Timetable integration in public transport planning

Fonseca, Joao Filipe Paiva

Publication date: 2020

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Timetable integration in public transport planning

PhD Thesis

João Filipe Paiva Fonseca
Kgs. Lyngby, August 2019
Recent years have seen a general increase in the demand for transport and mobility. As the transportation numbers rise, and with the increased efforts in reducing emissions on a global scale, public transport becomes an excellent alternative to private transport due to being cost efficient, environmentally friendly, and for its abilities to reduce congestion. In 2009, the Danish Parliament passed the transport policy agreement entitled “A green transport agreement”, which states that the public transport system should be expanded and made more efficient so that the expected growth in person transport should be able to be captured by the public transport system.

In Denmark, public transport holds a market share of 13% of all commuting, and in 2010 more than 290 million passengers used the railway network. Furthermore, in Copenhagen it was estimated that 65,000 hours are lost every day by passengers waiting for connecting services in public transport. In order to serve increased demand and provide an attractive service for the passengers at an affordable cost, public transport should be both time and cost efficient. Research has through a number of years clearly shown that operations research techniques can help make public transport systems more efficient and thereby increase its competitiveness compared to other transport modes.

Public transport planning comprises a range of planning problems at the strategic, the tactical as well as the operational planning levels. Problems like line planning, timetabling, or vehicle scheduling are traditionally solved independently and sequentially. Today, several models and efficient solution methods for solving these exist. However, solving these problems independently will often result in suboptimal solutions for passengers with respect to the service level offered. Solutions with increased passenger service could be obtained by integrating different planning problems. However, due to the size of real world instances, solving the integrated problem is, usually, not possible without dedicated solution methods, which, in turn, requires huge computational efforts.

The goal of this thesis is to study the integration of timetabling with other planning problems such as line planning, passenger routing, and vehicle scheduling. We propose a number of advanced planning tools, which are able to find integrated solutions to the
complex planning problems seen within public transport. We investigate the application of integrated models to real life case studies and we assess the quality of integrated solutions in comparison to non-integrated solutions.

The thesis studies two problems in the bus transport domain and one problem in the rail transport domain. In the bus transport domain, we integrate timetabling with both vehicle scheduling and passenger routing. The relations between departure and arrival times and how passengers travel in the network are investigated, while taking into account the costs associated with operating the timetables. In the train domain, we investigate partial integration of timetabling and line planning. Specifically, how timetables can be improved in terms of passenger service by allowing changes in train frequencies and changes in stops at stations included in the train journey. We formulate the problems using mathematical models, and we propose heuristic solution methods for all of them. The solution methods are tested in real life case studies of the Greater Copenhagen area for the bus transport problems, and of a subset of the intercity network in The Netherlands for the rail transport problem. We show that the timetables obtained with integrated models provide an increased passenger service while being able to keep operating costs at the same level or within a pre-defined budget.

I Danmark har det kollektive transportsystem en markedsandel på 13% af transporten mellem arbejde og hjem og der blev i 2010 gennemført mere end 290 millioner togrejser i det danske jernbanenet. Det er endvidere tidligere blevet estimeret, at der hver dag spildes 65.000 timer af passagerer, der venter på forbindelser i den kollektive transport. For at imødekomme den forventede vækst og for at kunne tilbyde passagererne en attraktiv service til en fornuftig pris skal den kollektive transport være så effektiv som mulig. Forskningen har gennem de seneste mange år demonstreret, at matematiske optimeringsmetoder kan gøre det kollektive transportsystem mere effektivt og dermed mere konkurrencedygtigt i forhold til andre transportformer.

Målet med denne afhandling er at studere integrationen af køreplansplanlægningen med andre planlægningsproblemer såsom linjeplanlægning, rutevalg og planlægning af køretøjer-nes opgaver (omløbsplanlægning). Vi udvikler en række avancerede planlægningsværktøjer, der kan finde integrerede løsninger til de komplekse planlægningsproblemer man møder inden for den kollektive trafik. Vi undersøger anvendelsen af disse integrerede modeller til at løse virkelige casestudier, og vurderer kvaliteten af de fundne integrerede løsninger sammenlignet med løsninger baseret på ikke-integrerede modeller.

The here presented thesis was carried out at the Division of Management Science of DTU Management - Department of Technology, Management and Economics, at the Technical University of Denmark. It is a partial fulfillment of the requirements for the degree of Doctor of Philosophy (Ph.D.) in Operations Research. The work has been supervised by prof. dr. Allan Larsen and co-supervised by dr. Evelien van der Hurk and dr. Roberto Roberti. Part of the work was conducted at the Department of Supply Chain Analytics at the Vrije Universiteit Amsterdam, between November 2017 and March 2018, and hosted by dr. Roberto Roberti and dr. Gábor Mároti. The thesis deals with the integration of public transport timetabling with other planning problems such as line planning, vehicle scheduling, and passenger routing. Solution methods are proposed for the integration of timetabling, using Operations Research techniques such as mathematical programming and heuristics. The thesis is divided in three parts. First, the Introduction motivates the topic and depicts the overall public transport planning process and integration attempts previously addressed in the literature. Second, three stand-alone research papers written as part of the PhD are presented. Finally, in the Conclusion we present an outlook on the contributions of the thesis and directions for future research.

The PhD project is part of the IPTOP project, which is a collaboration between multiple DTU departments, Danish companies and institutions. The project aims at developing new methods for planning, designing, and optimizing the Danish public transport system. The project is funded by the Innovation Fund Denmark (IFD) for a period of 5 years between 2015-2019. IFD has supported this work solely financially and has not taken part in any research related activities.

Kgs. Lyngby, Denmark, August 2019

João Filipe Paiva Fonseca
Many say that obtaining a PhD is a lonely endeavor. The finish line of my PhD is crowded with people that helped me along the way and to whom I would like to express my most sincere thanks.

A lot happened during my PhD: I married the woman of my life and became a father. Carla and Olivia, I am blessed to have you both in my life, and I would not be able to finish this without your help. I love you.

I was very lucky to have an all-star team of supervisors: Allan Larsen, Evelien van der Hurk, and Roberto Roberti. All three of you were extremely helpful and always had the right words to motivate me towards the next milestone, each in its very own way. You always demonstrated true enthusiasm with what I was doing, even when didn’t go quite as expected. I am very grateful to you all, not only for making me grow as a researcher, but also as a person. Thanks go also to Stefan Røpke, who was on-board in the beginning of the project and helped shaping the structure of the first research activity.

I would like to express my thanks to Roberto Roberti, Gábor Maróti and to all the members of the Supply Chain Analytics group for welcoming me at the Vrije Universiteit Amsterdam and making me feel at home from day one during my external research stay. Thanks go also to Gert-Jaap Polinder at NS for providing me with data. Special thanks go to Christos Orlis, Rosario Paradiso, and (often) Martina Fischetti, with whom I shared offices, meals, philosophical talks, and nights out. My external research stay would not have been possible without the financial support of Stibo Fonden, P.A. Fiskers Fond, Thomas B. Thrige Fond, and Otto Mønsteds Fond.

My PhD was part of a larger research project named IPTOP. I would like to thank all researchers and partners in the project for valuable feedback and productive meetings. Special thanks go to Poul Bayer, Mette Clausen, and Maiken Krogdal Larsen at Movia, and to Rasmus Dyhr Frederiksen and Philip Bagger at Rapidis, who helped me getting data for my research.
At DTU, I was first part of DTU Transport, and later DTU Management. I was surrounded by bright professionals during all these years, and I am extremely thankful for all I learned from you all.

Finally, I would like to thank my parents Margarida and Vitor, my siblings Pedro and Carolina, and my grandparents Fernanda and Leonel. You are far away in distances but very close to my heart. I would also like to thank my family in law, Carlos, Zé, and Tiago for being the best second family one could ask for. To all the rest of my family and friends back in Portugal, thank you!
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Summary</strong></td>
<td>i</td>
</tr>
<tr>
<td><strong>Resumé (Danish Summary)</strong></td>
<td>iii</td>
</tr>
<tr>
<td><strong>Preface</strong></td>
<td>v</td>
</tr>
<tr>
<td><strong>Acknowledgements</strong></td>
<td>vii</td>
</tr>
<tr>
<td><strong>I Introduction</strong></td>
<td>1</td>
</tr>
<tr>
<td>1. Introduction and thesis outline</td>
<td>3</td>
</tr>
<tr>
<td>1.1 Context: IPTOP</td>
<td>5</td>
</tr>
<tr>
<td>1.2 Research questions and scientific contributions</td>
<td>6</td>
</tr>
<tr>
<td>1.2.1 Answering the research questions</td>
<td>6</td>
</tr>
<tr>
<td>1.2.2 Scientific contributions</td>
<td>7</td>
</tr>
<tr>
<td>1.3 Document outline</td>
<td>9</td>
</tr>
<tr>
<td><strong>2 Public transport planning</strong></td>
<td>11</td>
</tr>
<tr>
<td>2.1 Planning horizon for public transport problems</td>
<td>11</td>
</tr>
<tr>
<td>2.1.1 Network Design</td>
<td>13</td>
</tr>
<tr>
<td>2.1.2 Line Planning</td>
<td>13</td>
</tr>
<tr>
<td>2.1.3 Timetabling</td>
<td>15</td>
</tr>
<tr>
<td>2.1.4 Vehicle Scheduling and Rolling Stock Planning</td>
<td>17</td>
</tr>
<tr>
<td>2.1.5 Crew Scheduling and Crew Rostering</td>
<td>18</td>
</tr>
<tr>
<td>2.1.6 Disruption Management and Real Time Control</td>
<td>19</td>
</tr>
<tr>
<td>2.2 Integration of public transport planning problems</td>
<td>20</td>
</tr>
<tr>
<td>2.3 Final remarks</td>
<td>22</td>
</tr>
</tbody>
</table>
## II Research Work

### 3 A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

#### 3.1 Introduction

#### 3.2 The integration of timetabling and vehicle scheduling

#### 3.3 Literature review

- 3.3.1 Multiple depot vehicle scheduling
- 3.3.2 Transit network timetabling
- 3.3.3 Integrated timetabling and vehicle scheduling

#### 3.4 A mathematical model for the IT-VSP

- 3.4.1 Lines and trips
- 3.4.2 Timetable modifications
- 3.4.3 Passenger transfers
- 3.4.4 Compatible trips
- 3.4.5 Vehicle scheduling
- 3.4.6 Passenger and operating costs
- 3.4.7 Decision variables and mathematical model

#### 3.5 A matheuristic approach

- 3.5.1 Outline
- 3.5.2 Trip selection strategies

#### 3.6 Case study

#### 3.7 Computational experiments

- 3.7.1 Parameter tuning
- 3.7.2 Computational performance
- 3.7.3 Transfers and operational costs

#### 3.8 Conclusion

### 4 Passenger service optimization through timetabling with free passenger route choice

#### 4.1 Introduction

#### 4.2 Literature Review

#### 4.3 The IPAT-VSP

- 4.3.1 Example

#### 4.4 Solution method

- 4.4.1 Integrated timetabling and vehicle scheduling – the IT-VSP matheuristic approach
- 4.4.2 Passenger route choice model – the PTTA model
- 4.4.3 IPAT-VSP matheuristic – the MHeuPA

#### 4.5 Case study

#### 4.6 Computational experiments

- 4.6.1 Results for the base scenario
- 4.6.2 Designing timetables - changes in the public transport network
- 4.6.3 Changes in passenger demand

#### 4.7 Conclusions and future research

#### 4.8 Appendix 1 - IT-VSP mathematical model
## 5 Aperiodic public transport timetabling with flexible line plans

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Introduction</td>
<td>98</td>
</tr>
<tr>
<td>5.2 Literature Review</td>
<td>100</td>
</tr>
<tr>
<td>5.2.1 Line Planning</td>
<td>100</td>
</tr>
<tr>
<td>5.2.2 Timetabling</td>
<td>101</td>
</tr>
<tr>
<td>5.2.3 Integrated line planning and timetabling</td>
<td>102</td>
</tr>
<tr>
<td>5.3 Problem description</td>
<td>103</td>
</tr>
<tr>
<td>5.3.1 Example</td>
<td>105</td>
</tr>
<tr>
<td>5.4 Solution method</td>
<td>106</td>
</tr>
<tr>
<td>5.4.1 Lower Bound method</td>
<td>108</td>
</tr>
<tr>
<td>5.4.2 Construction heuristic</td>
<td>110</td>
</tr>
<tr>
<td>5.4.3 Track section fixer and solution check</td>
<td>112</td>
</tr>
<tr>
<td>5.4.4 Operators</td>
<td>113</td>
</tr>
<tr>
<td>5.4.5 Evaluation</td>
<td>116</td>
</tr>
<tr>
<td>5.4.6 Acceptance of solutions and restarting the search</td>
<td>116</td>
</tr>
<tr>
<td>5.5 Case study</td>
<td>117</td>
</tr>
<tr>
<td>5.6 Computational experiments</td>
<td>119</td>
</tr>
<tr>
<td>5.6.1 Tuning experiments</td>
<td>120</td>
</tr>
<tr>
<td>5.6.2 Analysis of results</td>
<td>124</td>
</tr>
<tr>
<td>5.7 Conclusions and future research</td>
<td>132</td>
</tr>
</tbody>
</table>

### III Conclusion

6 Conclusion 135

Bibliography 139
Part I

Introduction
Recent years have seen a general increase in demand for transport and mobility. At the same time, environmental awareness is growing and there is an international focus on reducing CO\textsubscript{2} emissions. According to the European Union, the transport sector is responsible for almost one quarter of the total greenhouse gas emissions in Europe and is the main cause of air pollution in cities (European Union 2019). Public transport is an excellent alternative to private transport due to being cost efficient, environmentally friendly, and for its abilities to reduce congestion. In 2009, the Danish Parliament passed the transport policy agreement entitled “A green transport agreement”, which states that the public transport system should be expanded and made more efficient so that the expected growth in person transport should be able to be captured by the public transport system (IPTOP 2018). In Denmark, commuting distances have reached an average value of 22 kilometers per day in 2017 (Statistics Denmark 2017). Public transport holds a market share of 13\% of all commuting, and in 2010 more than 290 million passengers used the railway network (Ministry of Transport 2011). Furthermore, in Copenhagen it was estimated that 65,000 hours are lost every day by passengers waiting for connecting services in public transport (Petersen et al. 2013). In order to serve increased demand and provide an attractive service for the passengers at an affordable cost, public transport should be both time and cost efficient. Research has through a number of years clearly shown that operations research techniques can help making public transport systems more efficient and thereby increase its competitiveness compared to other transport modes.

In public transport planning, decisions are made at different levels, strategic, tactical, and operational. Traditionally, the decisions to make follow a sequence driven by how decisive they are. For example, before it is decided what stations a train should visit it has to be decided where to build tracks and stations. The traditional sequence is addressing network design first, followed by line planning. Then, the timetable is produced, followed by vehicle
scheduling or rolling stock planning. Next, to cover the vehicle schedules and rolling stock plans, crew planning problems are solved, consisting of crew scheduling and crew rostering. Finally, disruption management and real time control are the last steps in the planning process. In order to transport large numbers of passengers in a fast and reliable way, public transport networks must be planned as efficiently as possible. Public transport planning is a complex process and historically there has been a limited tradition in using transport models and mathematical optimization tools for public transport planning (Ceder, 2007). The use of decision support tools can help make public transport more efficient and attractive by providing faster planning, cheaper operations, and offer higher levels of passenger service. Research in the fields of operations research and optimization has paid a large attention to public transport planning in recent years. However, a big part of that research is focused on reducing operating costs rather than focused on the quality of service provided to the passengers. Furthermore, a large body of research is devoted to solving the public transport planning problem as broken down into smaller subproblems that are easier to handle, model, and solve, while benefits can be expected from an integrated approach. Addressing these problems individually yields loss of optimality, given that the solutions obtained may be optimal for the single problem but not so good when considering the big picture.

This thesis investigates how integrated public transport timetabling can potentially improve passenger service at the same or lower operating costs, or under a given extra budget for operating costs. Given an existing infrastructure, lines and suggested frequencies to be operated, the timetabling problem consists of assigning departure and arrival times to each service to all stations or stops visited by that service. In the public transport planning process, the timetabling problem is placed right after the line planning problem, which determines lines and hourly frequencies, and right before the vehicle scheduling problem, which assigns vehicles to services. The sequential approach ignores the fact that these problems are highly dependent on each other. The lines created in the line planning problem will influence how passengers travel in the network, partially determine what stations have large passenger flows, and indirectly influence the vehicle schedules. Timetabling also influences which services can be included in the same vehicle schedule, by determining the start and end times of services. If passengers have multiple options for going from their origin to their destination, timetabling influences directly their route choice. A passenger might choose to travel via a longer route if for example a shorter route that involved transferring yields a high transfer waiting time.

Integrating timetabling with line planning, passenger routing, and vehicle scheduling can potentially find solutions with higher passenger service than the solutions found when solving the problems independently. In this thesis we study the integration of these problems, provide mathematical models and solution approaches to them, and report their performance on real life case studies. Results indicate that passenger service can be improved at lower or within budget operating costs, proving the advantages of considering integrated problems in public transport planning.
1.1 Context: IPTOP

The work developed during this PhD is part of the Integrated Public Transport Optimization and Planning (IPTOP) research project, a project funded by Innovation Fund Denmark (IFD) during the 5-year period 2015-2019. IPTOP is a research project across multiple departments at DTU and conducted in close collaboration with Danish transport authorities and companies: BaneDanmark (railway infrastructure manager), Trafikstyrelsen (Danish Transport, Construction and Housing Authority), MOVIA (main Danish bus operator) and DSB (main railway operator), and Rapidis (transport software company). Recent years’ fast technological advances allowed large amounts of data to be available, which motivate a renewed look at large scale optimization methods for public transport. The project aims at an integrated approach to public transport planning and the main goal is to develop new methods for planning, designing, controlling, and optimizing Danish public transport systems, with a focus on passenger perspectives without disregarding the optimization of operations. IPTOP focuses on recent advances in the fields of data analysis, mathematical optimization, operations research, and simulation. Figure 1.1 shows the structure of IPTOP, all its work packages, and links between them.

This PhD is part of WP5 Optimization of schedules and is focused on develop new methodologies to integrate timetable optimization with other public transport planning problems such as line planning, passenger routing, and vehicle scheduling. The objective is to take into account both passenger service considerations (consumer surplus) and operating costs. Furthermore, synchronization across different modes of transport is considered. The methodologies developed use operations research techniques such as mathematical
modeling, heuristics algorithms, and matheuristics. The thesis analyses case studies in the Greater Copenhagen area, working with real life instances provided by MOVIA, and in a subset of the intercity network in The Netherlands, with real life data provided by NS (main rail operator in The Netherlands).

1.2 Research questions and scientific contributions

In this thesis we investigate whether it is possible to integrate timetabbling with different planning problems in an efficient way. In other words, we study whether or not mathematical models can include the necessary level of detail to represent the problems fully, and remain an efficient way to integrate them. It may be that the formulations to solve these problems become so complex that they are no longer useful. We aim at finding solution methods that allow integrating different planning problems using reasonable computational time. Timetabling is a tactical planning problem which is solved months in advance of operation, therefore, we consider a computation time between 3–12 hours as reasonable for the real life instances used in this thesis. Finally, it is important to assess the quality of solutions obtained not only in terms of operating costs but also, and more importantly, in terms of how they serve passengers. We determine whether the integrated solutions are in fact better in terms of passenger service than solving timetabling independently from the other problems. To do this, we compare the solutions obtained with base input timetables and with timetables generated using non integrated approaches. We consider the following research questions, which are answered in the context of each individual research paper of the thesis:

1. Can timetabling be efficiently integrated with line planning, passenger routing, and vehicle scheduling?

2. Can heuristics and matheuristics be used to solve real life instances of the integrated approaches in reasonable computational time?

3. Are the solutions to the integrated approaches in fact better, from a passenger perspective, than solving timetabling independently?

1.2.1 Answering the research questions

We answer the first research question by integrating timetabling with vehicle scheduling (Chapter 3), with vehicle scheduling and passenger routing (Chapter 4), and with line planning and passenger routing (Chapter 5). We formulate problems with more detail than previously done in the literature, and include highly detailed passenger assignment which has not been done before in integrated timetabling problems. Therefore, we show
that timetabling can be efficiently integrated with other planning problems in public transport.

The thesis answers the second research question by creating solution approaches that provide good quality solutions for real life case studies in all three problems considered. They do so within three hours of computational time, which is considered reasonable for timetable planning problems of this size. Therefore, we show that it is feasible to find good quality solutions for integrated timetabling problems in a real case study setting.

The solutions that we find with our solution approaches are better than solving timetabling independently, hence we answer the third research question. In Chapter 3, we show that the integrated approach solutions reduce transfer costs by up to 10% without increasing operating costs. In Chapter 4, the addition of detailed passenger routing allows savings of approximately 40,000 DKK in passenger costs expressed in value of time, when compared with solutions that integrate timetabling and vehicle scheduling but use static passenger routing. Finally, in Chapter 5 allowing line planning changes, using an extra budget of operating costs, leads to solutions with considerable savings in passenger service when compared to solutions that keep line plans unchanged.

1.2.2 Scientific contributions

This thesis considers three distinct scientific research projects that answer the three research questions posed above. Each of the projects is included in a self-contained chapter, which is either published, submitted, or soon to be submitted to an international peer-reviewed journal. The problems considered in the projects are formally described, modeled using mathematical programming, and we propose solution methods to solve real life instances. Therefore, the scientific contributions to the state of the art of timetabling integration in public transport are threefold: provide mathematical models for new or existing problems in the literature, propose new methodologies and solution frameworks to solve them, and provide insights on their performance on real life case studies.

In Chapter 3, we address the Integrated Timetabling and Vehicle Scheduling Problem (IT-VSP), with the objective of minimizing operating costs and excess transfer times given fixed demands for transfers. The problem is formulated as a bi-objective mixed integer programming problem that allows timetable modifications to an initial timetable. Timetable modifications consist of shifts in departure times and addition of dwell times at stations with transfers. The timetables resulting from the modifications must respect a set of headway constraints. We propose an iterative matheuristic solution method that uses the mathematical formulation of the problem. Each iteration allows modifications to a subset of trips in the timetable, while all other trips remain fixed. Different strategies for selecting the subset of trips are compared, and the solution method is tested in a real life case study in the Greater Copenhagen area. Results indicate that the solution method is able to find better solutions faster than a commercial solver and that the addition of
dwell times allows reducing excess transfer times at fixed operating costs.

With Chapter 3, we contribute to the state of the art by: (i) presenting a mathematical formulation for the IT-VSP that allows a wider set of timetable modifications than other papers in the literature, (ii) proposing a matheuristic that is successful in integrating timetable modifications with vehicle scheduling, generating good quality solutions for real life instances in reasonable computational times and faster than a commercial solver, and (iii) testing the method in a real life case study for the express bus network in the Greater Copenhagen area. The work in Chapter 3 has been disseminated as follows:

- a paper published in *Transportation Research Part B: Methodological* (Fonseca et al., 2018b);
- a presentation at the 28th *European Conference on Operational Research (EURO 2016)*, in Poznan, Poland (Fonseca et al., 2016);
- a presentation at the *Annual Workshop of the EURO Working Group on Vehicle Routing and Logistics optimization (VeRoLog 2017)*, in Amsterdam, The Netherlands (Fonseca et al. 2017).

In Chapter 4 we study the *Integrated Passenger Assignment, Timetabling and Vehicle Scheduling Problem* (IPAT-VSP), which integrates timetabling and vehicle scheduling with free route choice of passengers. The objective of the IPAT-VSP is to maximize passenger service by minimizing weighted travel time, while keeping the operating costs at the same level as in an initial timetable. Weighted travel time is composed by in-vehicle time, initial waiting time, and transfer time. Operating costs are defined by the minimal cost vehicle schedules required to operate the timetable. We propose a solution approach that integrates an adapted version of the matheuristic in Chapter 3 with a passenger assignment model. Different forms of integrating the two models are evaluated and we test the method in a case study in the Greater Copenhagen area. Results indicate that the solution method consistently leads to timetables with lower weighted travel times in comparison to the initial timetable and to solutions obtained with the model assuming fixed passenger route choice. Furthermore, we show that suggesting potentially interesting transfers for passengers allows the model to find timetables with higher passenger service than providing precise passenger route choice information on the current timetable.

With Chapter 4, we contribute to the state of the art by: (i) integrating free route choice in a timetabling and vehicle scheduling problem, modeling passenger assignment in a more detailed way than available so far in the timetabling literature, (ii) proposing a modular matheuristic approach that combines two state of the art models and is able to find good solutions in reasonable computational times, and (iii) showing that, for a real life case study in the Greater Copenhagen area, the inclusion of free route choice results in timetables with higher passenger service while maintaining the same operating costs. The work in Chapter 4 has been disseminated as follows:
In Chapter 5, we address the *Aperiodic public transport Timetabling problem with flexible Line Plans* (AT-LP) for rail public transport. We include line planning flexibility by allowing changes in the suggested hourly frequencies (by adding and removing services) and in the stopping patterns (by skipping stops at stations). The AT-LP minimizes a weighted sum of costs associated with passenger service: in-vehicle time, initial waiting time, transfer time, and transfer penalties. Timetable feasibility is imposed mainly by respecting upper and lower bounds on the allowed line planning modifications, and by respecting safety headway constraints. Passenger costs are calculated by routing the passengers in an input origin-destination-time matrix using shortest path computations. We propose a heuristic solution approach to solve the AT-LP that starts by creating a feasible timetable and then applies timetable and line planning modifications to that timetable. To escape local minima, we propose two methods to restart the search and accept worse solutions. We use a lower bound method on passenger costs to assess the quality of the solutions obtained. A subset of the IC network in The Netherlands is used as a case study, and results show that including line planning modifications allows finding solutions with improved timetable quality in terms of passenger service, given a budget on operating costs.

With Chapter 5, we contribute to the state of the art by: (i) defining the AT-LP as an aperiodic timetabling problem that includes some elements of line planning and passenger routing, (ii) proposing an iterative heuristic to solve the AT-LP and propose a passenger cost lower bound method that is valid for all tree shaped networks, and (iii) showing how different versions of the solution approach behave in a real life case study of a subset of the IC network in The Netherlands, and showing that with the inclusion of line planning modifications our method finds timetables with lower passenger costs than with timetable modifications only. The work in Chapter 5 has been disseminated as follows:

- a manuscript, which is currently under preparation and will soon be submitted to an international peer-reviewed journal (Fonseca et al. 2019a);
- a presentation at the *30th European Conference on Operational Research (EURO 2019)*, in Dublin, Ireland (Fonseca et al. 2019b).

### 1.3 Document outline

The remainder of this thesis is organized as follows. Chapters 3–5 are self contained papers and can be read independently.
Chapter 2 Public transport planning. This chapter introduces the reader to the domain of public transport planning, providing the necessary knowledge for understanding the underlying problems and introducing terminology to be used in the remainder of the thesis. This chapter provides also a review on previous work on the optimization of public transport planning problems and integration of different planning problems.

Chapter 3 A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling. This chapter addresses the integration of bus timetabling and vehicle scheduling. (Published in Transportation Research Part B: Methodological)

Chapter 4 Passenger service optimization through timetabling with free passenger route choice. This chapter integrates passenger route choice with bus timetabling and vehicle scheduling. (Submitted to OR Spectrum: Quantitative Approaches in Management)

Chapter 5 Aperiodic public transport timetabling with flexible line plans. This chapter integrates train timetabling with elements of line planning and passenger routing. (To be submitted)

Chapter 6 Conclusions. This chapter concludes the thesis with a summary of the main findings and concluding remarks. It also presents possible directions for future research in Public Transport Planning.
In this chapter we introduce the reader to the public transport planning process. We start by explaining how transport planning is usually decomposed into different planning problems, which are traditionally solved independently. Then, we explain each of those problems and refer to relevant literature to model and solve them using operations research techniques. Finally, we explain the benefits of integrating selected public transport planning problems and review previous research that addresses the topic.

### 2.1 Planning horizon for public transport problems

Public transport planning is a complex process that comprises decisions at the strategic, tactical, and operational levels. Given its complexity, usually it is divided into smaller subproblems that are easier to solve from a computational point of view. However, for real life scenarios these problems remain highly complex combinatorial problems, and the operations research community has been interested in the topic for a number of decades. We refer to Bussieck et al. (1997b), Huisman et al. (2005), and Borndörfer et al. (2018b) for reviews on operations research methods applied to rail public transport, and to Desaulniers and Hickman (2007) and Guihaire and Hao (2008a) for reviews on operations research methods applied to bus public transport. Figure 2.1 shows the different subproblems in public transport planning, the sequence they are usually solved in, and which planning level they belong to. In general, solutions for an upstream problem serve as input to the downstream problems.

The problems in strategic planning – Network Design and Line Planning – are usually
planned with lengthy time horizons, since they often involve resource acquisition. Especially for rail public transport, Network Design may involve building new infrastructure. The objective of problems at the strategic level is to maximize quality of service (given demand forecasts) while imposing budget restrictions. Tactical planning problems – Timetabling, Vehicle Scheduling / Rolling Stock Planning, and Crew Planning (Scheduling and Rostering) – are concerned with resource allocation once the network is fixed. These problems are solved more often than strategic problems, and their objective is to utilize resources as efficiently as possible given resource capacity constraints, security constraints, and labor constraints. Finally, operational planning problems deal with day-to-day situations that change what was planned in the tactical level. Their objective is to return the public transport system to its normal state as soon as possible and minimize the negative effect that disruptions and other disturbances cause on passengers.

Figure 2.1 suggests that these problems are solved sequentially, from top to bottom. However, as pointed out in Bussieck et al. (1997b), it might be needed to iterate back to re-solve upstream problems. For example, it might occur that the optimal solution to the line planning stage does not allow finding a feasible timetable due to track section conflicts. In that scenario, the line planning problem must be re-solved with additional constraints that ensure feasibility at the timetabling stage.
2.1 Planning horizon for public transport problems

2.1.1 Network Design

The first problem to solve at the strategic level of public transport planning is the network design problem. Network design influences all the planning problems that come after it, since it defines the public transport network (PTN). The PTN can be defined as a graph \((V, E)\), where the set of nodes \(V\) corresponds to stations or stops and the set of edges \(E\) corresponds to direct connections between stations or stops. In the case of rail transport, the edges correspond to tracks, while in the case of bus transport these correspond to the actual road network. The network design problem defines the location of stations and stops, and defines which stations and stops should be directly connected. Some of these decisions, such as the construction of a new rail line, demand major public and private investments and thus, involve a wide range of decision makers including government, regional institutions, city councils, and public transport operators. Although there are optimization models for network design in the literature, the limited number of studies on the topic may be explained by the difficulty to model risk and uncertainty associated with strategic decisions of this magnitude. Another issue is how to evaluate the quality of a solution to the network design problem, when political constraints play an important role in the decision.

The network design problem is proven NP-hard in Magnanti and Wong (1984). We direct the reader to Desaulniers and Hickman (2007) and Ibarra-Rojas et al. (2015) for literature overviews of the network design problem.

2.1.2 Line Planning

The line planning problem takes as input the PTN defined in the network design phase, and passenger demands to be serviced. Consider a public transport network represented as a finite and undirected graph \(PTN = (V, E)\), defined by a node set \(V\) representing stops or stations, and an edge set \(E = \{(u, v) : u, v \in V\}\), representing the existence of direct connections between two nodes. A line \(l\) is a simple path in the PTN, defined by a sequence of stations pairwise connected by edges, or equivalently by a sequence of adjacent edges. Assume that a line pool \(L\) composed of different lines exists, and assume as input a set \(R \subseteq V \times V\) of all origin-destination pairs \((a, b)\), where \(w_{ab}\) denotes the number of passengers willing to travel from station \(a\) to station \(b\). The line planning problem in public transport consists of choosing a subset of lines \(L \subseteq L\) and hourly frequencies \(f_l\) for each line \(l \in L\), such that travel demand is accommodated. Type of service and type of vehicles to use are often decisions of the line planning problem as well, since for example not all trains can operate a high-speed line. Feasibility of a line plan is mainly concerned with respecting the minimum and maximum frequencies allowed on each edge \(e \in E\) of the PTN (Bussieck et al., 1997a):
Schöbel (2012) presents an overview of different modelling and solution methods for solving line planning. Consider the PTN depicted in Figure 2.2, with the set of stations $V = \{1, 2, 3, 4, 5, 6\}$ and the set of undirected edges $E = \{(1, 3), (2, 3), (3, 4), (4, 6), (4, 6)\}$. Consider also that there is demand for all possible pairs of nodes. Coverage can be guaranteed by a minimum of two bi-directed lines, for example 1-3-4-5 and 2-3-4-6, but in principle all paths in the PTN can form a line, even if composed by a single edge. Considering only lines that stop at all intermediate stations, the PTN of Figure 2.2 contains 15 possible lines. When stopping patterns, frequencies, and types of service are also considered, the number of different options increases. Bussieck (1998) proved that the line planning problem is NP-complete. The definition of a line pool $L$ limits the number of options and thus makes the problem more tractable.

Mathematical models and heuristics for solving line planning problems can be divided according to what objective they address: cost oriented models, passenger oriented models, and models that address other aspects of line planning. Cost oriented models are aimed at minimizing the operating costs of line plans, subject to service constraints and capacity requirements. Claessens et al. (1998) present an algorithm based on constraint satisfaction and a branch-and-bound procedure to solve the problem. Goossens et al. (2006) present an extension of the cost-minimization model, considering stopping pattern determination. Other cost oriented models for line planning can be found in Bussieck et al. (2004), Goossens et al. (2004), and Borndörfer et al. (2018a). Passenger oriented models consider the effect of line plans on passenger perspectives and optimize line planning towards better serving passengers. Examples of passenger oriented line planning models are presented in Bussieck et al. (1997a) and Schöbel and Scholl (2006). The objective in Bussieck et al. (1997a) is to maximize the number of direct travelers – passengers that can travel from origin to destination without having to transfer. In Schöbel and Scholl (2006), the objective is to minimize the generalized travel time of passengers, which is a weighted sum of different travel time components and penalties for transferring. Recent research...
on line planning include other aspects, such as the generation of line pools (Gattermann et al., 2017), different stopping patterns (Burggraefe et al., 2017; Bull et al., 2018), and modeling line planning as a game (Schiewe et al., 2019).

### 2.1.3 Timetabling

The timetabling problem takes as input the lines, frequencies, and stopping patterns defined in the line planning stage. The problem consists of assigning time points to arrivals to and departures from visited stations or stops. Timetabling defines not only the published timetable, which is what is disseminated to the general public, but also defines other aspects such as dwell times (stopping time of a vehicle at a stop or station) or entry times at track sections. In railway oriented timetabling it is important to consider the capacity of the track system, with timetabling being solved together or after a routing step. This is due to the fact that train paths are not defined by the line planning step (Lusby et al., 2011). With fixed train paths, the problem can be addressed as a resource-constraint project-scheduling problem (Mascis and Pacciarelli, 2002). Papers in this domain use different objective functions, such as in Caprara et al. (2002) where the deviation to a reference timetable is minimized.

Timetabling problems can be divided in periodic and aperiodic. In periodic timetabling, the timetable is determined for a fixed period of time, usually a peak-hour, and is then repeated throughout the rest of the day. Aperiodic timetabling disregards this periodicity of departure times and plans a timetable for the whole planning horizon. Recent research investigates combinations of periodic and aperiodic timetabling, for example in Robenek et al. (2017). Periodic timetables are common practice in train timetabling, especially in Europe. Passengers tend to prefer these timetables since they are easier to remember. However, with the advent of new technologies for journey planning, passengers rely more and more on information retrieved from mobile apps and less on remembering the actual departure times of trains. Aperiodic timetabling increases the freedom in scheduling services and makes the problem easier to solve.

Public transport timetabling is widely studied in the Operations Research literature, with several models and heuristics available both for periodic and aperiodic timetabling. Timetabling appears as early as Szpigiel (1973), where the author presents a mixed integer formulation, adapted from the general job shop scheduling problem, with the objective of minimizing total travel time of scheduled trains. A branch-and-bound algorithm is presented to solve a case study of a single line in the Brazilian rail network. A complete overview on previous research work related with periodic and aperiodic timetabling is available in Lusby et al. (2011).

Most of the approaches to solve periodic timetabling are based on the NP-complete Periodic Event Scheduling Problem (PESP) introduced by Serafini and Ukovich (1989). Assuming fixed routes for passengers, the objective of the PESP is to minimize travel
time. Different constraints can be included in the PESP by exploring the modeling power of the formulation. Driving times, waiting times at stations, passenger transfers, and security headway constraints are added to the PESP by Odijk (1996) and Nachtigall (1998). Nachtigall (1998) introduce a cycle based integer programming formulation that is extended later by a variety of authors. Variability in driving times is added to the PESP by Kroon and Peeters (2003). Peeters (2003) propose alternative objective functions to the PESP (minimization of number of vehicles needed, maximization of robustness, and minimization of number of violated constraints) and propose ways to add departure synchronization and station capacities to the constraints of the model. Later, Liebchen and Möhring (2007) extend the modeling power of the PESP to include coupling and decoupling constraints, fixed times for certain events, and bundling of lines. The modeling power of the PESP is matched by the difficulty in solving it. The literature is rich in solution approaches and algorithms to solve it in computational times that are compatible with real life cases. The types of solution methods include genetic algorithms (Nachtigall and Voget, 1996, 1997), constraint generation (Odijk, 1996), modulo simplex heuristics (Nachtigall and Opitz, 2008; Goerigk and Schöbel, 2013; Goerigk and Liebchen, 2017), and satisfiability problems (Großmann et al., 2012; Kümmeling et al., 2015; Matos et al., 2018). For a recent overview of timetabling models, with a special focus on PESP models, we direct the reader to Sels et al. (2016).

Aperiodic timetabling models for trains are presented in the literature by Barrena et al. (2014a, b), Niu et al. (2015), Robenek et al. (2016), and Yin et al. (2017). Barrena et al. (2014a) minimize passenger average waiting time and present three formulations for modeling the problem, with different levels of detail. Barrena et al. (2014b) introduce two mathematical (non-linear) programming formulations for a single rail line under a dynamic demand pattern, generalizing the non-periodic train timetabling problem, with the same objective as Barrena et al. (2014a). The aperiodic timetabling problem considered in Niu et al. (2015) allows trains on a single corridor to skip stations according to predefined stopping patterns. The problem objective is to minimize total passenger waiting time at stations. Robenek et al. (2016) maximize the profit of a train company while keeping passenger satisfaction at the same level. Passenger satisfaction includes in-vehicle time, waiting time, number of transfers and scheduled delays. Yin et al. (2017) minimize operating costs and passenger waiting times in a metro corridor, using an aperiodic timetabling model that also includes dynamic passenger demands.

Research on public transport timetabling for buses, which is also referred to as Transit Network Timetabling (TNT), is also abundant in the literature. Common TNT objectives considered in the literature are minimization of excess transfer times (transfer waiting times), maximization of synchronization, or multi-objective approaches.

Minimization of excess transfer times as an objective to the TNT is introduced by Klement and Stemme (1988), who model the problem as a Quadratic Semi Assignment Problem (QSAP) and propose a constructive heuristic to solve it. Several other solution approaches are later proposed to solve the QSAP for the TNT: branch-and-bound local search and simulated annealing (Domschke 1989), combining simulation and mathemat-
2.1 Planning horizon for public transport problems

The planning horizon for public transport problems has been addressed using various optimization techniques. Bookbinder and Desilets (1992) introduced regret heuristics and improvement procedures combined with simulated annealing and tabu search (Daduna and Vöß 1995). Other authors start from an initial timetable and adjust it to minimize excess transfer times. These authors use linearization of the QSAP (Schröder and Solchenbach 2006), mixed integer programming combined with lagrangean heuristics (Wong et al. 2008), or genetic algorithms (Shafahi and Khani 2010).

TNT problems with the objective of maximizing synchronization are found in Ceder et al. (2001), Liu et al. (2007), and Ibarra-Rojas and Rios-Solis (2012). All these three papers consider timetables constructed based on upper and lower bounds for the headways, allowing aperiodic timetables. The authors model the problem as a MIP (Ceder et al. 2001), and add a pre-processing stage that eliminates variables and constraints (Ibarra-Rojas and Rios-Solis 2012). Solution approaches presented are heuristic algorithms (Ceder et al. 2001), nested tabu search (Liu et al. 2007), and iterated local search (Ibarra-Rojas and Rios-Solis 2012).

The TNT has also been considered as a multi-objective problem. Objectives considered are minimizing the cost of transfers and costs caused by deviations from an initial timetable (Kwan and Chang 2008), minimizing empty seat penalties and expected passenger waiting times (Hassold and Ceder 2012), minimizing expected total passenger waiting time and variations in vehicle occupancy (Liu and Ceder 2016), and maximizing the number of passengers benefited by smooth transfers and minimizing deviations from an initial timetable (Wu et al. 2016).

2.1.4 Vehicle Scheduling and Rolling Stock Planning

Once a timetable is created, there is a set of services (or trips) for each line that must be serviced. The vehicle scheduling (for buses) and rolling stock planning (for trains) problems create vehicle routes to service the timetabled trips. These routes are composed not only by the timetabled trips (the ones that figure in the timetable), but also by empty trips, where the vehicle transports no passengers: pull-out, pull-in, and deadhead. Pull-out trips occur when a vehicle is traveling from the depot to the start of a timetabled trip, pull-in trips occur when a vehicle is traveling from the end of a timetabled trip to the depot, and deadhead trips occur when a vehicle is traveling from the end of a timetabled trip to the start of another timetabled trip.

Vehicle scheduling problems are widely studied in the literature, and survey papers are available in Daduna and Paixão (1995) and Bunte and Kliwer (2009). Vehicle scheduling problems can be categorized according to how many vehicle types they consider (one or multiple) and in terms of how many depots they include (one or multiple). For the simplest case, with a single type and a single depot, Bertossi et al. (1987) demonstrate that the problem is solvable in polynomial time. The problem is also solvable in polynomial time when the number of vehicles is restricted or other constraints limit the number of
different combinations of routes. However, with multiple depots, the Multi Depot Vehicle Scheduling Problem (MDVSP), which is a well known problem in the field of operations research, is proven to be NP-hard. A detailed description of the problem is available in Desrosiers et al. (1995), including also a review on the literature existent at the time. Desaulniers and Hickman (2007) provides a wider survey on the MDVSP and extensions of the problem. With the help of column generation, Kliwer et al. (2006) are able to solve to optimality real life MDVSP problems with up to 7000 trips, given that these instances have a structure that helps finding optimal solutions in reasonable computational time. Carpaneto et al. (1989) presents a set of randomly generated instances, from which Hadjar et al. (2006) is able to solve the ones with up to 800 trips to optimality using an exact branch-and-bound approach. Due to the complexity of real life instances of the MDVSP, the problem is usually solved heuristically. In Pepin et al. (2009), a comparison is presented between five different heuristic approaches to solve instances with 500, 1000, and 1500 trips, and with 4 or 8 depots. The authors conclude that if enough computational time is available, then the best performing approach is truncated column generation. However, given restricted computational times, large neighborhood search is the best performing heuristic in finding good solutions fast. Rolling stock scheduling encompasses a wider set of decisions, mainly due to having different train compositions to select from. A train composition is composed by different train units that can be coupled and decoupled according to what trip they are servicing. The order in which train units are coupled is also important, since it will define in what order and in what combinations they can be uncoupled at the next stations. Since in this thesis we address only vehicle scheduling for buses, we direct the reader to Schrijver (1993), Alfieri et al. (2006), Fioole et al. (2006), and Peeters and Kroon (2008), where multiple formulations and solution approaches are proposed for rolling stock scheduling.

2.1.5 Crew Scheduling and Crew Rostering

Crew members play a major role in carrying out public transport operations. Crew members range from drivers to train conductors or other workers serving passengers. Each crew member has a specific set of skills and qualifications that have to match the duties they are assigned to. On the other hand, each task to be performed requires a certain number of crew members with specific skills or qualifications. Crew planning can be divided in Crew Scheduling and Crew Rostering. Crew Scheduling defines generic daily duties (start and end time) to cover a set of vehicle schedules determined in the vehicle scheduling stage. The main objective of Crew Scheduling is to minimize the costs associated with the duties, while satisfying labor regulation constraints: limited duration of work load, maximum intervals between meal breaks, maximum consecutive work time, and starting times for duties. The literature
2.1 Planning horizon for public transport problems

considers different objectives for the Crew Scheduling problem, e.g.: minimization of number of duties, minimization of idle times, minimization of penalty incurred by violating constraints, or minimization of uncovered duties. Models used for solving Crew Scheduling problems are often based on set partitioning and set covering, which are then solved using column generation and branch-and-bound.

Crew Rostering consists of assigning the daily duties created in the Crew Scheduling phase to specific crew members over a period of time, for example one month. Again, labor rules play a major role in the feasibility of solutions to the Crew Rostering problem, namely in terms of number of days-off, number of days-off on weekends and holidays, number of consecutive work days, number of night shifts over a certain period of time, minimum number of resting hours between shifts, or equity among different rosters for workers of the same type. Models to solve Crew Rostering are also based on set partitioning and set covering and usual objectives are minimization of salary costs, minimization of idle times, or minimization of penalties incurred by violating constraints.

A review of Crew Scheduling literature for bus systems is available in [Wren and Rousseau (1995)], while Crew Scheduling and Crew Rostering sections for bus systems are available in [Ibarra-Rojas et al. (2015)]. For rail Crew Scheduling and Rostering, we direct the reader to [Ernst et al. (2001)].

2.1.6 Disruption Management and Real Time Control

All the planning stages mentioned until now assume the system under specific "normal" conditions. However, when operating public transport systems in the real world, disruptions such as equipment breakdown, adverse weather conditions, or infrastructure malfunctions may happen. Disruption Management refers to techniques to restore the public transport system to its normal conditions. This field of research is also very present in other areas of public transport, mainly in the airline industry (see e.g. the review in [Clausen et al. (2010)]). Real Time Control strategies consist of monitoring and acting to maintain the system under its normal conditions. Real Time Control strategies for bus transport can be divided in station control strategies and inter station control strategies [Ibarra-Rojas et al. (2015)]. Station control strategies include holding (to increase regularity or to reduce transfer times), skipping stops, or imposing boarding limits. Inter station strategies include speed regulation strategies and traffic signal priority. In the train domain, [Corman et al. (2017)] recently integrated train scheduling and delay management in real-time railway traffic control. The authors propose passenger-centric models, solution algorithms, and lower bounds for the integrated problem.

Disruption Management for passenger rail transportation can be divided in three main subproblems: timetable adjustment, rolling stock rescheduling, and crew rescheduling [Jespersen-Groth et al. (2009)]. Timetable rescheduling consists of constructing a feasible timetable that recovers the system from the disruption, by adjusting the exist-
ing timetable. Cacchiani et al. (2014) and Corman and Meng (2014) present overviews of models and algorithms for timetable rescheduling under disruptions. Rolling stock rescheduling takes place if the original rolling stock schedule is not suited for handling the disruption scenario. When trains are canceled the rolling stock units will not be positioned at the same locations as originally planned, so the rolling stock schedule has to be re-planned. Rolling stock rescheduling is studied for example in Budai et al. (2010), Nielsen et al. (2012), and Lusby et al. (2017). Crew rescheduling assigns the new duties defined in timetable and rolling stock rescheduling to crew members and is addressed in for example Rezanova and Ryan (2010) and Potthoff et al. (2010).

2.2 Integration of public transport planning problems

Solving the public transport planning subproblems independently leads to suboptimal solutions (Schöbel, 2017). This motivates the growth in research aimed at problems that integrate two or more planning stages simultaneously. In this section we focus on integrated approaches for public transport planning, paying special attention to timetabling integration with other planning problems, as this is the main focus of this thesis.

Previous research integrating line planning and timetabling is mainly focused on heuristic approaches. A first attempt at including line planning elements in the PESP is presented in Liebchen and Möhring (2007), with the authors proposing a matching method for predefined line segments. Kaspi and Raviv (2012) propose a genetic algorithm to solve line planning and timetabling, minimizing operating costs and user inconvenience. Average travel time is reduced by 20%, while using the same amount of resources. Schöbel (2015) presents a mixed integer linear program that minimizes the planned travel time of passengers. In Burggraeve et al. (2017), the authors propose an iterative heuristic to integrate robust timetabling with line planning. The authors report that the robustness of the timetables is significantly improved.

Jiang et al. (2017) consider a train timetabling problem with elements of line planning, for highly congested lines, where an initial timetable is allowed extra services and extra stops or skipping stops. Skipping stops is also considered in Parbo et al. (2018) in a bi-level problem that modifies timetables to optimize passenger service. Blanco et al. (2019) recently proposed a mixed integer nonlinear programming model to integrate line planning and timetabling in automated metro networks. The authors propose a matheuristic to solve the problem and present extensive computational experiments to demonstrate its applicability and effectiveness in real world problems.

Timetabling and vehicle scheduling are integrated for the first time in Ceder (2001), for the single depot case, in an approach that, while still being sequential, includes a feedback loop. The solutions obtained are reported to be good for both operators and passengers.
2.2 Integration of public transport planning problems

For the multi depot case, Liu and Shen (2007) integrate timetabling and vehicle scheduling using bi-level programming: the upper level minimizes number of vehicles and deadhead costs, and the lower level minimizes excess transfer times. Van den Heuvel et al. (2008) use a tabu search algorithm to modify an initial timetable in order to find better vehicle schedules. The authors report operating cost reductions of up to about 8% when compared to the initial timetable. A tabu search approach is also proposed in Guihaire and Hao (2010). Ibarra-Rojas and Rios-Solis (2011) present an aperiodic timetabling and vehicle scheduling model for the single depot case, which minimizes the number of vehicles and maximizes timetable synchronization. Petersen et al. (2013) consider timetable modifications in the form of shifts and propose a large neighbourhood search metaheuristic, minimizing a weighted sum of passenger and operational costs. With the same number of vehicles used, the authors report a 20% decrease in excess transfer time, accompanied by a minor increase in operating costs. Yue et al. (2017) present a bi-level model where the upper level finds an aperiodic timetable and the lower level finds vehicle schedules to cover that timetable. The authors propose a simulated annealing heuristic to solve the problem.

Some authors considered integrating timetabling both with line planning and with vehicle scheduling. Michaelis and Schöbel (2009) propose a re-ordering of the sequential approach, first designing the vehicle routes, then splitting them into lines, and finally calculating a periodic timetable. Costs are included at all planning stages and the objective function is customer-oriented. The authors show results where the number of vehicles is reduced by 10% and the attractiveness of the timetables is increased by 1%. A framework for heuristics and models integrating line planning, timetabling, and vehicle scheduling – the eigenmodel – is presented in Schöbel (2017), where the idea of re-ordering the usual sequence for public transport planning takes a step further. The eigenmodel is a bi-objective model which iterates between the three planning stages, defining different paths for the planning process depending on the order the problems are solved. Pätzold et al. (2017) propose three look-ahead methods: a) using new costs in the line planning step, b) line pool generations with look-ahead, and c) solving vehicle scheduling before and after the timetabling module. The authors report a 40% reduction in vehicle costs when compared to the traditional approach. Li et al. (2018) add line planning and vehicle scheduling aspects to an aperiodic timetabling formulation for single track metro systems. The authors propose a branch-and-bound method and a rolling horizon heuristic to solve the problem. Recently, Lübbecke et al. (2019) present an integer linear programming formulation for integrating the three problems. The authors analyze the structure of the problem and study different decomposition approaches.

The integration of passenger routing in public transport planning problems is also an interesting topic of research in the operations research literature, and saw an increased number of research output in recent years. For an overview on the integration of routing decisions in public transport planning problems we direct the reader to Schmidt (2014). Specifically for timetabling, we direct the reader to Parbo et al. (2014) who review how passenger perspectives are included in timetabling models. Timetabling highly influences how passengers move in the network, and demand changes for each different timetable.
Hartleb et al. (2019) study what makes a timetable good from a passenger perspective, and compare different evaluation functions. With the objective of minimizing total travel time, Schmidt and Schöbel (2015) integrate passenger routing in a timetabling problem and analyze the computational complexity of the problem. A SAT model (boolean satisfiability problem) to integrate periodic timetabling and passenger routing is proposed in Gattermann et al. (2016). Borndörfer et al. (2017) consider different objective functions and allow passengers to follow different paths in their integrated models. The authors consider periodic timetables and report travel and transfer time reductions when compared to a base timetable. Using a bi-level model, Zhu et al. (2017) integrate passenger routing with timetabling: in the upper level, headways that minimize total passenger costs are determined, and then, in the lower level, passenger arrival times are determined. Robenek et al. (2018) address train timetabling design considering a probabilistic demand forecasting model. The authors use simulated annealing to solve the problem and report increases of 15% in revenues. Timetable coordination with passenger considerations is considered in Wu et al. (2019), who propose a bi-level model to perform timetable modifications that re-route passengers with missed transfers. Integration of timetabling, vehicle scheduling, and passenger routing is studied in Laporte et al. (2017), Liu and Ceder (2017), and Schiewe (2018).

Passenger route choice integration with transport optimization problems is also present in other fields. For example, in the airline booking process Dumas and Soumis (2008) combine a fleet assignment optimization model with a passenger flow simulation model. For disruption management, Cadarso et al. (2013) integrate an optimization model for timetabling and rolling stock scheduling with a model for passengers’ behavior, and Kroon et al. (2014) integrate free passenger flows with real-time rolling stock rescheduling. Other authors that integrate free passenger route choice in disruption management are Binder et al. (2017), Veelenturf et al. (2017), Wagenaar et al. (2017), Ortega et al. (2018), and Van der Hurk et al. (2018).

2.3 Final remarks

Although optimization of public transport planning has received a lot of attention during the past decades, with recent years seeing an increase in the number of publications that integrate two or more of the planning sub-problems, a lot remains to be done. Inclusion of detailed passenger information remains a challenge, especially for large case studies. The increased availability of data also motivates new approaches to how problems are tackled. With the advent of digitalization, public transport operators and planners are becoming more receptive to models and algorithms that aid them in planning operations, so there is a high potential for application of new technologies.
Part II

Research Work
Chapter 3

A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

with Evelien van der Hurka, Roberto Robertib, Allan Larsena

aDTU Management - Department of Technology, Management and Economics, Technical University of Denmark, Kgs. Lyngby, Denmark
bDepartment of Supply Chain Analytics, School of Business and Economics, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands


Abstract: Long transfer times often add unnecessary inconvenience to journeys in public transport systems. Synchronizing relevant arrival and departure times through small timetable modifications could reduce excess transfer times, but may also directly affect the operational costs, as the timetable defines the set of feasible vehicle schedules. Therefore better results in terms of passenger service, operational costs, or both, could be obtained by solving these problems simultaneously.

This paper addresses the tactical level of the Integrated Timetabling and Vehicle Scheduling Problem as a bi-objective mixed integer programming problem that minimizes transfer costs and operational costs. Given an initial non-cyclical timetable, and time-dependent service times and passenger demand, the weighted sum of transfer time cost and operational costs is minimized by allowing modifications to the
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

timetable that respect a set of headway constraints. Timetable modifications consist of shifts in departure time and addition of dwell time at intermediate stops with transfer opportunities.

A matheuristic is proposed that iteratively solves the mathematical formulation of the Integrated Timetabling and Vehicle Scheduling Problem allowing timetable modifications for a subset of timetabled trips only, while solving the full vehicle scheduling problem. We compare different selection strategies for defining the subproblems. Results for a realistic case study of the Greater Copenhagen area indicate that the matheuristic is able to find better feasible solutions faster than a commercial solver and that allowing the addition of dwell time creates a larger potential for reducing transfer costs.

Keywords: Public Transport · Bus Timetabling · Vehicle Scheduling · Mixed Integer Linear Programming · Matheuristic

3.1 Introduction

Transfers add substantial amounts of travel time to journeys in large public transport systems. Reduced transfer times resulting from a better synchronization of trips in timetables could increase ridership of public transport and thereby potentially diminish congestion (Ibarra-Rojas and Rios-Solis 2012). A higher mode share for public transportation furthermore aids to reduce rising pollution levels. Therefore, the integration of timetabling and vehicle scheduling is important because it improves passenger service at limited operating costs (Guihaire and Hao 2010). This paper addresses this issue in the context of tactical timetable and vehicle schedule design.

Timetables and vehicle schedules are closely related problems, but are traditionally solved sequentially (Desaulniers and Hickman 2007). Indeed, a small change in the timetable could render an initial vehicle schedule infeasible, or could create options for less costly vehicle schedules. Sequentially optimizing these plans can therefore result in suboptimal solutions. That there is a benefit in integrating timetabling and vehicle scheduling was already demonstrated by e.g. Ceder (2001), Van den Heuvel et al. (2008), Petersen et al. (2013), Ibarra-Rojas et al. (2014) and Laporte et al. (2017), who report savings of up to 20% in transfer waiting times while keeping operational costs at a similar level.

This paper addresses the tactical level of the Integrated Timetabling and Vehicle Scheduling Problem (IT-VSP) as a bi-objective mixed integer programming problem that minimizes transfer costs and operational costs. Given an initial non-cyclical timetable, and time-dependent service times and fixed passenger demand per transfer, the weighted sum of transfer time cost and operational costs is minimized by allowing modifications to the timetable that respect a set of headway constraints. Timetable modifications consist of shifts in departure time and addition of dwell time at intermediate stops with transfer
opportunities. Novelty of the current work lies in the far wider set of allowed timetable modifications in the IT-VSP, the detailed representation of vehicle schedules, and the new matheuristic that for the first time allows to compare results to a lower bound on the problem.

The contributions of this paper are threefold: (i) we present a mathematical formulation for the IT-VSP that allows for a far wider set of timetable modifications by allowing both a change in departure time as well as increases in dwell time under headway constraints; (ii) we propose a matheuristic approach that generates good quality solutions for real size instances faster than a general purpose commercial solver; and (iii) we apply our methodology to a real case study for the express bus network in the Greater Copenhagen area. Results of the case study indicate that the integrated planning of timetables and vehicle schedules can reduce both excess transfer times for passengers and the operational costs. The solutions of the matheuristic in one hour of computation time are substantially better than the solutions of a general purpose solver after seven days of computation time.

![Flow diagram of the matheuristic approach](figure3.png)

The matheuristic, depicted in Figure 3.1, selects and solves a sub-problem of the IT-VSP in each iteration. The input consists of the set of timetabled trips, an initial timetable, a fixed passenger demand per transfer opportunity, and a selection strategy for defining the subproblem. We propose and compare four selection strategies. The matheuristic consists of the two blocks in Figure 3.1 executed iteratively. First a subset of timetable trips is selected according to the selection strategy. Next the IT-VSP mixed integer programming formulation is solved, rescheduling timetables for selected trips only while simultaneously optimizing the vehicle schedules. Specific features of the model are the dynamic assignment of transferring passengers to transfer-to trips, the allowance of non-cyclic timetables, pre-defined changes in on- and off-peak travel times of vehicles, and the creation of detailed vehicle schedules for the planning horizon (e.g. 24 hours). The output defines the new, best known timetable, as well as vehicle schedules that cover that timetable. The iterations stop when either a maximum time or a maximum number of iterations have been reached.

The remainder of this paper consists of a problem description (Section 3.2), a literature review (Section 3.3), a formal problem definition including the MIP formulation (Section 3.4), the matheuristic (Section 3.5), and a case study and discussion of results (Sections 3.6 and 3.7), as well as conclusions and suggestions for future research (Section 3.8).
3.2 The integration of timetabling and vehicle scheduling

Given a set of bus lines and desired frequencies, the Transit Network Timetabling (TNT) problem defines the departure and arrival times for each stop visited by each trip. The general TNT problem aims at maximizing passenger service, and may consider schedule synchronization and transfer times. The Multiple Depot Vehicle Scheduling Problem (MDVSP) has the goal of operating a set of timetabled trips using vehicles from a set of depots at minimum cost. When only one depot is available, the problem is referred to as the Single Depot Vehicle Scheduling Problem (SDVSP), which is solvable in polynomial time. Inputs to the problem are the number of vehicles available at each depot, a set of timetabled trips to be serviced (which is the output of the TNT), and a distance matrix between all terminal stops and depots. Vehicles must start and end at the same depot.

The IT-VSP applies modifications to a provided timetable to minimize a weighted sum of passenger costs and operational costs resulting from the vehicle schedules. Passenger demand is known and fixed, and defined as a number of passengers that wish to transfer from a specific trip to another line at a specific stop. As an example, Figure 3.2 depicts a transfer opportunity between two lines, 300S and 400S, at a Lyngby station. Passengers are assumed to transfer from an arriving trip of a feeder line (e.g., 300S) to the first available trip of their desired line (e.g., 400S). Operational costs are defined as the costs for vehicles waiting at stops (when stretching), vehicles performing non-service trips (dead heading), and fixed costs per vehicle schedule.

A transfer opportunity exists at the crossing of two bus or train lines, such as the crossing of bus lines 300S and 400S at Lyngby station in Figure 3.2. The transfer depicted in Figure 3.3 requires a minimum time of 4 minutes to disembark the 300S vehicle, walk to the stop of the 400S line and embark the 400S vehicle at Lyngby Station. In the current timetable, trip \( i \) of the 300S line arrives at 9:30 leaving a 10 minute transfer time to the next departing 400S line trip \( j \) at 9:40, thus resulting in 6 minutes of excess transfer time.

Timetable modifications that postpone trip \( j_2 \)’s departure time from 9:30 to 9:34 would remove all excess transfer time for the 300S trip \( i \) transfer to line 400S. However, such a change could also influence the departure times of other trips on this line. Figure 3.4 depicts the allowed timetable modifications. The scheduled time of each trip is allowed to vary within half the scheduled headway time (interval of time between two consecutive trips in the same line) to the next trip, thus allowing trip \( j_2 \)’s departure time to be within 9:26 and 9:35. A shift changes the departure of a trip from its first stop. A stretch adds dwell time at any of the intermediate stops of a trip, as represented by the grey nodes and resulting arcs in Figure 3.4. The addition of stretches could represent the distribution of buffer time over a trip, in such a way that it enables better transfer opportunities. The increase in arrivals and departures at Lyngby station for line 400S in Figure 3.4 illustrates that timetable modifications are limited to respect a minimum headway \( h_u \) and maximum
3.2 The integration of timetabling and vehicle scheduling

Figure 3.2: Geographic representation of two bus lines, 300S and 400S, and a station (Lyngby St.), where passengers can transfer.

Figure 3.3: Example of passenger transfer from trip i of line 300S to line 400S at Lyngby St. given a minimum transfer time of 4 minutes, passengers can embark into trip j3 only after waiting 6 minutes.

Figure 3.4: Representation of allowed timetable modifications and headway bounds for trip j2 at Lyngby station.

Timetable modifications may influence which groups of trips can be serviced together by one vehicle, as shifts and stretches applied to trips can alter the required arrival time of a vehicle at the start terminal, or change the arrival time of the selected vehicle at the end terminal of the trip. Trips in Figure 3.5 are depicted as a short arc between a start node and an arrival node (i.e., the start and end terminals of the trip). The grey bars represent the possible shifts forward and backward in time of the trip. The dashed arcs in Figure 3.5 represent a feasible vehicle schedule that starting from depot k in O(k) serves trip i − 1, deadheads from i − 1 to serve trip j1, deadheads from j1 to serve trip j3 and finally returns to the depot in D(k). An earlier departure of trip j1 would make the depicted
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

vehicle schedule infeasible, while alternative shifts may allow to serve more trips with the same vehicle.

![Diagram of vehicle scheduling part of the problem](image)

**Figure 3.5:** Representation of the vehicle scheduling part of the problem

The MIP and the matheuristic we propose in Sections 3.4 and 3.5 aim at simultaneously optimizing the timetabling problem and the vehicle scheduling problem by allowing changes in the input timetables in the form of shifts and stretches illustrated in Figures 3.2-3.5.

### 3.3 Literature review

In this section, we review previous work on the domains of the MDVSP, public transport timetabling, and integration of timetabling and vehicle scheduling. We refer to Guihaire and Hao (2008a) for a review on transit network design and scheduling, who classify and describe over 60 approaches dealing with design, frequency setting, timetabling and combinations of these problems. For a review on the topics of planning, operation and control of bus transport systems we refer to Ibarra-Rojas et al. (2015), who present an extensive literature review with two chapters devoted to timetabling and vehicle scheduling problems.

#### 3.3.1 Multiple depot vehicle scheduling

The IT-VSP is an extension of the MDVSP, which is a classical problem within Operations Research and more specifically in the domain of transport optimization problems. In their book chapter on Time Constrained Routing and Scheduling, Desrosiers et al. (1995) present a detailed description of the problem and provide a review on the literature existent at the time. For a vast survey on the MDVSP, we refer the reader to Desaulniers...
3.3 Literature review

and Hickman (2007), where also some extensions to the MDVSP are discussed. Real-life MDVSP instances of up to 7000 trips can be solved to optimality using column generation by Kliewer et al. (2006). The instances in Kliewer et al. (2006) possess a specific structure that contributes to finding optimal solutions in reasonable time for these large instances. Hadjar et al. (2006) is able to solve randomly generated instances developed by Carpaneto et al. (1989) with up to 800 trips only, with an exact branch-and-bound approach. Due to its complexity, the MDVSP is usually solved using heuristics or metaheuristics. Pepin et al. (2009) compare the performance of five different heuristics to solve instances of the problem with 500, 1000, and 1500 trips, both with 4 and 8 depots. Their computational experiments indicate that when enough computational time is available truncated column generation is the best performing approach, with upper bounds on the average with 0.17% and 0.837% from the optimal solution, and using up to one hour of computing time. However, if the goal is to obtain good quality solutions in fast computational times then large neighbourhood search performs best.

3.3.2 Transit network timetabling

The TNT determines arrival and departure times at stops visited, meeting a given frequency, and satisfying demand. The TNT problem is addressed in the literature using different objectives, for example minimization of excess transfer times (transfer waiting times), maximization of synchronization, or multi-objective approaches. In this section, we review previous research work on the TNT, and group the contributions according to similarities in the objectives considered.

Klemt and Stemme (1988) are the first authors to use minimization of excess transfer times as an objective to the TNT, and proposed a quadratic semi-assignment formulation to model the problem. The authors describe a constructive heuristic to schedule trips one by one and considering transfer synchronization. Domschke (1989) propose a branch-and-bound, local search, and simulated annealing algorithms that outperform Klemt and Stemme (1988) in a simplified example for West Berlin’s subway. Bookbinder and Desilets (1992) minimize transfer costs (costs associated with excess transfer times) considering stochastic travel times and constant headways for each line. Their decision variables are defined as the departures of the first trip in each line. They employ a combination of simulation procedure and the mathematical formulation of Klemt and Stemme (1988). Small example networks are used for the experimental results, and results are compared for optimizing a single transfer node and multiple transfer nodes. Daduna and Voß (1995) minimize excess transfer times and modify the quadratic semi-assignment formulation to restrict transfer opportunities to be within a maximum waiting time. The authors use regret heuristics and unidirectional improvement procedures to compute initial solutions, which are then improved using simulated annealing and tabu search procedures. The different approaches are tested in a series of examples of different sizes, ranging from 14 to 27 lines and from 15 to 38 transfer nodes. Schröder and Solchenbach (2006) start from an original timetable and allow shifts in the start times of trips to optimize the quality
of transfers; they also introduce a new way to assess transfer quality dependent on the amount of time available for transferring and the perceived quality by the users. The TNT is modelled as a quadratic semi-assignment problem, and they solve a linearization of the problem in CPLEX for a small real-life case study. Their results indicate improvements between 0.5% and 5% in comparison to the original timetables, depending on which shifts are allowed and how many nodes are synchronized. Wong et al. (2008) also adjust an original timetable to minimize excess transfer times, but formulate the TNT as a MIP and solve it using Lagrangian heuristics. They allow a wide range of timetable modifications in the form of headway variation, dispatch times, station dwell times and train run times. Their heuristic showed improvements in the solutions obtained for a real case study with 4 lines and 16 transfer stops, when comparing with fixed headways and trip times. Shafahi and Khani (2010) minimize the waiting time at transfer stations and allow the addition of dwell time to improve the transfers. The effect of the added dwell time on through passengers is considered. A large case study is solved using a genetic algorithm approach and reductions of 11.5% of transfer waiting times are reported. Wu et al. (2015) present a timetabling model that minimizes the total waiting time costs for three classes of passengers: transferring, boarding, and through passengers. They consider stochastic travel times and allow the addition of slack time to benefit the passenger transfer feasibility. The authors report that the model is especially effective if the ratio of through-passengers to transfer passengers is below a certain threshold.

Ceder et al. (2001), Liu et al. (2007), and Ibarra-Rojas and Rios-Solis (2012) use maximization of synchronization. The three studies address similar problems where timetables are constructed using minimum and maximum values for headways, and allow solutions with headway variations. Ceder et al. (2001) formulate the TNT as a MIP and develop a heuristic algorithm to solve small examples with up to 4 transfer nodes. Synchronization is defined as the number of simultaneous bus arrivals at connection nodes of the network. Their solution method is able to find 240 simultaneous bus arrivals for a real-life example with 3 transfer nodes and 14 bus lines, and with approximately 3 hours of operation. Liu et al. (2007) redefine synchronization as a coefficient relating the number of lines with synchronized arrivals at a transfer stop with the total number of lines visiting that stop. They develop a nested Tabu Search procedure to generate feasible solutions to a small example with 8 lines and 3 transfer stops, but computational times are not reported. Ibarra-Rojas and Rios-Solis (2012) prove that the TNT is NP-hard and create a preprocessing stage that eliminates variables and constraints, improving the tractability of their MIP model. Synchronization is defined as the arrival of two trips of different lines within a certain time window, at a specific stop. The authors propose an Iterated Local Search (ILS) procedure to solve large test instances of the problem, with up to 200 lines and 40 transfer points, and compare it with a branch-and-bound procedure. The ILS obtains a gap of 15.55% from the lower bound in under one minute of computational time, while the branch-and-bound only gets to 22.64% in two hours of computational time.

Kwan and Chang (2008), Hassold and Ceder (2012), Liu and Ceder (2016), and Wu et al. (2016) consider the TNT as a multi-objective problem. Kwan and Chang (2008) start from an original timetable and allow changes in frequency, dwell time, layover time, and
run time to minimize the cost of transfers and costs caused by deviations from the initial timetable. They use an upper bound on the number of vehicles available to limit the changes made to the timetables. The authors implement a genetic algorithm, a multi-objective evolutionary algorithm, and a local search procedure to solve the problem. They report computational experiments on an example with 6 lines and 5 transfer points. Hassold and Ceder (2012) compare their results with the current timetable and minimize empty seat penalties and expected passenger waiting times. The authors formulate the TNT as a network flow bi-objective problem and consider different vehicle types. They use a multi-objective label-correcting algorithm to solve the problem and computational experiments for a real life case show savings of up to 43% in passenger waiting times, associated with acceptable passenger loads on all vehicles. Liu and Ceder (2016) present a two-objective MIP that minimizes the expected total passenger waiting time and variation in vehicle occupancy. The authors consider fluctuating passenger demand and multiple vehicle types. A decomposition method is used to solve an example and a real life network. Comparing with the current timetables, their approach obtains solutions that reduce total passenger waiting times by approximately 70%, while also reducing variation in vehicle occupancy by approximately 60%. Wu et al. (2016) study a multi-objective re-synchronizing problem for bus timetables, characterized by headway-sensitive passenger demand, uneven headways, service regularity, and flexible synchronization. The objectives considered are the maximization of the number of passengers benefited by smooth transfers and the minimization of the deviation from the existing initial timetable. They use a genetic algorithm to solve the problem and report that high-quality non-dominated solutions are obtained within reasonable CPU time.

For cyclical timetabling problems, a variety of authors apply methodologies using the Periodic Event Scheduling Problem (PESP), introduced by Serafini and Ukovich (1989). The idea is to schedule the events for a cycle, which is then repeated throughout the day. PESP-based approaches are not suitable for solving the IT-VSP in our case, since they impose a periodicity constraint which we want to break apart from. We refer the reader to Nachtigall (1998) for a strong formulation of the PESP applied to railway timetabling. Also in the railway timetabling field, Liebchen and Möhring (2007) extend the PESP to include important decisions of network planning, line planning, and vehicle scheduling into the task of periodic timetabling. A recent state-of-the-art review on cyclic railway timetabling can be found in Kümmling et al. (2015) and other PESP applications to solve the cyclic railway timetabling problem can be found in Liebchen and Möhring (2002), Peeters (2003), or Kroon et al. (2007).

### 3.3.3 Integrated timetabling and vehicle scheduling

To our best knowledge, Ceder (2001) is the first to study the integration of timetabling and the SDVSP. The author develops a 4-step sequential approach with a single feedback loop that determines a timetable and vehicle schedules, obtaining good solutions for both the operator and passengers. The approach is tested in an example with 3 hours of operations
and two lines, with a total of 22 trips. Chakroborty et al. (2001) are the first to include in the TNT the decision of an “optimal fleet size”, which is the number of buses available for each line. They propose a genetic algorithm to solve the problem and present results for a 3-line test instance and total scheduling period of 4 hours. Liu and Shen (2007) use bi-level programming to integrate the TNT formulation of Liu et al. (2007) with the MDVSP, where the upper level minimizes the number of vehicles and deadhead costs and the lower level minimizes the excess transfer time of passengers at intermediate stops. They develop a bi-level nesting tabu search algorithm to solve the problem and present results for an example network with four lines and 3 connection stops, with 3 hours of operation, where the algorithm runs in under 3 seconds of computational time.

Van den Heuvel et al. (2008) integrate the TNT and the MDVSP with the objective of minimizing operational costs, and do not include passenger transfer costs in their model. The authors allow shifts in trip starting times. A Tabu Search algorithm is presented, where timetables are first modified and then the MDVSP problem is optimized. The proposed approach is applied in a real case study with up to 49 lines and 1862 trips, and indicates operational cost reductions of up to about 8% when compared to the original timetable. Guillaume and Hao (2008b) include a weighted objective function considering the number of vehicles, number and quality of transfers, and headway evenness. The problem is solved using an ILS procedure, where at each iteration trips are shifted and the VSP problem is solved. The performance of the solution approach is analyzed for an example with 318 trips, and results show that the number of vehicles is reduced by up to 26%, while the number of feasible transfers increases by up to 44%. Guillaume and Hao (2010) propose a Tabu Search approach that adjusts the timetables by shifts in departure and arrival times, after the vehicle and driver scheduling is solved, with the purpose of providing better transfer opportunities. Without changing either the vehicle or the driver schedules, the operational costs remain constant.

Ceder (2011) creates timetables with even headways and balanced vehicle occupancy, considering multiple vehicle types. The problem is formulated as a minimum cost-flow network problem with NP-hard complexity level, and a heuristic is developed for solving it. The author demonstrates the application of the algorithm in an example with 8 trips and three terminals. Petersen et al. (2013) integrate the MDVSP with timetable modifications in the form of a set of shifts, limiting this set to a pre-defined maximum size, and assume fixed passenger demand. They propose a large neighbourhood search metaheuristic with the objective to optimize a weighted sum of passenger service and operational costs. Results for a case study in the Copenhagen area indicate a decrease in excess transfer time of up to 20%, using the same number of vehicles and a small increase in deadhead cost. Ibarra-Rojas et al. (2014) propose a bi-objective problem that solves the SDVSP and the timetable synchronization problem, assuming passenger demand is fixed. They limit shifts within given time windows around the departure time in a base timetable. With the objectives of minimizing fleet size and maximizing the number of passengers benefited by synchronized transfers, an \( \epsilon \)-constraint method is implemented to obtain Pareto optimal solutions. Results on case study instances with up to 50 lines, up to 5 transfer nodes, and 4 hours of operation show that in some instances increasing the
number of vehicles by one could improve considerably the passenger transfers. Recently, [Laporte et al., 2017] integrate timetabling and vehicle scheduling with special attention to route choice. The timetables are designed taking into account operational costs, expressed as number of vehicles per line and restricted to a budget. The authors use an $\epsilon$-constraint solution approach to obtain the exact Pareto front of solutions. [Liu and Ceder, 2017] also extend the integration of timetabling and vehicle scheduling to include passenger assignment. They present a bi-objective, bi-level IP formulation that optimizes fleet size and user travel and waiting times. An initial vehicle scheduling is given as input. Their integrated vehicle scheduling component does not allow vehicles to deadhead from one trip to another, which greatly reduces the complexity of vehicle scheduling, but would lead to unreasonably costly vehicle schedules for dense networks, like the Copenhagen Network, considered in this paper. Timetabling allows for shifts in departure time. They use a deficit function based sequential search to solve small examples with up to 4 unidirectional lines, 4 transfer stops, and one hour of operations.

Our main contribution in comparison to the related papers of [Petersen et al., 2013], [Ibarra-Rojas et al., 2014], and [Laporte et al., 2017], is that we allow a far wider set of timetable modifications in the form of newly defined stretches, and further extend the set of shifts by removing constraints on the set size. Moreover, we are able to quantify the matheuristic solutions for the IT-VSP in relation to the best feasible and best lower bounds derived from solving the new formulation directly using a general purpose solver for 24 hours. Our results for the case study indicate that the wider set of timetable modifications result in less excess transfer time, while maintaining the same level of operational costs.

### 3.4 A mathematical model for the IT-VSP

In this section, we formally define the IT-VSP and formulate it as a mixed integer linear program. In our formulation the assumptions are:

- The number of passengers wishing to transfer at a stop is fixed;
- Passengers transfer to the earliest trip departing after their arrival time plus a minimum time required for transferring;
- The minimum transfer time is the same for all passengers, and may depend on the transfer stop, the feeder line, and the receiving transfer line;
- The travel time between two stops is deterministic and may depend on the time of day;
- Passenger demands are fixed and given, described as transfer opportunities as explained in the subsection “Passenger transfers”. The model assumes that all transfer opportunities passed as input have a feasible transfer in both the original and final timetables;
• The minimum required transfer time at a transfer node is assumed constant and independent of the individual passenger.

3.4.1 Lines and trips

Let $S$ be the set of all stops. We define a direct line $l \in L$ as a sequence of stops visited by a vehicle, with $L$ being the set of all directed lines. Let a timetable be defined by a set $T = \{1, \ldots, n\}$ of all timetabled trips. Each trip $i \in T$ is defined as having an id, a directed line $l_i$, a total minimum travel time $t_i$, and a set of visited stops $S_i \subseteq S$. Notice that, as the travel time is specified for each trip, then two trips $i,j \in T$ in the same line can have different travel times $t_i, t_j$, which can depend, for example, on the time of the day the trip is scheduled. We define $st_i \in S_i$ as the start terminal, $et_i \in S_i$ as the end terminal, and $J_i \subseteq S_i$ as the set of all intermediate stops visited by trip $i \in T$, i.e., $J_i = S_i \setminus \{st_i, et_i\}$. For each directed line $l \in L$, we define $T_l \subseteq T$ as the subset of all trips in the directed line $l$ - notice that $T = \bigcup_{l \in L} T_l$ and $T_l \cap T_{l'} = \emptyset$ for all $l', l'' \in L, l' \neq l''$. Furthermore, we define the set $T^1$ as the set of all trips which are the first in their directed line.

3.4.2 Timetable modifications

We define minimum and maximum headways, $h^-_{is}$ and $h^+_{is}$ respectively, in relation to the timetable’s headways, for each trip $i \in T$ at each stop $s \in J_i \cup \{st_i\}$. The departure time from $st_i \in S_i$ of any trip $i \in T$ can be modified by a shift within an interval $\{d^-_{i, st_i}, d^+_{i, st_i}\}$ defined in relation to its departure time in the original timetable. A dwell time extension is allowed for all intermediate stops of trip $i$. Let $w^-_{is}$ be the dwell time in the original timetable of a trip $i \in T$ at stop $s \in J_i$, and $w^+_{is}$ the maximum allowed dwell time at the same stop. An upper limit $w$ is imposed on the total added dwell time to all stops of any trip of the set $T$. For each trip $i \in T$, all timetable modifications define earliest and latest arrival times $\{a^-_{is}, a^+_{is}\}$ at all stops $s \in J_i \cup \{et_i\}$, and earliest and latest departure times $\{d^-_{is}, d^+_{is}\}$ from all stops $s \in J_i \cup \{st_i\}$.

3.4.3 Passenger transfers

Let $R$ be the set of all transfer opportunities, where a transfer opportunity $r \in R$, defined by a triplet $(i, l, s)$, represents a transfer request from passengers disembarking trip $i \in T$ at stop $s \in J_i \cup \{et_i\}$ with the intent of embarking a trip $j \in T_l$ of line $l \in L$ such that $l \neq l_i$ and $s \in J_j \cup \{st_j\}$. Let $f_r$ be the number of passengers requesting transfer $r \in R$. Passengers are assumed to transfer to the earliest feasible trip $j \in T_l$. Let $e_r$ be the minimum transfer time for transfer $r \in R$. For a transfer $r = (i, l, s) \in R$, transferring to trip $j \in T_l$ is feasible when $e_r$ is not greater than the difference between the departure
time of trip \( j \) from stop \( s \) and the arrival time of trip \( i \) at stop \( s \). All transfer opportunities in \( R \) are feasible in any feasible timetable. That is, for all \( r = (i, l, s) \in R \) there is at least one trip \( j \in T_l \) such that a transfer from trip \( i \) to \( j \) at stop \( s \) is feasible. For example, if trip \( j_3 \) of Figure 3.3 did not exist, transfer opportunity \((i, 400S, LyngbySt.)\) would not be feasible in the original timetable and thus not be part of \( R \).

### 3.4.4 Compatible trips

The minimum and maximum turnaround times are denoted by \( \{q^-, q^+\} \) respectively, and consist of a buffer time that guarantees arriving with enough time to physically turnaround the vehicle. As extending dwell times in the form of stretches redistributes buffer time, the minimum turnaround time can be reduced by the amount of additional dwell time added to a trip. Two trips \( i, j \in T \) are compatible in a given timetable if three conditions hold. First, the distance between \( et_i \) and \( st_j \), \( Dist(et_i, st_j) \), has to be smaller than the maximum deadhead distance \( u \). Second, the earliest arrival time of trip \( i \) to its end terminal \( (a_{i,et}) \) plus the minimum turnaround time \( (q^-) \) plus the driving time between \( et_i \) and \( st_j \), denoted as \( b_{ij} \), has to be smaller or equal to the latest departure time of trip \( j \) from its start terminal \( (d_{j,st}^+) \). And third, the latest arrival time of trip \( i \) to its end terminal \( (a_{i,et}) \) plus the maximum turnaround time \( (q^+) \) plus \( b_{ij} \) has to be greater or equal to the earliest departure time of trip \( j \) from its start terminal \( (d_{j,st}^-) \).

### 3.4.5 Vehicle scheduling

Let \( K \) denote the set of depots, with each depot \( k \in K \) housing \( v_k \) vehicles. Each vehicle used in a feasible solution covers a sequence of compatible trips and must return to the depot from which it departed. Each depot \( k \in K \) is associated with a graph \( G_k = (V_k, A_k) \). The set of nodes \( V_k \) contains a node for each trip \( i \in T \), as well as for depot \( k \in K \) which is denoted \( n + k \), thus \( V_k = T \cup \{n + k\} \). The set of arcs \( A_k \) defines the deadhead trips \( I = \{(i, j)\mid i, j \in T : i \neq j, Dist(et_i, st_j) \leq u, a_{i,et_i} + q^- + b_{ij} \leq d_{j,st_j}^+, a_{i,et_i} + q^+ + b_{ij} \geq d_{j,st_j}^-\} \), the pull-out trips \( \{n + k\} \times T \) and the pull-in trips \( T \times \{n + k\} \). A deadhead trip exists for any set of pairwise compatible trip nodes in \( V_k \). Thus, \( A_k \) is defined as \( A_k = I \cup (\{n + k\} \times T) \cup (T \times \{n + k\}) \).

The movement of vehicles is defined using triplets \( (i, j, k) \) representing a vehicle from depot \( k \in K \) covering the pair of trips \( (i, j) \in A_k \). Let \( Q = Q^D \cup Q^O \cup Q^H \) be the set of all compatible triplets \( (i, j, k) \), where \( Q^D \) is the set of all deadhead triplets \( Q^D = \{(i, j, k) : k \in K, (i, j) \in I\} \), \( Q^O \) is the set of all pull-out triplets \( Q^O = \{(n + k, j, k) : k \in K, j \in T\} \), and \( Q^H \) is the set of all pull-in triplets \( Q^H = \{(i, n + k, k) : i \in T, k \in K\} \). Let us also define \( T(Q) \) as the set of all pairs of trips \( i, j \in T \) for which a triplet involving \( i \) and \( j \) exists, \( T(Q) = \{(i, j)\mid i, j \in T : \exists(i, j, k) \in Q\} \).
3.4.6 Passenger and operating costs

The IT-VSP aims at minimizing a weighted sum of operating costs (defined as vehicle driving costs, fixed costs per schedule, and additional dwell time costs) and passenger costs (defined as excess transfer time costs and travel cost increase due to additional dwell times).

Vehicle driving costs and fixed costs for schedule creation are captured by costs $c_{ijk}$ associated with each triplet $(i, j, k) \in Q$. The cost $c_{ijk}$ of triplet $(i, j, k) \in Q$ is equal to the deadhead time $b_{ij}$ multiplied by a driving cost per time unit. The costs for creating new schedules are included in the pull-out trips: if $(i, j, k) \in Q^O$, so the arc represents a vehicle leaving the depot, $c_{ijk}$ includes a fixed cost for creating a new schedule in addition to the costs for deadheading to the service trip. The costs for creating a new schedule correspond to the fixed cost for using a vehicle. The operating costs associated with additional dwell time are captured by costs $c_{i}^{DWO}$, affected to each minute of additional dwell time in trip $i \in T$.

Passenger costs are defined as the sum of excess transfer time per passenger and the increase in in-vehicle travel time for on-board passenger due to additional dwell times. In the model, the excess transfer time per transfer opportunity is calculated exactly from the timetable adjustments, and is penalized by a factor $c_{i}^{TR}$ per passenger. The addition of dwell time to a trip is penalized by $c_{i}^{DW_P}$ per minute of additional dwell time. This cost reflects the travel cost increase from one minute of additional dwell time multiplied by the number of on-board passengers of trip $i \in T$. Note that since $c_{i}^{DW_O}$ and $c_{i}^{DW_P}$ are constants which depend on trip $t \in T$, they can be joined in a cost $c_{i}^{DW}$.

The relative weights assigned to $c_{ijk}$, $c_{i}^{TR}$, $c_{i}^{DW}$ will influence the solution outcome. In the case study, we have discussed with public transport authority Movia how to select the weights so that they express costs in monetary units. This calibration allows to directly compare the objectives. In addition, a sensitivity analysis will be performed.

3.4.7 Decision variables and mathematical model

The problem is formulated using the following sets of decision variables:

- Binary variables $x_{ijk}$ for all $(i, j, k) \in Q$, which take the value 1 if and only if a vehicle from depot $k$ travels from node $i$ directly to node $j$, and 0 otherwise;
- Non-negative integer variables $\tau_{is}^{d}$ indicating the departure time of trip $i \in T$ from stop $s \in J_i \cup \{st_i\}$ in minutes from midnight;
- Non-negative integer variables $\tau_{is}^{a}$ indicating the arrival time of trip $i \in T$ at stop $s \in J_i \cup \{et_i\}$ in minutes from midnight;
A mathematical model for the IT-VSP

- Non-negative real variables $\gamma_r$ which store the excess transfer time for passengers using transfer opportunity $r \in R$ in minutes;
- Binary variables $\alpha_{ij,s}$ which take the value 1 if and only if passengers of transfer opportunity $r = (i, l, s) \in R$ are embarking trip $j \in T$, and 0 otherwise;
- Non-negative integer variables $\delta_i$ which store the amount of dwell time added to trip $i \in T$ in minutes. These variables are not necessary in our formulation and are only used to simplify the presentation of the model and improve readability.

The $\alpha_{ij,s}$ variables indicating transfer opportunities are created only for each transfer opportunity $r = (i, l, s) \in R$ and for a set $W(r) = \{(i, j, s)| j \in T_i, i \neq j, a_{is}^r + e_r \leq d_{js}^r, a_{is}^r + e_r + 1.5h_l \geq d_{js}^r\}$, where $h_l$ is the largest frequency observed for line $l \in L$ throughout the day. Moreover, let $W$ be the set of all triplets for all transfer opportunities defined as $W = \cup_{r \in R} W(r)$. In this way, the number of $\alpha_{ij,s}$ variables created is reduced, improving tractability of the model without imposing any practical constraints, since at least one transfer to a trip in $l \in L$ will be available given the timetable modifications.

A MIP formulation for the IT-VSP is:

$$\begin{align}
\text{min} & \quad \sum_{(i,j,k) \in Q} c_{ijk}x_{ijk} + \sum_{i \in T} c_i^{DW} \delta_i + c_i^{TR} \sum_{r \in R} f_r \gamma_r \\
\text{s.t.} & \quad \sum_{(i,j,k) \in Q} x_{ijk} = 1 & i \in T \\
& \quad \sum_{(i,j,k) \in Q} x_{ijk} - \sum_{(j,i,k) \in Q} x_{jik} = 0 & k \in K \quad j \in V_k \\
& \quad \sum_{(i,j,k) \in Q} x_{ijk} \leq v_k & k \in K \\
& \quad d_{i,st_i}^d - \gamma_i \leq \tau_{i,st_i} - d_{i,st_i}^d & i \in T \\
& \quad 0 \leq \tau_{is} - \tau_{is} - w_{is}^r \leq w_{is}^r & i \in T \quad s \in J_i \\
& \quad \delta_i \leq w & i \in T \\
& \quad \delta_i = \tau_{i,et_i} - \tau_{i,et_i} - t_i & i \in T \\
& \quad h_{is}^l \leq \tau_{is}^d - \tau_{i-1,s} \leq h_{is}^r & l \in L \quad i \in T_i : i \notin T^1 \quad s \in J_i \cup \{st_i\}
\end{align}$$

$$\begin{align}
& \quad \tau_{i,et_i} + b_{ij} + q^- - \delta_i - M(1 - \sum_{(i,j,k) \in Q} x_{ijk}) \leq \tau_{j, st_j}^d & (i,j) \in T(Q) \\
& \quad \sum_{(i,j,k) \in W(r)} \alpha_{ijk} = 1 & r = (i,l,s) \in R \\
& \quad \tau_{js}^d - \tau_{is}^d - e_r \geq M(\alpha_{ij,s} - 1) & r \in R \quad (i,j,s) \in W(r) \\
& \quad M \sum_{(i,k,s) \in W(r), k \leq j} \alpha_{iks} \geq \tau_{js}^d - \tau_{is}^d - e_r & r \in R \quad (i,j,s) \in W(r)
\end{align}$$
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

\[ \tau_{js}^d - \tau_{is}^a - e_r - M(1 - \alpha_{ijs}) \leq \gamma_r \]
\[ x_{ijk} \in \{0, 1\} \]
\[ r \in R \quad (i, j, s) \in W(r) \] (3.14)
\[ \tau_{is}^d \in \mathbb{Z}_+ \]
\[ i \in T, \quad s \in J_i \cup \{st_i\} \] (3.15)
\[ \tau_{is}^a \in \mathbb{Z}_+ \]
\[ i \in T, \quad s \in J_i \cup \{et_i\} \] (3.16)
\[ \delta_i \in \mathbb{Z}_+ \]
\[ i \in T \] (3.17)
\[ \gamma_r \in \mathbb{R}_+ \]
\[ r \in R \] (3.18)
\[ \alpha_{ijs} \in \{0, 1\} \]
\[ (i, j, s) \in W \] (3.19)

The objective function (3.1) minimizes a weighted sum of operational and passenger costs. The first term accounts for the deadhead, pull out, and pull in costs for the vehicle movements selected by the model. The second term penalizes operational and on-board passenger costs incurred when adding dwell times. The third term addresses the transfer costs. Indeed, the addition of dwell time to trips increases the travel time for on-board passengers, as well as the in-service time of vehicles.

Constraints (3.2) - (3.4) are classical MDVSP constraints. Constraints (3.2) guarantee that each trip \( i \in T \) is included in exactly one vehicle schedule. Constraints (3.3) are flow conservation constraints for the trip and depot nodes, guaranteeing the continuity of the vehicle schedules created. Constraints (3.4) are capacity constraints that limit the number of pull-out trips to the maximum number of vehicles available at each depot \( k \in K \).

The allowed timetable modifications are modelled in constraints (3.5) - (3.9). Constraints (3.5) force the departure time from the first stop of each trip to lie within the bounds defined for its lower and upper shifts. Constraints (3.6) ensure that the dwell time at each stop of a trip is increased by no more than the maximum dwell time allowed, with respect to the original timetable. Constraints (3.7) impose that the total added dwell time to all stops of a trip does not exceed the maximum allowed \( w \). Constraints (3.8) set the values of the \( \delta_i \) variables to the total added dwell time in the corresponding trip. The minimum and maximum headways between each trip \( i \in T \) and its precedent trip in the same directed line at each stop \( s \in J_i \cup \{st_i\} \) are modelled with constraints (3.9).

The vehicle scheduling and the timetable modification parts of the problem are linked in constraints (3.10). These guarantee that if trips \( i \) and \( j \) are operated consecutively by the same vehicle, then the vehicle has time to deadhead from \( et_i \) to \( st_j \) without violating the minimum turnaround time \( q^- \).

Constraints (3.11) guarantee that passengers from all transfer opportunities \( r = (i, l, s) \in R \) are able to transfer, by selecting exactly one transfer to trip \( j \in T_l \). The transfer variables \( \alpha_{ijs} \) are linked with the departure and arrival times of trips through constraints (3.12) and (3.13). Constraints (3.12) prevent variable \( \alpha_{ijs} \) from taking value 1 whenever passengers do not have enough time to transfer from trip \( i \) to trip \( j \) at stop \( s \), where \( (i, l(j), s) \in R \). Constraints (3.13) are lifting constraints which ensure that passengers arriving from trip \( i \) at stop \( s \) transfer to one of the trips \( j \), such that \( (i, l(j), s) \in R \),
if the arrival and departure times allow the transfer to take place. Constraints (3.13) are in fact not needed for the model to produce feasible solutions, but strengthen the performance of the model. The excess transfer times are stored in the $\gamma_r$ variables by constraints (3.14), which determine this value for each transfer opportunity based on the selected transfers. Together, constraints (3.11)-(3.14) ensure that passengers transfer to the first available trip in the desired line. The selected trip depends on the timetable modifications. Finally, the range of the sets of decision variables used in the model is defined in constraints (3.15)-(3.20).

**Numerical example of constraints (3.11)-(3.14):** To illustrate how constraints (3.11)-(3.14) influence the feasibility of the solutions, consider the transfer opportunity $r = (i, l, s) \in R$ depicted in Figure 3.3, where $l = 400S$ and $s = Lyngby St$. Let us assume that trip $i$ can arrive between 9:23 and 9:42 and that trips $j_1, j_2$ and $j_3$ can shift 5 minutes forward or backward, derived from the headway of line 400S. Input to the model is the set $W(r)$, that in this example contains two triplets, $(i, j_2, s)$ and $(i, j_3, s)$. Because timetable modifications do not allow to transfer from trip $i$ to trip $j_1$, the triplet $(i, j_1, s)$ is not part of $W(r)$. Indeed, trip $i$ arrives at $s$ at 9:23 at the earliest, and trip $j_1$ can depart at 9:25 at the latest, so the minimum transfer time of 4 minutes does not allow passengers to embark on trip $j_1$. Constraints (11)-(14) ensure that at least one of the transfer options in $W(r)$ is feasible, and that transfer time is calculated as the minimum transfer time over all feasible options. Specifically, constraint (3.11) becomes

$$\alpha_{i,j_2,s} + \alpha_{i,j_3,s} = 1,$$

modeling that passengers disembarking from trip $i$ have to transfer either to trip $j_2$ or to trip $j_3$, distinguishing two cases: either a) $\alpha_{i,j_2,s} = 1$, or b) $\alpha_{i,j_3,s} = 1$. Let’s consider case a). Constraints (3.11) force $\alpha_{i,j_3,s}$ to be equal to zero, since passengers of a transfer opportunity transfer to only one trip. Constraints (3.12) become $\tau_{j_2,s} - \tau_{a,s} - e_r \geq 0$ and $\tau_{j_2,s} - \tau_{a,s} - e_r \geq -M$ respectively for $j_2$ and $j_3$, forcing the transfer from $i$ to $j_2$ or to trip $j_3$, which force $\gamma_r$ to equal the waiting time for the transfer $i$-$j_2$. Case b) is similar to case a), with the inverse order of which constraints must hold.

### 3.5 A matheuristic approach

The matheuristic approach, which is denoted as $\text{MHeu}$, is based on the MIP formulation of Section 3.4. Real-life instances of the IT-VSP are intractable when solving the MIP directly with a general solver, so a heuristic approach is needed.

#### 3.5.1 Outline

The $\text{MHeu}$ is an iterative algorithm where at each iteration timetable modifications are allowed for a subset of timetabled trips $T' \subseteq T$ only. An iteration consists of solving the
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

thus restricted IT-VSP problem, which we denote as IT-VSP\((T')\), where variables \(\tau_{is}^d\) and \(\tau_{is}^a\) for all trips \(i \not\in T'\) are fixed to their values in the current best solution. Although timetable changes are only allowed for \(T'\), the vehicle scheduling part of the IT-VSP is solved for all trips. Different selection strategies for constructing \(T'\) are compared and will be explained in Section 3.5.2.

Algorithm 1 outlines the MHeu in pseudo code. The input consists of a set of original timetabled trips \(T\), an initial timetable \(T_0\), the set of transfer opportunities \(R\), a stopCriterion, a selection strategy \(\text{Strat}\), and a set of \(\text{Strat}\) dependent parameters \(\vartheta(\text{Strat})\). The stopCriterion is either the number of iterations or total running time, depending on \(\text{Strat}\). The algorithm starts by solving the MDVSP in Line 1 without allowing timetable modifications, thus \(\tau_{is}^d\), \(\tau_{is}^a\) fixed to \(T_0\) for all \(i \in T\). This generates an initial solution \(S^*\) composed by vehicle schedules \(X_0\) and the initial timetable \(T_0\). The iterative procedure is described in Lines 3 - 7, which runs until the stopCriterion is met.

### Algorithm 1: MHeu

**Input:** \(T\), \(T_0\), \(R\), stopCriterion, \(\text{Strat}\), \(\vartheta(\text{Strat})\)

**Initialization:**

1: \(S^* = (X_0, T_0) \leftarrow \text{solve IT-VSP}(1)-(20)\) with \(\tau_{is}^d, \tau_{is}^a\) fixed to \(T_0\) for all \(i \in T\)
2: \(\eta = 0\)

**Matheuristic:**

3: while stopCriterion not reached do
4: \(T' \leftarrow \text{selectTrips}(S^*, \text{Strat}, \vartheta(\text{Strat}))\)
5: \(S^* = (X_\eta, T_\eta) \leftarrow \text{solve IT-VSP}(1)-(20)\) with \(\tau_{is}^d, \tau_{is}^a\) fixed to \(T_{\eta-1}\) for all \(i \in T \setminus T'\)
6: \(\eta = \eta + 1\)
7: end while
8: return \(S^*\)

Each iteration starts in Line 4 by selecting the subset of trips \(T' \subset T\) according to \(\text{Strat}\) (one of the trip selection strategies described in Section 3.5.2), and using the set of parameters \(\vartheta(\text{Strat})\). These parameters define how many and which trips are selected, and the maximum running time for each iteration. Timetable modifications in arrival and departure times are allowed for trips in \(T'\) only. A new solution is calculated in Line 5 by solving the restricted IT-VSP\((T')\), with \(\tau_{is}^d, \tau_{is}^a\) fixed to \(T_{\eta-1}\) for all \(i \in T \setminus T'\). The solution obtained is always at least as good as the current best solution, since the current best solution is always feasible. To ensure that a solution is always found, each iteration starts from the current best solution, using CPLEX warm-start. The best solution \(S^*\) found is returned once the stopCriterion is met.

### 3.5.2 Trip selection strategies

A run of the MHeu uses one and only one of the four trip selection strategies defined in this section. A trip selection strategy consists of a parameter specifying the size of the
3.5 A matheuristic approach

selected trip set, and a rule for selecting trips from $S^*$. Each selection strategy also uses a parameter $\psi$ that defines the maximum running time of each iteration. We propose the following selection strategies:

- **Random** ($\text{Rand}$, $\vartheta(\text{Rand}) = \{\psi, \kappa\}$): selects $\kappa$ trips of $S^*$, where any trip $t \in S^*$ has an equal probability of being selected for $T'$ at each iteration. The stopping criterion for the $\text{Rand}$ strategy is total running time.

- **Rolling Horizon** ($\text{RolH}$, $\vartheta(\text{RolH}) = \{\psi, \Omega, \xi\}$): deterministic procedure that defines a set $\Omega$ of equally long time intervals. Each time interval $\omega \in \Omega$ is defined by a time window $[s_\omega, e_\omega]$, where $s_\omega$ is the start time and $e_\omega$ is the end time of the interval. The strategy runs in $|\Omega|$ iterations, each of them referring to one time interval $\omega \in \Omega$. At iteration $\omega$, $T'$ is composed by all trips in $S^*$ with start time belonging to $[s_\omega, e_\omega]$. Consecutive intervals overlap each other by a percentage defined by a parameter $\xi$. The stopping criterion is the number of iterations, and these are indirectly constrained by a maximum total running time.

Consider a small example with trips starting between 6:00 and 22:00. Suppose we want to solve the $\text{MHeu}$ with the $\text{RolH}$ with $|\Omega| = 5$ and $\xi = 25\%$. We define the 5 equally long intervals: $[6:00, 10:00]$, $[9:00, 13:00]$, $[12:00, 16:00]$, $[15:00, 19:00]$, $[18:00, 22:00]$, which overlap each other by 60 minutes (25% of the size of the interval). If the total computational time allowed is equal to one hour, then $\psi = \frac{60}{5} = 12$ minutes.

- **Cost Probability** ($\text{CostP}$, $\vartheta(\text{CostP}) = \{\psi, \kappa\}$): selects $\kappa$ trips of $S^*$, where the probability of selecting a trip $t \in S^*$ is calculated as

$$p(t) = \frac{\text{TrC}(S^*, t)}{\sum_{i \in T'} \text{TrC}(S^*, i)}$$

where $\text{TrC}(S^*, t)$ are the transfer costs associated with trip $t$ in solution $S^*$, i.e., transfer costs incurred in $S^*$ by transferring to or from $t$. If the number of trips with transfer costs is lower than $\kappa$, all trips with transfer costs are selected and trips without transfer costs are randomly selected until $\kappa$ is reached. The stopping criterion for the $\text{CostP}$ strategy is total running time.

- **Relatedness** ($\text{Relat}$, $\vartheta(\text{Relat}) = \{\psi, \kappa\}$): We define a new subset $\hat{T}_t \subseteq S^*$, which contains all trips $\hat{t}$ related to a trip $t \in T$, in $S^*$. A trip $\hat{t}$ is related to $t$ in $S^*$ when either $t$ and $\hat{t}$ belong to the same vehicle schedule in $S^*$, or when there exist passenger transfers between trips $t$ and $\hat{t}$ in $S^*$. Starting with $T' = \emptyset$, this selection strategy iteratively selects a random trip $t \in S^*$, and adds $t$ and all trips $\hat{t} \in \hat{T}_t$ to $T'$. This process is repeated until $|T'| = \kappa$. The stopping criterion for the $\text{Relat}$ strategy is total running time.

The $\text{Rand}$ strategy provides a base scenario. The $\text{RolH}$ represents a methodology where start time of trips is important, selecting for modifications trips that have a higher chance
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

of sharing transfers. The \texttt{CostP} strategy addresses transfer cost optimization, by selecting trips that in the current solution have high transfer costs. Finally, the \texttt{Relat} strategy is directly linked to the objectives of minimizing excess transfer time and vehicle schedule costs, by selecting trips that are related with each other in the current solution, in terms of vehicle schedules or transfers.

3.6 Case study

In this work, we focus on the 8 bi-directional express-bus lines (S-Bus) in the Greater Copenhagen area, which provide faster routes than regular bus lines, with fewer stops, and complement the local train service (S-Train) lines across and radially. Figure 3.6 is a geographical representation of the S-Bus and S-Train service. The bus network in the Greater Copenhagen area acts as a supplement to the S-Train service, which forms the so-called five finger plan resembling the five fingers of a hand. Movia is the public transport agency responsible for the planning of buses in the region of Zealand, and provided data for the case study. Shifts and stretches are allowed for S-Bus lines’ trips, and the S-Train lines operate according to a fixed timetable. The vehicle scheduling component of the problem includes the bus trips only; the vehicle scheduling of trains is not included.

![Central Station](image)

Figure 3.6: Geographic representation of the S-Bus network. The dashed lines represent the S-Train lines.

The IT-VSP data input components are: (i) an initial timetable for the bus lines, which
defines the set of all trips and also bus line frequencies used to calculate minimum and maximum headways in the new solution; (ii) fixed timetables for train lines; (iii) a distance matrix which includes all distances between trip terminal stops and depots; (iv) the number of transferring passengers using each transfer opportunity, which can be a bus-bus, bus-train, or train-bus transfer; (v) costs and parameters, namely minimum and maximum turnaround times, minimum transfer times at different transfer opportunities, vehicle operational costs, fixed costs per vehicle schedule, value of time costs for passenger excess transfer time, driving speed for vehicles while deadheading, maximum deadhead distance, maximum added dwell time per trip and per stop, and depot capacities.

Input components (i) and (ii) are publicly available. Deadhead distances (iii) were obtained using geographical data. The number of passengers using each transfer opportunity (iv) was estimated based on Movia’s data on the number of embarking and disembarking passengers at each stop for each bus trip. The set of transfer opportunities $R$ is input to our model, and is the same from iteration to iteration in the matheuristic. It was defined and provided by Movia, and the computation of this set is therefore not part of our model. The costs and parameters (v) were estimated in collaboration with Movia. They provided estimates of operational waiting time, distance, and schedule costs expressed in monetary units, which together defines the operational costs. Excess waiting time is weighted by a value of time factor, this weighted sum defining transfer costs. The objective minimizes the sum of operational costs and transfer costs. Due to lack of data on on-board passengers, our case study assumes a unique $c_t^{DW}$ for all trips.

We derive 3 instances of different sizes from the case study, with respectively 3, 5, and 8 S-Bus lines, which are described in Table 3.1. The 3-line instance consists of the most central circular bus lines (200S, 300S, and 400S), the 5-line instance adds 2 more rural circular bus lines to the 3-line instance (500S, and 600S), and the 8-line instance adds 3 additional radial bus lines to the 5-line instance (150S, 250S, and 350S). The first column in Table 3.1 is the instance index, and the second column is the number of undirected bus lines considered. The three remaining columns are the number of transfers for respectively bus-bus, bus-train, and train-bus transfer opportunities.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Bus Lines</th>
<th>Trips</th>
<th>Bus-Bus Transfers</th>
<th>Bus-Train Transfers</th>
<th>Train-Bus Transfers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>556</td>
<td>48</td>
<td>375</td>
<td>853</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>864</td>
<td>128</td>
<td>554</td>
<td>1150</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>1585</td>
<td>360</td>
<td>1109</td>
<td>1644</td>
</tr>
</tbody>
</table>

It can be observed that the instances increase in both number of trips and number of transfer opportunities. All three instances include 7 train lines with a total of 1308 trips with fixed timetables.

To create vehicle schedules that cover the initial timetables, we solve the MDVSP without allowing timetable modifications. In the 8 line instance, the vehicle schedules consist of 1585 bus trips assigned to 205 vehicles. Each schedule starts and ends in the same depot.
and it is allowed to service trips from different lines in the same schedule, thus allowing
deadheading between consecutive trips in a schedule.

The case study includes time dependent service bus travel times and constant deadhead
speeds along the day. Vehicles can service trips from different bus lines in the same sched-
ule, known at Movia as *interlining*. The maximum deadhead distance is 15 kilometres
(i.e., \( u = 15 \)), the minimum turnaround time is 12 minutes (i.e., \( q^- = 12 \)), and the max-
imum turnaround time is 30 minutes (i.e., \( q^+ = 30 \)). The dwell time at each stop with
transfers can be increased by up to 3 minutes (i.e., \( w_{is} = 3 \), \( i \in T \), \( s \in J_i \)), and a maximum
of 10 minutes of dwell time can be added in total to a trip (i.e., \( w = 10 \)). The additional
dwell time is deducted from the buffer in the turnaround time at the end of the trip. The
shifts allowed in each trip departure time were created based on the original timetables
for each bus line. Considering consecutively timetabled trips \( (i - 1), i, (i + 1) \in T \) and
with departure time from the first stop \( d_{i-1,st_i}, d_{i,st_i}, d_{i+1,st_i} \) respectively, the lower and
upper shift limits for trip \( i \) are calculated with the expressions

\[
\begin{align*}
  d^-_{i,st_i} & = d_{i,st_i} - \left\lfloor \frac{d_{i,st_i} - d_{i-1,st_i} - 1}{2} \right\rfloor \\
  d^+_{i,st_i} & = d_{i,st_i} + \left\lfloor \frac{d_{i+1,st_i} - d_{i,st_i}}{2} \right\rfloor
\end{align*}
\]

ensuring that trips can never overtake each other in the timetable. At each stop, trip
modifications are also bounded by the minimum and maximum headways. For each trip
\( i \in T \) at each stop \( s \in J_i \cup \{st_i\} \), minimum and maximum headways, \( h^-_{is} \) and \( h^+_{is} \),
are calculated based on the scheduled headway in the original timetable. Table 3.2 shows the
allowed variations on headways based on the scheduled headways.

<table>
<thead>
<tr>
<th>Scheduled headway (minutes)</th>
<th>Minimum and maximum headway variation (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \leq 4 )</td>
<td>+/- 1</td>
</tr>
<tr>
<td>( \leq 12 )</td>
<td>+/- 2</td>
</tr>
<tr>
<td>( \leq 20 )</td>
<td>+/- 3</td>
</tr>
<tr>
<td>( \leq 39 )</td>
<td>+/- 4</td>
</tr>
<tr>
<td>( \geq 40 )</td>
<td>+/- 5</td>
</tr>
</tbody>
</table>

### 3.7 Computational experiments

This section evaluates the performance of the MHeu in a set of computational experiments.
It consists of a set of parameter tuning experiments for the selection strategies (Section
3.7.1) an analysis of computational performance (Section 3.7.2), and an analysis of solution quality in terms of transfer cost, operational costs and total cost (Section 3.7.3). The parameter tuning is based on the 3-line instance allowing the addition of stretches for a total running time of 30 minutes. Computational performance and solution quality are evaluated and compared for solving the MHeu with and without allowing stretches for all selection strategies, for all 3 instances (3, 5 and 8 lines) and for a total running time of 1, 5 and 12 hours, inspired by the running time limits used in Petersen et al. (2013) and justified by the fact that the problem is addressed at the tactical level. Furthermore, convergence, trade-off between operational and transfer costs, the quality of resulting vehicle schedules, and the distribution of excess transfer time are discussed.

The algorithm was implemented in C and the mathematical formulations were solved using CPLEX version 12.6. The experiments were conducted on HPC servers, using Intel Xeon E5-2660 v3 2.60GHz processors, and 1 computation core. Each iteration used CPLEX warm-start to start from the best solution found so far. All results presented are average results over five runs for each different setting, except for the RolH strategy as it is deterministic.

The following measures are used to express computational performance and solution quality. The computational performance of the MHeu algorithm is expressed in relation to solving the IT-VSP(T) directly in CPLEX with a maximum computation time of 24 hours. Performance is expressed as \( \text{Gap} \): the average percentage gap to the best lower bound obtained in 24 hours; as well as \( \text{Gap}^* \): the average percentage gap to the best known upper bound obtained in 24 hours, e.g. the best obtained integer solution. The formulas to calculate \( \text{Gap} \) and \( \text{Gap}^* \) are:

\[
\text{Gap} = \frac{S_{\text{MHeu}}^{AVG} - S_{\text{IT-VSP}}^{LB}}{S_{\text{IT-VSP}}^{LB}}, \quad \text{Gap}^* = \frac{S_{\text{MHeu}}^{AVG} - S_{\text{IT-VSP}}^{UB}}{S_{\text{IT-VSP}}^{UB}},
\]

where \( S_{\text{MHeu}}^{AVG} \) is the average objective value over all five runs of an instance and setting of MHeu, \( S_{\text{IT-VSP}}^{LB} \) is the best lower bound obtained with CPLEX solving IT-VSP(T) in 24 hours, and \( S_{\text{IT-VSP}}^{UB} \) is the objective value of the best integer solution obtained with CPLEX solving IT-VSP(T) in 24 hours. The values \( S_{\text{IT-VSP}}^{LB} \) and \( S_{\text{IT-VSP}}^{UB} \) are computed for each instance and each setting of timetable modifications (with or without stretches) separately. Even when extending the computation time from 24 hours to 7 days we were not able to identify the optimal solution, and the decrease in \( \text{Gap} \) is only between 0.04% to 0.64% over all instances.

The solution quality of \( S_{\text{MHeu}}^{AVG} \) is expressed in terms of transfer cost (\( \text{TrC} \)), operational costs (\( \text{OpC} \)), and total costs (\( \text{TC} \)), which are average percentage differences to the optimal base solution \( S_{\text{NDVSP}} \) of IT-VSP(\( \emptyset \)). The \( S_{\text{NDVSP}} \) represents the best solution without integration of vehicle scheduling and timetabling, on which we aim to improve in terms of transfer time and operational costs. Let \( x = \{ \text{TrC}, \text{OpC}, \text{TC} \} \) and \( f_x(S) \) denote the \( x \)-type cost of a solution \( S \), then these quality measures are computed as:
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

\[ x = \frac{f_x(S_{\text{MHeu}}^{\text{AVG}}) - f_x(S_{\text{MDVSP}})}{f_x(S_{\text{MDVSP}})} \times 100\% \]

A negative percentage for \( x \) corresponds to a reduction of costs in the MHeu solution in comparison to the non-integrated \( S_{\text{MDVSP}} \) base solution.

### 3.7.1 Parameter tuning

Table 3.3 shows the parameter tuning results for the Rand, CostP, and Relat selection strategies. All three strategies require the same set of parameters as input, which consists of the number of trips selected \( \kappa \) for \( T' \) at each iteration, and the maximum running time of each iteration in minutes \( \psi \). We tested \( \kappa = 150, 200, 250, 300, 350, 400, 450 \) trips and \( \psi = 0.5, 1, 2, 5, 10 \) minutes, and report average results over five runs.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>( \psi ) (m)</th>
<th>150</th>
<th>200</th>
<th>250</th>
<th>300</th>
<th>350</th>
<th>400</th>
<th>450</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rand</td>
<td>0.5</td>
<td>-2.15%</td>
<td>-3.21%</td>
<td>-4.36%</td>
<td>-5.71%</td>
<td>-6.51%</td>
<td>-6.32%</td>
<td>5.01%</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-1.87%</td>
<td>-3.74%</td>
<td>-4.57%</td>
<td>-5.80%</td>
<td>-6.02%</td>
<td>-6.35%</td>
<td>-4.23%</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>-1.85%</td>
<td>-2.90%</td>
<td>-4.61%</td>
<td>-5.67%</td>
<td>-6.23%</td>
<td><strong>-6.70%</strong></td>
<td>-3.67%</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>-2.13%</td>
<td>-2.51%</td>
<td>-4.43%</td>
<td>-4.90%</td>
<td>-6.18%</td>
<td>-5.69%</td>
<td>-4.74%</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>-2.42%</td>
<td>-3.05%</td>
<td>-4.36%</td>
<td>-5.09%</td>
<td>-4.82%</td>
<td>-3.41%</td>
<td>-3.37%</td>
</tr>
<tr>
<td>CostP</td>
<td>0.5</td>
<td>-2.36%</td>
<td>-2.98%</td>
<td>-4.74%</td>
<td>-5.25%</td>
<td>-5.21%</td>
<td>-6.01%</td>
<td>5.01%</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-1.94%</td>
<td>-3.35%</td>
<td>-4.73%</td>
<td>-5.44%</td>
<td>-5.99%</td>
<td><strong>-6.16%</strong></td>
<td>-4.98%</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>-2.11%</td>
<td>-2.61%</td>
<td>-4.52%</td>
<td>-5.55%</td>
<td>-5.66%</td>
<td>-6.00%</td>
<td>-3.75%</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>-1.87%</td>
<td>-3.54%</td>
<td>-4.58%</td>
<td>-5.19%</td>
<td>-5.43%</td>
<td>-6.01%</td>
<td>-5.40%</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>-1.96%</td>
<td>-2.66%</td>
<td>-5.01%</td>
<td>-4.93%</td>
<td>-5.69%</td>
<td>-5.13%</td>
<td>-3.89%</td>
</tr>
<tr>
<td>Relat</td>
<td>0.5</td>
<td>-2.35%</td>
<td>-2.80%</td>
<td>-4.51%</td>
<td>-5.47%</td>
<td>-6.06%</td>
<td>-4.58%</td>
<td>5.01%</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-2.42%</td>
<td>-3.28%</td>
<td>-3.93%</td>
<td>-5.17%</td>
<td><strong>-6.45%</strong></td>
<td>-5.39%</td>
<td>1.78%</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>-2.69%</td>
<td>-3.58%</td>
<td>-4.57%</td>
<td>-4.97%</td>
<td>-6.01%</td>
<td>-6.41%</td>
<td>-3.20%</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>-2.51%</td>
<td>-3.05%</td>
<td>-4.04%</td>
<td>-4.85%</td>
<td>-5.21%</td>
<td>-4.94%</td>
<td>-4.08%</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>-2.79%</td>
<td>-3.76%</td>
<td>-4.07%</td>
<td>-4.51%</td>
<td>-4.63%</td>
<td>-3.23%</td>
<td>-3.37%</td>
</tr>
</tbody>
</table>

Table 3.3 contains the average \( \text{Gap}^* \) per selection strategy for all parameters, where minimal average \( \text{Gap}^* \) per selection strategy is marked in bold. The average \( \text{Gap}^* \) per selection strategy is smallest for \( \psi = 2 \), \( \psi = 1 \), and \( \psi = 1 \) for Rand, CostP and Relat respectively, thus indicating that running more iterations with a short computational time may be more beneficial than running less iterations with a long computational time for solving
3.7 Computational experiments

CPLEX in each iteration. Furthermore, the lowest average \( \text{Gap}^* \) is obtained for \( \kappa = 400 \), \( \kappa = 400 \), and \( \kappa = 350 \) for \text{Rand}, \text{CostP} \) and \text{Relat} \) respectively, indicating that a larger sub-problem, with a larger solution space, may provide better solutions. However, for \( \kappa = 450 \), computation times of 0.5 minutes may be too small: all selection strategies find worse solutions for this case, and for the \text{Relat} \) this also holds for \( \psi = 1 \) minute. For all other parameters the MHeu improves on the CPLEX solution, with \( \text{Gap}^* \) between -1.85\% and -6.70\%. However, differences per strategy for improved solutions can be up to 4\%, while the second-best setting is no more than 0.35\% from the best one and in 1 out of 3 cases is obtained with different values for both \( \kappa \) and \( \psi \) than the ones for the lowest gap. Furthermore, \( \text{Gaps} \) sometimes form an oscillating pattern, for example for \text{Rand} \( \kappa = 150 \).

Table 3.4 contains average \( \text{Gap}^* \) for the RolH selection strategy, with the lowest average \( \text{Gap}^* \) marked in bold. The parameters for this selection strategy are the number of time intervals \( |\Omega| \) and the percentage overlap \( \xi \) between consecutive time intervals. Provided the overall computation time limit of 30 minutes and a fixed number of iteration for this approach, the time limit per iteration is set to \( \psi = \frac{30}{|\Omega|} \). The values tested for the number of time intervals are \( |\Omega| = 2, 4, 6, 8, 10, 12, 14, 16 \) intervals, and the percentage overlaps tested are \( \xi = 20, 25, 30, 35, 40 \). Both RolH parameters influence how many trips are selected for \( T' \) at each iteration. For each setting, we report the size of \( |\Omega| \) and the maximum number of trips \( \kappa^{max} \) selected in each time interval of \( \Omega \) between brackets: \( |\Omega|(\kappa^{max}) \). Since this strategy is deterministic, only one run of each setting is conducted.

### Table 3.4: Average \( \text{Gap}^* \) for the RolH selection strategy

<table>
<thead>
<tr>
<th>Strategy</th>
<th>( \xi ) (%)</th>
<th>16 (75)</th>
<th>14 (82)</th>
<th>12 (95)</th>
<th>10 (117)</th>
<th>8 (134)</th>
<th>6 (167)</th>
<th>4 (220)</th>
<th>2 (413)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RolH</td>
<td>20</td>
<td>-2.48%</td>
<td>-3.05%</td>
<td>-2.26%</td>
<td>-2.25%</td>
<td>-1.96%</td>
<td>-2.74%</td>
<td>-1.17%</td>
<td>-1.04%</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>-3.03%</td>
<td>-2.97%</td>
<td>-4.60%</td>
<td>-3.49%</td>
<td>-4.55%</td>
<td>-3.58%</td>
<td>-1.40%</td>
<td>-1.13%</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>-3.07%</td>
<td>-3.99%</td>
<td>-4.27%</td>
<td>-4.33%</td>
<td>-5.34%</td>
<td>-1.43%</td>
<td>-2.42%</td>
<td>2.33%</td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>-4.14%</td>
<td>-3.06%</td>
<td>-2.68%</td>
<td>-3.46%</td>
<td>-5.83%</td>
<td>-1.33%</td>
<td>-3.31%</td>
<td>-1.28%</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>-4.29%</td>
<td>-5.28%</td>
<td>-3.17%</td>
<td>-4.55%</td>
<td>-3.37%</td>
<td>-2.57%</td>
<td>-1.36%</td>
<td>-1.11%</td>
</tr>
</tbody>
</table>

The results presented in Table 3.4 show that contrary to previous selection methods the RolH strategy has the best performance for a relatively small sub-problem of around 134 trips \( (|\Omega|=8) \) rather than for larger sub-problems of around 400 trips \( (|\Omega|=2) \). The \( \text{Gap}^* \) decreases when the number of time intervals increases from 4 to 8. The overall smallest \( \text{Gap}^* \) of -5.83\% is obtained for \( \xi = 35\% \). Further increasing \( |\Omega| \) does not improve the results.

The parameter settings resulting from the parameter tuning experiments are summarized in Table 3.5 per instance. It was necessary to increase the runtime per iteration for the 8-line instance as its increased size and complexity did not allow building the model within the specified time. The number of time intervals for the RolH strategy in the 5 and 8 lines instances were selected to resemble the \( \kappa^{max} \) parameter of the 3-line instance at \( |\Omega| = 8 \) and \( \xi = 30\% \).
Table 3.5: Parameter values used for each of the 3 instances and each of the selection strategies

<table>
<thead>
<tr>
<th>Instance</th>
<th>Parameter</th>
<th>3 Lines</th>
<th>5 Lines</th>
<th>8 Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 Lines</td>
<td>(\kappa)</td>
<td>400</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>5 Lines</td>
<td>(\psi) (m)</td>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>8 Lines</td>
<td>(\varOmega(\kappa_{\text{max}}))</td>
<td>8 (134)</td>
<td>14 (141)</td>
<td>24 (149)</td>
</tr>
<tr>
<td></td>
<td>(\xi) (%)</td>
<td>35</td>
<td>35</td>
<td>35</td>
</tr>
</tbody>
</table>

3.7.2 Computational performance

In this section, we analyze the computational performance of the different selection strategies given different total running time limits for the 3, 5 and 8 lines instances.

3.7.2.1 Selection Strategies and Running Time

Table 3.6 contains the (upper bound) \(\text{Gap}^*\) and (lower bound) \(\text{Gap}\) defined in Section 3.7, which express the performance of the MHeu to the performance of a standard solver (CPLEX) for solving the IT-VSP directly within a time limit of 24 hours.

<table>
<thead>
<tr>
<th>Stretches</th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance</td>
<td>3 Lines</td>
<td>5 Lines</td>
</tr>
<tr>
<td>1</td>
<td>Rand</td>
<td>-5.59</td>
</tr>
<tr>
<td></td>
<td>RolH</td>
<td>-4.89</td>
</tr>
<tr>
<td></td>
<td>CostP</td>
<td>-5.43</td>
</tr>
<tr>
<td></td>
<td>Relat</td>
<td>-5.54</td>
</tr>
<tr>
<td>5</td>
<td>Rand</td>
<td>-5.75</td>
</tr>
<tr>
<td></td>
<td>CostP</td>
<td>-5.84</td>
</tr>
<tr>
<td></td>
<td>Relat</td>
<td>-5.74</td>
</tr>
<tr>
<td>12</td>
<td>Rand</td>
<td>-6.09</td>
</tr>
<tr>
<td></td>
<td>CostP</td>
<td>-6.00</td>
</tr>
<tr>
<td></td>
<td>Relat</td>
<td>-5.76</td>
</tr>
</tbody>
</table>

The MHeu always improves on the CPLEX solution even when running it for only 1 hour, with the only exception being the RolH selection strategy that requires more than 1 hour to find.
a better feasible solution. For 12 hours of total computational time, the \( \text{Gap}^* \) to the CPLEX solution is between -4.61% and -8.52% when allowing stretches and between -2.81% and -6.48% when not allowing stretches. The improvement in \( \text{Gap}^* \) for 12 hours compared to 5 hours computation time is only between 0.13% and 1.5%, and therefore one could also opt for a shorter computation time than 12 hours. However, a computation time of one hour seems insufficient especially for larger instances. This is especially evidenced by the poor performance of the RolH strategy at one hour computation times. Since the integrated timetabling and vehicle scheduling problem is a tactical level problem, computation times of several hours are non-prohibitive for using this algorithm in practice. Furthermore, for the sake of comparison, all experiments in this paper were run using only one computation core. Increasing the computation cores would most likely reduce the computational times.

As the size of the instance grows, the lower bound \( \text{Gap} \) increases from around 2% and 4% for the 3-line instance to around 10% and higher for the 8-line instance. Thus the quality of the lower bounds seems to decrease when the instance size increases. The lower bound \( \text{Gap} \) is somewhat higher when allowing stretches, which is intuitively explained by the wider solution space when allowing stretches.

The gap differences between selection strategies are small for a minimal computation time of five hours: the difference in upper bound \( \text{Gap}^* \) ranges between 0.40% and 1.69% when not allowing stretches, and between 0.65% and 2.97% when allowing stretches. Moreover, there isn’t one strategy that consistently returns better results than the others, and all strategies return the best result in at least one of the different combinations of runtime, instances, and allowing or disallowing stretches for a minimum of five hours computation time. The performance over the five runs is relatively stable with standard deviation below 0.82% for these instances.

### 3.7.2.2 Convergence of the Different Selection Strategies

The convergence of each selection strategy for the 8-line instance, with stretches, and with a total running time of 5 hours, is depicted in Figure 3.7. The x-axis reflects the running time expressed in seconds and the y-axis represents the percentage reduction in total costs. Each curve represents the best run of the algorithm for a specific selection strategy. Markers indicate when the strategy finds an improved feasible solution.

All selection strategies start with a steep decline, which slows down generally after around 100 minutes of computation time. After 3 hours most strategies have converged, the Relat being the only exception in this specific case, however similar small improvements have been observed for Rand after this amount of computation time. Only minor improvements are obtained extending the computation time further than 5 hours, as discussed in Section 3.7.2.1.

The RolH finds the solution with the lowest total costs for this instance, and does so within half of the total running time limit. The RolH runs shorter when at any of the
iterations, the optimal solution for the IT-VSP \( (T') \) is found within the running time limit of the iteration.

### 3.7.3 Transfers and operational costs

This section expresses the performance of the MHeu in comparison to the base MDVSP solution to discuss the benefits in terms of passenger service and operational cost in comparison to current practice. Thus, it focuses on the value that the algorithm may provide public transport agencies, with a specific focus on the allowance of extra dwell time in timetabled trips (stretches).

#### 3.7.3.1 Value of Stretches

Table 3.7 contains the transfer cost \( TrC \), operational cost \( OpC \), and total costs \( TC \) for the 3, 5 and 8 lines instances, with and without stretches, for all four selection strategies and a total running time of 5 hours. The values in bold represent which selection strategy performed best in each category (\( TrC \), \( OpC \), and \( TC \)) per instance, for allowing and disallowing stretches independently.

The integration of timetabling and vehicle scheduling can reduce the total costs between 10.19% and 11.76% for the 3-line instance, between 9.58% and 10.99% for the 5-line instance, and between 6.91% and 8.39% for the 8-line instance. The inclusion of stretches in the timetable modifications achieves solutions with total costs comparable to the solutions
Table 3.7: MHeu results with a run time of 5 hours, with and without stretches, expressed in TrC, OpC, and TC.
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling

with only shifts. In a few cases, the addition of stretches shows a small improvement in the total costs, ranging from approximately 0.08% to 0.75% points.

The results obtained in terms of transfer costs are comparable for all selection strategies, over all 6 cases (3, 5 and 8 lines instances, with and without stretches). The Relat strategy may be especially suitable to reduce transfer costs: it provides lowest transfer costs in 3 out of 6 cases, and a close second-least transfer costs in two more instances, however differences are too small to draw any definitive conclusions.

To illustrate the value of allowing stretches to decrease transfer times, we define a slightly changed IT-VSP$^B$ that minimizes transfer costs within a budget for operational costs, rather than the weighted sum of both. Specifically, the IT-VSP (3.1) - (3.20) is changed by replacing objective (3.1) with (3.21) and adding budget constraint (3.22). Thus IT-VSP$^B$ is defined as

$$\begin{align*}
\min & \quad c^{TR} \sum_{r \in R} f_r \gamma_r \\
& \quad \sum_{(i,j,k) \in Q} c_{ijk}x_{ijk} + c_i^{DW} \sum_{i \in T} \delta_i \leq \Delta
\end{align*}$$

The objective function (3.21) considers only transfer cost minimization. The operational cost components of (3.1) are removed from the objective and an additional constraint is added to the model, (3.22), where the operational cost components removed from (3.1) are kept below or equal to a certain budget $\Delta$. In these experiments, $\Delta$ is defined as the operational costs in the optimal solution to the IT-VSP($\emptyset$) for the respective instance.

Table 3.8 contains transfer costs, operational costs, and total costs for the 3, 5 and 8 lines instances, with and without allowing stretches. Experiments were run for 5 hours total computation time and the Rand selection strategy, as there was no strategy that consistently performed best. Moreover, the increased complexity introduced by the budget constraint and changed objective in IT-VSP$^B$ required to increase run time per iteration of the 8-line instance to 20 minutes.

Table 3.8: Budget results with a run time of 5 hours, with and without stretches, expressed in TrC, OpC, and TC

<table>
<thead>
<tr>
<th>Instance</th>
<th>3 Lines</th>
<th>5 Lines</th>
<th>8 Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TrC</td>
<td>OpC</td>
<td>TC</td>
</tr>
<tr>
<td>Stretches</td>
<td>No</td>
<td>-35.75</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>-47.19</td>
<td>-0.06</td>
</tr>
</tbody>
</table>
The results in Table 3.8 indicate that indeed adding stretches can reduce transfer costs by approximately 11% in all instances, without increasing the operational costs. With stretches, transfer costs reductions are increased from between 30.61% and 37.81% to between 42.03% and 48.81%.

### 3.7.3.2 Schedules and Excess Transfer Time

In this section, we compare the vehicle schedules and the excess transfer times in the original timetable MDVSP solution and in the best solution obtained using our MHeu approach. Results refer to the 8-line instance, allowing stretches, and with a total computation time of 5 hours. Figure 3.8 analyzes the quality of transfers in terms of excess transfer time, which is represented in the x-axis. The y-axis shows the percentage of passengers that experience each value of excess transfer time.

![Figure 3.8](image)

**Figure 3.8:** Comparison of transfers in the original solution and in the Relat best solution

In the MHeu solution, less passengers experience high excess transfer times. It can be observed that the number of passengers with ideal transfer time (experiencing zero minutes excess transfer time) increased by approximately 175%. In the MHeu solution a total of 34% of all passengers experience 0 minutes of excess transfer time, while only 12.8% of passengers experience 0 minutes excess transfer time in the original timetables. Furthermore, the average excess transfer time decreased from 4.5 minutes to 2.8 minutes in the MHeu solution, and the worst case excess transfer time was also reduced.

Figure 3.9 shows the number and duration of vehicle schedules for the best solution of the MHeu and for the MDVSP solution resulting from the original timetable. The x-axis contains the duration of schedules in minutes, while the y-axis indicates the number of schedules with a duration up to x minutes. The schedule durations are discretized in intervals of 50 minutes, so a value of 2 in the 600 minute duration means that there are two schedules with duration between 550 and 600 minutes.
The *MHeu* solution reduces the number of schedules from 205 to 195, which corresponds to a decrease of approximately 5%. Furthermore, the average duration of schedules increased by approximately 30 minutes. The percentage of modified trips in the original timetable, by either shifts, stretches, or both, is approximately 74%. A total of 78 trips are stretched. On average, circa 1.5 minutes of dwell time are added per stretched trip. This indicates that the increase in in-vehicle time for on-board passengers will be very limited.

We present further information on the vehicle schedules in Figure 3.10 and Tables 3.9 and 3.10. Figure 3.10 shows the number of trips per schedule in the original timetable MDVSP solution and in the best solution obtained with the *MHeu* approach. It can be observed that the *MHeu* generates schedules with a higher number of trips, being the average number of trips per schedule 8.13, while in the MDVSP solution this average amounts to 7.73.

![Figure 3.9](image.png)

**Figure 3.9:** Comparison of schedules in the original solution and in the *Relat* best solution

![Figure 3.10](image.png)

**Figure 3.10:** Number of trips in schedules in the original solution and in the *MHeu* best solution

Table 3.9 shows the number of trips and number of schedules assigned to each depot.
3.7 Computational experiments

Table 3.9: Number of trips and number of schedules assigned to each of the four depots in the original and in the MHeu schedules

<table>
<thead>
<tr>
<th>Depot</th>
<th>Trips</th>
<th>Schedules</th>
<th>Trips</th>
<th>Schedules</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>616</td>
<td>75</td>
<td>630</td>
<td>75</td>
</tr>
<tr>
<td>2</td>
<td>461</td>
<td>55</td>
<td>456</td>
<td>52</td>
</tr>
<tr>
<td>3</td>
<td>376</td>
<td>55</td>
<td>356</td>
<td>49</td>
</tr>
<tr>
<td>4</td>
<td>132</td>
<td>20</td>
<td>143</td>
<td>19</td>
</tr>
</tbody>
</table>

In both solutions, depot 1 is at its maximum capacity, from which we conclude it is the depot with most convenient location given the set of trips to be serviced. Furthermore, the number of trips assigned to this depot in the improved solution increases, reinforcing the importance of depot 1. Depot 4, although being the depot with lowest number of schedules, also has an increased number of trips in the MHeu solution. Depots 2 and 3 see a decrease in both number of trips and number of schedules when comparing to the MDVSP solution.

Table 3.10: Distribution of trips per depot in the original and in the MHeu schedules

<table>
<thead>
<tr>
<th>Depot</th>
<th>MHeu schedules</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Original Schedules</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>383 (24.2%)</td>
</tr>
<tr>
<td>2</td>
<td>134 (8.5%)</td>
</tr>
<tr>
<td>3</td>
<td>93 (5.9%)</td>
</tr>
<tr>
<td>4</td>
<td>20 (1.3%)</td>
</tr>
</tbody>
</table>

Table 3.10 provides information on how trips are assigned to depots, both in absolute value and in percentage of total number of trips. For example, trips in entry (1,1) are assigned to depot 1 in both the original and in the MHeu solutions, while trips in entry (1,2) are assigned to depot 1 in the original solution and to depot 2 in the MHeu solution. While 57.2% of all trips (907 trips, sum of the diagonal values) are assigned to the same depot in both solutions, 42.8% of all trips shift depots in different solutions, confirming the significant impact that timetable modifications have on vehicle schedules.

3.7.3.3 Trade-off between Operational Costs and Transfer Costs

In this section, we analyze the trade-off between operational costs and transfer costs by varying the value of time (VOT) for passengers. The VOT considered in the previous experiments was 100 DKK/hour, which corresponds to approximately 14 USD/hour. We considered the additional VOTs= \{50, 200, 400\} DKK/hour, with 50 DKK/hour representing a very low value of time, 100 DKK/hour a standard value of time for commuters, 200 DKK/hour representing business travelers, and 400 DKK/hour representing an extreme high value of time. These values were inspired by value of time studies conducted at
the transport modelling center at the Technical University of Denmark, and are available at their website[1]. The experiments were run for 5 hours of total running time, for the 8-line instance, allowing stretches, and the Rand strategy. The plots in Figure [3.11] have different VOTs represented in the x-axis and the percentage differences in the y-axis.

![Figure 3.11: Analysis of the influence of Value of Time](image)

The plots show that increasing the VOT leads to lower transfer costs and higher operational costs. However, the sensitivity of this relation appears to be low. For a value of time of 400 DKK/hour instead of 100 DKK/hour, the operational costs increase by 3.12% (-0.47 - -3.59) for a decrease of -9.42% (-46.61 - -37.19) in transfer costs.

### 3.8 Conclusion

This paper proposed a new model for the integrated timetabling and vehicle scheduling problem. Provided an initial timetable, it defines a set of timetable modifications and a set of vehicle schedules with the objective of minimizing passenger excess transfer times and operational costs. Modifications consist of changes in the start time of a trip (shifts), and addition of dwell time at intermediate stops (stretches). The new idea to include the addition of stretches could represent the redistribution of buffer time over trips to create better-timed transfers. Results for a realistic case study for the Greater Copenhagen area indicate that the integration of timetabling and vehicle scheduling may lead to a potential reduction in both transfer costs and operational costs. Moreover, our findings suggest that allowing a wider set of timetable modifications in the form or stretches creates a potential for further reducing transfer costs by up to 10%, without increasing operational costs.

We propose a matheuristic that in each iteration solves the MIP formulation for a sub-problem of the IT-VSP. The sub-problems restrict timetable modifications to a subset of all timetabled trips, while it solves the full integrated vehicle scheduling problem. Several methods for constructing sub-problems are proposed and compared. Results indicate that the matheuristic is able to produce better solutions in terms of transfer time and operational costs than a general purpose solver in 7 days, and does so faster in 1 to 5

hours of computation time. Solutions reduce average excess transfer time in comparison to the current timetable from 4.5 to 2.8 minutes, while increasing the number of passengers with ideal transfer times by almost 175% and decreasing worst case excess transfer times. In addition, reductions in operational costs are found in comparison to optimal vehicle schedules for the current timetable. Results for our case study are therefore promising that also for larger networks gains could be obtained from the integrated approach.

Several opportunities for future research exist. First, one could aim to include a dynamic passenger route choice component into the optimization. Indeed, if more favourable transfer connections are provided some passengers may change their route, thus leading to a change in passenger flows. This would also allow a better analysis of the trade-off between added dwell time and increase in travel costs for on-board passengers. The modelling of accurate passenger route choice is a non-trivial task, and therefore left for future research. One could also aim to include crew scheduling constraints into the model, to ensure that there exist feasible crew schedules for the resulting vehicle schedules. Finally, it would be interesting to consider the integration of timetabling and vehicle scheduling in a real-time setting. This would require a focus on increasing computation speed, include new practical constraints for en-route vehicles and drivers, and evaluation of results in a dynamic setting.
A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling
Chapter 4

Passenger service optimization through timetabling with free passenger route choice

with Tobias Zündorf\textsuperscript{a}, Evelien van der Hurk\textsuperscript{b}, Allan Larsen\textsuperscript{b}

\textsuperscript{a}Institute of Theoretical Informatics, Karlsruhe Institute of Technology, Karlsruhe, Germany
\textsuperscript{b}DTU Management - Department of Technology, Management and Economics, Technical University of Denmark, Kgs. Lyngby, Denmark

Publication Status: submitted to \textit{OR Spectrum: Quantitative Approaches in Management}, on July 3\textsuperscript{rd} 2019

Abstract: Designing a public transport timetable that maximizes passenger service, measured in weighted travel time, is an intricate problem. The weighted travel time depends on the free route choice of passengers. Passenger route choice depends on the timetable. In turn, the timetable that minimizes weighted travel time depends on the route choice of passengers – and therefore requires passenger route choice information. Consequently, a sequential approach where timetables are designed provided pre-fixed passenger assignment to routes, may not find the optimal timetable.

This paper aims to integrate passenger route choice and timetabling. It addresses the problem of designing maximal passenger service public transport timetables in systems with free route choice within a budget for operating costs. Operating costs are defined by the minimal cost vehicle schedule required to operate the timetable.
The proposed methodology integrates a matheuristic for timetabling and vehicle scheduling with a passenger assignment model in an iterative framework, where different forms of integration are evaluated. Focus is on long to medium term timetabling, provided an initial timetable. Results for a realistic case study in the Greater Copenhagen area indicate that our approach consistently leads, at no additional cost, to timetables that represent a reduction in passenger weighted travel time in comparison to both an initial timetable and a non-integrated timetabling method that receives a single passenger assignment as input.

**Keywords:** Public Transport · Bus Timetabling · Passenger Route Choice · Mixed Integer Linear Programming · Matheuristic

### 4.1 Introduction

Timetabling consists of assigning specific points in time to a set of events. In bus timetabling, this set of events follows from the service network designing, determining a set of lines, each with a given ordered list of stops and a frequency. The output of the timetabling phase serves as the input to the vehicle scheduling problem, that assigns vehicles to specific services in the timetable. Timetabling generally represents a balance between high quality passenger service, and low operating costs. Passenger service depends on the timetable, while the operating costs depend on the vehicle assignment. Passenger service can be expressed in terms of weighted travel time (WTT): a weighted sum of the components of initial waiting time (IWT), in-vehicle time (IVT), and transfer time (TrT) ([Comi et al.](2017)). Operating costs are expressed as a function of the number of required vehicles and dead-heading distance.

Integrating passenger route choice and timetabling is an intricate problem, as there is a co-dependency between passenger route choice and timetable design. Indeed, minor changes in the timetable can have large effects on the WTT of routes with a transfer. Especially in dense networks, small changes in the timetable could lead to far larger changes in transfer times, and consequently may cause passengers preferring alternative geographical routes. As a result, providing a fixed passenger assignment as input to the timetabling phase could lead to finding sub-optimal timetables. Furthermore, also the vehicle schedules can be strongly affected by small changes in the timetable. To ensure reasonable operating costs of the resulting timetable, it is vital to consider vehicle scheduling as well.

This paper studies the Integrated Passenger Assignment, Timetabling and Vehicle Scheduling Problem (IPAT-VSP) at a tactical level. The objective of the IPAT-VSP is to maximize passenger service in terms of WTT within a budget for operating costs and a set of headway constraints. Input consists of a non-cyclical initial timetable with time-dependent service times, and an Origin-Destination-Time (ODt) passenger demand matrix. Timetable decisions consist of shifting departure times of trips, or extending dwell time at transfer stops, with respect to the provided initial timetable. Operating costs are
Key characteristics of passenger route choice as considered in this study are that (i) passengers have free route choice, and (ii) passengers may have different route preferences. Thus passengers with the same origin, destination, and departure time, may still choose different routes. The first is in contrast to the common setting of a central controller assigning passengers to maximize a social optimum. The second is in contrast to assuming all passengers choose the minimum weight path, and reflects that passengers may have different preferences. Specifically, it will ensure that when two almost equivalent paths exist connecting an origin and destination, passengers will be assigned to both. This assumption fits with commonly accepted route choice theory (Ben-Akiva et al. [2004]).

A matheuristic approach for the IPAT-VSP is proposed that consists of an iterative framework between a passenger route choice model, and an integrated timetabling and vehicle scheduling model. Different forms of integration are evaluated. The implementation of this modular framework combines the integrated timetabling and vehicle scheduling model of Fonseca et al. (2018b) with the passenger route choice model of Briem et al. (2017), that satisfies the above two key characteristics of free passenger route choice, and different preferences for passengers. The latter also serves as a ground-truth for evaluating the passenger service of any timetable.

A realistic case study representing a large part of the multi-modal public transport network of the Greater Copenhagen Area, Denmark, serves to investigate the value of the IPAT-VSP timetabling approach: (i) in comparison to a the status-quo reflected by an initial timetable; (ii) in case of a change in the line network; and (iii) in case of a change in the OD matrix. Results indicate that including free passenger route choice results in timetables with higher passenger service in all three situations compared to a fixed passenger route choice approach as proposed in Fonseca et al. (2018b). Moreover, our computational studies, supported by a simple clarifying example, illustrate that in order to find timetables with high passenger service indicating potentially interesting transfers is more important than estimating accurate usage of transfers in a current timetable.

To summarize the contributions of this work: (i) we investigate the maximal passenger service timetabling problem in the context of a free passenger route choice; (ii) we propose a modular matheuristic approach for the IPAT-VSP that, in an iterative framework, combines two state of the art models: (1) the IT-VSP matheuristic of Fonseca et al. (2018b), which maximizes passenger service through minimizing excess transfer time under the assumption of fixed passenger route choice, with (2) the passenger route choice model of Briem et al. (2017), which represents free route choice of passengers; and (iii) we find that the inclusion of free passenger route choice results in timetables with higher passenger service for a realistic case study of the Greater Copenhagen area. Thereby the current study is different from Fonseca et al. (2018b) by (a) indicating the value of integrating passenger route choice and timetabling, where Fonseca et al. (2018b) assumed the passenger route choice as fixed input; (b) demonstrating the value of this approach for a larger, more complex network case study, and contrasting this against the approach
Passenger service optimization through timetabling with free passenger route choice

Fonseca et al. (2018b), and (c) the evaluation of the value of the timetabling approach not only in comparison to the status-quo, but also in case of a small network re-design, and a change in passenger demand. The later two would lead to a change in (expected) passenger flows, which this model demonstratively is better capable of handling than the model of Fonseca et al. (2018b).

The remainder of this paper is organized as follows: Section 4.2 reviews previous work on the integration of timetabling and passenger route choice, Section 4.3 describes the IPAT-VSP and presents a small example, Section 4.4 describes all components the solution approach for solving the IPAT-VSP, Section 4.5 describes the case study, Section 4.6 discusses the results for the case study, and Section 4.7 shows conclusions and suggestions for future research.

4.2 Literature Review

Recent years saw an increase in research output that integrates passenger decisions into the optimization models, especially in line planning, timetabling, and delay management models. Schmidt (2014) provided an overview on public transport problems integrated with routing decisions.

Schmidt and Schöbel (2015) integrate timetabling with passenger route choice minimizing total travel time, and investigate the computational complexity of the problem. Gattermann et al. (2016) present a boolean satisfiability problem (SAT) model that integrates periodic timetabling with passenger routing, distributing OD pairs temporally using time slices to make the problem tractable. The model is tested on Germany’s long-distance passenger railway network and, for a restricted set of OD pairs, results show better objective values when compared with previous results. Computational time increases considerably when the number of OD pairs increases. Borndörfer et al. (2017) study the integration of passenger routing with periodic timetabling models and propose a variety of models that allow different passenger paths and different objective functions. For a case study in the city of Wuppertal, the authors report a reduction of 1.24% in travel time and 23.57% in transfer waiting time in comparison to a real-world reference solution. Zhu et al. (2017) present a bi-level model to integrate single line timetabling with passenger routing. The first level determines the headways to minimize total passenger costs (perceived travel time and travel penalties), and the second level determines the passenger arrival times given the headways. The authors use a two stage genetic algorithm to solve hypothetical examples of the problem. Chu (2018) presents a mixed integer program to integrate network design and timetable, while routing passengers through a procedure based on the breadth-first search and path enumeration algorithms. A branch-and-price-and-cut algorithm is presented to solve the problem, and results are presented for example networks. Robenek et al. (2018) address train timetabling design considering a probabilistic demand forecasting model. The demand elasticities are calibrated using a logit model. The problem is solved using a simulated annealing heuristic and results for a case study of Israeli
Railways show revenue increases of 15%. Wu et al. (2019) present a bi-level program to coordinate timetabling and consider passengers’ behavior to the timetable modifications by rerouting passengers that in the new timetable have missed transfers. The first level uses a mixed integer non-linear program to design the timetable, minimizing system cost composed by operating and user costs. The second level is a passenger route choice model. The authors use a heuristic algorithm to solve the problem and show results for two small examples: one with 3 lines and 3 transfer stops, and another with 4 lines and 4 transfer stops.

Laporte et al. (2017) integrate timetabling and vehicle scheduling including special attention to route choice. Their problem designs timetables keeping operating costs (expressed as number of vehicles per line) under a certain budget. The Pareto front of solutions is calculated using an $\epsilon$-constraint solution approach. Also in Liu and Ceder (2017), passenger route choice is integrated with timetabling and vehicle scheduling. The problem is modeled using a bi-objective, bi-level IP formulation, optimizing fleet size and WTT. The authors allow timetable modifications by shifting departure times and initial vehicle schedules are given as input. Deadheading is not allowed, meaning that vehicles are assigned to a single line and can only service trips belonging to that line, which significantly reduces the complexity of the problem. They propose a deficit function based sequential search to solve small examples with up to 4 unidirectional lines, 4 transfer stops, and one hour of operations.

Integration of free passenger route choice with transport optimization problems is also present in other fields. Dumas and Soumis (2008) combine a fleet assignment optimization model with a passenger flow simulation model in an integrated model for the airline booking process. The passenger simulation model in Dumas et al. (2009) describes the passengers’ reaction to capacity restrictions and how their itineraries change if their preferred flights are sold out. Cadarso et al. (2013) address disruption management in rapid transit networks, integrating an optimization model for timetabling and rolling stock schedules with a model for passengers’ behavior. Kroon et al. (2014) study the integration of free passenger flows in a real-time rolling stock rescheduling model for disruption management. The authors present a heuristic approach that iterates between a simulation model for passenger flows and an optimization model for the rolling stock, updating the objective function of the optimization model at each iteration according to the current passenger flows. Recent research in integration of free passenger route choice in disruption management can be found in Binder et al. (2017), Veelenturf et al. (2017), Wagenaar et al. (2017), Ortega et al. (2018), and Van der Hurk et al. (2018). Canca et al. (2016) present a mixed integer non linear program to optimize line frequencies (minimizing operating costs and fleet acquisition costs) and simultaneously compute passenger assignments (minimizing average trip time and number of transfers). They use a cutting plane algorithm and present results for a simplification of the Madrid Metropolitan Railway Network.

In comparison to Laporte et al. (2017), we allow a wider set of timetable modifications. We consider free passenger route choice, where passengers may have different preferences, while Fonseca et al. (2018b) considered passenger route choice as input. The proposed
matheuristic framework is inspired by the framework proposed in [Dumas and Soumis (2008)] for the airline booking process and [Kroon et al. (2014)] for disruption management. In comparison to [Wu et al. (2019)], we consider free route choice for all passengers, while in [Wu et al. (2019)] only passengers with missed transfers are re-routed. Furthermore, [Wu et al. (2019)] test the performance of their method in small examples, while we test our method in a much larger real life case study.

4.3 The IPAT-VSP

The objective of the IPAT-VSP is to find the maximum passenger service timetable $T^*$ of all feasible timetables $T \in T$, which we define as the timetable with minimum total weighted travel time. Let $l \in L$ be the set of directed lines, where each line is defined by a sequence of stops $s \in S$, with $S$ the set of all stops. Let a trip $i$ represent a vehicle servicing all stops of a line $l$ once. Each line $l$ is associated with a frequency, leading to a set $T_l$ of trips in the timetable for this line. The timetable is defined by the set of all trips $T = \bigcup_{l \in L} T_l$ and $T_{l'} \cap T_{l''} = \emptyset$ for all $l', l'' \in L, l' \neq l''$. The timetabling problem is to for all trips $i \in T$ assign arrival and departure times to all stops of the trip $s \in S_i$.

Passenger service, measured in weighted travel time, is defined by the route choice of passengers, and depends on the timetable. A (passenger) route choice model calculates a passenger assignment (PA) provided a timetable $T$ and an origin-destination-departure-time matrix $ODt$. A passenger assignment consists of a set of paths $P$, where each path $p \in P$ is associated with a number of passengers that select this path. A path $p \in P$ represents an ordered list of trips $i \in T$ that connect the origin to the destination of the passenger in time and space, where the departure time of the first trip, from the origin stop of the passenger, is not before the departure time of the passenger. The path specifies per trip the boarding and departure stop of the passenger, where when disembarking a trip $i$ passengers have either arrived at their destination, or will transfer to a trip of another line $l \in L$.

The total weighted travel time, that follows from the passenger assignment, is defined as:

$$WTT = \sum_{p \in P} w_p \cdot (\alpha \cdot IWT_p + IVT_p + \beta \cdot TrT_p)$$

where $WTT$ is the total weighted travel time of the passenger assignment, $w_p$ the number of passengers that will select path $p$, $IWT_p$ the initial waiting time of path $p$, $IVT_p$ the in-vehicle time of path $p$, and $TrT_p$ the transfer penalty of path $p$. We assume $\alpha, \beta \geq 1$.

The IPAT-VSP is a bi-level optimization problem: (i) the operator aims to find the maximum passenger service timetable within an operating budget, and (ii) the passengers aim
to find their individual best paths in this timetable. Here we assume that (ii) represents free route choice of passengers. In other words, the social optimum for all passengers may not be equal to the individual optimum of each passenger. In addition, we assume that passengers may have different preferences. The later will ensure that when two paths exist of (almost) equal WTT, some passengers will prefer the one path, others the other path. As a consequence of these properties, it is not straight forward to translate timetabling with passenger free route choice to the context of a mathematical optimization model, where e.g. minimizing total WTT would lead to a social optimum, rather than an individual optimum. Furthermore, timetabling itself is already a computationally hard problem, thus the integration of the two is expected to be intractable as well.

Many models, like Fonseca et al. (2018b), assume a PA as input. Specifically, Fonseca et al. (2018b) assumed as input the number of passengers per transfer location, where a transfer location is defined by a specific (feeder) trip $i \in T$, at a stop $s \in S$, to another (receiver) line $l \in L$, so $i \notin T_l$. The problem is that the number of passengers that will use such a transfer location, depends on the quality of the transfer (that is, excess waiting time above of the minimum transfer time), as well as the availability of alternative paths. Both transfer times and quality of alternative paths depend on the timetable. As the timetable is unknown, the demand per transfer location is unknown, and the input of a non-optimal PA may lead to the selection of non-optimal timetables. Moreover, while the objective to minimize excess transfer time (transfer time longer than the minimum required time to transfer), as in Fonseca et al. (2018b), will lead to minimizing total WTT in case of fixed route choice; when passengers have free route choice and may change their routes, higher total weighted transfer times may actually represent solutions with lower total WTT. The example in Section 4.3.1 will illustrate this.

The example of Section 4.3.1 shows that even perfect information on the PA of an initial timetable cannot guarantee the finding of the optimal passenger service timetable. Specifically, long transfer times in an initial timetable can cause these transfer locations to be overlooked when they are not used in the initial timetable. We hypothesize that identifying these potential beneficial transfer locations, rather than including the set of current transfer locations, can lead to finding timetables with higher passenger service. We demonstrate by a set of case studies that indeed our matheuristic approach is able to find timetables with higher passenger service than (1) models assuming a static passenger assignment as input (specifically, than Fonseca et al. (2018b)), and (2) integration where input consists only of the passenger assignment for an initial, or current, timetable. In Section 4.4 we will present our modular math-heuristic approach to the IPAT-VSP.

### 4.3.1 Example

Consider the example in Figure 4.1 with three bus lines (1, 2, 3) and four stops (A, B, C, D). Consider that all bus lines have a headway of 20 minutes. Information on travel times is indicated along the edges, and a minimum transfer time of 4 minutes is required.
Figure 4.1: Example of a public transport network with three lines and four stops

Passenger service optimization through timetabling with free passenger route choice

to guarantee a successful transfer, as indicated by the curved arrows. Three transfer opportunities exist in this network: at B where lines 1 and 3 meet, at C where lines 1 and 2 meet, and at D where lines 2 and 3 meet.

This example is used to (1) illustrate that when the timetabling model receives a PA input that is different from the PA of the optimal timetable, the timetabling model may not find the optimal timetable; and (2) that although given a fixed PA, minimizing the weighted sum of (excess) transfer time, as in Fonseca et al. (2018b), will lead to minimizing total WTT, this is not true when the final PA is different from the input PA. Even more, timetables with a higher weighted sum of (excess) transfer time could be associated to timetables with a lower total WTT, and thus higher passenger service.

Passengers traveling from A to D have two routing options: (i) traveling with line 1 from A to C, then transfer to line 2 to travel towards D; and (ii) traveling with line 1 from A to B, then transferring to line 3 to travel towards D. The passenger service is reflected in the weighted travel time, which is calculated as $WTT = \alpha \cdot IWT + IVT + \beta \cdot TrT$, where $\alpha, \beta \geq 1$. We may focus on IVT and TrT alone, as for both (i) and (ii) the IWT (dependent on the frequency of the first line) is equal. The IVT of (i) is $5 + 10 + 10 = 25$, which is longer than the IVT of (ii) $5 + 10 = 15$. Passengers will however prefer the longer IVT of route (i) when $25 + \beta \cdot TrT_{(i)} < 15 + \beta \cdot TrT_{(ii)}$.

Consider a timetable where the transfer from line 1 to 2 is perfectly synchronized at C, with no excess transfer time. Moreover, line 3 departs from B one minute before the arrival of line 1 at B, resulting in 19 minutes of transfer time at this location. For a value of $\beta = 3$ even under free route choice all passengers will prefer route (i) with a WTT $= 25 + 3 \cdot 4 = 37$ over route (ii) with a WTT $= 15 + 3 \cdot 19 = 72$. However, passenger service could be increased in this example if timetabling decisions would reduce the transfer time from line 1 to line 3 at B in route (ii) such that $15 + \beta \cdot TrT_{(ii)} < 25 + \beta \cdot 4$. This alternative has much lower IVT in exchange for a transfer penalty and some additional waiting time, as required for the transfer.
4.4 Solution method

Indeed, in our example reducing the transfer time at stop B could improve passenger service by attracting passengers to route (ii). However, as in the current PA no passengers are using this transfer location, the timetabling model has no incentive to improve the synchronization of the two lines at this location. This shows that a PA different from the optimal PA could prevent finding the optimal timetable.

The reduction in transfer time at B could improve total WTT already at a positive excess transfer time (when $15 + \beta \cdot Tr_{(ii)} < 25 + \beta \cdot 4$, with $\beta \geq 1$). However, this would lead to an increase in total weighted excess transfer time of the model in comparison to passengers only using the perfectly synchronized transfer at C. In fact, passengers only transferring at the perfect synchronized transfer in C leads to a minimal objective of 0 minutes excess transfer time, which could suggest the optimal timetable is found. This is true if route choice was fixed. However, as it is not fixed, the total WTT would be lower when there is a low transfer time at B, even if the transfer is not perfectly synchronized.

4.4 Solution method

This paper proposes a modular matheuristic, the $MHeuPA$, to solve the IPAT-VSP. The objective is to maximize passenger service in terms of minimizing WTT by modifying an initial timetable under the assumption of free passenger route choice and respecting a budget on operating costs. It does so by integrating an integrated timetabling and vehicle scheduling model, the $MHeu$ (Section 4.4.1), with a passenger route choice model, the $PTTA$ (Section 4.4.2). As the approach is modular, alternative models for timetabling and passenger route choice could be used as well.

The two main points that need to be taken into account for solving the IPAT-VSP, that result from the assumption of free route choice, are:

1. passenger volumes per transfer location in the current timetable may not represent the passenger volumes per transfer location in the maximal passenger service timetable

2. higher objective values of the model of Fonseca et al. (2018b) could actually be associated with lower overall WTT, and thus higher passenger service.

These two points were illustrated by the example in Section 4.3.1. To address these two points, (1) different passenger routings, by changing the waiting cost parameter in the $PTTA$ model of Briem et al. (2017), are tested in the integration, and (2) timetable quality in terms of WTT is always calculated by the passenger route choice model.

Figure 4.2 outlines the IPAT-VSP matheuristic, the $MHeuPA$, that links the IT-VSP matheuristic ($MHeu$) of Fonseca et al. (2018b) with the passenger route choice model of Briem et al. (2017). Section 4.4.3 will discuss this algorithm in detail.
Input consists of an initial timetable $T_0$ defining departure and arrival times for the set of all timetabled trips $T$, an ODt matrix $ODt$ representing passenger demand over time, a stop criterion $stopCriterion$, and a waiting cost function $CostF$ for the PTTA (Section 4.4.3.1). These cost functions are the key to how the route choice model and timetabling model are working together to solve the IPAT-VSP. We compare different cost functions. Initialization consists of the PTTA calculating passenger assignment for an initial timetable according to the realistic waiting cost value ($\phi^R$), and determining the operating cost budget by calculating the minimum cost vehicle schedule by running a Multi Depot Vehicle Scheduling Problem (MDVSP) for the initial timetable.

The core of the method consists of three blocks executed iteratively. First a PTTA is run, where the waiting cost function $CostF$ may have waiting cost values different from $\phi^R$. This is done to generate different passenger assignments than the ones that would be generated with $\phi^R$, in an attempt to obtain timetables with higher passenger service. The PTTA provides the IT-VSP with a set of passenger transfer demands $R'$ and a vector $\Lambda$ of passengers on board at each stop. Next, the IT-VSP $MHeu$ is solved maximizing passenger service in terms of minimizing excess transfer time and additional in-vehicle time resulting from a possible extension of vehicle dwell time. The timetable is rescheduled within a budget for operating costs.

The resulting new timetable $T'$ is evaluated in terms of passenger service according to the PTTA using $\phi^R$. The iterative procedure stops after the maximum running time is reached, and outputs the computed timetable with highest passenger service $T^*$ (the one with lowest WTT), the vehicle schedules $X^*$ that cover that timetable and respect the budget constraint, and the passenger assignment $A^*$ associated with timetable $T^*$ computed by PTTA. Note that the framework is modular in that any passenger route choice model could be used instead of PTTA, and the timetabling module IT-VSP could be replaced by a different model.
4.4 Solution method

4.4.1 Integrated timetabling and vehicle scheduling – the IT-VSP matheuristic approach

Input to the IT-VSP consists of an initial timetable for the set of all trips $i \in T$, passenger route choice information, a budget for the operating costs, and costs and parameters related with the case study, such as allowed headways, maximum dwell times, or turnaround times. The passenger route choice information consists of a set of transfer opportunities $R$, where each $r \in R$ defines a transfer stop, a transfer-from trip $i \in T$, a desired to transfer to line $l$, and a number of passengers that are expected to make this transfer. Furthermore, the PA contains the expected on-board passengers per trip $i \in T$ at stop $s \in S_i$, $\Lambda_{is}$. The passenger route choice information is computed in the Public Transport Traffic Assignment (PTTA) model (Section 4.4.2), which requires a timetable as input.

The objective of the IT-VSP is to minimize a weighted sum of passenger costs incurred by extending dwell times at stops for passengers on-board, and passenger costs incurred when transferring. Passengers incur transfer time costs when transfers are above the minimum transfer time. Transfers below the minimum transfer time are infeasible.

Decision variables consists of: a) binary assignment variables $x_{ijk} \in \{0, 1\}$ storing which vehicles are assigned to which trips, b) departure and arrival time variables $\tau_{ids}$ and $\tau_{ais} \in \mathbb{Z}_0^+$ for each trip $i \in T$ and each stop $s \in S_i$, c) excess transfer time variables $\gamma_r \in \mathbb{R}_0^+$, which store the amount of excess transfer time for passengers using each transfer location $r \in R$, d) binary transfer variables $\alpha_{ijis} \in \{0, 1\}$, which indicate which trip $j \in T$ passengers of transfer location $r = (i, l, j, s) \in R$ embark, and e) dwell time variables $\delta_{is} \in \mathbb{Z}_0^+$, which store the number of minutes of dwell time added to trip $i \in T$ at stop $s \in J_i$.

Constraints ensure the assignment of all expected transferring passengers to specific transfers, ensure headway constraints to be within a specific range, and limit the amount of added dwell time per stop and per trip, in addition to feasibility constraints for both the timetable and the vehicle schedule. Trips can only have additional dwell time and not less dwell time than in the initial timetable, since the dwell time in the initial timetable is considered to be the minimum dwell time. In fact, for the case study addressed in this paper, most trips have a dwell time of zero at all visited stops. The output consist of a timetable and a vehicle schedule.

The full mathematical model is presented in Appendix 1 (Section 4.8). This is an extended version of the model in Fonseca et al. (2018b). This extended version allows to explicitly include the effect of extended dwell time for on-board passengers.

The IT-VSP is solved by a matheuristic approach based on the MILP formulation described above, which we denote by $MHeu$. A heuristic is necessary since solving real-life instances of the IT-VSP directly with a general solver is not feasible, due to the size of the instances making the problem intractable. The $MHeu$ solves the IT-VSP by iteratively solving a sub-problem IT-VSP($T''$) that only allows timetable modifications for a subset
of trips $T' \subseteq T$, but still solves the full vehicle scheduling problem. The heuristic stores the solution with lowest weighted travel time encountered so far.

4.4.2 Passenger route choice model – the PTTA model

We measure passenger service using the PTTA to evaluate the WTT of a timetable. Furthermore, the PTTA provides input to the \texttt{MHeu} in terms of the set of transfer opportunities $R$ and the number of passengers on board $\Lambda_{is}$.

Input to the passenger route choice model PTTA is a timetable $T$, a set of possible transfer locations, the minimum required transfer time for a transfer to be feasible, and an ODt matrix. Output of the PTTA are the set of transfer opportunities $R$, the number of passengers on board for each trip $i \in T$ and each stop $s \in S_i$, $\Lambda_{is}$, and the WTT of the resulting passenger assignment.

To evaluate the quality of a timetable, cost parameters are set to use perceived arrival times (PAT) where the actual arrival time is weighted with a factor of 1, waiting time is weighted with a factor of 2, and each transfer receives an additional penalty of 5 minutes on top of the waiting time. When the PTTA model serves as input to the timetabling heuristic IT-VSP, different weights for the waiting time component are evaluated. In the route choice set generation, maximum difference in PAT is set to $\Delta_{\text{max}} = 15$ minutes. As random utility model we use the linear decision model that was used in Briem et al. (2017).

The PTTA model is used to estimate which routes are likely to be chosen by passengers in a system with free route choice, as well as the expected number of passengers per route. The underlying model and algorithm to compute these routes were first presented in Briem et al. (2017). Conceptually, the PTTA model is a sequential route choice model. This means, that decisions are not made based on complete routes, but one journey leg at a time. Given a passenger, a current location, and a destination, the model specifies which step is probably taken next by the passenger in order to reach the destination. The probability for every possible next step is determined using a random utility model, with the utility being influenced by several cost functions, such as travel time, waiting time, and number of transfers. The PTTA model iteratively repeats this process until every passenger reached its destination, thereby compiling the complete routes used by the passengers.

For a given destination and journey leg, the random utility model in the PTTA characterizes the likelihood of the journey leg being used as next leg of a route leading to the destination. Thus, the first step of computing the overall passenger assignment in the PTTA model consists of computing the utilities for all pairs of possible journey legs and destinations. The PAT at the destination when using the specific leg in turn determines the utility of a leg (for a given destination). The PAT is a linear combination, which besides the
actual arrival time, factors in all criteria that effect the route choice. For this work, the PAT is the weighted sum of the actual arrival time, the number of transfers, and the time spent waiting for the next trip. An important aspect of the PTTA model is the algorithm that allows for an efficient computation of PATs for all pairs of journey legs and destinations. To this end, PAT values are computed for one destination at a time. For a given destination, the PATs of all possible journey legs are computed iteratively, sorted by time in decreasing order. A detailed description of the process can be found in Briem et al. (2017). The benefit of processing the journey legs in decreasing order of time is that for a given leg the PATs of all possible journey continuations are already known. Thus, the PAT of every leg can be computed quite efficiently. If the leg itself ends at the destination then the PAT of this leg is given by its actual arrival time (since a single leg does not comprise transfers by definition). If the leg does not end at the destination, then the route has to be continued with another leg. In this case the PAT is given as the PAT of the following leg (which is already known), plus the additional cost (transfer time, waiting time) to connect the legs.

After all PATs have been computed, the actual passenger route choice is determined using a simulation approach. For every passenger, a sequence of decisions is made based on a random utility model. Each of these decisions determines the next journey leg the passenger takes towards the destination. The choice set for this decision is determined on basis of the PAT values. It consists of the leg with the lowest PAT as well as all other legs, such that the difference of PATs in the choice set does not surpass a certain, user-defined limit ($\Delta_{\text{max}}$). The utility of each leg $\ell$ in the choice set is then defined as $\max(0, \min_{\ell' \neq \ell} \text{PAT}_{\ell'} - \text{PAT}_\ell + \Delta_{\text{max}})$. Finally, the probability of each leg in the choice set can be obtained using a random utility model. Proportional to these probabilities one leg is chosen, e.g. the passenger is assigned to this leg as part of his route. This process is repeated until all passengers have been assigned to full routes, reaching their destinations. In order to obtain a distribution of several routes that could be used by a passenger (alongside with their respective probabilities), several virtual passengers can be simulated for every actual passenger.

4.4.3 IPAT-VSP matheuristic – the MHeuPA

Algorithm 2 presents the pseudo code for the MHeuPA approach.

Steps 1-5 are the initialization procedure. The algorithm starts by calculating an assignment for the initial timetable $T_0$ in step 1, using the PTTA with $\phi^R$. The set of transfer opportunities $R_0$ and the vehicle occupancy $\Lambda_0$ computed by the PTTA are used as input to solve the MDVSP in step 2 without allowing timetable modifications, thus the departure and arrival times of all trips $i \in T$ from/to stop $s \in S$, $\tau_{is}^d$, $\tau_{is}^a$, are fixed to $T_0$ (meaning that these trips will have arrival and departure times at all stops visited equal to the ones in the initial timetable $T_0$). An initial solution $S_0$ is defined in step 3, composed by vehicle schedules $X_0$, the initial timetable $T_0$, and the initial assignment $A_0$, and since
Algorithm 2: MHeuPA

**Input:** $T$, $T_0$, $ODt$, stopCriterion, $\phi^R$, CostF

Initialization procedure:

1: $(A_0, R_0, \Lambda_0) \leftarrow \text{PTTA}(T_0, ODt, \phi^R)$
2: $(X_0, T_0) \leftarrow \text{solve IT-VSP (1)-(21)}$ using $R_0$ and $\Lambda_0$ and with departure and arrival times $(\tau^d_{is}, \tau^a_{is})$ fixed to $T_0$ for all $i \in T$
3: $S_0 = (X_0, T_0, A_0)$
4: $S^* = S_0$
5: $\eta = 0$

Iterative procedure:

6: while stopCriterion not reached do
7:    $\eta = \eta + 1$
8:    $T' \leftarrow \text{selectTrips}(S_{\eta-1})$
9:    $(A', R', \Lambda') \leftarrow \text{PTTA}(T_{\eta-1}, ODt, \text{CostF})$
10:   $(X_{\eta}, T_{\eta}) \leftarrow \text{solve IT-VSP (1)-(21)}$ using $R'$ and $\Lambda'$ and with departure and arrival times $(\tau^d_{is}, \tau^a_{is})$ fixed to $T_{\eta-1}$ for all $i \in T \setminus T'$
11:   $(A_{\eta}, R_{\eta}, \Lambda_{\eta}) \leftarrow \text{PTTA}(T_{\eta}, ODt, \phi^R)$
12:   $S_{\eta} = (X_{\eta}, T_{\eta}, A_{\eta})$
13:   if WTT$(S_{\eta}) < \text{WTT}(S^*)$ then
14:       $S^* \leftarrow S_{\eta}$
15:   end if
16: end while
17: return $S^*$
this is the only solution so far, in step 4 it is also saved as the current best solution in terms of WTT. The iterative procedure is described in Lines 6 - 16, which runs until the stop criterion \textit{stopCriterion} is met.

Each iteration \( \eta \) starts by selecting in step 8 the subset of trips \( T' \subset T \) to modify, being \( \kappa \) the number of trips selected. Trips in \( T' \) are allowed modifications in arrival and departure times (shifts and stretches), while all other trips \( i \in T \setminus T' \) remain fixed to the timetable in solution \( S_{\eta-1} \). In step 9, an assignment of passengers is calculated for the current timetable \( T_{\eta-1} \) according to \textit{CostF}, generating transfer opportunities \( R' \) and vehicle occupancy \( \Lambda' \). A new timetable \( T_\eta \) and vehicles schedules \( X_\eta \) are calculated in step 10, solving the restricted IT-VSP(\( T' \)), with \( \tau^d_i, \tau^a_i \) fixed to \( T_{\eta-1} \) for all \( i \in T \setminus T' \), optimizing the transfer opportunities \( R' \) considering the vector \( \Lambda' \) of passengers on board. The realistic assignment (obtained using \( \phi^R \)) of passengers \( A_\eta \) to the new timetable is calculated in step 11 by running the realistic PTTA. The iteration solution \( S_\eta \) is set in step 12. Steps 13-15 save \( S_\eta \) as the best solution \( S^* \) if the weighted travel time associated with it is lower than the weighted travel time associated with the current best solution. The IPAT-VSP concludes in step 17 by returning the best solution \( S^* \) found once the stop criterion \textit{stopCriterion} is met.

### 4.4.3.1 Waiting cost functions

As demonstrated in the example in Section 4.3.1, the objective is to find all \textit{potentially beneficial transfer locations}, that is, transfer locations that, with a good synchronization of the transfer, could be used by passengers. Whether the transfer location will be used by passengers, depends on the quality of the set of alternative paths available, and therefore cannot be determined per transfer location independently. Different waiting cost functions are used in a desire to find all potentially beneficial transfer locations.

Each run of the \textit{MHeuPA} uses exactly one of five different waiting cost functions \textit{CostF}. These functions change the weight attributed to the waiting costs when computing a new passenger assignment to serve as input to the IT-VSP \textit{MHeu}.

- **Realistic (Realistic):** this waiting cost function runs the PTTA model with the realistic value \( \phi^R \) for the waiting costs at every iteration. It reflects a base-case where the PA of a current timetable is provided as input to the timetabling module.

- **No waiting costs (NoCosts):** this waiting cost function runs the PTTA model without waiting costs at every iteration. This will lead to passengers selecting the path with the minimal IVT.

- **Linear ascending costs (LinAsc):** this waiting cost function increases the waiting costs in the PTTA model per iteration of the \textit{MHeuPA}. Assuming a total running time of \( maxT \) seconds, the PTTA iterations in the first \( maxT/10 \) seconds use a waiting costs parameter of 0. In the remainder of the iterations, the waiting costs increase
linearly until the realistic value is achieved by the end of the experiment. The
waiting costs at each iteration can be calculated using

\[ WC = \frac{\phi t}{\max T} \]

where \( t \) is the current total running time.

- Random waiting costs (Random): this waiting cost function runs the PTTA model
  with random waiting costs at every iteration of the MHeuPA, with a value between 0
  and the realistic waiting costs \( \phi = 2 \), with a uniform distribution.

- Random and linear ascending costs (RandLinAsc): this waiting cost function com-
bines the Random and LinAsc waiting cost functions. In the initial two thirds of the
computational time, the PTTA model iterations use a random value between 0 and
the realistic value for the waiting costs. In the last third, it uses linear ascending
costs, calculated using

\[ WC = \frac{\phi(t - 2\max T/3)}{\max T - 2\max T/3} \]

where \( t \) is the current total running time.

The advantage of a lower waiting time costs is that transfer locations that currently have
high waiting time, but would provide low in-vehicle time paths, will attract passengers –
and thus provide an incentive to the timetabling model to improve the synchronization of
trips at these transfer locations. The downside is that in dense networks it is unlikely that
all transfer locations will be able to provide perfect transfers. Thus, the real passenger
assignment is likely to be different from the zero or small waiting cost assignment; and
therefore the trade-offs made in the timetable may be non-optimal due to erroneous
numbers of expected passengers per transfer location, and expected passengers on-board.

Additionally we define Route_Fixed as running the MHeuPA with a fixed routing of pas-
sengers, using then the PTTA to evaluate the solutions in terms of WTT only. The results
using Route_Fixed correspond to the IT-VSP MHeu results evaluated in WTT, and are
included in this paper to enable a comparison with Fonseca et al. (2018b).

Finally, any timetable is evaluated by running an independent passenger assignment of
PTTA with a ground-truth costs for waiting time. In our experiments, we will use the value
of 2, which was selected in Briem et al. (2017).
4.5 Case study

For the experimental section, we focus on a subset of the public transport network in the Greater Copenhagen area. We consider 8 bi-directional express-bus lines, referred to as S-Bus lines. In comparison to regular bus lines, these are faster and with fewer stops, acting mainly as a complement to the local urban trains (S-Train) across and radially. Figure 4.3 depicts a geographical representation of the network, which includes not only the S-Bus and S-Train lines but also two bi-directional Metro lines, one bi-directional train line, and one high-frequency bus line that connects the city center to the airport. The public transport agency Movia, which is responsible for the planning of buses in the region of Zealand, provided data for the case study. We allow timetable modifications by shifts and stretches to the trips in the S-Bus lines, while all other lines in the case study operate according to a fixed timetable. The vehicle scheduling component of the IPAT-VSP is solved solely for the S-Bus trips.

The data input for the IPAT-VSP is composed by: (i) a initial timetable for all S-Bus lines; (ii) a fixed timetable for all other lines in the network; (iii) a distance matrix with all distances between stops and depots; (iv) an ODt matrix describing passenger demand for the full network; (v) costs and parameters specific to the case study: minimum and maximum turnaround times, minimum transfer times at stops, vehicle operating costs, fixed costs per vehicle schedule created, travel time and waiting time costs for passengers, driving speed for vehicles while deadheading, maximum deadhead distance, maximum...
added dwell time per trip and per stop, and depot capacities.

The timetables used in input components (i) and (ii) are publicly available and the distance matrix (iii) was obtained using geographical data. The ODt matrix (iv) was provided by Rapidis\footnote{Rapidis is a Danish company that develops tools for planning in Transportation and Logistics. website: http://www.rapidis.com/}. It describes minute-by-minute passenger demand between the stations in our network. The ODt contains 164,333 entries representing 170,117 passengers. As for costs and parameters (v), these were estimated together with Movia. We use estimates of operating waiting time, distance, and schedule costs expressed in Danish kroner (DKK, 1 euro is equivalent to approximately 7.5 DKK), which together define the operating costs of a solution. Travel times, initial waiting times, and transfer times are weighted by value of time (VOT) factors of respectively 100, 300, and 300 DKK. We used value of time studies developed at the Center for Transport Analytics at the Technical University of Denmark as inspiration for these values\footnote{Center for Transport Analytics website: http://www.cta.man.dtu.dk/}.

Table 4.1 shows information about all lines included in the case study network. The first column indicates the name of the line, followed by the mode of transport, the number of stops with transfer opportunities, the number of trips, an indication of whether the trips in the line are part of the timetabling design or not, the minimum headway in the initial timetable, and the maximum headway in the initial timetable.

For each trip $i \in T_i$ at each stop $s \in J_i \cup st_i$, minimum and maximum headways, $h_{is}^-$ and $h_{is}^+$, are calculated based on the scheduled headways between trip $i$ and its immediate precedent trip in the line, trip $i - 1$, as indicated in Table 4.2. Notice that in our case study the minimum scheduled headway is 4 minutes. In this case, the minimum and maximum headways allowed will be 3 and 5 minutes respectively.

The maximum dwell time added at each stop is 3 minutes (i.e., $w_{is}^+ = 3, i \in T, s \in J_i$), meaning that a maximum of 10 minutes of dwell time can be added in total to a trip (i.e., $w = 10$). The added dwell time is deducted from the buffer in the turnaround time at the end of the trip. The shifts allowed in each trip departure time were created based on the initial timetable for each bus line. Considering consecutively timetabled trips $(i - 1), i, (i + 1) \in T_i$ and with departure time from the first stop $d_{i-1,st_i}, d_{i,st_i}, d_{i+1,st_i}$ respectively, the lower and upper shift limits for trip $i$ are calculated with the expressions

$$d^-_{i,st_i} = d_{i,st_i} - \left[ \frac{d_{i,st_i} - d_{i-1,st_i} - 1}{2} \right], \quad d^+_{i,st_i} = d_{i,st_i} + \left[ \frac{d_{i+1,st_i} - d_{i,st_i}}{2} \right]$$

ensuring that trips can never overtake each other in the timetable.

As they are not part of the input, vehicle schedules that cover the initial timetable for
### Table 4.1: Lines in the case study network

<table>
<thead>
<tr>
<th>Line name</th>
<th>Mode</th>
<th>Stops</th>
<th>Trips</th>
<th>Timetabling</th>
<th>Min headway</th>
<th>Max headway</th>
</tr>
</thead>
<tbody>
<tr>
<td>150S</td>
<td>S-Bus</td>
<td>5</td>
<td>256</td>
<td>yes</td>
<td>4</td>
<td>21</td>
</tr>
<tr>
<td>200S</td>
<td>S-Bus</td>
<td>7</td>
<td>186</td>
<td>yes</td>
<td>6</td>
<td>23</td>
</tr>
<tr>
<td>250S</td>
<td>S-Bus</td>
<td>7</td>
<td>161</td>
<td>yes</td>
<td>8</td>
<td>24</td>
</tr>
<tr>
<td>300S</td>
<td>S-Bus</td>
<td>8</td>
<td>230</td>
<td>yes</td>
<td>5</td>
<td>23</td>
</tr>
<tr>
<td>350S</td>
<td>S-Bus</td>
<td>12</td>
<td>304</td>
<td>yes</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>400S</td>
<td>S-Bus</td>
<td>7</td>
<td>140</td>
<td>yes</td>
<td>7</td>
<td>21</td>
</tr>
<tr>
<td>500S</td>
<td>S-Bus</td>
<td>8</td>
<td>167</td>
<td>yes</td>
<td>7</td>
<td>31</td>
</tr>
<tr>
<td>600S</td>
<td>S-Bus</td>
<td>7</td>
<td>141</td>
<td>yes</td>
<td>4</td>
<td>34</td>
</tr>
<tr>
<td>A</td>
<td>S-Train</td>
<td>10</td>
<td>202</td>
<td>no</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>B</td>
<td>S-Train</td>
<td>8</td>
<td>202</td>
<td>no</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>Bx</td>
<td>S-Train</td>
<td>4</td>
<td>8</td>
<td>no</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>C</td>
<td>S-Train</td>
<td>8</td>
<td>205</td>
<td>no</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>E</td>
<td>S-Train</td>
<td>7</td>
<td>200</td>
<td>no</td>
<td>9</td>
<td>21</td>
</tr>
<tr>
<td>F</td>
<td>S-Train</td>
<td>3</td>
<td>374</td>
<td>no</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>H</td>
<td>S-Train</td>
<td>12</td>
<td>115</td>
<td>no</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>KB</td>
<td>Train</td>
<td>6</td>
<td>246</td>
<td>no</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>M1</td>
<td>Metro</td>
<td>7</td>
<td>487</td>
<td>no</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>M2</td>
<td>Metro</td>
<td>6</td>
<td>450</td>
<td>no</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>5C</td>
<td>Bus</td>
<td>8</td>
<td>542</td>
<td>no</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

### Table 4.2: Allowed headway variations based on scheduled headways

<table>
<thead>
<tr>
<th>Scheduled headway (m)</th>
<th>Minimum and maximum headway variation (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>= 4</td>
<td>+/- 1</td>
</tr>
<tr>
<td>≤ 12</td>
<td>+/- 2</td>
</tr>
<tr>
<td>≤ 20</td>
<td>+/- 3</td>
</tr>
<tr>
<td>≥ 21</td>
<td>+/- 4</td>
</tr>
</tbody>
</table>
Passenger service optimization through timetabling with free passenger route choice

the S-Bus trips are calculated using an MDVSP model. The solution consists of 205 vehicle schedules that cover the 1585 S-Bus trips. It uses constraints (4.3)-(4.5) of the mathematical model in Section 4.4.1. The initial timetable considers time dependent service times, but the mathematical model uses constant deadhead speeds along the day. Trips from different lines can be included in the same vehicle schedule, which is known as interlining, thus allowing deadheading between consecutive trips in a schedule. The maximum deadhead distance is 15 kilometers (i.e., \(u = 15\)), the minimum turnaround time is 12 minutes (i.e., \(q^- = 12\)), and the maximum turnaround time is 30 minutes (i.e., \(q^+ = 30\)).

4.6 Computational experiments

This section evaluates the performance of the MHeuPA through a set of computational experiments for the case study of the Greater Copenhagen Area described in Section 4.5. Results of the MHeuPA for different waiting cost functions (Section 4.4.3.1) are compared to the fixed passenger route choice model of Fonseca et al. (2018b), the IT-VSP MHeu. Note that results between this paper and Fonseca et al. (2018b) cannot be directly compared as the current case study represents a larger network, with new detailed passenger demand information that was not available yet during the Fonseca et al. (2018b) study; and secondly due to a different measure of passenger service, which in this paper is represented as WTT calculated by the passenger route choice model of Briem et al. (2017). Route_Fixed is an exact representation of the model of Fonseca et al. (2018b) in this new setting. Thus, the comparison between the MHeuPA and the Route_Fixed demonstrates the value of including free passenger route choice.

We evaluate our approach in the following three situations:

- **In comparison to an initial timetable representing the current timetable for our case study area** (Section 4.6.1). This case study is similar to the setting of Fonseca et al. (2018b), and therefore allows the most direct comparison between fixed and free passenger route choice. This section presents a detailed analysis of the results for the different components of weighted travel time, benefits specifically for transferring passengers, and the resulting vehicle schedules.

- **In case of a change in the public transport network** (Section 4.6.2). A change in the public transport network results in a timetabling situation where one would expect a change in passenger route choice. This situation is simulated by offsetting the timetables of one, or a set, of public transport lines in the network, such that headway constraints and time-dependent vehicle travel times are still respected, but transfers are likely offset.

- **In case of a change in passenger demand** (Section 4.6.3). A change in the passenger demand matrix could lead to a change the relevance of transfer opportunities:
making some transfer opportunities more important than others; for instance in case of a special event. This could also make it important to consider free passenger route choice. We assume that, whatever the change in demand, sufficient capacity is available, as measures to increase capacity on routes (e.g. longer vehicles, or higher frequencies) are not part of the timetabling decisions considered in this paper.

The algorithm is implemented in C++ and uses CPLEX version 12.6 to solve the mathematical program at each iteration. All experiments were conducted on HPC servers, using Intel Xeon E5-2660 v3 2.60GHz processors, and 8 computation cores. Each iteration uses CPLEX warm-start to start from the previous solution. Presented are average results over five runs with each setting, with a 3 hour computation time limit. Fixed parameters are the number of trips selected per iteration $\kappa = 350$, the maximum running time per iteration $\psi = 30$, and the realistic value for waiting costs $\phi = 2$. The values for the parameters $\kappa$ and $\psi$ are based on the computational results of Fonseca et al. (2018b) and taking into account that the current case study is larger both in terms of network and OD matrix, while the value for $\phi$ is based on the findings of Briem et al. (2017).

The solution quality is expressed in terms of weighted travel time (WTT) and its components: in-vehicle time (IVT), initial waiting time (IWT), and transfer time (TrT). We also compare the operating costs (OpC) across experiments. To compare the solutions obtained with the MHeuPA with the initial timetable and with the solutions obtained with the Route\_Fixed, we use percentage improvements to initial and percentage improvements to Route\_Fixed. For $x = \{\text{WTT, IVT, IWT, TrT, OpC}\}$ and $\bar{f}_x(S)$ being the $x$-type average cost of a solution $S$ over $n$ runs, we calculate the percentage improvement to the initial timetable as

$$\frac{\bar{f}_x(S_{\text{MHeuPA}}) - \bar{f}_x(S_{\text{Initial}})}{\bar{f}_x(S_{\text{Initial}})} \times 100\%$$

and the percentage improvement to the solutions obtained with the Route\_Fixed as

$$\frac{\bar{f}_x(S_{\text{MHeuPA}}) - \bar{f}_x(S_{\text{Route\_Fixed}})}{\bar{f}_x(S_{\text{Route\_Fixed}})} \times 100\%$$

since there is only one solution for the initial timetable but $n$ solutions for the Route\_Fixed (one for each run). A negative percentage for $x$ corresponds to a reduction of costs in the MHeuPA solution in comparison to the initial timetable or to the Route\_Fixed solutions. Since we use the budget version of the IT-VSP MHeu in all experiments, we keep the operating costs under a budget. We consider as budget the operating costs obtained by solving the MDVSP for the initial timetable.
Table 4.3: WTT results for the base scenario for 1) the full ODt matrix and 2) for the transferring passengers only

<table>
<thead>
<tr>
<th>Solution</th>
<th>Full ODt Matrix</th>
<th>Zoom on transferring passengers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WTT (DKK)</td>
<td>Improv. to Base (%)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improv. to WTT (DKK)</td>
</tr>
<tr>
<td>initial timetable</td>
<td>14,952,021</td>
<td>-</td>
</tr>
<tr>
<td>RouteFixed</td>
<td>14,844,516</td>
<td>-0.72</td>
</tr>
<tr>
<td>Realistic</td>
<td>14,832,916</td>
<td>-0.80</td>
</tr>
<tr>
<td>NoCosts</td>
<td>14,805,563</td>
<td>-0.98</td>
</tr>
<tr>
<td>LinAsc</td>
<td>14,810,788</td>
<td>-0.94</td>
</tr>
<tr>
<td>Random</td>
<td>14,811,079</td>
<td>-0.94</td>
</tr>
<tr>
<td>RandomLinAsc</td>
<td>14,811,457</td>
<td>-0.94</td>
</tr>
</tbody>
</table>

4.6.1 Results for the base scenario

In this section, we present the results for the base scenario: initial timetable and base ODt matrix. We present results in terms of WTT and different WTT components, operating costs, timetable modifications and characteristics of vehicle schedules. Additionally we show a convergence analysis for the different waiting cost functions considered and histograms of WTT variations.

Table 4.3 shows the WTT results for the different waiting cost functions in absolute WTT values, percentage improvement to initial timetable, and percentage improvement to RouteFixed solution. We present results both for the full ODt matrix and for a version of the ODt matrix that considers only passengers that, given the public transport network lines, will have to transfer at least one time.

The results in Table 4.3 support the hypothesis that inclusion of free route choice leads to timetables with higher passenger service, and that using alternative cost models, to find potentially beneficial transfer locations, also enable finding timetables with higher passenger service. Indeed, all MHeuPA solutions improve passenger service in comparison to RouteFixed solutions by 0.08% to 0.26% in the full ODt matrix and by 0.13% to 0.39% in the zoom on transferring passengers. The NoCosts is the best performing waiting cost function, both for the full ODt matrix and for transferring passengers only, while all alternative cost functions improve on input from the Realistic assignment model. All approaches improve passenger service in terms of WTT in relation to the initial timetable. The improvement in passenger service stems mainly from improved WTT for passengers with a transfer in their path, which is observed when comparing the results for the full ODt matrix with the results for transferring passengers only.

Figure 4.4 shows the convergence of WTT for the three best CostF functions LinAsc, NoCosts, and RandomLinAsc. The horizontal axis shows the algorithm total computational time in minutes and the vertical axis shows the percentage reduction in WTT in comparison to the initial timetable. The figure shows that the LinAsc waiting cost function has a steeper initial decline in WTT, while NoCosts finds the overall minimum WTT.
4.6 Computational experiments

Figure 4.4: WTT convergence of the LinAsc, NoCosts, and RandomLinAsc solutions over time

Table 4.4: Passenger WTT improvements for the base scenario

<table>
<thead>
<tr>
<th>Solution</th>
<th>Avg. improvement (DKK)</th>
<th>Pax better off (%)</th>
<th>Pax worse off (%)</th>
<th>Perc of pax with WTT reduction &gt;10 DKK (%)</th>
<th>Perc of pax with WTT increase &gt;10 DKK (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route_Fixed</td>
<td>-1.12</td>
<td>34.37</td>
<td>28.52</td>
<td>10.23</td>
<td>6.84</td>
</tr>
<tr>
<td>Realistic</td>
<td>-1.21</td>
<td>35.41</td>
<td>29.64</td>
<td>12.43</td>
<td>8.17</td>
</tr>
<tr>
<td>NoCosts</td>
<td>-1.45</td>
<td>35.20</td>
<td>31.19</td>
<td>13.22</td>
<td>9.64</td>
</tr>
<tr>
<td>LinAsc</td>
<td>-1.40</td>
<td>35.72</td>
<td>30.79</td>
<td>13.40</td>
<td>9.28</td>
</tr>
<tr>
<td>Random</td>
<td>-1.39</td>
<td>35.86</td>
<td>30.94</td>
<td>13.54</td>
<td>9.40</td>
</tr>
<tr>
<td>RandomLinAsc</td>
<td>-1.39</td>
<td>36.03</td>
<td>30.94</td>
<td>13.74</td>
<td>9.59</td>
</tr>
</tbody>
</table>

from around 120 minutes of computational time. This indicates that the LinAsc waiting cost function allows to find good timetables fast, but in the long run it is better to use the NoCosts waiting cost function. For all waiting cost functions, the improvements in WTT decline after the first 1.5 hours of computational time.

Table 4.4 shows for all waiting cost functions the average improvement in WTT expressed in DKK, the percentages of passengers better and worse off, the percentage of passengers better off by more than 10 DKK of WTT, and the percentage of passengers worse off by more than 10 DKK of WTT. The value of 10 DKK of WTT is equivalent to 6 minutes of in-vehicle time or 2 minutes of excess transfer time or initial waiting time.

Table 4.4 shows that the NoCosts waiting cost function achieves the best average improvement in WTT per passenger, with a value of -1.45 DKK, which follows from the results in Table 4.3. The percentage of passengers better and worse off is similar across waiting cost functions, with more passengers being better off than worse off when compared with the initial timetable. Specifically for differences in WTT larger than ±10%, the percentages of passengers better off are higher than the percentages of passengers worse off, as evidenced by the last two columns in Table 4.4. Although the NoCosts waiting cost function achieves the best average improvement, it is not the one that achieves the
Passenger service optimization through timetabling with free passenger route choice

Figure 4.5: Histogram of variation in WTT for the Route_Fixed solution and for the best solution obtained with our algorithm

highest percentage of passengers better off, with the RandomLinAsc surpassing the 36% mark. If the objective is to maximize the percentage of passengers better off by a certain threshold or to minimize the percentage of passengers worse off it might be preferable to use other waiting cost functions than the NoCosts waiting cost function for this instance.

For better understanding how the MHeuPA solutions improve the WTT for passengers, Figures 4.5 and 4.6 show histograms of changes in WTT in the best solutions obtained by the Route_Fixed and by the MHeuPA. The horizontal axis shows the change in WTT experienced by passengers and the vertical axis shows the absolute number of passengers that experience changes in each interval. The histogram is divided in two figures due to the difference in magnitude of the number of passengers.

Figures 4.5 and 4.6 show that in the MHeuPA solution more passengers experience high decreases in WTT, especially in the interval \([-70, -20]\). From Figure 4.6 it is clearly observed that the amount of passengers on the left hand side of the histogram (passengers better off) is larger than the one on the right hand side (passengers worse off), which is linked to the results in Table 4.4. However, Figure 4.5 shows that in the Route_Fixed solution more passengers experience smaller changes in WTT than in the MHeuPA solution, between -10 DKK and 10 DKK. Additionally, Figures 4.5 and 4.6 show that the Route_Fixed solution has less passengers worse off than the MHeuPA solution, which is explained by there being in general less changes in the timetable. The timetables of
Figure 4.6: Histogram of variation in WTT for the Route Fixed solution and for the best solution obtained with our algorithm.
Table 4.5: WTT details for the base scenario

<table>
<thead>
<tr>
<th>Solution</th>
<th>IVT (DKK)</th>
<th>IWT (DKK)</th>
<th>TrT (DKK)</th>
<th>Improv to Base (%)</th>
<th>Improv to Base (%)</th>
<th>Improv to Base (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial timetable</td>
<td>7,399,710</td>
<td>4,043,178</td>
<td>3,509,133</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Route_Fixed</td>
<td>7,390,300</td>
<td>4,036,102</td>
<td>3,418,115</td>
<td>-0.13</td>
<td>-0.18</td>
<td>-2.59</td>
</tr>
<tr>
<td>Realistic</td>
<td>7,389,179</td>
<td>4,041,434</td>
<td>3,402,302</td>
<td>-0.14</td>
<td>-0.04</td>
<td>-3.04</td>
</tr>
<tr>
<td>NoCosts</td>
<td>7,374,506</td>
<td>4,026,984</td>
<td>3,404,074</td>
<td>-0.34</td>
<td>-0.40</td>
<td>-2.99</td>
</tr>
<tr>
<td>LinAsc</td>
<td>7,378,596</td>
<td>4,034,440</td>
<td>3,397,752</td>
<td>-0.29</td>
<td>-0.22</td>
<td>-3.17</td>
</tr>
<tr>
<td>Random</td>
<td>7,378,806</td>
<td>4,033,539</td>
<td>3,398,735</td>
<td>-0.28</td>
<td>-0.24</td>
<td>-3.15</td>
</tr>
<tr>
<td>RandomLinAsc</td>
<td>7,379,542</td>
<td>4,033,954</td>
<td>3,397,962</td>
<td>-0.27</td>
<td>-0.23</td>
<td>-3.17</td>
</tr>
</tbody>
</table>

The MHeuPA represent a different trade-off between passenger groups, and although not a strict improvement for all passengers; the disbenefits for some passengers are offset by the benefits for a larger group of other passengers.

Table 4.5 shows the results in terms of each of the components of WTT, for the same set of experiments as in Tables 4.3 and 4.4. The table contains the absolute values in DKK of in-vehicle time (IVT), initial waiting time (IWT), and transfer time (TrT), along with their percentage improvements in relation to the initial timetable.

The results in Table 4.5 show that all components of WTT are improved in relation to the initial timetable, as evidenced by the negative percentages. Most of the improvement in WTT comes from the improvement in transfer times, with decreases ranging from -2.59% to -3.17% compared to decreases of -0.13% to -0.34% in in-vehicle time and of -0.04% to -0.40% in initial waiting time. This is expected, since transfer time is the WTT component that is specifically considered in the IT-VSP objective function. Among the MHeuPA results, the NoCosts waiting cost function obtains the smaller reduction in TrT, with a value of -2.99%, but obtains higher reductions in IVT and IWT, respectively -0.34% and -0.40%. Improvements in IVT show that passenger routes actually change in comparison to the initial timetable, because trips can only have additional dwell time and not less dwell time than in the initial timetable. This indicates that passengers are able to find better routes to travel from origin to destination, spending less time in-vehicle. Improvements in IWT are incidental since the IT-VSP objective does not include the effect of timetable modifications on IWT. Trips are shifted to cater for transfer synchronization, but in the process also IWT could be reduced.

Figure 4.7 visualizes the excess transfer time (transfer time minus the minimum required transfer time for a feasible transfer) experienced by passengers in the initial timetable and in the best MHeuPA solution. The horizontal axis shows the excess transfer time in minutes and the vertical axis shows the percentage of transferring passengers that experiences that value of excess transfer time.

Figure 4.7 shows that in the MHeuPA timetable there are significantly more passengers experiencing perfectly synchronized transfers, with 0 minutes of excess transfer time. A
### 4.6 Computational experiments

![Comparison of excess transfer time in the initial timetable and in the best MHeuPA solution timetable](image)

**Figure 4.7:** Comparison of excess transfer time in the initial timetable and in the best MHeuPA solution timetable

**Table 4.6:** Operating costs, timetable and vehicle schedules for the 8 line case and for the base OD matrix

<table>
<thead>
<tr>
<th>Solution</th>
<th>OpC (DKK)</th>
<th>OpC improv to Base (%)</th>
<th>Trips with shifts only (%)</th>
<th>Trips with stretches only (%)</th>
<th>Trips with shifts and stretches (%)</th>
<th>Avg added dwell (m)</th>
<th>Avg added shift (m)</th>
<th>Number Schedules</th>
<th>Avg schedule duration (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial timetable</td>
<td>323,744</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>205</td>
<td>784</td>
</tr>
<tr>
<td>Route_Fixed</td>
<td>317,417</td>
<td>-1.95</td>
<td>48.3</td>
<td>3.8</td>
<td>15.4</td>
<td>2.3</td>
<td>1.7</td>
<td>200</td>
<td>802</td>
</tr>
<tr>
<td>Realistic</td>
<td>317,608</td>
<td>-1.90</td>
<td>52.8</td>
<td>3.3</td>
<td>14.2</td>
<td>2.4</td>
<td>2.0</td>
<td>201</td>
<td>800</td>
</tr>
<tr>
<td>NoCosts</td>
<td>318,124</td>
<td>-1.74</td>
<td>55.1</td>
<td>3.8</td>
<td>14.0</td>
<td>2.2</td>
<td>1.9</td>
<td>201</td>
<td>798</td>
</tr>
<tr>
<td>LinAsc</td>
<td>318,437</td>
<td>-1.64</td>
<td>56.6</td>
<td>2.6</td>
<td>13.6</td>
<td>2.3</td>
<td>2.0</td>
<td>201</td>
<td>798</td>
</tr>
<tr>
<td>Random</td>
<td>316,754</td>
<td>-2.16</td>
<td>54.0</td>
<td>4.0</td>
<td>17.1</td>
<td>2.3</td>
<td>2.0</td>
<td>200</td>
<td>802</td>
</tr>
<tr>
<td>RandomLinAsc</td>
<td>316,649</td>
<td>-2.19</td>
<td>54.4</td>
<td>3.8</td>
<td>16.6</td>
<td>2.4</td>
<td>2.1</td>
<td>200</td>
<td>803</td>
</tr>
</tbody>
</table>

The results in Table 4.6 show that all waiting cost functions use less operating costs than the budget of the initial timetable, but also considerably decrease them, with percentages between 1.64% and 2.19%. The solutions obtained with the MHeuPA shift 52.8% to 56.6% of the trips, add stretches to 2.6% to 4.0% of the trips, and add both shifts and stretches to 13.6% to 17.1% of the trips. The values of added dwell time and added shifts are averages over the total number of trips with added stretches and added shifts respectively. On total of 22% of transferring passengers experience perfectly synchronized transfers in the MHeuPA solution, while in the initial timetable this value is 18%. For all other excess transfer time values, the initial timetable has more passengers experiencing each value. Furthermore, the average excess transfer time decrease from 2.29 m to 2.06 m in the MHeuPA solution, for more than 100,000 transferring passengers.
average, just 2 minutes of dwell time are added to modified trips. Regarding the vehicle schedules, the Route_Fixed and MHeuPA solutions use 200 to 201 schedules to cover all trips, while the base solution uses 205. Furthermore, schedules are on average longer in the Route_Fixed and MHeuPA solutions with values ranging between 798 and 803 minutes, compared to the 784 minutes in the initial timetable. This means that the Route_Fixed and MHeuPA solutions use resources more efficiently, with vehicles covering on average more trips.

The overall improvement in WTT in comparison to the initial timetable is approximately 1%, of which 0.25% is due to the inclusion of free passenger route choice. The 0.25% is equivalent to a daily reduction of approximately 40,000 DKK when expressed as value of time. Due to the budget constraints, these savings come at no additional operating costs, and in fact allow a reduction of operational costs, as demonstrated in Table 4.6.

4.6.2 Designing timetables - changes in the public transport network

In this section, we analyze solutions obtained when we start from a randomized timetable. Note that given a line plan and frequency, we can always create an initial timetable. It may however be of bad quality. This section demonstrates that also provided such a ”bad” starting point, the MHeuPA can be used for constructing a timetable. Thus in principle it is suitable for designing new timetables. Results for the best performing waiting cost functions from the previous section (NoCosts, LinAsc, and RandomLinAsc) are compared to solutions obtained with the Route_Fixed (fixed passenger route choice). We test four different scenarios:

1. offsetting the timetable for line 350S only, which is the line that transports the largest volume of passengers in the network;
2. offsetting the timetables for lines 250S, 300S, and 400S, which are the lines with largest volumes of passenger transfers;
3. offsetting the timetables for lines 350S, 500S, and 600S, which are the lines with largest volumes of passenger transfers involving a bus trip;
4. offsetting the full S-Bus network timetable.

When designing the timetable for one line, we start from its trips in the initial timetable. We then shift all trips of that line by a random amount of time between 0 and double the minimum periodicity of that line. The minimum periodicity of a line in the initial timetable is between 13% and 41% of its maximum periodicity, so in most cases the timetable modifications will allow attaining a timetable close to the initial timetable. Each direction in the offset line is shifted by a different random amount of time. Timetable
modifications by shifts and stretches are allowed for all S-Bus lines, in all scenarios. Offsetting the timetable may also change the vehicle schedules obtained when solving the MDVSP. In order to enable a fair comparison between results in this section and in the previous section, we use as budget for the operating costs the same budget used before, i.e. the operating costs obtained by solving the MDVSP for the initial timetable. Table 4.7 shows results for all four scenarios in terms of absolute WTT and DKK savings in comparison to the Route\textsubscript{Fixed} solutions. Each scenario is associated with an $\alpha$ value, which is the percentage increase in WTT of the offset initial timetable in comparison to the initial timetable. The results in Table 4.7 show that all MHeuPA solutions have a lower WTT than the solutions obtained with the Route\textsubscript{Fixed}, similarly to what is observed in Section 4.6.1. Again, the NoCosts waiting cost function is the one that shows the largest decreases in WTT. By comparing Tables 4.3 and 4.7 we see that the solutions in Table 4.7 also have a lower WTT than the initial timetable of Section 4.6.1, despite starting from a worse timetable (evidenced by all $\alpha$ values being positive). Solutions obtained in the previous sections with the same waiting cost functions are better in terms of WTT by 0.05% to 0.56% than solutions obtained in this section, since they start from a timetable with more transfer synchronization and therefore lower WTT.

### 4.6.3 Changes in passenger demand

In this section, we test the MHeuPA for different ODt matrices than the base matrix. We test two different scenarios:

1. a random variation in the base ODt matrix (Random $\pm 10\%$);
2. an event simulation ODt matrix (Event Simulation).

The random variation scenario was generated by varying OD hourly demand in the base ODt matrix randomly by a value between -10% and 10%. As for the Special Event Simulation scenario, we selected three stations in the city center and simulated a two hour event happening between 6p.m. and 8p.m. Consequently, we increase by 50% all OD pairs in the base ODt towards these three stations with departure time during the

---

**Table 4.7: WTT results for the four scenarios of designing timetables**

<table>
<thead>
<tr>
<th>Offset lines $\alpha$</th>
<th>Solution</th>
<th>WTT (DKK)</th>
<th>Savings ict Route\textsubscript{Fixed} (DKK)</th>
<th>WTT (DKK)</th>
<th>Savings ict Route\textsubscript{Fixed} (DKK)</th>
<th>WTT (DKK)</th>
<th>Savings ict Route\textsubscript{Fixed} (DKK)</th>
<th>WTT (DKK)</th>
<th>Savings ict Route\textsubscript{Fixed} (DKK)</th>
</tr>
</thead>
<tbody>
<tr>
<td>350S 0.07</td>
<td>Route\textsubscript{Fixed}</td>
<td>14,858,155</td>
<td>-</td>
<td>14,862,350</td>
<td>-</td>
<td>14,886,838</td>
<td>-</td>
<td>14,928,154</td>
<td>-</td>
</tr>
<tr>
<td>250S, 300S, 400S 0.16</td>
<td>NoCosts</td>
<td>14,813,879</td>
<td>-44,276</td>
<td>14,817,685</td>
<td>-44,665</td>
<td>14,840,451</td>
<td>-46,388</td>
<td>14,879,730</td>
<td>-48,424</td>
</tr>
<tr>
<td>350S, 500S, 600S 0.13</td>
<td>Linker</td>
<td>14,818,344</td>
<td>-39,811</td>
<td>14,832,089</td>
<td>-30,261</td>
<td>14,843,607</td>
<td>-43,231</td>
<td>14,882,466</td>
<td>-45,689</td>
</tr>
<tr>
<td>All S-Bus network 0.39</td>
<td>RandomLinAsc</td>
<td>14,822,558</td>
<td>-35,596</td>
<td>14,832,355</td>
<td>-29,995</td>
<td>14,840,764</td>
<td>-46,074</td>
<td>14,894,508</td>
<td>-33,646</td>
</tr>
</tbody>
</table>
two hours prior to the event. We also increase by 50% all OD pairs in the base ODt originating from these three stations in the two hours after the event.

Table 4.8 shows the results for the two scenarios, in absolute WTT and percentage improvement in relation to the Route_Fixed solutions. Similarly to Section 4.6.2, each scenario is associated with an \( \alpha \) value, indicating the percentage increase of the initial solution in comparison to the initial timetable for the base ODt matrix.

From Table 4.3, the initial timetable has a WTT of 14,952,021 DKK. We observe that, despite the changes in passenger demand, the MHeuPA is able to obtain solutions that have lower WTT than the initial timetable. Furthermore, the integration with a PTTA model proved to be beneficial, evidenced by the solutions with lower WTT obtained with the MHeuPA solutions in comparison to the Route_Fixed solutions. The NoCosts waiting cost function once again outperforms the other cost functions, with a reduction in WTT of -0.29% in the random ODt scenario and of -0.31% in the event simulation scenario, in comparison to the Route_Fixed solutions. Furthermore, the NoCosts is also the best performing waiting cost function when comparing with the initial timetable, with WTT reductions of -0.92% in the random ODt scenario and of -0.61% in the special event simulation scenario.

We acknowledge that, in general, in case of a large event it is important to evaluate if there is capacity to transport the higher volumes of passengers. Since our approach does not take into account capacity constraints, it is out of the scope of the current work to consider the analysis of capacity restrictions. The purpose of the above described cases is to demonstrate new timetables can also be found in case of a change in demand scenarios. Thus, the MHeuPA is suitable to use in a wide range of situations: improving on the current timetable, dealing with a change in the network, dealing with a change in passenger demand.

### 4.7 Conclusions and future research

This paper addresses the problem of maximizing passenger service through timetabling under the assumptions of free passenger route choice within a fixed budget for operating
costs at a tactical level. Free route choice implies that passengers follow their individually preferred path, rather than one that optimizes a social optimum, and that passengers with the same origin, destination, and departure time may have different preferences. The latter ensures that in case two equivalent routes exist, passengers are assumed to use both.

The proposed matheuristic for the IPAT-VSP combines two state-of-the-art models: the integrated timetabling and vehicle scheduling model of Fonseca et al. (2018b) with the passenger route choice model of Briem et al. (2017). Provided an initial timetable and an ODt matrix describing passenger demand over time, the objective of the MHeuPA is to maximize passenger service, expressed as weighted travel time, through modifications of the timetable. These modifications consist of changes in the starting time of trips (shifts), and addition of dwell time (stretches) at transfer stops, in comparison to the initial timetable within a set of headway constraints and a budget on operating costs. Operating costs are defined by the minimum cost vehicle schedules for a timetable, which problem is simultaneously solved during the timetabling procedure.

A realistic case study focused on timetabling bus lines in the context of the multi-modal network of the Greater Copenhagen area illustrates that (i) including free passenger route choice leads to timetables with higher passenger service than assuming fixed passenger route choice such as in Fonseca et al. (2018b), (ii) that the indication of potentially interesting transfers for passengers results in timetables with a higher passenger service than providing the timetabling model information on the precise passenger route choice on the current timetable, and (iii) that benefits of including free passenger route choice can be found in comparison to the current timetable of our case study area, in case of a change in the network, and in case of a change in passenger demand. The latter also suggests that the proposed MHeuPA approach could be used to design new timetables in case of changes in the network, e.g. due to planned maintenance, or in case of an expected change in the demand matrix, e.g. due to special events.

Although the higher passenger service in our case study results from a trade-off between passenger groups, the increase in service results foremost from a sizable decrease in WTT for a large group of passengers that offsets the increase in WTT for others. Overall improvement in WTT in comparison to the initial timetable is approximately 1%, of which 0.25% is due to the inclusion of free passenger route choice. The 0.25% is equivalent to a daily reduction of approximately 40,000 DKK when expressed as value of time. Due to the budget constraints, these savings come at no additional operating costs.

In summary, this paper contributes to the field of timetabling and public transport planning by studying integrated maximal passenger service timetabling and vehicle scheduling in the context of a realistic free passenger route choice model representing free route choice of passengers; demonstrating that the inclusion of free passenger route choice leads to timetables with higher passenger service and that the indication of potential important transfers for passengers is more important than providing a timetabling model with accurate information on the passenger route choice in a current, initial timetable.
Future research may focus on a further integration of passenger route choice decisions into the timetabling and vehicle scheduling model; or on extending the timetabling procedure to include decisions on stops per line and frequency, which have a major influence on passenger service but are currently generally fixed in the previous planning stage of line planning and network design. Moreover, future research could focus on finding exact lower bounds for the maximal passenger service timetabling problem.

### 4.8 Appendix 1 - IT-VSP mathematical model

**Sets**

- \( S \): Set of all stops
- \( L \): Set of all directed lines
- \( T = \{1, \ldots, n\} \): Set of all timetabled trips
- \( T_l \subseteq T \): Subset of all trips in the directed line \( l \in L \)
- \( T^1 \subseteq T \): Set of all trips which are the first in their directed line
- \( S_i \subseteq S \): Set of all stops visited by trip \( i \in T \), i.e., \( J_i = S_i \setminus \{st_i, et_i\} \)
- \( R \): Set of all transfer opportunities, each defined by a triplet \((i, l, s)\): passengers disembarking trip \( i \in T \) at stop \( s \in J_i \cup \{et_i\} \) with the intent of embarking a trip \( j \in T_l \) of line \( l \in L \) such that \( l \neq l_i \) and \( s \in J_j \cup \{st_j\} \)
- \( K \): Set of all depots
- \( I \): Set of all compatible trips, \( I = \{(i, j) | i, j \in T : i \neq j, Dist(et_i, st_j) \leq u, a_{i,et_i} + q^- + b_{ij} \leq d_{j,st_j}^-, a_{i,et_i} + q^+ + b_{ij} \geq d_{j,st_j}^+\} \)
- \( V_k \): Set of nodes, which contains a node for each trip \( i \in T \), as well as for depot \( k \in K \) which is denoted \( n + k \), thus \( V_k = T \cup \{n + k\} \)
- \( A_k \): Set of arcs, including deadhead trips, pull-out trips, and pull-in trips, thus \( A_k = I \cup \{(n + k) \times T \} \cup (T \times \{n + k\}) \)
- \( G_k = (V_k, A_k) \): Graph associated with depot \( k \in K \)
- \( Q^D \): Set of all deadhead triplets \( Q^D = \{(i, j, k) : k \in K, (i, j) \in I\} \)
- \( Q^O \): Set of all pull-out triplets \( Q^O = \{(n + k, j, k) : k \in K, j \in T\} \)
- \( Q^H \): Set of all pull-in triplets \( Q^H = \{(i, n + k, k) : i \in T, k \in K\} \)
- \( Q \): Set of all compatible triplets \((i, j, k)\), representing a vehicle from depot \( k \in K \) covering the pair of trips \((i, j) \in A_k\). \( Q = Q^D \cup Q^O \cup Q^H \)
- \( T(Q) \): Set of all pairs of trips \( i, j \in T \) for which a triplet involving \( i \) and \( j \) exists, \( T(Q) = \{(i, j) | i, j \in T : \exists (i, j, k) \in Q\} \).

**Parameters**

- \( l_i \): Directed line of trip \( i \in T \)
- \( t_i \): Total travel time of trip \( i \in T \) in the initial timetable
- \( st_i \in S_i \): Start terminal of trip \( i \in T \)
- \( et_i \in S_i \): End terminal of trip \( i \in T \)
### 4.8 Appendix 1 - IT-VSP mathematical model

- $h_{is}^{-}, h_{is}^{+}$: Minimum and maximum headways, respectively, in relation to the timetabled headways, for each trip $i \in T$ at each stop $s \in J_i \cup \{st_i\}$
- $d_{i,st_i}^{-}, d_{i,st_i}^{+}$: Minimum and maximum departure shift from the first station for trip $i \in T$, defined in relation to its departure time in the initial timetable
- $w_{is}^{-}, w_{is}^{+}$: Dwell time in the initial timetable of a trip $i \in T$ at stop $s \in J_i$
- $w_s$: Maximum allowed dwell time of a trip $i \in T$ at stop $s \in J_i$
- $\Lambda_{is}$: Upper bound on the total added dwell time to all stops of any trip
- $a_{is}^{-}, a_{is}^{+}$: Earliest and latest arrival times of trip $i \in T$ at stop $s \in J_i \cup \{et_i\}$, determined by the possible timetable modifications
- $d_{is}^{-}, d_{is}^{+}$: Earliest and latest departure times of trip $i \in T$ from stop $s \in J_i \cup \{st_i\}$, determined by the possible timetable modifications
- $f_r$: Number of passengers requesting transfer $r \in R$
- $c_{ij}$: Operating cost associated with servicing triplet $(i,j,k) \in Q$. The cost $c_{ij}^k$ of triplet $(i,j,k) \in Q$ is equal to the deadhead time $b_{ij}$ multiplied by a driving cost per time unit; if $(i,j,k) \in Q^D$, $c_{ij}^k$ also includes a fixed cost for creating a new schedule, corresponding to the fixed cost for using a vehicle.
- $c_{ij}^{DW}$: Operating cost per minute of extra dwell time
- $c_{ij}^{OB}$: Cost per minute of extra dwell time per on board passenger
- $c_{ij}^{TR}$: Cost per minute of excess transfer time at transfers per passenger

### Decision variables

- $x_{ijk} \in \{0,1\}$: 1 if and only if a vehicle from depot $k$ travels from node $i$ directly to node $j$, 0 otherwise
- $\tau_{is}^{-}, \tau_{is}^{+} \in \mathbb{Z}_0^{+}$: Departure time of trip $i \in T$ from stop $s \in J_i \cup \{st_i\}$
- $\tau_{is}^{-}, \tau_{is}^{+} \in \mathbb{Z}_0^{+}$: Arrival time of trip $i \in T$ at stop $s \in J_i \cup \{et_i\}$
- $\gamma_r \in \mathbb{R}_0^{+}$: Excess transfer time for passengers using transfer location $r \in R$
- $\alpha_{ij,s} \in \{0,1\}$: 1 if and only if passengers of transfer location $r = (i,l,s) \in R$ embark trip $j \in T$, 0 otherwise
- $\delta_{is} \in \mathbb{Z}_0^{+}$: Minutes of dwell time added to trip $i \in T$ at stop $s \in J_i$

We define a directed line $l \in L$ as a sequence of stops $s \in S$ visited by a vehicle, the set of all trips $T = \bigcup_{l \in L} T_l$ and $T_{l'} \cap T_{l''} = \emptyset$ for all $l', l'' \in L, l' \neq l''$. Passengers are assumed to transfer to the earliest feasible trip $j \in T_i$. A transfer $r$ from trip $i$ to line $l$ at stop $s$, $r = (i,l,s) \in R$, is feasible when the minimum transfer time for transfer $r \in R$, $e_r$, is not greater than the difference between the departure time of trip $j \in T_i$ from stop $s$...
and the arrival time of trip $i$ at stop $s$. The turnaround time should generally be in the interval $[q^-, q^+]$. Buffer time added to the trip in the form of dwell time is subtracted from the minimum turnaround time $q^-$. Each vehicle used in a feasible solution covers a sequence of compatible trips and must return to the depot from which it departed. Two trips $i, j \in T$ are compatible if the following three conditions hold: (a) $\text{Dist}(et_i, st_j)$ is smaller than $u$; (b) The sum of $a_{i, et_i}, q^-$, and $b_{ij}$ is smaller or equal to $d_{j, st_j}^+$; and (c) the sum of $a_{i, et_i}, q^+$, and $b_{ij}$ is greater or equal to $d_{j, st_j}^-$. The $\alpha_{ij}$ variables indicating transfer opportunities are defined only for a set $W = \{(i, j, s) | i, j \in T, s \in S : r = (i, l, s) \in R, j \in T_i, i \neq j, a_{is}^- + \epsilon_r \leq d_{js}^-, a_{is}^+ + \epsilon_r + 1.5 h_l \geq d_{js}^+\}$, where $h_l$ is the largest frequency observed for line $l \in L$ throughout the day. This improves the tractability of the model by reducing the number of $\alpha_{ij}$ variables created, without imposing any practical constraints, since at least one transfer to a trip in $l \in L$ will be available given the timetable modifications.

The MILP formulation for the IT-VSP is:

$$\min \sum_{i \in T} \sum_{s \in J_i} c^{OB} a_{is} \delta_{is} + \sum_{r \in R} c^{TR} f_r \gamma_r$$  \hspace{1cm} (4.1)

s.t. \hspace{1cm} \sum_{(i,j,k) \in Q} c_{ijk} x_{ijk} + \sum_{i \in T} \sum_{s \in J_i} c^{DW} \delta_{is} \leq B  \hspace{1cm} (4.2)

$$\sum_{(i,j,k) \in Q} x_{ijk} = 1  \hspace{1cm} i \in T \hspace{1cm} (4.3)$$

$$\sum_{(i,j,k) \in Q} x_{ijk} - \sum_{(j,i,k) \in Q} x_{jik} = 0  \hspace{1cm} k \in K \hspace{1cm} j \in V_k \hspace{1cm} (4.4)$$

$$\sum_{(i,j,k) \in Q^O} x_{ijk} \leq v_k  \hspace{1cm} k \in K \hspace{1cm} (4.5)$$

$$d_{i, st_i}^- \leq \delta_{i, st_i}^d \leq d_{i, st_i}^+  \hspace{1cm} i \in T \hspace{1cm} (4.6)$$

$$0 \leq \tau_{is}^d - \tau_{is}^a - w_{is}^- \leq w_{is}^+  \hspace{1cm} i \in T \hspace{1cm} s \in J_i \hspace{1cm} (4.7)$$

$$\sum_{s \in J_i} \delta_{is} \leq w  \hspace{1cm} i \in T \hspace{1cm} (4.8)$$

$$\delta_{is} = \tau_{is}^d - \tau_{is}^a - w_{is}^-  \hspace{1cm} i \in T \hspace{1cm} s \in J_i \hspace{1cm} (4.9)$$

$$h_{is}^- \leq \tau_{is}^d - \tau_{i-1, s}^a \leq h_{is}^+  \hspace{1cm} l \in L \hspace{1cm} i \in T_l : i \notin T_l \hspace{1cm} s \in J_l \cup \{s_i\} \hspace{1cm} (4.10)$$

$$\tau_{i, et_i}^a + b_{ij} + q^- - \sum_{s \in J_i} \delta_{is} - M(1 - \sum_{(i,j,k) \in Q} x_{ijk}) \leq \tau_{j, st_j}^d  \hspace{1cm} (i,j) \in T(Q) \hspace{1cm} (4.11)$$

$$\sum_{k \in T_l : (i,k,s) \in W, k \leq j} \alpha_{iks} \geq \tau_{js}^d - \tau_{is}^a - \epsilon_r  \hspace{1cm} r \in R \hspace{1cm} (i,j,s) \in W \hspace{1cm} (4.12)$$

$$\tau_{js}^d - \tau_{is}^a - \epsilon_r \geq M(\alpha_{ij} - 1)  \hspace{1cm} r \in R \hspace{1cm} (i,j,s) \in W \hspace{1cm} (4.13)$$

$$\sum_{j \in T_l : (i,j,s) \in W} \alpha_{ij} = 1  \hspace{1cm} r = (i,l,s) \in R \hspace{1cm} (4.14)$$

$$\tau_{js}^d - \tau_{is}^a - \epsilon_r - M(1 - \alpha_{ij}) \leq \gamma_r  \hspace{1cm} r \in R \hspace{1cm} (i,j,s) \in W \hspace{1cm} (4.15)$$

$$x_{ijk} \in \{0, 1\}  \hspace{1cm} (i,j,k) \in Q \hspace{1cm} (4.16)$$

$$\tau_{is}^d \in \mathbb{Z}_+  \hspace{1cm} i \in T \hspace{1cm} s \in J_i \cup \{s_i\} \hspace{1cm} (4.17)$$
The objective function (4.1) minimizes a weighted sum of passengers’ costs. The first term refers to on board passenger costs incurred when adding dwell time to trips. The second term refers to costs associated with excess transfer times.

Constraints (4.2) impose an upper bound (budget) on the operating costs. The first term considers driving operating costs for deadhead, pull out, and pull in trips, and the second term considers operating costs associated with additional dwell times. Constraints (4.3) - (4.5) model classical MDVSP constraints: assignment constraints (4.3) guarantee coverage of each trip \( i \in T \) by including it in exactly one vehicle schedule; flow conservation constraints (4.4) on trip and depot nodes guarantee the continuity of the vehicle schedules created; and capacity constraints (4.5) limit the number of pull-out trips to the maximum number of vehicles available at each depot \( k \in K \).

Constraints (4.6) - (4.10) model timetable modifications: constraints (4.6) force lower and upper shift bounds on the departure time from the first stop of each trip; constraints (4.7) impose a maximum added dwell time at each stop of a trip; constraints (4.8) bound the total added dwell time to all intermediate stops of a trip; constraints (4.9) define the \( \delta_{is} \) variables to the added dwell time in the corresponding intermediate stop \( s \in J_i \) of trip \( i \in T \); and constraints (4.10) model the minimum and maximum headways between each trip \( i \in T \) and its precedent trip in the same directed line at each stop \( s \in J_i \cup \{st_i\} \).

Linking constraints (4.11) relate the vehicle scheduling and the timetable modification parts of the problem. These guarantee that if trips \( i \) and \( j \) are serviced consecutively by the same vehicle, then the vehicle has time to deadhead from \( et_i \) to \( st_j \) without violating the minimum turnaround time \( q^- \).

Linking constraints (4.12) and (4.13) relate the transfer variables \( \alpha_{ij}s \) and the departure and arrival times of trips: constraints (4.12) ensure that passengers arriving from trip \( i \) at stop \( s \) transfer to one of the trips \( j \), such that \((i, l(j), s) \in R\), if the arrival and departure times allow the transfer to take place; constraints (4.13) prevent variable \( \alpha_{ij}s \) from taking value 1 whenever passengers do not have enough time to transfer from trip \( i \) to trip \( j \) at stop \( s \), where \((i, l(j), s) \in R\). Constraints (4.14) impose that each transfer location is performed by transferring to exactly one trip \( j \in T_i \). Constraints (4.15) define the values of \( \gamma_r \) variables to the excess transfer times, determining this value for each transfer location based on the selected transfers. Constraints (4.16)-(4.21) define the range of all sets of decision variables.
Passenger service optimization through timetabling with free passenger route choice
Chapter 5

Aperiodic public transport timetabling with flexible line plans

with Evelien van der Hurka, Roberto Robertib, Gábor Marótic

aDTU Management - Department of Technology, Management and Economics, Technical University of Denmark, Kgs. Lyngby, Denmark
bDepartment of Supply Chain Analytics, School of Business and Economics, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
cNetherlands Railways (NS), Utrecht, The Netherlands

Publication Status: to be submitted

Abstract: Line planning and timetabling are intrinsically related problems in railway planning. Line planning consists of determining which lines are operated, i.e. the sequence of stations visited by each line, and the frequency at which each line is operated, in order to maximize some function of passenger service and operating costs. Timetabling consists of assigning departure and arrival times to all services of each line, at all stations visited, with the objective of maximizing passenger service. Furthermore, a feasible timetable ensures that safety headway constraints and station capacities are respected. The output of line planning is part of the input for timetabling, hence the line plans highly influence the quality of the timetable. While integration of the two problems can potentially yield timetables with increased passenger service, this is not a straight forward task. Additionally, the problem becomes even more complex if detailed passenger routing is to be integrated as well.

We consider a timetabling problem integrated with elements of line planning – the aperiodic public transport timetabling problem with flexible line plans (AT-LP). The
objective of the AT-LP is to minimize a weighted sum of passenger costs. Passenger costs include in-vehicle time, initial waiting time, transfer time, and transfer penalties. Input to the problem are the lines and suggested hourly frequencies from the line planning stage, an origin-destination-time matrix, and network characteristics. The output is an optimized timetable that minimizes passenger costs, a passenger assignment for that timetable, and an estimate of costs for operating the timetable. We allow changing the suggested frequencies for the lines, provided that they remain within defined upper and lower bounds. We also allow to change the stopping pattern of lines by skipping stops, imposing a maximum number of stops canceled per trip and imposing that consecutive services in a line do not cancel the same stop. Timetable safety headways must be respected and we impose a budget on operating costs. We use shortest path computations to assess the quality of timetables in terms of passenger service and compare them to a lower bound on passenger costs. We propose a heuristic solution approach to solve the problem. Results for a real life case study for a subset of the IC network in The Netherlands indicate that including line planning elements in our methodology improves the quality of timetables in terms of passenger service, using the additional budget on operating costs.

**Keywords:** Public Transport Optimization · Train Aperiodic Timetabling · Line Planning · Heuristics

## 5.1 Introduction

Public transport planning consists of a set of problems that are traditionally solved in sequence (Schöbel, 2017). Long term decisions include infrastructure construction and line planning problems. Strategic and operational decisions include timetabling, rolling stock (or vehicle scheduling), and crew scheduling, among others. Line planning consists of deciding the composition of lines and their frequency. Given the line plan, timetabling consist of assigning times to events such as arrivals and departures from stations. The goal of timetabling is to provide high quality passenger service, while allowing low operating costs. Timetables can be periodic (if a certain pattern is repeated after a certain period of time) or aperiodic (if services are scheduled freely for the whole planning horizon). Solving these problems in sequence yields sub-optimal solutions and recent research is aimed at integrating different planning problems, in an attempt to find better solutions.

We address the Aperiodic Public Transport Timetabling Problem with Flexible Line Plans (AT-LP), which integrates line planning ideas in an aperiodic timetabling problem. Input from the line planning phase consists of a given set of lines, each line to be operated at a given hourly frequency. We define a *service* as a timetabled realization of a line, with departure and arrival times at stations included in the line. We assume known and fixed passenger demands in an input ODt matrix, with a number of passengers associated with each entry of the matrix. The objective of the AT-LP is to minimize passenger
inconvenience, based on initial waiting time, in-vehicle time, transfer time, and number of transfers.

The AT-LP includes timetabling decisions and line planning decisions. Timetabling decisions consist of determining arrival and departure times from stations. Line planning decisions consist of deciding the number of services for each line, thus modifying the initial suggested frequencies within defined bounds, and deciding stopping patterns for each service, thus allowing skipping stops at stations. The AT-LP imposes an upper bound on the number of stops canceled per service and ensures that two consecutive services in the same line do not cancel the same stop. By defining an upper bound on the headway between two consecutive services of the same line, the AT-LP ensures regularity and prevents solutions with high imbalances of services per direction. Safety headways at stations and track sections ensure the feasibility of solutions found, guaranteeing that all solutions do not contain station or track section conflicts.

Furthermore, the inclusion of passengers constitutes by itself a challenge, since two conflicting objectives collide. On one hand, passengers want to travel as fast and cheap as possible. On the other hand, public transport operators want to minimize operating costs. In order to keep the estimated operating costs balanced, and prevent too costly solutions due to the addition of services, a budget on the estimated costs of operating the timetable is imposed - in our case, measured as a cost related to the train utilization in train in-service minutes.

The considered line planning modifications impact the quality of the timetable both in terms of passenger service and operating costs. Changing the frequency of a line has a direct consequence on passenger waiting times and transfers. Skipping stops in a service decreases the travel time for on-board passengers that use that service, but increases waiting time for passengers wanting to transfer/board/alight at those stations. Skipping stops also reduces the usage of rolling stock, so it potentially decreases operating costs. The AT-LP outputs a timetable to operate the lines in $L$ with possibly modified frequencies and stopping patterns, along with expected passengers flows and a measure of operating costs expressed in train in-service minutes. Passenger service quality is measured in terms of the defined passenger costs (with different weights for each category), and by routing passengers in the ODt using shortest path computations.

We propose an iterative heuristic approach to solve the AT-LP. The heuristic starts by determining a feasible timetable according to the suggested line plans and frequencies. Once an initial timetable is found, we apply a set of operators that modify the timetable and assess the results of those modifications. The timetable operator considered is a shift operator, that shifts trips forward or backward in time. The line planning operators considered are the addition and removal of services, and the skipping or addition of stops at stations. In order to escape local minima, the heuristic accepts worse solutions, and multiple methods to restart the search are studied. The quality of the heuristic solutions is compared with a valid lower bound on the passenger inconvenience, computed using a lower bound method.
We test our solution approach in a case study composed by a subset of the IC lines in The Netherlands railway network. We show that, allowing a limited additional budget on operating costs, the inclusion of line planning elements helps find timetables that reduce passenger costs in comparison to using our methodology allowing timetable modifications only. We analyze different versions and combinations of parameters in the solution approach, and we analyze the solutions in terms of convergence and gap to the calculated lower bound on passenger costs.

Most of the authors that integrate line planning and timetabling do it for the periodic case. We consider an aperiodic timetabling problem that minimizes passenger costs and is allowed to modify some of the elements of line planning, namely stopping patterns and frequencies. It does so while guaranteeing that a certain budget on operating costs is respected, and it includes passenger routing. We propose an iterative heuristic to solve the AT-LP for real life cases and propose a method to calculate lower bounds on passenger costs that is valid for all tree shaped networks. Furthermore, we assess different versions of the solution approach in a real life case study and we show that our solution approach including line planning operators obtains timetables with lower passenger costs than our solution approach with timetabling modifications only, using an additional budget on operating costs.

In Section 5.2 we review relevant research on the topics of line planning, timetabling, and the integration of both. Section 5.3 describes the AT-LP in detail and provides an example, and Section 5.4 describes the solution method. We present the case study in Section 5.5 and discuss the computational experiments in Section 5.6. Finally, Section 5.7 shows conclusions and points to future research avenues on the topic.

5.2 Literature Review

5.2.1 Line Planning

The line planning problem is well studied in the literature. Models for solving line planning can be divided in cost oriented models, passenger oriented models, and models that address other aspects of line planning. For an overview of different modelling and solution methods to line planning we direct the reader to Schöbel (2012).

Claessens et al. (1998) present a cost oriented model for solving line planning. Their model minimizes the operating costs of line plans, subject to service constraints and capacity requirements. To solve the problem, the authors present an algorithm based on constraint satisfaction and a branch-and-bound procedure. Goossens et al. (2006) present an extension of the cost-minimization model, considering the determination of stopping patterns. Other cost oriented models for line planning are Bussieck et al. (2004), Goossens et al. (2004), and Borndörfer et al. (2018a).
Passenger oriented models optimize line plans for the passengers and are presented in Bussieck et al. (1997a) and Schöbel and Scholl (2006). These models use different objective functions to reflect how passengers are affected by the line plan. Bussieck et al. (1997a) maximizes the number of direct travelers, i.e., passengers that do not have to transfer to reach their final destination. Schöbel and Scholl (2006) minimize the generalized travel time of passengers, meaning a weighted sum of travel time where transfers are penalized by a fixed amount.

Recently, other aspects of line planning are studied in the literature. The generation of line pools, which are assumed input for most line planning problems, is studied in Gattermann et al. (2017). Line pools should be large enough, but the larger they become the more intractable the line planning problem gets. Different stopping patterns along the same network are studied in Burggraeve et al. (2017) and Bull et al. (2018). Recently, Schiewe et al. (2019) model line planning as a game, where passengers are players trying to minimize their individual objective functions composed of different passenger service criteria.

### 5.2.2 Timetabling

Assuming known passenger demands for each vehicle or line and assuming known transfer patterns between lines at stations, passenger-oriented timetabling aims at minimizing travel times. In the case of railway planning, capacity constraints can be approximated by headway constraints between activities that use the same part of the network. Parbo et al. (2016) present an up-to-date overview of passenger considerations in railway timetabling. Railway timetabling problems can be divided in periodic and aperiodic. Periodic timetabling consists of determining arrivals and departure times from stations for an example period of time (usually one hour), which is then repeated throughout the day. Aperiodic timetable does not require this pattern repetition.

The periodic event scheduling problem (PESP) of Serafini and Ukovich (1989) is generally used for finding periodic timetables with minimal sum of travelling times, and serves as basis for many timetabling approaches in the literature (e.g., Nachtigall (1996), Peeters (2003), Liebchen (2008), or Schmidt and Schöbel (2015)). Originally, the PESP was modelled without an objective function but over the years different authors considered different objective functions to it. As an example, Burggraeve et al. (2017) use an objective function in the timetabling module that maximizes the minimum buffer times between services using the same part of the infrastructures, in an attempt to increase robustness. Also different constraints are added depending on the characteristics of the problem to model. For a recent overview of timetabling literature, specifically using PESP models, we direct the reader to Sels et al. (2016).

Aperiodic timetabling models that consider passenger service are presented in Barrena et al. (2014a,b), Niu et al. (2015), Robenek et al. (2016), and Yin et al. (2017). Barrena
et al. (2014a) present three formulations for the aperiodic timetabling problem, minimizing passenger average waiting time, that differ on the level of detail in modeling variables. Barrena et al. (2014b) present two mathematical (non-linear) programming formulations which generalize the non-periodic train timetabling problem on a single line under a dynamic demand pattern, also minimizing average passenger waiting time at the stations. Niu et al. (2015) consider an aperiodic timetabling problem for a single corridor and one direction that minimizes total passenger waiting time at stations. Services are allowed to use predefined patterns that skip stations. Robenek et al. (2016) maximize the profit of a train company while maintaining the level of passenger satisfaction, measured as an utility function considering in-vehicle time, waiting time, number of transfers and schedule passenger delay. Yin et al. (2017) include dynamic passenger demands in an aperiodic timetabling problem for a urban metro corridor, with the objective to jointly minimize operating costs and passenger waiting times.

5.2.3 Integrated line planning and timetabling

Previous research that integrates line planning and timetabling can be found in Liebchen and Möhring (2007), Michaelis and Schöbel (2009), Kaspi and Raviv (2012), Schöbel (2015, 2017), Pätzold et al. (2017), Burggraeve et al. (2017), and Jiang et al. (2017).

Liebchen and Möhring (2007) consider how the PESP can be integrated with important decisions of network planning, line planning, and vehicle scheduling. To integrate the PESP with line planning, the authors propose a way to model matching of pre-defined line segments into lines. This enables the creation of entirely new lines by combining segments of existing lines.

Michaelis and Schöbel (2009) address the integration of line planning, timetabling, and vehicle scheduling for bus systems, proposing that the sequential approach is reordered. First the vehicle routes are designed, then they are split into lines and finally a periodic timetable is calculated. The re-ordering allows modeling costs in all planning stages, but the approach still solves a sequence of optimization problems that are ordered in a different way than traditionally done. The objective function is customer-oriented in all three steps. The authors propose a heuristic to solve real-life cases. For a case study in the German city of Göttingen, the authors report a reduction of 10% in the number of buses while increasing the attractiveness of the timetables by 1%, with a computational time of 20 hours.

Kaspi and Raviv (2012) use a genetic algorithm to solve line planning and timetabling from scratch, minimizing both user inconvenience and operating costs. They use an initial line pool in which each line has a fixed number of potential services. They present a case study on the Israeli Railways network, where they report a reduction of 20% of average user travel time using the same amount of resources.
Problem description

In this section we formally describe the AT-LP and introduce the notation used in the remainder of the paper.

Given a public transportation network $\text{PTN} = (V, E)$, where $V$ is the set of stations and $E$ is the set of direct links between stations, consider that a line is a path in the PTN. Input from the line planning stage consists of a given set of lines $L$, each line $l \in L$ to...
be operated at a hourly frequency $f_l$. We define a service as a timetabled realization of a line $l \in L$, with departure and arrival times at stations included in the line. Based on the frequencies and on the size of the planning horizon $H$, measured in hours, we define the minimum and maximum number of services of each line $l \in L$, respectively $\delta_l^-$, $\delta_l^+$, depending on how much variation is allowed on the line plan’s frequency. Also based on the frequencies, we define the maximum headway $h_l^+$ between consecutive services of the same line $l \in L$.

Consider $T_l$ the set of services per line $l \in L$, where the size of $T_l$ is equal to the maximum number of services for line $l$. We define the set of all possible services as $\mathcal{T} = \bigcup_{l \in L} T_l$. We define $V_j \subset V$ as the set of all stations in the geo route of service $j \in \mathcal{T}$, and $V_l \subset V$ as the set of all stations in the geo route of line $l \in L$. $Q$ is the set of all track sections in the PTN, with $Q_j \subset Q$ being the set of all track sections in the geo route of service $j \in \mathcal{T}$, and $Q_l \subset Q$ being the set of all track sections in the geo route of line $l \in L$.

We assume known and fixed passenger demands in an input origin-destination-matrix ODt on the set $V$, i.e. $(o,d,t) \in \text{ODt}, (o,d) \subseteq V \times V$, where $o$ is an origin station, $d$ is a destination station, and $t$ is a desired departure time. Given a tree shaped PTN, consider also $\text{ODt}^{Tr} \subset \text{ODt}$ being the subset of all $(o,d,t)$ entries that need to transfer to reach their destination. Each entry $(o,d,t) \in \text{ODt}$ is associated with a number of passengers $\text{pax}_{(o,d,t)}$ and each $(o,d,t) \in \text{ODt}^{Tr}$ is associated with transfer station $\text{tr}_{(o,d,t)}$. The travel time between two stations $(u,v) \in V$ or between two track sections $(u,v) \in Q$ is defined as $s_{u,v}$.

The objective AT-LP is to minimize a weighted sum of costs associated with passengers. These are divided in four terms: in-vehicle time (IVT), initial waiting time (IWT), transfer time (TrT), and transfer penalty (TrP). The costs are calculated based on the cost per passenger minute $k_{\text{PM}}$, and each component has a different weight factor: $\pi_{\text{IWT}}$ is the penalty for each minute of initial waiting time, $\pi_{\text{TrP}}$ is the penalty for each transfer, and $\pi_{\text{TrT}}$ is the penalty for each minute of transfer time. In-vehicle time has a factor of 1. The objective function of the AT-LP is then:

$$\min k_{\text{PM}} \cdot \sum_{(o,d,t) \in \text{ODt}} \text{pax}_{(o,d,t)} \cdot (\text{IVT}_{(o,d,t)} + \pi_{\text{IWT}} \cdot \text{IWT}_{(o,d,t)} + \pi_{\text{TrT}} \cdot \text{TrT}_{(o,d,t)} + \pi_{\text{TrP}} \cdot \text{TrP}_{(o,d,t)})$$

(5.1)

The decisions to make are: (a) how many and which services should be launched, (b) at which stations should each launched service stop, and (c) what should be the departure and arrival times of each launched service from / to each visited station. These variables will define entering times at track sections and the order of services entering stations. More importantly, these decisions determine the services used by passengers in the ODt, defining the IVT, IWT, TrT, and TrP terms in the objective function. Passenger service relates with the timetable through shortest path computations that respect the passenger
assignment and transfer constraints, and that minimize the objective of the AT-LP.

The main constraints of the problem are:

- **Number of services.** The number of services scheduled for each line \( l \in L \) has to respect the minimum and maximum number of services for that line, \( \delta_l^-, \delta_l^+ \);

- **Canceled stops.** Services are allowed to cancel (skip stop at) intermediate stations. For each service \( j \in T_l \) and each line \( l \in L \), the maximum number of canceled stops \( c_l \) must be respected. Furthermore, two consecutive services of the same line \( l \in L \) cannot cancel a stop at the same station \( v \in V_l \);

- **Safety constraints.** The minimum headway \( h_q \) between consecutive services entering the same track section \( q \in Q \) must be respected. For consecutive services arriving or departing from a station \( v \in V \), arrival and departure headways \( h_{av}, h_{dv} \) must be respected;

- **Fixed dwell times.** If a service \( j \in T \) stops at a station \( v \in V_j \), then its dwell time is equal to the fixed dwell time \( w \);

- **Maximum line headway.** Two consecutive services of the same line \( l \in L \) are separated in time by at most the the maximum headway \( h_l^+ \);

- **Operating costs budget.** The number of train in-service minutes of all services in the timetable multiplied by the cost per train minute \( k^{TM} \) has to respect the budget on operating costs \( B \);

- **Passenger assignment.** Passengers of \((o,d,t) \in ODt\) can only board services departing from \( o \) later than their desired departure time \( t \);

- **Passenger transfers.** Transfers for passengers of \((o,d,t) \in ODt^{Tr}\) can only occur between services arriving and departing from the transfer station \( tr_{(o,d,t)} \) separated by at least the minimum transfer time \( e \).

### 5.3.1 Example

To illustrate the modifications allowed by the AT-LP, consider the PTN of 5 stops \{1,2,3,4,5\} and 4 direct links \{(1,2),(2,3),(3,4),(2,5)\} represented in Figure 5.1. Bi-directed lines \( l_1, l_2 \) and \( l_3 \) are also depicted, visiting respectively stops \{1,2,5\}, \{2,3,4\}, and \{1,2,3,4\}, and with hourly frequencies \( f_1, f_2, \) and \( f_3 \). At the line level, the AT-LP allows the frequencies to be slightly changed. Assuming a planning horizon of \( H \) hours, the number of services of line \( l \in L \) in the AT-LP is \( H \cdot f_l \pm \delta_l \), where \( \delta_l \) is a parameter that indicates how much frequency variations are allowed in line \( l \). To prevent services in the same line to be too far apart in time, maximum service based headways \( h_l^+ \) must be respected between each pair of services in the line.
Services may deviate from the stopping pattern associated with their line by including fewer stops. Figure 5.2 illustrates the modifications allowed by representing three different services \( \{j_1, j_2, j_3\} \) of line \( l_1 \). Service \( j_1 \) follows the same stopping pattern as line \( l_1 \). Service \( j_2 \) skips Station 2 and travels directly from Station 5 to Station 1. We impose a maximum number of canceled stops \( c_l \) for each service of line \( l \in L \). Furthermore, in the AT-LP it is not allowed that two consecutive services of the same line cancel a stop at the same station (thus inclusion of \( j_2 \) would enforce that \( j_1 \) and \( j_3 \) stop at Station 2). At each station \( v \), minimum arrival (\( h_v^a \)) and departure (\( h_v^d \)) headways must be respected for all services visiting that station.

5.4 Solution method

The solution method proposed to solve the AT-LP is a multi start heuristic, where a timetable is modified taking the passenger considerations into account, with guidance
towards solutions with low passenger costs. The diagram in Figure 5.3 presents the different building blocks of the solution method, which will be explained in the following subsections.

**Figure 5.3:** Flow diagram of the solution method

Input to the solution method is the set of lines $L$ and suggested frequencies $f_l$ for each line $l \in L$, an ODt matrix, passenger and operating costs information, and case study specific parameters: minimum transfer time at stations, minimum headways at track sections and stations, and fixed dwell times at stations. A lower bound $LB$ on passenger costs and the size of the planning horizon in hours $H$ are also part of the input. The lower bound on the passenger costs is calculated as described in Section 5.4.1.

The solution approach starts by calculating an initial timetable according to the construction heuristic described in Section 5.4.2, which includes checking for feasibility and fixing potential track section conflicts with the procedure of Section 5.4.3. If the solution is feasible and there are no track section conflicts, the solution method proceeds, otherwise a new initial solution is created. Once a feasible initial solution is found, the solution method selects randomly one operatorMove of the list of operators described in Section 5.4.4 and modifies the timetable. Again, feasibility and track section conflicts are checked and solved, and if the solution is feasible it is evaluated with the Evaluate procedure of Section 5.4.5. We check whether the new solution should be accepted as current solution, local best solution, and/or global best solution, according to the procedure accept() described in Section 5.4.6. In order to escape local minima, the solution method checks if the search should be restarted. Two options are available for restarting, either create
a new initial solution or return to a previously visited solution. Both these options are described in Section 5.4.6. Once the stopping criteria is met, the solution method outputs the global best solution found, its timetable, its associated passenger assignment, and its operating costs.

5.4.1  Lower Bound method

We compare solutions to a lower bound on passenger costs. The lower bound method (LB) used in our solution method is valid for all types of tree networks, where passenger paths are fixed as there is a single path connecting a pair of stations. For networks that contain cycles, where the passenger routes can vary depending on the timetable, the lower bound method should be modified to handle multiple paths for passengers.

The LB is calculated by summing independent lower bounds on the four components associated with passenger costs: in-vehicle time (IVT), initial waiting time (IWT), transfer time (TrT) and transfer penalty (TrP):

\[ LB = LB^{IVT} + LB^{IWT} + LB^{TrT} + LB^{TrP} \]

5.4.1.1  In-vehicle time

For each \((o, d, t) \in OD_t\), the lower bound on in-vehicle time corresponds to the minimum travel time between \(o\) and \(d\), assuming a direct service with no stops at intermediate stations. For \((o, d, t)\) entries that require transferring, it is assumed that the service stops additionally at the transfer station, and the minimum travel time is the sum of the unconstrained travel time from the origin to the transfer station and the unconstrained travel time from the transfer station to the destination station. The lower bound on in-vehicle time, \(LB^{IVT}\), is determined by summing the minimum transfer time \(IVT^{\min}_{(o,d,t)}\) over all \((o, d, t)\) entries, multiplied by the number of passengers in each \((o, d, t)\) and by the cost per passenger minute \(k^{PM}\):

\[ LB^{IVT} = k^{PM} \cdot \sum_{(o,d,t) \in OD_t} pax_{(o,d,t)} \cdot IVT^{\min}_{(o,d,t)} \]
5.4 Solution method

5.4.1.2 Initial waiting time

The lower bound on the initial waiting time is calculated per station $v \in V$ and per direction $g \in G_v$, where $G_v$ is the set of directions at station $v \in V$. Notice that in a tree network each station might have more than two directions. We determine the number of directions of a station $v \in V$ depending on the lines passing through that station, and cluster the $(o,d,t) \in OD_t, o = v$ entries originating in that station in directions according to their destination $d$. Once this is done, we determine the number of services passing through the origin station $o$ in each direction, assuming that all lines $l \in L$ are operated with the maximum number of services $\delta_l^+$, given the allowed modifications to the suggested frequencies.

We calculate the minimum passenger waiting time for each station and each direction using a dynamic programming recursion. The states in the dynamic recursion are defined as $W_{vg}[R][N]$, where $R$ represents the time in the planning horizon, $N$ represents the number of services already scheduled in direction $g \in G_v$ for station $v \in V$. $W_{vg}[r][n]$ is the minimum initial waiting time of all passengers traveling in direction $g \in G_v$ and departing from station $v \in V$ if $n$ services are scheduled in that direction up to time $r$, with the last services scheduled at time $r$. We initialize $W_{vg}[0][0] = 0$ and all other $W_{vg}$ to a large number. Then we expand for each time instant $r$, each number of scheduled services $n$, and all instants of time $\phi$ in the planning horizon using the recursion

$$ W[\phi][n] = \min_{\{r=0,\ldots,\phi-1\}} (W[r][n-1] + \text{paxArr}[v][g][r+1][\phi]) $$

where $\text{paxArr}[v][g][r+1][\phi]$ is the sum of all passengers departing from station $v \in V$ between the instants of time $r + 1$ and $\phi$, traveling in direction $g \in G_v$.

$$ \text{paxArr}[v][g][r+1][\phi] = \sum_{(o,d,t) \in OD_t: o=v, r+1 \leq t \leq \phi, d \in g} \text{pax}(o,d,t) $$

If the number of services to schedule in direction $g \in G_v$ at station $v \in V$ is larger than the number of $(o,d,t)$ entries in that direction, then the minimum waiting time is equal to zero for that direction, since the dynamic recursion will assume that services are scheduled matching the departure times of the $(o,d,t)$ entries. The lower bound on the initial waiting time, $\text{LB}_{IWT}$, is calculated by summing, over all stations $v \in V$ and all directions $g \in G_v$, the minimum initial waiting time $\text{IWT}_{v,g}^{\min}$ and multiplying it by the cost per passenger minute $k^{PM}$ and by the IWT penalty $\pi_{IWT}$.
Aperiodic public transport timetabling with flexible line plans

\[ LB^{IWT} = k^{PM} \cdot \pi^{IWT} \sum_{v \in V} \sum_{g \in G_v} IWT_{vg}^{\text{min}} \]

where the minimum initial waiting time \( IWT_{vg}^{\text{min}} \) is determined by the minimum value \( W_{vg}[R][N] \), with \( N \) being the number of services to schedule at that station in that direction, and \( R \) being an instant of time larger than the arrival time of the last \((o,d,t)\) entry at station \( v \in V \) with direction \( g \in G_v \). Given a granularity \( \gamma \) for the departures of passengers in the ODt matrix, we only need to expand the recursion for time instants \( r \) multiple of \( \gamma \), since these will provide the minimal initial waiting time.

### 5.4.1.3 Transfer time and transfer penalty

In a tree shaped PTN, the number of passengers that need to transfer to reach their destination is known. The lower bound on transfer time, \( LB^{TrT} \), is calculated assuming that all passengers that need to transfer, i.e., all passengers from entries \((o,d,t) \in ODt^{Tr}\), transfer at the minimum transfer time \( e \), multiplied by the cost per passenger minute \( k^{PM} \) and by the TrT penalty \( \pi^{TrT} \)

\[ LB^{TrT} = k^{PM} \cdot \pi^{TrT} \cdot e \cdot \sum_{(o,d,t) \in ODt^{Tr}} \text{pax}_{(o,d,t)} \]

The lower bound on transfer penalty, \( LB^{TrP} \), is calculated by multiplying the penalty for transfers \( \pi^{TrP} \) by the cost per passenger minute \( k^{PM} \) and by the number of transferring passengers

\[ LB^{TrP} = k^{PM} \cdot \pi^{TrP} \cdot \sum_{(o,d,t) \in ODt^{Tr}} \text{pax}_{(o,d,t)} \]

### 5.4.2 Construction heuristic

The \texttt{constructionHeuristic} creates an initial solution given a suggested line plan \( L \), suggested frequencies \( f_l \) for each line \( l \in L \), and a planning horizon \( H \) in hours. For each line \( l \in L \), the number of services to be scheduled is \( H \cdot f_l \). The \texttt{constructionHeuristic} assumes that all launched services visit all stations in their respective line, and that the
dwell times at each of these stations is equal to the fixed dwell time $w_v^-$. The departure time of the first service in each line $l \in L$ is determined randomly from the interval $[b, b + \frac{60}{f_l}]$, where $b$ is the starting time of the planning horizon. The arrival and departure times at subsequent stations are calculated according to the travel times between stations and the dwell times at each station. The services in each line are equally distributed in time, i.e., the difference between the departure of two consecutive services in the same line from the same stop is equal to $\frac{60}{f_l}$. Once the constructionHeuristic repeats this process for every line in the line plan, we have a initial timetable that may or may not be feasible in terms of track section and station headways. The construction heuristic is an iterative procedure that uses the feasibility checks and solution fixer presented in Section 5.4.3 until a feasible solution is found. The pseudocode for the constructionHeuristic is presented in Algorithm 3.

Algorithm 3: constructionHeuristic

\begin{algorithm}
\begin{algorithmic}[1]
\State \textbf{Input}: $H$, $L$, $f_l$, $w_v^-$, $s(u,v)$
\For {each line $l \in L$ do}
\State $|T_l| = H \cdot f_l$
\State rand_start = rand($b, b + \frac{60}{f_l} - 1$)
\For {each service $j \in T_l$ do}
\State $\tau_{d_jv}^j = \text{rand}\_\text{start} + j \cdot \frac{60}{f_l}$
\For {each station $v \in V_l \setminus \{v_j^1, v_j^N\}, j \in T_l$ do}
\State $\tau_{a_jv} = \tau_{d_j,v-1} + s(v-1,v)$
\State $\tau_{a_j} = \tau_{a_j} + w_v^-$
\EndFor
\State $\tau_{j,v}^N = \tau_{j,v}^{N-1} + s(v_N-1,v_N)$
\State determine entry times at track sections from departure times
\EndFor
\EndFor
\State \textbf{fixTrackSections}
\State \textbf{solutionChecker}
\If {solution not feasible}
\State go to step 1
\Else
\State return solution
\EndIf
\end{algorithmic}
\end{algorithm}

Although arrival time for the first station $v_j^1$ and departure time from the last station $v_j^N$ for each service $j$ are important for track availability, we do not assign them any value since they are not relevant for passengers. Step 11 fills the entry times at all track sections visited by each service. In fact, to find the travel times between each pair of stations in a direct link, we sum up the travel times across all track sections that compose the direct link. We need the entry times at track sections to guarantee that the safety headway constraints are respected.
5.4.3 Track section fixer and solution check

The solutionChecker is used to assess if all constraints are respected in a given solution to the AT-LP: no track section conflicts, consecutive services in the same line not canceling the same station, maximum number of canceled stops in a service respected, maximum headway between services in the same line respected for each station visited, minimum and maximum number of services in each line respected, and fixed dwell times at intermediate stations respected. If the solution has track section conflicts, the fixTrackSections procedure tries to solve them. Otherwise, a solution is considered infeasible by the solutionChecker.

The fixTrackSections procedure resolves track section conflicts in a candidate timetable $T^c$, if there are any. Conflicts in track sections are due to services entering the same track section $q \in Q$ without respecting the minimum headway at track sections $h_q$. The pseudocode for the fixTrackSections procedure is presented in Algorithm 4.

**Algorithm 4: fixTrackSections**

- **Input:** $T^c, Z_{T^c}, h_q, TL$
- **while** $TL$ not reached **do**
  - **while** $|Z_{T^c}| > 0$ **do**
    - Select $z = (j_1, j_2, q) \in Z_{T^c}$
    - Add $h_q - (\tau_{a,j_2,q}^a - \tau_{a,j_1,q}^a)$ to all arrival and departures from stations and to all entries at track sections of service $j_2$
    - Remove $z$ from $Z_{T^c}$
    - Add possible new conflicts to $Z_{T^c}$
  - **end while**
  - return $T^c$
- **end while**
- return infeasible

Input to the fixTrackSections procedure consists of a candidate timetable $T^c$ (which is a timetable that may not be feasible), a set of track section conflicts $Z_{T^c}$, the minimum headway at track sections $h_q$, and a computational time limit $TL$. Each track conflict in $Z_{T^c}$ is a triplet $(j_1, j_2, q)$ where services $j_1, j_2 \in T$ have a conflict in track section $q \in Q$. Furthermore, all triplets are sorted such that $\tau_{a,j_2,q}^a \leq \tau_{a,j_1,q}^a$. While the time limit is not reached and there are conflicts, the procedure picks a conflict $z = (j_1, j_2, q) \in Z_{T^c}$ and fixes it by advancing $j_2$ by an amount of time that solves the conflict. Conflict $z$ is solved and therefore removed from $Z_{T^c}$. If new conflicts are created with this change, they are added to $Z_{T^c}$. If $|Z_{T^c}| = 0$, the solution $T^c$ is conflict free and is returned. If the time limit $TL$ is reached, the procedure isn’t able to solve all conflicts in the stipulated time limit and returns that the solution is infeasible.
Table 5.1: Probabilities of selection given a number of feasible solutions after an operator modification

<table>
<thead>
<tr>
<th>Number of feasible solutions</th>
<th>Probabilities (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>{0, 0, 0, 0, 0}</td>
</tr>
<tr>
<td>1</td>
<td>{1, 0, 0, 0, 0}</td>
</tr>
<tr>
<td>2</td>
<td>{0.7, 0.3, 0, 0, 0}</td>
</tr>
<tr>
<td>3</td>
<td>{0.55, 0.3, 0.15, 0, 0}</td>
</tr>
<tr>
<td>4</td>
<td>{0.4, 0.25, 0.2, 0.15, 0}</td>
</tr>
<tr>
<td>5</td>
<td>{0.3, 0.25, 0.2, 0.15, 0.1}</td>
</tr>
</tbody>
</table>

5.4.4 Operators

Our method relies on a set of independent operators that modify the timetable and line plans of a solution. The operators described in this section modify the timetable of a feasible solution in an attempt to reduce the solution passenger costs. The timetable operator is the shift operator (\(\text{Sh}\)). Line planning operators are the two service operators (\(\text{Serv}^+\) and \(\text{Serv}^-\)) and the two stop operators (\(\text{Stp}^+\) and \(\text{Stp}^-\)). Different versions for each of the operators are presented in the following subsections. Some of the versions of the operators create 5 different solutions to select from based on the passenger costs associated with them. Table [5.1] shows the probabilities of selection given a number of feasible solutions to choose from, with the higher probabilities being the ones for the solutions with lower passenger costs.

Other versions of the operators select hourly time intervals \(i \in I\) of the planning horizon according to a probability dependent on the number of passengers traveling in that time interval in the current solution. The probability of selecting a time interval \(i\) is \(p_i = \frac{(o,d,t)_i}{(o,d,t)^{TOTAL}}\), where \((o,d,t)_i\) is the number of passengers traveling in travel interval \(i\) and \((o,d,t)^{TOTAL}\) is the total number of passengers in the OD\(t\) matrix.

5.4.4.1 Shift operator

The shift operator \(\text{Sh}\) modifies a single service \(j \in T\) by shifting all its departure and arrival times by the same amount of time \(\sigma\) in minutes. If \(\sigma\) is a negative number, all the arrivals and departures of service \(j\) will occur earlier in time, while if \(\sigma\) is a positive number they will occur later in time. In our method we test five versions of the \(\text{Sh}\) operator:

- \(\text{Sh}_1\): creates 5 different solutions, each of them by selecting a random service \(j \in T\) and shifting it by a random \(\sigma \in [-5, 5] \setminus \{0\}\). After evaluation, one of the 5 solutions is selected based on the probabilities of Table [5.1].

- \(\text{Sh}_2\): selects a random service \(j \in T\). Then creates 5 different solutions, each of them shifting \(j\) by a different random \(\sigma \in [-5, 5] \setminus \{0\}\). After evaluation, one of the
5 solutions is selected based on the probabilities of Table 5.1

- Sh₃: creates 5 different solutions, each of them by selecting a random service \( j \in T \), trying all possible shifts \( \sigma \in [-5, 5] \setminus \{0\} \), and selecting the \( \sigma \) value that results in the solution with lower passenger costs for each \( j \). After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1

- Sh₄: selects a time interval \( i \in I \) with probability \( p^i \). Then applies the same procedure as Sh₃, selecting only services \( j \in T^i \), where \( T^i \) is the set of services with start time within \( i \);

- Sh₅: selects a time interval \( i \in I \) with probability \( p^i \). Then applies the same procedure as Sh₂, selecting a service \( j \in T^i \), where \( T^i \) is the set of services with start time within \( i \).

### 5.4.4.2 Add service operator

The add service operator \( \text{Serv}^+ \) adds a service to a determined line \( l \in L \), given that it does not violate the maximum number of services for line \( l \), in an attempt to increase the frequency of this line in relation to the suggested frequency. All the remaining services in the timetable remain the same as in the previous solution, except for changes made by the \text{fixTrackSections} procedure. In our method we test 3 versions of the \( \text{Serv}^+ \) operator:

- \( \text{Serv}^+_1 \): creates 5 different solutions, each of them by adding a service to a random line \( l \in L \) and a random start time \( \tau \). After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1

- \( \text{Serv}^+_2 \): selects a time interval \( i \in I \) with probability \( p^i \), and selects a random line \( l \in L \). Then creates 5 different solutions, each of them by adding a service to line \( l \) at a random start time within \( \tau \in i \). After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1

- \( \text{Serv}^+_3 \): selects a time interval \( i \in I \) with probability \( p^i \), and selects a random line \( l \in L \). Then creates 12 solutions, each of them by adding a service to line \( l \) at a start time \( \tau \) multiple of 5, i.e., \( \tau \in \{00, 05, ..., 55\} \in i \). From the 12 solutions it selects the one with lower passenger costs.

### 5.4.4.3 Remove service operator

The remove service operator \( \text{Serv}^- \) creates 5 different solutions, each of them by removing a service from a random line \( l \in L \), given that it does not violate the minimum number of services for line \( l \). Once \( l \) is chosen, the operator removes the service \( j \in T_i \) which is used by the least passengers. The services of line \( l \) immediately before and after \( j \) are adjusted
in order to fulfill the maximum headway constraints for services of the same line. After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1.

5.4.4.4 Cancel stop operator

The cancel stop operator $Stp^-$ cancels a stop of a service $j \in T_l, l \in L$ at a determined intermediate station $v \in V_l \setminus \{v^1_j, v^N_j\}$ currently visited by that service. The timetable modifications associated with $Stp^-(j, v)$ is that the departure and arrival times of $j$ at $v$ will take the same value. Additionally, all departure and arrival times at stations after $v$ are reduced by an amount of time equal to the dwell time of $j$ at $v$ before the operator is applied. In our method we test 4 versions of the $Stp^-$ operator:

- $Stp^-_1$: creates 5 solutions, each of them by selecting a random service $j \in T$ and canceling a random station $v \in V_j \setminus \{v^1_j, v^N_j\}$. After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1.

- $Stp^-_2$: selects a time interval $i \in I$ with probability $p^i$. Then creates 5 solutions, each of them by selecting a random service $j \in T^i$, where $T^i$ is the set of services with start time within $i$, and canceling a random station $v \in V_j \setminus \{v^1_j, v^N_j\}$. After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1.

- $Stp^-_3$: selects a time interval $i \in I$ with probability $p^i$ and selects a random service $j \in T^i$, where $T^i$ is the set of services with start time within $i$. Then, it creates as many solutions as the number of stations $v \in V_j \setminus \{v^1_j, v^N_j\}$, each solution canceling one of the intermediate stations. The solution with lower passenger costs is selected.

- $Stp^-_4$: selects a random service $j \in T$. Then, it creates as many solutions as the number of stations $v \in V_j \setminus \{v^1_j, v^N_j\}$, each solution canceling one of the intermediate stations. The solution with lower passenger costs is selected.

5.4.4.5 Add stop operator

The add stop operator $Stp^+$ creates 5 solutions, each by adding a stop at station $v$ to a random service $j \in T$. Only stops that are on the original line plans can be added to a service, meaning that $Stp^+(j, v)$ can only be called if previously we have called $Stp^-(j, v)$ (assuming that we start from a solution where all services visit all stations in their respective line plan). When a stop is added to a service with this operator, it is added with a dwell time equal to the fixed dwell time. Therefore, the timetable modifications associated with $Stp^+(j, v)$ is a delay in the departure from $v$ equal to the fixed dwell time, and a shift in all departure and arrival times at stations after $v$ equal to the fixed dwell time. After evaluation, one of the 5 solutions is selected based on the probabilities of Table 5.1.
5.4.5 Evaluation

The Evaluation of a solution $S$ consists of calculating the passenger costs associated with the solution. Given a solution, we use shortest path computations to determine the passenger assignment and the passenger costs (IVT, IWT, TrT, TrP) associated with it. The Evaluation procedure assumes that if there are lines that visit both the origin $o$ and the destination $d$ of an $(o, d, t) \in \text{ODt}$, then passengers will take a service from one of those lines. It also assumes that passengers always take the service that minimizes their arrival time at the destination. When there are no lines that visit $o$ and $d$, passengers will transfer at a location that minimizes their arrival time at the destination.

In order to speed up the computation of the Evaluation procedure, only services with departure time from the first station within a time window around the departure time $d$ of the ODt entry are considered. Furthermore, if the Evaluation procedure is evaluating a change in a line or a service in a line of a solution for which the passenger costs were already computed before, the recalculation is done only for ODt entries that could potentially use the changed line. An entry in the ODt can potentially use a line if it has used a service in that line in previous iterations of the Evaluation procedure. Whenever a solution that can potentially improve the global best solution ($S^*$) is found, the information of which lines can be used by each ODt is not taken into account, hence a more precise evaluation procedure is run, which also takes slightly more time to compute.

5.4.6 Acceptance of solutions and restarting the search

Our method allows accepting up to $\mu$ worse solutions in an attempt to escape local minima. Furthermore, after a certain number of iterations without improvement, which we denote by restart parameter $\nu$, the search is restarted in order to attempt different search avenues. When a new solution $S$ is evaluated we have to decide whether to accept it as global best solution ($S^*$), local best solution ($S^+$), and/or current solution ($S'$). These decisions depend on the passenger costs of $S$, which we denote as $PC_S$ and are the value returned by the Evaluation procedure, and on the current number of solutions without improvement $\theta$ and current number of worse solutions accepted $\kappa$. Algorithm 5 is the pseudocode for the acceptance of solutions.

A solution $S$ that is accepted as $S^*$ is also accepted as $S^+$ and $S'$, and a solution $S$ that is accepted as $S^+$ is also accepted as $S'$. When a new solution is accepted as $S^*$ or $S^+$, the current number of worse solutions accepted and current number of iterations without improvement are reset to zero. Once the restart parameter $\nu$ is reached, the search is restarted. We tested two different methods for restarting the search:

- **Restart** flex: a new solution is generated using the construction heuristic described in Section 5.4.2. This has the drawback of restarting from a solution that is probably...
Algorithm 5: accept()

Input: $S, S^*, S^+, S', \mu, \nu, \theta, \kappa$

1: \textbf{if} $PC_S < PC_{S^*}$ \textbf{then}
2: \{\(S^*, S^+, S'\)\} $\leftarrow$ $S$
3: \{\(\mu, \nu\)\} $\leftarrow$ 0
4: \textbf{else if} $PC_S < PC_{S^+}$ \textbf{then}
5: \{\(S^+, S'\)\} $\leftarrow$ $S$
6: \{\(\mu, \nu\)\} $\leftarrow$ 0
7: \textbf{else if} $\kappa < \mu$ \textbf{then}
8: \{\(S'\)\} $\leftarrow$ $S$
9: \text{increase} $\kappa$ \text{by} 1
10: \textbf{else}
11: \text{increase} $\theta$ \text{by} 1
12: \textbf{end if}
13: \textbf{if} $\theta = \nu$ \textbf{then}
14: restart using one of the restart methods
15: \textbf{end if}

much worse that the current solution, but it can completely reshuffle the timetables and get to better solutions;

- \textbf{Restart}$_{\text{prev}}$: the new solution is a previously visited solution selected at random from a list of potential solutions to return to. When this restart option is selected, our method keeps a list of 5 potential solutions to return to, all of them separated by at least 0.5% in gap to the lower bound on passenger costs. Once a new solution is added to this list, the worst in the list is removed. Consequently, at any given point during the search the five solutions saved as possible return to solutions will be up to \{0.5, 1, 1.5, 2, 2.5\}% worse than the current solution, with the exception being at the start of the search, where the five slots are initialized as the initial solution.

5.5 Case study

We test our solution method in a network composed by a subset of the intercity train lines in The Netherlands. Consider Figure 5.4 which is a geographical representation of the network considered, that consists of the six IC lines that stop in Gouda, 18 stations and 17 direct links. The lines are operated by Netherlands Railways (NS), the main rail operator in The Netherlands, who is also responsible for the line planning and timetabling. Data for the case study was provided by NS, including line plans, planned frequencies, track sections traversed by each direct link, and estimates of costs. The suggested hourly frequencies $f_l$ for each line $l \in L$ in the network are indicated in Figure 5.4.

Input data for the AT-LP consists of: (i) stations information (name, coordinates, number
of platforms), (ii) track sections information (connection points, distance, average speed), (iii) line information (frequency, stations and track sections visited), (iv) an ODt matrix (origin, destination, time of departure, and number of passengers), and (v) case study specific costs and parameters: costs per train minute, minimum transfer time at stations, minimum headways at track sections and stations, fixed dwell times at stations, passenger costs (different for IWT, IVT, TrT, and TrP).

We obtained from NS data related to (i)-(iii) and the discussed with them the parameters of (v). From (ii) and (iii) it was possible to calculate average travel times for all the direct links in the network. The maximum number of canceled stops $c_l$ in a service $j \in T_l$ is calculated based on the number of stations in the line $l \in L$, using the formula

$$c_l = \left\lceil \frac{|V_l|}{4} \right\rceil$$

where $|V_l|$ is the number of stations in the geo route of line $l$. Given a planning horizon $H$, we can calculate the minimum and maximum number of services in each line $l \in L$, $\delta_l^-$ and $\delta_l^+$ respectively, using the formulas

$$\delta_l^- = H \cdot f_l - \left\lfloor \frac{H \cdot f_l}{4} \right\rfloor$$

$$\delta_l^+ = H \cdot f_l + \left\lceil \frac{H \cdot f_l}{4} \right\rceil$$

The maximum headway $h_l^+$ between services of the same line $l \in L$ is also calculated based on the frequency $f_l$, using the formula

$$h_l^+ = \frac{60}{f_l} + \left\lceil \frac{25}{f_l} \right\rceil$$
The track section minimum headway $h_q$ is 3 minutes, and this is also the value for arrival and departure headways, $h^a_v$ and $h^d_v$, at each station $v \in V$. Other than the minimum track section headway, no minimum headway is considered for services of the same line in order to allow extra services at times of the day with high demand. The minimum transfer time at stations $e$ is 3 minutes. The fixed dwell time at stations is 1 minute.

We estimated the yearly cost of a train to be 5.5 millions of euros, which results in a train minute cost $k_{TM}$ of 10.5 euros. The budget on the operating costs is 10%, thus solutions with operating costs 10% higher than the operating costs of the initial solution are considered feasible. Considering an average year income of 36500 euros and considering 8 hours of work in each of the days of the year, the passenger minute cost $k_{PM}$ is 0.21 euros. The IVT cost is equal to the passenger minute cost $k_{PM}$, while the IWT and TrT costs are multiplied by factors $\pi_{IWT}$ and $\pi_{TrT}$ of 3. We consider a transfer penalty $\pi_{TrP}$ of $10 \cdot k_{PM}$ euros.

Regarding the ODt matrix, for confidentiality reasons NS does not provide information on OD data. Instead, OD data was created using an algorithm which generates ODt matrices proportional to the population and inversely proportional to the distance. It also generates higher demand volumes in morning and afternoon peak hours. Input to the generator is composed by:

- List of all stations, including x and y coordinates and population;
- Granularity of the ODt matrix to be generated. In our case we chose a granularity $\gamma$ of 5 minutes, meaning that for each OD pair there is an entry for every instant of time multiple of 5;

In order to balance the granularity of 5 minutes, all initial waiting times are rounded down to a multiple of 5 using

$$IWT = IWT - (IWT \mod \gamma)$$

## 5.6 Computational experiments

The solution method proposed to solve the AT-LP is tested through various computational experiments for the case study presented in Section 5.5. The quality of solutions is evaluated in comparison to the lower bound values obtained for passenger costs, using the lower bound method of Section 5.4.1.

All experiments are run for the two different methods to restart the search presented in Section 5.4.6 $\text{Restart}^\text{Rev}$ and $\text{Restart}^\text{Prev}$, and for for three different instances with 5,
12, and 19 hours of planning horizon, all starting at 5am. The 5 hour planning horizon instance represents the morning peak hour, the 12 hour planning horizon instance represents the period including both morning and afternoon peak hours, and the 19 hour instance represented the period from the morning peak hour until the end of the day.

This section starts with tuning experiments for the restart parameter $\nu$ and for the different operators. All operators are initially evaluated using their most random version, i.e., $\text{Sh}_1$, $\text{Serv}_1^+$, and $\text{Stp}_1^-$, in which five different random solutions are compared and one of them is chosen based on a probability dependent on the associated passenger costs. We illustrate the value of integrated timetabling and line planning by contrasting the performance of our integrated solution method, with the performance of the solution method allowing timetabling modifications only. We show convergence and passenger costs analysis for the best solution obtained, and we perform a sensitivity analysis on the budget on operating costs.

The solution method is implemented in C++ and all experiments used a HPC server, running on Intel Xeon E5-2660 v3 2.60GHz processors, and 1 computation core. All results presented are average results over five runs with each setting of parameters, and all experiments used a 3 hour computation time limit. Solutions are compared with the lower bound results in terms of gap, calculated as

$$\text{Gap} = \frac{PC^S - \text{LB}}{\text{LB}} \cdot 100\%.$$ 

Note that different planning horizons have different lower bound values $\text{LB}$, so when calculating the gap we have to use the respective $\text{LB}$ value. Solution quality is expressed in terms of average gap (Avg Gap), best gap (Best Gap), standard deviation of the gap (Std dev), number of iterations (Iterations), and number of hyper-iterations (HyperIterations). We start a new hyper-iteration each time we use one of the restart methods.

### 5.6.1 Tuning experiments

#### 5.6.1.1 Tuning of the restart parameter ($\nu$)

In this section we present results of experiments for tuning the restart parameter $\nu$. This parameter directly influences how often the method restarts the search (number of hyper-iterations), and therefore has an influence on how many local minima it can escape. The reasoning is: each time the search restarts we go back to a solution with a higher gap than the solution the method was stuck at, in order to aim to escape the previous local minimum. The restart comes at the cost of investing computation time in returning to solutions of similar quality as before the restart.
5.6 Computational experiments

Table 5.2 shows results for the tuning of $\nu$. In these experiments we test $\nu = \{20, 50, 100, 200\}$ and we tune it for all three sizes of planning horizon considered, $H = \{5, 12, 19\}$, and for both methods to restart the search, $\text{Restart}_{\text{New}}$ and $\text{Restart}_{\text{Prev}}$. The other operators are fixed to $\text{Sh}_1$, $\text{Serv}^+_1$, and $\text{Stp}^-_1$.

**Table 5.2:** Results for the experiments to tune the restart parameter

<table>
<thead>
<tr>
<th>$H$</th>
<th>$\nu$</th>
<th>Avg Gap</th>
<th>Min Gap</th>
<th>Std dev</th>
<th>Iterations</th>
<th>HyperIterations</th>
<th>Avg Gap</th>
<th>Min Gap</th>
<th>Std dev</th>
<th>Iterations</th>
<th>HyperIterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>20</td>
<td>49.64</td>
<td>49.64</td>
<td>0.00</td>
<td>30490</td>
<td>87</td>
<td>48.02</td>
<td>45.86</td>
<td>1.79</td>
<td>28033</td>
<td>258</td>
</tr>
<tr>
<td>50</td>
<td>47.72</td>
<td>46.62</td>
<td>0.91</td>
<td>30305</td>
<td>25</td>
<td>45.84</td>
<td>43.92</td>
<td>2.14</td>
<td>25004</td>
<td>89</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>47.98</td>
<td>46.62</td>
<td>0.98</td>
<td>27715</td>
<td>17</td>
<td>47.38</td>
<td>43.37</td>
<td>2.42</td>
<td>21685</td>
<td>47</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>48.72</td>
<td>47.18</td>
<td>1.38</td>
<td>27479</td>
<td>13</td>
<td>46.88</td>
<td>43.36</td>
<td>2.50</td>
<td>19246</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>20</td>
<td>37.54</td>
<td>37.20</td>
<td>0.19</td>
<td>10943</td>
<td>38</td>
<td>38.19</td>
<td>37.57</td>
<td>0.58</td>
<td>10751</td>
<td>60</td>
</tr>
<tr>
<td>5</td>
<td>38.24</td>
<td>37.56</td>
<td>0.46</td>
<td>10653</td>
<td>10</td>
<td>37.40</td>
<td>34.78</td>
<td>2.07</td>
<td>8762</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>38.39</td>
<td>37.56</td>
<td>0.61</td>
<td>9717</td>
<td>8</td>
<td>37.50</td>
<td>35.58</td>
<td>1.30</td>
<td>7644</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>38.59</td>
<td>37.56</td>
<td>0.77</td>
<td>8653</td>
<td>6</td>
<td>38.06</td>
<td>36.91</td>
<td>1.06</td>
<td>6396</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>32.70</td>
<td>32.12</td>
<td>0.32</td>
<td>6419</td>
<td>20</td>
<td>34.72</td>
<td>32.82</td>
<td>0.97</td>
<td>6185</td>
<td>34</td>
</tr>
<tr>
<td>5</td>
<td>33.78</td>
<td>33.28</td>
<td>0.44</td>
<td>5893</td>
<td>6</td>
<td>34.30</td>
<td>33.26</td>
<td>0.87</td>
<td>5253</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>33.71</td>
<td>32.83</td>
<td>0.62</td>
<td>5659</td>
<td>4</td>
<td>34.46</td>
<td>32.52</td>
<td>1.28</td>
<td>4469</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>33.63</td>
<td>32.83</td>
<td>0.58</td>
<td>4755</td>
<td>4</td>
<td>34.75</td>
<td>33.38</td>
<td>1.22</td>
<td>3643</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

Results of Table 5.2 indicate that, although there is no single value for $\nu$ that is associated with the lowest average and minimum gaps over all planning horizons and restart methods, there is a tendency for smaller values of $\nu$ to achieve smaller gaps. Furthermore, the difference between average gaps for the same planning horizon is small, when using different values for $\nu$.

For the experiments with $\text{Restart}_{\text{New}}$, $\nu = 20$ obtains the lowest average gap for large planning horizons, respectively 37.54% and 32.70% for $H = 12$ and $H = 19$. Furthermore, $\nu = 20$ also obtains the minimum gap solution for these two planning horizons, respectively 37.20% and 32.12% for $H = 12$ and $H = 19$. Also, it does so at the minimum standard deviation. However, it is the worse performing value for $\nu$ for the planning horizon of 5 hours. Since it is the value with lowest best gap and lowest average gap for two out of three values of $H$, also because it performs well for large planning horizons, we select $\nu = 20$ for all the remainder of the experiment using $\text{Restart}_{\text{New}}$.

Experiments using $\text{Restart}_{\text{Prev}}$ show that $\nu = 50$ provides the lowest average gap for all three planning horizons considered – respectively 45.84%, 37.40%, and 34.30% for $H = 5$, $H = 12$, and $H = 19$. This value for $\nu$ does not obtain the minimum gap solution for all planning horizons, but when it does not it provides a gap with a small difference to the minimum gap obtained. Due to this consistence in performance, we fix $\nu = 50$ for all the remaining experiments that use $\text{Restart}_{\text{Prev}}$.

As mentioned before, increasing $\nu$ means decreasing the number of hyper-iterations, which is supported by the results in Table 5.2. However, as $\nu$ increases, the number of iterations also decreases. This might be related with the number of solutions found that improve the global best solution. The quick Evaluation method to evaluate the passenger costs after a change in the timetable is not as precise as the full Evaluation of passenger costs. Hence, when we find a solution that potentially improves $S^*$ (according to the quick evaluation),
we also run the full Evaluation to be sure. If we are in a situation where our quick evaluation finds several solutions that are potentially better, the solution method takes longer to compute the Evaluation and consequently the number of iterations decreases.

5.6.1.2 Tuning the shift operator

In this section we present tuning results for the different versions of the shift operator Sh. Table 5.3 shows results for all five versions of the Sh operator presented in Section 5.4.4.1 for all three sizes of planning horizon considered, and for both methods to restart the search, Restart$^\text{New}$ and Restart$^\text{Prev}$. The other operators with different versions are fixed to Serv$^+_1$, and Stp$_1$.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Sh$_1$</td>
<td>49.64</td>
<td>49.64</td>
<td>0.00</td>
<td>30490</td>
<td>87</td>
<td>45.84</td>
<td>43.92</td>
<td>2.14</td>
<td>25004</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>Sh$_2$</td>
<td>50.98</td>
<td>50.98</td>
<td>0.00</td>
<td>30778</td>
<td>79</td>
<td>49.28</td>
<td>45.19</td>
<td>2.43</td>
<td>20636</td>
<td>116</td>
</tr>
<tr>
<td></td>
<td>Sh$_3$</td>
<td>47.04</td>
<td>47.04</td>
<td>0.00</td>
<td>10224</td>
<td>24</td>
<td>46.75</td>
<td>42.93</td>
<td>2.99</td>
<td>8898</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>Sh$_4$</td>
<td>48.63</td>
<td>47.32</td>
<td>0.68</td>
<td>10789</td>
<td>23</td>
<td>47.66</td>
<td>44.73</td>
<td>2.49</td>
<td>9357</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Sh$_5$</td>
<td>47.97</td>
<td>47.73</td>
<td>0.48</td>
<td>31264</td>
<td>84</td>
<td>48.81</td>
<td>45.81</td>
<td>1.82</td>
<td>20596</td>
<td>116</td>
</tr>
<tr>
<td>12</td>
<td>Sh$_1$</td>
<td>37.54</td>
<td>37.20</td>
<td>0.19</td>
<td>10943</td>
<td>38</td>
<td>37.40</td>
<td>34.78</td>
<td>2.07</td>
<td>8762</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Sh$_2$</td>
<td>37.16</td>
<td>37.16</td>
<td>0.00</td>
<td>11139</td>
<td>34</td>
<td>37.93</td>
<td>36.32</td>
<td>0.99</td>
<td>8246</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>Sh$_3$</td>
<td>37.76</td>
<td>36.86</td>
<td>0.68</td>
<td>4116</td>
<td>14</td>
<td>37.01</td>
<td>35.06</td>
<td>1.32</td>
<td>3470</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Sh$_4$</td>
<td>37.32</td>
<td>36.94</td>
<td>0.52</td>
<td>4102</td>
<td>13</td>
<td>37.19</td>
<td>35.10</td>
<td>1.23</td>
<td>3458</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Sh$_5$</td>
<td>37.28</td>
<td>36.24</td>
<td>0.53</td>
<td>11604</td>
<td>36</td>
<td>38.43</td>
<td>36.62</td>
<td>1.41</td>
<td>8490</td>
<td>25</td>
</tr>
<tr>
<td>19</td>
<td>Sh$_1$</td>
<td>32.70</td>
<td>32.12</td>
<td>0.32</td>
<td>6419</td>
<td>20</td>
<td>34.30</td>
<td>33.26</td>
<td>0.87</td>
<td>5253</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Sh$_2$</td>
<td>33.86</td>
<td>33.73</td>
<td>0.16</td>
<td>6422</td>
<td>20</td>
<td>34.73</td>
<td>33.19</td>
<td>0.97</td>
<td>5990</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Sh$_3$</td>
<td>34.26</td>
<td>33.26</td>
<td>0.61</td>
<td>2514</td>
<td>7</td>
<td>33.65</td>
<td>32.57</td>
<td>0.95</td>
<td>2198</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Sh$_4$</td>
<td>34.73</td>
<td>34.01</td>
<td>0.59</td>
<td>2423</td>
<td>7</td>
<td>34.47</td>
<td>33.61</td>
<td>0.66</td>
<td>2062</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Sh$_5$</td>
<td>34.14</td>
<td>33.21</td>
<td>0.75</td>
<td>6537</td>
<td>19</td>
<td>35.21</td>
<td>34.23</td>
<td>0.71</td>
<td>5069</td>
<td>15</td>
</tr>
</tbody>
</table>

Experiments using Restart$^\text{New}$ show that using different versions of the shift operator leads to similar average gaps for each planning horizon. Especially for $H = 12$, the average gaps across all shift versions differ by at most 0.60%. There is not one clear version that performs better than the others, but Sh$_1$ obtains the lower average gap and minimum gap for $H = 19$. Furthermore, this version of the shift operator presents low standard deviations, and low average and minimum gaps for $H = 12$. For this reason, we fix the shift operator to Sh$_1$ for all remaining experiments using Restart$^\text{New}$.

For experiments using Restart$^\text{Prev}$ we observe that the Sh$_3$ version obtains the lowest average gaps for both $H = 12$ and $H = 19$, 37.01% and 33.65% respectively. Additionally, this version obtains also the minimum gaps for $H = 5$ and $H = 19$, and although is does not obtain the lowest average gap for $H = 5$ it obtains the second lowest. We fix the shift operator to Sh$_3$ for the remainder of experiments that use the Restart$^\text{Prev}$ method.
5.6 Computational experiments

5.6.1.3 Tuning the add service operator

The results presented in this section refer to the experiments for tuning the different versions of the add service operator of Section 5.4.4.2. Table 5.4 shows results for all three versions of the $\text{Serv}_+^1$, for all three sizes of planning horizon considered, and for all methods to restart the search, $\text{Restart}^\text{New}$ and $\text{Restart}^\text{Prev}$. The other operators with multiple versions are fixed to $\text{Sh}_1$ (for $\text{Restart}^\text{New}$ experiments), $\text{Sh}_3$ (for $\text{Restart}^\text{Prev}$ experiments), and $\text{Stp}_1^-$. The results of Table 5.4 indicate that for the $\text{Restart}^\text{New}$ method, and for planning horizons $H = 5$ and $H = 12$, the $\text{Serv}_3^+$ version of the add service operator obtains the lowest average gaps and the lowest minimum gaps. In the $\text{Restart}^\text{Prev}$ experiments, this version of the operator obtains the lowest average gaps and the lowest minimum gaps for large planning horizons $H = 12$ and $H = 19$. Considering the results of Table 5.4, we fix the add service operator to version $\text{Serv}_3^+$ for the remainder of the experiments using both $\text{Restart}^\text{New}$ and $\text{Restart}^\text{Prev}$ methods.

Table 5.4: Results for the experiments to tune the add service operator

<table>
<thead>
<tr>
<th>H</th>
<th>Version</th>
<th>$\text{Restart}^\text{New}, \nu = 20, \text{Sh}_1, \text{Stp}_1^-$</th>
<th>$\text{Restart}^\text{Prev}, \nu = 50, \text{Sh}_1, \text{Stp}_1^-$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>$\text{Serv}_1^+$</td>
<td>Avg Gap: 49.64, Min Gap: 49.64, Std dev: 0.00, HyperIterations: 987</td>
<td>Avg Gap: 46.75, Min Gap: 42.91, Std dev: 2.99, HyperIterations: 87</td>
</tr>
<tr>
<td></td>
<td>$\text{Serv}_2^+$</td>
<td>Avg Gap: 49.80, Min Gap: 47.95, Std dev: 1.01, HyperIterations: 28714</td>
<td>Avg Gap: 49.49, Min Gap: 45.37, Std dev: 2.39, HyperIterations: 8972</td>
</tr>
<tr>
<td></td>
<td>$\text{Serv}_3^+$</td>
<td>Avg Gap: 47.49, Min Gap: 47.10, Std dev: 0.32, HyperIterations: 17114</td>
<td>Avg Gap: 49.03, Min Gap: 46.47, Std dev: 1.70, HyperIterations: 7236</td>
</tr>
<tr>
<td>12</td>
<td>$\text{Serv}_1^+$</td>
<td>Avg Gap: 37.54, Min Gap: 37.20, Std dev: 0.19, HyperIterations: 10943</td>
<td>Avg Gap: 37.01, Min Gap: 35.06, Std dev: 1.32, HyperIterations: 3470</td>
</tr>
<tr>
<td></td>
<td>$\text{Serv}_2^+$</td>
<td>Avg Gap: 37.60, Min Gap: 36.97, Std dev: 0.65, HyperIterations: 10531</td>
<td>Avg Gap: 37.48, Min Gap: 35.64, Std dev: 1.56, HyperIterations: 3343</td>
</tr>
<tr>
<td></td>
<td>$\text{Serv}_3^+$</td>
<td>Avg Gap: 35.68, Min Gap: 34.87, Std dev: 0.85, HyperIterations: 6490</td>
<td>Avg Gap: 35.63, Min Gap: 34.17, Std dev: 1.39, HyperIterations: 2862</td>
</tr>
<tr>
<td>19</td>
<td>$\text{Serv}_1^+$</td>
<td>Avg Gap: 32.70, Min Gap: 32.12, Std dev: 0.32, HyperIterations: 6419</td>
<td>Avg Gap: 33.65, Min Gap: 32.57, Std dev: 0.95, HyperIterations: 2198</td>
</tr>
<tr>
<td></td>
<td>$\text{Serv}_2^+$</td>
<td>Avg Gap: 33.18, Min Gap: 32.45, Std dev: 0.63, HyperIterations: 6184</td>
<td>Avg Gap: 34.18, Min Gap: 33.66, Std dev: 0.59, HyperIterations: 2156</td>
</tr>
<tr>
<td></td>
<td>$\text{Serv}_3^+$</td>
<td>Avg Gap: 33.81, Min Gap: 33.30, Std dev: 0.36, HyperIterations: 3808</td>
<td>Avg Gap: 32.44, Min Gap: 31.75, Std dev: 0.61, HyperIterations: 1688</td>
</tr>
</tbody>
</table>

5.6.1.4 Tuning the cancel stops operator

In this section we present results for tuning the different versions of the cancel stop operator $\text{Stp}^-$ presented in Section 5.4.4.4. Table 5.5 shows results for all versions of the cancel stop operator, for all three sizes of planning horizon considered, and for both methods to restart the search, $\text{Restart}^\text{New}$ and $\text{Restart}^\text{Prev}$. All other operators with multiple versions are fixed to $\text{Sh}_1$ (for $\text{Restart}^\text{New}$ experiments), $\text{Sh}_3$ (for $\text{Restart}^\text{Prev}$ experiments), and $\text{Serv}_3^+$. The experiments of Table 5.5 indicate that for the smallest planning horizon $\text{Stp}_3^-$ of the cancel stop operator obtains the lowest average gaps and lowest minimum gaps. For $H = 12$, is version $\text{Stp}_1^-$ that obtains the the lowest average gaps and lowest minimum gaps. However, for $H = 19$ there is not a clear best performing version, with $\text{Stp}_4^-$ obtaining the lowest average gap of 32.84% for the $\text{Restart}^\text{New}$ method and $\text{Stp}_3^-$ obtaining the lowest average gap of 32.17% for the $\text{Restart}^\text{Prev}$ method. We decide to fix the
cancel stop operator to Stp_3^- for the Restart^{Rev} experiments, due to obtaining the lowest minimum gap on two out of three planning horizons, to obtaining the average gap for $H = 5$ and the second lowest average gap for $H = 12$. Regarding the Restart^{Prev} method, we fix the remaining experiments to Stp_2^-, since this version obtains the lowest average and minimum gaps for $H = 19$, obtains the second lowest average gap for $H = 12$, and obtains the second lowest minimum gap for $H = 5$.

5.6.1.5 Summary of tuning experiments

Summarizing the tuning experiment, the results in terms of parameter and operators settings are shown in Table 5.6.

<table>
<thead>
<tr>
<th>$H$</th>
<th>$\nu$</th>
<th>Version</th>
<th>Stp_1^-</th>
<th>Stp_2^-</th>
<th>Stp_3^-</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>20</td>
<td>Avg Gap</td>
<td>47.49</td>
<td>47.10</td>
<td>47.40</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min Gap</td>
<td>47.10</td>
<td>47.10</td>
<td>46.43</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Std dev</td>
<td>0.32</td>
<td>0.64</td>
<td>0.49</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Iterations</td>
<td>17111</td>
<td>14025</td>
<td>16113</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>Avg Gap</td>
<td>49.03</td>
<td>47.17</td>
<td>47.09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min Gap</td>
<td>46.47</td>
<td>45.20</td>
<td>41.88</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Std dev</td>
<td>1.70</td>
<td>3.06</td>
<td>3.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Iterations</td>
<td>7236</td>
<td>7238</td>
<td>7044</td>
</tr>
<tr>
<td>12</td>
<td>20</td>
<td>Avg Gap</td>
<td>35.68</td>
<td>35.16</td>
<td>36.48</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min Gap</td>
<td>34.87</td>
<td>35.67</td>
<td>36.31</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Std dev</td>
<td>0.85</td>
<td>0.87</td>
<td>0.10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Iterations</td>
<td>6490</td>
<td>6429</td>
<td>5651</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>Avg Gap</td>
<td>35.63</td>
<td>35.35</td>
<td>36.46</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min Gap</td>
<td>34.17</td>
<td>35.59</td>
<td>36.51</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Std dev</td>
<td>1.39</td>
<td>0.82</td>
<td>0.11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Iterations</td>
<td>2862</td>
<td>2774</td>
<td>2628</td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>Avg Gap</td>
<td>33.81</td>
<td>33.65</td>
<td>32.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min Gap</td>
<td>33.30</td>
<td>31.61</td>
<td>32.41</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Std dev</td>
<td>0.36</td>
<td>1.01</td>
<td>0.33</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Iterations</td>
<td>3808</td>
<td>3735</td>
<td>3468</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>Avg Gap</td>
<td>32.44</td>
<td>33.17</td>
<td>32.71</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min Gap</td>
<td>31.75</td>
<td>32.29</td>
<td>31.53</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Std dev</td>
<td>0.61</td>
<td>0.96</td>
<td>0.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Iterations</td>
<td>1688</td>
<td>1693</td>
<td>1639</td>
</tr>
</tbody>
</table>

5.6.2 Analysis of results

From the results in the tuning section, we observe in Tables 5.2, 5.3, 5.4, and 5.5 that the gaps obtained for larger values of $H$ are lower, which indicates that the lower bound method is more accurate for larger instances. The LB^{TrP} determines the exact value of transfer penalties, since in a tree shaped network this determined. LB^{IVT} are fairly well estimated, missing only the stopping times at intermediate stops. LB^{IWT} and LB^{TrT} are worse estimations, but when the number of services increases this is mitigated in the lower bound method and it attains a lower gap. Other reason for having larger gaps in $H = 5$ than in the other values of $H$ is that $H = 5$ is a peak-hour only period of time, thus with
more passengers per unit of time in the planning horizon, while the other two planning horizons include both peak and off-peak hour periods.

In general, both the average gaps and the best gaps obtained by restarting from a previously visited solution with \texttt{Restart}^{\text{Prev}} are lower than restarting from a completely new solution with \texttt{Restart}^{\text{New}}. This indicates that the algorithm is able to escape local minima by applying the operators in a different order than in previous iterations. On the other hand, it indicates that when determining a new initial solution, which likely has a higher gap than the local minimum, a lot of computational time is used in getting to at least the same value of lower bound. This results in the solution method not being able to perform as well as when the \texttt{Restart}^{\text{Prev}} method is used.

The results from the tuning experiments can be resumed in the following points:

- in general, small values of \( \nu \) allow the solution method to obtain lower gaps than large values of \( \nu \);
- for \texttt{Restart}^{\text{New}} experiments, deciding shifts in a completely random fashion with the \texttt{Sh}1 version allows in general finding the lowest gaps. On the other hand, for \texttt{Restart}^{\text{Prev}} experiments it appears that the \texttt{Sh}3 version, which extensively tries all possible shift values once a service is selected, allows obtaining the lowest gaps;
- for both \texttt{Restart}^{\text{New}} and \texttt{Restart}^{\text{Prev}} experiments, the selection of the \texttt{Serv}^+ version shows that it is worthy to spend more computational time evaluating all alternatives for adding a service, and select when to schedule it in a deterministic way, even if this means running less iterations. Furthermore, it shows the value of selecting time intervals with high numbers of passengers traveling for determining where to add the extra services;
- The selected versions of the cancel stops operator also show the importance of time interval definitions for this operator.

### 5.6.2.1 Inclusion of line planning operators

In this section we assess the impact of including line planning operators in the solutions obtained with our solution method. We compare results using the timetabling operator only – the shifts operator – with results using both timetabling and line planning operators. Furthermore, we allow an increased computational time limit, and show results for 3, 5, and 12 hours of computational time. The results presented in Table \ref{table:results} show results for all planning horizons and all restart methods. The restart parameter and the operators are fixed to their previously tuned values and versions – \( \nu = 20, \texttt{Sh}1, \texttt{Serv}^+_3, \) and \texttt{Stp}^−_3 for \texttt{Restart}^{\text{New}}, and \( \nu = 50, \texttt{Sh}3, \texttt{Serv}^+_3, \) and \texttt{Stp}^−_2 for \texttt{Restart}^{\text{Prev}}.

Table \ref{table:results} shows, on the left hand side of the table, that all average and minimum gaps obtained when using line planning operators are lower than the ones on the right hand.
### Table 5.7: Results for the inclusion of line planning operators

<table>
<thead>
<tr>
<th>Restart</th>
<th>Method</th>
<th>Running time (h)</th>
<th>Avg Gap</th>
<th>Min Gap</th>
<th>Std dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>35.65</td>
<td>33.70</td>
<td>1.15</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>38.85</td>
<td>37.85</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>38.85</td>
<td>37.85</td>
<td>1.15</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>31.75</td>
<td>30.75</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>32.72</td>
<td>31.72</td>
<td>0.96</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
side of the table, when using the shift operator only. Regarding the experiments with 3 hours of computational time, the decrease in average gaps ranges from 4.58% (for \( H = 19 \) using \( \text{Restart}^{\text{Rev}} \), with average gaps 33.65% and 38.23% respectively with and without line planning operators) to 6.72% (for \( H = 19 \) using \( \text{Restart}^{\text{Prev}} \), with average gaps 32.17% and 38.89% respectively with and without line planning operators). A similar magnitude is observed for the minimum gaps obtained with and without line planning operators. However, line planning operators, especially the add service operator, make use of the available extra budget for operating costs, while the timetable operator cannot use that extra budget since it does not add driving time to the services it shifts. In general, when the available computational time increases, we observe lower gaps than with 3 hours of computational time. Increasing the computational time from 3 to 12 hours yields decreases of average gap ranging between 0.17% (using \( \text{Restart}^{\text{Rev}} \) on the instance with 5 hours of planning horizon) and 2.5% (using \( \text{Restart}^{\text{Prev}} \) on the instance with 5 hours of planning horizon).

5.6.2.2 Analysis of best solution obtained

In this section we analyze the overall best solution obtained with our heuristic approach, for the largest planning horizon of 19 hours, and with 3 hours of computational time. This solution is obtained using the restart method \( \text{Restart}^{\text{Prev}} \), restart parameter \( \nu = 50 \), and the versions \( \text{Sh}_{3}, \text{Serv}_{3}^{+}, \) and \( \text{Stp}_{2}^{-} \). The solution has a passenger costs gap of 30.91%, and runs 1659 iterations in 5 hyper-iterations. Figure 5.5 shows the convergence of the solution method for this specific solution (in blue), and shows how much of the 10% budget on operating costs increase is being used over time (in red). Additionally, the figure also shows the convergence of the best solution found without line planning operators (in black). As the timetabling operator does not use additional operating cost budget, this is not represented in the figure.

![Figure 5.5: Convergence of the solution method in terms of gap to the lower bound, and percentage of additional budget used over time](image)

Figure 5.5: Convergence of the solution method in terms of gap to the lower bound, and percentage of additional budget used over time.
Figure 5.5 shows that, when using line planning operators, the solution method finds many improving solutions in the first 10 minutes of computation, quickly reducing the passenger costs gap to the lower bound. In the first 10 minutes, we observe a gap reduction of approximately 8%. In contrast, the remaining two hours and fifty minutes of computational time provide a gap reduction of only 3% approximately. The solution without line planning operators converges less abruptly, but still manages to reduce passenger costs by approximately 4% in comparison to the lower bound, while keeping operating costs at the same level. We observe that the solution with line planning operators uses almost the full available budget from after ten minutes of computational time. This explains why the gap reduction is steep in the first ten minutes and less abrupt in the remainder of the experiment, since the operating cost budget is already close to its maximum. It also indicates that the operators defined in our solution method hardly decrease operating costs, which could free some budget for the following iterations. This could motivate introducing a new operator to free budget once the maximum operating costs are reached. It could also be done by defining a new restart method that would return to a previously visited solution that uses at most a given budget on operating costs. However, the fact that our solution method is able to reduce the gap by approximately 3% once the budget on operating costs is reached indicates that the defined operators are able to improve the solution at fixed budget.

The best solution has a passenger cost gap of 30.91% when compared to the passenger costs lower bound. However, the contribution of the different passenger costs components differs. The transfer penalty (TrP) gap is 0%, since in a tree shaped network the number of transfers cannot change with the allowed line planning modifications. Additionally, our solution method and problem definition do not allow changing whether the passengers need to transfer or not, which would in principle be part of line planning decisions, but are fixed in our problem definition. The in-vehicle time (IVT) gap is 2.63% which is justified by the lower bound assuming no stops at intermediate stations for the calculation of unconstrained travel times. In reality, not all those stops can be canceled and our estimation will be off. Initial waiting time is harder to estimate, and the IWT gap for the best solution found is 57.12%. Finally, the transfer time is the less well estimated term of passenger costs, with a gap of 376.33%. We strongly believe that these gaps are high due to the lower bound, since all transferring passengers are assumed to transfer at the minimum transfer time, which in reality is impossible to happen. A way to improve the lower bound calculations would be to include line frequencies and headway restrictions in the calculation of LB<sub>IWT</sub> and LB<sub>TrT</sub>. Nevertheless, a percentage of the gaps is also due to the solution our algorithm finds most likely not being the optimal solution.

We compare the best solutions found with both timetabling and line planning operators and with timetabling operators only in Table 5.8. The comparison is made in terms of gap to the lower bound and in terms of each element of passenger costs. Additionally, we show the percentage of budget increase used in each solution.

Table 5.8 shows that adding the line planning operators reduces the gap to the lower bound by 6.23%, at the expense of an increase of 9.97% in train minute costs, corresponding to an
Table 5.8: Comparison between the best solutions found with timetabling and line planning operators and with timetabling operators only

<table>
<thead>
<tr>
<th></th>
<th>Best Solution</th>
<th>Best Solution</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Timetabling and Line Planning</td>
<td>Timetabling</td>
<td></td>
</tr>
<tr>
<td>Gap LB (%)</td>
<td>30.91</td>
<td>37.14</td>
<td>-6.23</td>
</tr>
<tr>
<td>IVT (eur)</td>
<td>18903603.81</td>
<td>18907935.06</td>
<td>-4331.25</td>
</tr>
<tr>
<td>IWT (eur)</td>
<td>15597760.5</td>
<td>17175699.45</td>
<td>-1577939</td>
</tr>
<tr>
<td>TrT (eur)</td>
<td>3990134.61</td>
<td>4284093.24</td>
<td>-293958.63</td>
</tr>
<tr>
<td>TrP (eur)</td>
<td>930759.9</td>
<td>930759.9</td>
<td>0</td>
</tr>
<tr>
<td>Additional budget used (%)</td>
<td>9.97</td>
<td>0.00</td>
<td>9.97</td>
</tr>
</tbody>
</table>

increase of approximately 26,500 euros. Adding the line planning operators is specially effective in reducing the gaps for initial waiting times and transfer times. Based on these results, we estimate that a reduction of 1% in gap corresponds to a reduction of approximately 300,000 euros in passenger costs, calculated by summing the savings in each component and dividing by the gap to the lower bound.

In Table 5.9, we present details of the timetable of the best solution found. For each line and each direction, we show the suggested hourly frequency $f_i$, the difference in number of services in relation to the number of services using $f_i$, the total number of skipped stops throughout the day, and the minimum, maximum and average headways between services of the same line.

Table 5.9: Best solution timetable characteristics

<table>
<thead>
<tr>
<th>Line</th>
<th>Direction</th>
<th>$f_i$</th>
<th>Difference services</th>
<th>Total skipped stops</th>
<th>Minimum headway (m)</th>
<th>Maximum headway (m)</th>
<th>Average headway (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>Rotterdam Central - Groningen</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>19</td>
<td>69</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Groningen - Rotterdam Central</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>18</td>
<td>71</td>
<td>57</td>
</tr>
<tr>
<td>1700</td>
<td>Enschede - Den Haag</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>30</td>
<td>64</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>Den Haag - Enschede</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>10</td>
<td>66</td>
<td>52</td>
</tr>
<tr>
<td>2000</td>
<td>Utrecht - Den Haag</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>7</td>
<td>41</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>Den Haag - Utrecht</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>11</td>
<td>42</td>
<td>27</td>
</tr>
<tr>
<td>2800</td>
<td>Utrecht - Rotterdam Central</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>42</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>Rotterdam Central - Utrecht</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>43</td>
<td>29</td>
</tr>
<tr>
<td>11700</td>
<td>Amersfoort - Den Haag</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>21</td>
<td>81</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>Den Haag - Amersfoort</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>81</td>
<td>59</td>
</tr>
<tr>
<td>12500</td>
<td>Leeuwarden - Rotterdam Central</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>28</td>
<td>65</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>Rotterdam Central - Leeuwarden</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>80</td>
<td>54</td>
</tr>
</tbody>
</table>

Table 5.9 shows that we add services to all directions of all lines except in one case – line 11700, direction Den Haag - Amersfoort. In general, the solution obtained does not skip many stops, with a total of 11 skipped stops throughout the day and a maximum of 5 skipped stops in a single direction of a single line. With the addition of services, it is expected that the minimum headway decreases. Although the maximum headways increase the headways in the suggested frequencies by up to 21 minutes, it is curious to observe that the average headways remain very close to the headways of the suggested
frequencies. This means that, on average, passengers do not experience a large negative effect due to headway increase.

5.6.2.3 Sensitivity analysis on operating costs budget

As seen in the previous section, the percentage of additional budget on operating costs influences how much the solution method is able to reduce passenger costs. In this section we extend our analysis of the operating costs budget by considering additional values for the percentage budget increase. Table 5.10 shows results for three values of operating cost budget, \{2, 5, 10\}% , for all planning horizons and restart methods.

<table>
<thead>
<tr>
<th>H</th>
<th>Additional budget (%)</th>
<th>Restart\textsuperscript{rev}</th>
<th>Restart\textsuperscript{prev}</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg Gap</td>
<td>Min Gap</td>
<td>Std dev</td>
</tr>
<tr>
<td>5</td>
<td>47.40</td>
<td>46.43</td>
<td>0.49</td>
</tr>
<tr>
<td></td>
<td>51.56</td>
<td>50.17</td>
<td>1.02</td>
</tr>
<tr>
<td></td>
<td>52.84</td>
<td>51.32</td>
<td>0.89</td>
</tr>
<tr>
<td>12</td>
<td>36.48</td>
<td>36.31</td>
<td>0.10</td>
</tr>
<tr>
<td></td>
<td>39.06</td>
<td>38.59</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>41.07</td>
<td>41.07</td>
<td>0.00</td>
</tr>
<tr>
<td>19</td>
<td>33.65</td>
<td>31.51</td>
<td>2.15</td>
</tr>
<tr>
<td></td>
<td>36.22</td>
<td>35.89</td>
<td>0.21</td>
</tr>
<tr>
<td></td>
<td>37.20</td>
<td>36.98</td>
<td>0.11</td>
</tr>
</tbody>
</table>

The results of Table 5.10 show that as the budget on operating costs decreases the passenger cost average and minimum gaps increase. The magnitude of the increase varies according to planning horizon and restart method. We show in Figures 5.6 and 5.7 convergence and budget utilization graphs for solutions using each additional budget option, obtained using Restart\textsuperscript{Prev} , and for \( H = 19 \).

We observe similar convergence patterns for all three budget options up until their budget limit is reached. From that point, we observe that the solution method continues to be able to find timetables with lower passenger costs, but at a slower pace. The final solution for an additional budget of 5% is not much lower in gap than the final solution for additional budget of 2%, but we observe that the first is in a local minimum from around 60 minutes of computational time.
Figure 5.6: Gap to lower bound convergence for different budget values

Figure 5.7: Percentage of additional budget used over time
5.7 Conclusions and future research

We address the aperiodic public transport timetabling problem with flexible line plans, with the objective of minimizing a weighted sum of passenger costs. Passenger costs are composed of in-vehicle time, initial waiting time, transfer time, and transfer penalties. Timetable quality is calculated using shortest path computations to determine passenger costs and we compare our solutions to a lower bound on passenger costs. Feasible solutions have to respect timetabling constraints such as safety headways, and respect a budget on operating costs.

The proposed heuristic starts by calculating an initial timetable for the input line plans using the suggested frequencies. Then it iteratively applies operators that modify the initial timetable. The timetable operator considered is service shifts, where the departure and arrival times of a service are shifted forward or backward in time. Line planning operators are the addition and removal of services, and skipping stops at stations or re-adding stops at stations. To escape local minima, we study different methods to restart the search and we accept worse solutions.

We test our solution approach on a real life case study composed by a subset of the IC network in The Netherlands. Several different versions and combinations of operators are investigated. Results indicate that the inclusion of line planning operators allows obtaining solutions with gaps to the calculated lower bounds approximately 4.5-6.5% lower than when using only the timetabling operator, using an additional budget on operating costs. Furthermore we show that our solution method is able to find many improving solutions within 10 minutes of computational time. When the budget on operating costs is lower, our algorithm can still find improving solutions, but at a slower pace.

Future research could explore the influence of the budget on the quality of the timetable. For example, by developing operators focused on reducing operating budget once its limit has been reached, the current operators have room to further improve the solution. Another interesting aspect to be investigated is considering additional line planning elements, such as allowing short turning a service (and thus reducing operating costs and allowing rolling stock to be utilized earlier) or allowing changes to the definition of a line (for example, matching segments of existing lines into new lines, thus influencing the number of transfers). Our solution method can be applied to networks with cycles, but the lower bound cannot. Implementing a lower bound procedure that can be used in networks containing cycles would therefore be a reasonable extension to the approach. Furthermore, the quality of the lower bound for initial waiting time and transfer time could be improved by considering line headways, safety headways, and possibly other constraints in its computation.
Part III

Conclusion
Conclusion

Public transport planning is composed of multiple problems at different planning stages, and solving these sequentially one by one may lead to suboptimal solutions in terms of passenger service. The aim of this thesis has been to investigate the integration of timetabling and other public transport planning problems. To answer the research questions, we considered the integration of timetabling with line planning, vehicle scheduling, and passenger routing, in three distinct scientific research projects. The formulated problems show that the integration can be done in an efficient way, and the solution methods show that good solutions can be obtained within reasonable computation times, even for large real life case studies. Furthermore, we investigate the value of integration by comparing the solutions from integrated approaches with initial solutions and with solutions for timetabling only, and conclude that indeed better solutions in terms of passenger service can be obtained by integrating timetabling with selected public transport planning problems.

Chapter 3 addresses the integration of timetabling with vehicle scheduling. We formulate the problem as a mixed integer programming model that allows timetabling modifications to an initial timetable, with the purpose of reducing transfer time costs while minimizing operating costs. Our solution method is tested on a real life case study and we show that allowing modifications of dwell time at stations leads to solutions with transfer costs up to 10% lower than allowing shift modifications only, while keeping the operating costs at the same level.

In Chapter 4 we extend the integration of timetabling and vehicle scheduling to include free passenger route choice. The proposed solution method combines two state of the art models and shows that, for a real life case study, the addition of free route choice leads to daily savings of up to 40,000 DKK when compared to using static route choice, at no
additional operating costs.

Chapter 5 addresses an aperiodic train timetabling problem where line plan flexibility is allowed. Our solution method shows that line planning changes, such as addition of services and skipping stops, allow reducing passenger costs by up to 6.23% at the expense of 10% additional operating costs. However, when expressed in monetary values, the passenger cost savings outnumber the additional operating costs by a wide margin.

Outlook and future directions

The models proposed in this thesis open several possibilities for future research, not only in terms of methodology but also in terms of research topics.

Methodologically, the solution approaches could be extended in several directions. Extending the number of modifications and operators used could lead to even better solutions in terms of passenger service. Specifically for Chapter 5, short-turning services could lead to solutions where the passenger service improvements lead also to better utilization of rolling stock. Also in this paper, a way to measure operating costs more accurately and testing the solution method in a case study with high frequencies could provide interesting results. The solution methods presented in this thesis obtain good solutions in acceptable running times, but it is possible that when testing their scalability for even larger networks, with more lines and stations, the computational time needed for obtaining good solutions increases. One interesting research avenue to allow the application of these methods in larger case studies would be to reduce the size of the problem without loss of optimality. For example, partial re-routing of OD entries when recomputing passenger routes is an interesting field of research and some of its insights could be applied in these problems.

Future research could also be aimed at including crew scheduling in the framework. For example, in Chapters 3 and 4, the addition of crew scheduling constraints will restrict vehicle scheduling to only create schedules that are feasible from a crew planning point of view. From a practical point of view, the inclusion of vehicle capacities is also of major importance. The passenger routing used in the solution approaches in this thesis does not consider capacity constraints, thus it can lead to solutions that are infeasible in practice simply because there is no physical space in a vehicle for all the passengers trying to board it.

The traditional sequential approach for public transport planning ends up influencing a lot of the integrated approaches in the literature. One could instead think about different ways to decompose the whole process than the traditional approach. In fact, Lubbecke et al. (2019) present different decompositions that might be interesting to integrate line planning, timetabling, and vehicle scheduling. The usage of techniques such as machine learning might also be beneficial for integrated problems, especially if one can identify relevant structural properties that can relate them with other known problems.
Almost as important as finding good feasible solutions is knowing how good they are. Research aimed at improving existing lower bounds on passenger service is an interesting topic of research, given that when a line plan and a timetable do not exist this is indeed not a trivial task. Specifically for Chapter 5, the extension of the lower bound procedure to consider networks with cycles would be of major importance.

The models and solution methods presented in the thesis are general enough to allow their transferability to other case studies, independently of city, country, or network characteristics, provided that the needed input data is available. The exception is the lower bound model of Chapter 5 which has to be adapted for new case studies and for networks that contain cycles. Better data on passenger travel patterns, due to advances in big data and machine learning, can provide additional insights for adapting the solutions approaches, motivating the development of new search strategies.

In a dynamic world with increased focus on on-demand services, one may wonder what the future holds for public transport. We may expect that, in the next years, the public transport system will change to become more demand responsive. Timetables are still needed for the public transport operators, but passengers highly rely on mobile apps for determining their best travel routes. In part, this motivated our research to focus more on aperiodic timetabling.

In conclusion, the problems studied in the thesis, the results from the case studies, and the gaps and shortcomings identified motivate several future research avenues within integration of public transport timetabling.
Bibliography


