into morning shift duties, ending somewhere between 12pm and 2pm, and evening shift duties, starting somewhere between 12pm and 2pm. As it is interesting to simulate this period with many driver reliefs, we simulate a duration of 6 hours, from 10am until 4pm. Moreover, when we construct an instance with $2x$ drivers, we generate x morning shift drivers whose duties end between 12pm and 2pm and x evening shift drivers, whose duties start between 12pm and 2pm. The exact starting and end times are uniformly generated within this interval. For all drivers, Ut serves as the crew base. The planned duty lengths are all set equal to 8 hours. The maximum allowed working time is set equal to 10 hours. In other words, a driver is allowed to work at most 2 hours past his/her planned end-of-duty time. The maximum working time without a break is set equal to 4.5 hours.

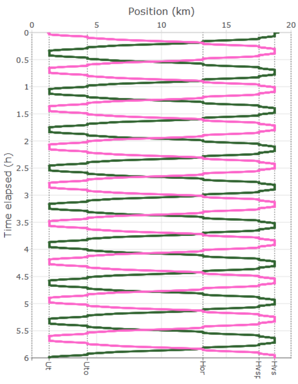
6.5.2 Comparison of Rolling Stock Strategies

In the first experiment, we compare the performance of the four rolling stock dispatching strategies, without simulating the crew. Initially, we assume that there are six trains available in this region.

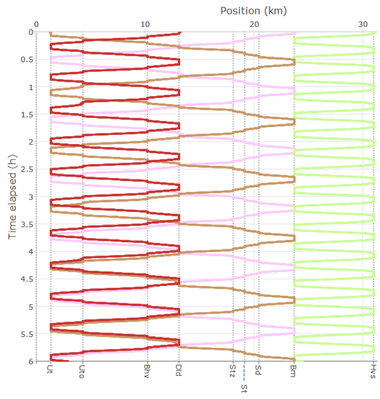
We first analyze the time-distance diagrams that visualize all train trajectories during the simulation. Figure 6.3 and Figure 6.4 present the time-distance diagrams for the four rolling stock dispatching strategies. The line Ut-Hvs is depicted in subfigures (a) and (c). The three other lines are depicted in subfigures (b) and (d). The different shades of gray in the diagrams represents a different train. When we compare the STAT strategies with the DYN strategies, it can be observed that in accordance with the definition of these strategies, when a STAT strategy is applied, trains stick with their initial line, whereas when a DYN strategy is applied, trains can switch between lines. Especially when the ASAP-DYN strategy is used, trains are often exchanged between lines, to serve the line that needs a departure most urgently. Note that there are no switches between different lines at station Brn, as this is prohibited by the infrastructure at that station. When we compare the ASAP strategies with the SYNC strategies, we find that the time-distance diagrams of ASAP-STAT and ASAP-DYN appear to be more cluttered and irregular compared to those of SYNC-STAT and SYNC-DYN. On the other hand, it appears that the ASAP strategies are able to achieve higher frequencies. In none of the diagrams, long delays or queuing of trains can be observed. Moreover, the time-distance diagrams do not give any signs of a long warm-up period required before a steady state is reached. Instead, the behavior of the system seems rather homogeneous over time.

Figure 6.5 presents the operational measures obtained with the different strategies. This figure supports the observations made using the time-distance diagrams. The delay measure is very close to 1.00 for all strategies, indicating that there are hardly any delays. The frequency measure is slightly over 1.00 for the ASAP strategies and slightly below 1.00 for the SYNC strategies. This shows that the SYNC strategies lead to frequencies that are a bit below the target frequencies, while the ASAP strategies lead to frequencies above the target frequencies. This is caused by the fact that the ASAP principle instructs trains to depart as soon as possible, without regard to the desired interdeparture. We find that the SYNC strategies perform better for the regularity measure, with values very close to 1.00. This confirms the observation that the services realized by these strategies are almost perfectly regular. The ASAP strategies score worse in terms of regularity, especially ASAP-DYN. There are only minor differences in the measures obtained with SYNC-STAT and SYNC-DYN. This could be caused by the number of trains available in these experiments, as in the absence of delays, six trains are sufficient to meet the target headways.

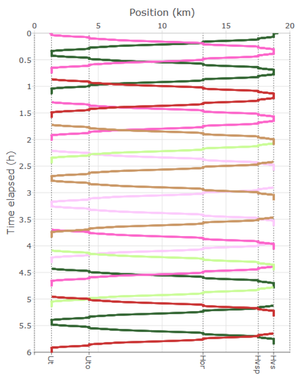
Larger differences in the performance of the strategies become apparent when we analyze the measures per line, especially in terms of frequency. Figure 6.6 presents the frequency measure per line for the four strategies. The static turning principle



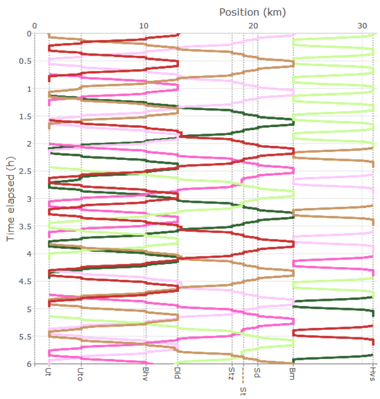
(a) Ut-Hvs with ASAP-STAT



(b) Ut-Brn-Hvs with ASAP-STAT

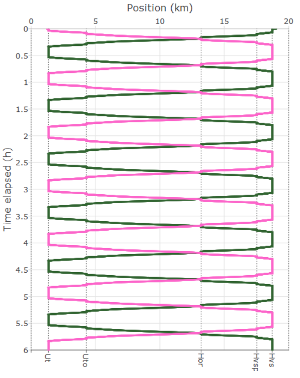


(c) Ut-Hvs with ASAP-DYN

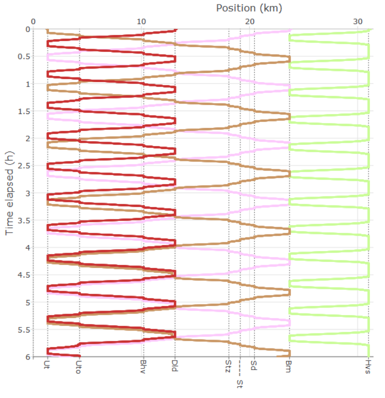


(d) Ut-Brn-Hvs with ASAP-DYN

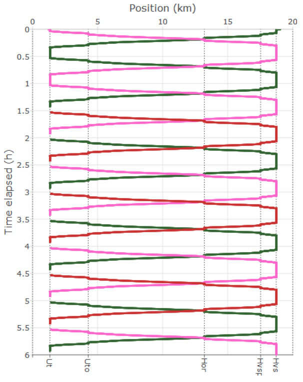
Figure 6.3: Time-distance diagrams obtained by simulating the ASAP strategies with six trains.



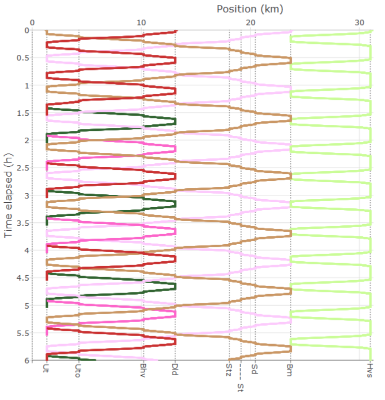
(a) Ut-Hvs with SYNC-STAT



(b) Ut-Brn-Hvs with SYNC-STAT



(c) Ut-Hvs with SYNC-DYN



(d) Ut-Brn-Hvs with SYNC-DYN

Figure 6.4: Time-distance diagrams obtained by simulating the SYNC strategies with six trains.

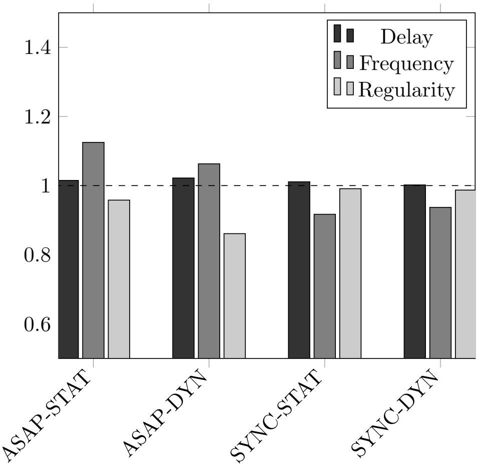


Figure 6.5: The operational measures obtained by simulating the strategies with six trains, averaged over all lines in the network.

can be seen to lead to a much larger dispersion in the frequency. For example, with the ASAP-STAT strategy, half the lines experience a frequency much larger than 1 (up to 1.5), and the other half experience a frequency smaller than 1. This occurs since the lines Ut-Hvs and Hvs-Brn are assigned more trains relative to their trip time. With the ASAP-DYN strategy, the differences in frequency between lines is much smaller, as trains are swapped between lines. To a lesser extent, the same holds for the SYNC-STAT strategy and the SYNC-DYN strategy. The dynamic turning principle hence leads to a more balanced division of resources over the lines.

Varying the number of trains

Besides conducting the experiment with six available trains, we also repeat the experiment with four, five, seven and eight trains. Figure 6.7 presents the performance measures as a function of the number of trains. We find that with ASAP strategies every increase in the number trains translates to an increase in frequency. This is not the case with the SYNC strategies, where the frequency stops increasing after six trains. This aligns with the definition of these strategies, as the SYNC principle instructs trains to wait to meet the target interdeparture time, such that the frequency measure cannot be above 1.0 by definition. The difference in frequency between the STAT principle and the DYN principle again only becomes apparent when the frequency is analyzed per line, as the STAT principle leads to large differences in

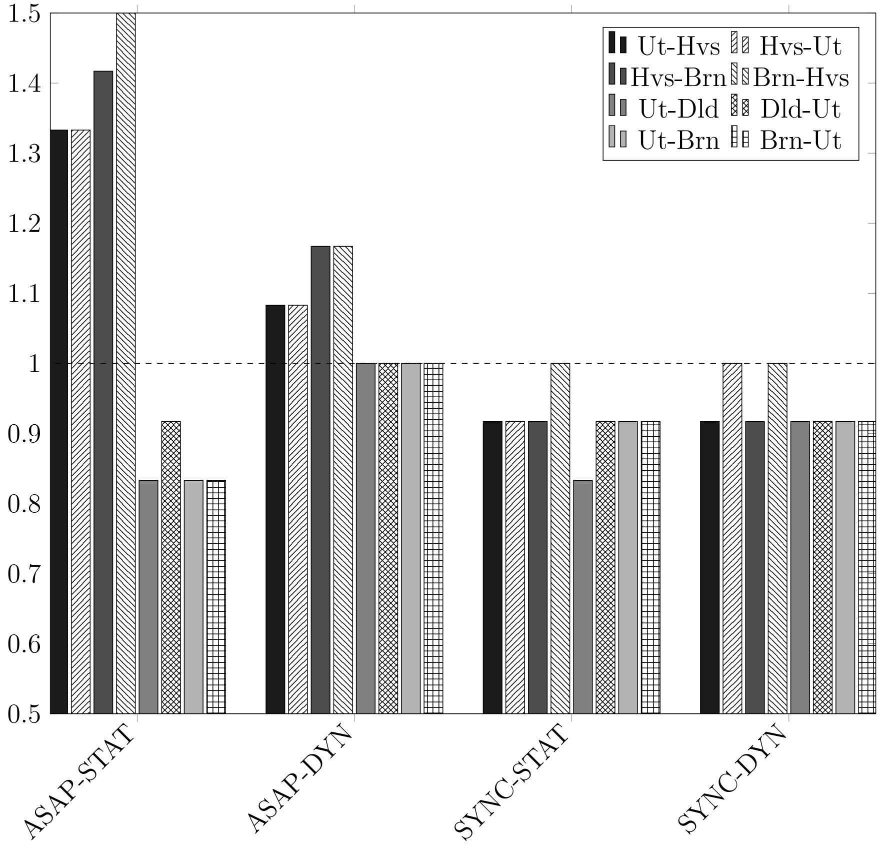


Figure 6.6: Frequency measure per line with six trains.

frequency per line, whereas the DYN principle leads to a more evenly distribution of services over the lines. How the number of the trains affects the other measures is less unambiguous. There appears to be a positive relationship between delay and the number of trains when one of the SYNC strategies is used. This delay can be attributed to the single track part between Dld and Brn, where the abundance of 'slack' in the number of trains causes trains to have to wait for each other at the passing loop at Soest. There are no significant delays in the other parts of the network. As for the regularity, we find that all strategies have fairly a high regularity. The STAT strategies attain a higher or equal regularity than their DYN counterparts, except when there are eight trains. This is caused by the fact that the STAT strategies have a constant number of trains per line, leading to a higher regularity.

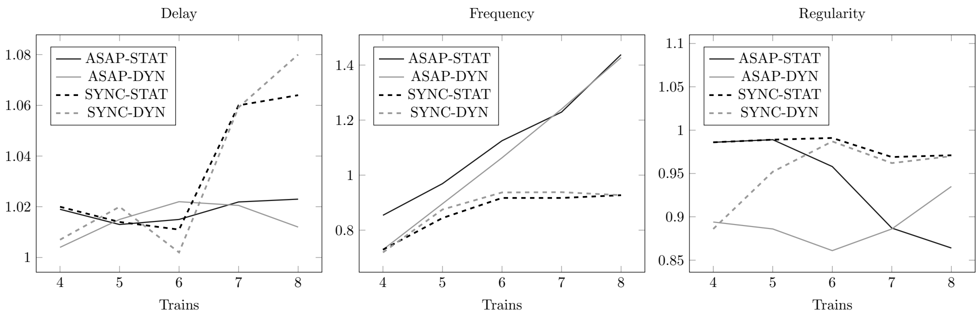


Figure 6.7: Performance measures with a different number of trains.

Higher line frequencies

Furthermore, we analyze the effect of increasing the frequencies of the lines in the network. Specifically, the frequency of the lines Ut-Dld, Ut-Hvs and Hvs-Brn is increased to 4 per hour. The frequency of the line Ut-Brn remains 2 per hour, as the single track cannot manage higher frequencies. We perform this experiment with ten trains. Figures 6.9 and 6.10 visualize the time-distance diagrams. The main finding is that the increased frequencies lead to a higher incidence of delays, which can be observed as vertical lines in the time-distance diagram. This happens occasionally when at the single track part of the network and also right before entering station Hvs. Still, there is no sign of queuing of trains and all delays remain relatively small. This is also reflected in the performance measures, presented in Figure 6.8. Especially the SYNC strategies experience larger delays compared to the case with lower line frequencies. With respect to the frequency measure, the ASAP strategies also outperform the SYNC strategies. On the other hand, the SYNC strategies do score much better on the regularity measure.

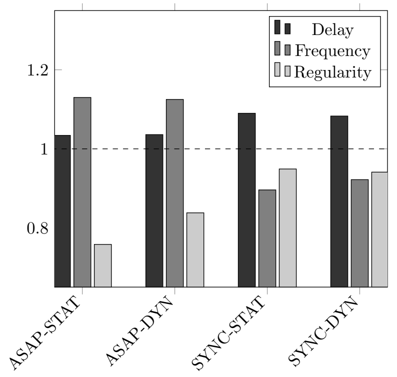
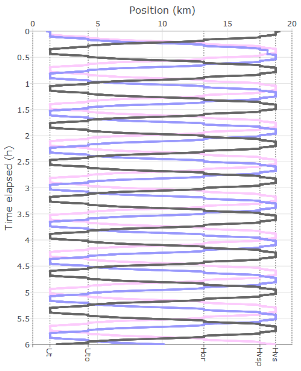
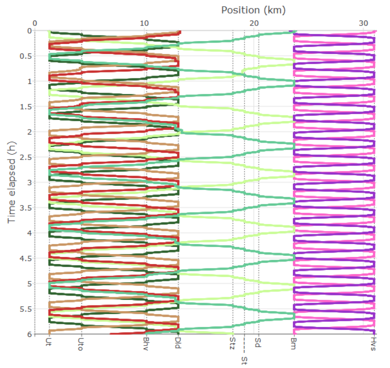


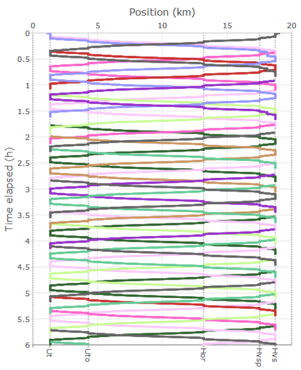
Figure 6.8: The operational measures obtained by simulating the strategies with ten trains, with increased line frequencies, averaged over all lines in the network.



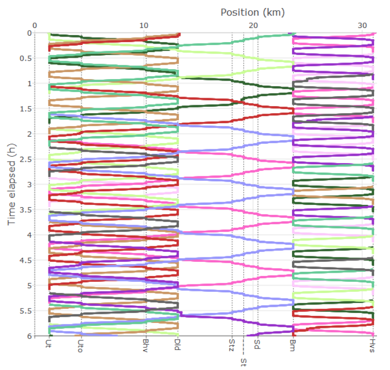
(a) Ut-Hvs with ASAP-STAT



(b) Ut-Brn-Hvs with ASAP-STAT

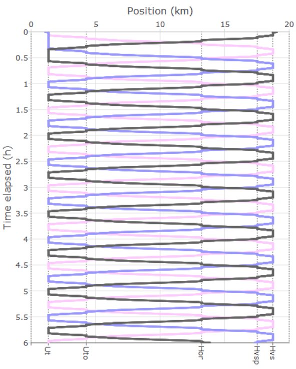


(c) Ut-Hvs with ASAP-DYN

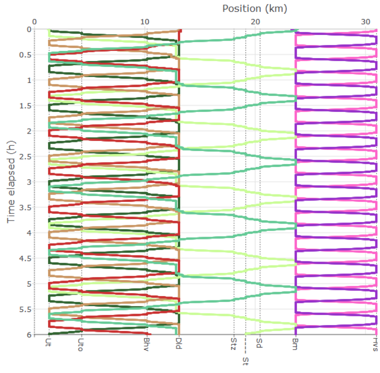


(d) Ut-Brn-Hvs with ASAP-DYN

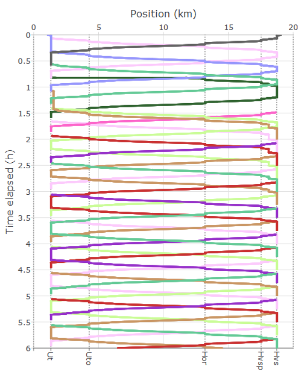
Figure 6.9: Time-distance diagrams obtained by simulating the ASAP strategies with ten trains, with increased line frequencies.



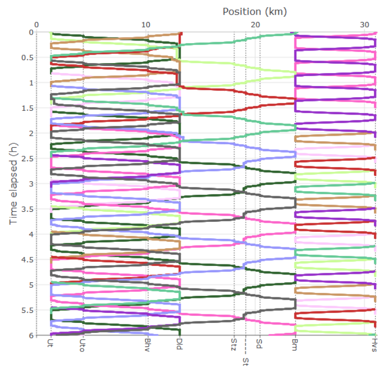
(a) Ut-Hvs with SYNC-STAT



(b) Ut-Brn-Hvs with SYNC-STAT



(c) Ut-Hvs with SYNC-DYN



(d) Ut-Brn-Hvs with SYNC-DYN

Figure 6.10: Time-distance diagrams obtained by simulating the SYNC strategies with ten trains, with increased line frequencies.

6.5.3 Comparison of Crew Strategies

In the second experiment, we compare the performance of the two crew dispatching strategies. As a static rolling stock strategy does not combine well with the flexible switching of drivers and the SYNC-DYN performed well without drivers, we choose the SYNC-DYN strategy for the rolling stock in this experiment. We use six trains, six drivers in the morning shift and six in the evening shift. We perform five runs for every setting, with different crew data.

First, we examine the impact of the inclusion of the crew dispatching in the simulation on the metrics. Figure 6.11 presents the operational measures for the OneStepAhead strategy, the TwoStepAhead strategy, and the case without driver dispatching. We observe that regardless of the crew strategy, the impact of including driver dispatching is small, with only minor differences in the obtained delay, frequency and regularity. Hence, we find that both strategies are successful in maintaining a high level of service. With the TwoStepAhead strategy, the frequency measure is even higher than without driver dispatching. A possible reason for this is that the TwoStepAhead strategy can instruct a train to leave before the desired departure time if that is required to avoid violating driver constraints.

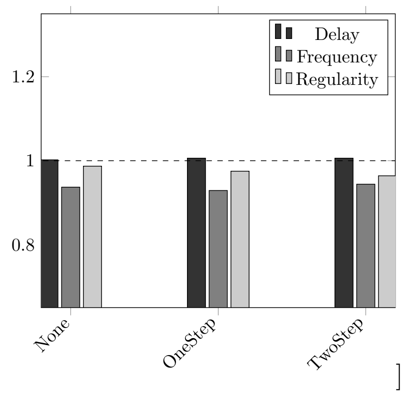
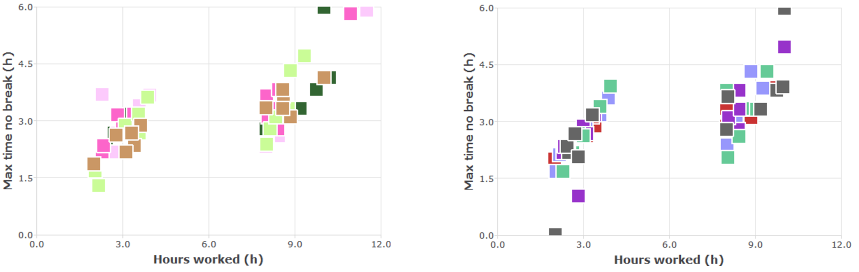


Figure 6.11: Performance measures with different crew dispatching strategies.

Next, we analyze the realized durations of the duty length of drivers and how long drivers have worked without having a break. Figure 6.12 visualizes these statistics for every driver in the five simulation runs. Every shade of gray corresponds to a different run. For both strategies, all points can be divided into two clouds, which correspond to the morning and evening shift drivers, respectively. Recall that the simulation



(a) OneStepAhead (b) TwoStepAhead

Figure 6.12: Scatter plot of the realized duty duration and working time without break for the two strategies, with six trains and twelve drivers (six in the morning shift, six in the evening shift). The rolling stock strategy is SYNC-DYN. Every shade corresponds to a run.

starts at 10am, that the workday of the morning shift drivers starts between 4am and 6am and for the evening shift drivers between 12pm and 2pm. Hence, the left cloud corresponds to the evening shift drivers, who worked a couple of hours at most when the simulation ends. It is more interesting to consider the right cloud, corresponding to the morning shift drivers. We find that with both strategies, the majority of drivers needs to work between 8 and 9 hours. As the planned duty durations are 8 hours, this corresponds to at most 1 hour overtime. Moreover, for most drivers, the constraint that the maximum working time without a break is 4.5 hours is not violated. The difference between the strategies becomes apparent in the outliers, where we find that with the OneStepAhead strategy, five drivers worked more than 10 hours, including two drivers that worked over 11 hours. These drivers got stuck at stations other than the crew base Utrecht, and were required to travel back to Utrecht as a passenger. With the TwoStepAhead strategy, only two drivers worked more than 10 hours, but only with a maximum of 10 hours and 3 minutes, due to a driver operating a train that faced a delay. The TwoStepAhead strategy also leads to fewer violations of the constraint that drivers should have a break every 4.5 hours. Therefore, this indicates that the TwoStepAhead strategy is an effective strategy for avoiding severe violations of the break and duty length constraints.

Higher line frequencies

When we repeat the experiments with increased line frequencies of 4 per hour for all lines except Ut-Brn, we find the results presented in Figure 6.13 and Figure 6.14. We

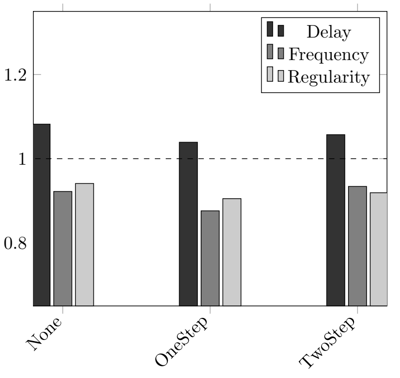
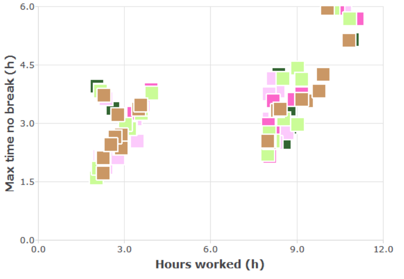
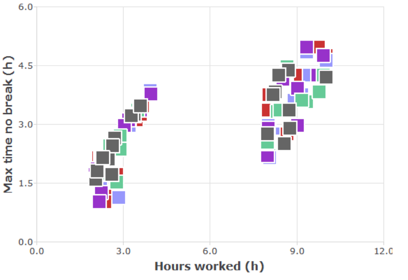


Figure 6.13: Performance measures with different crew dispatching strategies, with increased line frequencies.



(a) OneStepAhead



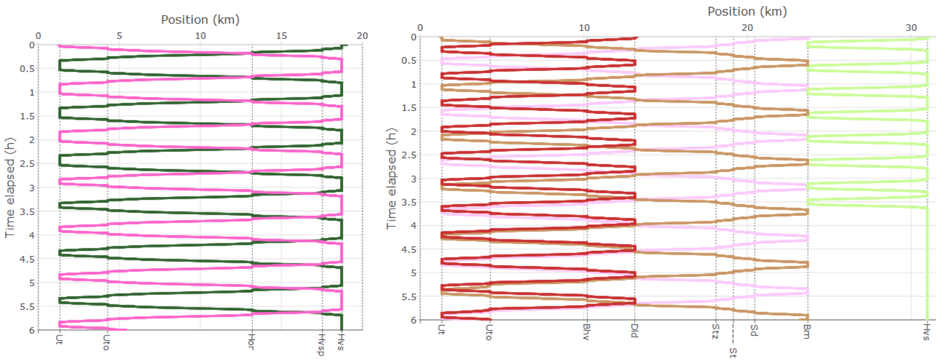
(b) TwoStepAhead

Figure 6.14: Scatter plot of the realized duty duration and working time without break for the two strategies, with ten trains and twenty drivers (ten in the morning shift, ten in the evening shift). The rolling stock strategy is SYNC-DYN. Every shade corresponds to a run.

again find that the impact on the measures of including driver dispatching is small. Furthermore, the TwoStepAhead strategy results in fewer and less severe exceedances of the end-of-duty time of drivers and of the maximum time without a break than the OneStepAhead strategy.

Crew dispatching with a STAT strategy

In Section 6.2.3, we mentioned that rolling stock dispatching with static turning may be ineffective if crew is considered, since it may lead to drivers having to switch trains at stations that are not a crew base, where it is unlikely that there is a driver



(a) Ut-Hvs

(b) Ut-Brn-Hvs

Figure 6.15: Time-distance diagrams, with six trains and twelve drivers. The rolling stock strategy is SYNC-STAT and the crew strategy is TwoStepAhead.

available to start operating the abandoned train. Experiments confirm that when crew is considered, static rolling stock strategies indeed perform badly. Figures 6.15a-b illustrate the issue. The figures show the obtained time-distance diagrams with SYNC-STAT as the rolling stock strategy and TwoStepAhead as the crew dispatching strategy. It can be observed that after about 3.5 hours, the train operating the Hvs-Brn line stops in Hvs and does not operate any more trips. The reason is that the driver originally operating this train, has to switch to the Ut-Hvs line at Hvs, in order to arrive at the crew base Ut before violating the duty length constraint. If the SYNC-DYN strategy would be used, the driver would stay on the train and simply operate a trip of the line to Ut. However, with the SYNC-STAT strategy, the train is not allowed to switch lines, such that the driver has to abandon the train and get on a different train to travel towards Ut. Moreover, as dispatching is done locally, the dispatcher at Ut is unaware of the driver shortage at Hvs. Of course, it is possible that the driver that abandoned the train at Hvs informs the dispatcher at Ut, but it would still take a long time for the replacement driver to arrive at Hvs.

6.6 Conclusion

In this chapter, we tested the performance of decentralized strategies for dispatching rolling stock and drivers in a railway system. Such strategies could serve as a back-up plan when traditional dispatching approaches become infeasible due to disruptions. To analyze whether decentralized dispatching can be a viable alternative, we developed a simulation platform that is able to simulate dispatching strategies on a microscopic representation of the railway system. Experiments on a part of the Dutch railway system indicate that on small instances, easy-to-implement decentralized dispatching strategies can attain high performance, meeting target frequencies with a high degree of regularity and small delays. Strategies where trains are allowed to switch between lines attain the same average frequency as strategies where trains are fixed to lines. However, with these latter strategies, the frequency per line deviates much more strongly from the average frequency, indicating that dynamic switching leads to a more balanced performance. The advantage of dynamic switching strategies become even more clear when drivers are also considered: with static strategies, trains can be left without a driver because the driver needs to switch to a different line to travel to his/her crew base, which is avoided by using a dynamic strategy. Due to the complicatedness of microscopic railway simulation, we have only considered a relatively small instance with four lines and at most 10 trains. It would be interesting to scale up and investigate whether the performance of decentralized dispatching degrades in larger instances with multiple types of trains and lines.

Chapter 7

Summary and Conclusions

In this thesis, we investigated approaches for planning and operating public transport. The first part of this thesis focused on integrating the steps in the public transport planning process, with the general aim to find schedules that are more attractive for passengers, operators and the environment. The second part of this thesis investigated decentralized strategies for operating public transport. Although such strategies could be preferable over centralized and schedule-based control in various scenarios, in this thesis we specifically targeted strategies suitable for out-of-control situations in railway systems.

In Chapter 2, we considered a combination of line planning and vehicle scheduling, by developing methods that estimate how many vehicles are required to operate a line plan, without having to compute a timetable. This allows operators to quickly assess and compare the cost-effectiveness of line plans, avoiding the time-consuming and computationally intensive timetabling step. We considered different restrictions on the number of lines that may be combined in a vehicle circulation and analyzed the impact on computational complexity. For the case where at most two lines can be combined, we developed an exact algorithm exploiting low treewidth, as well as an approximation algorithm based on matching.

In Chapter 3, we developed a novel solution approach to jointly optimize a periodic timetable and the vehicle circulation schedule, allowing operators to make informed trade-offs between passenger service and operating costs. We provided new theoretical results on the vehicle circulation scheduling problem, and used these results to improve the algorithmic performance of the integrated model. Computational experiments showed that this approach finds timetables that require fewer vehicles, without compromising the passengers' perspective.

In Chapter 4, we provided an in-depth theoretical analysis of a decentralized dispatching policy. The proposed policy can easily be applied without the use of a computer, as it only involves a trivial calculation using the most recent departure times of the lines at a station. We showed that, in idealized conditions, this simple policy results in a self-organizing system: once converged, it matches the efficiency of centralized control. We also established worst-case bounds on the time until convergence, as well as the maximum headway deviation if the number of vehicles is insufficient. Numerical experiments illustrated that the policy still performs reasonably well when the assumptions required for the theoretical analysis are not met.

In Chapter 5, we investigated a novel disruption management strategy for dealing with out-of-control situations in railways. This strategy involves modifying the line plan in the affected region, which is subsequently operated using decentralized dispatching strategies. We developed a solution approach based on combinatorial Benders' decomposition that finds passenger-optimal line plans, given infrastructural and rolling stock restrictions that result from the disruption(s). In addition, we proposed several rolling stock dispatching strategies, requiring varying degrees of flexibility and coordination. Computational experiments based on disruptions in the Dutch railway network indicate that the algorithm performs well, finding workable and passenger-oriented line plans within a couple of minutes. Moreover, a macroscopic simulation demonstrated that the produced line plans can be operated smoothly without depending on central coordination.

In Chapter 6, we analyzed the effectiveness of decentralized strategies in a microscopic railway simulation. Besides rolling stock, this simulation involves train drivers, for which we also proposed decentralized strategies. We tested the strategies on a subnetwork of the Dutch railway network, containing eleven stations linked by four train lines. The results showed that with the decentralized dispatching strategies, target frequencies of the lines are approximately met and train services are highly regular without large delays. Especially strategies that allow rolling stock to switch between lines result in a high performance.

7.1 Practical Implications

Chapters 2 and 3 highlight opportunities for public transport operators to improve their planning process. The benefit of integrated planning is most clearly demonstrated in Chapter 3: compared to sequential planning, the integrated approach enables a considerable reduction in the number of required vehicles, driving down costs, at the expense of marginal increases in passenger service. Put differently, integrated planning leads to opportunities to increase passenger service without increasing operating costs. Chapter 3 also provides evidence that allowing vehicle circulations to contain at most two lines (combined circulations) achieves a good balance between costs and the complicatedness of the vehicle circulation schedule, which impacts the robustness of the system. Compared to fixed circulations, combined circulations realize a large decrease in the number of required vehicles, while the additional benefit of allowing more than two lines in a circulation is fairly limited. Under this restriction, the matching heuristic proposed in Chapter 2 provides a fast and intuitive method to estimate the number of vehicles line plans require, allowing operators to better assess and compare the cost-effectiveness of line plans.

Chapters 4, 5 and 6 show that there is certainly merit to decentralized control. Especially among railway practitioners, it is often believed that there is no alternative to centralized control and stick-to-the-plan rescheduling approaches. Our work refutes that belief: flexible, decentralized dispatching strategies are **not** guaranteed to cause a plunge into chaos. In out-of-control situations, the strategies that we propose could even provide a way out of chaos, yielding a reasonable and relatively stable level of service in the affected region. On the other hand, the effectiveness of the considered strategy is dependent on the assumption that there are no interactions between the disrupted region (operated using decentralized strategies), and the rest of the network (operated using conventional approaches). The Netherlands has seen recent examples of out-of-control situation where this assumption is met due to fallen trees blocking tracks, essentially cutting up the network. However, in the majority of the cases, one has to manually enforce the separation between the disrupted region and the rest of the network, which disrupts a considerable proportion of rolling stock and crew schedules, placing a large burden on dispatchers. In other words, imposing a strict boundary between the disrupted region and the rest of the network may solve one problem, but potentially leads to new problems elsewhere. These considerations limit the applicability of decentralized control to out-of-control situations.

In the introduction, we also discussed other possible applications of decentralized strategies: transit networks in developing countries or remote areas, replacement bus services, high-frequency services and first-phase rescheduling. Our promising results indicate that decentralized control is a viable alternative in certain scenarios. However, the mentioned applications all come with their own intricacies, such that more research is required to accurately assess the applicability of such strategies.

7.2 Further Research

There are many interesting and relevant research directions that extend the research in this thesis. A limitation of the work in Chapters 2 and 3 is the assumption that a single vehicle suffices to perform a trip. However, in railway systems, rolling stock scheduling is more involved, as train units can be coupled to increase capacity. In this setting, the methods developed in Chapters 2 and 3 estimate and optimize the number of required *compositions* rather than the number of train units. It would be interesting to investigate the extension of the proposed techniques by taking into account coupling of train units. Another promising research direction is to integrate the predictive model for estimating the number of required vehicles in Chapter 2 into a prescriptive line planning model, to find line plans that can efficiently be covered by a small number of vehicles. Our work could also be applied in different fields than public transport. For example, the formulation proposed in Chapter 3 for optimizing the number of vehicles in a periodic timetable can be applied to optimize other resources in systems operated using periodically repeating schedules.

On the topic of decentralized control in public transport, the opportunities for further research are plentiful. In addition to the mentioned other applications of decentralized strategies that could be researched further, it is worth noting that in this thesis, we primarily assessed performance by analyzing frequencies, regularity and delay. These measures unquestionably affect the experience of passengers. However, it would be interesting to move passengers more to the foreground. Besides modeling the impact of certain strategies on passengers in more detail, also the impact the passengers have on the system, for example due to bunching effects, is worth investigating. Furthermore, one could even analyze decentralized strategies that depend directly on the passengers (for example, "depart when the number of vehicles is at least x% of the capacity"). These aspects could be investigated both analytically, as an extension to Chapter 4, or empirically, in the style of Chapters 5 and 6.

References

Abbink, E.J., Albino, L., Dollevoet, T., Huisman, D., Roussado, J. and Saldanha, R.L. (2011). Solving large scale crew scheduling problems in practice. *Public Transport* 3(2), 149–164.

Argote-Cabanero, J., Daganzo, C.F. and Lynn, J.W. (2015). Dynamic control of complex transit systems. *Transportation Research Part B: Methodological* 81, 146–160.

Balinski, M.L. (1965). Integer programming: methods, uses, computations. *Management Science* 12(3), 253–313.

Bampas, E., Gąsieniec, L., Hanusse, N., Ilcinkas, D., Klasing, R. and Kosowski, A. (2009). Euler tour lock-in problem in the rotor-router model. *International Symposium on Distributed Computing*. Springer, 423–435.

Bampas, E., Gąsieniec, L., Hanusse, N., Ilcinkas, D., Klasing, R., Kosowski, A. and Radzik, T. (2017). Robustness of the Rotor–Router Mechanism. *Algorithmica* 78(3), 869–895.

Bartholdi, J.J. and Eisenstein, D.D. (2012). A self-coördinating bus route to resist bus bunching. *Transportation Research Part B: Methodological* 46(4), 481–491.

Berrebi, S.J., Watkins, K.E. and Laval, J.A. (2015). A real-time bus dispatching policy to minimize passenger wait on a high frequency route. *Transportation Research Part B: Methodological* 81, 377–389.

Bodlaender, H.L. (1996). A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM Journal on Computing* 25(6), 1305–1317.

Bodlaender, H.L. and Kloks, T. (1996). Efficient and constructive algorithms for the pathwidth and treewidth of graphs. *Journal of Algorithms* 21(2), 358–402.

Borndörfer, R., Grötschel, M. and Pfetsch, M.E. (2007). A column-generation approach to line planning in public transport. *Transportation Science* 41(1), 123–132.

Borndörfer, R., Hoppmann, H., Karbstein, M. and Löbel, F. (2016). The Modulo Network Simplex with Integrated Passenger Routing. *Operations Research Proceedings 2016*. Springer, 637–644.

Borndörfer, R., Karbstein, M., Liebchen, C. and Lindner, N. (2018a). A Simple Way to Compute the Number of Vehicles That Are Required to Operate a Periodic Timetable. *18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik,

Borndörfer, R., Klug, T., Lamorgese, L., Mannino, C., Reuther, M. and Schlechte, T. (2018b). *Handbook of Optimization in the Railway Industry*. Vol. 268. Springer.

Borndörfer, R., Lindner, N. and Roth, S. (2019). A concurrent approach to the periodic event scheduling problem. *Journal of Rail Transport Planning & Management*, 100175.

Bunte, S. and Kliewer, N. (2009). An overview on vehicle scheduling models. *Public Transport* 1(4), 299–317.

Burggraeve, S., Bull, S.H., Vansteenwegen, P. and Lusby, R.M. (2017). Integrating robust timetabling in line plan optimization for railway systems. *Transportation Research Part C: Emerging Technologies* 77, 134–160.

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

Caimi, G., Kroon, L. and Liebchen, C. (2017). Models for railway timetable optimization: Applicability and applications in practice. *Journal of Rail Transport Planning & Management* 6(4), 285–312.

Caprara, A., Kroon, L., Monaci, M., Peeters, M. and Toth, P. (2007). Passenger railway optimization. *Handbooks in Operations Research and Management Science* 14, 129–187.

Chalopin, J., Das, S., Gawrychowski, P., Kosowski, A., Labourel, A. and Uznański, P. (2015). Limit behavior of the multi-agent rotor-router system. *International Symposium on Distributed Computing*. Springer, 123–139.

Choi, I., Kim, J. and Suil, O. (2016). The difference and ratio of the fractional matching number and the matching number of graphs. *Discrete Mathematics* 339(4), 1382–1386.

Claessens, M.T., van Dijk, N.M. and Zwaneveld, P.J. (1998). Cost optimal allocation of rail passenger lines. *European Journal of Operational Research* 110(3), 474–489.

Codato, G. and Fischetti, M. (2006). Combinatorial Benders' cuts for mixed-integer linear programming. *Operations Research* 54(4), 756–766.

Daganzo, C.F. (2009). A headway-based approach to eliminate bus bunching: Systematic analysis and comparisons. *Transportation Research Part B: Methodological* 43(10), 913–921.

Dekker, M.M., Van Lieshout, R.N., Ball, R.C., Bouman, P.C., Dekker, S.C., Dijkstra, H.A., Goverde, R.M.P., Huisman, D., Schaafsma, A.A.M. and Van den Akker, J.M. (2021). A Next Step in Disruption Management: Combining Operations Research and Complexity Science. *Public Transport*. To appear.

Delgado, F., Munoz, J.C. and Giesen, R. (2012). How much can holding and/or limiting boarding improve transit performance? *Transportation Research Part B: Methodological* 46(9), 1202–1217.

Dereniowski, D., Kosowski, A., Pająk, D. and Uznański, P. (2016). Bounds on the cover time of parallel rotor walks. *Journal of Computer and System Sciences* 82(5), 802–816.

Desfontaines, L. and Desaulniers, G. (2018). Multiple depot vehicle scheduling with controlled trip shifting. *Transportation Research Part B: Methodological* 113, 34–53.

Dessouky, M., Hall, R., Nowroozi, A. and Mourikas, K. (1999). Bus dispatching at timed transfer transit stations using bus tracking technology. *Transportation Research Part C: Emerging Technologies* 7(4), 187–208.

Fan, W. and Machemehl, R.B. (2009). Do transit users just wait for buses or wait with strategies? Some numerical results that transit planners should see. *Transportation Research Record* 2111(1), 169–176.

Fioole, P.-J., Kroon, L., Maróti, G. and Schrijver, A. (2006). A rolling stock circulation model for combining and splitting of passenger trains. *European Journal of Operational Research* 174(2), 1281–1297.

Fonseca, J.P., Van der Hurk, E., Roberti, R. and Larsen, A. (2018). A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling. *Transportation Research Part B: Methodological* 109, 128–149.

Garey, M.R. and Johnson, D.S. (1979). *Computers and intractability. A guide to the theory of NP-completeness. A Series of Books in the Mathematical Sciences.*

Gattermann, P., Großmann, P., Nachtigall, K. and Schöbel, A. (2016). Integrating Passengers' Routes in Periodic Timetabling: a SAT approach. *16th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2016).* Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 3:1–3:15.

Ghaemi, N., Cats, O. and Goverde, R.M.P. (2017). Railway disruption management challenges and possible solution directions. *Public Transport* 9(1-2), 343–364.

Goerigk, M. and Liebchen, C. (2017). An Improved Algorithm for the Periodic Timetabling Problem. *17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, 1.

Goerigk, M. and Schöbel, A. (2013). Improving the modulo simplex algorithm for large-scale periodic timetabling. *Computers & Operations Research* 40(5), 1363–1370.

Goossens, J.-W., Van Hoesel, S. and Kroon, L.G. (2004). A branch-and-cut approach for solving railway line-planning problems. *Transportation Science* 38(3), 379–393.

Goverde, R.M. and Hansen, I.A. (2013). Performance indicators for railway timetables. *2013 IEEE International Conference on intelligent rail transportation proceedings.* IEEE, 301–306.

Großmann, P., Hölldobler, S., Manthey, N., Nachtigall, K., Opitz, J. and Steinke, P. (2012). Solving periodic event scheduling problems with SAT. *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems.* Springer, 166–175.

Hansen, I.A. and Pachl, J. (2008). *Railway Timetable and Traffic. Analysis, Modelling, Simulation.* Hamburg, Germany: Eurailpress.

Hernández, D., Muñoz, J.C., Giesen, R. and Delgado, F. (2015). Analysis of real-time control strategies in a corridor with multiple bus services. *Transportation Research Part B: Methodological* 78, 83–105.

Huisman, D., Kroon, L.G., Lentink, R.M. and Vromans, M.J. (2005). Operations research in passenger railway transportation. *Statistica Neerlandica* 59(4), 467–497.

Ibarra-Rojas, O.J., Giesen, R. and Rios-Solis, Y.A. (2014). An integrated approach for timetabling and vehicle scheduling problems to analyze the trade-off between level of service and operating costs of transit networks. *Transportation Research Part B: Methodological* 70, 35–46.

Ibarra-Rojas, O.J., López-Irarragorri, F. and Rios-Solis, Y.A. (2015). Multiperiod bus timetabling. *Transportation Science* 50(3), 805–822.

Jespersen-Groth, J., Potthoff, D., Clausen, J., Huisman, D., Kroon, L.G., Maróti, G. and Nielsen, M.N. (2009). Disruption management in passenger railway transportation. *Robust and online large-scale optimization.* Springer, 399–421.

Kaspi, M. and Raviv, T. (2013). Service-oriented line planning and timetabling for passenger trains. *Transportation Science* 47(3), 295–311.

Kliewer, N., Amberg, B. and Amberg, B. (2012). Multiple depot vehicle and crew scheduling with time windows for scheduled trips. *Public Transport* 3(3), 213–244.

Kroon, L., Huisman, D., Abbink, E., Fioole, P.-J., Fischetti, M., Maróti, G., Schrijver, A., Steenbeek, A. and Ybema, R. (2009). The new Dutch timetable: The OR revolution. *Interfaces* 39(1), 6–17.

Kroon, L., Peeters, L., Wagenaar, J. and Zuidwijk, R. (2013). Flexible connections in PESP models for cyclic passenger railway timetabling. *Transportation Science* 48(1), 136–154.

Liang, S., Zhao, S., Lu, C. and Ma, M. (2016). A self-adaptive method to equalize headways: Numerical analysis and comparison. *Transportation Research Part B: Methodological* 87, 33–43.

Liebchen, C. (2008). The first optimized railway timetable in practice. *Transportation Science* 42(4), 420–435.

Liebchen, C. and Möhring, R.H. (2007). The modeling power of the Periodic Event Scheduling Problem: railway timetables - and beyond. *Algorithmic methods for railway optimization.* Springer, 3–40.

Liebchen, C. and Peeters, L. (2009). Integral cycle bases for cyclic timetabling. *Discrete Optimization* 6(1), 98–109.

Liebchen, C., Proksch, M. and Wagner, F.H. (2008). Performance of algorithms for periodic timetable optimization. *Computer-Aided Systems in Public Transport.* Springer, 151–180.

Lovász, L. (1993). Random walks on graphs: A survey. *Combinatorics, Paul Erdös is Eighty* 2(1), 1–46.

Maróti, G. (2017). A branch-and-bound approach for robust railway timetabling. *Public Transport* 9(1-2), 73–94.

Monmerqué, L. (1828). *Les carrosses à cinq sols, ou les omnibus du 17. siecle.* Didot.

Nachtigall, K. (1994). *A branch and cut approach for periodic network programming.* Inst. für Mathematik.

Nachtigall, K. and Opitz, J. (2008). Solving periodic timetable optimisation problems by modulo simplex calculations. *8th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018).* Schloss Dagstuhl-Leibniz-Zentrum für Informatik,

Nührenberg, G. (2016). Integrated timetabling and rolling stock scheduling. MA thesis. ETH-Zürich.

Odijk, M.A. (1997). Railway Timetable Generation. PhD thesis. Delft University of Technology, Delft, The Netherlands.

Odijk, M.A. (1996). A constraint generation algorithm for the construction of periodic railway timetables. *Transportation Research Part B: Methodological* 30(6), 455–464.

Orlin, J.B. (1982). Minimizing the number of vehicles to meet a fixed periodic schedule: An application of periodic posets. *Operations Research* 30(4), 760–776.

Pätzold, J., Schiewe, A., Schiewe, P. and Schöbel, A. (2017). Look-ahead approaches for integrated planning in public transportation. *17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2017)*. Schloss Dagstuhl-Leibniz-Zentrum füer Informatik, 17:1–17:16.

Peeters, L. (2003). Cyclic railway timetable optimization. EPS-2003-022-LIS. PhD thesis. Erasmus University Rotterdam, Rotterdam, The Netherlands.

Petit, A., Lei, C. and Ouyang, Y. (2019). Multiline Bus Bunching Control via Vehicle Substitution. *Transportation Research Part B: Methodological* 126, 68–86.

Polinder, G.-J., Breugem, T., Dollevoet, T. and Maróti, G. (2019). An adjustable robust optimization approach for periodic timetabling. *Transportation Research Part B: Methodological* 128, 50–68.

Priezzhev, V.B., Dhar, D., Dhar, A. and Krishnamurthy, S. (1996). Eulerian walkers as a model of self-organized criticality. *Physical Review Letters* 77(25), 5079.

Quaglietta, E. (2014). A simulation-based approach for the optimal design of signalling block layout in railway networks. *Simulation Modelling Practice and Theory* 46, 4–24.

Quaglietta, E., Corman, F. and Goverde, R.M. (2013). Stability analysis of railway dispatching plans in a stochastic and dynamic environment. *Journal of Rail Transport Planning & Management* 3(4), 137–149.

Quaglietta, E., Wang, M. and Goverde, R.M. (2020). A multi-state train-following model for the analysis of virtual coupling railway operations. *Journal of Rail Transport Planning & Management* 15, 100195.

Robertson, N. and Seymour, P.D. (1986). Graph minors. II. Algorithmic aspects of tree-width. *Journal of Algorithms* 7(3), 309–322.

Sánchez-Martínez, G., Koutsopoulos, H. and Wilson, N. (2016). Real-time holding control for high-frequency transit with dynamics. *Transportation Research Part B: Methodological* 83, 1–19.

Schiewe, A., Albert, S., Schiewe, P., Schöbel, A. and Spühler, F. (2020). *Documentation for LinTim 2020.12.*

Schlechte, T., Borndörfer, R., Erol, B., Graffagnino, T. and Swarat, E. (2011). Micro–macro transformation of railway networks. *Journal of Rail Transport Planning & Management* 1(1), 38–48.

Schmid, V. and Ehmke, J.F. (2015). Integrated timetabling and vehicle scheduling with balanced departure times. *OR Spectrum* 37(4), 903–928.

Schöbel, A. (2012). Line planning in public transportation: models and methods. *OR spectrum* 34(3), 491–510.

Schöbel, A. (2017). An eigenmodel for iterative line planning, timetabling and vehicle scheduling in public transportation. *Transportation Research Part C: Emerging Technologies* 74, 348–365.

Schrijver, A. and Steenbeek, A. (1993). *Spoorweg Dienstregelingontwikkeling (Railway timetable development).* URL: `https://homepages.cwi.nl/~lex/sdo.pdf`.

Serafini, P. and Ukovich, W. (1989). A mathematical model for periodic scheduling problems. *SIAM Journal on Discrete Mathematics* 2(4), 550–581.

Solinen, E., Nicholson, G. and Peterson, A. (2017). A microscopic evaluation of railway timetable robustness and critical points. *Journal of Rail Transport Planning & Management* 7(4), 207–223.

Tamaki, H. (2017). Positive-Instance Driven Dynamic Programming for Treewidth. *25th Annual European Symposium on Algorithms (ESA 2017)*. Ed. by K. Pruhs and C. Sohler. Vol. 87. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 68:1–68:13.

Theeg, G. and Vlasenko, S. (2009). Railway Signalling & Interlocking. *International Compendium. Hamburg, Eurailpress.*

Van Lieshout, R.N., Bouman, P.C. and Huisman, D. (2020). Determining and Evaluating Alternative Line Plans in Out-of-Control Situations. *Transportation Science* 54(3), 740–761.

Van Lieshout, R.N. (2021). Integrated Periodic Timetabling and Vehicle Circulation Scheduling. *Transportation Science* 55(3), 768–790.

Van Lieshout, R.N. and Bouman, P.C. (2018). Vehicle Scheduling Based on a Line Plan. *18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 15:1–15:14.

Van Lieshout, R.N., Bouman, P.C., Van den Akker, J.M. and Huisman, D. (2021). A Self-Organizing Policy for Vehicle Dispatching in Public Transit Systems with Multiple Lines. *Transportation Research Part B: Methodological* 152, 46–64.

Vanderbeck, F. and Wolsey, L.A. (2010). Reformulation and decomposition of integer programs. *50 Years of Integer Programming 1958-2008*, 431–502.

Veelenturf, L.P., Kidd, M.P., Cacchiani, V., Kroon, L.G. and Toth, P. (2015). A railway timetable rescheduling approach for handling large-scale disruptions. *Transportation Science* 50(3), 841–862.

Veelenturf, L.P., Potthoff, D., Huisman, D., Kroon, L.G., Maróti, G. and Wagelmans, A.P.M. (2014). A quasi-robust optimization approach for crew rescheduling. *Transportation Science* 50(1), 204–215.

Visentini, M.S., Borenstein, D., Li, J.-Q. and Mirchandani, P.B. (2014). Review of real-time vehicle schedule recovery methods in transportation services. *Journal of Scheduling* 17(6), 541–567.

Wagner, I.A., Lindenbaum, M. and Bruckstein, A.M. (1999). Distributed covering by ant-robots using evaporating traces. *IEEE Transactions on Robotics and Automation* 15(5), 918–933.

Yanovski, V., Wagner, I.A. and Bruckstein, A.M. (2003). A distributed ant algorithm for efficiently patrolling a network. *Algorithmica* 37(3), 165–186.

Zhang, S. and Lo, H.K. (2018). Two-way-looking self-equalizing headway control for bus operations. *Transportation Research Part B: Methodological* 110, 280–301.

Samenvatting

Openbaar vervoer speelt in onze hedendaagse maatschappij een essentiële rol waar iedereen van profiteert. Effectief openbaar vervoer biedt mensen de mogelijkheid om van punt A naar punt B te reizen op een betaalbare, comfortabele en snelle manier, en is zo een grote stimulans voor economische mobiliteit en economische activiteit in het algemeen. Ook gebruikers van privaat vervoer profiteren van openbaar vervoer, aangezien openbaar vervoer de druk op het wegennet drastisch verlaagt door het samenvoegen van reizigers in voertuigen met hoge capaciteit. Een ander belangrijk voordeel van openbaar vervoer is de lage emissie per passagierskilometer ten opzichte van privaat wegvervoer, waardoor het een rol kan spelen in de strijd tegen klimaatverandering.

Gezien deze voordelen, is het belangrijk om openbaar vervoersystemen (ov-systemen) zo duurzaam, kostenefficiënt en aantrekkelijk mogelijk voor de reiziger te maken. Dit proefschrift draagt hieraan bij door nieuwe aanpakken voor het plannen, be- en bijsturen van ov-systemen te onderzoeken, en doet dat in twee delen.

Deel I richt zich op het integreren van verschillende stappen in het planningsproces van aanbieders van openbaar vervoer. In vergelijking met de gangbare sequentiële aanpak kan een dergelijke geïntegreerde aanpak in theorie betere oplossingen vinden, wat aanzienlijke besparingen en/of verbetering van het aanbod voor de reiziger op kan leveren.

Hoofdstuk 2 beschouwt een combinatie tussen lijnvoering en voertuigplanning, door het ontwikkelen van methodes die inschatten hoeveel voertuigen er nodig zijn om een gegeven lijnvoering uit te voeren, zonder dat hiervoor een volledige dienstregeling uitgerekend hoeft te worden. We besteden vooral aandacht aan het geval waarbij een voertuigcirculatie maximaal twee verschillende lijnen mag bevatten, aangezien dit de

complexiteit van het voertuigschema beperkt. Voor dit geval ontwikkelen we zowel een exact algoritme, als een efficiënte approximatiemethode.

Hoofdstuk 3 beschouwt een combinatie tussen dienstregeling en voertuigplanning. Hier ontwikkelen we een optimalisatiemodel dat geïntegreerd een dienstregeling en de voertuigcirculaties uitrekent, om zo een afweging te kunnen maken tussen kosten en serviceniveau voor de reiziger. We bewijzen nieuwe theoretische resultaten over het plannen van voertuigcirculaties en gebruiken deze resultaten om de rekenprestaties van het model te verbeteren. Experimenten laten het voordeel van onze geïntegreerde aanpak zien: er worden dienstregelingen gevonden die substantieel minder voertuigen nodig hebben om uitgevoerd te worden, zonder dat de reiziger erg moet inleveren qua serviceniveau.

Deel II onderzoekt decentrale aanpakken om ov-systemen te be- of bijsturen, met een focus op spoorsystemen. In reguliere situaties, waar we mogen uitgaan van functionerende communicatiesystemen, vlotte bijsturing en complete beschikbaarheid van informatie, is centrale bijsturing geprefereerd boven decentrale bijsturing. Wanneer een van deze aannames echter wegvalt worden de voordelen van decentrale besturing zichtbaar. Dit is bijvoorbeeld het geval in *out-of-control* situaties in spoorsystemen, welke het gevolg kunnen zijn van een opeenstapeling van grote verstoringen. In dit deel onderzoeken we of decentrale bijsturing een uitkomst kan zijn voor out-of-control situaties, en andere situaties waar de centrale aanpak ineffectief is.

Hoofdstuk 4 presenteert een uitgebreide theoretische analyse van een eenvoudig toe te passen decentrale beslisregel. De voorgestelde beslisregel bepaalt wanneer er op een eindstation een voertuig arriveert op basis van de vorige vertrektijden hoe laat en in welke richting dit voertuig weer moet vertrekken. We bewijzen dat, in geïdealiseerde situaties, deze beslisregel resulteert een een zelf-organiserend systeem: mits er genoeg voertuigen zijn, convergeert het systeem spontaan naar een stabiele dienstregeling. Hiernaast leiden we bovengrenzen af voor de tijd tot convergentie en de afwijking van het gewenste vertrekinterval indien er niet voldoende voertuigen zijn. In numerieke experimenten laten we zien dat wanneer de aannames van de theoretische analyse niet op gaan, de beslisregel nog steeds leidt tot goede prestaties van het systeem.

Hoofdstuk 5 onderzoekt een nieuwe strategie om met out-of-control situaties om te gaan. In deze strategie wordt eerst de lijnvoering in het verstoorde gebied aangepast, waarna dit gebied met decentrale beslisregels bestuurd kan worden. We ontwikkelen een oplosmethode om een lijnvoering te vinden die het ongemak van passagiers minimaliseert, rekening houdend met infrastructurele restricties en materieelbeperkingen. Daarnaast stellen we meerdere beslisregels voor om decentraal te bepalen wanneer en en in welke richting inkomende treinen moet vertrekken. Experimenten gebaseerd op verstoringen in het Nederlandse spoornetwerk laten zien dat het algoritme binnen enkele minuten passagiersgeoriënteerde lijnvoeringen kan vinden. Een macroscopische spoorsimulatie laat verder zien dat de gevonden lijnvoeringen in combinatie met de decentrale beslisregels werkbaar zijn, en niet leiden tot grote vertragingen of onregelmatigheden.

Ten slotte analyseert Hoofdstuk 6 de effectiviteit van decentrale besturing in een microscopische spoorsimulatie, waarin het spoorsysteem op een zeer gedetailleerd niveau wordt gesimuleerd. Naast het materieel stellen we hier ook beslisregels voor voor treinpersoneel. We testen de beslisregels opnieuw op een deel van het Nederlandse spoornetwerk. De resultaten laten zien dat de voorgestelde beslisregels een regelmatige treindienst kunnen opleveren, waarbij de doelfrequenties gehaald worden en er geen grote vertragingen optreden.

Ons onderzoek levert een aantal praktische implicaties op. Het onderzoek in Deel I laat mogelijkheden zien voor aanbieders van openbaar vervoer om hun planningsproces te verbeteren. De toegevoegde waarde van geïntegreerde planning wordt goed duidelijk uit de resultaten van Hoofdstuk 3: in vergelijking met sequentieel plannen leidt een geïntegreerde aanpak tot een aanzienlijke reductie in het aantal benodigde voertuigen, tegen een marginaal verlies in reistijd voor de reiziger. Met andere woorden, geïntegreerd plannen biedt mogelijkheden om het serviceniveau te verhogen zonder dat dit extra kosten meebrengt. Hoofdstuk 3 laat ook zien dat het toestaan van voertuigcirculaties met maximaal twee lijnen tot een goede balans leidt tussen kosten en de complexiteit van het voertuigschema. Onder deze restrictie levert Hoofdstuk 2 een snelle en intuïtieve methode om het aantal benodigde voertuigen voor een lijnvoering te bepalen, wat het mogelijk maakt om beter de kosteneffectiviteit van lijnvoeringen te vergelijken.

Het onderzoek in Deel II laat zien dat er kansen liggen om ov-systemen op een decentrale manier te besturen. Zeker onder mensen werkzaam in de spoorsector wordt er regelmatig gedacht dat er geen alternatief bestaat voor gecentraliseerd en *stick-to-the-plan* bijsturen. Ons onderzoek ondersteunt die hypothese echter niet: op flexibele wijze decentraal bijsturen hoeft niet in chaos te eindigen. Uiteraard geniet centrale bijsturing de voorkeur in het gros van de situaties, maar in bijvoorbeeld out-control-situaties, wanneer centraal bijsturen niet mogelijk is, kan decentraal bijsturen als alternatief dienen.

Een belangrijke beperking van ons onderzoek is wel dat we hebben aangenomen dat er geen interacties zijn tussen het verstoorde gebied, wat decentraal bestuurd wordt, en het niet-verstoorde gebied, wat met reguliere methoden bestuurd wordt. In enkele recente out-of-control situaties gaat deze aanname op, bijvoorbeeld wanneer er door een storm bomen op het spoor vallen, waardoor het netwerk vanzelf wordt opgeknipt. In de meeste out-of-control situaties gaat deze aanname echter niet moet, wat zou betekenen dat het netwerk handmatig opgedeeld zou moeten worden, met grote verstoringen in het materieelschema en personeelsschema tot gevolg. Met andere woorden, het oplossen van een probleem in een deel van het netwerk zou dan tot nieuwe problemen leiden andere delen van het netwerk. Dit maakt dat de toepassing van decentrale besturing in out-of-control situaties beperkt mogelijk is.

In de introductie van deze thesis bespreken we ook andere mogelijke toepassingen van decentrale besturing: ov-systemen in ontwikkelingslanden of afgelegen gebieden, vervangend busvervoer, hoogfrequente diensten en bijsturing in de eerste fase van een verstoring. Ons onderzoek laat zien dat decentrale beslisregels tot goede prestaties kunnen leiden en dus mogelijk waardevol zijn in de genoemde toepassingen. Echter hebben elk van deze toepassingen hun eigen specifieke eigenschappen, waardoor er verder onderzoek vereist is om de toepasbaarheid van decentrale beslisregels nauwkeuriger te bepalen.

About the author

Rolf Nelson van Lieshout was born in Rotterdam, The Netherlands, in 1994. Rolf holds a Bachelor's degree in Econometrics and Operations Research and a Master's degree in Econometrics and Management Science, with a specialization in Operations Research and Quantitative Logistics. Both degrees were obtained summa cum laude from the Erasmus University Rotterdam. In his research, Rolf applies advanced analytical techniques to improve decision-making in transportation and logistics, with a focus on public transport.

In 2017, Rolf started working 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). Rolf's work has been published in journals such as Transportation Science and Transportation Research Part B. Moreover, he was awarded second place in the 2018 INFORMS RAS student paper award at the INFORMS annual meeting. During his PhD, Rolf regularly visited the department of Process quality and Innovation (π) of Netherlands Railways, as well as the Innovation department of ProRail. He is currently working as an assistant professor at the Eindhoven University of Technology (TU/e) within the Operations, Planning, Accounting and Control (OPAC) Group.

Portfolio

Publications in International Journals

Van Lieshout, R.N. (2021). Integrated Periodic Timetabling and Vehicle Circulation Scheduling. *Transportation Science* 55(3), 768–790.

Van Lieshout, R.N., Bouman, P.C., Van den Akker, J.M. and Huisman, D. (2021). A Self-Organizing Policy for Vehicle Dispatching in Public Transit Systems with Multiple Lines. *Transportation Research Part B: Methodological* 152, 46–64.

Dekker, M.M., Van Lieshout, R.N., Ball, R.C., Bouman, P.C., Dekker, S.C., Dijkstra, H.A., Goverde, R.M.P., Huisman, D., Schaafsma, A.A.M. and Van den Akker, J.M. (2021). A Next Step in Disruption Management: Combining Operations Research and Complexity Science. *Public Transport*. To appear.

Van Lieshout, R.N., Bouman, P.C. and Huisman, D. (2020). Determining and Evaluating Alternative Line Plans in Out-of-Control Situations. *Transportation Science* 54(3), 740–761.

Van Lieshout, R.N., Mulder, J. and Huisman, D. (2018). The Vehicle Rescheduling Problem with Retiming. *Computers & Operations Research* 96, 131–140.

Publications in Conference Proceedings

Van Lieshout, R.N. and Bouman, P.C. (2018). Vehicle Scheduling Based on a Line Plan. *18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik (OASIcs), 15:1–15:14.

Technical Reports

Van Lieshout, R.N., van den Akker, J.M., Mendes Borges, R., Druijf, T. and Quaglietta, E. (2021). Microscopic Simulation of Decentralized Dispatching Strategies in Railways. Econometric Institute Research Papers, EI-1708

Lindner, N. and Van Lieshout, R.N. (2021). Benders Decomposition for the Periodic Event Scheduling Problem. ZIB-Report (21-29).

Conference Presentations

EURO 2021, Athens, Greece (online).

EURO 2019, Dublin, Ireland.

LNMB Conference 2019, Lunteren, The Netherlands.

INFORMS 2018, Phoenix, United States.

CASPT 2018, Brisbane, Australia.

NPCS 2018, Utrecht, The Netherlands.

OR2017, Berlin, Germany.

Awards

RAS Student Paper Award (2nd place), Railway Application Section, INFORMS, 2018.

Best Econometric Thesis Award (Bachelor, 1st place) (Veneficus, Econometric Institute and FAECTOR), 2016.

Teaching activities

Lecturer for the *Econometrics and Operations Research* bachelor course *Linear Programming*, 2021.

Supervision and co-reading of master theses in *Operations Research and Quantitative Logistics*, 2021.

Teaching assistant for the *Econometrics and Operations Research* bachelor courses *Introductory Case Studies, Academic Skills* and *Advanced Programming*, 2018-2021.

Supervision and co-reading of bachelor theses in *Econometrics and Operations Research*, 2018-2019.

PhD Courses and Certificates	
Algorithms and Complexity	Interior Point Methods
Column Generation School 2018	Integer Programming Methods
Algorithmic Game Theory	Robust Optimization
Stochastic Programming	English (CPE certificate)
Convex Analysis for Optimization	Networks and Polyhedra
Networks and Semidefinite Programming	Randomized Algorithms
	Scientific Integrity

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 Gökgöz, Z. Mobile Consumers and Applications: Essays on Mobile Marketing, , Promotors: Prof. G.H. van Bruggen & Dr. B. Ataman, EPS-2021-519-MKT, whttps://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 Platforms*, 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

Hartleb, J., *Public Transport and Passengers: Optimization Models that Consider Travel Demand*, Promotors: Prof. D. Huisman, Prof. M. Friedrich & Dr. M.E. Schmidt, EPS-2021-535-LIS, https://repub.eur.nl/pub/135664

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

Hoogervorst, R., *Improving the Scheduling and Rescheduling of Rolling Stock: Solution Methods and Extensions*, Promotors: Prof. D. Huisman & Dr. T.A.B. Dollevoet, EPS-2021-534-LIS, https://repub.eur.nl/pub/135726

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

Koritarov, V.D., *The Integration of Crisis Communication and Regulatory Focus: Deconstructing and Optimizing the Corporate Message*, Promotors: Prof. C.B.M. van riel, Dr. G.A.J.M. Berens & Prof. P. Desmet, EPS-2021-522-ORG, https://repub.eur.nl/pub/136979

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, Wei., *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

Nikulina, A., *Interorganizational Governance in Projects: Contracts and collaboration as alignment mechanisms*, Promotors: Prof. J.Y.F. Wynstra & Prof. L. Volker, EPS-2021-523- LIS, https://repub.eur.nl/pub/137001

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

Ramezan Zadeh, M.T., *How Firms Cope with Digital Revolution: Essays on Managerial and Organizational Cognition*, Promotors: Prof. H.W. Volberda & Prof. J.P. Cornelissen, EPS-2021-508-S&E, https://repub.eur.nl/pub/135682

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, EPS-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

Sluga, A., *Hour of Judgment: On judgment, decision making, and problem solving under accountability*, Promotors: Prof. F.G.H. Hartmann & Dr. M.A.S. Boksem, EPS-2021-520-F&A, https://repub.eur.nl/pub/136967

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, Prof. J.P. Cornelissen*, Promotor: P.P.M.A.R. Heugens, EPS-2019-483-S&E, https://repub.eur.nl/pub/118760

Stirnkorb, S., *Changes in the Information Landscape and Capital Market Communication*, Promotors: Prof. E. Peek & Prof. M. van Rinsum, EPS-2021-536-F&A, https://repub.eur.nl/pub/136970

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

Subasi, 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

Zhu, S., *Spare Parts Demand Forecasting and Inventory Management: Contributions to Intermittent Demand Forecasting, Installed Base Information and Shutdown Maintenance*, Promotors: Prof. R. Dekker & Dr. W.L. van Jaarsveld, EPS-2021-538-LIS, https://repub.eur.nl/pub/135684

Zon, M. van, *Cost Allocation in Collaborative Transportation*, Promotors: Prof. A.P.M. Wagelmans, Dr. R. Spliet & Dr. W. van den Heuvel, EPS-2021-530-LIS, https://repub.eur.nl/pub/136975

Public transport brings undisputed benefits to modern-day societies. Aside from providing an affordable means to get around, its supreme efficiency in comparison with private transport plays a crucial role in curbing congestion and pollution. Given these advantages, adequate design and operation of public transport systems is of utmost importance.

The first part of this thesis seeks to improve the planning process of public transport operators by integrating planning steps that are traditionally performed sequentially. The first study considers a combination of line planning and vehicle scheduling, and presents methods that estimate how many vehicles are required to operate a line plan, without having to compute a timetable. The second study combines timetabling and vehicle scheduling, and develops a novel optimization model for jointly optimizing a periodic timetable and vehicle circulation schedule.

The second part of this thesis investigates decentralized strategies for operating public transport, with a focus on railway systems. Such strategies could be preferable over conventional centralized and schedule-based control in various scenarios. The first study in this part presents a theoretical analysis of a simple, decentralized strategy for dispatching vehicles. The second study considers the application of decentralized control to out-of-control situations in railways, which includes the development of a solution algorithm to find line plans that are suited for these circumstances. The final study in this thesis tests decentralized dispatching of both vehicles and crew in a microscopic railway simulation.

ERIM

The Erasmus Research Institute of Management (ERIM) is the Research School (Onderzoekschool) in the field of management of the Erasmus University Rotterdam. The founding participants of ERIM are the Rotterdam School of Management (RSM), and the Erasmus School of Economics (ESE). ERIM was founded in 1999 and is officially accredited by the Royal Netherlands Academy of Arts and Sciences (KNAW). The research undertaken by ERIM is focused on the management of the firm in its environment, its intra- and interfirm relations, and its business processes in their interdependent connections.

The objective of ERIM is to carry out first rate research in management, and to offer an advanced doctoral programme in Research in Management. Within ERIM, over three hundred senior researchers and PhD candidates are active in the different research programmes. From a variety of academic backgrounds and expertises, the ERIM community is united in striving for excellence and working at the forefront of creating new business knowledge.

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